User's Manual

RZ/G3E Group

User's Manual: Hardware

RZ Family RZ/G Series

arm

This manual does not include NDA base information.
For more details, please refer to the User's Manual Additional Document under NDA.

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corp. without notice. Please review the latest information published by Renesas Electronics Corp. through various means, including the Renesas Electronics Corp. website (http://www.renesas.com).

Renesas Electronics
www.renesas.com

Rev.1.10 Jun, 2025

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:

www.renesas.com/contact/

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

© 2025 Renesas Electronics Corporation. All rights reserved.

Trademarks (continued)

For the “Cortex” notation, it is used as follows;
— Arm® Cortex®-A55
— Arm® Cortex®-M33
Note that in each section of the Manual, this may be noted as Cortex-A55 and Cortex-M33 respectively.

Examples of trademark or registered trademark used in the document of RZ/G3E;
CoreLink™: CoreLink is a trademark of Arm Limited.
CoreSight™: CoreSight is a trademark of Arm Limited.
Ethos™ is a trademark of Arm Limited.
Mali™: Mali is a trademark of Arm Limited.
TrustZone®: TrustZone is a trademark of Arm Limited.
Neon™: Neon is a trademark of Arm Limited.
Temper™: Temper is a trademark of Arm Limited.
OpenGL ES™: OpenGL ES is a trademark of Khronos Group, Inc.
eMMC™: eMMC is a trademark of MultiMediaCard Association
PCIe®: PCIe is a registered trademark of PCI-SIG, Inc.
PCI Express®: PCI Express is a registered trademark of PCI-SIG, Inc.
MIPI®: MIPI is a registered trademark of MIPI Alliance, Inc.
CSI-2®: CSI-2 is a registered trademark of MIPI Alliance, Inc.
DSI®: DSI is a registered trademark of MIPI Alliance, Inc.
I3C®: I3C is a registered trademark of MIPI Alliance, Inc.

Note that in each section of the Manual, trademark notation of ® and TM may be omitted.
All other trademarks and registered trademarks are the property of their respective owners.

© 2025 Renesas Electronics Corporation. All rights reserved.

Notes on the Use of Specific Technologies

1. You are required to obtain a license for the use of the technologies listed in the table below (hereinafter referred to as the “specific technology” or “specific technologies”), that are used in this LSI chip. You shall correctly obtain a license for the use of each specific technology that is in use or available for use and maintain the validity of all such licenses for as long as any of your products incorporating this LSI chip continue to be manufactured, sold, or in use. In addition, if you manufacture and supply products that incorporate this LSI chip for sale under the trademark of a third party (OEM supply), you are requested to notify such third parties that obtaining licenses for the use of the specific technologies will be required.

Technology Name
SD
H.264
H.265

2. Renesas Electronics do not grant any rights to you, and make no warranties of any kind, including but not limited to warranties of merchantability and compatibility with specific purposes and warranties of non-infringement of third party rights. In the event that you receive a claim from a third party due to your failure to correctly obtain a license to use a specific technology, you shall resolve the claim at your own responsibility and expense. Renesas Electronics shall not be liable for any damages, liabilities, or costs that may be incurred by you or any third party.

© 2025 Renesas Electronics Corporation. All rights reserved.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin
   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

How to Use This Manual

1. Objective and Target Users

This manual was written to explain the hardware functions and electrical characteristics of this LSI to the target users, i.e. those who will be using this LSI chip in the design of application systems. Target users are expected to understand the fundamentals of electrical circuits, logic circuits, and microcomputers.

This manual is structured to cover the following items: an overview of the product, descriptions of the CPU, system control functions, and peripheral functions, electrical characteristics of the device, and usage notes.

When designing an application system that includes this LSI chip, take all points to note into account. Points to note are given in their contexts, at the final parts of each of the sections, and in the section giving usage notes.

The list of revisions is a summary of major points of revision or addition for this and earlier versions. It does not cover all revised items. For details on the revised points, see the actual locations in the manual.

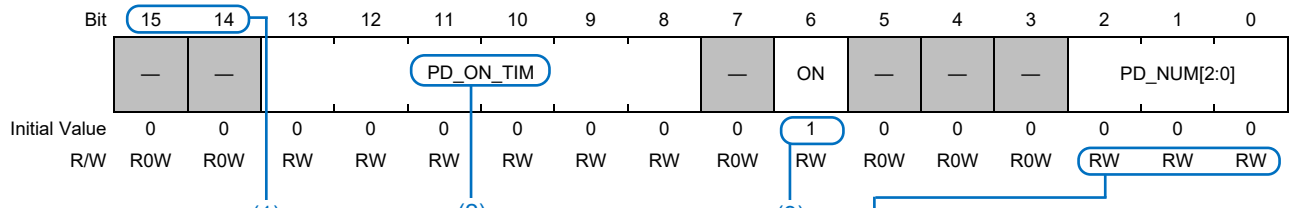
This manual is a simplified version. Please contact a Renesas Electronics sales representative for the detailed version of the manual.

---

2. Description of Registers

Each register description includes a bit chart, illustrating the arrangement of bits, and a table of bits, describing the meanings of the bit settings. The standard format and notation for use with bit charts and tables are described below.

[Bit Chart]



[Table of Bits]

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R0W	Reserved These bits are read as 0b. The write value should be 0b.
13 to 8	PD_ON_TIM	0h	RW	Set the power-on time for power domain unit.
7	—	0h	R0W	Reserved These bits are read as 0b. The write value should be always 0b.
6	ON	1h	RW	Writing to this register starts the process of turning on/off the power domain. 0b: Power off 1b: Power on
5 to 3	—	All 0	R0W	Reserved These bits are read as 0b. The write value should be always 0b.
2 to 0	PD_NUM[2:0]	0h	RW	Specify the power domain to be turned on/off. 000b: The power on/off process does not start. 001b: The processing target is PD_1. 010b: The processing target is PD_2. 100b: The processing target is PD_3. Settings other than the above are prohibited.

Note: The bit names and statements in the above figure and table are examples and have nothing to do with the main contents of this manual.

(1) Bit

Indicates the bit number or numbers.
In the case of a 64-bit register, the bits are arranged in order from 63 to 0.
In the case of a 32-bit register, the bits are arranged in order from 31 to 0.

(2) Bit name

Indicates the name of the bit or bit field.
When the number of bits must be clearly indicated for a bit field, appropriate notation is included (e.g., PD_NUM[2:0]).
Certain kinds of bits, such as those of timer counters, are not assigned bit names. In such cases, the entry under “Bit Name” is blank (e.g., PD_ON_TIM).
A reserved bit is indicated by “—”. Make sure to use the specified values when writing to such bits or fields. Otherwise, the correct operation of this LSI chip is not guaranteed.

(3) Initial value

Indicates the initial value of each bit.

0:	The initial value is 0.
1:	The initial value is 1.
x/—:	The initial value is undefined.

(4) R/W

For each bit and bit field, this entry indicates whether the bit or field is readable, writable, or both, and whether writing to and reading from the bit or field are impossible.

The notation is as follows:

RW:	The bit or field is readable and writable.
RW0:	The bit or field is readable and writable with 0. Writing 1 is invalid.
RW1:	The bit or field is readable and writable with 1. Writing 0 is invalid.
R0W:	The bit or field is readable as 0 and writable.
R1W:	The bit or field is readable as 1 and writable.
R:	The bit or field is readable. Writing is invalid.
R0W0:	The bit or field is readable as 0 and writable with 1. Writing 1 is invalid.
R0W1:	The bit or field is readable as 0 and writable with 1. Writing 0 is invalid.
R1W0:	The bit or field is readable as 1 and writable as 0. Writing 1 is invalid.
R1W1:	The bit or field is readable as and writable as 1. Writing 0 is invalid.
RCW0:	The bit or field is readable and writable with 0. Reading the bit initializes it. Writing 1 is invalid.
RCW1:	The bit or field is readable and writable with 1. Reading the bit initializes it. Writing 0 is invalid.

(5) Description

Describes the function of the bit or field and specifies the values for writing.

Note: Access to reserved addresses is prohibited.
Do not access these addresses. In case of access, correct operation of this LSI chip is not guaranteed.

Note: Access to the unavailable units is prohibited (see the product lineup). Use the related registers in the initial state.

3. Notation of Numbers and Symbols

Number notation: Binary numbers are given as XXXXb (or B'XXXX, b'XXXX), hexadecimal numbers are given as XXXXh (or H'XXXX, h'XXXX, 0xXXXX), and decimal numbers are given as XXXX.

4. List of Abbreviations

Abbreviation	Description
AES	Advanced Encryption Standard
AHB	Advanced High-performance Bus
AHB-AP	AHB Access Port
APB	Advanced Peripheral Bus
ATB	Advanced Trace Bus
AWO	Always in use (ON) power
AXI	Advanced Extensible Interface
BCD	Binary Coded Decimal
CBC	Cipher Block Chaining
CCM	Counter with CBC-MAC
CMAC	Cipher-based Message Authentication Code
CSI	Camera Serial Interface
CTR	Counter
DACK	DMA acknowledgement
DAP	Debug Access Port
DDR	Double Data Rate
DPA	Differential Power Analysis
DREQ	external DMA request
DSI	Display Serial Interface
DSP	Digital Signal Processor
ECB	Electronic Codebook
ECC	Elliptic Curve Cryptography
ECC	Error Correcting Code
ETF	Embedded Trace FIFO
ETM	Embedded Trace Macrocell
ETR	Embedded Trace Router
FIPS	Federal Information Processing Standards
FPU	Floating Point Unit
GCM	Galois/Counter Mode
GMAC	Galois Message Authentication Code
GPIO	General Purpose Input Output
I2S	Inter IC Sound (Audio format)
IRQ	Interrupt Request
JTAG	Joint Test Action Group
LPDDR	Low Power Double Date Rate
LSB	Least Significant Bit
MIPI	Mobile Industry Processor Interface
MMU	Memory Management Unit
MSB	Most Significant Bit
NIST	National Institute of Standards and Technology
NMI	Non Maskable Interrupt
PHY	PHYsical Layer
PLL	Phase Locked Loop
PWM	Pulse Width Modulation
RSA	Rivest Shamir Adleman

Abbreviation	Description
S/H	Sample and Hold
SDR	Single Data Rate
SHA	Secure Hash Algorithm
SPA	Simple Power Analysis
SSCG	Spread Spectrum Clock Generator
SWD	Serial Wire Debug
TDM	Time Division Multiplexing (Audio format)
TRNG	True Random Number Generator
UART	Universal Asynchronous Receiver/Transmitter
XTS	XEX encryption mode with tweak and ciphertext stealing

Table of Contents

SECTION 1 OVERVIEW 74

1.1 Overview 74

1.1.1 Features 74

1.1.2 Product Lineup 76

1.1.3 Functions 77

1.1.3.1 CPU 77

1.1.3.2 Boot 77

1.1.3.3 NPU 77

1.1.3.4 Graphics Unit 78

1.1.3.5 Video Processing Unit 78

1.1.3.6 Internal Memory 79

1.1.3.7 External Memory Interface 79

1.1.3.8 CPU Peripheral 80

1.1.3.9 Camera Interface 81

1.1.3.10 Display Interfaces 82

1.1.3.11 Storage and Network 83

1.1.3.12 Peripheral Module 84

1.1.3.13 Timer 86

1.1.3.14 Audio Interface 87

1.1.3.15 Analog 88

1.1.3.16 Security 88

1.1.3.17 Others 88

1.1.3.18 Power Supply Voltage 88

1.1.3.19 Temperature Range 89

1.1.3.20 Quality Level 89

1.1.3.21 Package 89

1.1.4 Block Diagram 90

1.2 Pin 93

1.2.1 Pin Assignment 93

1.2.2 External Pins and Multiplexed Functional Pins 93

1.3 Clocks 94

1.3.1 Overview 94

1.3.2 Clock Generating and Control Functions 95

1.3.3 Clock Pin Specifications 96

1.4 Reset 98

1.4.1 Overview 98

1.4.1.1 System Reset (External QRESN Pin) 98

1.4.1.2 Debug Reset (Software Reset by CoreSight) 98

1.4.1.3 Error Reset 98

1.4.1.4 Resetting of Individual Units 98

1.4.1.5 CA55 Reset 99

1.4.1.6 CM33 Reset 99

1.5 Interrupts 100

1.5.1 Overview 100

1.5.2 Overview of the ICU 101

1.5.2.1 Interrupt Control 101

1.5.2.2 Event Source Control 102
1.5.2.3 DMAC Control 102
1.5.2.4 Event Output Control 103
1.5.2.5 Error Output 103
1.5.2.6 Register Access Security Control 103
1.5.3 System Configuration Related to Interrupts 104
1.5.4 Error Processing 104
1.5.4.1 Notifying the CA55 and CM33 Cores of Errors 104
1.5.4.2 Resets Due to Errors 105
1.6 Power Management 106
1.6.1 Overview 106
1.6.2 Power Domain 106
1.6.3 Lower Power Management 107
1.7 Internal Bus 108
1.7.1 Overview 108
1.7.1.1 Features 108
1.7.1.2 LSI Internal Bus Configuration 109
1.7.2 Accessible Areas 110
1.7.3 Bus System Control 113
1.7.3.1 Security Control 113
1.7.3.1.1 Re-Setting the Security Attribute Output from Bus Masters 113
1.7.3.1.2 Determining the Security Levels of Bus Slaves 114
1.7.3.2 Address Translation 116
1.7.3.2.1 36-Bit Address Space Access 116
1.7.3.2.2 36-Bit Address Space Access for CM33 118
1.7.3.3 Bus Error Interrupt Generation 120
1.8 Address Map 121
1.8.1 Overall Address Space 121
1.8.2 CM33 Address Space 128
1.9 Boot Operation 142
1.9.1 CPU (CA55 or CM33) Normal Execution Mode 142
1.9.1.1 Operation of Booting 143
1.9.1.1.1 Boot Mode 0 (eSD) 143
1.9.1.1.2 Boot Mode 1 / Boot Mode 5 (eMMC 3.3V / 1.8V) 146
1.9.1.1.3 Boot Mode 2 / Boot Mode 6 (Serial Flash Memory 3.3V / 1.8V) 150
1.9.1.1.4 Boot Mode 3 (SCIF Download) 153
1.9.1.1.5 Boot Mode 4 (USB Download) 155
1.9.1.2 Allocation of the Loader Program 157
1.9.1.2.1 Boot Mode 0 (eSD) 157
1.9.1.2.2 Boot Mode 1 / Boot Mode 5 (eMMC 3.3V / 1.8V) 162
1.9.1.2.3 Boot Mode 2 / Boot Mode 6 (Serial Flash Memory 3.3V / 1.8V) 164
1.9.1.2.4 Boot Mode 3 (SCIF Download) 165
1.9.2 Boundary Scan Mode 166
1.9.3 Debug Mode 166

SECTION 2 PROCESSORS 167
2.1 Processors Overview 167
2.2 CPU 168
2.2.1 Overview 168
2.2.2 Application Processors Cortex-A55 (CA55) 168
2.2.2.1 Features 168
2.2.2.2 Resets 171
2.2.2.2.1 Cold Reset 172
2.2.2.2.2 Core Warm Reset 174
2.2.2.2.3 Cluster Warm Reset 178
2.2.2.2.4 Debug Reset 180
2.2.2.3 Interrupts 181
2.2.2.4 Low Power Consumption Mode 182
2.2.2.4.1 Application Sleep Mode 182
2.2.2.5 Function Reference 183
2.2.2.6 Boot Procedure 183
2.2.2.7 CoreSight Connection 184
2.2.2.8 Usage Note for Dual Core Products 185
2.2.3 System CPU Cortex-M33 (CM33) 186
2.2.3.1 Features 186
2.2.3.2 Resets 188
2.2.3.2.1 Reset Functions and Areas 188
2.2.3.2.2 Cold Reset 188
2.2.3.2.3 Warm Reset 189
2.2.3.3 Interrupts 190
2.2.3.3.1 Interrupt Pins 190
2.2.3.4 Address Space 191
2.2.3.4.1 IDAU Setting 191
2.2.3.5 Description of Functions 192
2.2.3.6 Low Power Consumption Mode 192
2.2.3.6.1 Supported Low Power Consumption Modes 192
2.2.3.6.2 Method for Executing System MPU Sleep Mode 192
2.2.3.6.3 Method for Executing Software Standby Mode 193
2.3 Ethos-U55 (NPU) 195
2.3.1 Overview 195
2.3.1.1 Features 195
2.3.1.2 Block Diagram 196
2.3.1.3 Registers 197
2.3.1.4 Usage Note 197
SECTION 3 MEMORY 198
3.1 Memory Overview 198
3.2 SRAM 199
3.2.1 Overview 199
3.2.1.1 SRAMm (m = 0) 200
3.2.1.2 SRAMm (m = 1) 201
3.2.1.3 Features 202

3.2.2 RAM Registers 203
3.2.2.1 List of Registers 203
3.2.3 RAM Register Descriptions 204
3.2.3.1 ECC Control Registers_n (SRAMm_CTL_n) (n: SRAMm Channel Number) 204
3.2.3.2 ECC Error Address Registers p_n (SRAMm_EADp_n) (n: SRAMm Channel Number) (p = 0 to 7) 206
3.2.4 Initializing the ECC Function 207
3.2.5 Initializing the Detection-Usage Areas of the On-Chip RAM 207
3.2.6 Usage Notes 208
3.2.6.1 Notes on Setting the Registers of This Module 208
3.2.6.2 8- or 16-Bit Access 208
3.2.6.2.1 8- or 16-bit reading 208
3.2.6.2.2 8- or 16-bit writing 208
3.2.6.3 Numbers of Cycles for Reading from or Writing to a RAM Area with the ECC Function Enabled 208
3.2.6.3.1 Reading 208
3.2.6.3.2 Writing 208
3.2.6.4 ECC Error Address Register Overwrite Conditions 209
3.4 LPDDR4/4X Controller (DDR) 210
3.4.1 Overview 210
3.4.1.1 Features 210
3.4.1.2 Block Diagram 211
3.4.1.3 External Pins 212
3.4.2 Usage Notes 214
3.5 TrustZone Address Space Controller (TZC) 215
3.5.1 Overview 215
3.5.1.1 Features 215
3.5.1.2 Block Diagram 216
3.5.1.3 External Pins 218
3.5.2 Register Configuration 219
3.5.3 Register Descriptions 220
3.5.4 Operation 220
3.5.5 Usage Note 220
SECTION 4 SYSTEM 221
4.1 System Overview 221
4.2 Pin Function Controller (PFC) 222
4.2.1 Overview 222
4.2.1.1 GPIO Control and Switching between Functions Multiplexed on the Pins 223
4.2.1.2 Switching and Controlling the Pins 223
4.2.1.3 OSC Mode Switching 224
4.2.1.4 Reset Latch Function 224
4.2.1.5 Digital Noise Filter Control 225
4.2.1.6 Event Link Controller 225
4.2.1.7 Functional Block Diagram 226
4.2.2 Register Description 227
4.2.2.1 List of Registers 227
4.2.2.2 Port Registers (PFC_P_mn) 235
4.2.2.3 Port Mode Registers (PFC_PM_mn) 237

4.2.2.4 Port Mode Control Registers (PFC_PMC_mn) 239
4.2.2.5 Port Function Control Registers (PFC_PFC_mn) 241
4.2.2.6 Port Pin Input Registers (PFC_PIN_mn) 243
4.2.2.7 IOLH Switching Registers (PFC_IOLH_mn) 245
4.2.2.8 Slew-Rate Switching Registers (PFC_SR_mn) 251
4.2.2.9 IEN Switching Registers (PFC_IEN_mn) 257
4.2.2.10 PU/PD Switching Registers (PFC_PUPD_mn) 259
4.2.2.11 Digital Noise Filter (FILONOFF) Registers (PFC_FILONOFF_mn) 265
4.2.2.12 Digital Noise Filter (FILNUM) Registers (PFC_FILNUM_mn) 267
4.2.2.13 Digital Noise Filter (FILCLKSEL) Registers (PFC_FILCLKSEL_mn) 269
4.2.2.14 ISEL Control Registers (PFC_ISEL_mn) 271
4.2.2.15 N-ch Open Drain Control Registers (PFC_NOD_mn) 273
4.2.2.16 Schmitt Control Registers (PFC_SMT_mn) 279
4.2.2.17 Port Group Specification Registers (PFC_ELC_PGRg) 281
4.2.2.18 Port Group Control Registers (PFC_ELC_PGCg) 282
4.2.2.19 Port Buffer Registers (PFC_ELC_PDBFg) 283
4.2.2.20 Event Connection Port Specification Registers (PFC_ELC_PELs) 284
4.2.2.21 Input Edge Detection Control Register (PFC_ELC_DPTC) 285
4.2.2.22 Port Event Control Register (PFC_ELC_ELSR2) 286
4.2.2.23 OSC Bypass Mode Switching Register (PFC_OSCBYPS) 287
4.2.2.24 Write Protect Register (PFC_PWPR) 289
4.2.2.25 IRQ Digital Noise Filter (FILONOFF) Register (PFC_FILONOFF_IRQ) 290
4.2.2.26 IRQ Digital Noise Filter (FILNUM) Register (PFC_FILNUM_IRQ) 292
4.2.2.27 IRQ Digital Noise Filter (FILCLKSEL) Register (PFC_FILCLKSEL_IRQ) 294
4.2.2.28 DMAC_REQ Digital Noise Filter (FILONOFF) Register (PFC_FILONOFF_DMAC_REQ) 296
4.2.2.29 DMAC_REQ Digital Noise Filter (FILNUM) Register (PFC_FILNUM_DMAC_REQ) 297
4.2.2.30 DMAC_REQ Digital Noise Filter (FILCLKSEL) Register (PFC_FILCLKSEL_DMAC_REQ) 298
4.2.2.31 NMI Digital Noise Filter (FILONOFF) Register (PFC_FILONOFF_NMI) 299
4.2.2.32 NMI Digital Noise Filter (FILNUM) Register (PFC_FILNUM_NMI) 300
4.2.2.33 NMI Digital Noise Filter (FILCLKSEL) Register (PFC_FILCLKSEL_NMI) 301
4.2.2.34 OEN Switching Register (PFC_OEN) 302
4.2.3 Functional Description 304
4.2.3.1 IO Block 304
4.2.3.1.1 List of IO blocks 304
4.2.3.2 IO_TOP_ISEL_CTL 305
4.2.3.3 IO_TOP_SMT_CTL 306
4.2.3.4 IO_TOP_NOD_CTL 307
4.2.3.5 RST_LATCH 308
4.2.3.6 DFIL_TOP 309
4.2.4 Operation Procedure 310
4.2.4.1 Control of Dedicated and Multiplexed Pins 310
4.2.4.2 Multiplexed Pins (in Port Mode): GPIO Control and Control of the Pins 310
4.2.4.3 Multiplexed Pins (in Control Mode): Switching between Functions Multiplexed on the Pins and Control of the Pins 310
4.2.4.4 OSC Mode Switching 313
4.2.4.5 Reset Latch Function 313
4.2.4.6 Digital Noise Filter Control 313
4.2.4.7 Event Link Controller — Port Event Control (PFC_ELC_GPIO) 313
4.2.4.8 Switching Power in the OTHERS Area Power Domain from ON to OFF 319

4.3 System Controller (SYS) 320
4.3.1 Functional Overview 320
4.3.1.1 Functions 320
4.3.2 Connection Schematic Diagram 321
4.3.3 Registers 322
4.3.3.1 List of Registers 322
4.3.3.2 Register Description 327
4.3.3.2.1 Master Access Control Register 0 (SYS_MSTACCCTL0) 327
4.3.3.2.2 Master Access Control Register 1 (SYS_MSTACCCTL1) 330
4.3.3.2.3 Master Access Control Register 3 (SYS_MSTACCCTL3) 331
4.3.3.2.4 Master Access Control Register 4 (SYS_MSTACCCTL4) 333
4.3.3.2.5 Master Access Control Register 5 (SYS_MSTACCCTL5) 334
4.3.3.2.6 Master Access Control Register 6 (SYS_MSTACCCTL6) 337
4.3.3.2.7 Master Access Control Register 7 (SYS_MSTACCCTL7) 339
4.3.3.2.8 Master Access Control Register 8 (SYS_MSTACCCTL8) 340
4.3.3.2.9 Master Access Control Register 12 (SYS_MSTACCCTL12) 342
4.3.3.2.10 Master Access Control Register 13 (SYS_MSTACCCTL13) 343
4.3.3.2.11 Master Access Control Register 14 (SYS_MSTACCCTL14) 345
4.3.3.2.12 Master Access Control Register 15 (SYS_MSTACCCTL15) 346
4.3.3.2.13 Master Access Control Register 19 (SYS_MSTACCCTL19) 347
4.3.3.2.14 Master Access Control Register 21 (SYS_MSTACCCTL21) 348
4.3.3.2.15 Slave Access Control Register 0 (SYS_SLVACCCTL0) 349
4.3.3.2.16 Slave Access Control Register 1 (SYS_SLVACCCTL1) 350
4.3.3.2.17 Slave Access Control Register 2 (SYS_SLVACCCTL2) 351
4.3.3.2.18 Slave Access Control Register 3 (SYS_SLVACCCTL3) 352
4.3.3.2.19 Slave Access Control Register 4 (SYS_SLVACCCTL4) 353
4.3.3.2.20 Slave Access Control Register 5 (SYS_SLVACCCTL5) 354
4.3.3.2.21 Slave Access Control Register 6 (SYS_SLVACCCTL6) 355
4.3.3.2.22 Slave Access Control Register 7 (SYS_SLVACCCTL7) 356
4.3.3.2.23 Slave Access Control Register 8 (SYS_SLVACCCTL8) 357
4.3.3.2.24 Slave Access Control Register 9 (SYS_SLVACCCTL9) 358
4.3.3.2.25 Slave Access Control Register 10 (SYS_SLVACCCTL10) 359
4.3.3.2.26 Slave Access Control Register 11 (SYS_SLVACCCTL11) 360
4.3.3.2.27 Slave Access Control Register 12 (SYS_SLVACCCTL12) 361
4.3.3.2.28 Slave Access Control Register 13 (SYS_SLVACCCTL13) 362
4.3.3.2.29 Slave Access Control Register 14 (SYS_SLVACCCTL14) 363
4.3.3.2.30 Slave Access Control Register 16 (SYS_SLVACCCTL16) 364
4.3.3.2.31 Slave Access Control Register 19 (SYS_SLVACCCTL19) 365
4.3.3.2.32 Slave Access Control Register 20 (SYS_SLVACCCTL20) 366
4.3.3.2.33 Slave Access Control Register 21 (SYS_SLVACCCTL21) 367
4.3.3.2.34 Slave Access Control Register 32 (SYS_SLVACCCTL32) 368
4.3.3.2.35 Slave Access Control Register 35 (SYS_SLVACCCTL35) 369
4.3.3.2.36 Slave Access Control Register 36 (SYS_SLVACCCTL36) 370

4.3.3.2.37 Slave Access Control Register 37 (SYS_SLVACCCTL37) 371
4.3.3.2.38 Slave Access Control Register 38 (SYS_SLVACCCTL38) 372
4.3.3.2.39 Slave Access Control Register 39 (SYS_SLVACCCTL39) 373
4.3.3.2.40 Slave Access Control Register 40 (SYS_SLVACCCTL40) 374
4.3.3.2.41 Slave Access Control Register 41 (SYS_SLVACCCTL41) 375
4.3.3.2.42 Slave Access Control Register 43 (SYS_SLVACCCTL43) 376
4.3.3.2.43 Slave Access Control Register 44 (SYS_SLVACCCTL44) 377
4.3.3.2.44 Slave Access Control Register 47 (SYS_SLVACCCTL47) 378
4.3.3.2.45 Slave Access Control Register 48 (SYS_SLVACCCTL48) 379
4.3.3.2.46 Slave Access Control Register 49 (SYS_SLVACCCTL49) 380
4.3.3.2.47 Slave Access Control Register 50 (SYS_SLVACCCTL50) 381
4.3.3.2.48 Slave Access Control Register 51 (SYS_SLVACCCTL51) 382
4.3.3.2.49 Slave Access Control Register 52 (SYS_SLVACCCTL52) 383
4.3.3.2.50 Slave Access Control Register 53 (SYS_SLVACCCTL53) 384
4.3.3.2.51 Slave Access Control Register 55 (SYS_SLVACCCTL55) 385
4.3.3.2.52 Slave Access Control Register 56 (SYS_SLVACCCTL56) 386
4.3.3.2.53 Slave Access Control Register 57 (SYS_SLVACCCTL57) 387
4.3.3.2.54 Slave Access Control Register 64 (SYS_SLVACCCTL64) 388
4.3.3.2.55 Slave Access Control Register 65 (SYS_SLVACCCTL65) 389
4.3.3.2.56 Slave Access Control Register 66 (SYS_SLVACCCTL66) 390
4.3.3.2.57 Slave Access Control Register 67 (SYS_SLVACCCTL67) 391
4.3.3.2.58 Slave Access Control Register 68 (SYS_SLVACCCTL68) 392
4.3.3.2.59 Slave Access Control Register 80 (SYS_SLVACCCTL80) 393
4.3.3.2.60 Slave Access Control Register 81 (SYS_SLVACCCTL81) 394
4.3.3.2.61 Slave Access Control Register 87 (SYS_SLVACCCTL87) 395
4.3.3.2.62 Slave Access Control Register 89 (SYS_SLVACCCTL89) 396
4.3.3.2.63 Slave Access Control Register 91 (SYS_SLVACCCTL91) 397
4.3.3.2.64 Slave Access Control Register 92 (SYS_SLVACCCTL92) 398
4.3.3.2.65 Slave Access Control Register 93 (SYS_SLVACCCTL93) 399
4.3.3.2.66 LSI Mode Signal Register (SYS_LSI_MODE)..... 400
4.3.3.2.67 LSI Device ID Register (SYS_LSI_DEVID)..... 402
4.3.3.2.68 LSI Product Register (SYS_LSI_PRR) 403
4.3.3.2.69 TSU1 Trimming Register 0 (SYS_LSI_OTPTSU1TRMVAL0)..... 404
4.3.3.2.70 TSU1 Trimming Register 1 (SYS_LSI_OTPTSU1TRMVAL1)..... 405
4.3.3.2.71 Address Offset Register n (SYS_AOFn) (n = 0 to 2, 5, 6, 8, 9, 16 to 22, 24, 25) 406
4.3.3.2.72 Address Offset Register n (SYS_AOFn) (n = 32) 408
4.3.3.2.73 Address Offset Register n (SYS_AOFn) (n = 33, 42, 43, 48 to 50) 409
4.3.3.2.74 CA55 Peripheral Port Region Start Address Low Register (SYS_ACPU_CFG_SMPL) 411
4.3.3.2.75 CA55 Peripheral Port Region Start Address High Register (SYS_ACPU_CFG_SMPH)..... 412

4.3.3.2.76 CA55 Peripheral Port Region End Address Low Register (SYS_ACPU_CFG_EMPL) 413
4.3.3.2.77 CA55 Peripheral Port Region End Address High Register (SYS_ACPU_CFG_EMPH) 414
4.3.3.2.78 CA55 Core 0 Reset Vector Address Low Configuration Register (SYS_ACPU_CFG_RVAL0) 415
4.3.3.2.79 CA55 Core 0 Reset Vector Address High Configuration Register (SYS_ACPU_CFG_RVAH0) 416
4.3.3.2.80 CA55 Core 1 Reset Vector Address Low Configuration Register (SYS_ACPU_CFG_RVAL1) 417
4.3.3.2.81 CA55 Core 1 Reset Vector Address High Configuration Register (SYS_ACPU_CFG_RVAH1) 418
4.3.3.2.82 CA55 Core 2 Reset Vector Address Low Configuration Register (SYS_ACPU_CFG_RVAL2) 419
4.3.3.2.83 CA55 Core 2 Reset Vector Address High Configuration Register (SYS_ACPU_CFG_RVAH2) 420
4.3.3.2.84 CA55 Core 3 Reset Vector Address Low Configuration Register (SYS_ACPU_CFG_RVAL3) 421
4.3.3.2.85 CA55 Core 3 Reset Vector Address High Configuration Register (SYS_ACPU_CFG_RVAH3) 422
4.3.3.2.86 CM33 Config Register 2 (SYS_MCPU_CFG2) 423
4.3.3.2.87 CM33 Config Register 3 (SYS_MCPU_CFG3) 424
4.3.3.2.88 CM33 Config Register 4 (SYS_MCPU_CFG4) 425
4.3.3.2.89 CM33 Lock Register (SYS_MCPU_CFG5) 426
4.3.3.2.90 xSPI CS0 Start Address Register (SYS_SPI_STAADDCS0) 427
4.3.3.2.91 xSPI CS0 End Address Register (SYS_SPI_ENDADDCS0) 428
4.3.3.2.92 xSPI CS1 Start Address Register (SYS_SPI_STAADDCS1) 429
4.3.3.2.93 xSPI CS1 End Address Register (SYS_SPI_ENDADDCS1) 430
4.3.3.2.94 SRAM0 ECC Setting Register (SYS_SRAM0_ECC) 431
4.3.3.2.95 SRAM0 Access Control Register (SYS_SRAM0_EN) 432
4.3.3.2.96 SRAM1 ECC Setting Register (SYS_SRAM1_ECC) 433
4.3.3.2.97 SRAM1 Access Control Register (SYS_SRAM1_EN) 434
4.3.3.2.98 WDT0 Control Register (SYS_WDT0_CTRL) 435
4.3.3.2.99 WDT1 Control Register (SYS_WDT1_CTRL) 436
4.3.3.2.100 WDT2 Control Register (SYS_WDT2_CTRL) 437
4.3.3.2.101 WDT3 Control Register (SYS_WDT3_CTRL) 438
4.3.3.2.102 VSP Control Register (SYS_VSP_CLK) 439
4.3.3.2.103 GBETH0 Config Register (SYS_GBETH0_CFG) 440
4.3.3.2.104 GBETH1 Config Register (SYS_GBETH1_CFG) 441
4.3.3.2.105 PCIE INTX Ch 0 Register (SYS_PCIE_INTX_CH0) 442
4.3.3.2.106 PCIE MSI Ch 0 Register 1 (SYS_PCIE_MSI1_CH0) 443
4.3.3.2.107 PCIE MSI Ch 0 Register 2 (SYS_PCIE_MSI2_CH0) 444
4.3.3.2.108 PCIE MSI Ch 0 Register 3 (SYS_PCIE_MSI3_CH0) 445

4.3.3.2.109 PCIE MSI Ch 0 Register 4 (SYS_PCIE_MSI4_CH0) ... 446
4.3.3.2.110 PCIE MSI Ch 0 Register 5 (SYS_PCIE_MSI5_CH0) ... 447
4.3.3.2.111 PCIE PME Ch 0 Register (SYS_PCIE_PME_CH0) ... 448
4.3.3.2.112 PCIE ACK Ch 0 Register (SYS_PCIE_ACK_CH0) ... 449
4.3.3.2.113 PCIE Misc Function Ch 0 Register (SYS_PCIE_MISC_CH0) ... 450
4.3.3.2.114 PCIE MODE Ch 0 Register (SYS_PCIE_MODE_CH0) ... 451
4.3.3.2.115 ADC Config Register (SYS_ADC_CFG) ... 452
4.3.3.2.116 General Register 0 (SYS_GPREG_0) ... 453
4.3.3.2.117 General Register 1 (SYS_GPREG_1) ... 454
4.3.3.2.118 General Register 2 (SYS_GPREG_2) ... 455
4.3.3.2.119 General Register 3 (SYS_GPREG_3) ... 456
4.3.4 Functional Description ... 457
4.3.4.1 Security Level Setting ... 457
4.3.4.2 Stopping of Counting by WDTm ... 460
4.4 Clock Pulse Generator (CPG) ... 461
4.4.1 Overview ... 461
4.4.1.1 Functions for the Generation and Control of Clock Signals ... 462
4.4.1.2 Function for the Generation and Control of Resets ... 463
4.4.1.3 Function for the Control of Booting ... 463
4.4.1.4 Function for the Control over the Power Supply Domains ... 463
4.4.1.5 Function for the Control over Low Power Consumption ... 463
4.4.1.6 Functional Block Diagram ... 464
4.4.2 List of Clock Signals ... 465
4.4.3 List of Reset Signals ... 467
4.4.4 Registers ... 473
4.4.4.1 pllname Standby Control Registers (CPG_pllname_STBY) ... 479
4.4.4.2 pllname Output Clock Setting Registers 1 (CPG_pllname_CLK1) ... 481
4.4.4.3 pllname Output Clock Setting Registers 2 (CPG_pllname_CLK2) ... 483
4.4.4.4 pllname Monitor Registers (CPG_pllname_MON) ... 485
4.4.4.5 Static Mux Control Registers (CPG_SSELm) (m = 0 to 3) ... 486
4.4.4.6 Dynamic Gear Control Registers (CPG_CDDIVm) (m = 0 to 5) ... 488
4.4.4.7 Static Gear Control Registers (CPG_CSDIVm) (m = 0, 1) ... 491
4.4.4.8 CGC Control Registers (CPG_CLKON_m) (m = 0 to 27) ... 493
4.4.4.9 Dynamic Gear/Mux Status Monitor Register (CPG_CLKSTATUS0) ... 501
4.4.4.10 CGC Monitor Registers (CPG_CLKMON_m) (m = 0 to 11) ... 503
4.4.4.11 Reset Control Registers (CPG_RST_m) (m = 0 to 18) ... 509
4.4.4.12 Reset Monitor Registers (CPG_RSTMON_m) (m = 0 to 9) ... 515
4.4.4.13 Error Reset Select Register 1 (CPG_ERRORRST_SEL1) ... 521
4.4.4.14 Error Reset Select Registers (CPG_ERRORRST_SELm) (m = 2 to 8) ... 522
4.4.4.15 Error Reset Registers (CPG_ERROR_RSTm) (m = 2 to 8) ... 525
4.4.4.16 Low-Power-Consumption Sequence Control Register 1 (CPG_LP_CTL1) ... 528
4.4.4.17 Low-Power-Consumption Sequence Control Register 2 (CPG_LP_CTL2) ... 529
4.4.4.18 GE3D Low-Power-Consumption Sequence Control Register (CPG_LP_GE3D_CTL) ... 530
4.4.4.19 CM33 Control Register (CPG_CM33_CTL) ... 533
4.4.4.20 Low-Power Sequence Cortex-M33 Control Register 0 (CPG_LP_CM33_CTL0) ... 534
4.4.4.21 Low-Power Sequence Cortex-M33 Control Register 1 (CPG_LP_CM33_CTL1) ... 535
4.4.4.22 Cortex-A55 Clock Control Register 1 (CPG_LP_CA55_CTL1) ... 537
4.4.4.23 Cortex-A55 Clock Control Register 2 (CPG_LP_CA55_CTL2) ... 538

4.4.4.24 Cortex-A55 Clock Control Register 3 (CPG_LP_CA55_CTL3)....539
4.4.4.25 Cortex-A55 Clock Control Register 6 (CPG_LP_CA55_CTL6)....540
4.4.4.26 Cortex-A55 Clock Control Register 7 (CPG_LP_CA55_CTL7)....541
4.4.4.27 Low-Power Sequence Control Register (CPG_LP_PMU_CTL1)....542
4.4.4.28 SRAM Standby Control Registers (CPG_LP_SRAM_STBY_CTLm) (m = 1 to 3)..543
4.4.4.29 CST Control Register 2 (CPG_LP_CST_CTL2)....547
4.4.4.30 OTP Handshaking Monitor Register (CPG_OTP_HANDSHAKE_MON)....548
4.4.4.31 Reset Control for the PD_OTHERS Domain (CPG_OTHERS_INI)....549
4.4.4.32 NPU Low-Power-Consumption Sequence Control Register 1 (CPG_LP_NPU_CTL1)....550
4.4.4.33 NPU Low-Power-Consumption Sequence Control Register 2 (CPG_LP_NPU_CTL2)....552
4.4.4.34 VSPI Low-Power-Consumption Sequence Control Register 1 (CPG_LP_VSPI_CTL1)....553
4.4.4.35 VSPI Low-Power-Consumption Sequence Control Register 2 (CPG_LP_VSPI_CTL2)....555
4.4.4.36 FDP1 Low-Power-Consumption Sequence Control Register (CPG_LP_FDP1_CTL)....556
4.4.4.37 MSTOP Registers (CPG_BUS_m_MSTOP) (m = 1 to 13)....558
4.4.5 Functions for the Generation and Control of Clock Signals....562
4.4.5.1 Block Diagram of the Clock System....562
4.4.5.2 Control of Turning Unit Clock Signals On and Off....562
4.4.5.3 Functions for Controlling, Setting, and Monitoring PLLs....562
4.4.5.3.1 PLL operating mode control....563
4.4.5.3.2 Setting output clock signals....564
4.4.5.3.3 Monitoring the PLL states....564
4.4.5.3.4 Timing setting....565
4.4.5.3.5 PLL settings....567
4.4.5.4 Control Over Frequencies for the CA55....570
4.4.5.4.1 Setting PLLCA55 by BOOTPLLCA pins....571
4.4.5.5 Clock Dividers and Selectors....572
4.4.5.5.1 Dynamic switching variable divider....573
4.4.5.5.2 Static switching selector....573
4.4.5.5.3 Static switching variable divider....574
4.4.5.6 Temporarily Halting Clock Signals After Reset Signals are De-asserted....575
4.4.6 Functions for the Generation and Control of Resets....576
4.4.6.1 Block Diagram of the Reset System....576
4.4.6.2 Reset Application List....577
4.4.6.3 System Reset (External Pin)....578
4.4.6.4 Switching Unit Reset On and Off....578
4.4.6.5 Error Reset....579
4.4.6.5.1 Error reset circuit....579
4.4.6.5.2 WDTUDFCM pin....580
4.4.6.5.3 WDTUDFCA pin....581
4.4.6.5.4 Error reset registers (CPG_ERROR_RSTm)....581
4.4.6.6 Debug Reset (Software Reset by CoreSight)....582
4.4.6.7 CA55 Reset....583
4.4.6.7.1 CA55 system cold reset (normal mode)....584
4.4.6.7.2 CA55 system cold reset (debug mode)....584

4.4.6.7.3 CA55 error cold reset 584
4.4.6.7.4 CA55 software cold reset 584
4.4.6.7.5 CA55 software warm reset 584
4.4.6.7.6 CA55 reset timing constraints 584
4.4.6.8 CM33 Reset 585
4.4.6.8.1 CM33 system cold reset 585
4.4.6.8.2 CM33 error cold reset 585
4.4.6.8.3 CM33 software cold reset 586
4.4.6.8.4 CM33 warm reset 586
4.4.6.8.5 CM33 reset timing constraints 586
4.4.7 Control of Booting 587
4.4.7.1 SRAM Rescue Circuit Control 588
4.4.7.2 CM33 Boot (Normal Mode) 588
4.4.7.3 CM33 Boot (Debug Mode) 589
4.4.7.4 CA55 Boot (Normal Mode) 590
4.4.7.5 CA55 Boot (Debug Mode) 591
4.4.8 Control over Lower Power Consumption 592
4.4.8.1 State Transition for Each Mode 593
4.4.8.2 Power Mode Control 594
4.4.8.3 Low-Frequency Mode 594
4.4.8.3.1 Overview 594
4.4.8.3.2 Procedure for transition to the mode 594
4.4.8.3.3 Procedure for return from the mode 594
4.4.8.4 Module Standby Mode 595
4.4.8.4.1 Overview 595
4.4.8.4.2 Procedure for transition to the mode 595
4.4.8.4.3 Procedure for return from the mode 595
4.4.8.4.4 Points to note 596
4.4.8.5 SRAM Power-Saving Mode 597
4.4.8.5.1 Overview 597
4.4.8.5.2 Procedure for transition to the mode 598
4.4.8.5.3 Procedure for return from the mode 598
4.4.8.6 CM33 Sleep Mode 599
4.4.8.6.1 Overview 599
4.4.8.6.2 Procedure for transition to the mode 599
4.4.8.6.3 Procedure for return from the mode 599
4.4.8.7 CA55 Sleep Mode 600
4.4.8.7.1 Overview 600
4.4.8.7.2 Procedure for transition to the mode 600
4.4.8.7.3 Procedure for return from the mode 600
4.4.8.8 Software Standby Mode (only for the CM33 Cold Boot) 601
4.4.8.8.1 Overview 601
4.4.8.8.2 Procedure for transition to the mode 601
4.4.8.8.3 Procedure for return from the mode 601
4.4.9 Operation 603
4.4.9.1 Procedure for Switching Unit Clock Signals On and Off 603

4.4.9.2 Procedure for Switching Unit Resets On and Off 604
4.4.9.3 Procedure for Starting up Units 604
4.4.9.4 Procedure for Setting PLLs 605
4.4.9.4.1 Procedure for setting the PLL normal mode (changing the output frequency and SSC mode) 605
4.4.9.4.2 Procedure for setting the PLL for standby mode 606
4.4.9.5 Procedure for Setting Frequency Division Ratios of Dynamic Switching Dividers 607
4.4.9.6 Procedure for Setting Static Switching Dividers and Selectors 608
4.4.9.7 Boot Sequence (Normal Mode) 609
4.4.9.8 Boot Sequence (Debug Mode) 611
4.4.9.9 Procedure for Writing to Registers 612
4.4.9.10 Procedure for CM33 Warm Reset 613
4.4.10 Restrictions 614
4.4.10.1 CM33 Systick Timer 614
4.4.10.2 Restrictions on Using the Individual PLLs 614
4.4.10.3 Low Power Mode Transition Conditions 615
4.5 Power Management Unit (PMU) 616
4.5.1 Overview 616
4.5.2 Pin Description 616
4.5.3 Power Domains 617
4.5.3.1 Power Domain Control 620
4.5.3.1.1 Power mode (AWO/AWO_DDR_RETENTION → ALL_ON) 621
4.5.3.1.2 Power mode (ALL_ON → AWO/AWO_DDR_RETENTION) 622
4.5.3.1.3 Power mode (ALL_ON → DDR_RETENTION) 623
4.5.3.1.4 Power mode (DDR_RETENTION → ALL_ON) 623
4.5.3.1.5 PD_CA55 area voltage change 624
4.5.4 Usage Note 625
4.5.4.1 Initialization of PD_OTHERS 625
4.6 Interrupt Controller 626
4.6.1 Interrupt Control Unit (ICU) 626
4.6.1.1 Functional Block Diagram 627
4.6.1.2 Registers 628
4.6.1.2.1 List of Registers 628
4.6.1.2.2 Register Description 632
4.6.1.3 Detailed Specifications 690
4.6.1.3.1 Control over input events 690
4.6.1.3.2 CPU interrupt output 697
4.6.1.3.3 Control over the event link controller (ELC) 705
4.6.1.3.4 Control over the DMAC 707
4.6.1.3.5 Other control 711
4.6.1.4 Restrictions 714
4.6.1.5 Event List 715
4.6.1.5.1 List of input events 715
4.6.1.5.2 List of output events 771
4.6.1.5.3 List of bus error events 773
4.6.1.5.4 List of SRAM error events 775
4.6.1.5.5 List of the DMA ACK signals 776

4.6.2 Generic Interrupt Controller (GIC) 779
4.6.2.1 Features 779
4.6.2.2 GIC-600 and NVIC Register Configuration 779
4.6.2.3 GIC-600 and NVIC Register Descriptions 780
4.6.3 Message Handling Unit (MHU) 781
4.6.3.1 Features 781
4.6.3.2 Interrupt 783
4.6.3.3 Registers 788
4.6.3.3.1 List of Registers 789
4.6.3.3.2 Register Description 790
4.6.3.4 Function Description 805
4.6.3.4.1 Inter-CPU communication 805
4.6.3.4.2 Software interrupt 807
4.6.3.5 Operation Sequence 808
4.6.3.5.1 Message transmission sequence 808
4.6.3.5.2 Response transmission sequence 809
4.6.3.5.3 Software interrupt sequence 810
4.7 DMA Controller (DMAC) 811
4.7.1 Functional Overview 811
4.7.1.1 Features 811
4.7.1.2 DMA Transfer Specifications 812
4.7.1.3 Restrictions 812
4.7.2 Connection Configuration 813
4.7.3 Pin Functions 814
4.7.3.1 List of External Pins 814
4.7.3.2 List of Internal Pins 814
4.7.3.2.1 DMA Interfaces 815
4.7.3.2.2 Interrupt Interfaces 817
4.7.4 Address Space 818
4.7.5 Register Descriptions 819
4.7.5.1 List of Registers 819
4.7.5.2 Register Descriptions 820
4.7.5.2.1 Next0 Source Address Register n (DMACm_N0SA_n) (m = 0 to 4, n = 0 to 15) 822
4.7.5.2.2 Next0 Destination Address Register n (DMACm_N0DA_n) (m = 0 to 4, n = 0 to 15) 823
4.7.5.2.3 Next0 Transaction Byte Register n (DMACm_N0TB_n) (m = 0 to 4, n = 0 to 15) 824
4.7.5.2.4 Next1 Source Address Register n (DMACm_N1SA_n) (m = 0 to 4, n = 0 to 15) 825
4.7.5.2.5 Next1 Destination Address Register n (DMACm_N1DA_n) (m = 0 to 4, n = 0 to 15) 826
4.7.5.2.6 Next1 Transaction Byte Register n (DMACm_N1TB_n) (m = 0 to 4, n = 0 to 15) 827
4.7.5.2.7 Current Source Address Register n (DMACm_CRSA_n) (m = 0 to 4, n = 0 to 15) 828

4.7.5.2.8 Current Destination Address Register n (DMACm_CRDA_n) (m = 0 to 4, n = 0 to 15) ... 829
4.7.5.2.9 Current Transaction Byte Register n (DMACm_CRTB_n) (m = 0 to 4, n = 0 to 15) ... 830
4.7.5.2.10 Channel Status Register n (DMACm_CHSTAT_n) (m = 0 to 4, n = 0 to 15) ... 831
4.7.5.2.11 Channel Control Register n (DMACm_CHCTRL_n) (m = 0 to 4, n = 0 to 15) ... 834
4.7.5.2.12 Channel Configuration Register n (DMACm_CHCFG_n) (m = 0 to 4, n = 0 to 15) ... 836
4.7.5.2.13 Channel Interval Register n (DMACm_CHITVL_n) (m = 0 to 4, n = 0 to 15) ... 839
4.7.5.2.14 Channel Extension Register n (DMACm_CHEXT_n) (m = 0 to 4, n = 0 to 15) ... 840
4.7.5.2.15 Next Link Address Register n (DMACm_NXLA_n) (m = 0 to 4, n = 0 to 15)841
4.7.5.2.16 Current Link Address Register n (DMACm_CRLA_n) (m = 0 to 4, n = 0 to 15) ... 842
4.7.5.2.17 DMA Control Register (DMACm_DCTRL) (m = 0 to 4) ... 843
4.7.5.2.18 DMA Status EN Register (DMACm_DST_EN) (m = 0 to 4) ... 844
4.7.5.2.19 DMA Status ER Register (DMACm_DST_ER) (m = 0 to 4) ... 845
4.7.5.2.20 DMA Status END Register (DMACm_DST_END) (m = 0 to 4) ... 846
4.7.5.2.21 DMA Status TC Register (DMACm_DST_TC) (m = 0 to 4) ... 847
4.7.5.2.22 DMA Status SUS Register (DMACm_DST_SUS) (m = 0 to 4) ... 848
4.7.6 Functional Description ... 849
4.7.6.1 DMA Mode ... 849
4.7.6.1.1 Mode setting ... 849
4.7.6.1.2 Register mode ... 850
4.7.6.1.3 Link mode ... 857
4.7.6.2 DMA Transfer ... 866
4.7.6.2.1 Transfer mode ... 866
4.7.6.2.2 Control over the priority of DMAC channels ... 868
4.7.6.2.3 DMA transfer request ... 870
4.7.6.2.4 DMA acknowledge output ... 888
4.7.6.2.5 DMA transfer completed interrupt ... 893
4.7.6.2.6 DMA terminal count signal output function ... 894
4.7.6.2.7 DMA error interrupt (DMAERR) ... 895
4.7.6.2.8 Interval counting ... 896
4.7.6.2.9 Differences in operation with the transfer size ... 897
4.7.6.2.10 Transfer state ... 900
4.7.6.3 AXI Address Conversion ... 906
4.7.6.3.1 Prohibited items ... 907
4.7.7 Operating Procedure ... 908
4.7.7.1 Example DMAC Settings ... 908
4.7.7.1.1 Example setting 1 (hardware request in register mode) ... 909

4.7.7.1.2 Example setting 2 (software request in register mode) ... 911
4.7.7.1.3 Example setting 3 (continuous execution in register mode) ... 913
4.7.7.1.4 Example setting 4 (link mode) ... 915
4.7.7.1.5 Setting for next register continuous execution ... 918
4.7.8 Points for Caution ... 921
4.7.8.1 DMA Transfer Requests by External Devices ... 921
4.7.8.2 Problem of the Interrupt Signal Overtaking DMA Transfer ... 921
4.7.8.2.1 Overview ... 921
4.7.8.2.2 Workaround ... 922
4.8 Renesas Secure IP (RSIP-E50B) ... 923
4.8.1 Overview ... 923
4.8.2 Operation ... 926
4.8.2.1 Encryption Engine ... 926
4.8.2.2 Encryption and Decryption ... 927
4.8.3 Usage Notes ... 928
4.8.3.1 SPA/DPA Protections ... 928
4.8.3.2 Renesas Secure IP(RSIP-E50B) Driver ... 928
4.9 Debug Interface ... 929
4.9.1 Overview ... 929
4.9.1.1 Features ... 929
4.9.1.2 Block Diagram ... 930
4.9.1.3 External Pins ... 932
4.9.2 CST Registers ... 934
4.9.2.1 List of Registers ... 936
4.9.2.2 CST Register Descriptions ... 937
4.9.2.2.1 ID Register (CST_BSID) ... 937
4.9.2.2.2 Other Registers ... 937
4.9.3 Debug Connection ... 938
4.9.3.1 Debug Connection Mode ... 938
4.9.4 Debug Access Range ... 938
4.9.5 Operation ... 939
4.9.5.1 Debug Related Reset ... 939
4.9.5.2 DAP ... 939
4.9.5.3 Trace ... 940
4.9.5.3.1 Data flow ... 940
4.9.5.3.2 Timestamp ... 940
4.9.5.4 Cross Trigger ... 941
4.9.5.4.1 CoreSight cross trigger connection ... 941
4.9.5.4.2 CA55 cross trigger connection ... 942
4.9.5.4.3 CM33 cross trigger connection ... 945
4.9.5.5 WDT Counter Stop Control ... 945
4.9.5.6 SYC Counter Stop Control ... 945
SECTION 5 TIMER ... 946
5.1 Timer Overview ... 946
5.2 System Counter (SYC) ... 947
5.2.1 Overview ... 947

5.2.1.1 Features....947
5.2.1.2 Block Diagram of SYC....948
5.2.2 Registers....949
5.2.3 Operation....950
5.2.3.1 System Bus Access Control....950
5.2.3.2 Halt on Debug Function....950
5.3 Realtime Clock (RTC)....951
5.3.1 Overview....951
5.3.2 Registers....954
5.3.2.1 List of Registers....955
5.3.2.2 Register Description....956
5.3.2.2.1 64-Hz Counter (RTC_R64CNT)....956
5.3.2.2.2 Second Counter (RTC_RSECCNT)/Binary Counter 0 (RTC_BCNT0)....957
5.3.2.2.3 Minute Counter (RTC_RMINCNT)/Binary Counter 1 (RTC_BCNT1)....959
5.3.2.2.4 Hour Counter (RTC_RHRCNT)/Binary Counter 2 (RTC_BCNT2)....961
5.3.2.2.5 Day-of-Week Counter (RTC_RWKCNT)/Binary Counter 3 (RTC_BCNT3) 963
5.3.2.2.6 Date Counter (RTC_RDAYCNT)....965
5.3.2.2.7 Month Counter (RTC_RMONCNT)....966
5.3.2.2.8 Year Counter (RTC_RYRCNT)....967
5.3.2.2.9 Second Alarm Register (RTC_RSECAR)/Binary Counter 0 Alarm Register (RTC_BCNT0AR)....968
5.3.2.2.10 Minute Alarm Register (RTC_RMINAR)/Binary Counter 1 Alarm Register (RTC_BCNT1AR)....970
5.3.2.2.11 Hour Alarm Register (RTC_RHRAR)/Binary Counter 2 Alarm Register (RTC_BCNT2AR)....972
5.3.2.2.12 Day-of-Week Alarm Register (RTC_RWKAR)/Binary Counter 3 Alarm Register (RTC_BCNT3AR)....974
5.3.2.2.13 Date Alarm Register (RTC_RDAYAR)/Binary Counter 0 Alarm Enable Register (RTC_BCNT0AER)....976
5.3.2.2.14 Month Alarm Register (RTC_RMONAR)/Binary Counter 1 Alarm Enable Register (RTC_BCNT1AER)....978
5.3.2.2.15 Year Alarm Register (RTC_RYRAR)/Binary Counter 2 Alarm Enable Register (RTC_BCNT2AER)....980
5.3.2.2.16 Year Alarm Enable Register (RTC_RYRAREN)/Binary Counter 3 Alarm Enable Register (RTC_BCNT3AER)....982
5.3.2.2.17 RTC Status Register (RTC_RSR)....984
5.3.2.2.18 RTC Control Register 1 (RTC_RCR1)....985
5.3.2.2.19 RTC Control Register 2 (RTC_RCR2)....986
5.3.2.2.20 Time Error Adjustment Register (RTC_RADJ)....988
5.3.3 Operation....989
5.3.3.1 Outline of Initial Settings of Registers after Power On....989
5.3.3.2 Clock and Count Mode Setting Procedure....990
5.3.3.3 Setting the Time....991
5.3.3.4 30-Second Adjustment....992
5.3.3.5 Reading 64-Hz Counter and Time....993

5.3.3.6 Alarm Function....994
5.3.3.7 Time Error Adjustment Function....995
5.3.3.7.1 Automatic adjustment....995
5.3.3.7.2 Adjustment by software....996
5.3.3.7.3 Procedure for changing the mode of adjustment....996
5.3.3.7.4 Procedure for stopping adjustment....997
5.3.4 Interrupt Sources....998
5.3.4.1 Alarm Interrupt (ALM)....998
5.3.4.2 Periodic Interrupt (PRD)....998
5.3.4.3 Carry Interrupt (CUP)....999
5.3.5 Event Link Output....1000
5.3.5.1 Interrupt Handling and Event Linking....1000
5.3.6 Usage Notes....1001
5.3.6.1 Register Writing during Counting....1001
5.3.6.2 Use of Periodic Interrupts....1001
5.3.6.3 Transitions to Low Power Consumption Modes after Setting Registers....1001
5.3.6.4 Notes on Writing to and Reading from Registers....1002
5.3.6.5 Changing the Count Mode....1002
5.3.6.6 Initialization Procedure When the RTC is Not to be Used....1002
5.4 Watchdog Timer (WDT)....1003
5.4.1 Overview....1003
5.4.2 Registers....1005
5.4.2.1 List of Registers (m = 0 to 3)....1005
5.4.2.2 Register Description....1006
5.4.2.2.1 WDT Refresh Register (WDTm_WDTRR)....1006
5.4.2.2.2 WDT Control Register (WDTm_WDTCR)....1007
5.4.2.2.3 WDT Status Register (WDTm_WDTSR)....1010
5.4.2.2.4 WDT Reset Control Register (WDTm_WDTRCR)....1012
5.4.3 Operation....1013
5.4.3.1 Register Start Mode....1013
5.4.3.2 Controlling Writes to the WDTCR and WDTRCR Registers....1015
5.4.3.3 Refresh Operation....1016
5.4.3.4 Status Flags....1018
5.4.3.5 Interrupt Sources....1018
5.4.3.6 Reading the Down-Counter Value....1019
5.4.4 Output to the Event Link Controller (ELC)....1020
5.4.5 Example WDT1 Setting for Error System Reset....1020
5.5 General Timer (GTM)....1021
5.5.1 Functional Overview....1021
5.5.1.1 Features of GTM....1021
5.5.2 Registers....1023
5.5.2.1 List of Registers....1023
5.5.2.2 Register Description....1024
5.5.2.2.1 OSTM Compare Register (OSTMnCMP)....1024
5.5.2.2.2 OSTM Counter Register (OSTMnCNT)....1025
5.5.2.2.3 OSTM Count Enable Status Register (OSTMnTE)....1026
5.5.2.2.4 OSTM Count Start Trigger Register (OSTMnTS)....1027
5.5.2.2.5 OSTM Count Stop Trigger Register (OSTMnTT)....1028

5.5.2.2.6 OSTM Control Register (OSTMnCTL) ... 1029
5.5.3 Functional Description ... 1030
5.5.3.1 Block Diagram ... 1030
5.5.3.2 Count Clock ... 1030
5.5.3.3 Generation of Interrupt Request ... 1031
5.5.3.4 Starting and Stopping the Timer ... 1032
5.5.3.5 Interval Timer Mode ... 1033
5.5.3.5.1 Basic Operation in Interval Timer Mode ... 1033
5.5.3.5.2 Operation when OSTMnCMP = 0000_0000h ... 1035
5.5.3.6 Free-Running Comparison Mode ... 1036
5.5.3.6.1 Basic Operation in Free-Running Comparison Mode ... 1036
5.5.3.6.2 Operation when OSTMnCMP = 0000_0000h ... 1038
5.6 Compare Match Timer W (CMTW) ... 1039
5.6.1 Overview ... 1039
5.6.2 Registers ... 1041
5.6.2.1 List of Registers ... 1041
5.6.2.2 Register Description ... 1042
5.6.2.2.1 Timer Start Register (CMTWm_CMWSTR) ... 1042
5.6.2.2.2 Timer Control Register (CMTWm_CMWCR) ... 1043
5.6.2.2.3 Timer I/O Control Register (CMTWm_CMWIOR) ... 1045
5.6.2.2.4 Timer Counter (CMTWm_CMWCNT) ... 1047
5.6.2.2.5 Compare Match Constant Register (CMTWm_CMWCOR) ... 1048
5.6.2.2.6 Input Capture Registers n (CMTWm_CMWICRn) (n = 0, 1) ... 1049
5.6.2.2.7 Output Compare Registers n (CMTWm_CMWOCRn) (n = 0, 1) ... 1050
5.6.3 Operation ... 1051
5.6.3.1 Period Counting Operation ... 1051
5.6.3.2 Compare Match Function ... 1052
5.6.3.3 Output Compare Function ... 1054
5.6.3.4 Input Capture Function ... 1056
5.6.3.5 Counter Size ... 1057
5.6.3.6 Count Timing Based on CMWCNT ... 1058
5.6.3.7 Output Compare Output Timing ... 1058
5.6.3.8 Input Capture Timing ... 1059
5.6.4 Interrupts ... 1060
5.6.4.1 CMTW Interrupt Sources and DMAC Transfer Requests ... 1060
5.6.4.2 Timing of Compare Match Interrupt Generation ... 1061
5.6.4.3 Timing of Output Compare Interrupt Generation ... 1061
5.6.4.4 Timing of Input Capture Interrupt Generation ... 1062
5.6.5 Event Link Operation ... 1063
5.6.5.1 Issuing Event Signals to the ELC ... 1063
5.6.5.1.1 Compare match event ... 1063
5.6.5.1.2 Output compare event ... 1063
5.6.6 Usage Notes ... 1064
5.6.6.1 Contention between CMWCNT Counter Writing and Compare Match ... 1064
5.6.6.2 Contention between CMWCNT Counter Writing and Incrementing or Clearing ... 1064
5.6.6.3 Contention between CMWCOR Register Writing and Compare Match ... 1065
5.6.6.4 Contention between CMWOCRn Register Writing and Compare Match ... 1065
5.6.6.5 Contention between CMWCNT Counter Reading and Incrementing or Clearing ... 1066

5.6.6.6 Contention Between CMWICRn Register Reading and Input Capture 1066
5.7 General-Purpose Timer (GPT) 1068
5.7.1 Overview 1068
5.7.2 Registers 1073
5.7.2.1 List of Registers 1073
5.7.2.2 Register Description 1075
5.7.2.2.1 General-Purpose Timer Write-Protection Register (GPTm_n_GTWP) 1075
5.7.2.2.2 General-Purpose Timer Software Start Register (GPTm_n_GTSTR) 1078
5.7.2.2.3 General-Purpose Timer Software Stop Register (GPTm_n_GTSTP) 1080
5.7.2.2.4 General-Purpose Timer Software Clear Register (GPTm_n_GTCLR) 1082
5.7.2.2.5 General-Purpose Timer Start Source Select Register (GPTm_n_GTSSR) 1083
5.7.2.2.6 General-Purpose Timer Stop Source Select Register (GPTm_n_GTPSR) 1087
5.7.2.2.7 General-Purpose Timer Clear Source Select Register (GPTm_n_GTCSR) 1091
5.7.2.2.8 General-Purpose Timer Up Count Source Select Register (GPTm_n_GTUPSR) 1095
5.7.2.2.9 General-Purpose Timer Down Count Source Select Register (GPTm_n_GTDNSR) 1099
5.7.2.2.10 General-Purpose Timer Input Capture Source Select Register A (GPTm_n_GTICASR) 1103
5.7.2.2.11 General-Purpose Timer Input Capture Source Select Register B (GPTm_n_GTICBSR) 1107
5.7.2.2.12 General-Purpose Timer Control Register (GPTm_n_GTCR) 1111
5.7.2.2.13 General-Purpose Timer Count Direction and Duty Setting Register (GPTm_n_GTUDDTYC) 1113
5.7.2.2.14 General-Purpose Timer I/O Control Register (GPTm_n_GTIOR) 1116
5.7.2.2.15 General-Purpose Timer Interrupt Output Setting Register (GPTm_n_GTINTAD) 1120
5.7.2.2.16 General-Purpose Timer Status Register (GPTm_n_GTST) 1123
5.7.2.2.17 General-Purpose Timer Buffer Enable Register (GPTm_n_GTBER) 1130
5.7.2.2.18 General-Purpose Timer Interrupt and A/D Converter Start Request Setting Register (GPTm_n_GTITC) 1135
5.7.2.2.19 General-Purpose Timer Counter (GPTm_n_GTCNT) 1138
5.7.2.2.20 General-Purpose Timer Compare Capture Register k (GPTm_n_GTCCRk) (k = A to F) 1139
5.7.2.2.21 General-Purpose Timer Cycle Setting Register (GPTm_n_GTPR) 1140
5.7.2.2.22 General-Purpose Timer Cycle Setting Buffer Register (GPTm_n_GTPBR) 1141
5.7.2.2.23 General-Purpose Timer Cycle Setting Double-Buffer Register (GPTm_n_GTPDBR) 1142

5.7.2.2.24 A/D Converter Start Request Timing Register k (GPTm_n_GTADTRk) (k = A, B) 1143
5.7.2.2.25 A/D Converter Start Request Timing Buffer Register k (GPTm_n_GTADTBRk) (k = A, B) 1144
5.7.2.2.26 A/D Converter Start Request Timing Double-Buffer Register k (GPTm_n_GTADTDBRk) (k = A, B) 1145
5.7.2.2.27 General-Purpose Timer Dead Time Control Register (GPTm_n_GTDTCR) 1146
5.7.2.2.28 General-Purpose Timer Dead Time Value Register k (GPTm_n_GTDVk) (k = U, D) 1148
5.7.2.2.29 General-Purpose Timer Dead Time Buffer Register k (GPTm_n_GTDBk) (k = U, D) 1149
5.7.2.2.30 General-Purpose Timer Output Protection Function Status Register (GPTm_n_GTSOS) 1150
5.7.2.2.31 General-Purpose Timer Output Protection Function Temporary Release Register (GPTm_n_GTSOTR) 1151
5.7.2.2.32 General-Purpose Timer Extended Interrupt Skipping Counter Control Register (GPTm_n_GTEITC) 1152
5.7.2.2.33 General-Purpose Timer Extended Interrupt Skipping Setting Register 1 (GPTm_n_GTEITLI1) 1155
5.7.2.2.34 General-Purpose Timer Extended Interrupt Skipping Setting Register 2 (GPTm_n_GTEITLI2) 1158
5.7.2.2.35 General-Purpose Timer Extended Buffer Transfer Skipping Setting Register (GPTm_n_GTEITLB) 1160
5.7.2.2.36 General-Purpose Timer Inter Channel Logical Operation Function Setting Register (GPTm_n_GTICLF) 1162
5.7.2.2.37 General-Purpose Timer Operation Enable Bit Simultaneous Control Channel Select Register (GPTm_n_GTSECSR) 1165
5.7.2.2.38 General-Purpose Timer Operation Enable Bit Simultaneous Control Register (GPTm_n_GTSECR) 1167
5.7.3 Operation 1169
5.7.3.1 Basic Operation 1169
5.7.3.1.1 Counter Operation 1169
5.7.3.1.2 Waveform Output by Compare Match 1175
5.7.3.1.3 Input Capture Function 1179
5.7.3.2 Buffer Operation 1181
5.7.3.2.1 GTPR Register Buffer Operation 1181
5.7.3.2.2 Buffer Operation for GTCCRA and GTCCRB Registers 1184
5.7.3.2.3 Buffer Operation for GTADTRA and GTADTRB 1191
5.7.3.3 PWM Output Operating Mode 1195
5.7.3.3.1 Saw-wave PWM Mode 1195
5.7.3.3.2 Saw-Wave One-Shot Pulse Mode 1197
5.7.3.3.3 Triangle-Wave PWM Mode 1 (32-Bit Transfer at Trough) 1200

5.7.3.3.4 Triangle-Wave PWM Mode 2 (32-Bit Transfer at Crest and Trough) 1202
5.7.3.3.5 Triangle-Wave PWM Mode 3 (64-Bit Transfer at Trough) 1205
5.7.3.4 Automatic Dead Time Setting Function 1208
5.7.3.5 Count Direction Changing Function 1214
5.7.3.6 Function of Output Duty 0% and 100% 1215
5.7.3.7 Hardware Count Start/Count Stop and Clear Operation 1217
5.7.3.7.1 Hardware Start Operation 1217
5.7.3.7.2 Hardware Stop Operation 1218
5.7.3.7.3 Hardware Clear Operation 1220
5.7.3.8 Synchronized Operation 1223
5.7.3.8.1 Synchronized Operation by Software 1223
5.7.3.8.2 Synchronized Operation by Hardware 1225
5.7.3.9 PWM Output Operation Examples 1227
5.7.3.9.1 Synchronized PWM Output 1227
5.7.3.9.2 3-Phase Saw-Wave Complementary PWM Output 1228
5.7.3.9.3 3-Phase Saw-Wave Complementary PWM Output with Automatic Dead Time Setting 1229
5.7.3.9.4 3-Phase Triangle-Wave Complementary PWM Output 1230
5.7.3.9.5 3-Phase Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting 1231
5.7.3.9.6 3-Phase Asymmetric Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting 1232
5.7.3.10 Phase Counting Function 1233
5.7.3.11 Inter Channel Logical Operation Function 1243
5.7.4 Interrupt Sources 1245
5.7.4.1 Interrupt Sources 1245
5.7.4.1.1 GPT_Ux_gpt_gtcia_n_m 1246
5.7.4.1.2 GPT_Ux_gpt_gtcib_n_m 1246
5.7.4.1.3 GPT_Ux_gpt_gtcic_n_m 1246
5.7.4.1.4 GPT_Ux_gpt_gtcid_n_m 1246
5.7.4.1.5 GPT_Ux_gpt_gtcie_n_m 1246
5.7.4.1.6 GPT_Ux_gpt_gtcif_n_m 1247
5.7.4.1.7 GPT_Ux_gpt_gtciv_n_m 1247
5.7.4.1.8 GPT_Ux_gpt_gtciu_n_m 1247
5.7.4.1.9 GPT_Ux_gpt_gtdei_n_m 1247
5.7.4.1.10 GPT_Ux_gpt_gtcih_n_m 1247
5.7.4.1.11 GPT_Ux_gpt_gtcil_n_m 1248
5.7.4.1.12 GPT_Ux_gpt_gtciada_n_m 1248
5.7.4.1.13 GPT_Ux_gpt_gtciadb_n_m 1248
5.7.4.2 DMAC Activation 1249
5.7.4.3 Interrupt and A/D Converter Start Request Skipping Function 1249
5.7.4.3.1 Interrupt Skipping Function by GTITC Register 1249
5.7.4.3.2 Extended Interrupt Skipping Function 1253
5.7.5 A/D Converter Start Request 1256
5.7.6 Operations Linked by ELC 1259

5.7.6.1 Event Signal Output to ELC 1259
5.7.6.2 Event Signal Inputs from ELC 1259
5.7.7 Noise Filter Function 1260
5.7.8 Protection Function 1261
5.7.8.1 Write-Protection for Registers 1261
5.7.8.2 Disabling of Buffer Operation 1261
5.7.8.2.1 Simultaneous Control of Buffer Operations of Multiple Channels 1263
5.7.8.2.2 Repeated Double-Buffered Operation when GTCCR Buffer Transfer is Inhibited 1264
5.7.8.2.3 Repeated Double-Buffered Operation when GTADTR Buffer Transfer is Inhibited 1270
5.7.8.3 GTIOCnm Pin Output Negate Control (n = 0 to 15, m = A, B) 1271
5.7.8.4 GTIOCnm Protection Function for GTIOCnm Pin Output (n = 0 to 15; m = A, B) 1273
5.7.9 Initialization Method of Output Pins 1283
5.7.9.1 Pin Settings after Reset 1283
5.7.9.2 Pin Initialization Due to Error during Operation 1284
5.7.10 Usage Notes 1285
5.7.10.1 Settings of the GTCCRn Register during Compare Match Operation (n = A to F) 1285
5.7.10.2 Setting Range for GTCNT Counter 1286
5.7.10.3 Starting and Stopping the GTCNT Counter 1286
5.7.10.4 Priority Order of Each Event 1287
5.8 Port Output Enable for GPT (POEG) 1289
5.8.1 Overview 1289
5.8.2 Registers 1291
5.8.2.1 List of Registers 1291
5.8.2.2 Register Description 1292
5.8.2.2.1 POEG Group n Setting Register (POEG_POEGGn) (n = A to H) 1292
5.8.3 Output-Disable Control Operation 1294
5.8.3.1 Pin Input Level Detection Operation 1295
5.8.3.1.1 Digital Filter 1295
5.8.3.2 Output-Disable Requests from the GPT 1296
5.8.3.3 Output-Disable Control Using Registers 1296
5.8.3.4 Release from Output-Disable 1296
5.8.4 Interrupt Sources 1297
5.8.5 External Trigger Output to the GPT 1298
5.8.6 Usage Notes 1299
5.8.6.1 Transition to Software Standby Mode 1299
5.8.6.2 Specifying Pins Associated with the GPT 1299
SECTION 6 HIGH-SPEED INTERFACE 1300
6.1 High-Speed Interface Overview 1300
6.2 SD/MMC Host Interface (SD) 1301
6.2.1 Overview 1301
6.2.1.1 Features 1301
6.2.1.2 Block Diagram 1302
6.2.1.3 External Pins 1303
6.2.2 SD Registers 1304
6.2.2.1 List of Registers 1304

6.2.2.2 SD Register Description 1306
6.2.2.2.1 Command Type Register (SDm_SD_CMD) 1306
6.2.2.2.2 Command Argument Registers 1308
6.2.2.2.3 Data Stop Register (SDm_SD_STOP) 1310
6.2.2.2.4 Block Count Register (SDm_SD_SECCNT) 1312
6.2.2.2.5 SD Card Response Registers 1313
6.2.2.2.6 SD Card Interrupt Flag Register 1 (SDm_SD_INFO1) 1321
6.2.2.2.7 SD Card Interrupt Flag Register 2 (SDm_SD_INFO2) 1324
6.2.2.2.8 SD_INFO1 Interrupt Mask Register (SDm_SD_INFO1_MASK) 1327
6.2.2.2.9 SD_INFO2 Interrupt Mask Register (SDm_SD_INFO2_MASK) 1328
6.2.2.2.10 SD Clock Control Register (SDm_SD_CLK_CTRL) 1329
6.2.2.2.11 Transfer Data Length Register (SDm_SD_SIZE) 1331
6.2.2.2.12 SD Card Access Control Option Register (SDm_SD_OPTION) 1332
6.2.2.2.13 SD Error Status Register 1 (SDm_SD_ERR_STS1) 1334
6.2.2.2.14 SD Error Status Register 2 (SDm_SD_ERR_STS2) 1336
6.2.2.2.15 SD Buffer Read/Write Register (SDm_SD_BUF0) 1337
6.2.2.2.16 SDIO Mode Control Register (SDm_SDIO_MODE) 1338
6.2.2.2.17 SDIO Interrupt Flag Register (SDm_SDIO_INFO1) 1340
6.2.2.2.18 SDIO_INFO1 Interrupt Mask Register (SDm_SDIO_INFO1_MASK) 1341
6.2.2.2.19 DMA Mode Enable Register (SDm_CC_EXT_MODE) 1342
6.2.2.2.20 Software Reset Register (SDm_SOFT_RST) 1343
6.2.2.2.21 Version Register (SDm_VERSION) 1344
6.2.2.2.22 Host Interface Mode Setting Register (SDm_HOST_MODE) 1345
6.2.2.2.23 SD Interface Mode Setting Register (SDm_SDIF_MODE) 1346
6.2.2.2.24 SD Status Register (SDm_SD_STATUS) 1347
6.2.2.2.25 DMAC Transfer Mode Register (SDm_DM_CM_DTRAN_MODE) 1348
6.2.2.2.26 DMAC Transfer Control Register (SDm_DM_CM_DTRAN_CTRL) 1349
6.2.2.2.27 DMAC Reset Register (SDm_DM_CM_RST) 1350
6.2.2.2.28 DMAC Interrupt Register 1 (SDm_DM_CM_INFO1) 1351
6.2.2.2.29 DM_CM_INFO1 Interrupt Mask Register (SDm_DM_CM_INFO1_MASK) 1353
6.2.2.2.30 DMAC Interrupt Register 2 (SDm_DM_CM_INFO2) 1354
6.2.2.2.31 DM_CM_INFO2 Interrupt Mask Register (SDm_DM_CM_INFO2_MASK) 1355
6.2.2.2.32 DMAC Transfer Address Register (SDm_DM_DTRAN_ADDR) 1356
6.2.3 Operation 1357
6.2.3.1 SD Interface 1357
6.2.3.1.1 SD data format 1357
6.2.3.1.2 Bus signal voltage switch 1361
6.2.3.2 Card Detect/Write Protect 1362
6.2.3.2.1 Card detect 1362
6.2.3.2.2 Write protect 1363
6.2.3.3 Interrupt Request 1364

6.2.3.3.1 Interrupt request 1364
6.2.3.4 Communications Errors and Timeouts 1365
6.2.4 Usage Example 1367
6.2.4.1 Command without Data Transfer 1367
6.2.4.1.1 Flowchart 1367
6.2.4.1.2 Operation for command without data transfer 1368
6.2.4.2 Single Block Read 1369
6.2.4.2.1 Flowchart 1369
6.2.4.2.2 Operation for single block read 1370
6.2.4.3 Single Block Write 1371
6.2.4.3.1 Flowchart 1371
6.2.4.3.2 Operation for single block write 1372
6.2.4.4 Multiple Block Read 1373
6.2.4.4.1 Flowchart 1373
6.2.4.4.2 Operation for multiple block read 1374
6.2.4.5 Multiple Block Write (when Using Internal Timer) 1375
6.2.4.5.1 Flowchart 1375
6.2.4.5.2 Operation for multiple block write 1376
6.2.4.6 Multiple Block Write (when Using External Timer) 1377
6.2.4.6.1 Flowchart 1377
6.2.4.6.2 Operation for multiple block write 1378
6.2.4.7 IO_RW_DIRECT Command (CMD52) 1379
6.2.4.7.1 Flowchart 1379
6.2.4.8 IO_RW_EXTENDED (CMD53/Multiple Block Read) 1380
6.2.4.8.1 Flowchart 1380
6.2.4.9 IO_RW_EXTENDED (CMD53/Multiple Block Write) 1382
6.2.4.9.1 Flowchart 1382
6.2.4.10 DMA Transfer 1384
6.2.4.10.1 SD_BUF DMA transfer 1384
6.2.4.11 High-Priority Interrupt (without Data Transfer) 1386
6.2.4.11.1 Flowchart 1386
6.2.4.11.2 Operation for high-priority interrupt without data transfer 1387
6.2.4.12 High-Priority Interrupt (at Single Block Write) 1388
6.2.4.12.1 Flowchart 1388
6.2.4.12.2 Operation for HPI at single block write 1389
6.2.4.13 High-Priority Interrupt (at Multiple Block Write) 1390
6.2.4.13.1 Flowchart 1390
6.2.4.13.2 Operation for HPI at multiple block transfer 1392
6.2.4.14 Example of SD_CMD Register Setting 1394
6.2.5 Usage Notes 1397
6.2.5.1 SD_BUF Illegal Write Access 1397
6.2.5.2 Block Number Limitation for Multiple Block Read 1397
6.2.5.3 Automatic Control of SDCLK Output 1399
6.2.5.4 Control of the C52PUB Setting for Multiple Block Write 1399
6.2.5.5 Notes on SD_CLK_CTRL Register Settings 1399
6.2.5.6 Restrictions on specifications 1399
6.2.5.7 STP bit setting during multiple block read 1400

6.2.6 Sampling Clock Controller (SCC) 1401
6.2.6.1 Features 1401
6.2.6.2 SCC Block Diagram 1401
6.2.7 SCC Registers 1402
6.2.7.1 List of Registers 1402
6.2.7.2 SCC Register Descriptions 1403
6.2.7.2.1 Initial Setting Register (SDm_SCC_DTCNTL) 1403
6.2.7.2.2 Sampling Clock Position Setting Register (SDm_SCC_TAPSET) 1404
6.2.7.2.3 Sampling Clock Selection Register (SDm_SCC_CKSEL) 1405
6.2.7.2.4 Sampling Clock Position Correction Register (SDm_SCC_RVSCNTL) ... 1406
6.2.7.2.5 Sampling Clock Position Correction Request Register (SDm_SCC_RVSREQ) 1407
6.2.7.2.6 Hardware Adjustment Register 1 (SDm_SCC_DT2FF) 1408
6.2.7.2.7 Sampling Data Comparison Register (SDm_SCC_SMPCMP) 1409
6.2.7.2.8 Hardware Adjustment Register 2 (SDm_SCC_TMPPORT) 1410
6.2.8 Usage Example of SCC 1411
6.2.8.1 Tuning 1411
6.2.8.2 Sampling Clock Position Correction after Tuning 1414
6.2.8.2.1 Manual correction of the sampling clock position 1414
6.2.8.2.2 Automatic correction of the sampling clock position 1415
6.2.8.3 Change point of the input data 1416
6.3 Gigabit Ethernet Interface (GBETH) 1417
6.3.1 Overview 1417
6.3.1.1 Features 1417
6.3.1.2 Block Diagram 1419
6.3.1.3 External Pins 1420
6.4 USB3.2 Gen2x1 Interface (USB3) 1421
6.4.1 Functional Overview 1421
6.4.2 Connection Configuration 1422
6.4.3 Pin Functions 1423
6.4.4 USB3.2 Gen1 Host Controller (USB3HOST) 1424
6.4.4.1 Introduction 1424
6.4.4.1.1 Features 1424
6.4.4.2 Interrupts 1426
6.4.4.2.1 Interrupt diagram 1426
6.4.4.2.2 Interrupt sources 1427
6.4.4.3 Register Specification 1428
6.4.4.3.1 Register Overview 1429
6.4.4.3.2 Register List 1430
6.4.4.4 Register Descriptions 1433
6.4.4.4.1 Host Controller Capability Registers 1434
6.4.4.4.2 Host Controller Operational Registers 1443
6.4.4.4.3 Port Register Sets in Host Controller Operational Registers 1454
6.4.4.4.4 xHCI Extended Capabilities 1467
6.4.4.4.5 Host Controller Runtime Registers 1485
6.4.4.4.6 Doorbell Registers 1493

6.4.4.4.7 Core Defined Registers 1495
6.4.4.4.8 PIPE Status and Control Registers 1518
6.4.4.5 Description of Functions 1523
6.4.4.5.1 Battery Charging 1523
6.4.4.5.2 Initialization Flow 1525
6.4.4.5.3 xHCD Initialization Flow 1526
6.4.4.5.4 Interrupt Flow 1528
6.4.4.5.5 xHCI Command 1533
6.4.4.5.6 Examples of Transaction Flows 1534
6.4.4.5.7 Functional Implementation Constraint 1536
6.4.5 USB Test Module (USB3TEST/USB2TEST) 1537
6.4.5.1 Features 1537
6.5 USB2.0 Interface 1538
6.5.1 USB2.0 PHY Controller 1538
6.5.1.1 Features 1538
6.5.1.1.1 Ch0: USB2.0 OTG/DRD (host/function) interface 1539
6.5.1.1.2 Ch1: USB2.0 host interface 1539
6.5.1.1.3 Block diagram 1540
6.5.1.1.4 External signal pins 1541
6.5.1.1.5 Power and GND pins 1541
6.5.1.1.6 Interrupt 1542
6.5.1.2 Interface Registers 1543
6.5.1.2.1 USB2PHY Control Register 1543
6.5.1.3 Operation 1543
6.5.2 USB2.0 Host Controller 1544
6.5.2.1 Overview 1544
6.5.2.1.1 Features 1544
6.5.2.1.2 Support of USB-related specifications 1548
6.5.2.2 Host Registers 1549
6.5.2.2.1 List of Registers 1550
6.5.2.2.2 OHCI Operational Register Descriptions 1552
6.5.2.2.3 EHCI Controller Capability Register Descriptions 1580
6.5.2.2.4 HCI Operational Register Descriptions 1585
6.5.2.2.5 AHB Bridge Register Descriptions 1599
6.5.2.2.6 UCOM Register Descriptions 1611
6.5.2.3 Clock Signals 1617
6.5.2.3.1 Clock Gating Specifications 1617
6.5.2.4 Interrupt Sources 1619
6.5.2.4.1 Interrupt Signals 1619
6.5.2.4.2 Interrupt Sources and Control 1619
6.5.2.4.3 Timing of De-asserting Interrupt Signals 1623
6.5.2.5 Power-Saving Function 1624
6.5.2.5.1 Controlling the SUSPENDM and SLEEPM Pins of the USB2PHY 1624
6.5.2.5.2 Controlling the Clock-Gating Function 1624
6.5.2.6 Battery Charging 1625

6.5.2.6.1 Support of charging port ... 1625
6.5.2.6.2 Support of portable device ... 1625
6.5.2.7 Bus Master ... 1626
6.5.2.7.1 Functional specifications of the bus master ... 1626
6.5.2.8 Overcurrent Control and VBUS Control ... 1629
6.5.2.8.1 OVRCURN/VBUSEN pin ... 1629
6.5.2.8.2 Overcurrent detection timer setting ... 1629
6.5.2.8.3 Port Power (VBUS) control specifications ... 1630
6.5.2.8.4 Timing Chart for Overcurrent Detection and Recovery ... 1631
6.5.2.9 Procedure for Setting this Module ... 1632
6.5.2.9.1 Host/Peripheral Common Setting Sequence ... 1632
6.5.2.9.2 Initialization Sequence ... 1633
6.5.2.9.3 Flow of Error Handling ... 1633
6.5.3 USB2.0 Function Controller ... 1634
6.5.3.1 Overview ... 1634
6.5.3.1.1 Features ... 1635
6.5.3.1.2 Overview of functions ... 1637
6.5.3.1.3 Restrictions and notes ... 1638
6.5.3.2 Function Controller Registers ... 1639
6.5.3.2.1 How to read the register table ... 1640
6.5.3.2.2 List of Registers ... 1641
6.5.3.2.3 System Configuration Control Registers ... 1644
6.5.3.2.4 System Configuration Status ... 1647
6.5.3.2.5 USB Signal Control Registers ... 1648
6.5.3.2.6 Test Mode Register ... 1650
6.5.3.2.7 FIFO Port Registers ... 1651
6.5.3.2.8 Interrupt Enable Registers ... 1659
6.5.3.2.9 SOF Control Register ... 1663
6.5.3.2.10 Interrupt Status ... 1664
6.5.3.2.11 Frame Number Registers ... 1673
6.5.3.2.12 USB Address ... 1676
6.5.3.2.13 USB Request Registers ... 1677
6.5.3.2.14 DCP Configuration ... 1681
6.5.3.2.15 Pipe Configuration Registers ... 1686
6.5.3.2.16 Pipe Control Registers ... 1697
6.5.3.2.17 Transaction Counters ... 1704
6.5.3.2.18 Low Power Control Register ... 1707
6.5.3.2.19 Low Power Status Register ... 1708
6.5.3.2.20 PHY Function Control Register ... 1709
6.5.3.2.21 PHY_OTG Control Register ... 1710
6.5.3.2.22 Peripheral L1 Control Register 1 ... 1711
6.5.3.2.23 Peripheral L1 Control Register 2 ... 1713
6.5.3.2.24 Next Register Set ... 1714
6.5.3.2.25 Current Register Set ... 1717

6.5.3.2.26 Channel Register Set 1720
6.5.3.2.27 Link Register Set 1732
6.5.3.2.28 Skip Register Set 1734
6.5.3.2.29 DMA Register Set 1740
6.5.3.3 Functions 1747
6.5.3.3.1 System Control and Oscillation Control 1747
6.5.3.3.2 Interrupt Function 1748
6.5.3.3.3 Pipe Control 1754
6.5.3.3.4 FIFO Buffer 1758
6.5.3.3.5 FIFO Port Functions 1760
6.5.3.3.6 Control Transfer (DCP) 1763
6.5.3.3.7 Bulk Transfer (Pipes 1 to 5) 1765
6.5.3.3.8 Interrupt Transfer (Pipes 6 to 9) 1765
6.5.3.3.9 Isochronous Transfer (Pipes 1 and 2) 1766
6.5.3.3.10 SOF Interpolation Function 1772
6.5.3.3.11 Link Power Management Processing 1773
6.5.3.3.12 DMA Mode 1775
6.5.3.3.13 DMA Transfer 1793
6.5.3.3.14 Access Type 1804
6.5.3.3.15 Arbitration between DMACs 1808
6.5.3.3.16 Notes 1809
6.6 PCI Express 3.0 Interface (PCIe) 1810
6.6.1 Overview 1810
6.6.2 TOP Pins 1812
6.6.2.1 List of external pins 1812
6.6.2.2 List of internal pins 1813
6.6.3 Block Diagram 1818
6.6.3.1 Reset 1818
6.6.4 Registers 1819
6.6.4.1 Register Descriptions (Root Complex Mode) 1820
6.6.4.1.1 List of AXI Bridge Registers 1821
6.6.4.1.2 List of PCI Express Configuration Registers (Type1) 1827
6.6.4.1.3 AXI Bridge Register Descriptions 1829
6.6.4.1.4 PCI Express Configuration Register Descriptions (Type1) 1923
6.6.4.2 Register Descriptions (Endpoint Mode) 2004
6.6.4.2.1 List of AXI Bridge Registers 2005
6.6.4.2.2 List of PCI Express Configuration Registers (Type0) 2011
6.6.4.2.3 AXI Bridge Register Descriptions 2013
6.6.4.2.4 PCI Express Configuration Register Descriptions (Type0) 2102
6.6.4.3 PHY Register Descriptions 2191
6.6.4.3.1 List of Physical Layer Control/Status Registers 2191
6.6.4.3.2 Register Descriptions 2192
6.6.5 Functional Description 2200
6.6.5.1 PCIe Core Functions 2200
6.6.5.2 Issuing of PCIe Requests and Register Access (by AXI) 2201

6.6.5.2.1 PCIe Requests which can be Issued (Supported Commands for TX) 2201
6.6.5.2.2 Register Access 2202
6.6.5.2.3 Issuing Memory Requests 2202
6.6.5.2.4 Issuing Special Requests 2204
6.6.5.3 Initiation of AXI Transactions and Register Access (by PCIe) 2207
6.6.5.3.1 PCIe Requests which can be Received (Supported Commands for RX) 2207
6.6.5.3.2 Initiation of AXI Transactions 2208
6.6.5.3.3 Narrow Transfer 2209
6.6.5.4 DMAC Functions 2210
6.6.5.4.1 Register-Type Transfer 2210
6.6.5.4.2 Descriptor-Type Transfer 2214
6.6.5.4.3 Method of Transfer 2223
6.6.5.4.4 Inter-Channel Arbitration 2223
6.6.5.4.5 DMA Completed Interrupt 2224
6.6.5.5 Reception of PCIe Commands 2225
6.6.5.5.1 Reception of MSIs (Root Complex) 2225
6.6.5.5.2 Setting the MSI Window 2227
6.6.5.5.3 Reception of an Interrupt in Response to a Message Request (Root Complex) 2228
6.6.5.5.4 Reception of Message Requests 2228
6.6.5.6 Interrupt 2229
6.6.5.6.1 Error and Event Interrupt Notification 2229
6.6.5.6.2 INTx/MSI Interrupt Notification (Root Complex) 2230
6.6.5.6.3 Issuing INTx/MSI Interrupts (Endpoint) 2231
6.6.5.6.4 Root Complex Interrupt Notification 2235
6.6.5.6.5 DMA Interrupt 2242
6.6.5.7 Power Management (Root Complex Mode) 2243
6.6.5.7.1 PCI Power Management (PCI-PM) 2243
6.6.5.7.2 Active State Power Management (ASPM) 2244
6.6.5.8 Power Management (Endpoint Mode) 2245
6.6.5.8.1 PCI Power Management (PCI-PM) 2245
6.6.5.8.2 Active State Power Management (ASPM) 2247
6.6.5.9 Power Management (Common) 2248
6.6.5.9.1 Turn Off (PME Turn Off Message Recept) 2248
6.6.5.9.2 Wake Up (PME Message transmit) 2248
6.6.5.10 Error Processing 2249
6.6.5.10.1 Error Classification 2249
6.6.5.10.2 Error Checking Mechanism 2250
6.6.5.10.3 Error Message 2251
6.6.6 Operation 2252
6.6.6.1 Setting Up 2252
6.6.6.1.1 Changing the Initial Values of the Registers 2253
6.6.6.1.2 De-asserting the Reset 2254
6.6.6.1.3 Checking if the Link is Up 2254
6.6.6.2 Setting the Windows (Root Complex Mode) 2255

6.6.6.3 Setting the Windows (Endpoint Mode) 2255
6.6.6.3.1 BAR Specification (Two-Function Configuration) 2255
6.6.6.3.2 Setting the AXI Windows 2257
6.6.6.3.3 Setting the PCIe Address Space 2260
6.6.6.4 Data Transfer 2263
6.6.6.4.1 PCIe MWr (when the AXI Slave is in Use) 2263
6.6.6.4.2 PCIe MWr (when the DMAC is in Use) 2265
6.6.6.4.3 PCIe MRd (when the AXI Slave is in Use) 2267
6.6.6.4.4 PCIe MRd (when the DMAC is in Use) 2269
6.6.6.5 PCIe Initialization Procedure 2271
6.6.6.5.1 Initialization Procedure 2271
6.6.7 Points for Caution and Restrictions....2275
6.6.7.1 Points for Caution and Prohibited Items in the Issuing of Requests 2275
6.6.7.2 Ordering Specifications of Received Non-Posted and Posted Requests....2275
6.6.7.3 Caution when Changing the Speed Spontaneously by the EP Unit....2275
6.6.7.4 Error Processing of Unsupported Requests....2276
6.6.7.5 Processing on Reception of the Message....2277
6.6.7.6 Notes on Window Settings 2278
6.6.7.7 Other Points for Caution 2279
SECTION 7 LOW-SPEED INTERFACE....2280
7.1 Low-Speed Interface Overview 2280
7.2 Expanded Serial Peripheral Interface (xSPI)....2281
7.2.1 Overview 2281
7.2.2 Registers....2284
7.2.2.1 List of Registers....2284
7.2.2.2 Register Description 2286
7.2.2.2.1 xSPI Configuration Registers 2286
7.2.2.2.2 xSPI Control Registers 2294
7.2.2.2.3 xSPI Status Registers 2320
7.2.2.2.4 xSPI Interrupt Registers 2324
7.2.3 Operation 2330
7.2.3.1 xSPI Bus....2330
7.2.3.1.1 Supported protocol mode 2330
7.2.3.1.2 xSPI frame interval 2337
7.2.3.1.3 xSPI signals timing control 2337
7.2.3.1.4 Automatic calibration 2341
7.2.3.2 Manual-command....2342
7.2.3.2.1 Direct mode 2342
7.2.3.2.2 Periodic mode 2342
7.2.3.3 Memory-mapping....2343
7.2.3.3.1 Configuration 2343
7.2.3.3.2 Write access operation 2344
7.2.3.3.3 Combination function.... 2346
7.2.3.3.4 Read access operation.... 2347
7.2.3.3.5 Prefetch function 2349

7.2.3.3.6 XiP mode ... 2350
7.2.3.4 Pattern Control ... 2350
7.2.3.4.1 XiP disable pattern ... 2350
7.2.3.4.2 Reset pattern ... 2351
7.2.3.4.3 CS only pattern ... 2351
7.2.3.5 Integrity Checking ... 2352
7.2.3.5.1 Interrupts ... 2352
7.2.3.6 Flows of Operations ... 2354
7.2.3.6.1 Flow of startup ... 2354
7.2.3.6.2 Flow of configuration ... 2354
7.2.3.6.3 Flow of communication stop ... 2356
7.2.3.6.4 Flow of automatic calibration ... 2357
7.2.3.6.5 Flow of manual-command procedure ... 2358
7.2.3.6.6 Flow of memory-mapping ... 2360
7.2.3.6.7 Flow of memory-mapping stop ... 2361
7.2.3.6.8 Flow of pattern request ... 2362
7.2.3.6.9 Flow of XiP mode ... 2363
7.2.3.7 Usage Notes ... 2364
7.3 Serial Communications Interface (RSCI) ... 2365
7.3.1 Overview ... 2365
7.3.2 SCI Registers ... 2369
7.3.2.1 List of Registers ... 2371
7.3.2.2 SCI Register Descriptions ... 2372
7.3.2.2.1 Receive Shift Register (RSCIm_RSR) ... 2372
7.3.2.2.2 Receive Data Register (RSCIm_RDR) ... 2373
7.3.2.2.3 Transmit Data Register (RSCIm_TDR) ... 2375
7.3.2.2.4 Transmit Shift Register (RSCIm_TSR) ... 2376
7.3.2.2.5 Common Control Register 0 (RSCIm_CCR0) ... 2377
7.3.2.2.6 Common Control Register 1 (RSCIm_CCR1) ... 2380
7.3.2.2.7 Common Control Register 2 (RSCIm_CCR2) ... 2385
7.3.2.2.8 Common Control Register 3 (RSCIm_CCR3) ... 2393
7.3.2.2.9 Common Control Register 4 (RSCIm_CCR4) ... 2397
7.3.2.2.10 Communication Enable Status Register (RSCIm_CESR) ... 2400
7.3.2.2.11 Half Data Communication Control Register (RSCIm_HCR) ... 2401
7.3.2.2.12 Simple I2C Control Register (RSCIm_ICR) ... 2402
7.3.2.2.13 FIFO Control Register (RSCIm_FCR) ... 2405
7.3.2.2.14 Driver Control Register (RSCIm_DCR) ... 2407
7.3.2.2.15 Simple-LIN (SCIX) Control Register 0 (RSCIm_XCR0) ... 2409
7.3.2.2.16 Simple-LIN (SCIX) Control Register 1 (RSCIm_XCR1) ... 2412
7.3.2.2.17 Simple-LIN (SCIX) Control Register 2 (RSCIm_XCR2) ... 2414
7.3.2.2.18 Common Status Register (RSCIm_CSR) ... 2415
7.3.2.2.19 Simple I2C Status Register (RSCIm_ISR) ... 2421
7.3.2.2.20 FIFO Receive Status Register (RSCIm_FRSR) ... 2422
7.3.2.2.21 FIFO Transmit Status Register (RSCIm_FTSR) ... 2424

7.3.2.2.22 Simple-LIN (SCIX) Status Register 0 (RSCIm_XSR0) ... 2425
7.3.2.2.23 Simple-LIN (SCIX) Status Register 1 (RSCIm_XSR1) ... 2428
7.3.2.2.24 Common Flag Clear Register (RSCIm_CFCLR) ... 2429
7.3.2.2.25 Simple I2C Flag Clear Register (RSCIm_ICFCLR) ... 2430
7.3.2.2.26 FIFO Flag Clear Register (RSCIm_FFCLR) ... 2431
7.3.2.2.27 Simple-LIN (SCIX) Flag Clear Register (RSCIm_XFCLR) ... 2432
7.3.3 Asynchronous Mode ... 2433
7.3.3.1 Serial Data Transfer Format ... 2434
7.3.3.2 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode ... 2435
7.3.3.3 Clock ... 2436
7.3.3.4 Double-Speed Operation and Frequency of 6 Times the Bit Rate ... 2436
7.3.3.5 CTS and RTS Function ... 2437
7.3.3.6 Address Match (Receive Data Match Detection) Function ... 2437
7.3.3.7 SCI Initialization (Asynchronous Mode) ... 2441
7.3.3.8 Serial Data Transmission (Asynchronous Mode) ... 2444
7.3.3.9 Serial Data Reception (Asynchronous Mode) ... 2451
7.3.3.10 The Function of Adjust Receive Sampling Timing (Asynchronous Mode) ... 2458
7.3.3.11 The Function of Adjust Transmit Timing (Asynchronous Mode) ... 2461
7.3.4 Multi-Processor Communication Function ... 2467
7.3.4.1 Multi-Processor Serial Data Transmission ... 2469
7.3.4.2 Multi-Processor Serial Data Reception ... 2471
7.3.5 Smart Card Interface Mode ... 2478
7.3.5.1 Sample Connection ... 2478
7.3.5.2 Data Format (Except in Block Transfer Mode) ... 2478
7.3.5.3 Block Transfer Mode ... 2480
7.3.5.4 Receive Data Sampling Timing and Reception Margin ... 2480
7.3.5.5 Initialization of SCI (Smart Card Interface Mode) ... 2481
7.3.5.6 Serial Data Transmission (Except in Block Transfer Mode) ... 2483
7.3.5.7 Serial Data Reception (Except in Block Transfer Mode) ... 2485
7.3.5.8 Clock Output Control ... 2487
7.3.6 Simple-LIN Mode ... 2489
7.3.6.1 Simple-LIN Start Frame Transmission ... 2490
7.3.6.2 Simple-LIN Start Frame Reception ... 2493
7.3.6.2.1 Simple-LIN Normal Reception of Start Frame (PIB Not Used) ... 2493
7.3.6.2.2 Simple-LIN Start Frame reception (using the priority interrupt bit) ... 2497
7.3.6.3 Simple-LIN Bus Conflict Detection Function ... 2498
7.3.6.4 Simple-LIN Bit Rate Measurement Function ... 2500
7.3.7 Simple I2C Mode ... 2501
7.3.7.1 Generation of Start, Restart, and Stop Conditions ... 2502
7.3.7.2 Clock Synchronization ... 2503
7.3.7.3 SDA Output Delay ... 2504
7.3.7.4 SCI Initialization (Simple I2C Mode) ... 2505
7.3.7.5 Operation in Master Transmission (Simple I2C Mode) ... 2506
7.3.7.6 Master Reception (Simple I2C Mode) ... 2512
7.3.8 Clock Synchronous Mode ... 2515
7.3.8.1 Clock ... 2515
7.3.8.2 CTS and RTS Functions ... 2516
7.3.8.3 Initializing the SCI (Clock Synchronous Mode) ... 2516
7.3.8.4 Serial Data Transmission (Clock Synchronous Mode) ... 2518
7.3.8.5 Serial Data Reception (Clock Synchronous Mode) ... 2523

7.3.8.6 Serial Data Transmission and Reception (Clock Synchronous Mode) 2528
7.3.8.7 Reception Sampling Timing Adjustment Function in Clock Synchronous Mode with Internal Clock Used 2530
7.3.9 Simple-SPI Mode 2531
7.3.9.1 Pin Status in Master and Slave Mode 2531
7.3.9.2 SS Function in Master Mode 2532
7.3.9.3 SS Function in Slave Mode 2532
7.3.9.4 Relationship between Clock and Transmit/Receive Data 2533
7.3.9.5 SCI Initialization in Simple-SPI Mode 2533
7.3.9.6 Serial Data Transmission and Reception in Simple-SPI Mode 2534
7.3.9.7 Reception Sampling Timing Adjustment Function in Simple-SPI Mode with Internal Clock Used 2534
7.3.10 Bit Rate Modulation Function 2535
7.3.11 Noise Cancellation Function 2536
7.3.12 RS-485 Driver Control Function 2537
7.3.13 Loopback Function 2538
7.3.14 Half-Duplex Communication Function 2539
7.3.15 Synchronizer Bypass Function 2540
7.3.16 Half Data Communication Function 2541
7.3.16.1 Reception in Half Data Communication 2541
7.3.16.2 Transmission in Half Data Communication 2543
7.3.16.3 Register Settings for Half Data Communication 2544
7.3.17 Interrupt Signal 2545
7.3.17.1 Buffer Operations for TXI and RXI Interrupts 2545
7.3.17.2 Interrupt in Asynchronous Mode, Clock Synchronous Mode, and Simple-SPI Mode 2545
7.3.17.3 Interrupt in Smart Card Interface Mode 2547
7.3.17.4 Interrupts in Simple-I2C Mode 2548
7.3.17.5 Interrupts in Simple-LIN Mode 2549
7.3.18 Usage Notes 2550
7.3.18.1 Notes on Module Standby Mode 2550
7.3.18.2 Break Detection and Processing 2554
7.3.18.3 Mark State and Production of Breaks 2555
7.3.18.4 Receive Error Flags and Transmit Operations (Clock Synchronous Mode and Simple SPI Mode) 2555
7.3.18.5 Writing Data to TDR 2555
7.3.18.6 Restrictions on Clock Synchronous Transmission (Clock Synchronous Mode and Simple SPI Mode) 2555
7.3.18.7 Notes on Starting Transfer 2557
7.3.18.8 Limitations on Simple SPI Mode 2557
7.3.18.9 Notes on Transmit Enable Bit (TE) 2558
7.3.18.10 Notes on Simple-LIN mode 2558
7.3.18.11 Notes on RS-485 Driver Control Function 2558
7.3.18.12 Notes on Loopback Function 2559
7.3.18.13 Notes Regarding Register Access when Operation Clock (RSCI_m_TCLK) is Slower than Bus Clock (RSCI_m_PCLK) 2559
7.3.18.14 Notes on Interrupting Operation 2559
7.3.18.15 Notes on Communication Mode Transition 2559
7.4 Serial Communications Interface with FIFO (SCIF) 2560
7.4.1 Overview 2560
7.4.2 Registers 2562

7.4.2.1 List of Registers 2562
7.4.2.2 Register Description 2563
7.4.2.2.1 Receive Shift Register (SCIFm_RSR) 2563
7.4.2.2.2 Transmit Shift Register (SCIFm_TSR) 2563
7.4.2.2.3 Serial Mode Register (SCIFm_SMR) 2564
7.4.2.2.4 Bit Rate Register (SCIFm_BRR) 2566
7.4.2.2.5 Serial Control Register (SCIFm_SCR) 2569
7.4.2.2.6 Modulation Duty Register (SCIFm_MDDR) 2571
7.4.2.2.7 Transmit FIFO Data Register (SCIFm_FTDR) 2573
7.4.2.2.8 Serial Status Register (SCIFm_FSR) 2574
7.4.2.2.9 Receive FIFO Data Register (SCIFm_FRDR) 2578
7.4.2.2.10 FIFO Control Register (SCIFm_FCR) 2579
7.4.2.2.11 FIFO Data Count Register (SCIFm_FDR) 2581
7.4.2.2.12 Serial Port Register (SCIFm_SPTR) 2582
7.4.2.2.13 Line Status Register (SCIFm_LSR) 2583
7.4.2.2.14 Serial Extended Mode Register (SCIFm_SEMR) 2584
7.4.2.2.15 FIFO Trigger Control Register (SCIFm_FTCR) 2586
7.4.3 Operation 2587
7.4.3.1 Overview 2587
7.4.3.2 Asynchronous Serial Communication 2588
7.4.3.3 Transmit/Receive Formats 2589
7.4.3.4 Transmitting and Receiving Data 2590
7.4.3.4.1 SCIF initialization 2590
7.4.3.4.2 Transmitting serial data 2592
7.4.3.4.3 Receiving serial data 2594
7.4.4 Bit Rate Modulation 2598
7.4.5 Interrupt Sources 2599
7.4.6 Event Link Output 2600
7.4.7 Serial Port Register (SPTR) and SCIF-Related Pins 2600
7.4.8 Noise Cancellation 2601
7.4.9 Usage Notes 2602
7.4.9.1 FTDR Register Writing and TDFE Flag 2602
7.4.9.2 FRDR Register Reading and RDF Flag 2602
7.4.9.3 Break Detection and Processing 2602
7.4.9.4 Writing to the SPTR Register 2602
7.4.9.5 Break Signal Transmission 2603
7.4.9.6 Receive Data Sampling Timing and Receive Margin 2604
7.4.9.7 Note on FER Flag and PER Flag in Serial Status Register (FSR) 2605
7.5 Serial Peripheral Interface (RSPI) 2606
7.5.1 Overview 2606
7.5.1.1 Features 2606
7.5.2 Registers 2610
7.5.2.1 List of Registers 2611
7.5.2.2 Register Description 2613
7.5.2.2.1 SPI Data Register (RSPIm_SPDR/SPDR_HA/SPDR_BY) 2613
7.5.2.2.2 SPI Clock Delay Register (RSPIm_SPCKD) 2616

7.5.2.2.3 SPI Slave Select Negation Delay Register (RSPIm_SSLND) 2617
7.5.2.2.4 SPI Next-Access Delay Register (RSPIm_SPND)..... 2619
7.5.2.2.5 SPI Clock Digital Control Register for Master Receive (RSPIm_MRCKD)2620
7.5.2.2.6 SPI Control Register (RSPIm_SPCR)..... 2621
7.5.2.2.7 SPI Control Register for Master Receive Only (RSPIm_SPCRRM) 2627
7.5.2.2.8 SPI Control Register for Received Data Ready Detection (RSPIm_SPDRCR) 2628
7.5.2.2.9 SPI Pin Control Register (RSPIm_SPPCR) 2629
7.5.2.2.10 SPI Control Register 2 (RSPIm_SPCR2)..... 2630
7.5.2.2.11 SPI Slave Select Polarity Register (RSPIm_SSLP) 2631
7.5.2.2.12 SPI Bit Rate Register (RSPIm_SPBR)..... 2632
7.5.2.2.13 SPI Sequence Control Register (RSPIm_SPSCR) 2633
7.5.2.2.14 SPI Command Register n (RSPIm_SPCMDn) (n = 0 to 7) 2634
7.5.2.2.15 SPI Data Control Register (RSPIm_SPDCR) 2637
7.5.2.2.16 SPI Data Control Register 2 (RSPIm_SPDCR2) 2639
7.5.2.2.17 SPI Sequence Status Register (RSPIm_SPSSR)..... 2640
7.5.2.2.18 SPI Status Register (RSPIm_SPSR) 2641
7.5.2.2.19 SPI Transfer FIFO Status Register (RSPIm_SPTFSR) 2647
7.5.2.2.20 SPI Receive FIFO Status Register (RSPIm_SPRFSR) 2648
7.5.2.2.21 SPI Polling Register (RSPIm_SPPSR) 2649
7.5.2.2.22 SPI Status Clear Register (RSPIm_SPSRC) 2650
7.5.2.2.23 SPI FIFO Clear Register (RSPIm_SPFCR) 2651
7.5.3 Operation 2652
7.5.3.1 Overview of SPI Operations 2652
7.5.3.2 SPI Pin Control 2653
7.5.3.3 SPI System Configuration Examples..... 2655
7.5.3.3.1 Master/single slave (this LSI = master) 2655
7.5.3.3.2 Master/single slave (this LSI = slave)..... 2655
7.5.3.3.3 Master/multi-slave (this LSI = master)..... 2656
7.5.3.3.4 Master/multi-slave (with this LSI acting as slave) 2657
7.5.3.3.5 Master (clock synchronous operation)/slave (clock synchronous operation) (this LSI = master) 2658
7.5.3.3.6 Master (clock synchronous operation)/slave (clock synchronous operation) (this LSI = slave)..... 2659
7.5.3.4 Data Format..... 2660
7.5.3.4.1 Data format of one frame 2660
7.5.3.4.2 When parity is disabled (SPPE = 0b) 2660
7.5.3.4.3 When parity is enabled (SPPE = 1b)..... 2664
7.5.3.4.4 Byte swap transmission..... 2668
7.5.3.4.5 Byte swap reception 2671
7.5.3.5 Transfer Format (Frame Format)..... 2674
7.5.3.5.1 CPHA = 0b..... 2674
7.5.3.5.2 CPHA = 1b..... 2675
7.5.3.6 Communications Operating Mode 2676

7.5.3.6.1 Transmit-receive serial communication (TXMD[1:0] = 00b) 2676
7.5.3.6.2 Transmit-only serial communication (TXMD[1:0] = 01b) 2676
7.5.3.6.3 Receive-only serial communication (TXMD[1:0] = 10b) 2677
7.5.3.7 Transmission buffer empty/receive buffer full interrupt 2678
7.5.3.8 Idle Interrupt 2680
7.5.3.8.1 In the Motorola-SPI case 2680
7.5.3.8.2 In the TI-SSP case 2681
7.5.3.9 Communication End Interrupt 2681
7.5.3.9.1 Transmit-receive/transmit-only in master mode 2681
7.5.3.9.2 Receive-only in master mode 2683
7.5.3.9.3 Transmit-receive/transmit-only in slave mode on SPI operation (4-wire) 2685
7.5.3.9.4 Receive only in slave mode on SPI operation (4-wire) 2687
7.5.3.9.5 Transmit-receive/transmit-only in slave mode on clock synchronous operation (3-wire) 2689
7.5.3.9.6 Receive only in slave mode on clock synchronous operation (3-wire) 2689
7.5.3.9.7 Common operation 2690
7.5.3.10 Error Detection 2691
7.5.3.10.1 Overrun error 2692
7.5.3.10.2 Parity error 2696
7.5.3.10.3 Mode fault error 2697
7.5.3.10.4 Underrun error 2697
7.5.3.11 Received Data Ready Detection 2698
7.5.3.12 SPI Initialization 2698
7.5.3.12.1 Initialization by clearing the SPE bit 2698
7.5.3.12.2 System reset 2699
7.5.3.13 SPI Operation 2699
7.5.3.13.1 Master mode operation 2699
7.5.3.13.2 Slave mode operation 2715
7.5.3.14 Clock Synchronous Operation 2721
7.5.3.14.1 Master mode operation 2722
7.5.3.14.2 Slave mode operation 2726
7.5.3.15 Loopback Mode 2728
7.5.3.16 Self-Diagnosis of Parity Function 2728
7.5.3.17 Interrupt Requests 2729
7.5.4 Usage Notes 2731
7.5.4.1 Notes on Communication Start 2731
7.5.4.2 Notes on the Module-Stop Function 2731
7.5.4.3 Notes on SPRF and SPTEF Flags 2731
7.5.4.4 Notes on Receive Sampling Function 2731
7.5.4.5 Notes on Burst Transfer in Master Mode 2731
7.5.4.6 Notes on TI-SSP in Slave Mode 2731
7.5.4.7 Notes at SPE = 1b 2732
7.6 CRC Operation Unit (CRC) 2734
7.6.1 Overview 2734
7.6.2 CRC Registers 2736
7.6.2.1 List of Registers 2736
7.6.2.2 CRC Register Descriptions 2737

7.6.2.2.1 CRC Control Register 0 (CRCCR0) ... 2737
7.6.2.2.2 CRC Control Register 1 (CRCCR1) ... 2738
7.6.2.2.3 CRC Data Input Register (CRCDIR) ... 2739
7.6.2.2.4 CRC Data Output Register (CRCDOR) ... 2740
7.6.2.2.5 Snoop Address Register (CRCSAR) ... 2741
7.6.3 Operation ... 2742
7.6.3.1 Basic Operation ... 2742
7.6.3.2 CRC Snoop ... 2746
7.6.4 Usage Notes ... 2747
7.6.4.1 Note on Transmission ... 2747
7.7 I²C Bus Interface (RIIC) ... 2748
7.7.1 Overview ... 2748
7.7.2 RIIC Registers ... 2753
7.7.2.1 List of Registers ... 2753
7.7.2.2 RIIC Register Descriptions ... 2754
7.7.2.2.1 I²C Bus Control Register 1 (RIICm_ICCR1) ... 2754
7.7.2.2.2 I²C Bus Control Register 2 (RIICm_ICCR2) ... 2757
7.7.2.2.3 I²C Bus Mode Register 1 (RIICm_ICMR1) ... 2761
7.7.2.2.4 I²C Bus Mode Register 2 (RIICm_ICMR2) ... 2762
7.7.2.2.5 I²C Bus Mode Register 3 (RIICm_ICMR3) ... 2764
7.7.2.2.6 I²C Bus Function Enable Register (RIICm_ICFER) ... 2766
7.7.2.2.7 I²C Bus Status Enable Register (RIICm_ICSER) ... 2768
7.7.2.2.8 I²C Bus Interrupt Enable Register (RIICm_ICIER) ... 2770
7.7.2.2.9 I²C Bus Status Register 1 (RIICm_ICSR1) ... 2772
7.7.2.2.10 I²C Bus Status Register 2 (RIICm_ICSR2) ... 2775
7.7.2.2.11 Slave Address Register L y (RIICm_SARLy) (y = 0 to 2) ... 2779
7.7.2.2.12 Slave Address Register U y (RIICm_SARUy) (y = 0 to 2) ... 2780
7.7.2.2.13 I²C Bus Bit Rate Low-Level Register (RIICm_ICBRL) ... 2781
7.7.2.2.14 I²C Bus Bit Rate High-Level Register (RIICm_ICBRH) ... 2782
7.7.2.2.15 I²C Bus Transmit Data Register (RIICm_ICDRT) ... 2784
7.7.2.2.16 I²C Bus Receive Data Register (RIICm_ICDRR) ... 2785
7.7.2.2.17 I²C Bus Shift Register (RIICm_ICDRS) ... 2786
7.7.3 Operation ... 2787
7.7.3.1 Communication Data Format ... 2787
7.7.3.2 Initial Settings ... 2789
7.7.3.3 Master Transmit Operation ... 2790
7.7.3.4 Master Receive Operation ... 2794
7.7.3.5 Slave Transmit Operation ... 2800
7.7.3.6 Slave Receive Operation ... 2803
7.7.4 SCL Synchronization Circuit ... 2806
7.7.5 SDA Output Delay Function ... 2808
7.7.6 Digital Noise Filter Circuits ... 2809
7.7.7 Address Match Detection ... 2810
7.7.7.1 Slave-Address Match Detection ... 2810
7.7.7.2 Detection of General Call Address ... 2813
7.7.7.3 Device ID Address Detection ... 2814

7.7.7.4 Host Address Detection 2816
7.7.8 Automatic Low-Hold Function for SCL 2817
7.7.8.1 Function to Prevent Wrong Transmission of Transmit Data 2817
7.7.8.2 NACK Reception Transfer Suspension Function 2818
7.7.8.3 Function to Prevent Failure to Receive Data 2819
7.7.8.3.1 1-byte receive operation and automatic low-hold function using the WAIT bit 2819
7.7.8.3.2 1-byte receive operation (ACK/NACK transmission control) and automatic low-hold function using the RDRFS bit 2819
7.7.9 Arbitration-Lost Detection Functions 2821
7.7.9.1 Master Arbitration-Lost Detection (MALE Bit) 2821
7.7.9.2 Function to Detect Loss of Arbitration during NACK Transmission (NALE Bit) 2824
7.7.9.3 Slave Arbitration-Lost Detection (SALE Bit) 2826
7.7.10 Start, Restart, and Stop Condition Issuing Function 2827
7.7.10.1 Issuing a Start Condition 2827
7.7.10.2 Issuing a Restart Condition 2827
7.7.10.3 Issuing a Stop Condition 2830
7.7.11 Bus Hanging 2831
7.7.11.1 Timeout Function 2831
7.7.11.2 Extra SCL Clock Cycle Output Function 2833
7.7.11.3 IIC Reset and Internal Reset 2834
7.7.12 SMBus Operation 2835
7.7.12.1 SMBus Timeout Measurement 2835
7.7.12.1.1 Measuring slave device timeout 2835
7.7.12.1.2 Measuring master device timeout 2835
7.7.12.2 Packet Error Code (PEC) 2836
7.7.12.3 SMBus Host Notification Protocol (Notify ARP Master Command) 2837
7.7.13 Interrupt Sources 2838
7.7.14 Even Link Output 2839
7.7.15 Resets, Registers, and Function States when Issuing Each Condition 2840
7.8 I3C Bus Interface (I3C) 2841
7.8.1 Overview 2841
7.8.1.1 Functional Overview 2841
7.8.2 Block Diagram [I^2C/I3C common] 2844
7.8.3 Registers 2845
7.8.3.1 List of Registers 2845
7.8.3.2 Register Description 2849
7.8.3.2.1 Protocol Selection Register (I3Cm_PRTS) 2849
7.8.3.2.2 Bus Control Register (I3Cm_BCTL) 2850
7.8.3.2.3 Master Device Address Register (I3Cm_MSDVAD) 2852
7.8.3.2.4 Reset Control Register (I3Cm_RSTCTL) 2853
7.8.3.2.5 Present State Register (I3Cm_PRSST) 2855
7.8.3.2.6 Internal Status Register (I3Cm_INST) 2858
7.8.3.2.7 Internal Status Enable Register (I3Cm_INSTE) 2860
7.8.3.2.8 Internal Interrupt Enable Register (I3Cm_INIE) 2861
7.8.3.2.9 Internal Status Force Register (I3Cm_INSTFC) 2862
7.8.3.2.10 Device Characteristic Table Register (I3Cm_DVCT) 2863

7.8.3.2.11 IBI Notify Control Register (I3Cm_IBINCTL) 2864
7.8.3.2.12 Bus Function Control Register (I3Cm_BFCTL) 2865
7.8.3.2.13 Slave Control Register (I3Cm_SVCTL) 2867
7.8.3.2.14 Reference Clock Control Register (I3Cm_REFCKCTL) 2869
7.8.3.2.15 Standard Bit Rate Register (I3Cm_STDBR) 2870
7.8.3.2.16 Extended Bit Rate Register (I3Cm_EXTBR) 2873
7.8.3.2.17 Bus Free Condition Detection Time Register (I3Cm_BFRECDT) 2875
7.8.3.2.18 Bus Available Condition Detection Time Register (I3Cm_BAVLCDT) 2876
7.8.3.2.19 Bus Idle Condition Detection Time Register (I3Cm_BIDLCDT) 2877
7.8.3.2.20 Output Control Register (I3Cm_OUTCTL) 2878
7.8.3.2.21 Input Control Register (I3Cm_INCTL) 2880
7.8.3.2.22 Timeout Control Register (I3Cm_TMOCTL) 2881
7.8.3.2.23 Wake-Up Unit Control Register (I3Cm_WUCTL) 2883
7.8.3.2.24 Acknowledge Control Register (I3Cm_ACKCTL) 2885
7.8.3.2.25 SCL Stretch Control Register (I3Cm_SCSTRCTL) 2887
7.8.3.2.26 SCL Stalling Control Register (I3Cm_SCSTLCTL) 2889
7.8.3.2.27 Slave Transfer Data Length Register (I3Cm_SVTDLG0) 2891
7.8.3.2.28 Synchronous Timing Control Register (I3Cm_STCTL) 2892
7.8.3.2.29 Asynchronous Timing Control Register (I3Cm_ATCTL) 2893
7.8.3.2.30 Asynchronous Timing Trigger Register (I3Cm_ATTRG) 2894
7.8.3.2.31 Asynchronous Timing Control Counter Enable Register (I3Cm_ATCCNTE) 2895
7.8.3.2.32 Condition Control Register (I3Cm_CNDCTL) 2896
7.8.3.2.33 Normal Command Queue Port Register (I3Cm_NCMDQP) 2898
7.8.3.2.34 Normal Response Queue Port Register (I3Cm_NRSPQP) 2899
7.8.3.2.35 Normal Transfer Data Buffer Port Register (I3Cm_NTDTBP0/NTDTBP0_BY) 2900
7.8.3.2.36 Normal IBI Queue Port Register (I3Cm_NIBIQP) 2902
7.8.3.2.37 Normal Receive Status Queue Port Register (I3Cm_NRSQP) 2903
7.8.3.2.38 Normal Queue Threshold Control Register (I3Cm_NQTHCTL) 2904
7.8.3.2.39 Normal Transfer Data Buffer Threshold Control Register (I3Cm_NTBTHCTL0) 2906
7.8.3.2.40 Normal Receive Status Queue Threshold Control Register (I3Cm_NRQTHCTL) 2908
7.8.3.2.41 Bus Status Register (I3Cm_BST) 2909
7.8.3.2.42 Bus Status Enable Register (I3Cm_BSTE) 2913
7.8.3.2.43 Bus Interrupt Enable Register (I3Cm_BIE) 2915
7.8.3.2.44 Bus Status Force Register (I3Cm_BSTFC) 2917
7.8.3.2.45 Normal Transfer Status Register (I3Cm_NTST) 2918
7.8.3.2.46 Normal Transfer Status Enable Register (I3Cm_NTSTE) 2924
7.8.3.2.47 Normal Transfer Interrupt Enable Register (I3Cm_NTIE) 2926
7.8.3.2.48 Normal Transfer Status Force Register (I3Cm_NTSTFC) 2928
7.8.3.2.49 Bus Condition Status Register (I3Cm_BCST) 2930

7.8.3.2.50 Slave Status Register (I3Cm_SVST) 2932
7.8.3.2.51 Wake-Up Unit Operating Status Register (I3Cm_WUST) 2936
7.8.3.2.52 MsyncCNT Counter Capture Register (I3Cm_MRCCPT) 2938
7.8.3.2.53 Device Address Table Basic Register n (I3Cm_DATBASn) (n = 0 to 7) 2939
7.8.3.2.54 Slave Device Address Table Basic Register n (I3Cm_SDATBASn) (n = 0 to 2) 2941
7.8.3.2.55 Master Device Characteristic Table Register n (I3Cm_MSDCTn) (n = 0 to 7) 2943
7.8.3.2.56 Extended Device Address Table Basic Register (I3Cm_EXDATBAS) 2945
7.8.3.2.57 Slave Device Characteristic Table Register (I3Cm_SVDCT) 2946
7.8.3.2.58 Slave Device Characteristic Table Provisional ID Low Register (I3Cm_SDCTPIDL) 2948
7.8.3.2.59 Slave Device Characteristic Table Provisional ID High Register (I3Cm_SDCTPIDH) 2949
7.8.3.2.60 Slave Device Address Register n (I3Cm_SVDVADn) (n = 0 to 2) 2950
7.8.3.2.61 CCC Slave Events Command Register (I3Cm_CSECMD) 2952
7.8.3.2.62 CCC Enter Activity State Register (I3Cm_CEACTST) 2954
7.8.3.2.63 CCC Max Write Length Register (I3Cm_CMWLG) 2955
7.8.3.2.64 CCC Max Read Length Register (I3Cm_CMRLG) 2956
7.8.3.2.65 CCC Enter Test Mode Register (I3Cm_CETSTMD) 2958
7.8.3.2.66 CCC Get Device Status Register (I3Cm_CGDVST) 2959
7.8.3.2.67 CCC Max Data Speed W (Write) Register (I3Cm_CMDSPW) 2961
7.8.3.2.68 CCC Max Data Speed R (Read) Register (I3Cm_CMDSPR) 2962
7.8.3.2.69 CCC Max Data Speed T (Turnaround) Register (I3Cm_CMDSPT) 2963
7.8.3.2.70 CCC Exchange Timing Support Information M (Mode) Register (I3Cm_CETSM) 2964
7.8.3.2.71 CCC Exchange Timing Support Information S (State) Register (I3Cm_CETSS) 2966
7.8.3.2.72 Bit Count Register (I3Cm_BITCNT) 2968
7.8.3.2.73 Normal Queue Status Level Register (I3Cm_NQSTLV) 2969
7.8.3.2.74 Normal Data Buffer Status Level Register (I3Cm_NDBSTLV0) 2970
7.8.3.2.75 Normal Receive Status Queue Status Level Register (I3Cm_NRSQSTLV) 2971
7.8.3.2.76 Present State Debug Register (I3Cm_PRSTDBG) 2972
7.8.3.2.77 Master Error Counters Register (I3Cm_MSERRCNT) 2973
7.8.3.2.78 SC1 Capture monitor Register (I3Cm_SC1CPT) 2974
7.8.3.2.79 SC2 Capture monitor Register (I3Cm_SC2CPT) 2975
7.8.4 Operation 2976
7.8.4.1 Data Structures 2976
7.8.4.1.1 Command Descriptor 2976
7.8.4.1.2 Response Descriptor 2991
7.8.4.1.3 IBI Status Descriptor 2994
7.8.4.1.4 Receive Status Descriptor 2996

7.8.4.2 Details of Function 2998
7.8.4.2.1 Operation Mode 2998
7.8.4.2.2 Data Handler 3031
7.8.4.2.3 I²C/I3C Protocol 3034
7.8.4.2.4 Error Detection 3088
7.8.4.2.5 Low Power Function [I²C/I3C common] 3095
7.8.4.2.6 Other 3107
7.8.4.3 Operation 3111
7.8.4.3.1 Initial Setting Flow 3111
7.8.4.3.2 I3C Communication Flow 3114
7.8.4.3.3 Master Mode Communication Flow 3116
7.8.4.3.4 Slave Mode Communication Flow 3123
7.8.5 Interrupt Sources 3130
7.8.5.1 Overview 3130
7.8.6 Event Link Output 3131
7.8.6.1 Interrupt Handling and Event Linking 3132
7.8.7 Reset Descriptions 3133
7.9 CAN-FD Interface (CANFD) 3143
7.9.1 Overview 3143
7.9.2 CANFD Registers 3147
7.9.2.1 List of Registers 3147
7.9.2.2 CANFD Register Description 3156
7.9.2.2.1 Channel n Nominal Bit Rate Configuration Register (CFDCnNCFG) (n = 0 to 5) 3156
7.9.2.2.2 Channel n Control Register (CFDCnCTR) (n = 0 to 5) 3158
7.9.2.2.3 Channel n Status Register (CFDCnSTS) (n = 0 to 5) 3164
7.9.2.2.4 Channel n Error Flag Register (CFDCnERFL) (n = 0 to 5) 3167
7.9.2.2.5 Channel n Data Bit Rate Configuration Register (CFDCnDCFG) (n = 0 to 5) 3173
7.9.2.2.6 Channel n CAN-FD Configuration Register (CFDCnFDCFG) (n = 0 to 5) 3175
7.9.2.2.7 Channel n CAN-FD Control Register (CFDCnFDCTR) (n = 0 to 5) 3180
7.9.2.2.8 Channel n CAN-FD Status Register (CFDCnFDSTS) (n = 0 to 5) 3181
7.9.2.2.9 Channel n CAN-FD CRC Register (CFDCnFDCRC) (n = 0 to 5) 3184
7.9.2.2.10 Global IP Version Register (CFDGIPV) 3185
7.9.2.2.11 Global Configuration Register (CFDGCFG) 3186
7.9.2.2.12 Global Control Register (CFDGCTR) 3189
7.9.2.2.13 Global Status Register (CFDGSTS) 3192
7.9.2.2.14 Global Error Flag Register (CFDGERFL) 3194
7.9.2.2.15 Global TX Interrupt Status Register 0 (CFDGTINTSTS0) 3197
7.9.2.2.16 Global TX Interrupt Status Register 1 (CFDGTINTSTS1) 3200
7.9.2.2.17 Global Timestamp Counter Register (CFDGTSC) 3203
7.9.2.2.18 Global Acceptance Filter List Entry Control Register (CFDGAFLECTR) .. 3204
7.9.2.2.19 Global Acceptance Filter List Configuration Register w (CFDGAFLCFGw) (w = 0, 1, 2) 3205

7.9.2.2.20 Global Acceptance Filter List ID Register n (CFDGAFLIDn) (n = 1 to 16) 3206
7.9.2.2.21 Global Acceptance Filter List Mask Register n (CFDGAFLMn) (n = 1 to 16) 3208
7.9.2.2.22 Global Acceptance Filter List Pointer 0 Register n (CFDGAFLP0n) (n = 1 to 16) 3210
7.9.2.2.23 Global Acceptance Filter List Pointer 1 Register n (CFDGAFLP1n) (n = 1 to 16) 3213
7.9.2.2.24 RX Message Buffer Number Register (CFDRMNB) 3216
7.9.2.2.25 RX Message Buffer New Data Register t (CFDRMNDt) (t = 0 to 2)..... 3217
7.9.2.2.26 RX FIFO Configuration/Control Register n (CFDRFCCn) (n = 0 to 7)..... 3218
7.9.2.2.27 RX FIFO Status Register n (CFDRFSTSn) (n = 0 to 7)..... 3221
7.9.2.2.28 RX FIFO Pointer Control Register n (CFDRFPCTRn) (n = 0 to 7) 3223
7.9.2.2.29 Common FIFO Configuration/Control Register n (CFDCFCCn) (n = 0 to 17) 3224
7.9.2.2.30 Common FIFO Configuration/Control Enhancement Register n (CFDCFCCEn) (n = 0 to 17)..... 3228
7.9.2.2.31 Common FIFO Status Register n (CFDCFSTSn) (n = 0 to 17) 3230
7.9.2.2.32 Common FIFO Pointer Control Register n (CFDCFPCTRn) (n = 0 to 17) 3234
7.9.2.2.33 FIFO Empty Status Register (CFDFESTS)..... 3235
7.9.2.2.34 FIFO Full Status Register (CFDFFSTS) 3236
7.9.2.2.35 FIFO Message Lost Status Register (CFDFMSTS)..... 3237
7.9.2.2.36 RX FIFO Interrupt Flag Status Register (CFDRFISTS) 3238
7.9.2.2.37 Common FIFO RX Interrupt Flag Status Register (CFDCFRISTS)..... 3239
7.9.2.2.38 Common FIFO TX Interrupt Flag Status Register (CFDCFTISTS)..... 3240
7.9.2.2.39 FIFO FDC Full Status Register (CFDFFFSTS)..... 3241
7.9.2.2.40 Common FIFO Message Overwrite Status Register (CFDCFMOWSTS). 3242
7.9.2.2.41 Common FIFO One Frame RX Interrupt Flag Status Register (CFDCFOFRISTS) 3243
7.9.2.2.42 Common FIFO One Frame TX Interrupt Flag Status Register (CFDCFOFTISTS)..... 3244
7.9.2.2.43 DMA Transfer Control Register (CFDCDTCT) 3245
7.9.2.2.44 DMA Transfer Status Register (CFDCDTSTS) 3247
7.9.2.2.45 DMA TX Transfer Control Register (CFDCDTTCT) 3249
7.9.2.2.46 DMA TX Transfer Status Register (CFDCDTTSTS)..... 3251
7.9.2.2.47 Global RX Interrupt Status Register n (CFDGRINTSTSn) (n = 0 to 5)..... 3253
7.9.2.2.48 TX Message Buffer Control Register n (CFDTMCn) (n = 0 to 383)..... 3255
7.9.2.2.49 TX Message Buffer Status Register n (CFDTMSTSn) (n = 0 to 383)..... 3257
7.9.2.2.50 TX Message Buffer Transmission Request Status Register f (CFDTMTRSTSf) (f = 0 to 11) 3259
7.9.2.2.51 TX Message Buffer Transmission Abort Request Status Register f (CFDTMTARSTSf) (f = 0 to 11) 3261
7.9.2.2.52 TX Message Buffer Transmission Completion Status Register f (CFDTMTCSTSf) (f = 0 to 11) 3263

7.9.2.2.53 TX Message Buffer Transmission Abort Status Register f (CFDTMTASTSf) (f = 0 to 11)..... 3265
7.9.2.2.54 TX Message Buffer Transmission Interrupt Enable Register f (CFDTMIECf) (f = 0 to 11)..... 3267
7.9.2.2.55 TX Queue Configuration/Control Register 0n (CFDTXQCC0n) (n = 0 to 5) 3269
7.9.2.2.56 TX Queue Configuration/Control Register 1n (CFDTXQCC1n) (n = 0 to 5) 3272
7.9.2.2.57 TX Queue Configuration/Control Register 2n (CFDTXQCC2n) (n = 0 to 5) 3275
7.9.2.2.58 TX Queue Configuration/Control Register 3n (CFDTXQCC3n) (n = 0 to 5) 3278
7.9.2.2.59 TX Queue Status Register 0n (CFDTXQSTS0n) (n = 0 to 5) 3281
7.9.2.2.60 TX Queue Status Register 1n (CFDTXQSTS1n) (n = 0 to 5) 3284
7.9.2.2.61 TX Queue Status Register 2n (CFDTXQSTS2n) (n = 0to 5) 3287
7.9.2.2.62 TX Queue Status Register 3n (CFDTXQSTS3n) (n = 0 to 5) 3290
7.9.2.2.63 TX Queue Pointer Control Register 0n (CFDTXQPCTR0n) (n = 0 to 5)... 3293
7.9.2.2.64 TX Queue Pointer Control Register 1n (CFDTXQPCTR1n) (n = 0 to 5)... 3294
7.9.2.2.65 TX Queue Pointer Control Register 2n (CFDTXQPCTR2n) (n = 0 to 5)... 3295
7.9.2.2.66 TX Queue Pointer Control Register 3n (CFDTXQPCTR3n) (n = 0 to 5)... 3296
7.9.2.2.67 TX Queue Empty Status Register (CFDTXQESTS) 3297
7.9.2.2.68 TX Queue Full Interrupt Status Register (CFDTXQFISTS) 3299
7.9.2.2.69 TX Queue Message Lost Status Register (CFDTXQMSTS) 3301
7.9.2.2.70 TX Queue Interrupt Status Register (CFDTXQISTS) 3303
7.9.2.2.71 TX Queue One Frame TX Interrupt Status Register (CFDTXQOFTISTS) 3305
7.9.2.2.72 TX Queue One Frame RX Interrupt Status Register (CFDTXQOFRISTS)3307
7.9.2.2.73 TX Queue Full Status Register (CFDTXQFSTS)..... 3309
7.9.2.2.74 TX History List Configuration/Control Register n (CFDTHLCCn) (n = 0 to 5) 3311
7.9.2.2.75 TX History List Status Register n (CFDTHLSTSn) (n = 0 to 5)..... 3313
7.9.2.2.76 Channel n TX History List Access Register 0 (CFDTHLACC0n) (n = 0 to 5) 3315
7.9.2.2.77 Channel n TX History List Access Register 1 (CFDTHLACC1n) (n = 0 to 5) 3316
7.9.2.2.78 TX History List Pointer Control Register n (CFDTHLPCTRn) (n = 0 to 5) 3317
7.9.2.2.79 Global Reset Control Register (CFDGRSTC)..... 3318
7.9.2.2.80 Global Flexible CAN Mode Configuration Register (CFDGFCMC)..... 3319
7.9.2.2.81 Global Flexible Transmission Buffer Assignment Configuration Register (CFDGFTBAC) 3320
7.9.2.2.82 Global Test Configuration Register (CFDGTSTCFG) 3322
7.9.2.2.83 Global Test Control Register (CFDGTSTCTR) 3323
7.9.2.2.84 Global FD Configuration Register (CFDGFDCFG) 3324
7.9.2.2.85 Global Lock Key Register (CFDGLOCKK)..... 3325

7.9.2.2.86 RAM Test Page Access Register n (CFDRPGACCn) (n = 0 to 63) 3326
7.9.2.2.87 Channel n Bus Load Control Register (CFDCnBLCT) (n = 0 to 5) 3327
7.9.2.2.88 Channel n Bus Load Status Register (CFDCnBLSTS) (n = 0 to 5) 3328
7.9.2.2.89 Identifier Bits Alignment 3329
7.9.2.3 Message Buffer Component Structure 3330
7.9.2.3.1 Start Addresses 3330
7.9.3 Operation 3354
7.9.3.1 Overview 3354
7.9.3.2 Global Modes 3354
7.9.3.2.1 Global Sleep Mode 3355
7.9.3.2.2 Global Reset Mode 3357
7.9.3.2.3 Global Halt Mode 3359
7.9.3.2.4 Global Operation Mode 3361
7.9.3.3 Channel Modes 3363
7.9.3.3.1 CAN Channel Sleep Mode 3363
7.9.3.3.2 CAN Channel Reset Mode 3363
7.9.3.3.3 CAN Channel Halt Mode 3365
7.9.3.3.4 CAN Channel Operation Mode 3365
7.9.3.3.5 CAN Channel Bus-Off State 3366
7.9.3.4 Global Mode and Channel Mode Transition Interactions 3369
7.9.3.4.1 Timing of Global Mode Change 3370
7.9.3.4.2 Timing of Channel Mode Change 3370
7.9.4 Initialization 3371
7.9.4.1 Initialization of CAN Clock, Bit Timing and Baud Rate 3371
7.9.4.1.1 Bit Timing Conditions 3371
7.9.4.1.2 CAN Bit Timing 3372
7.9.4.1.3 Baud Rate 3373
7.9.4.1.4 Setting of CAN Clock, Bit Timing and Baud Rate 3375
7.9.4.1.5 Transmitter Delay Compensation 3376
7.9.4.1.6 CAN Module Configuration after Hardware Reset 3379
7.9.5 Acceptance Filtering Function using Global Acceptance Filter List (AFL) 3381
7.9.5.1 Overview 3381
7.9.5.2 Allocation of AFL Entries to Each CAN Channel 3383
7.9.5.3 AFL Entry Description 3384
7.9.5.4 Entering Entries in the AFL 3386
7.9.5.5 Loopback Modes 3390
7.9.5.6 IDE Masking 3391
7.9.6 FIFO Buffers and Normal Message Buffer Configuration 3392
7.9.6.1 Normal RX Message Buffers 3394
7.9.6.1.1 Normal RX Message Buffer Configuration 3394
7.9.6.2 FIFO Buffer 3394
7.9.6.2.1 FIFO Buffers Configuration 3395
7.9.6.2.2 FIFO Buffers Control 3399
7.9.7 Interrupt and DMA 3401
7.9.7.1 Interrupts 3401
7.9.7.2 DMA Transfer 3405

7.9.8 Reception and Transmission 3409
7.9.8.1 Reception....3409
7.9.8.1.1 Message Storage in RX Message Buffers 3409
7.9.8.1.2 Message Storage in FIFO Buffers....3411
7.9.8.1.3 Timestamp 3414
7.9.8.2 Transmission 3415
7.9.8.2.1 Transmission Priority 3416
7.9.8.2.2 Normal Transmission.... 3418
7.9.8.2.3 TX FIFO or GW FIFO Transmission.... 3421
7.9.8.2.4 TX Queue 3427
7.9.8.2.5 TX History List 3432
7.9.8.2.6 TX Data Padding 3436
7.9.9 Test Mode 3437
7.9.9.1 Channel Specific Test Modes....3437
7.9.9.1.1 Basic Test Mode 3437
7.9.9.1.2 Listen-Only Mode 3437
7.9.9.1.3 Self-Test Mode 0 (External Loop Back Mode).... 3438
7.9.9.1.4 Self-Test Mode 1 (Internal Loop Back Mode).... 3439
7.9.9.1.5 Restricted Operation Mode 3439
7.9.9.2 Global Test Modes....3439
7.9.9.2.1 RAM Test Mode 3441
7.9.9.2.2 Internal CAN Bus Communication Mode 3443
7.9.9.2.3 CRC Error Test 3444
7.9.10 Bus Traffic Measurement....3445
7.9.10.1 How to Count the CAN Bus Idle Time 3445
7.9.10.2 Operations and Measurement Procedure 3445
7.9.11 Flexible CAN mode 3447
7.9.12 Flexible Transmission Buffer Assignment....3448
7.10 12-Bit A/D Converter (ADC) 3450
7.10.1 Overview 3450
7.10.2 ADC Registers 3455
7.10.2.1 List of Registers....3455
7.10.2.2 ADC Register Descriptions....3457
7.10.2.2.1 A/D Data Registers.... 3457
7.10.2.2.2 A/D Control Register (ADCm_ADCSR).... 3461
7.10.2.2.3 A/D Conversion Status Register (ADCm_ADREF) 3462
7.10.2.2.4 A/D Channel Select Register (ADCm_ADANSA0).... 3463
7.10.2.2.5 A/D-Converted Value Addition/Average Function Channel Select Register 0 (ADCm_ADADS0) 3464
7.10.2.2.6 A/D-Converted Value Addition/Average Count Select Register (ADCm_ADADC).... 3465
7.10.2.2.7 A/D Control Extended Register (ADCm_ADCER) 3466
7.10.2.2.8 A/D Conversion Start Trigger Select Register (ADCm_ADSTRGR)..... 3467
7.10.2.2.9 A/D Channel Select Register B0 (ADCm_ADANSB0) 3471
7.10.2.2.10 A/D Error Control Register (ADCm_ADERCR).... 3472

7.10.2.2.11 A/D Error Clear Register (ADCm_ADERCLR) 3473
7.10.2.2.12 A/D Overwrite Error Register (ADCm_ADOWER) 3474
7.10.2.2.13 A/D Overwrite Error Extended Register (ADCm_ADOWEER) 3475
7.10.2.2.14 A/D Event Link Control Register (ADCm_ADELCCR) 3476
7.10.2.2.15 A/D Group Scan Priority Control Register (ADCm_ADGSPCR) 3477
7.10.2.2.16 A/D Compare Function Window A/B Status Monitoring Register (ADCm_ADWINMON) 3478
7.10.2.2.17 A/D Compare Function Control Register (ADCm_ADCMPCR) 3479
7.10.2.2.18 A/D Compare Function Window A Channel Select Register 0 (ADCm_ADCMPANSR0) 3480
7.10.2.2.19 A/D Compare Function Window A Comparison Condition Setting Register 0 (ADCm_ADCMPLR0) 3481
7.10.2.2.20 A/D Comparison Function Window A Lower Level Setting Register (ADCm_ADCMPDR0) 3482
7.10.2.2.21 A/D Comparison Function Window A Upper Level Setting Register (ADCm_ADCMPDR1) 3483
7.10.2.2.22 A/D Comparison Function Window A Channel Status Register 0 (ADCm_ADCMPSR0) 3484
7.10.2.2.23 A/D Compare Function Window B Channel Select Register (ADCm_ADCMPBNSR) 3485
7.10.2.2.24 A/D Compare Function Window B Lower-Side Level Setting Register (ADCm_ADWINLLB) 3486
7.10.2.2.25 A/D Compare Function Window B Upper-Side Level Setting Register (ADCm_ADWINULB) 3487
7.10.2.2.26 A/D Compare Function Window B Status Register (ADCm_ADCMPBSR) 3488
7.10.2.2.27 A/D Channel Select Register C0 (ADCm_ADANSC0) 3489
7.10.2.2.28 A/D Group C Trigger Select Register (ADCm_ADGCTRGR) 3490
7.10.3 Operation 3493
7.10.3.1 Scanning Operation 3493
7.10.3.2 Single Scan Mode 3494
7.10.3.2.1 Basic operation 3494
7.10.3.2.2 A/D conversion in double trigger mode 3495
7.10.3.2.3 Extended operations when double trigger mode is selected 3496
7.10.3.3 Continuous Scan Mode 3498
7.10.3.3.1 Basic operation (not use S/H circuits) 3498
7.10.3.4 Group Scan Mode 3499
7.10.3.4.1 Basic operation 3499
7.10.3.4.2 A/D conversion in double trigger mode 3501
7.10.3.4.3 Operation under Group Priority Control 3503
7.10.3.5 Compare Function (Window A, Window B) 3507
7.10.3.5.1 Compare function window A/B 3507
7.10.3.5.2 Event output in compare function 3509
7.10.3.5.3 Compare function constraint 3511

7.10.3.6 A/D Data Register Automatic Clearing Function 3512
7.10.3.7 A/D-Converted Value Addition/Average Function 3512
7.10.3.8 Starting A/D Conversion with External Trigger 3512
7.10.3.9 Starting A/D Conversion with Internal Trigger from Peripheral Module 3513
7.10.3.10 Error Detection Function (Overwrite Check Function) 3513
7.10.3.11 Event Link Function 3514
7.10.3.11.1 Event output to ELC 3514
7.10.3.11.2 Starting A/D conversion with ELC event 3514
7.10.3.11.3 Notes on ELC trigger 3514
7.10.4 Usage Notes 3515
7.10.4.1 Notes on Reading Data Registers 3515
7.10.4.2 Notes on Stopping A/D Conversion 3515
7.10.4.2.1 A/D conversion stop procedure 3515
7.10.4.2.2 Notes on mode/status bits 3516
7.10.4.3 A/D conversion Forced Stop and Operation Timing at Starts 3516
7.10.4.4 Notes on Scan End Interrupt Processing 3516
7.10.4.5 Notes on Using Hardware and Software Triggers Simultaneously. 3516
7.10.4.6 Power-down Mode Settings 3517
7.10.4.7 Notes on Entering Low Power Consumption States 3517
7.11 Temperature Sensor Unit (TSU) 3518
7.11.1 Functional Overview 3518
7.11.2 Connection Configuration 3519
7.11.3 Pin Description 3520
7.11.3.1 List of Internal Pins 3520
7.11.3.2 Internal Pin Functions 3521
7.11.3.2.1 Clock and Reset 3521
7.11.3.2.2 Control Interface 3521
7.11.4 Clock 3522
7.11.4.1 Clock Signals 3522
7.11.5 Interrupts 3523
7.11.5.1 Conversion Complete Interrupt 3523
7.11.5.2 Comparison Result Interrupt 3523
7.11.6 Registers 3525
7.11.6.1 List of Registers 3525
7.11.6.2 Register Descriptions 3526
7.11.6.2.1 Sensor Suspend Register (TSUm_SSUSR) 3526
7.11.6.2.2 Sensor Trigger Register (TSUm_STRGR) 3527
7.11.6.2.3 Sensor Operation Setting 1 Register (TSUm_SOSR1) 3528
7.11.6.2.4 Sensor Code Read Register (TSUm_SCRR) 3529
7.11.6.2.5 Sensor Status Register (TSUm_SSR) 3530
7.11.6.2.6 Compare Mode Setting Register (TSUm_CMSR) 3531
7.11.6.2.7 Lower Limit Setting Register (TSUm_LLSR) 3532
7.11.6.2.8 Upper Limit Setting Register (TSUm_ULSR) 3533
7.11.6.2.9 Sensor Interrupt Status Register (TSUm_SISR) 3534
7.11.6.2.10 Sensor Interrupt Enable Register (TSUm_SIER) 3535
7.11.6.2.11 Sensor Interrupt Clear Register (TSUm_SICR) 3536
7.11.7 Functional Description 3537

7.11.7.1 Acquiring Temperature Code 3537
7.11.7.2 Temperature Code Averaging Function 3537
7.11.7.3 Temperature Code Reading Function 3537
7.11.7.4 Temperature Code Comparison Functions 3538
7.11.7.5 Interrupt Output Function....3539
7.11.7.6 Conversion Start Trigger 3539
7.11.7.7 Single Scan....3539
7.11.7.8 Temperature Compensation Calculation....3540
7.11.7.9 Comparison Setting Value Calculation....3540
7.11.8 Operation Procedure....3541
7.11.8.1 Using Interrupt for Each Completion of Conversion....3541
7.11.8.1.1 Single Scan, Start by Software.... 3541
7.11.8.1.2 Single Scan, Start by an ELC Trigger 3543
7.11.8.2 Using Comparison Result Interrupt 3545
7.11.8.2.1 Single Scan, Start by Software.... 3545
7.11.8.2.2 Single Scan, Start by an ELC Trigger 3547
7.11.8.3 Initialization Procedure 3549
7.11.8.4 Procedure for Changing Operating Mode 3549
7.11.9 Usage Notes 3550
7.11.9.1 Release from Power-down 3550
7.11.9.2 Conversion Start Trigger 3550
7.11.9.3 Restarting the Stopped Clock....3550
SECTION 8 AUDIO 3551
8.1 Audio Overview....3551
8.1.1 Overview 3551
8.1.1.1 Features....3551
8.1.1.2 Block Diagram 3552
8.1.1.3 External Pins....3554
8.1.1.4 Register Configuration....3554
8.1.1.5 Connected Module....3554
8.1.2 Register Description....3555
8.1.3 Operation 3556
8.1.3.1 Audio Module Description....3556
8.1.3.2 List of Routings....3559
8.1.3.3 DMAC 3560
8.1.3.4 Data Format on Audio Local Bus....3562
8.1.3.5 Rearranging the Order of Data 3563
8.1.3.6 Clock and Reset 3566
8.1.3.6.1 Clock and Reset Configuration.... 3566
8.1.3.6.2 System Clock for IP.... 3568
8.1.3.6.3 ADG Divider Input Clocks.... 3574
8.1.3.6.4 ADG Output Audio Master Clocks.... 3574
8.1.3.6.5 Reset for IP.... 3575
8.1.3.6.6 Module Reset for SCU 3578
8.1.3.7 Procedure for Releasing IP from Module Standby....3580
8.1.3.7.1 Procedure for Releasing ADG from Module Standby 3580
8.1.3.7.2 Procedure for Releasing ADMAC from Module Standby 3581
8.1.3.7.3 Procedure for Releasing SCU from Module Standby 3582

8.1.3.7.4 Procedure for Releasing SSIU from Module Standby ... 3583
8.1.3.7.5 Procedure for Releasing SPDIF from Module Standby ... 3585
8.1.3.7.6 Procedure for Releasing PDM from Module Standby ... 3586
8.1.3.8 SSIU/SCU Setup Procedure Overview ... 3587
8.1.3.8.1 Transfer Flow ... 3589
8.1.3.9 SPDIF Setup Procedure Overview ... 3591
8.1.3.10 PDM Setup Procedure Overview ... 3593
8.1.3.11 Module Standby Function ... 3594
8.1.3.11.1 Transition to Module Standby Mode ... 3594
8.1.3.11.2 Releasing and Restarting from Module Standby Mode ... 3596
8.1.4 Usage Notes ... 3597
8.1.4.1 Notes on Route Switching and Setting Changes ... 3597
8.1.4.2 Note when SSI0, SSI1, and SSI2, or SSI0, SSI1, SSI2, and SSI9 are Used for Single-Source Audio Data ... 3597
8.1.4.3 Note on Transfer ... 3597
8.1.4.4 Note on Usage of the SCU Module ... 3597
8.1.4.5 Note on Software Reset During Data Transfer ... 3597
8.1.4.6 Notes when Issuing SCU/SSIU Transfer Instructions ... 3598
8.2 Sampling Rate Converter Unit (SCU) ... 3600
8.2.1 Overview ... 3600
8.2.1.1 Features ... 3600
8.3 Audio Clock Generator (ADG) ... 3602
8.3.1 Overview ... 3602
8.3.1.1 Features ... 3602
8.4 Audio-DMAC-Peripheral-Peripheral (ADMAC) ... 3603
8.4.1 Overview ... 3603
8.4.1.1 Features ... 3603
8.5 Serial Sound Interface Unit (SSIU) ... 3604
8.5.1 Overview ... 3604
8.5.1.1 Features ... 3604
8.6 SPDIF Interface (SPDIF) ... 3606
8.6.1 Overview ... 3606
8.6.1.1 Features ... 3606
8.6.1.2 SPDIF Frame Format ... 3607
8.6.1.3 About Sampling Frequency Selection Method ... 3608
8.6.1.4 Block Diagram ... 3610
8.6.1.5 External Pins ... 3610
8.6.1.6 Connected Units ... 3611
8.6.2 Registers ... 3612
8.6.2.1 List of Registers ... 3612
8.6.3 Register Description ... 3613
8.6.3.1 Transmitter Channel 1 Audio Register (SPDIFm_TLCA) ... 3613
8.6.3.2 Transmitter Channel 2 Audio Register (SPDIFm_TRCA) ... 3614
8.6.3.3 Transmitter Channel 1 Status Register (SPDIFm_TLCS) ... 3615
8.6.3.4 Transmitter Channel 2 Status Register (SPDIFm_TRCS) ... 3616
8.6.3.5 Transmitter User Data Register (SPDIFm_TUI) ... 3617
8.6.3.6 Receiver Channel 1 Audio Register (SPDIFm_RLCA) ... 3618
8.6.3.7 Receiver Channel 2 Audio Register (SPDIFm_RRCA) ... 3619

8.6.3.8 Receiver Channel 1 Status Register (SPDIFm_RLCS) 3620
8.6.3.9 Receiver Channel 2 Status Register (SPDIFm_RRCS) 3621
8.6.3.10 Receiver User Data Register (SPDIFm_RUI) 3622
8.6.3.11 Control Register (SPDIFm_CTRL) 3623
8.6.3.12 Status Register (SPDIFm_STAT) 3626
8.6.3.13 Transmitter DMA Audio Data Register (SPDIFm_TDAD) 3628
8.6.3.14 Receiver DMA Audio Data Register (SPDIFm_RDAD) 3629
8.6.4 Operation 3630
8.6.4.1 Transmitter Control 3630
8.6.4.1.1 Overview of Transmitter Control 3630
8.6.4.1.2 Transmitter Initialization 3631
8.6.4.1.3 Transmitter Initialization Settings 3631
8.6.4.1.4 Data Transfer to the Transmitter 3632
8.6.4.1.5 DMA Data Transfer to the Transmitter 3633
8.6.4.2 Receiver Control 3635
8.6.4.2.1 Overview of Receiver Control 3635
8.6.4.2.2 Receiver Initialization 3636
8.6.4.2.3 Data Transfer from the Receiver 3636
8.6.4.2.4 DMA Data Transfer from the Receiver 3639
8.6.4.3 Stopping Transmitter and Receiver 3641
8.6.4.4 Compressed Mode Data 3641
8.6.4.5 Reset and Clock Control 3642
8.6.5 Interrupt 3643
8.6.5.1 Interrupt List 3643
8.6.6 Usage Notes 3644
8.6.6.1 Clearing TUIR 3644
8.6.6.2 Frequency of Audio Input Clock 3644
8.7 PDM Interface (PDM) 3645
8.7.1 Overview 3645
8.7.1.1 Feature 3645
8.7.1.2 Block Diagram 3646
8.7.1.3 External Pins 3647
8.7.1.4 Connected Unit 3647
8.7.2 Registers 3648
8.7.2.1 List of Registers 3648
8.7.2.2 Register Description 3654
8.7.2.2.1 Channel Software Start Trigger Register (PDMm_PDCSTRTR) (m = 0, 1) 3654
8.7.2.2.2 Channel Software Stop Trigger Register (PDMm_PDCSTPTR) (m = 0, 1) 3655
8.7.2.2.3 Channel Software Change Trigger Register (PDMm_PDCCHGTR) (m = 0, 1) 3656
8.7.2.2.4 Channel Interrupt Control Register (PDMm_PDCICR) (m = 0, 1) 3657
8.7.2.2.5 Channel Status Register (PDMm_PDCSR) (m = 0, 1) 3658
8.7.2.2.6 Channel Status Clear Register (PDMm_PDCSCR) (m = 0, 1) 3660
8.7.2.2.7 Channel Sound Detection Control Register (PDMm_PDCSDCR) (m = 0, 1) 3661

8.7.2.2.8 Channel Data Read Control Register (PDMm_PDCDRCR) (m = 0, 1) 3662
8.7.2.2.9 Channel Data Clear Register (PDMm_PDCDCR) (m = 0, 1)..... 3663
8.7.2.2.10 Version Register (PDMm_PDVR) (m = 0, 1)..... 3664
8.7.2.2.11 Software Start Trigger Register Channel n (PDMm_PDSTRTRCHn) (m = 0 to 1, n = 0 to 2) 3665
8.7.2.2.12 Software Stop Trigger Register Channel n (PDMm_PDSTPTRCHn) (m = 0 to 1, n = 0 to 2) 3666
8.7.2.2.13 Software Change Trigger Register Channel n (PDMm_PDCHGTRCHn) (m = 0 to 1, n = 0 to 2) 3667
8.7.2.2.14 Interrupt Control Register Channel n (PDMm_PDICRCHn) (m = 0 to 1, n = 0 to 2) 3668
8.7.2.2.15 Status Detection Control Register Channel n (PDMm_PDSDCRCHn) (m = 0 to 1, n = 0 to 2) 3669
8.7.2.2.16 Status Register Channel n (PDMm_PDSRCHn) (m = 0 to 1, n = 0 to 2).. 3670
8.7.2.2.17 Status Clear Register Channel n (PDMm_PDSCRCHn) (m = 0 to 1, n = 0 to 2) 3673
8.7.2.2.18 Mode Setting Register Channel n (PDMm_PDMDSRCHn) (m = 0 to 1, n = 0 to 2) 3674
8.7.2.2.19 Sinc Filter Control Register Channel n (PDMm_PDSFCRCHn) (m = 0 to 1, n = 0 to 2) 3676
8.7.2.2.20 High-Pass Filter Coefficient s(0) Register Channel n (PDMm_PDHFCS0RCHn) (m = 0 to 1, n = 0 to 2)..... 3678
8.7.2.2.21 High-Pass Filter Coefficient k(1) Register Channel n (PDMm_PDHFCK1RCHn) (m = 0 to 1, n = 0 to 2)..... 3679
8.7.2.2.22 High-Pass Filter Coefficient h(p) Register Channel n (PDMm_PDHFCHpRCHn) (m = 0 to 1, n = 0 to 2, p = 0 to 1)..... 3680
8.7.2.2.23 Compensation Filter Coefficient h(p) Register Channel n (PDMm_PDCFCHpRCHn) (m = 0 to 1, n = 0 to 2, p = 00 to 10)..... 3681
8.7.2.2.24 Low-Pass Filter Coefficient h0(10) Register Channel n (PDMm_PDLFCH010RCHn) (m = 0 to 1, n = 0 to 2)..... 3682
8.7.2.2.25 Low-Pass Filter Coefficient h1(p) Register Channel n (PDMm_PDLFCH1pRCHn) (m = 0 to 1, n = 0 to 2, p = 0 to 19)..... 3683
8.7.2.2.26 Sound Detection Lower Threshold Register Channel n (PDMm_PDSDLTRCHn) (m = 0 to 1, n = 0 to 2) 3684
8.7.2.2.27 Sound Detection Upper Threshold Register Channel n (PDMm_PDSDUTRCHn) (m = 0 to 1, n = 0 to 2)..... 3685
8.7.2.2.28 Data Buffer Control Register Channel n (PDMm_PDDBCRCHn) (m = 0 to 1, n = 0 to 2) 3686
8.7.2.2.29 Short Circuit Threshold Setting Register Channel n (PDMm_PDSCTSRCHn) (m = 0 to 1, n = 0 to 2)..... 3687
8.7.2.2.30 Overvoltage Lower Threshold Register Channel n (PDMm_PDOVLTRCHn) (m = 0 to 1, n = 0 to 2)..... 3688

8.7.2.2.31 Overvoltage Upper Threshold Register Channel n (PDMm_PDOVUTRCHn) (m = 0 to 1, n = 0 to 2) ... 3689
8.7.2.2.32 Data Read Control Register Channel n (PDMm_PDDRCRCHn) (m = 0 to 1, n = 0 to 2) ... 3690
8.7.2.2.33 Data Clear Register Channel n (PDMm_PDDCRCHn) (m = 0 to 1, n = 0 to 2) ... 3691
8.7.2.2.34 Data Read Register Channel n (PDMm_PDDRRCHn) (m = 0 to 1, n = 0 to 2) ... 3692
8.7.2.2.35 Data Status Register Channel n (PDMm_PDDSRCHn) (m = 0 to 1, n = 0 to 2) ... 3693
8.7.2.3 Mirroring Register ... 3694
8.7.3 Operation ... 3695
8.7.3.1 Clock Specifications ... 3695
8.7.3.2 Channel Start and Channel Stop ... 3696
8.7.3.3 PDM_CLKn's Frequency Change ... 3697
8.7.3.4 Filter ... 3698
8.7.3.4.1 Data Clipping and Shifting between Filters ... 3698
8.7.3.4.2 Sinc Filter ... 3702
8.7.3.4.3 High-Pass Filter ... 3703
8.7.3.4.4 Compensation Filter ... 3704
8.7.3.4.5 Low-Pass (Half-Band Decimation) Filter ... 3705
8.7.3.4.6 Moving Average Filter ... 3707
8.7.3.4.7 Settling Time of Channel Activation / Setting Change ... 3708
8.7.3.5 Data Input Control for Stereo Sound ... 3710
8.7.3.6 Sound Detection ... 3712
8.7.3.7 Data Buffer ... 3713
8.7.3.8 Short-Circuit Detection ... 3714
8.7.3.9 Overvoltage Detection ... 3715
8.7.3.10 Error ... 3716
8.7.3.11 Interrupt Function ... 3716
8.7.4 Setting Flow ... 3717
8.7.4.1 Start Flow ... 3720
8.7.4.2 Stop Flow ... 3721
8.7.4.3 Normal Processing Flow ... 3722
8.7.4.4 Low Power Mode Transition Flow ... 3723
8.7.4.5 Normal Mode Transition Flow ... 3724
8.7.4.6 Setting PDM_CLKn (n = 0, 1, 2) Frequency and Sampling Frequency Generated by the Unit ... 3725
8.7.4.7 DMA Transfer Method ... 3728
8.7.4.8 Reset Clock Control ... 3731
8.7.5 Interrupt ... 3732
8.7.5.1 Interrupt List ... 3732
8.7.6 Usage Notes ... 3733
8.7.6.1 Data Reception Interrupt Threshold Register Setting ... 3733
8.7.6.2 Frequency Characteristics of Filters with Default Settings ... 3733

SECTION 9 IMAGE 3735
9.1 Image Overview 3735
9.1.1 Variation of Display Output 3736
9.2 Camera Data Receiver Unit (CRU) 3738
9.2.1 Overview 3738
9.2.2 Connected Units 3739
9.2.3 MIPI CSI2 3740
9.2.3.1 Overview 3740
9.2.3.1.1 Features 3740
9.2.3.1.2 Block Diagram 3742
9.2.3.1.3 External Pins 3743
9.2.3.2 MIPI CSI2 Registers 3744
9.2.3.2.1 List of Registers 3744
9.2.3.2.2 Register Descriptions 3748
9.2.3.3 Operation (Short Packet Reception) 3787
9.2.3.3.1 Data Reception 3787
9.2.3.4 Interrupts 3792
9.2.4 Image Converter 3796
9.2.4.1 Overview 3796
9.2.4.1.1 Features 3797
9.2.4.1.2 Block Diagram 3800
9.2.4.2 IMC Registers 3803
9.2.4.2.1 List of Registers [m = 0] 3803
9.2.4.2.2 Register Descriptions 3809
9.2.4.3 Function Description 3940
9.2.4.3.1 Input Formats and Image Processing 3940
9.2.4.3.2 Output Formats 3947
9.2.4.3.3 Transfer to DRAM 3953
9.2.4.3.4 Pattern Generator (for Debugging) 3972
9.2.4.3.5 VC/Data Type Selector (for Image Processing) 3978
9.2.4.3.6 8-bit Expansion 3979
9.2.4.3.7 Image Clipping 3980
9.2.4.3.8 Frame Subsampling 3981
9.2.4.3.9 Data Arrangers 1 and 2 3981
9.2.4.3.10 Data Clipping 3982
9.2.4.3.11 Demosaicing 3982
9.2.4.3.12 Linear Matrix 3985
9.2.4.3.13 Color Space Conversion 3986
9.2.4.3.14 LUT (Lookup Table) 3988
9.2.4.3.15 YC Post Processing 3989
9.2.4.3.16 VC/Data Type Selector (for Non-Image Processing) 3989
9.2.4.3.17 Full Bypass function 3990
9.2.4.3.18 Statistics 3991
9.2.4.4 Interrupts 3998

9.2.4.4.1 Interrupt list ... 3998
9.2.4.4.2 Designated address forwarding completion interrupt ... 4000
9.2.4.5 Usage Notes ... 4001
9.2.5 Operation ... 4003
9.2.5.1 Starting Reception for the MIPI CSI-2 Input ... 4003
9.2.5.1.1 Initializing the D-PHY ... 4005
9.2.5.1.2 Initializing the AXI Master ... 4006
9.2.5.1.3 Initializing the Image Converter ... 4007
9.2.5.1.4 Storing LUT Data ... 4009
9.2.5.1.5 Initializing the LINK ... 4010
9.2.5.2 Stopping Reception for the MIPI CSI-2 Input ... 4011
9.2.5.2.1 Determining LINK Packet Stop ... 4012
9.2.5.2.2 Stopping the D-PHY ... 4012
9.2.5.2.3 Stopping Image Converter Reception ... 4013
9.2.5.3 Frame Synchronization Settings ... 4015
9.2.5.3.1 Changing Frame Synchronization Settings (by Setting Registers) ... 4016
9.2.5.3.2 Changing Frame Synchronization Settings (by Using the DMAC) ... 4017
9.2.5.4 Starting Reception for the Pattern Generator Input ... 4018
9.2.5.5 Stopping Reception for the Pattern Generator Input ... 4019
9.4 LCD Controller (LCDC) ... 4020
9.4.1 Overview ... 4020
9.4.1.1 Features ... 4020
9.4.1.2 Block Diagram ... 4022
9.4.1.2.1 FCPVD ... 4024
9.4.1.2.2 VSPD ... 4024
9.4.1.2.3 DU ... 4032
9.4.1.3 Clock and Interrupt signal ... 4033
9.4.2 FCPVD Registers ... 4034
9.4.2.1 List of FCPVD Registers ... 4034
9.4.2.2 FCPVD Register Description ... 4035
9.4.2.2.1 FCPVD Version Control Register (LCDC_FCPVD_FCP_VCR) ... 4035
9.4.3 VSPD Registers ... 4036
9.4.3.1 List of VSPD Registers ... 4036
9.4.3.2 VSPD Register Description ... 4041
9.4.3.2.1 Notational Conventions for Registers and Bit Fields ... 4041
9.4.3.2.2 Register Classification ... 4041
9.4.3.2.3 Restrictions on Access to Registers and Lookup Tables ... 4042
9.4.3.2.4 General Control Registers ... 4043
9.4.3.2.5 Display List Control Registers ... 4061
9.4.3.2.6 RPF Control Registers ... 4070
9.4.3.2.7 WPF Control Registers ... 4109
9.4.3.2.8 DPR Control Registers ... 4129
9.4.3.2.9 LUT Control Register ... 4135
9.4.3.2.10 BRS Control Registers ... 4139
9.4.3.2.11 LIF Control Registers ... 4152

9.4.4 DU Registers 4160
9.4.4.1 List of DU Registers 4160
9.4.4.2 DU Register Description 4161
9.4.4.2.1 DU Module Control Register 0 (LCDC_DU_MCR0) 4161
9.4.4.2.2 DU Module Status Register 0 (LCDC_DU_MSR0) 4162
9.4.4.2.3 DU Module Status Register 1 (LCDC_DU_MSR1) 4163
9.4.4.2.4 DU Interrupt Mask Register 0 (LCDC_DU_IMR0) 4164
9.4.4.2.5 DU Display I/F Timing Register 0 (LCDC_DU_DITR0) 4165
9.4.4.2.6 DU Display I/F Timing Register 1 (LCDC_DU_DITR1) 4166
9.4.4.2.7 DU Display I/F Timing Register 2 (LCDC_DU_DITR2) 4168
9.4.4.2.8 DU Display I/F Timing Register 3 (LCDC_DU_DITR3) 4169
9.4.4.2.9 DU Display I/F Timing Register 4 (LCDC_DU_DITR4) 4170
9.4.4.2.10 DU Module Control Register 1 (LCDC_DU_MCR1) 4171
9.4.4.2.11 DU PBUF Control Register 0 (LCDC_DU_PBCR0) 4172
9.4.4.2.12 DU PBUF Control Register 1 (LCDC_DU_PBCR1) 4173
9.4.4.2.13 DU PBUF Control Register 2 (LCDC_DU_PBCR2) 4174
9.4.5 Operation 4175
9.4.5.1 Operation Control Setting 4175
9.4.5.2 Reset Operation 4175
9.4.5.3 Registers to set fixed value 4175
9.4.5.4 Concept of VSPD Operation Starting and Stopping 4176
9.4.5.5 Display List 4178
9.4.5.5.1 Functional description 4178
9.4.5.5.2 Normal display list mode 4180
9.4.5.5.3 Header-less display list mode 4185
9.4.5.5.4 Restrictions on display list usage 4185
9.4.5.6 Interrupt Processing 4186
9.4.5.6.1 vs1o_intreq_n 4186
9.4.5.6.2 duo_intr_n 4186
9.4.5.7 Lookup Table Settings 4187
9.4.5.7.1 LUT 4187
9.4.5.8 Linked with DU 4188
9.4.5.8.1 Operation flow of VSPD and DU 4188
9.4.5.8.2 Controlling two register planes using display lists 4191
9.4.6 Usage Note 4193
9.4.6.1 Assignment in Memory Space 4193
9.4.6.2 Limitations on Software Reset and Module Standby 4193
9.4.6.3 Input Image Size 4193
9.4.6.4 Output Image Size 4194
9.4.6.5 Restriction 4196
9.5 MIPI DSI Interface (DSI) 4197
9.5.1 Overview 4197
9.5.1.1 Features 4197
9.5.1.2 Block Diagram 4200
9.5.1.3 External Pins 4201
9.5.1.4 Clocks 4202

9.5.2 DSI Registers 4203
9.5.2.1 List of Registers 4203
9.5.2.1.1 Registers in LINK 4203
9.5.2.1.2 Registers in D-PHY 4208
9.5.2.2 MIPI DSI Register Descriptions 4209
9.5.2.2.1 Interrupt Status Register (DSI_LINK_ISR) 4209
9.5.2.2.2 Link Status Register (DSI_LINK_LINKSR) 4210
9.5.2.2.3 General Purpose Output 0 Register (DSI_LINK_GPO0R) 4211
9.5.2.2.4 Tx Set Register (DSI_LINK_TXSETR) 4212
9.5.2.2.5 HS Clock Set Register (DSI_LINK_HSCLKSETR) 4213
9.5.2.2.6 ULPS Set Register (DSI_LINK_ULPSSETR) 4214
9.5.2.2.7 ULPS Control Register (DSI_LINK_ULPSCR) 4215
9.5.2.2.8 Reset Control Register (DSI_LINK_RSTCR) 4216
9.5.2.2.9 Reset Status Register (DSI_LINK_RSTSR) 4217
9.5.2.2.10 DSI Set Register (DSI_LINK_DSISETR) 4218
9.5.2.2.11 Receive Buffer Size Register (DSI_LINK_RXBUFSZR) 4219
9.5.2.2.12 Tx Packet Payload Data 0 Register (DSI_LINK_TXPPD0R) 4220
9.5.2.2.13 Tx Packet Payload Data 1 Register (DSI_LINK_TXPPD1R) 4221
9.5.2.2.14 Tx Packet Payload Data 2 Register (DSI_LINK_TXPPD2R) 4222
9.5.2.2.15 Tx Packet Payload Data 3 Register (DSI_LINK_TXPPD3R) 4223
9.5.2.2.16 Rx Status Register (DSI_LINK_RXSR) 4224
9.5.2.2.17 Rx Status Clear Register (DSI_LINK_RXSCR) 4226
9.5.2.2.18 Rx Interrupt Enable Register (DSI_LINK_RXIER) 4228
9.5.2.2.19 Peripheral Response Timeout BTA Set Register (DSI_LINK_PRESPTOBTASETR) 4230
9.5.2.2.20 Peripheral Response Timeout LP Set Register (DSI_LINK_PRESPTOLPSETR) 4231
9.5.2.2.21 Peripheral Response Timeout HS Set Register (DSI_LINK_PRESPTOHSSETR) 4232
9.5.2.2.22 Acknowledge and Error Report Packet Parameter Latest Info Register (DSI_LINK_AKEPLATIR) 4233
9.5.2.2.23 Acknowledge and Error Report Packet Parameter Accumulate Status Register (DSI_LINK_AKEPACMSR) 4234
9.5.2.2.24 Acknowledge and Error Report Packet Parameter Status Clear Register (DSI_LINK_AKEPSCR) 4235
9.5.2.2.25 Rx Result Saved Status Register (DSI_LINK_RXRSSR) 4236
9.5.2.2.26 Rx Result Saved Status Clear Register (DSI_LINK_RXRSSCR) 4237
9.5.2.2.27 Rx Result Info Overwrite Status Register (DSI_LINK_RXRINFOOWSR) 4238
9.5.2.2.28 Rx Result Info Overwrite Status Clear Register (DSI_LINK_RXRINFOOWSCR) 4239
9.5.2.2.29 Rx Result Save Slot x Register (DSI_LINK_RXRSSxR) (x = 0, 1, 2, 3) ... 4240
9.5.2.2.30 Rx Packet Payload Data 0 Register (DSI_LINK_RXPPD0R) 4241
9.5.2.2.31 Rx Packet Payload Data 1 Register (DSI_LINK_RXPPD1R) 4242

9.5.2.2.32 Rx Packet Payload Data 2 Register (DSI_LINK_RXPPD2R) 4243
9.5.2.2.33 Rx Packet Payload Data 3 Register (DSI_LINK_RXPPD3R) 4244
9.5.2.2.34 HSTX Timeout Set Register (DSI_LINK_HSTXTOSETR) 4245
9.5.2.2.35 LRX-H Timeout Set Register (DSI_LINK_LRXHTOSETR) 4246
9.5.2.2.36 TA Timeout Set Register (DSI_LINK_TATOSETR) 4247
9.5.2.2.37 Fatal Error Status Register (DSI_LINK_FERRSR) 4248
9.5.2.2.38 Fatal Error Status Clear Register (DSI_LINK_FERRSCR) 4249
9.5.2.2.39 Fatal Error Interrupt Enable Register (DSI_LINK_FERRIER)..... 4250
9.5.2.2.40 Clock Lane Stop Time Set Register (DSI_LINK_CLSTPTSETR) 4251
9.5.2.2.41 LP Transition Time Set Register (DSI_LINK_LPTRNSTSETR) 4253
9.5.2.2.42 Physical Lane Status Register (DSI_LINK_PLSR) 4254
9.5.2.2.43 Physical Lane Status Clear Register (DSI_LINK_PLSCR) 4256
9.5.2.2.44 Physical Lane Interrupt Enable Register (DSI_LINK_PLIER)..... 4257
9.5.2.2.45 Video-Input Channel 1 Set 0 Register (DSI_LINK_VICH1SET0R)..... 4258
9.5.2.2.46 Video-Input Channel 1 Set 1 Register (DSI_LINK_VICH1SET1R)..... 4260
9.5.2.2.47 Video-Input Channel 1 Status Register (DSI_LINK_VICH1SR)..... 4261
9.5.2.2.48 Video-Input Channel 1 Status Clear Register (DSI_LINK_VICH1SCR) ... 4262
9.5.2.2.49 Video-Input Channel 1 Interrupt Enable Register (DSI_LINK_VICH1IER) 4263
9.5.2.2.50 Video-Input Channel 1 Pixel Packet Set Register (DSI_LINK_VICH1PPSETR)..... 4264
9.5.2.2.51 Video-Input Channel 1 Vertical Size Set Register (DSI_LINK_VICH1VSSETR)..... 4265
9.5.2.2.52 Video-Input Channel 1 Vertical Porch Set Register (DSI_LINK_VICH1VPSETR)..... 4266
9.5.2.2.53 Video-Input Channel 1 Horizontal Size Set Register (DSI_LINK_VICH1HSSETR)..... 4267
9.5.2.2.54 Video-Input Channel 1 Horizontal Porch Set Register (DSI_LINK_VICH1HPSETR)..... 4268
9.5.2.2.55 Sequence Channel x Set 0 Register (DSI_LINK_SQCHxSET0R)..... 4269
9.5.2.2.56 Sequence Channel x Set 1 Register (DSI_LINK_SQCHxSET1R)..... 4270
9.5.2.2.57 Sequence Channel x Status Register (DSI_LINK_SQCHxSR) 4271
9.5.2.2.58 Sequence Channel x Status Clear Register (DSI_LINK_SQCHxSCR) 4273
9.5.2.2.59 Sequence Channel x Interrupt Enable Register (DSI_LINK_SQCHxIER) 4275
9.5.2.2.60 Sequence Channel x Descriptor y-A Register (DSI_LINK_SQCHxDSCyAR) 4277
9.5.2.2.61 Sequence Channel x Descriptor y-B Register (DSI_LINK_SQCHxDSCyBR) 4279
9.5.2.2.62 Sequence Channel x Descriptor y-C Register (DSI_LINK_SQCHxDSCyCR) 4280
9.5.2.2.63 Sequence Channel x Descriptor y-D Register (DSI_LINK_SQCHxDSCyDR) 4281
9.5.2.3 Registers in D-PHY.....4282
9.5.2.3.1 DSI PLL Enable Register (DSI_DPHY_PLLENR)..... 4282

9.5.2.3.2 DSI PHY Reset Register (DSI_DPHY_PHYRSTR) 4283
9.5.2.3.3 DSI PLL Clock Setting 0 Register (DSI_DPHY_PLLCLKSET0R)..... 4284
9.5.2.3.4 DSI PLL Clock Setting 1 Register (DSI_DPHY_PLLCLKSET1R)..... 4285
9.5.2.3.5 DSI PHY TCLK Setting Register (DSI_DPHY_PHYTCLKSETR) 4286
9.5.2.3.6 DSI PHY THS Setting Register (DSI_DPHY_PHYTHSSETR) 4287
9.5.2.3.7 DSI PHY TLPX Setting Register (DSI_DPHY_PHYTLPXSETR)..... 4288
9.5.2.3.8 DSI PHY Control Register (DSI_DPHY_PHYCR)..... 4289
9.5.2.3.9 DSI PHY Control 1 Register (DSI_DPHY_PHYC1R)..... 4290
9.5.2.3.10 DSI PHY Control 2 Register (DSI_DPHY_PHYC2R)..... 4291
9.5.2.3.11 DSI PHY Control 3 Register (DSI_DPHY_PHYC3R)..... 4292
9.5.3 Operation 4293
9.5.3.1 Power on Reset and Initial Settings for All Operations.....4293
9.5.3.2 Reset 4299
9.5.3.2.1 Software Reset..... 4299
9.5.3.3 Start/Stop of HS Clock.....4301
9.5.3.4 Sequence Operation.....4304
9.5.3.4.1 Basic Running Sequence..... 4304
9.5.3.4.2 Single Packet Transmission 4304
9.5.3.4.3 Single Packet Reception 4306
9.5.3.4.4 Reset Trigger Transmission..... 4308
9.5.3.4.5 Execute Operation in Sequence 4308
9.5.3.4.6 Receiving EoTp 4309
9.5.3.4.7 Acknowledge and Error Reporting Mechanism..... 4309
9.5.3.5 Video-Input Operation 4310
9.5.3.5.1 Video-Input Timing..... 4310
9.5.3.5.2 Video-Input Start, End, Change Setting 4314
9.5.3.5.3 Start of Video-Input Operation..... 4314
9.5.3.5.4 End of Video-Input Operation..... 4316
9.5.3.5.5 Burst Mode 4317
9.5.3.5.6 Non-Burst Mode 4317
9.5.3.5.7 LP Transition and Blanking..... 4317
9.5.3.5.8 Non video packet action during Video-Input Operation Running..... 4317
9.5.3.5.9 Delay about Video-Input Operation..... 4318
9.5.3.6 Enter/Exit of ULPS.....4319
9.5.3.7 Error Handling.....4323
9.5.3.7.1 Error Detection by Peripheral..... 4323
9.5.3.7.2 Error Detection in D-PHY 4323
9.5.3.7.3 Timeout Error..... 4324
9.5.3.7.4 Rx Related Error in Sequence Operation..... 4325
9.5.3.7.5 Tx Packet Related Error in Sequence Operation..... 4328
9.5.3.7.6 Error in Video-Input Operation 4328
9.5.4 Interrupts 4329
9.5.4.1 Interrupt List.....4329
9.5.5 Usage Notes 4332

9.5.5.1 Restriction and Notes 4332
9.6 H.265/H.264 Multi Codec (VCD) 4333
9.6.1 Functional Overview 4333
9.6.2 Connection Configuration 4334
9.6.3 Operation Procedure 4334
9.7 3D Graphics Engine (GE3D) 4335
9.7.1 Overview 4335
9.7.1.1 Features 4335
9.7.1.2 Block Diagram 4336
9.7.1.3 External Pins 4337
9.7.1.4 Register Configuration 4337
9.7.1.5 Usage Note 4337
9.7.1.6 Starting SYC 4337
9.9 Video Signal Processor (VSP) 4338
9.9.1 VSPI 4339
9.9.1.1 Features 4339
9.9.1.1.1 Block diagram of VSPI 4340
9.9.1.1.2 Memory Access Unit (MAU) 4340
9.9.1.1.3 Command Transfer Unit (CTU) 4341
9.9.1.1.4 Read Pixel Formatter (RPF) 4341
9.9.1.1.5 Data Path Router (DPR) 4342
9.9.1.1.6 Super Resolution Unit (SRU) 4342
9.9.1.1.7 Up Down Scalar (UDS) 4342
9.9.1.1.8 Look Up Table (LUT) 4342
9.9.1.1.9 Write Pixel Formatter (WPF) 4342
9.9.1.1.10 Cubic Look Up table (CLU) 4343
9.9.1.1.11 Hue Saturation value Transform (HST) 4343
9.9.1.1.12 Hue Saturation value Inverse transform (HSI) 4343
9.9.1.1.13 Histogram Generator-One dimension (HGO) 4343
9.9.1.1.14 Histogram Generator-Two dimension (HGT) 4343
9.9.1.1.15 SHarPness (SHP) 4343
9.9.1.1.16 Detail Function 4344
9.9.1.1.17 Restrictions 4347
9.9.1.2 Register Configuration 4348
9.9.1.2.1 Base Address 4348
9.9.1.2.2 Register/Table Address 4348
9.9.1.3 Register Description 4354
9.9.1.3.1 Notational Conventions for Registers and Bit Fields 4354
9.9.1.3.2 Register Classification 4354
9.9.1.3.3 Restrictions on Access to Registers and Lookup Tables 4355
9.9.1.3.4 General Control Registers 4356
9.9.1.3.5 Display List Control Registers 4371
9.9.1.3.6 RPF Control Registers 4379
9.9.1.3.7 WPF Control Registers 4420
9.9.1.3.8 DPR Control Registers 4442

9.9.1.3.9 SRU Control Registers 4448
9.9.1.3.10 UDS Control Registers 4452
9.9.1.3.11 LUT Control Register 4467
9.9.1.3.12 CLU Control Register 4471
9.9.1.3.13 HST Control Register 4475
9.9.1.3.14 HSI Control Register 4476
9.9.1.3.15 HGO Control Registers 4477
9.9.1.3.16 HGT Control Registers 4495
9.9.1.3.17 SHP Control Registers 4511
9.9.1.3.18 VSPI Internal Registers 4516
9.9.1.3.19 RPF0 OSD-CLUT table 4517
9.9.1.3.20 LUT table 4517
9.9.1.3.21 CLU table address register and table data register 4517
9.9.1.4 Operation 4518
9.9.1.4.1 Registers to set fixed value 4518
9.9.1.4.2 Concept of VSPI Operation Starting and Stopping 4518
9.9.1.4.3 Display List 4521
9.9.1.4.4 Interrupt Processing 4529
9.9.1.4.5 Lookup Table Settings 4530
9.9.1.4.6 Image partition for VSPI processing 4534
9.9.1.4.7 The step to read HGO histogram registers of 256 step mode by indirect addressing 4555
9.9.1.5 Usage Notes 4556
9.9.1.5.1 Assignment in Memory Space 4556
9.9.1.5.2 Input Image Size 4556
9.9.1.5.3 Output Image Size 4558
9.9.1.5.4 Configuration of VSPI 4559
9.9.1.5.5 Performance of VSPI 4560
9.9.1.5.6 Legend of Terms 4560
9.9.2 FCPVI 4562
9.9.2.1 Features 4562
9.9.2.2 Block Diagram 4563
9.9.2.3 IP Set and Bus Access Functions 4563
9.9.2.4 Register Configuration 4564
9.9.2.5 Register Description 4565
9.9.2.6 Operation 4569
9.9.2.6.1 Operation Control Setting IP set VSPI + FCPVI 4569
9.9.2.6.2 Reset Operation 4569
9.9.2.6.3 Register Settings and Restrictions on Connected VSPI 4570
9.9.2.7 Usage Notes 4570
9.9.2.7.1 Manipulating FCPVI, VSPI with CPG module stop, software reset 4570
9.9.3 Procedure for Starting and Stopping VSP 4571
9.9.3.1 Procedure for Starting VSP 4571
9.9.3.2 Procedure for Stopping VSP 4572

9.10 Frame Data Processor (FDP) 4573
9.10.1 FDP1B 4574
9.10.1.1 Features 4574
9.10.1.2 Block Diagram 4575
9.10.1.2.1 ABI (AXI Bus Interface) 4575
9.10.1.2.2 RPF (Read Pixel Formatter) 4575
9.10.1.2.3 WPF (Write Pixel Formatter) 4575
9.10.1.2.4 Input/Output Data Format 4576
9.10.1.3 Register Configuration 4577
9.10.1.4 Register Description 4579
9.10.1.4.1 Access Restrictions 4579
9.10.1.4.2 Register Type Definitions 4579
9.10.1.4.3 Notational Conventions for Registers and Bit Fields 4579
9.10.1.4.4 Register Classification 4579
9.10.1.4.5 Restrictions on Access to Registers and Lookup Tables 4580
9.10.1.4.6 General Control Registers 4581
9.10.1.4.7 RPF Control Registers 4596
9.10.1.4.8 WPF Control Registers 4607
9.10.1.4.9 Other Control Registers 4615
9.10.1.5 Operation 4618
9.10.1.5.1 Register Usages for FDP1B Start and Stop 4618
9.10.1.5.2 Register Setting Order for FDP1B Start and Stop 4619
9.10.1.6 Usage Notes 4621
9.10.1.6.1 Assignment in Memory Space 4621
9.10.1.6.2 FCPF Connection 4621
9.10.1.6.3 Manipulate FDP with Software Reset of CPG 4621
9.10.1.6.4 Terminology 4621
9.10.2 FCPF 4622
9.10.2.1 Features 4622
9.10.2.2 Block Diagram 4623
9.10.2.2.1 IP Set and Bus Access Functions 4623
9.10.2.3 Register Configuration 4624
9.10.2.4 Register Descriptions 4625
9.10.2.4.1 FCP Version Control Register (FCP_VCR) 4625
9.10.2.4.2 FCP Status Register (FCP_STA) 4626
9.10.2.4.3 FCP Tile/Linear Control Register (FCP_TL_CTRL) 4627
9.10.2.4.4 FCP Picture Information 1 (FCP_PICINFO1) 4628
9.10.2.4.5 FCP Base Address Reference picture of plane m (FCP_BA_REF_m) (m = Y, C) 4629
9.10.2.5 Operation 4630
9.10.2.5.1 Operation Control Setting for the IP Set FDP 4630
9.10.2.5.2 Reset Operation 4631
9.10.2.5.3 Register Settings and Restrictions on Connected IP (FDP) 4632
9.10.2.5.4 Decode Data 4633
9.10.2.5.5 Operation Status 4635

9.10.2.6 Usage Notes 4636
9.10.2.6.1 Manipulating FDP with CPG Software Reset 4636
9.11 Low Voltage Differential Signaling (LVDS) 4637
9.11.1 Overview 4637
9.11.1.1 Feature 4637
9.11.1.2 Block Diagram 4638
9.11.1.3 External Pins 4639
9.11.1.4 Connected Module 4639
9.11.2 Registers 4640
9.11.2.1 List of Registers 4640
9.11.2.2 LVDS Common Register (LVDS_CMN) 4641
9.11.2.3 LVDS0 PHY Register (LVDS_0_PHY) 4642
9.11.2.4 LVDS0 Control Register (LVDS_0_CTL) 4643
9.11.2.5 LVDS1 PHY Register (LVDS_1_PHY) 4644
9.11.2.6 LVDS1 Control Register (LVDS_1_CTL) 4645
9.11.3 Operation 4646
9.11.3.1 Power On Sequence 4646
9.11.3.2 Power Off Sequence 4650
9.11.3.3 LVDS Operating Modes 4651
9.11.3.4 Input Format 4654
9.11.3.5 Output Format Selection 4655
9.11.3.6 Input/Output Data Description 4659
9.11.3.7 Output Pattern 4663
9.11.3.8 Control Bit Switching 4664
9.11.3.9 Output Lane Selection 4665
9.11.3.10 Dual-link Output Data Swap 4666
9.11.3.11 Asynchronous FIFO 4667
9.11.3.12 PHY Reset 4667
9.11.3.13 PHY Skew Adjustment 4667
9.11.3.14 PHY IO Buffer Enable 4667
9.11.4 Interrupt 4668
9.11.5 Usage Note 4668
SECTION 10 ELECTRICAL CHARACTERISTICS 4669
10.1 Electrical Characteristics 4669
10.1.1 Absolute Maximum Ratings 4669
10.1.2 Recommended Operating Range 4671
10.1.3 Power-On/Off Sequence 4673
10.1.3.1 CM33 Boot Mode (State Diagram) 4673
10.1.3.2 CM33 Boot Mode (Power-On) 4674
10.1.3.3 CM33 Boot Mode (DDR Retention Enter) 4676
10.1.3.4 CM33 Boot Mode (DDR Retention Exit) 4678
10.1.3.5 CM33 Boot Mode (Power-Off) 4680
10.1.3.6 CA55 Boot Mode (State Diagram) 4682
10.1.3.7 CA55 Boot Mode (Power-On) 4683
10.1.3.8 CA55 Boot Mode (DDR Retention Enter) 4685
10.1.3.9 CA55 Boot Mode (DDR retention exit) 4687
10.1.3.10 CA55 Boot Mode (Power-Off) 4689
10.1.4 DC Characteristics 4691
10.1.4.1 Maximum Supply Current 4691
10.1.4.2 Standard I/O Characteristics 4693

10.1.5 AC Characteristics 4698
10.1.5.1 Clock Timing 4698
10.1.5.2 CMTW Timing 4699
10.1.5.3 POEG and GPT Trigger Timings 4700
10.1.5.4 Watchdog Timer Access Timing 4700
10.1.5.5 DMAC Timing 4701
10.1.5.6 LPDDR4 PHY Characteristics 4702
10.1.5.7 SD Access Timing 4702
10.1.5.7.1 SD Access Timing (SDR 3.3-V) 4702
10.1.5.8 eMMC Access Timing 4703
10.1.5.8.1 eMMC host interface timing (default) 4703
10.1.5.8.2 eMMC host interface timing (HS-SDR) 4704
10.1.5.8.3 eMMC host interface timing (HS-DDR) 4704
10.1.5.8.4 eMMC host interface timing (HS200) 4706
10.1.5.9 Ethernet Interface Timing 4707
10.1.5.10 USB 3.2 PHY Characteristics 4709
10.1.5.11 USB 2.0 PHY Characteristics 4709
10.1.5.12 PCI Express PHY Characteristics 4709
10.1.5.13 xSPI Timing 4710
10.1.5.14 Serial Communications Interface (RSCI) Access Timing 4714
10.1.5.15 Renesas Serial Peripheral Interface (RSPI) Access Timing 4719
10.1.5.16 Renesas IIC Bus Interface (RIIC) Access Timing 4724
10.1.5.17 I3C Timing 4726
10.1.5.18 CANFD Interface Access Timing 4738
10.1.5.19 A/D Converter Access Timing 4739
10.1.5.20 SSIU Timing 4740
10.1.5.21 PDM Timing 4743
10.1.5.22 MIPI CSI-2 PHY Characteristics 4744
10.1.5.23 MIPI DSI Tx D-PHY Characteristics 4744
10.1.5.24 Control Signal Access Timing 4745
10.1.5.25 JTAG Debugger Interface Access Timing 4746
10.1.5.26 LCDC 4748
10.1.5.27 LVDS Interface 4749
10.1.6 Analog Characteristics 4750
10.1.6.1 ADC Characteristics 4750
10.1.6.2 Temperature Sensor Characteristics 4751
10.1.7 Oscillation Circuits for Connecting Crystal Resonators (OSC) 4752
Appendix A PACKAGE DIMENSIONS 4754
A.1 Package Dimensions 4754

SECTION 1 OVERVIEW

1.1 Overview

1.1.1 Features

This LSI includes:

■ CPU

- On-chip Quad 64-bit Arm® Cortex®-A55 Core processors [Quad/Dual option]

  Application processing (up to 1.8 GHz)
- 32-bit Arm® Cortex®-M33 processor

  System management (up to 200 MHz)

■ Boot

- Selectable boot CPU from Arm® Cortex®-M33 or Arm® Cortex®-A55

■ Micro Neural network processing unit (NPU)

- NPU of Arm® Ethos-U55 (1 GHz) [option]

■ Video and Graphics

- H.264/H.265 Video codec unit (VCD)
- 3D Graphics Engine of Arm® Mali-G52
- Image Scaling Unit (VSP)
- Frame Data Processor (FDP)

■ On-chip SRAM and external memory interfaces

- On-chip shared SRAM (512-Kbyte on-chip SRAM with ECC)
- External DDR memory interface

  1-channel memory controller for LPDDR4-3200 or LPDDR4X-3200 with a 32-bit bus width
- xSPI interface
- SDHI (eMMC/SD (1-, 4-, 8-bit bus width) supported)

■ Camera and display interfaces

- MIPI CSI-2 (1 ch.: 1, 2, or 4 lanes)
- Selectable 2 display interfaces from the following

  MIPI DSI (1 ch.: 1, 2, or 4 lanes)

  Parallel interface (1 ch.)

  LVDS (2 ch.)

■ Various communication/storage/network interfaces

- Ethernet (2 ch.: 10/100/1000 BASE)
- USB2.0 (1 ch.: Host/Function, 1 ch.: Host-only)
- USB3.2 Gen2 × 1 (Host-only)
- PCIe Gen3 × 1 (1 or 2 lanes)
- CAN/CANFD (compliant with ISO11898-1) (6 ch.)
- SCI (10 ch.: UART/SPI/I2C-host)
- SPI (3 ch.)
- I2C (9 ch.)
- I3C (1 ch.)

■ Extended-function timers

- 32-bit general-purpose timer (16 ch.)
- 32-bit CMTW (8 ch.)

■ Audio

- Asynchronous sampling rate converter unit (SCU) (up to 192 kHz)
- DMAC for Audio (ADMAC) is available to transfer audio formats of I2S with SCU.
- Flexible audio clock generator (ADG) for audio functions.
- I2S (TDM) input/output interfaces (half-duplex 10 ch.; full-duplex 5 ch.)
- SPDIF input/output interfaces (3 ch.)
- Pulse density modulation (PDM) input interfaces (6 ch.)

■ Analog/Digital converter (ADC) and sensors

- 2.5 Msps 12-bit ADC (8 ch.)
- Internal temperature sensors (1 ch.)

■ Security

- Hardware cryptographic engine

NOTE

- 15×15 mm FCBGA is available, but 21×21 mm FCBGA is under development.
- The products with #AC0 or #BC0 in the part number have the following restrictions.
  - Not supported controlling MIPI LCDs with the Display Command Set over MIPI-DSI.
  - Not supported the applications of the Ethos-U55 utilizing the LPDDR4/LPDDR4X.

1.1.2 Product Lineup

Figure 1.1-1 shows the product part number information. **Table 1.1-1** shows a list of products.

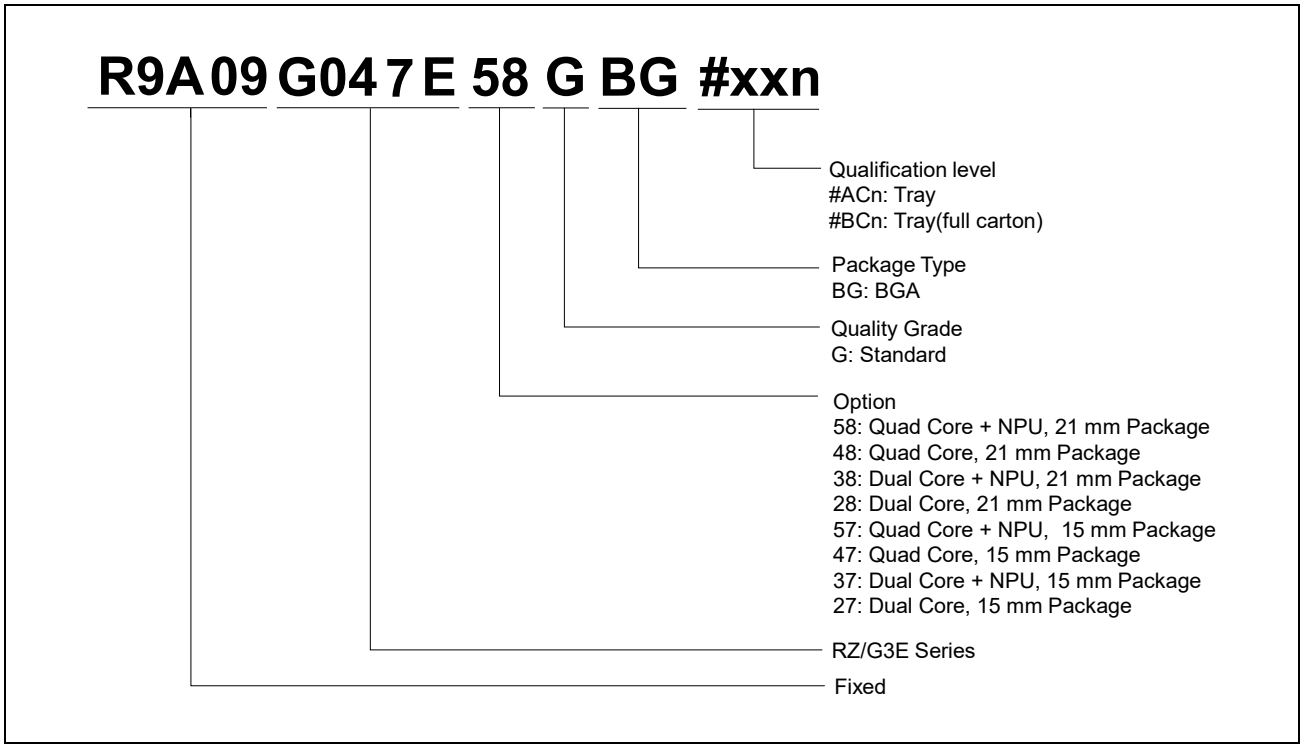


Figure 1.1-1 Part Numbering Scheme

Table 1.1-1 Product Lineup

Group	Package	Part Number	NPU	CPU
RZ/G3E	21 mm FCBGA*1	R9A09G047E58GBG*1	Ethos-U55	4x Cortex-A55, 1x Cortex-M33
		R9A09G047E48GBG*1	—	
		R9A09G047E38GBG*1	Ethos-U55	2x Cortex-A55, 1x Cortex-M33
		R9A09G047E28GBG*1	—	
	15 mm FCBGA	R9A09G047E57GBG	Ethos-U55	4x Cortex-A55, 1x Cortex-M33
		R9A09G047E47GBG	—	
		R9A09G047E37GBG	Ethos-U55	2x Cortex-A55, 1x Cortex-M33
		R9A09G047E27GBG	—	

Note 1. Under development

Note: The products with #AC0 or #BC0 in the part number have the following restrictions.
— Not supported controlling MIPI LCDs with the Display Command Set over MIPI-DSI.
— Not supported the applications of the Ethos-U55 utilizing the LPDDR4/LPDDR4X.

1.1.3 Functions

1.1.3.1 CPU

Item	Description
Application Processor Cortex-A55 (CA55) [Quad/Dual option]	• Arm Cortex-A55 Quad Core 1.8 GHz with 0.9 V, 1.1 GHz with 0.8 V • L1 I-cache 32 Kbytes (with parity) and D-cache 32 Kbytes (with ECC) per core • L2 cache: 0 Kbyte • L3 cache: 1 Mbyte (with ECC) • MMU supported • Neon™ and FPU supported • Cryptographic extension supported • Armv8-A architecture
System Manager Cortex-M33 (CM33)	• Arm Cortex-M33 Processor 200 MHz • FPU supported • DSP Extension supported • Security Extension supported • Armv8-M architecture
Debug Interface	• Arm® CoreSight® architecture • JTAG and SWD interfaces supported • ETF: Total of 60 Kbytes for program flow tracing • JTAG disabling supported [option]

1.1.3.2 Boot

Item	Description
Boot	• Selectable boot from Arm Cortex-A55 or Arm Cortex-M33 • CA55 boot – Boot Mode 0: Booting from eSD – Boot Mode 1: Booting from eMMC 3.3 V – Boot Mode 2: Booting from a serial flash memory connected to the xSPI bus space 3.3 V – Boot Mode 3: Booting from SCIF download – Boot Mode 4: Booting from USB download – Boot Mode 5: Booting from eMMC 1.8 V – Boot Mode 6: Booting from a serial flash memory connected to the xSPI bus space 1.8 V • CM33 boot – Boot Mode 2: Booting from a serial flash memory connected to the xSPI bus space 3.3 V – Boot Mode 3: Booting from SCIF download – Boot Mode 6: Booting from a serial flash memory connected to the xSPI bus space 1.8 V

1.1.3.3 NPU

Item	Description
Micro Neural network Processing Unit (NPU) [option]	• Arm Ethos-U55 1.0 GHz • 256MAC • 0.5 TOPS

1.1.3.4 Graphics Unit

Item	Description
3D Graphics Engine (GE3D)	• Arm Mali-G52 630 MHz • 64 Kbytes L2 cache • Vulkan 1.2 supported • OpenGL ES™ 1.1, 2.0, and 3.2 supported • OpenCL 2.0 full profile supported

1.1.3.5 Video Processing Unit

Item	Description
Video Codec Unit (VCD)	• H.264/H.265 codec module • Support for encoding and decoding – H.264/AVC (High Profile, level 4.2; Main Profile, level 4.2; Baseline Profile, level 4.2) – H.265/HEVC (Main Profile, level 5) • I-/P-slice supported for H.264/H.265 encoding I-/P-/B-slice supported for H.264/H.265 decoding • Maximum size – (H.264) 1920 × 1080 × 60 fps*1 – (H.265) 3840 × 2160p × 30 fps*1
Frame Data Processor (FDP)	• 1 channel • Tile-Linear conversion supported • 500 Mpix/s for output performance • Supports various image formats • Input: YCbCr444/422/420 • Output: YCbCr444/422/420 and RGB/αRGB • Input image Size (max): 3840 × 2160 • Output image Size (max): 3840 × 2160 • Not support Interlace format
Video Signal Processor (VSP)	• Scaling up & down function • Supports Various Data Formats and Conversion – Color space conversion and changes to the number of colors by dithering – Color keying • Supports combination between pixel alpha and global alpha • Supports generating pre multiplied alpha • Video Processing – Up and down scaling with arbitrary scaling ratio(×16 to ×1/15.9997) – Super resolution processing – Image rotation/reversal function: Reverses an image vertically/horizontally or rotates it by 90°/270° • Picture Quality/Color Correction with 1D/3D Look Up Table (LUT) – Hue, brightness, and saturation adjustment – 1D and 2D histogram • Following functions will be supported by Renesas software portfolio. – γ correction and gain correction – Correction of color • Input image size (max): 3840 × 2160 • Output image size (max): 3840 × 2160

Note 1. Maximum frame rate for this size. The number of streams can be defined within this specification by software.

1.1.3.6 Internal Memory

Item	Description
System RAM	• 512 Kbytes (with ECC)

1.1.3.7 External Memory Interface

Item	Description
External Bus Controller for LPDDR4/4X SDRAM (DDR)	• 1 channel • Support LPDDR4X-3200 Bus width: 32-bit Memory Size: Up to 4 GBytes, Single rank (15 mm PKG, 21 mm PKG) • Support LPDDR4-3200 Bus width: 32-bit Memory Size: Up to 4 GBytes, Single rank (15 mm PKG, 21 mm PKG) Memory Size: Up to 8 GBytes, Dual rank (15 mm PKG) • In line ECC supported (support for error detection interrupts) • Auto Refresh/Self Refresh/IO Retention supported • Memory access protection for secure regions using TZC-400 (Arm® TrustZone® supported)
xSPI Controller (xSPI)	• 1 channel (2 chip select signals) • Compliant with the xSPI protocol • Protocol mode 1, 4, or 8 pins with SDR or DDR (1S-1S-1S, 4S-4D-4D*1, 8D-8D-8D*1) 2 or 4 pins with SDR (1S-2S-2S, 2S-2S-2S, 1S-4S-4S, 4S-4S-4S) • Support for XiP mode • Support for up to 256-Mbyte address space (support for up to 128 Mbytes per channel address space in boot sequence)
SD Card Host Interface/ Multimedia Card Interface (SD/MMC)	• 3 channels Channel 0 supports SDHI/e-MMC (boot supported) Channel 1 supports SDHI/e-MMC Channel 2 supports SDHI • SD memory I/O card interface (1-bit/4-bit SD bus) • SD, SDHC and SDXC SD memory card access supported • Compliant with SD specification version 3.01 • Default, high-speed, UHS-I/SDR50, SDR104 and DDR50 transfer modes supported • Error check function: CRC7 (Command), CRC16 (Data) • Support for card detection and write protection • MMC interface Channel 0 (1-bit/4-bit/8-bit MMC bus) Channel 1 (1-bit/4-bit MMC bus) • e-MMC device access supported • Compliant with eMMC 4.51 • High-speed, HS200 and HS-DDR transfer modes supported

Note 1. DDR access without DS (XSPI0_DS) is not supported.

1.1.3.8 CPU Peripheral

Item	Description
Clock Pulse Generator (CPG)	• Generates the clocks from an external clock or external resonator (24 MHz). – Maximum Arm Cortex-A55 clock: 1.8 GHz with Over Drive (0.9 V) – Maximum Arm Cortex-M33 clock: 200 MHz – Maximum DDR clock: 800 MHz (LPDDR4/4X-3200) – Maximum 3DGE clock: 630 MHz – Maximum VSP clock: 800 MHz – Maximum NPU clock: 1.0 GHz – Maximum H.264/H.265 clock: 400 MHz – Maximum System Bus clock: 400 MHz • SSC (Spread Spectrum Clock) supported
Direct Memory Access Controller (DMAC)	• 80 channels • Transfer modes: Single transfer mode and block transfer mode • LINK mode (DMA transfer under descriptor control) supported • Transfer size: 1, 2, 4, 8, 16, 32, 64, or 128 bytes • Transfer request: Software trigger, external DMA requests (DREQ) and interrupt requests from peripheral functions • A specific DMA transfer interval can be specified to adjust the bus occupancy.
Interrupt Controller	• Arm® CoreLinkTM Generic Interrupt Controller (GIC-600) for Arm Cortex-A55 (32 priority levels available) • Nested Vectored Interrupt Controller (NVIC) for Arm Cortex-M33 • External Interrupt pins (NMI, IRQ0 to IRQ15, and TINT0 to TINT31) • On-chip peripheral Interrupts: priority level set for each module
Event Link Controller (ELC)	• Up to 455 event signals can be interlinked with the operation of modules • In particular, the operation of timer modules can be started by input event signals • Event-linked operation of signals of 16 ports P10~17, P70~77 is to be possible.
Error Controller	• Error events from CPU and peripherals are captured and merged to interrupt with mask for Arm Cortex-A55 and Arm Cortex-M33 respectively. • System reset can be generated by error events.
Message Handling Unit (MHU)	• Message handling function between Arm Cortex-A55 and Arm Cortex-M33 • Assert interrupt to inform message and response from/to every core
General-purpose I/O (GPIO)	• 141 General-purpose I/O ports

1.1.3.9 Camera Interface

Item	Description
MIPI CSI-2 Interface with camera image processing (CRU)	• Number of lanes: 1, 2, or 4 lanes • Maximum bandwidth: 2.1 Gbps per lane • Support for the throughput up to FullHD RAW12 60 fps • Support for 4 virtual channels selected from VC0 to VC15 • Support for input data formats: • YUV422 8 bits or 10 bits • RGB444, RGB555, RGB565, RGB666, RGB888 • RAW6, RAW7, RAW8, RAW10, RAW12, RAW14, RAW16, RAW20 • YUV420 8-bits or 10-bits (image processing not supported) • Legacy YUV420 8-bits (image processing not supported) • YUV420 8-bits or 10-bits (chroma shifted pixel sampling) (image processing not supported) • User defined byte-based data • The other formats from the MIPI CSI-2 interface can also be output without image processing. • Generic long packet data types 1 to 4 • User defined 8-bit data types 1 to 8

1.1.3.10 Display Interfaces

Item	Description
LCD Controller	• 2 channels • 2 planes blending (can blend 2 different size images) • Support Image Processing: – Dither processing (RGB666) – Clipping – RGB Gamma Correction LUT • Support Input Data Format: – RGB565/RGB666/RGB888 – ARGB1555/ARGB4444/ARGB8888 – YCbCr444 8-bit/YCbCr422 8-bit/YCbCr420 8-bit • Selectable 2 display interfaces from the following – 1 channel MIPI DSI (1/2/4-Lane) interface – 1 parallel interface – 2 LVDS interfaces • Support 1920 pixels × 1920 lines Pixel clock must be within the following range – MIPI DSI = 5.4 to 187.5 MHz – Parallel = 5.4 to 87 MHz – LVDS (single) = 25 to 87 MHz – LVDS (dual) = 50 to 174 MHz
MIPI DSI Interface	• 1 channel • The number of Lanes: 1,2, or 4-lane • Support 1920 × 1200, 60 fps (RGB888) • Maximum Bandwidth: 1.5 Gbps per lane • Support Output Data Format: RGB666/RGB888
Parallel Output Interface	• 1 channel • Support WXGA (1280×800), 60 fps • Support Output Data Format: RGB666/RGB888 • CLK/HD/VD timing signal supported
LVDS Interface	• 2 channels • Number of support lanes – Single-link: 4 lanes (Data) + 1 lane (Clock) – Single-link (2 ch.): 8 lanes (Data) + 2 lanes (Clock) – Dual-link：8 lanes (Data) + 2 lanes (Clock) • Single-link target – RGB888 640×480 (VGA) – RGB888 800×480 (QVGA) – RGB888 1366×768 60 Hz (WXGA) – RGB888 1280×800 60 Hz (WXGA) • Dual-link target – RGB888 1920×1200 60 fps (~3.32 Gbps) – RGB888 1920×1080 60 fps (~2.99 Gbps) • Output clock range: 25 to 87 MHz • Clock Duty: 4:3 • Color: 8 bits • ViD: 250 to 450 mV • Vcm (Vos): 1.125 to 1.375 V • SSC supported

1.1.3.11 Storage and Network

Item	Description
USB3.2 Host (USB3)	• 1 channel • Compliance with USB3.2 Gen2 • Maximum rate: 10 Gbps • Supports Control/Bulk/Interrupt/Isochronous transfer • Internal dedicated DMA
USB2.0 Host/Function (USB2)	• 2 channels (ch. 0: Host-Function, ch. 1: Host only) • Compliance with USB2.0 • Support for On-The-Go (OTG) functionality (ch. 0 only) • Supports Control/Bulk/Interrupt/Isochronous transfer • Internal dedicated DMA
PCI Express 3.0 (PCIE)	• 1 channel • PCIe Gen3 • Root complex or Endpoint selectable • 2-lane
Gigabit Ethernet Interface (GBETH)	• 2 channels • Compliant with IEEE802.3 • Compliant with IEEE802.1Qav, IEEE802.1Qat, IEEE802.1AS • Compliant with IEEE1588-2008 with nano second timer in ch. 0 (master) and ch. 1 (slave) • Supports 10BASE, 100BASE, 1000BASE • Supports full duplex and half duplex • Supports RGMII/MII Interfaces
CANFD Interface (RS-CANFD)	• 6 channels • CAN-FD ISO 11898-1 (2015) compliant • Support up to 8 MHz with payload transfer • Message buffer – 64 transmit message buffers per channel – 256 shared buffers for RXMB and FIFO buffers per channel

1.1.3.12 Peripheral Module

Item	Description
I3C Bus Interface (I3C)	• 1 channel • Support for 1.2 V and 1.8 V • Master or Slave mode selectable • Support for the multi-master • Compliant with MIPI I3C v1.0 and I3C Basic v1.0 • The following functions are not supported: – Bridge device (I3C v1.0 and I3C Basic v1.0) – Asynchronous timing control async mode 2 & 3 (I3C v1.0) • Support for DMAC and event linking
I2C Bus Interface (RIIC)	• 9 channels • Master or Slave mode selectable • Support for the multi-master • Support for Standard mode (100 kHz), Fast mode (400 kHz), and Fast mode+ (1 MHz) • Support for DMAC and event linking
Renesas Serial Communication Interface (RSCI)	• 10 channels • 6 communication mode – Asynchronous interfaces – 8-bit clock synchronous interface – Simple IIC (master-only) – Simple SPI – Smart card interface – Simple LIN (expanded SCIX mode) • 32-stage FIFO registers for transmission and reception • Clock sources is select from among four internal clock signals • Bit rate specifiable with the on-chip baud rate generator • Full-duplex and Half-duplex communication • Data length: 7 to 9 bits • Bit rate modulation • Double speed mode • Loopback function to enable self-diagnosis • Support for DMAC and event linking • Support for CRC calculation by the CRC unit
Renesas Serial Peripheral Interface (RSPI)	• 3 channels • SPI transfer facility – Using the MOSI (master out slave in), MISO (master in slave out), SSL (slave select), and RSPCK (SPI clock) signals enables serial transfer through SPI operation (four lines) or clock-synchronous operation (three lines) – Capable of handling serial transfer as a master or slave • Data formats – Switching between MSB first and LSB first – The number of bits in each transfer can be changed to any number of bits from 8 to 16, or 20, 24, or 32 bits. – 32 bits × 16 stage buffers for transmission and reception – Up to for frames can be transmitted or received in a single transfer operation (with each frame having up to 32 bits) • Buffered structure – 16 stages/channels for MOSI and MISO independently – Double buffers for both transmission and reception • RSPCK can be stopped automatically with the reception buffer full for master reception • Support for DMAC and event linking • Support CRC calculation by the CRC unit

Item	Description
CRC calculator (CRC)	• 1 channel • CRC code generation for arbitrary amounts of data in 8-, 16-, or 32-bit units • Select any of four generating polynomials: – $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$ (CRC-32) – $X^{32} + X^{28} + X^{27} + X^{26} + X^{25} + X^{23} + X^{22} + X^{20} + X^{19} + X^{18} + X^{14} + X^{13} + X^{11} + X^{10} + X^9 + X^8 + X^6 + 1$ (CRC-32C) – $X^{16} + X^{15} + X^2 + 1$ (CRC-16) – $X^{16} + X^{12} + X^5 + 1$ (CRC-CCITT) – $X^8 + X^2 + X + 1$ (CRC-8) • Support for RSCI and RSPI interfaces
Serial Communication Interface with FIFO (SCIF)	• 1 channel • Asynchronous mode • Simultaneous transmission and reception (full-duplex communication) supported • Dedicated baud-rate generator • Separate 16-byte FIFO registers for transmission and reception

1.1.3.13 Timer

Item	Description
General PWM Timer (GPT)	• 32 bits × 16 channels • Counting up or down (sawtooth-wave), counting up and down (triangle-wave) selectable for all channels • 2 input/output pins per channel • 2 output compare/input capture registers per channel • For the 2 output compare/input capture registers of each channel, 4 registers are provided as buffer registers and are capable of operating as comparison registers when buffering is not in use. • In output compare operation, buffer switching can be at peaks or troughs, enabling the generation of laterally asymmetrically PWM waveforms. • Registers for setting up frame intervals on each channel (with capability for generating interrupts on overflow or underflow) • Enabling synchronized operation of the several counters between 2 units • Modes of synchronized operation (synchronized, or displaced by desired times for phase shifting) • Generation of dead times in PWM operation • Automatic generation of three-phase PWM waveforms incorporating dead times through the combination of three counters • Starting, clearing, and stopping counters in response to external or internal triggers • Internal trigger sources: Software and compare-match • Generation of triggers for A/D converter conversion • Digital noise filter functions for signals on the input capture and external trigger pins • Event linking by the ELC • Support for phase counting mode
Port Output Enable for GPT (POEG)	• Controlling the output disable for GPT waveform output • Initiation by input level detection of GTETRG pins • Initiation by output disable request from GPT • Initiation by detection of oscillation stop or by software
Compare match timer W (CMTW)	• 32 bits × 8 channels • Compare-match, input-capture input, and output-comparison output are available (ch. 0 to ch. 3) • Interrupt requests can be output in response to compare-match, input-capture, and output-comparison events
Watchdog Timer (WDT)	• 4 channels • A counter underflow can reset the LSI
General Timer (GTM)	• 32 bits × 8 channels • Two operating modes: – Interval timer mode – Free-running comparison mode
Real Time Clock (RTC)	• A 100-year calendar from 2000 to 2099 • BCD code display • Clock source is an oscillator dedicated to RTC (32.768 kHz) • Automatic adjustment function for leap years • Alarm function

1.1.3.14 Audio Interface

Item	Description
Sampling Rate Converter Unit (SCU)	• 10 channels • Sampling Rate : Max.192 kHz • Asynchronous sampling rate conversion is available • Supports resolutions up to 24 bits • High-sound-quality type (THD + N*1 is −132 dB) and general-sound-quality type (THD + N*1 is −96 dB) • Automatically generates antialiasing filter coefficients • Four modules support one, two, four, six, or eight channels, and six modules support one or two channels. *Note 1.* Total harmonic distortion plus noise.
Audio Clock Generator Unit (ADG)	• Supplies clock signals to the SSIU, SCU and SPDIF module
Direct Access Memory Controller for Audio (ADMAC)	• Allows transfer of L/R data via I2S • 29 channels • Controls data transfer between the audio modules (SSIU, SCU)
Serial Sound Interface Unit (SSIU)	• 10 channels for half-duplex communication with transmit or receive function • 5 channels for full-duplex communication (full-duplex pairing: ch. 0 & 9, ch.1 & 2, ch. 3 & 4, ch. 5 & 6, ch. 7 & 8) • Support for I2S, monaural, and TDM audio formats • Support for master and slave functions • Generation of programmable word clocks and bit clocks • Multi-channel formats • Support for 8, 16, 18, 20, 22, 24, and 32-bit data formats • Support for WS (word select) signal continuation with which the WS signal is not stopped • Support for ADMAC
SPDIF Interface (SPDIF)	• 3 channels • Support of IEC60958 standard (stereo and consumer use modes only) • Sampling Rate: 32 kHz, 44.1 kHz, and 48 kHz • Audio word sizes of 16 to 24 bits per sample • Biphase mark encoding • Double buffered data • Parity encoded serial data • Supports DMAC
Pulse Density Modulation (PDM)	• 6 channels • Direction: input • Sampling Rate: 8, 10, 12, 15, 16, 20, 24, 25, 30, 40, or 48 kHz • Capable of filtering 1-bit digital input data and converting them into 20-bit or 16-bit digital data • supports stereo microphone (L/R sampling by rising/falling clock edge) • supports sound activity detector to wake up CPU from WFI • Supports DMAC

1.1.3.15 Analog

Item	Description
A/D Converter (ADC)	• 8 channels • Resolution: 12 bits • Input range: 0 V to 1.8 V • Conversion rate: 2.5 Msps, 2.0 Msps, 1.0 Msps, 0.5 Msps, 0.25 Msps • Operation mode: Single scan, continuous scan, group scan • Condition for starting A/D conversion – Software trigger – Asynchronous trigger: External ADTRG trigger supported – Synchronous trigger: ELC and GPT timers • Interrupt sources: A/D scan end, window compare match, compare match/mismatch, data register overwrite
Temperature Sensor Unit (TSU)	• 1 unit for internal temperature • Includes a 12-bit A/D convertor per unit • Resolution: 0.0625°C/code • Rang: -40°C to125°C • Precision: ±5°C • Conversion Rate: 14.9 ksps • Operation Mode: Single Scan • Condition for measurement start – Software trigger – Synchronous trigger: ELC • Interrupt sources: conversion end, window compare match

1.1.3.16 Security

Item	Description
Secure IP (RSIP)	• Security algorism – Common key encryption: AES – Non-common key encryption: RSA, ECC • Other features – TRNG (true-random number generator) – Hash value generation: SHA-1, SHA-2, SHA-3 – Support of Unique ID

1.1.3.17 Others

Item	Description
Boundary Scan	Boundary scan based on IEEE 1149.1 via JTAG interface is supported. Note that some module pins are not available on this boundary scan.

1.1.3.18 Power Supply Voltage

Item	Description
Power supply voltage	• VDD (core): 0.8 V • VDD (CA55): 0.8 V@Normal Drive, 0.9 V@OverDrive • VDD (DDR IO): 1.1 V/0.6 V (LPDDR4/4X) • VDD (except DDR IO): 1.2 V, 1.8 V, 3.3 V

1.1.3.19 Temperature Range

Item	Description
Temperature range	• Tj: −40°C to +125°C

1.1.3.20 Quality Level

Item	Description
Quality level	• Industrial usage, etc.

1.1.3.21 Package

Item	Description
Package	• 625 pin FCBGA / 21 mm square / 0.8 mm pitch • 529 pin FCBGA / 15 mm square / 0.5 mm pitch

1.1.4 Block Diagram

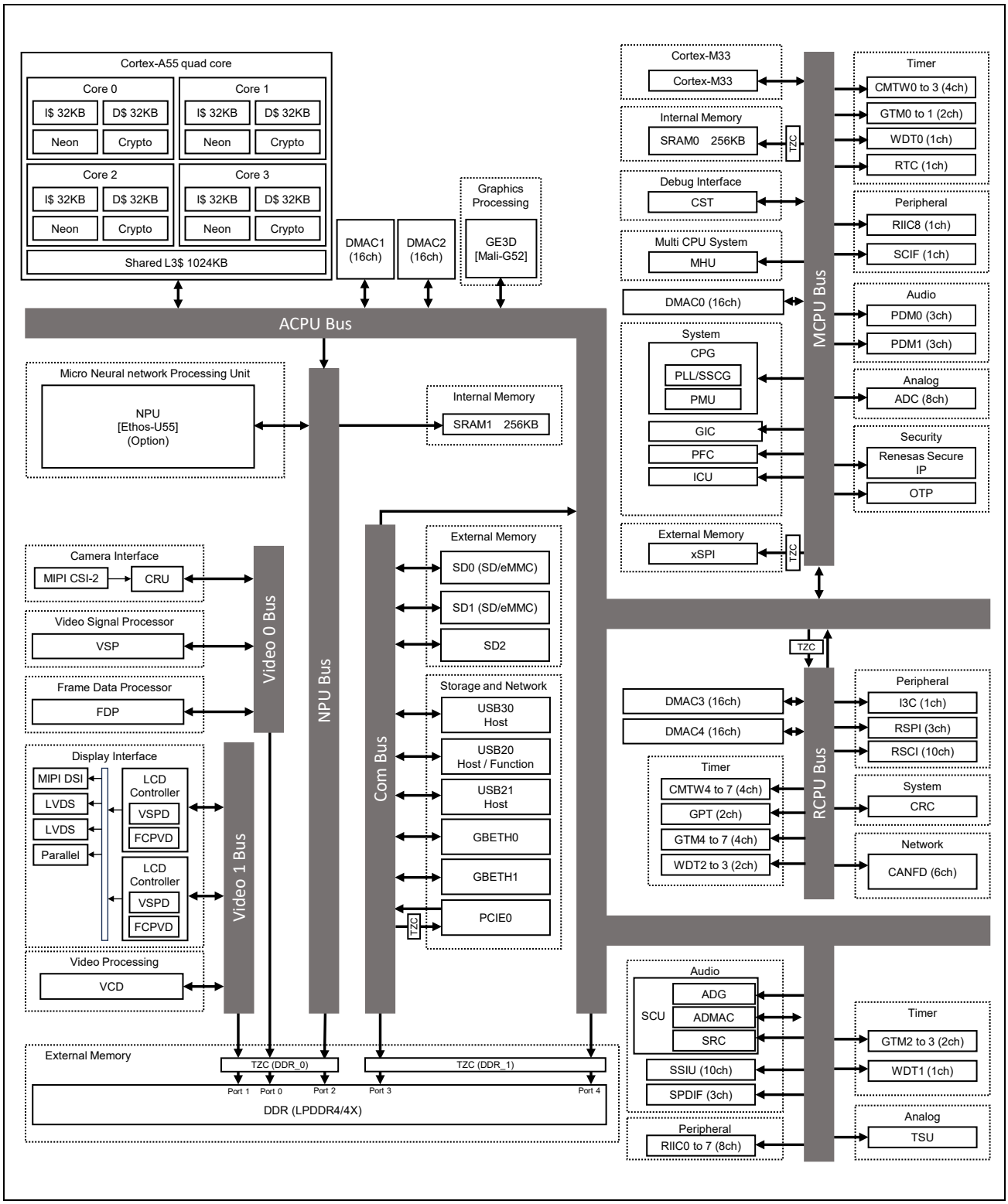


Figure 1.1-2 Block Diagram

Table 1.1-2 List of Units (1/2)

Unit Name	Unit Number	Function
ADC	ADC0	A/D converter
ADG	—	Audio clock generator
ADMAC	—	DMAC for audio
CA55	—	Arm Cortex-A55
CANFD	CANFD0	CAN-FD interface
CM33	—	Arm Cortex-M33
CMTW	CMTW0 to CMTW7	Compare match timer
CPG	—	Clock pulse generator
CRC	—	CRC operation unit
CRU	CRU0	Camera data receive unit (MIPI CSI-2 interface)
CST	—	Debug interface (Arm CoreSight)
DDR	DDR0	LPDDR4/4X controller
DMAC	DMAC0 to DMAC4 (each 16 ch.)	Direct memory access (DMA) controller
NPU	—	Micro Neural network processing unit
ELC	—	Event link controller
FDP	—	Frame Data Processor
GBETH	GBETH0, GBETH1	Gigabit Ethernet interface
GE3D	—	3D graphics engine
GIC	—	Generic interrupt controller
GPT	GPT0, GPT1 (each 16 ch.)	General purpose timer
GTM	GTM0 to GTM7	General timer
I3C	I3C0	I3C bus interface
ICU	—	Interrupt control unit
LCDC	—	LCD controller
MHU	—	Message handling unit
OTP	—	One time programmable memory
PCIE	PCIE0	PCIe Express 3.0 interface
PCU	—	Power control unit
PDM	PDM0, PDM1	Pulse density modulation (PDM) interface
PFC	—	Pin function controller
POEG	POEG0, POEG1	Port output enable for GPT
PMU	—	Power management unit
PWC	—	Power sequence controller
RIIC	RIIC0 to RIIC8	I2C bus interface
RSCI	RSCI0 to RSCI9	Serial communication interface
RSPI	RSPI0 to RSPI2	Serial peripheral interface
RTC	—	Real time clock
SCIF	SCIF0	Serial communication interface with FIFO
SD	SD0 to SD2	SD/MMC host interface
Secure IP	—	Trusted secure IP

Table 1.1-2 List of Units (2/2)

Unit Name	Unit Number	Functional Overview
SRAM	SRAM0 to SRAM11	SRAM
SRC	—	Sampling rate controller
SSIU	—	Serial sound interface unit
SYC	—	System counter
SYS	—	System controller
SYSTEM BUS	—	Internal bus
ACPU Bus	—	A bus connected to Cortex-A55, DDR memory controllers, SRAM, and its peripheral units
RCPU Bus	—	A bus connected to peripheral units
MCPU Bus	—	A bus connected to Cortex-M33, SRAM, its peripheral units, and the system control units
NPU Bus	—	A bus connected to NPU, SRAM, and DDR memory controllers
Video 0 Bus Video 1 Bus	—	A bus connected to image processing units and DDR memory controllers
Com Bus	—	A bus connected to communication interface units and DDR memory controllers
TSU	TSU1	Temperature sensor unit
TZC	—	CoreLink™ TrustZone Address Space Controller
USB2	USB20, USB21	USB2.0 host / function interface
USB3	USB30	USB3.2 host interface
VCD	—	H.265/H.264 multi codec
VSP	—	Video Signal Processor
WDT	WDT0 to WDT3	Watchdog timer
xSPI	xSPI0	xSPI controller

SECTION 1 OVERVIEW

1.2 Pin

This section describes the pins of this LSI.

1.2.1 Pin Assignment

Refer to another excel file for the “ball view” about pin assignment of this LSI.

(Please double-click the icon on the right side)

1.2.2 External Pins and Multiplexed Functional Pins

Refer to attached excel file for the “pin function list” about information of external pins and multiplexed functional pins of this LSI.

(Please double-click the icon on the right side)

SECTION 1 OVERVIEW

1.3 Clocks

This section describes the clocks of this LSI.

1.3.1 Overview

The main functions of the CPG are to generate and control clocks for the respective units, to generate and control resets, to control booting, to control low power consumption, and to control the power supply domain.

Table 1.3-1 Types and Overview of Unit Clocks

Function	Subset	Details
Clock Pulse Generator	CPG	Clock generating and control functions
		Reset sequential control functions
		Boot sequential control functions
Power Management Unit	PMU	Power sequential control functions

1.3.2 Clock Generating and Control Functions

The following functions are provided.

- The clocks to be supplied to various units are generated from external input clocks or PLL output clocks. Especially when the CA55 is cold boot, the CA55 operating frequency by the external pin is set.
- The selection of frequency divider ratio is implemented (the setting can be changed by the register).
- The clock path is selected by the clock selector (the setting can be changed by the register).
- Clock supply ON/OFF control (the setting can be changed by register)
- SSCG control of PLL, multiplication ratio setting, ON/OFF control (the setting can be changed by the register)

Table 1.3-2 PLL Overview

PLL	Details	Initial Frequency	SSCG Default Value
PLLCM33	For the MCPU bus and various units (PD_AWO area)	1,600 MHz	OFF (fixed)
PLLCLN	For buses other than the MCPU bus and various units	1,600 MHz	OFF (fixed)
PLLDTY	For buses other than the MCPU bus and various units	1,600 MHz	Depends on MD_CLKS
PLLCA55	CA55	1,700 MHz*1	Depends on MD_CLKS
PLLVDO	CRU / GE3D / CA55	1,260 MHz	Depends on MD_CLKS
PLLETH	GBETH / NPU / DSI	1,000 MHz	OFF (fixed)
PLLDSI0	DSI / LCDC / LVDS	297 MHz	OFF (fixed)
PLLDSI1	DSI / LCDC / LVDS	297 MHz	OFF (fixed)
PLLDDR0	For DDR ch. 0	800 MHz	OFF (fixed)

Note 1. When the CA55 is cold boot, the value is set by the external pin. The operation frequency of CA55 is selected by the BOOTPLLCA0 (pin name: ps2)*2 and BOOTPLLCA1 (pin name: ps3)*2 settings.

Note 2. This terminal is used as a dual-use terminal.

1.3.3 Clock Pin Specifications

Table 1.3-3 List of Pin Functions (1/2)

Pin Name	Input/ Output	Function	Frequency*1	Supply Source/ Destination Unit
LSI clock				
QXTAL	Output	24 MHz main clock	24 MHz	CPG
QEXTAL	Input	24 MHz main clock	24 MHz	CPG
RTXOUT	Output	32.768 kHz real-time clock	32.768 kHz	RTC
RTXIN	Input	32.768 kHz teal-time clock	32.768 kHz	RTC
AUDIO_XTAL	Output	4 to 48 MHz audio clock (Internal OSC mode)	48 MHz	ADG
AUDIO_EXTAL	Input	4 to 48 MHz audio clock (Internal OSC mode) 4 to 50 MHz audio clock (Input clock from the external device)	50 MHz	ADG
AUDIO_CLKB*2	Input	4 to 50 MHz audio clock-in	50 MHz	ADG
AUDIO_CLKC*2	Input	4 to 50 MHz audio clock-in	50 MHz	ADG
AUDIO_CLKOUT*2	Output	2 to 25 MHz audio clock-out (Half frequency of input clock)	25 MHz	ADG
Debugger interface				
TCK_SWCLK	Input	Test clock pin for the on-chip emulator functions as the SWCLK pin in serial wire debug (SWD) mode	20 MHz	CST
Expanded serial peripheral interface (xSPI)				
XSPI0_CKP*3	Output	Clock output pins (positive)	133 MHz	xSPI
XSPI0_CKN*3	Output	Clock output pins (negative)	133 MHz	xSPI
SD/MMC interface				
SD0CLK	Output	Output the clock signal to external SD/MMC device	200 MHz	SD0
SD interface				
SD1CLK	Output	Output the clock signal to external SD device	200 MHz	SD1
SD2CLK	Output	Output the clock signal to external SD device	200 MHz	SD2
PCIe Gen3				
PCIE_REFCLKP0	I/O	Differential reference clock (positive), 100 MHz ±300 ppm	100 MHz	PCIE0
PCIE_REFCLKN0	I/O	Differential reference clock (negative), 100 MHz ±300 ppm	100 MHz	PCIE0
Gbit Ethernet ch. 0				
ET0_MDC	Output	Output management data clocks	2.5 MHz	GBETH0
ET0_RXC_RXCLK	Input	RX clocks	125 MHz	GBETH0
ET0_TXC_TXCLK	I/O	TX clocks	125 MHz	GBETH0
GbEthernet ch. 1				
ET1_MDC	Output	Output management data clocks	2.5 MHz	GBETH1
ET1_RXC_RXCLK	Input	RX clocks	125 MHz	GBETH1
ET1_TXC_TXCLK	I/O	TX clocks	125 MHz	GBETH1
MIPI-DSI				
DSI_DPCLK	Output	Output clocks (positive)	750 MHz	DSI
DSI_DNCLK	Output	Output clocks (negative)	750 MHz	DSI
LVDS ch. n (n = 0 to 1)				
LVDSn_CLK_P	Output	Output clocks (positive)	87MHz	LVDS
LVDSn_CLK_N	Output	Output clocks (negative)	87MHz	LVDS
MIPI-CSI2 ch. n (n = 0)				
CSIn_CLKP	Input	Input clocks (positive)	1050 MHz	CRUn
CSIn_CLKN	Input	Input clocks (negative)	1050 MHz	CRUn

Table 1.3-3 List of Pin Functions (2/2)

Pin Name	Input/ Output	Function	Frequency*1	Supply Source/ Destination Unit
Serial peripheral interface (RSPI) ch. n (n = 0, 1, 2)				
RSPCKn	I/O	Synchronous clock signal	50 MHz	RSPIn
Serial communication interface (RSCI) ch. n (n = 0 to 9)				
SCKn	I/O	Clock pins (simple SPI mode)	37.5 MHz	RSCIn
SCLn	I/O	I2C clocks (simple I2C mode)	400 kHz	RSCIn
I2C bus interface (RIIC) ch. n (n = 0 to 8)				
SCLn	I/O	Clock pins with N-ch open drain	1 MHz	RIICn
I3C bus interface (I3C)				
SCL30	I/O	Clock pin	3 MHz	I3C0
Pulse density modulation interface (PDM) ch. n (n = 0 to 5)				
PDMCLK00	Output	Output PDM sampling clocks	2.4 MHz	PDM0
PDMCLK01	Output	Output PDM sampling clocks	2.4 MHz	PDM0
PDMCLK02	Output	Output PDM sampling clocks	2.4 MHz	PDM0
PDMCLK10	Output	Output PDM sampling clocks	2.4 MHz	PDM1
PDMCLK11	Output	Output PDM sampling clocks	2.4 MHz	PDM1
PDMCLK12	Output	Output PDM sampling clocks	2.4 MHz	PDM1
Serial sound interface (SSIU) ch. n (n = 0 to 9)				
SSIn_SCK	I/O	Output clocks	12.5 MHz	SSIU

Note 1. The clock frequencies are either variable or fixed. In the case of the variable clocks, the maximum frequencies which can be output are listed for the output clocks and the maximum frequencies which can be input to this LSI are listed for the input clocks.

Note 2. For use of these pins, the function selection of multiplexed function pins is required. For details, refer to **4.2 Pin Function Controller (PFC)**.

Note 3. CKP and CKN waves have opposite phase.

SECTION 1 OVERVIEW

1.4 Reset

This section describes resets of this LSI.

1.4.1 Overview

Each reset source generates a reset signal. The types of reset are described below.

For further details regarding the resets, see **4.4 Clock Pulse Generator (CPG)**.

1.4.1.1 System Reset (External QRESN Pin)

Asserting the reset signal on the external QRESN pin initializes the whole LSI.

1.4.1.2 Debug Reset (Software Reset by CoreSight)

When the external QRESN, TRSTN, and MD_BOOT3 pins are at the high level, the JTAG debugger is capable of resetting the CoreSight module (except the DAP) and Cortex-A55.

1.4.1.3 Error Reset

A reset can be generated in response to the maskable error sources. Generation of a reset in response to the main sources of errors due to the system such as a WDT timeout error, ECC error in internal SRAM, or a bus access error, as well as to error sources in individual units, is possible.

1.4.1.4 Resetting of Individual Units

Switching reset signals for individual units off and on can be realized by switching a reset pin for the unit off and on according to enable conditions. Sources of control for switching reset sources off and on are listed below.

- Off and on control by a register setting
- Reset control in a boot sequence
- Debug reset control
- CA55 cold reset control
- CM33 cold reset control

1.4.1.5 CA55 Reset

The CA55 has several types of reset, differing in terms of hardware or software control and the CA55 reset signal through which control is applied. At the time of a reset, the 4 cores and the cluster require handshake control by the individual P-channels.

■ **Hardware control: by the CPG**

System cold reset (by the function or the debugger),
WDT error cold reset

■ **Software control: by the CM33**

Cold reset (normal/debug mode), warm reset (cluster, core 0/1/2/3)

1.4.1.6 CM33 Reset

The CM33 has several types of reset, differing in terms of hardware or software control and the CM33 reset signal through which control is applied.

■ **Hardware control: by the CPG**

System cold reset, WDT error cold reset

■ **Software control:**

Cold reset (normal/debug mode), warm reset

SECTION 1 OVERVIEW

1.5 Interrupts

This section describes the interrupt functions of this LSI.

1.5.1 Overview

The interrupt controller accepts interrupt requests from peripheral modules and from external pins including the external DMA request pins. An interrupt accepted by the interrupt controller is set either as an interrupt for the CPU or as an activating trigger signal for the DMAC and ELC.

Table 1.5-1 Interrupt Overview

Item		Description
Interrupts	Interrupt destinations	• CA55 cores 0, 1, 2, and 3 • CM33 • Five DMAC units • ELC
	Peripheral function interrupts	Interrupts from peripheral modules detected by edge or level
	External pin interrupts	• Interrupts from pins: IRQ0 to IRQ15 • Number of sources: 16 • Detection: One of four methods can be set for each source. Low level, falling edge, rising edge, both rising and falling edges. • Digital noise filter supported
	Interrupt between CPU cores	Mutual interrupts between the four CA55 cores and CM33 with or without safety.
	Error events	Error events from the CPU and peripherals are captured and merged to interrupts with masking. All are masked in the initial state (no interrupts will be generated).
	DMAC control	Interrupts can be assigned as DMA activation sources.
	External DMA request	• DREQ (external DMA request) pins can output DACK (DMA acknowledgement) signals and TEND signals when DMA transfer is completed. • Number of sources: 5 (one per DMAC unit) • Digital noise filter supported
Non maskable interrupts	NMI pin interrupts	• Interrupts from NMI pin • Interrupt detection: Falling edge, rising edge • Digital noise filter supported
Restoration from the sleep state		Restoration due to non-maskable interrupt and non-masked interrupt sources

1.5.2 Overview of the ICU

The interrupts of this LSI are controlled by the interrupt control unit (ICU). An overview of the ICU function is given below. For details, see **4.6 Interrupt Controller**.

1.5.2.1 Interrupt Control

- Interrupt output for individual CPUs

  This LSI has a large number of interrupts, which are classified as follows. As stated, some interrupts are not assigned in common to all of the CA55 and CM33.

  - Interrupts common to all CPUs (common)
  - Independent interrupts for the CA55 and CM33 (system)
  - Interrupts other than those listed above
    Each interrupt for the CA55 cores is assigned as an SPI because the number of interrupts the CA55 supports is fewer than the maximum of 960 interrupts (single).
    Interrupts for the CM33 are selectable by software, and fewer than 480 interrupts are assigned as SPIs (select).

Table 1.5-2 Assignment of Interrupts to the Three Types of CPU

SPI No.	CA55	CM33
0 to 251	Common	
252 to 352	SYSTEM(CA55) SINGLE(CA55)	SYSTEM(CM33)
353 to 479		SELECT(CM33)
480 to 959		—

- Control of error interrupts
  - Generation of an extended pseudo error interrupt under software control
  - This error interrupt signal is connected as a single interrupt to the CM33 and CA55, respectively.
- The software interrupt is generated by writing to a register in the ICU.
- The ICU applies masking control of the interrupt for the CM33 by using the MASK signal externally input to the ICU.

1.5.2.2 Event Source Control

- Bundled RAM error interrupt control: the error source can be checked by reading a register.
- Bundled bus error interrupt control: the error source can be checked by reading a register.
- Bundled GPT control: the error source can be checked by reading a register.
- Control of error interrupt assignment to the CA55 and CM33 cores
  - Error interrupts for the CA55 cores
    All error events can be bundled into one interrupt for notification of a CA55. Masking an interrupt so that it is not conveyed is also possible.
  - Error interrupts for the CM33 cores
    All error events can be bundled into one interrupt for notification of a CM33. Masking an interrupt so that it is not conveyed is also possible.

  Units that produce error event signals
  CA55, CM33, SRAM, SYSTEM BUS, DDR, ICU, CPG, GPT, WDT, and ADC
- Retention of the error source flags in the SYS register
- Input selection control of TINT

  Pin functions multiplexed on 82 pins can be used as TINT interrupts. Interrupts from TINT0 to TINT31 are selectable.
- Exclusive output control of signals as interrupt signals or DMAC triggers

  The control when a source is assigned as both an interrupt and a DMA request is exclusive, and either an interrupt or DMA request will be enabled.
- Exclusive output control of signals as interrupt signals or event triggers

  The control when a source is assigned as both an interrupt and an event signal output is exclusive, and either an interrupt or event signal output will be enabled.

1.5.2.3 DMAC Control

- DMA request signals are output from the ICU to the DMAC.
- The ICU has functionality to assign the target units per DMAC channel, and outputs the events for the assigned units as DMA requests.
- Ack signals from the DMAC are output from the ICU to some units.
- The generation of software DMAC requests is controlled by writing to a register in the ICU.
- The channel settings are cleared in response to the DMAC transfer completion interrupt.
- The channel settings are controlled to be the same for a DMAC request.

1.5.2.4 Event Output Control

- Event signals that are input to the ICU are output to the target units.
- The ICU has functionality to assign the target units for event output, and outputs the event signals for the assigned units.
- The generation of software events is controlled by writing to a register in the ICU.

1.5.2.5 Error Output

The ICU outputs the underflow error produced by the WDT and errors for which the source is other than the WDT to the CPG.

The CPG uses these signals to control the system reset and resets for individual CPUs as well as the external WDTUDFCM and WDTUDFCA pins in response to underflow errors of the WDT for the CM33 and CA55.

1.5.2.6 Register Access Security Control

Registers in the ICU are classified into two groups (Gr0 and Gr1) and security control can be applied per group.

1.5.3 System Configuration Related to Interrupts

Figure 1.5-1 is a schematic diagram of the system related to interrupts in this LSI.

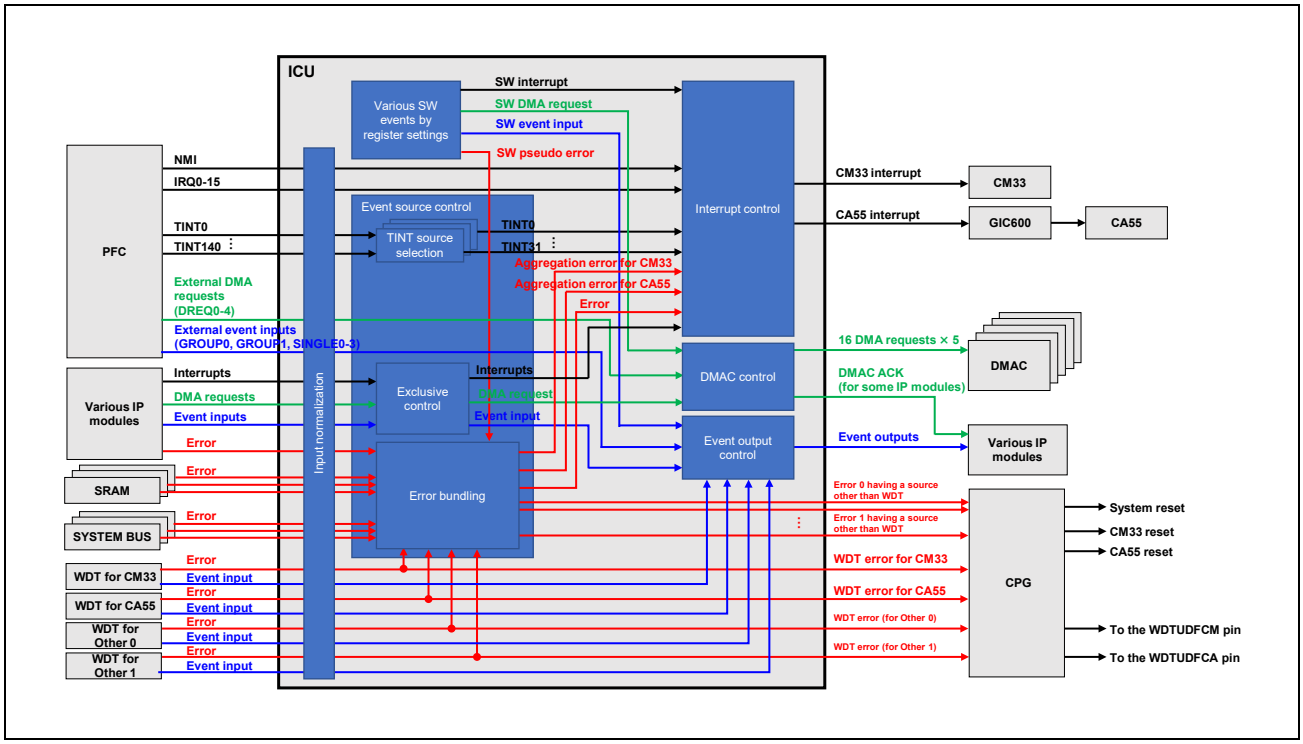


Figure 1.5-1 Schematic Diagram of the System Related to Interrupts

1.5.4 Error Processing

1.5.4.1 Notifying the CA55 and CM33 Cores of Errors

All error events can be aggregated as interrupts for conveyance to the CA55 and CM33 cores.

Masking an interrupt so that it is not conveyed is also possible. The CPU to which the interrupt was conveyed checks the error source retention register to identify the source of the error, and proceeds with processing corresponding to the source.

1.5.4.2 Resets Due to Errors

Table 1.5-3 shows the relationship between errors for which the source is a WDT or other than a WDT and each type of reset, and the WDTUDFCM and WDTUDFCA pins. In the CPG, whether a reset is applied to the CPU, a system reset is applied to the whole chip, or the WDTUDFCM and WDTUDFCA pins are asserted can be selected by masking.

Table 1.5-3 Errors and Resets

Error Source	CA55 Cold Reset	CM33 Cold Reset	System Reset	Asserting WDTUDFCA (Hardware Control)	Asserting WDTUDFCM (Hardware Control)
WDT for CA55 (all cores)	✓	—	✓	✓	—
WDT for CM33	—	✓	✓	—	✓
WDT for Other 0	—	—	✓	—	—
WDT for Other 1	—	—	✓	—	—
Error 0 having a source other than a WDT	—	—	✓	—	—
Error 1 having a source other than a WDT	—	—	✓	—	—
…	—	—	✓	—	—
Error N having a source other than a WDT	—	—	✓	—	—

SECTION 1 OVERVIEW

1.6 Power Management

This section describes the power management of this LSI.

1.6.1 Overview

The PMU controls the power off sequence except for the PD_AWO area.

When the CM33 is coldbooted, the power supply to areas other than the PD_AWO area can be shut down.

When the CA55 is coldbooted, the power must be always on in all power supply areas.

1.6.2 Power Domain

The power domain is classified into two main units. For details, see **4.4 Clock Pulse Generator (CPG)** and **4.5 Power Management Unit (PMU)**.

Table 1.6-1 PD_AWO (Always On)

Item	Target Unit
System	SYSTEM BUS, CPG, ICU, PFC, ROM, OTP, CST, secure IP, DMAC (16 ch.), TZC, etc.
CPU	Cortex-M33 (FPU/DSP extension)
SRAM	256 Kbyte
Timer	GTM (2 ch.), CMTW (4 ch.), WDT (1 ch.), RTC
Interface	xSPI, RIIC (1 ch.), SCIF (1 ch.), GPIO (141 pins)/IRQ/NMI, ADC (8 ports), PDM (6 ch.)

Table 1.6-2 PD_OTHERS (Including PD_CA55 and PD_DDR)

Item	Target Unit
System	SYSTEM BUS, CPG, PFC, TSU (1 ch.), DMAC (64 ch.), TZC, etc.
CPU	Cortex-A55
AI accelerator	Ethos-U55
Video & Graphics	GE3D*1, VCD, CRU, DSI, VSP, FDP, LCDC, LVDS
Audio	ADG, SCU, ADMAC, SSIU, SPDIF
SRAM	256 Kbytes
Timer	GTM (6 ch.), CMTW (4 ch.), WDT (3 ch.), GPT (16 ch.)
Interface	DDR (1 ch.), eMMC/SD (2 ch.), SD (1 ch.), USB3 (1 ch.), USB2 (2 ch.), GBETH (2 ch.), PCIE, I3C0 (1 ch.), RIIC (8 ch.), RSCI (10 ch.), RSPI (3 ch.), CANFD (6 ch.)

Note 1. This unit is optional.

CAUTION

The state where PD_AWO = OFF and PD OTHERS = ON is prohibited.

To avoid the power to be accidentally supplied in the above state, measures such as placing a diode between the AWO-OTHERS power supplies on the mounting board are required.

1.6.3 Lower Power Management

The low-power management of the system is supported. In some low-power modes, handshake control is performed with the CPU.

- PD_OTHERS area power control
- SRAM power-saving mode control
- Cortex-A55 sleep mode
- Cortex-M33 sleep mode
- Module standby mode
- Low frequency mode
- Software standby mode

For the operation procedure, refer to **4.4 Clock Pulse Generator (CPG)**.

SECTION 1 OVERVIEW

1.7 Internal Bus

This section describes the internal bus configuration of this LSI.

1.7.1 Overview

1.7.1.1 Features

The bus system of this LSI provides a physical address space of 64 Gbytes (address bus width of 36 bits). The internal bus of this LSI incorporates Arm CoreLink NIC-400, etc., and controls the following bus functions.

Security control:

Security attribute re-setting and security level determination

Address translation:

35-bit or 36-bit address space access

Interrupt generation:

AXI bus error interrupt generation

Unit state detection:

Slave unit stop state detection

NOTE

The security control function is only valid for secure products. For information on secure products, please contact our sales representative.

1.7.1.2 LSI Internal Bus Configuration

The internal bus of the LSI consists of the ACPU bus, RCPU bus, MCPU bus, NPU bus, Video 0 bus, Video 1 bus, and COM bus. The configuration of the buses is shown in **1.1 Overview**.

ACPU bus:

A bus connected to CA55, DDR memory controllers, and its peripheral units

RCPU bus:

A bus connected to peripheral units

MCPU bus:

A bus connected to CM33, SRAM, its peripheral units, and the system control units

NPU bus:

A bus connected to Cognitive Engine, and DDR memory controllers

Video 0 bus and Video 1 bus:

A bus connected to image processing units and DDR memory controllers

COM bus:

A bus connected to communication interface units and DDR memory controllers

1.7.2 Accessible Areas

In the bus system of this LSI, each bus master unit can only access the areas that are used for register access or data transfer. **Table 1.7-1** shows the areas that can be accessed from each master.

Table 1.7-1 Accessible Areas (1/3)

Slave Unit	Master Unit																			
	CA55 (Main)	CA55 (Peripheral)	CM33 (C-AHB)	CM33 (S-AHB)	DMAC (0,1,2,3,4)	CST (AXI-AP)	CST (ETR)	GE3D (Mali-G52)	VSP	FDP	NPU- (Ethos-U55)	SD (0,1,2)	PCIE0	USB30	USB20, USB21	GBETH (0,1)	CRU0 (Video, Statistics)	LCDC (0,1)	DSI	VCD
CPG	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
SYS	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
PFC	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
ICU	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
CST	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
OTP	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
MHU	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
TZC400 (xSPI)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
TZC400 (SRAM0)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
SRAM0 (Reg)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
SRAM0 (Memory)	✓	×	✓	✓	✓	✓	✓	✓	×	×	×	✓	✓	✓	✓	✓	×	×	×	×
PDM (0,1)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
xSPI (REG)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
xSPI (Memory)	✓	×	×	✓	✓	✓	✓	✓	×	×	×	✓	✓	✓	✓	✓	×	×	×	×
DMAC0	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
GTM (0,1)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
CMTW (0,1,2,3)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
SCIF	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
WDT0	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
ADC	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
RTC (Read Only)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
RTC	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
RIIC8	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
DMAC (3,4)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
I3C	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
CANFD	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
RSPI (0,1,2)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
RSCI	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
GTM (4,5,6,7)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
CRC	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
GPT (0,1)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×

Table 1.7-1 Accessible Areas (2/3)

Slave Unit	Master Unit																			
	CA55 (Main)	CA55 (Peripheral)	CM33 (C-AHB)	CM33 (S-AHB)	DMAC (0,1,2,3,4)	CST (AXI-AP)	CST (ETR)	GE3D (Mali-G52)	VSP	FDP	NPU- (Ethos-U55)	SD (0,1,2)	PCIE0	USB30	USB20, USB21	GBETH (0,1)	CRU0 (Video, Statistics)	LCDC (0,1)	DSI	VCD
POEG0A	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
POEG0B	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
POEG0C	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
POEG0D	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
POEG1A	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
POEG1B	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
POEG1C	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
POEG1D	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
CMTW (4,5,6,7)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
WDT (2,3)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
PCIE0	✓	✓ *1	×	✓	✓	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×
CRU	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
VSP	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
FDP	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
DSI (LINK)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
DSI (PHY)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
LCDC0 (VSPD)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
LCDC0 (FCPVD)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
LCDC0 (DU)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
LCDC1 (VSPD)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
LCDC1 (FCPVD)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
LCDC1 (DU)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
VCD (VLC)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
VCD (CE)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
VCD (FCPC)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
LVDS	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
USB30 (Host)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
USB30 (PHY)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
USB20 (Host,Function)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
USB20 (PHY)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
USB21 (Host)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
USB21 (PHY)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
SDHI (0,1,2)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
GBETH (0,1)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
NPU	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
SRAM1 (Reg)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
SRAM1 (Memory)	✓	×	✓	✓	✓	✓	✓	✓	×	×	✓	✓	✓	✓	✓	✓	×	×	×	×

Table 1.7-1 Accessible Areas (3/3)

Slave Unit	Master Unit																			
	CA55 (Main)	CA55 (Peripheral)	CM33 (C-AHB)	CM33 (S-AHB)	DMAC (0,1,2,3,4)	CST (AXI-AP)	CST (ETR)	GE3D (Mali-G52)	VSP	FDP	NPU- (Ethos-U55)	SD (0,1,2)	PCIE0	USB30	USB20, USB21	GBETH (0,1)	CRU0 (Video, Statistics)	LCDC (0,1)	DSI	VCD
DDR (MEMC)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
DDR (DDRPHY)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
DMAC (1,2)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
GIC	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
GE3D	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
SCU	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
ADG	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
SSIU	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
ADMAC	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
TZC400 (RCPU Bus)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
TZC400 (DDR_0)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
TZC400 (DDR_1)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
TZC400 (PCIE0)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
SYC	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
GTM (2,3)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
TSU	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
SPDIF (0,1,2)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
WDT1	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
RIIC (0,1,2,3,4,5,6,7)	✓	✓	×	✓	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	×	×
DDR (port 0 (VIDEO0))	×	×	×	×	×	×	×	×	✓	✓	×	×	×	×	×	×	✓	×	×	×
DDR (port 1 (VIDEO1))	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	✓	✓	✓
DDR (port 2 (NPU))	×	×	×	×	×	×	×	×	×	×	✓	×	×	×	×	×	×	×	×	×
DDR (port 3 (COM))	×	×	×	×	×	×	×	×	×	×	×	✓	✓	✓	✓	✓	×	×	×	×
DDR (port 4 (ACPU))	✓	×	×	✓	✓	✓	✓	✓	×	×	×	×	×	×	×	×	×	×	×	×

Note: Using paths marked with × in the table is prohibited. If access to such a path is attempted, incorrect operation may occur.

Note: For details on access control of each unit and bus, refer to CPG (MSTOP Registers) and SYS (Master Access Control/Slave Access Control Registers).

Remarks: ✓: Accessible path, ×: Prohibited path

Note 1. The CA55 peripheral IF can only access the PCIE0 register regions (1340_0000h to 1340_FFFFh).

1.7.3 Bus System Control

1.7.3.1 Security Control

1.7.3.1.1 Re-Setting the Security Attribute Output from Bus Masters

This facility is for re-setting the security attributes of bus transactions that are by bus master units. Specifically, use the master access control register (SYS_MSTACCCTLn) to re-set the security attributes. For details on the SYS_MSTACCCTLn register, see **4.3 System Controller (SYS)**.

NOTE

The bus transaction signals ARPROT[1:0] and AWPROT[1:0] indicate the security attributes, which can be used to protect against illicit transactions.

- ARPROT[0] and AWPROT[0] being 0 or 1 respectively indicate non-privileged access or privileged access.
- ARPROT[1] and AWPROT[1] being 0 or 1 respectively indicate secure access or non-secure access.

The ARPROT and AWPROT signals are prescribed in as part of the AXI protocol. For details, see the AMBA AXI and ACE Protocol Specifications published by Arm Ltd.

Use the SYS_MSTACCCTLn register to make the following settings.

- Selection of security attribute source

  Selection of whether to use the unchanged values of ARPROT[1:0] or AWPROT[1:0] of the ARPROT[2:0] or AWPROT[2:0] signals, which are output from the bus master units, or to re-set ARPROT[1:0] or AWPROT[1:0] according to the settings of the SYS_MSTACCCTL register
- Value to be re-set for ARPROT[1:0] or AWPROT[1:0]

1.7.3.1.2 Determining the Security Levels of Bus Slaves

This facility is for enabling or disabling access to a bus slave unit by comparing the security attribute that is input to the bus slave unit in a bus transaction with the security setting of the given slave. The security level can be set in one of the following ways. The applicable way depends on the slave.

- Setting by using the SL[1:0] bits in the slave control register (SYS_SLVACCCTLn)
  For details on the SYS_SLVACCCTLn register, see **4.3 System Controller (SYS)**.
- Setting by using the nsaid_wr_en[31:16] and nsaid_rd_en[15:0] bits in the region ID access register (REGION_ID_ACCESS_<n>: n = 0 to 2) of the TrustZone address space controller (TZC). For details on the TZC, see **3.5 TrustZone Address Space Controller (TZC)**.

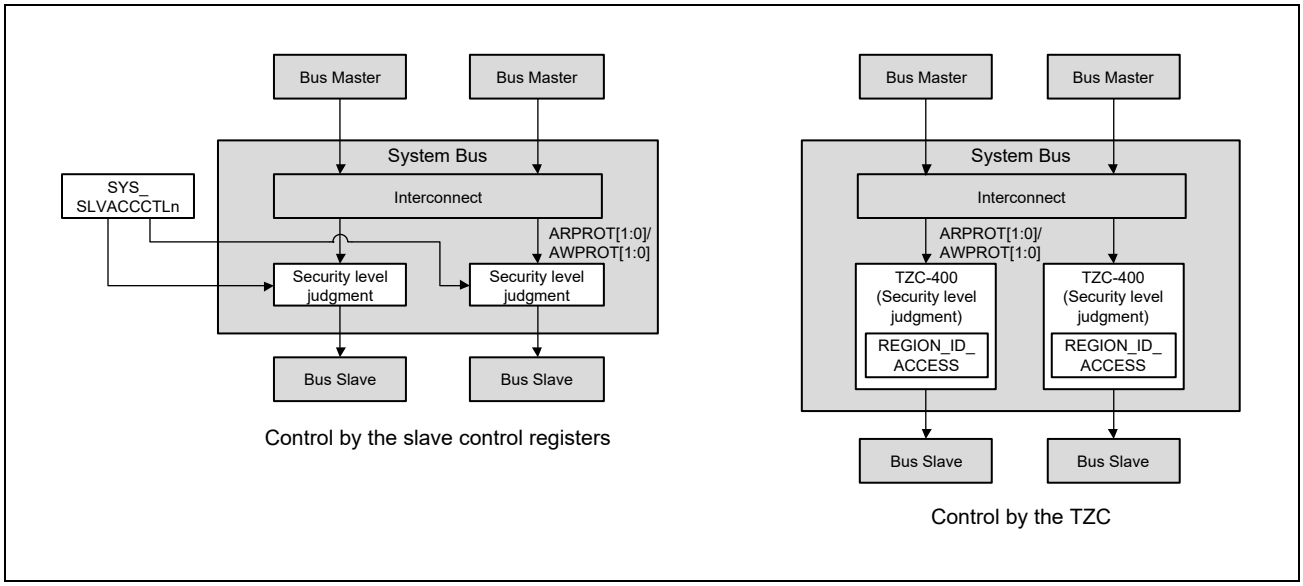


Figure 1.7-1 Security Level Determination Ways

NOTE

The bus transaction signals ARPROT[1:0] and AWPROT[1:0] indicate the security attributes, which can be used to protect against illicit transactions.

- ARPROT[0] and AWPROT[0] being 0 or 1 respectively indicate non-privileged access or privileged access.
- ARPROT[1] and AWPROT[1] being 0 or 1 respectively indicate secure access or non-secure access.

The ARPROT and AWPROT signals are prescribed in as part of the AXI protocol. For details, see the AMBA AXI and ACE Protocol Specifications published by Arm Ltd.

Table 1.7-2 shows the correspondence between the settings of the slave control register and the security levels for input transactions. **Table 1.7-3** shows the correspondence between the settings of the region ID access register of the TZC and the security levels for input bus transactions.

If comparison indicates that access is disabled, a bus error is generated in response to the corresponding attempted bus transaction. Stopping of a slave unit is detected by reference to the module stop state (MSTOP) setting in the CPG. Attempted access to a slave unit in the module stop state leads to a bus error.

Table 1.7-2 Security Levels Set by Using the Slave Control Register

	Security Attribute for Input Bus Transactions			
Setting of SL[1:0]	Non-privileged Non-secure	Privileged Non-secure	Non-privileged Secure	Privileged Secure
00b	Access allowed	Access allowed	Access allowed	Access allowed
01b	Access not allowed	Access allowed	Access allowed	Access allowed
10b	Access not allowed	Access not allowed	Access allowed	Access allowed
11b	Access not allowed	Access not allowed	Access not allowed	Access allowed

Table 1.7-3 Security Levels Set by Using the TZC

Setting of nsaid_wr_en[31:16] and nsaid_rd_en[15:0]	Security Attribute for Input Bus Transactions			
	Non-privileged Non-secure	Privileged Non-secure	Non-privileged Secure	Privileged Secure
000Fh	Access allowed	Access allowed	Access allowed	Access allowed
000Bh	Access not allowed	Access allowed	Access allowed	Access allowed
0003h	Access not allowed	Access not allowed	Access allowed	Access allowed
0002h	Access not allowed	Access not allowed	Access not allowed	Access allowed

1.7.3.2 Address Translation

1.7.3.2.1 36-Bit Address Space Access

This LSI has bus master units that can handle up to a 36-bit address space and bus master units that can only handle up to a 32-bit address space (actual size of 4 Gbytes). This function enables a bus master unit that can only handle up to a 32-bit address space (actual size of 4 Gbytes) to access an address space of greater than 4 Gbytes. The target bus master units are as follows.

1. Up to a 35-bit address space (access to DDR)
   - SD0, SD1, SD2
   - GBETH0, GBETH1
   - USB20, USB21
   - USB30
   - VCD
   - DSI
   - LCDC0, LCDC1
   - NPU
   - VSP
   - FDP
2. Up to a 36-bit address space (access to DDR and PCIE0)
   - DMAC0, DMAC1, DMAC2, DMAC3, DMAC4

This function translates bus transaction addresses that are output from a bus master unit in 1-Gbyte units according to the settings of the address offset registers (SYS_AOFn), as shown in the figure below. For details on the SYS_AOFn registers, see **4.3 System Controller (SYS)**.

- In the SYS_AOF registers, set bits [35:30] or [34:30] of the remapping destination address in each 1-Gbyte space corresponding to bits [31:30] of the address that is output from the bus master unit.
- Select the 6-bit value of the SYS_AOF register corresponding to the upper two bits (bits [31:30]) of the 32-bit address that is output from the bus master unit.
- In the selected 6-bit value, use the lower two bits to overwrite bits [31:30] of the original address that was output from the bus master unit, and add the upper four or three bits as bits [35:32] or [34:32] to the original address.

CAUTION

- For the bus master up to a 35-bit address space, the most significant bit([5]) of the SYS_AOF register must always be set to 0.
- When a transaction with [35] bit set to 1 is issued from the bus master up to a 35-bit address space, the operation is not guaranteed.

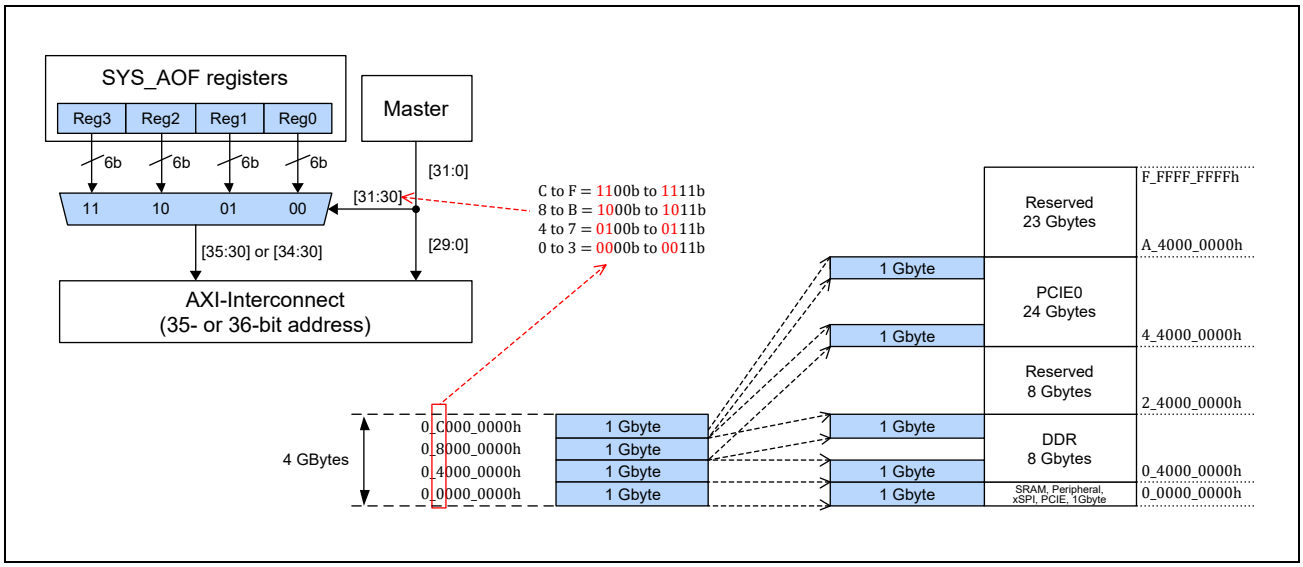


Figure 1.7-2 Address Space Extension (Overview)

CAUTION

When accessing DRAM from these bus master units, accesses that exceed a 4-Gbyte boundary are prohibited.

1.7.3.2.2 36-Bit Address Space Access for CM33

This LSI also supports a 36-bit address space for CM33. The hardware reallocates the CM33 address space to the address map of the LSI device as a whole. After that, it handles address space extension in units of 256 Mbytes. **Figure 1.7-3** shows reallocation from the CM33 address space to the address map of the LSI device as a whole. **Figure 1.7-4** shows the address space extension from the address map of the LSI device as a whole to spaces extending beyond 4 Gbytes.

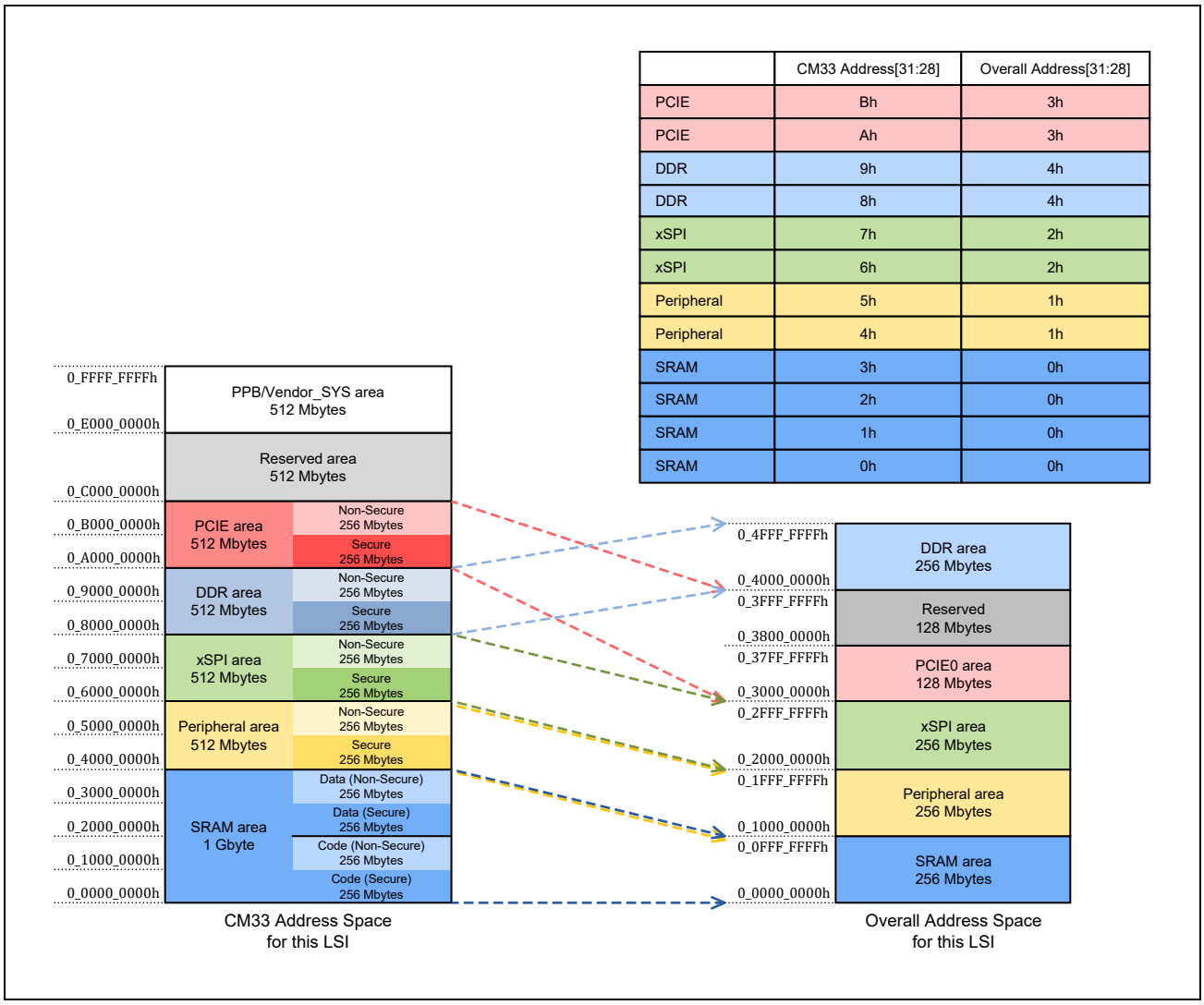


Figure 1.7-3 Reallocation from CM33 Address Space to Overall Address Space (Overview)

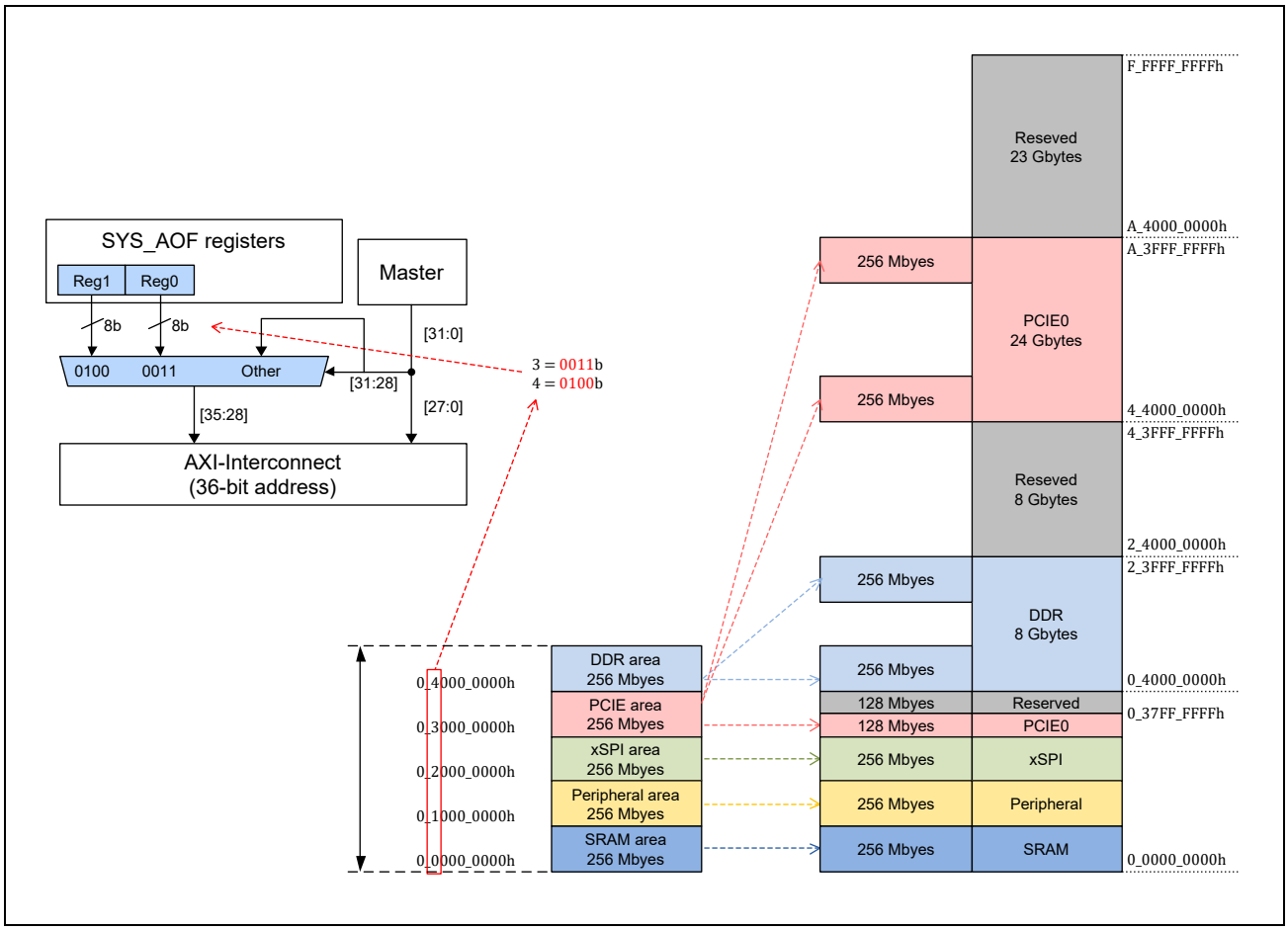


Figure 1.7-4 Address Space Extension for CM33 (Overview)

1.7.3.3 Bus Error Interrupt Generation

This LSI generates a bus error interrupt when any of the following conditions is met.

- A bus error is generated when access is disabled by the security setting.
- A bus error is generated when access to a slave unit in the module stop state (MSTOP) is attempted.
- A bus error is generated by a slave unit.

This function is supported for all bus master units included in this LSI.

Bus error interrupts are controlled (retained, cleared, and gathered into a single signal) by the interrupt control circuit (ICU). For details, including the types of the bus errors, see **4.6 Interrupt Controller**.

NOTE

This function cannot distinguish between a bus error generated when access is disabled by the security setting and a bus error generated due to other causes.

If bus errors occur consecutively, they will be notified to the CPU once, unless cleared.

In addition to the above bus error, an ERRINT interrupt is generated in any of the following cases.

- When the response to a write transaction with the bufferable attribute from the CM33 C-AHB interface is ERROR.
- When the response to a write transaction with the bufferable attribute from the CM33 S-AHB interface is ERROR.

SECTION 1 OVERVIEW

1.8 Address Map

This section describes the address space of this LSI.

1.8.1 Overall Address Space

Figure 1.8-1 shows the overall address space of this LSI and **Table 1.8-1** shows the detailed address space.

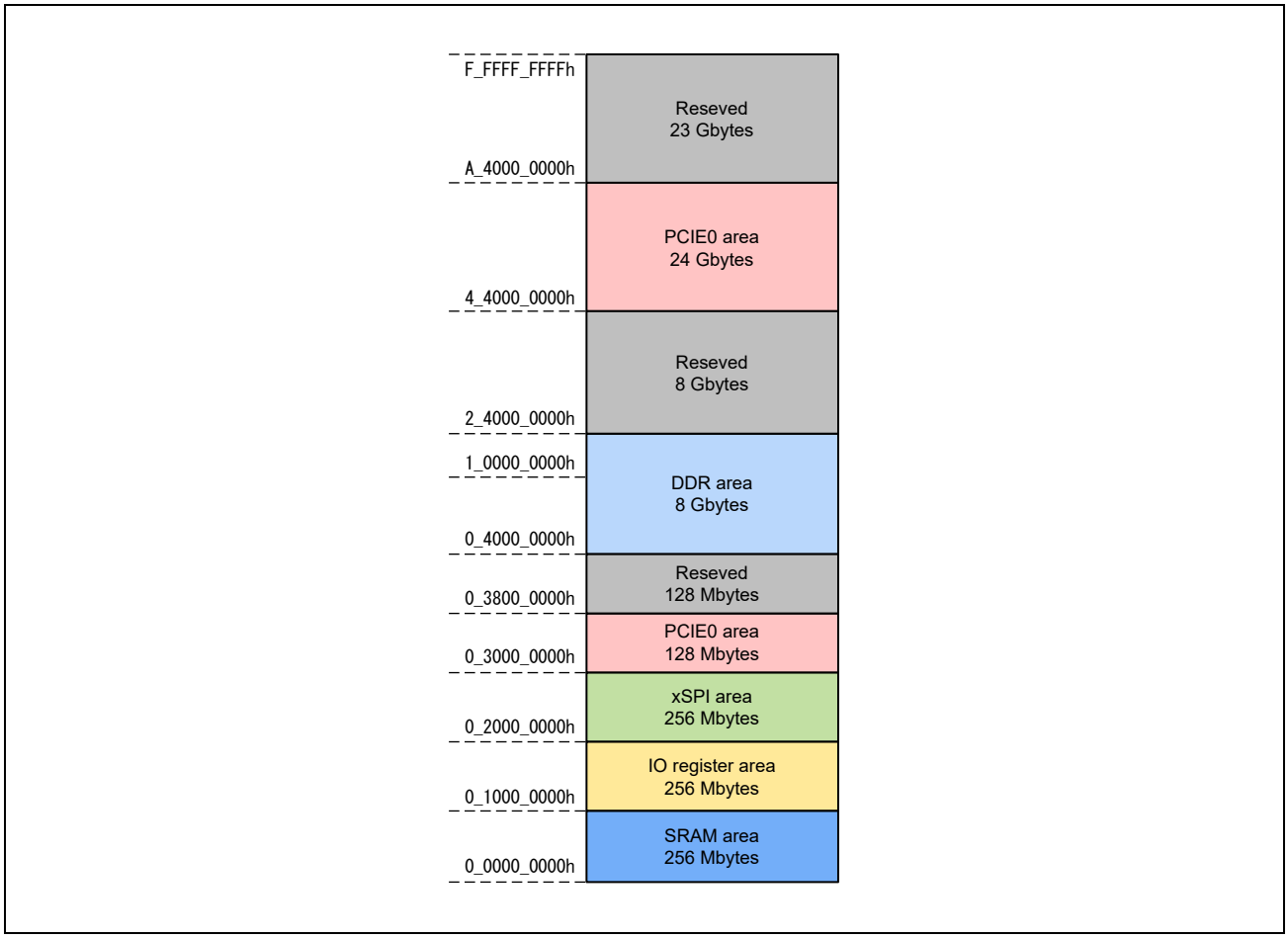


Figure 1.8-1 Overall Address Space

Some bus master units of this LSI require 35-bit or 36-bit address space access to access an address space of greater than 4 Gbytes in the whole 64-Gbyte address space of this LSI. For details, see **1.7 Internal Bus**.

The areas listed in **Table 1.8-1** are accessible by the CA55. Address space extension is needed for access to areas requiring addresses with widths of 33 or more bits.

The CM33 address space for this LSI differs from the overall address space. It is the same as the address space of the default memory map of CM33. For details, see **1.8.2 CM33 Address Space**.

Table 1.8-1 Detailed Address Space (1/6)

Start Address	End Address	Size		Space	Remarks
A_4000_0000h	F_FFFF_FFFFh	23	Gbytes	Reserved	*1
4_4000_0000h	A_3FFF_FFFFh	24	Gbytes	PCIE0	
2_4000_0000h	4_3FFF_FFFFh	8	Gbytes	Reserved	*1
0_4000_0000h	2_3FFF_FFFFh	8	Gbytes	DDR	
0_3800_0000h	0_3FFF_FFFFh	128	Mbytes	Reserved	*1
0_3000_0000h	0_37FF_FFFFh	128	Mbytes	PCIE0	
0_2000_0000h	0_2FFF_FFFFh	256	Mbytes	xSPI	
0_1F00_0000h	0_1FFF_FFFFh	16	Mbytes	CST	
0_1E40_0000h	0_1EFF_FFFFh	12	Mbytes	Reserved	*1
0_1E20_0000h	0_1E3F_FFFFh	2	Mbytes	Reserved	*1
0_1E10_0000h	0_1E1F_FFFFh	1	Mbyte	Reserved	*1
0_1E02_0000h	0_1E0F_FFFFh	896	Kbytes	Reserved	*1
0_1E01_0000h	0_1E01_FFFFh	64	Kbytes	Reserved	*1
0_1E00_0000h	0_1E00_FFFFh	64	Kbytes	DDR (MEMC)	
0_1C00_0000h	0_1DFF_FFFFh	32	Mbytes	Reserved	*1
0_1A00_0000h	0_1BFF_FFFFh	32	Mbytes	DDR (PHY)	
0_19E0_0000h	0_19FF_FFFFh	2	Mbytes	Reserved	*1
0_19D0_0000h	0_19DF_FFFFh	1	Mbyte	Reserved	*1
0_19CA_0000h	0_19CF_FFFFh	384	Kbytes	Reserved	*1
0_19C9_0000h	0_19C9_FFFFh	64	Kbytes	Reserved	*1
0_19C8_0000h	0_19C8_FFFFh	64	Kbytes	NPU	
0_19C7_0000h	0_19C7_FFFFh	64	Kbytes	Reserved	*1
0_19C6_0000h	0_19C6_FFFFh	64	Kbytes	Reserved	*1
0_19C5_0000h	0_19C5_FFFFh	64	Kbytes	Reserved	*1
0_19C4_0000h	0_19C4_FFFFh	64	Kbytes	Reserved	*1
0_19C3_0000h	0_19C3_FFFFh	64	Kbytes	Reserved	*1
0_19C2_0000h	0_19C2_FFFFh	64	Kbytes	Reserved	*1
0_19C1_0000h	0_19C1_FFFFh	64	Kbytes	Reserved	*1
0_19C0_0000h	0_19C0_FFFFh	64	Kbytes	SRAM1 (REG)	
0_1920_0000h	0_19BF_FFFFh	10	Mbytes	Reserved	*1
0_1910_0000h	0_191F_FFFFh	1	Mbyte	Reserved	*1
0_1900_0000h	0_190F_FFFFh	1	Mbyte	Reserved	*1
0_1800_0000h	0_18FF_FFFFh	16	Mbytes	Reserved	*1
0_1700_0000h	0_17FF_FFFFh	16	Mbytes	Reserved	*1
0_16C0_0000h	0_16FF_FFFFh	4	Mbytes	Reserved	*1
0_1680_0000h	0_16BF_FFFFh	4	Mbytes	Reserved	*1
0_1670_0000h	0_167F_FFFFh	1	Mbyte	Reserved	*1
0_1660_0000h	0_166F_FFFFh	1	Mbyte	Reserved	*1
0_1650_0000h	0_165F_FFFFh	1	Mbyte	Reserved	*1
0_164D_0000h	0_164F_FFFFh	192	Kbytes	Reserved	*1
0_164C_0000h	0_164C_FFFFh	64	Kbytes	LVDS	
0_164B_0000h	0_164B_FFFFh	64	Kbytes	LCDC1 (VSPD)	
0_164A_0000h	0_164A_FFFFh	64	Kbytes	LCDC1 (FCPVD)	
0_1649_0000h	0_1649_FFFFh	64	Kbytes	LCDC1 (DU)	
0_1648_0000h	0_1648_FFFFh	64	Kbytes	LCDC0 (VSPD)	

Table 1.8-1 Detailed Address Space (2/6)

Start Address	End Address	Size		Space	Remarks
0_1647_0000h	0_1647_FFFFh	64	Kbytes	LCDC0 (FCPVD)	
0_1646_0000h	0_1646_FFFFh	64	Kbytes	LCDC0 (DU)	
0_1645_0000h	0_1645_FFFFh	64	Kbytes	Reserved	*1
0_1644_0000h	0_1644_FFFFh	64	Kbytes	DSI (DPHY)	
0_1643_0000h	0_1643_FFFFh	64	Kbytes	DSI (LINK)	
0_1642_0000h	0_1642_FFFFh	64	Kbytes	VCD (CE)	
0_1641_0000h	0_1641_FFFFh	64	Kbytes	VCD (FCPC)	
0_1640_0000h	0_1640_FFFFh	64	Kbytes	VCD (VLC)	
0_1632_0000h	0_163F_FFFFh	896	Kbytes	Reserved	*1
0_1631_0000h	0_1631_FFFFh	64	Kbytes	FDP (FCPF)	
0_1630_0000h	0_1630_FFFFh	64	Kbytes	FDP	
0_1620_0000h	0_162F_FFFFh	1	Mbyte	Reserved	*1
0_1610_0000h	0_161F_FFFFh	1	Mbyte	Reserved	*1
0_1608_0000h	0_160F_FFFFh	512	Kbytes	Reserved	*1
0_1607_0000h	0_1607_FFFFh	64	Kbytes	Reserved	*1
0_1606_0000h	0_1606_FFFFh	64	Kbytes	VSP (FCPVI)	
0_1605_0000h	0_1605_FFFFh	64	Kbytes	VSP	
0_1604_0000h	0_1604_FFFFh	64	Kbytes	Reserved	*1
0_1603_0000h	0_1603_FFFFh	64	Kbytes	Reserved	*1
0_1602_0000h	0_1602_FFFFh	64	Kbytes	Reserved	*1
0_1601_0000h	0_1601_FFFFh	64	Kbytes	Reserved	*1
0_1600_0000h	0_1600_FFFFh	64	Kbytes	CRU0	
0_15F0_0000h	0_15FF_FFFFh	1	Mbyte	Reserved	*1
0_15E0_0000h	0_15EF_FFFFh	1	Mbyte	Reserved	*1
0_15D0_0000h	0_15DF_FFFFh	1	Mbyte	Reserved	*1
0_15C5_0000h	0_15CF_FFFFh	704	Kbytes	Reserved	*1
0_15C4_0000h	0_15C4_FFFFh	64	Kbytes	GBETH1	
0_15C3_0000h	0_15C3_FFFFh	64	Kbytes	GBETH0	
0_15C2_0000h	0_15C2_FFFFh	64	Kbytes	SDI2	
0_15C1_0000h	0_15C1_FFFFh	64	Kbytes	SD1	
0_15C0_0000h	0_15C0_FFFFh	64	Kbytes	SD0	
0_15B0_0000h	0_15BF_FFFFh	1	Mbyte	Reserved	*1
0_15A0_0000h	0_15AF_FFFFh	1	Mbyte	Reserved	*1
0_1590_0000h	0_159F_FFFFh	1	Mbyte	Reserved	*1
0_158A_0000h	0_158F_FFFFh	384	Kbytes	Reserved	*1
0_1589_0000h	0_1589_FFFFh	64	Kbytes	Reserved	*1
0_1588_0000h	0_1588_FFFFh	64	Kbytes	Reserved	*1
0_1587_0000h	0_1587_FFFFh	64	Kbytes	USB30 (PHY)	
0_1586_0000h	0_1586_FFFFh	64	Kbytes	Reserved	*1
0_1585_0000h	0_1585_FFFFh	64	Kbytes	USB30 (Host)	
0_1584_0000h	0_1584_FFFFh	64	Kbytes	USB21 (PHY)	
0_1583_0000h	0_1583_FFFFh	64	Kbytes	USB20 (PHY)	
0_1582_0000h	0_1582_FFFFh	64	Kbytes	USB20 (Function)	
0_1581_0000h	0_1581_FFFFh	64	Kbytes	USB21 (Host)	
0_1580_0000h	0_1580_FFFFh	64	Kbytes	USB20 (Host)	

Table 1.8-1 Detailed Address Space (3/6)

Start Address	End Address	Size		Space	Remarks
0_1540_0000h	0_157F_FFFFh	4	Mbytes	Reserved	*1
0_14E0_0000h	0_153F_FFFFh	6	Mbytes	Reserved	*1
0_14D0_0000h	0_14DF_FFFFh	1	Mbyte	Reserved	*1
0_14C0_0000h	0_14CF_FFFFh	1	Mbyte	Reserved	*1
0_14B0_0000h	0_14BF_FFFFh	1	Mbyte	Reserved	*1
0_14A0_0000h	0_14AF_FFFFh	1	Mbyte	Reserved	*1
0_1490_0000h	0_149F_FFFFh	1	Mbyte	GIC	
0_1486_0000h	0_148F_FFFFh	640	Kbytes	Reserved	*1
0_1485_0000h	0_1485_FFFFh	64	Kbytes	GE3D	
0_1484_0000h	0_1484_FFFFh	64	Kbytes	DMAC2	
0_1483_0000h	0_1483_FFFFh	64	Kbytes	DMAC1	
0_1482_0000h	0_1482_FFFFh	64	Kbytes	Reserved	*1
0_1481_0000h	0_1481_FFFFh	64	Kbytes	Reserved	*1
0_1480_0000h	0_1480_FFFFh	64	Kbytes	Reserved	*1
0_1440_3000h	0_147F_FFFFh	4084	Kbytes	Reserved	*1
0_1440_2C00h	0_1440_2FFFh	1	Kbyte	SPDIF2	
0_1440_2800h	0_1440_2BFFh	1	Kbyte	SPDIF1	
0_1440_2400h	0_1440_27FFh	1	Kbyte	SPDIF0	
0_1440_2000h	0_1440_23FFh	1	Kbyte	RIIC7	
0_1440_1C00h	0_1440_1FFFh	1	Kbyte	RIIC6	
0_1440_1800h	0_1440_1BFFh	1	Kbyte	RIIC5	
0_1440_1400h	0_1440_17FFh	1	Kbyte	RIIC4	
0_1440_1000h	0_1440_13FFh	1	Kbyte	RIIC3	
0_1440_0C00h	0_1440_0FFFh	1	Kbyte	RIIC2	
0_1440_0800h	0_1440_0BFFh	1	Kbyte	RIIC1	
0_1440_0400h	0_1440_07FFh	1	Kbyte	RIIC0	
0_1440_0000h	0_1440_03FFh	1	Kbyte	WDT1	
0_1420_0000h	0_143F_FFFFh	2	Mbytes	Reserved	*1
0_1410_0000h	0_141F_FFFFh	1	Mbyte	Reserved	*1
0_1402_0000h	0_140F_FFFFh	896	Kbytes	Reserved	*1
0_1401_0000h	0_1401_FFFFh	64	Kbytes	SYC	
0_1400_3000h	0_1400_FFFFh	52	Kbytes	Reserved	*1
0_1400_2000h	0_1400_2FFFh	4	Kbytes	TSU	
0_1400_1000h	0_1400_1FFFh	4	Kbytes	GTM3	
0_1400_0000h	0_1400_0FFFh	4	Kbytes	GTM2	
0_13E0_0000h	0_13FF_FFFFh	2	Mbytes	Reserved	*1
0_13D0_0000h	0_13DF_FFFFh	1	Mbyte	Reserved	*1
0_13CD_0000h	0_13CF_FFFFh	192	Kbytes	Reserved	*1
0_13CC_0000h	0_13CC_FFFFh	64	Kbytes	TZC400 (PCIE0)	
0_13CB_0000h	0_13CB_FFFFh	64	Kbytes	Reserved	*1
0_13CA_0000h	0_13CA_FFFFh	64	Kbytes	TZC400 (RCPU Bus)	
0_13C9_0000h	0_13C9_FFFFh	64	Kbytes	Reserved	*1
0_13C8_0000h	0_13C8_FFFFh	64	Kbytes	Reserved	*1
0_13C7_0000h	0_13C7_FFFFh	64	Kbytes	TZC400 (DDR_1)	
0_13C6_0000h	0_13C6_FFFFh	64	Kbytes	TZC400 (DDR_0)	

Table 1.8-1 Detailed Address Space (4/6)

Start Address	End Address	Size		Space	Remarks
0_13C5_0000h	0_13C5_FFFFh	64	Kbytes	ADMAC	
0_13C4_0000h	0_13C4_FFFFh	64	Kbytes	SSIU (DMAC)	
0_13C3_0000h	0_13C3_FFFFh	64	Kbytes	SSIU	
0_13C2_0000h	0_13C2_FFFFh	64	Kbytes	ADG	
0_13C1_0000h	0_13C1_FFFFh	64	Kbytes	SCU (DMAC)	
0_13C0_0000h	0_13C0_FFFFh	64	Kbytes	SCU	
0_1380_0000h	0_13BF_FFFFh	4	Mbytes	Reserved	*1
0_1360_0000h	0_137F_FFFFh	2	Mbytes	Reserved	*1
0_1350_0000h	0_135F_FFFFh	1	Mbyte	Reserved	*1
0_1342_0000h	0_134F_FFFFh	896	Kbytes	Reserved	*1
0_1341_0000h	0_1341_FFFFh	64	Kbytes	Reserved	*1
0_1340_0000h	0_1340_FFFFh	64	Kbytes	PCIE0	
0_1303_0000h	0_133F_FFFFh	3904	Kbytes	Reserved	*1
0_1302_0000h	0_1302_FFFFh	64	Kbytes	GPT1	
0_1301_0000h	0_1301_FFFFh	64	Kbytes	GPT0	
0_1300_3C00h	0_1300_FFFFh	49	Kbytes	Reserved	*1
0_1300_3800h	0_1300_3BFFh	1	Kbyte	POEG1D	
0_1300_3400h	0_1300_37FFh	1	Kbyte	POEG1C	
0_1300_3000h	0_1300_33FFh	1	Kbyte	POEG1B	
0_1300_2C00h	0_1300_2FFFh	1	Kbyte	POEG1A	
0_1300_2800h	0_1300_2BFFh	1	Kbyte	POEG0D	
0_1300_2400h	0_1300_27FFh	1	Kbyte	POEG0C	
0_1300_2000h	0_1300_23FFh	1	Kbyte	POEG0B	
0_1300_1C00h	0_1300_1FFFh	1	Kbyte	POEG0A	
0_1300_1800h	0_1300_1BFFh	1	Kbyte	CMTW7	
0_1300_1400h	0_1300_17FFh	1	Kbyte	CMTW6	
0_1300_1000h	0_1300_13FFh	1	Kbyte	CMTW5	
0_1300_0C00h	0_1300_0FFFh	1	Kbyte	CMTW4	
0_1300_0800h	0_1300_0BFFh	1	Kbyte	CRC	
0_1300_0400h	0_1300_07FFh	1	Kbyte	WDT3	
0_1300_0000h	0_1300_03FFh	1	Kbyte	WDT2	
0_12E0_0000h	0_12FF_FFFFh	2	Mbytes	Reserved	*1
0_12D0_0000h	0_12DF_FFFFh	1	Mbyte	Reserved	*1
0_12C2_0000h	0_12CF_FFFFh	896	Kbytes	Reserved	*1
0_12C1_0000h	0_12C1_FFFFh	64	Kbytes	Reserved	*1
0_12C0_4000h	0_12C0_FFFFh	48	Kbytes	Reserved	*1
0_12C0_3000h	0_12C0_3FFFh	4	Kbytes	GTM7	
0_12C0_2000h	0_12C0_2FFFh	4	Kbytes	GTM6	
0_12C0_1000h	0_12C0_1FFFh	4	Kbytes	GTM5	
0_12C0_0000h	0_12C0_0FFFh	4	Kbytes	GTM4	
0_1280_3400h	0_12BF_FFFFh	4083	Kbytes	Reserved	*1
0_1280_3000h	0_1280_33FFh	1	Kbyte	RSCI9	
0_1280_2C00h	0_1280_2FFFh	1	Kbyte	RSCI8	
0_1280_2800h	0_1280_2BFFh	1	Kbyte	RSCI7	
0_1280_2400h	0_1280_27FFh	1	Kbyte	RSCI6	

Table 1.8-1 Detailed Address Space (5/6)

Start Address	End Address	Size		Space	Remarks
0_1280_2000h	0_1280_23FFh	1	Kbyte	RSCI5	
0_1280_1C00h	0_1280_1FFFh	1	Kbyte	RSCI4	
0_1280_1800h	0_1280_1BFFh	1	Kbyte	RSCI3	
0_1280_1400h	0_1280_17FFh	1	Kbyte	RSCI2	
0_1280_1000h	0_1280_13FFh	1	Kbyte	RSCI1	
0_1280_0C00h	0_1280_0FFFh	1	Kbyte	RSCI0	
0_1280_0800h	0_1280_0BFFh	1	Kbyte	RSPI2	
0_1280_0400h	0_1280_07FFh	1	Kbyte	RSPI1	
0_1280_0000h	0_1280_03FFh	1	Kbyte	RSPI0	
0_1260_0000h	0_127F_FFFFh	2	Mbytes	Reserved	*1
0_1250_0000h	0_125F_FFFFh	1	Mbyte	Reserved	*1
0_1248_0000h	0_124F_FFFFh	512	Kbytes	Reserved	*1
0_1244_0000h	0_1247_FFFFh	256	Kbytes	CANFD	
0_1243_0000h	0_1243_FFFFh	64	Kbytes	Reserved	*1
0_1242_0000h	0_1242_FFFFh	64	Kbytes	Reserved	*1
0_1241_0000h	0_1241_FFFFh	64	Kbytes	Reserved	*1
0_1240_0000h	0_1240_FFFFh	64	Kbytes	I3C	
0_1220_0000h	0_123F_FFFFh	2	Mbytes	Reserved	*1
0_1210_0000h	0_121F_FFFFh	1	Mbyte	Reserved	*1
0_120C_0000h	0_120F_FFFFh	256	Kbytes	Reserved	*1
0_120A_0000h	0_120B_FFFFh	128	Kbytes	Reserved	*1
0_1208_0000h	0_1209_FFFFh	128	Kbytes	Reserved	*1
0_1206_0000h	0_1207_FFFFh	128	Kbytes	Reserved	*1
0_1204_0000h	0_1205_FFFFh	128	Kbytes	Reserved	*1
0_1202_0000h	0_1203_FFFFh	128	Kbytes	Reserved	*1
0_1201_0000h	0_1201_FFFFh	64	Kbytes	DMAC4	
0_1200_0000h	0_1200_FFFFh	64	Kbytes	DMAC3	
0_11C0_2800h	0_11FF_FFFFh	4086	Kbytes	Reserved	*1
0_11C0_2400h	0_11C0_27FFh	1	Kbyte	CMTW3	
0_11C0_2000h	0_11C0_23FFh	1	Kbyte	CMTW2	
0_11C0_1C00h	0_11C0_1FFFh	1	Kbyte	CMTW1	
0_11C0_1800h	0_11C0_1BFFh	1	Kbyte	CMTW0	
0_11C0_1400h	0_11C0_17FFh	1	Kbyte	SCIF	
0_11C0_1000h	0_11C0_13FFh	1	Kbyte	RIIC8	
0_11C0_0C00h	0_11C0_0FFFh	1	Kbyte	RTC (Read Only)	
0_11C0_0800h	0_11C0_0BFFh	1	Kbyte	RTC	
0_11C0_0400h	0_11C0_07FFh	1	Kbyte	WDT0	
0_11C0_0000h	0_11C0_03FFh	1	Kbyte	ADC	
0_11A0_0000h	0_11BF_FFFFh	2	Mbytes	Reserved	*1
0_1190_0000h	0_119F_FFFFh	1	Mbyte	Reserved	*1
0_1180_2000h	0_118F_FFFFh	1016	Kbytes	Reserved	*1
0_1180_1000h	0_1180_1FFFh	4	Kbytes	GTM1	
0_1180_0000h	0_1180_0FFFh	4	Kbytes	GTM0	
0_1160_0000h	0_117F_FFFFh	2	Mbytes	Reserved	*1
0_1150_0000h	0_115F_FFFFh	1	Mbyte	Reserved	*1

Table 1.8-1 Detailed Address Space (6/6)

Start Address	End Address	Size		Space	Remarks
0_1141_0000h	0_114F_FFFFh	960	Kbytes	Reserved	*1
0_1140_0000h	0_1140_FFFFh	64	Kbytes	DMAC0	
0_1130_0000h	0_113F_FFFFh	1	Mbyte	Reserved	*1
0_1120_0000h	0_112F_FFFFh	1	Mbyte	Reserved	*1
0_1110_0000h	0_111F_FFFFh	1	Mbyte	Reserved	*1
0_1106_0000h	0_110F_FFFFh	640	Kbytes	Reserved	*1
0_1105_0000h	0_1105_FFFFh	64	Kbytes	PDM1	
0_1104_0000h	0_1104_FFFFh	64	Kbytes	PDM0	
0_1103_0000h	0_1103_FFFFh	64	Kbytes	xSPI (REG)	
0_1102_0000h	0_1102_FFFFh	64	Kbytes	Reserved	*1
0_1101_0000h	0_1101_FFFFh	64	Kbytes	SRAM0 (REG)	
0_1100_1000h	0_1100_FFFFh	60	Kbytes	Reserved	*1
0_1100_0000h	0_1100_0FFFh	4	Kbytes	Reserved	*1
0_10D0_0000h	0_10FF_FFFFh	3	Mbytes	Reserved	*1
0_10C0_0000h	0_10CF_FFFFh	1	Mbyte	Reserved	*1
0_1080_0000h	0_10BF_FFFFh	4	Mbytes	Reserved	*1
0_104D_0000h	0_107F_FFFFh	3264	Kbytes	Reserved	*1
0_104C_0000h	0_104C_FFFFh	64	Kbytes	Reserved	*1
0_104B_0000h	0_104B_FFFFh	64	Kbytes	Reserved	*1
0_104A_0000h	0_104A_FFFFh	64	Kbytes	Reserved	*1
0_1049_0000h	0_1049_FFFFh	64	Kbytes	Reserved	*1
0_1048_0000h	0_1048_FFFFh	64	Kbytes	MHU	
0_1047_0000h	0_1047_FFFFh	64	Kbytes	TZC400 (xSPI)	
0_1046_0000h	0_1046_FFFFh	64	Kbytes	TZC400 (SRAM0)	
0_1045_0000h	0_1045_FFFFh	64	Kbytes	OTP	
0_1044_0000h	0_1044_FFFFh	64	Kbytes	Reserved	*1
0_1043_0000h	0_1043_FFFFh	64	Kbytes	SYS	
0_1042_0000h	0_1042_FFFFh	64	Kbytes	CPG	
0_1041_0000h	0_1041_FFFFh	64	Kbytes	PFC	
0_1040_0000h	0_1040_FFFFh	64	Kbytes	ICU	
0_1000_0000h	0_103F_FFFFh	4	Mbytes	Reserved	*1
0_0860_0000h	0_0FFF_FFFFh	122	Mbytes	Reserved	*1
0_0820_0000h	0_085F_FFFFh	4	Mbytes	Reserved	*1
0_0818_0000h	0_081F_FFFFh	512	Kbytes	Reserved	*1
0_0810_0000h	0_0817_FFFFh	512	Kbytes	Reserved	*1
0_0808_0000h	0_080F_FFFFh	512	Kbytes	Reserved	*1
0_0804_0000h	0_0807_FFFFh	256	Kbytes	SRAM1	
0_0800_0000h	0_0803_FFFFh	256	Kbytes	SRAM0	
0_07F8_2000h	0_07FF_FFFFh	504	Kbytes	Reserved	*1
0_07F8_0000h	0_07F8_1FFFh	8	Kbytes	Reserved	*1
0_0002_0000h	0_07F7_FFFFh	130432	Kbytes	Reserved	*1
0_0000_0000h	0_0001_FFFFh	256	Kbytes	Reserved	*1

Note 1. Access to the reserved areas is prohibited. If access to the reserved area is attempted, incorrect operation may occur.

1.8.2 CM33 Address Space

Figure 1.8-2 shows the default memory map of CM33 and an overview of the CM33 address space for this LSI, and **Table 1.8-2** shows the detailed address space. For the concept of the secure and non-secure states, see **2.2 CPU**.

In this LSI, addresses 0000_0000h to 1FFF_FFFFh are used as the code (program) area and addresses 2000_0000h to 3FFF_FFFFh are used as the data area in the 1-Gbyte SRAM area (0000_0000h to 3FFF_FFFFh).

In this LSI, the hardware automatically translates the addresses in the CM33 address space to those in the overall address space of this LSI (see **1.7.3.2.2 36-Bit Address Space Access for CM33**).

Users can therefore write programs using the addresses in the address space shown in **Figure 1.8-2** and **Table 1.8-2**, without considering the overall address space.

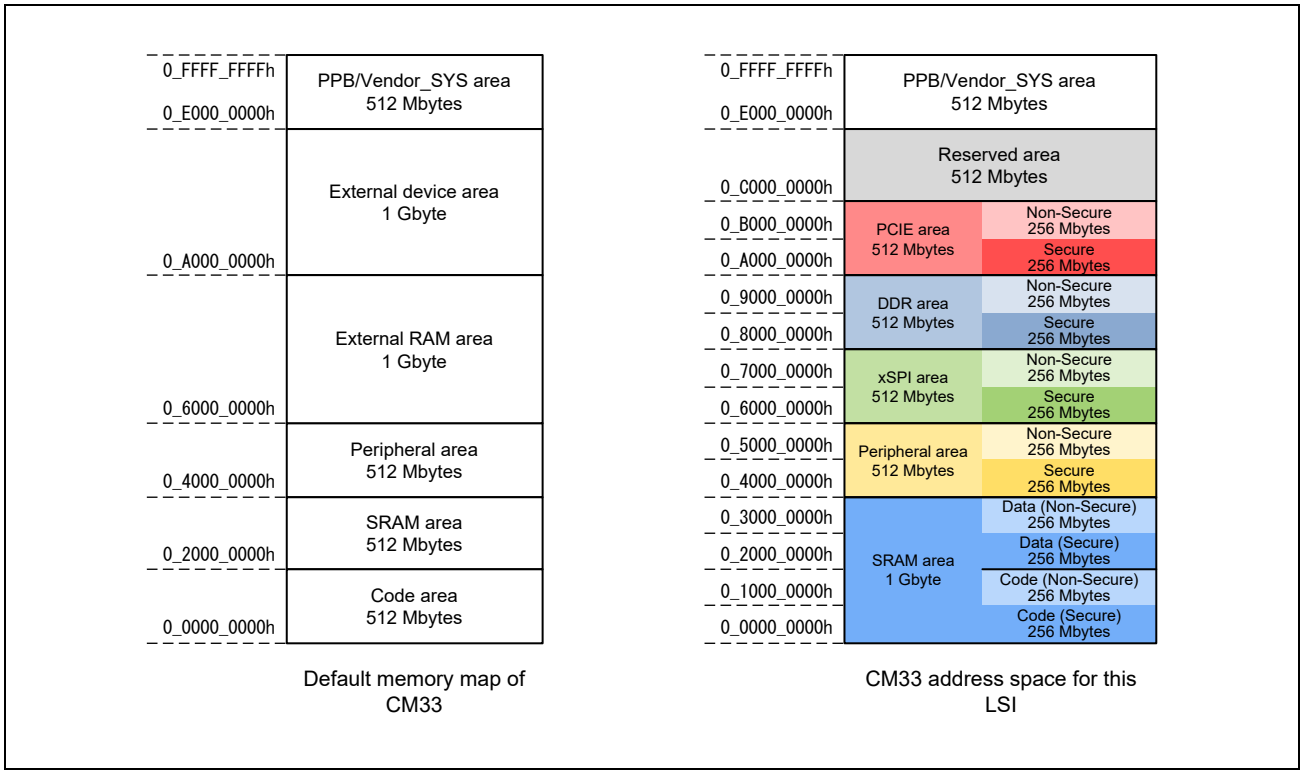


Figure 1.8-2 Overview of CM33 Address Space

Table 1.8-2 Detailed Address Space of CM33 (1/13)

Start Address	End Address	Size		Space	Remarks
E000_0000h	FFFF_FFFFh	512	Mbytes	PPB/Vendor_SYS	*2
C000_0000h	DFFF_FFFFh	512	Mbytes	Reserved	*1
B000_0000h	BFFF_FFFFh	256	Mbytes	PCIE (Non-Secure)	
A000_0000h	AFFF_FFFFh	256	Mbytes	PCIE (Secure)	
9000_0000h	9FFF_FFFFh	256	Mbytes	DDR (Non-Secure)	
8000_0000h	8FFF_FFFFh	256	Mbytes	DDR (Secure)	
7000_0000h	7FFF_FFFFh	256	Mbytes	xSPI (Non-Secure)	
6000_0000h	6FFF_FFFFh	256	Mbytes	xSPI (Secure)	
5F00_0000h	5FFF_FFFFh	16	Mbytes	CST (Non-Secure)	
5E40_0000h	5EFF_FFFFh	12	Mbytes	Reserved	*1
5E20_0000h	5E3F_FFFFh	2	Mbytes	Reserved	*1
5E10_0000h	5E1F_FFFFh	1	Mbyte	Reserved	*1
5E02_0000h	5E0F_FFFFh	896	Kbytes	Reserved	*1
5E01_0000h	5E01_FFFFh	64	Kbytes	Reserved	*1
5E00_0000h	5E00_FFFFh	64	Kbytes	DDR0 (MEMC) (Non-Secure)	
5C00_0000h	5DFF_FFFFh	32	Mbytes	Reserved	*1
5A00_0000h	5BFF_FFFFh	32	Mbytes	DDR0 (PHY) (Non-Secure)	
59E0_0000h	59FF_FFFFh	2	Mbytes	Reserved	*1
59D0_0000h	59DF_FFFFh	1	Mbyte	Reserved	*1
59CA_0000h	59CF_FFFFh	384	Kbytes	Reserved	*1
59C9_0000h	59C9_FFFFh	64	Kbytes	Reserved	*1
59C8_0000h	59C8_FFFFh	64	Kbytes	NPU (Non-Secure)	
59C7_0000h	59C7_FFFFh	64	Kbytes	Reserved	*1
59C6_0000h	59C6_FFFFh	64	Kbytes	Reserved	*1
59C5_0000h	59C5_FFFFh	64	Kbytes	Reserved	*1
59C4_0000h	59C4_FFFFh	64	Kbytes	Reserved	*1
59C3_0000h	59C3_FFFFh	64	Kbytes	Reserved	*1
59C2_0000h	59C2_FFFFh	64	Kbytes	Reserved	*1
59C1_0000h	59C1_FFFFh	64	Kbytes	Reserved	*1
59C0_0000h	59C0_FFFFh	64	Kbytes	SRAM1 (REG) (Non-Secure)	
5920_0000h	59BF_FFFFh	10	Mbytes	Reserved	*1
5910_0000h	591F_FFFFh	1	Mbyte	Reserved	*1
5900_0000h	590F_FFFFh	1	Mbyte	Reserved	*1
5800_0000h	58FF_FFFFh	16	Mbytes	Reserved	*1
5700_0000h	57FF_FFFFh	16	Mbytes	Reserved	*1
56C0_0000h	56FF_FFFFh	4	Mbytes	Reserved	*1
5680_0000h	56BF_FFFFh	4	Mbytes	Reserved	*1
5670_0000h	567F_FFFFh	1	Mbyte	Reserved	*1
5660_0000h	566F_FFFFh	1	Mbyte	Reserved	*1
5650_0000h	565F_FFFFh	1	Mbyte	Reserved	*1
564D_0000h	564F_FFFFh	192	Kbytes	Reserved	*1
564C_0000h	564C_FFFFh	64	Kbytes	LVDS (Non-Secure)	
564B_0000h	564B_FFFFh	64	Kbytes	LCDC1 (VSPD) (Non-Secure)	
564A_0000h	564A_FFFFh	64	Kbytes	LCDC1 (FCPVD) (Non-Secure)	
5649_0000h	5649_FFFFh	64	Kbytes	LCDC1 (DU) (Non-Secure)	

Table 1.8-2 Detailed Address Space of CM33 (2/13)

Start Address	End Address	Size		Space	Remarks
5648_0000h	5648_FFFFh	64	Kbytes	LCDC0 (VSPD) (Non-Secure)	
5647_0000h	5647_FFFFh	64	Kbytes	LCDC0 (FCPVD) (Non-Secure)	
5646_0000h	5646_FFFFh	64	Kbytes	LCDC0 (DU) (Non-Secure)	
5645_0000h	5645_FFFFh	64	Kbytes	Reserved	*1
5644_0000h	5644_FFFFh	64	Kbytes	DSI (DPHY) (Non-Secure)	
5643_0000h	5643_FFFFh	64	Kbytes	DSI (LINK) (Non-Secure)	
5642_0000h	5642_FFFFh	64	Kbytes	VCD (CE) (Non-Secure)	
5641_0000h	5641_FFFFh	64	Kbytes	VCD (FCPC) (Non-Secure)	
5640_0000h	5640_FFFFh	64	Kbytes	VCD (VLC) (Non-Secure)	
5632_0000h	563F_FFFFh	896	Kbytes	Reserved	*1
5631_0000h	5631_FFFFh	64	Kbytes	FDP (FCPF) (Non-Secure)	
5630_0000h	5630_FFFFh	64	Kbytes	FDP (Non-Secure)	
5620_0000h	562F_FFFFh	1	Mbyte	Reserved	*1
5610_0000h	561F_FFFFh	1	Mbyte	Reserved	*1
5608_0000h	560F_FFFFh	512	Kbytes	Reserved	*1
5607_0000h	5607_FFFFh	64	Kbytes	Reserved	*1
5606_0000h	5606_FFFFh	64	Kbytes	VSP (FCFP) (Non-Secure)	
5605_0000h	5605_FFFFh	64	Kbytes	VSP (Non-Secure)	
5604_0000h	5604_FFFFh	64	Kbytes	Reserved	*1
5603_0000h	5603_FFFFh	64	Kbytes	Reserved	*1
5602_0000h	5602_FFFFh	64	Kbytes	Reserved	*1
5601_0000h	5601_FFFFh	64	Kbytes	Reserved	*1
5600_0000h	5600_FFFFh	64	Kbytes	CRU0 (Non-Secure)	
55F0_0000h	55FF_FFFFh	1	Mbyte	Reserved	*1
55E0_0000h	55EF_FFFFh	1	Mbyte	Reserved	*1
55D0_0000h	55DF_FFFFh	1	Mbyte	Reserved	*1
55C5_0000h	55CF_FFFFh	704	Kbytes	Reserved	*1
55C4_0000h	55C4_FFFFh	64	Kbytes	GBETH1 (Non-Secure)	
55C3_0000h	55C3_FFFFh	64	Kbytes	GBETH0 (Non-Secure)	
55C2_0000h	55C2_FFFFh	64	Kbytes	SD2 (Non-Secure)	
55C1_0000h	55C1_FFFFh	64	Kbytes	SD1 (Non-Secure)	
55C0_0000h	55C0_FFFFh	64	Kbytes	SD0 (Non-Secure)	
55B0_0000h	55BF_FFFFh	1	Mbyte	Reserved	*1
55A0_0000h	55AF_FFFFh	1	Mbyte	Reserved	*1
5590_0000h	559F_FFFFh	1	Mbyte	Reserved	*1
558A_0000h	558F_FFFFh	384	Kbytes	Reserved	*1
5589_0000h	5589_FFFFh	64	Kbytes	Reserved	*1
5588_0000h	5588_FFFFh	64	Kbytes	Reserved	*1
5587_0000h	5587_FFFFh	64	Kbytes	USB30 (PHY) (Non-Secure)	
5586_0000h	5586_FFFFh	64	Kbytes	Reserved	*1
5585_0000h	5585_FFFFh	64	Kbytes	USB30 (Host) (Non-Secure)	
5584_0000h	5584_FFFFh	64	Kbytes	USB21 (PHY) (Non-Secure)	
5583_0000h	5583_FFFFh	64	Kbytes	USB20 (PHY) (Non-Secure)	
5582_0000h	5582_FFFFh	64	Kbytes	USB20 (Function) (Non-Secure)	
5581_0000h	5581_FFFFh	64	Kbytes	USB21 (Host) (Non-Secure)	

Table 1.8-2 Detailed Address Space of CM33 (3/13)

Start Address	End Address	Size		Space	Remarks
5580_0000h	5580_FFFFh	64	Kbytes	USB20 (Host) (Non-Secure)	
5540_0000h	557F_FFFFh	4	Mbytes	Reserved	*1
54E0_0000h	553F_FFFFh	6	Mbytes	Reserved	*1
54D0_0000h	54DF_FFFFh	1	Mbyte	Reserved	*1
54C0_0000h	54CF_FFFFh	1	Mbyte	Reserved	*1
54B0_0000h	54BF_FFFFh	1	Mbyte	Reserved	*1
54A0_0000h	54AF_FFFFh	1	Mbyte	Reserved	*1
5490_0000h	549F_FFFFh	1	Mbyte	GIC (Non-Secure)	
5486_0000h	548F_FFFFh	640	Kbytes	Reserved	*1
5485_0000h	5485_FFFFh	64	Kbytes	GE3D (Non-Secure)	
5484_0000h	5484_FFFFh	64	Kbytes	DMAC2 (Non-Secure)	
5483_0000h	5483_FFFFh	64	Kbytes	DMAC1 (Non-Secure)	
5482_0000h	5482_FFFFh	64	Kbytes	Reserved	*1
5481_0000h	5481_FFFFh	64	Kbytes	Reserved	*1
5480_0000h	5480_FFFFh	64	Kbytes	Reserved	*1
5440_3000h	547F_FFFFh	4084	Kbytes	Reserved	*1
5440_2C00h	5440_2FFFh	1	Kbyte	SPDIF2 (Non-Secure)	
5440_2800h	5440_2BFFh	1	Kbyte	SPDIF1 (Non-Secure)	
5440_2400h	5440_27FFh	1	Kbyte	SPDIF0 (Non-Secure)	
5440_2000h	5440_23FFh	1	Kbyte	RIIC7 (Non-Secure)	
5440_1C00h	5440_1FFFh	1	Kbyte	RIIC6 (Non-Secure)	
5440_1800h	5440_1BFFh	1	Kbyte	RIIC5 (Non-Secure)	
5440_1400h	5440_17FFh	1	Kbyte	RIIC4 (Non-Secure)	
5440_1000h	5440_13FFh	1	Kbyte	RIIC3 (Non-Secure)	
5440_0C00h	5440_0FFFh	1	Kbyte	RIIC2 (Non-Secure)	
5440_0800h	5440_0BFFh	1	Kbyte	RIIC1 (Non-Secure)	
5440_0400h	5440_07FFh	1	Kbyte	RIIC0 (Non-Secure)	
5440_0000h	5440_03FFh	1	Kbyte	WDT1 (Non-Secure)	
5420_0000h	543F_FFFFh	2	Mbytes	Reserved	*1
5410_0000h	541F_FFFFh	1	Mbyte	Reserved	*1
5402_0000h	540F_FFFFh	896	Kbytes	Reserved	*1
5401_0000h	5401_FFFFh	64	Kbytes	SYC (Non-Secure)	
5400_3000h	5400_FFFFh	52	Kbytes	Reserved	*1
5400_2000h	5400_2FFFh	4	Kbytes	TSU (Non-Secure)	
5400_1000h	5400_1FFFh	4	Kbytes	GTM3 (Non-Secure)	
5400_0000h	5400_0FFFh	4	Kbytes	GTM2 (Non-Secure)	
53E0_0000h	53FF_FFFFh	2	Mbytes	Reserved	*1
53D0_0000h	53DF_FFFFh	1	Mbyte	Reserved	*1
53CD_0000h	53CF_FFFFh	192	Kbytes	Reserved	*1
53CC_0000h	53CC_FFFFh	64	Kbytes	TZC400 (PCIE0) (Non-Secure)	
53CB_0000h	53CB_FFFFh	64	Kbytes	Reserved	*1
53CA_0000h	53CA_FFFFh	64	Kbytes	TZC400 (RCPU Bus) (Non-Secure)	
53C9_0000h	53C9_FFFFh	64	Kbytes	Reserved	*1
53C8_0000h	53C8_FFFFh	64	Kbytes	Reserved	*1
53C7_0000h	53C7_FFFFh	64	Kbytes	TZC400 (DDR_1) (Non-Secure)	

Table 1.8-2 Detailed Address Space of CM33 (4/13)

Start Address	End Address	Size		Space	Remarks
53C6_0000h	53C6_FFFFh	64	Kbytes	TZC400 (DDR_0) (Non-Secure)	
53C5_0000h	53C5_FFFFh	64	Kbytes	ADMAC (Non-Secure)	
53C4_0000h	53C4_FFFFh	64	Kbytes	SSIU (DMAC) (Non-Secure)	
53C3_0000h	53C3_FFFFh	64	Kbytes	SSIU (Non-Secure)	
53C2_0000h	53C2_FFFFh	64	Kbytes	ADG (Non-Secure)	
53C1_0000h	53C1_FFFFh	64	Kbytes	SCU (DMAC) (Non-Secure)	
53C0_0000h	53C0_FFFFh	64	Kbytes	SCU (Non-Secure)	
5380_0000h	53BF_FFFFh	4	Mbytes	Reserved	*1
5360_0000h	537F_FFFFh	2	Mbytes	Reserved	*1
5350_0000h	535F_FFFFh	1	Mbyte	Reserved	*1
5342_0000h	534F_FFFFh	896	Kbytes	Reserved	*1
5341_0000h	5341_FFFFh	64	Kbytes	Reserved	*1
5340_0000h	5340_FFFFh	64	Kbytes	PCIE0 (Non-Secure)	
5303_0000h	533F_FFFFh	3904	Kbytes	Reserved	*1
5302_0000h	5302_FFFFh	64	Kbytes	GPT1 (Non-Secure)	
5301_0000h	5301_FFFFh	64	Kbytes	GPT0 (Non-Secure)	
5300_3C00h	5300_FFFFh	49	Kbytes	Reserved	*1
5300_3800h	5300_3BFFh	1	Kbyte	POEG1D (Non-Secure)	
5300_3400h	5300_37FFh	1	Kbyte	POEG1C (Non-Secure)	
5300_3000h	5300_33FFh	1	Kbyte	POEG1B (Non-Secure)	
5300_2C00h	5300_2FFFh	1	Kbyte	POEG1A (Non-Secure)	
5300_2800h	5300_2BFFh	1	Kbyte	POEG0D (Non-Secure)	
5300_2400h	5300_27FFh	1	Kbyte	POEG0C (Non-Secure)	
5300_2000h	5300_23FFh	1	Kbyte	POEG0B (Non-Secure)	
5300_1C00h	5300_1FFFh	1	Kbyte	POEG0A (Non-Secure)	
5300_1800h	5300_1BFFh	1	Kbyte	CMTW7 (Non-Secure)	
5300_1400h	5300_17FFh	1	Kbyte	CMTW6 (Non-Secure)	
5300_1000h	5300_13FFh	1	Kbyte	CMTW5 (Non-Secure)	
5300_0C00h	5300_0FFFh	1	Kbyte	CMTW4 (Non-Secure)	
5300_0800h	5300_0BFFh	1	Kbyte	CRC (Non-Secure)	
5300_0400h	5300_07FFh	1	Kbyte	WDT3 (Non-Secure)	
5300_0000h	5300_03FFh	1	Kbyte	WDT2 (Non-Secure)	
52E0_0000h	52FF_FFFFh	2	Mbytes	Reserved	*1
52D0_0000h	52DF_FFFFh	1	Mbyte	Reserved	*1
52C2_0000h	52CF_FFFFh	896	Kbytes	Reserved	*1
52C1_0000h	52C1_FFFFh	64	Kbytes	Reserved	*1
52C0_4000h	52C0_FFFFh	48	Kbytes	Reserved	*1
52C0_3000h	52C0_3FFFh	4	Kbytes	GTM7 (Non-Secure)	
52C0_2000h	52C0_2FFFh	4	Kbytes	GTM6 (Non-Secure)	
52C0_1000h	52C0_1FFFh	4	Kbytes	GTM5 (Non-Secure)	
52C0_0000h	52C0_0FFFh	4	Kbytes	GTM4 (Non-Secure)	
5280_3400h	52BF_FFFFh	4083	Kbytes	Reserved	*1
5280_3000h	5280_33FFh	1	Kbyte	RSCI9 (Non-Secure)	
5280_2C00h	5280_2FFFh	1	Kbyte	RSCI8 (Non-Secure)	
5280_2800h	5280_2BFFh	1	Kbyte	RSCI7 (Non-Secure)	

Table 1.8-2 Detailed Address Space of CM33 (5/13)

Start Address	End Address	Size		Space	Remarks
5280_2400h	5280_27FFh	1	Kbyte	RSCI6 (Non-Secure)	
5280_2000h	5280_23FFh	1	Kbyte	RSCI5 (Non-Secure)	
5280_1C00h	5280_1FFFh	1	Kbyte	RSCI4 (Non-Secure)	
5280_1800h	5280_1BFFh	1	Kbyte	RSCI3 (Non-Secure)	
5280_1400h	5280_17FFh	1	Kbyte	RSCI2 (Non-Secure)	
5280_1000h	5280_13FFh	1	Kbyte	RSCI1 (Non-Secure)	
5280_0C00h	5280_0FFFh	1	Kbyte	RSCI0 (Non-Secure)	
5280_0800h	5280_0BFFh	1	Kbyte	RSPI2 (Non-Secure)	
5280_0400h	5280_07FFh	1	Kbyte	RSPI1 (Non-Secure)	
5280_0000h	5280_03FFh	1	Kbyte	RSPI0 (Non-Secure)	
5260_0000h	527F_FFFFh	2	Mbytes	Reserved	*1
5250_0000h	525F_FFFFh	1	Mbyte	Reserved	*1
5248_0000h	524F_FFFFh	512	Kbytes	Reserved	*1
5244_0000h	5247_FFFFh	256	Kbytes	CANFD (Non-Secure)	
5243_0000h	5243_FFFFh	64	Kbytes	Reserved	*1
5242_0000h	5242_FFFFh	64	Kbytes	Reserved	*1
5241_0000h	5241_FFFFh	64	Kbytes	Reserved	*1
5240_0000h	5240_FFFFh	64	Kbytes	I3C (Non-Secure)	
5220_0000h	523F_FFFFh	2	Mbytes	Reserved	*1
5210_0000h	521F_FFFFh	1	Mbyte	Reserved	*1
520C_0000h	520F_FFFFh	256	Kbytes	Reserved	*1
520A_0000h	520B_FFFFh	128	Kbytes	Reserved	*1
5208_0000h	5209_FFFFh	128	Kbytes	Reserved	*1
5206_0000h	5207_FFFFh	128	Kbytes	Reserved	*1
5204_0000h	5205_FFFFh	128	Kbytes	Reserved	*1
5202_0000h	5203_FFFFh	128	Kbytes	Reserved	*1
5201_0000h	5201_FFFFh	64	Kbytes	DMAC4 (Non-Secure)	
5200_0000h	5200_FFFFh	64	Kbytes	DMAC3 (Non-Secure)	
51C0_2800h	51FF_FFFFh	4086	Kbytes	Reserved	*1
51C0_2400h	51C0_27FFh	1	Kbyte	CMTW3 (Non-Secure)	
51C0_2000h	51C0_23FFh	1	Kbyte	CMTW2 (Non-Secure)	
51C0_1C00h	51C0_1FFFh	1	Kbyte	CMTW1 (Non-Secure)	
51C0_1800h	51C0_1BFFh	1	Kbyte	CMTW0 (Non-Secure)	
51C0_1400h	51C0_17FFh	1	Kbyte	SCIF (Non-Secure)	
51C0_1000h	51C0_13FFh	1	Kbyte	RIIC8 (Non-Secure)	
51C0_0C00h	51C0_0FFFh	1	Kbyte	RTC (Read Only) (Non-Secure)	
51C0_0800h	51C0_0BFFh	1	Kbyte	RTC (Non-Secure)	
51C0_0400h	51C0_07FFh	1	Kbyte	WDT0 (Non-Secure)	
51C0_0000h	51C0_03FFh	1	Kbytes	ADC (Non-Secure)	
51A0_0000h	51BF_FFFFh	2	Mbytes	Reserved	*1
5190_0000h	519F_FFFFh	1	Mbyte	Reserved	*1
5180_2000h	518F_FFFFh	1016	Kbytes	Reserved	*1
5180_1000h	5180_1FFFh	4	Kbytes	GTM1 (Non-Secure)	
5180_0000h	5180_0FFFh	4	Kbytes	GTM0 (Non-Secure)	
5160_0000h	517F_FFFFh	2	Mbytes	Reserved	*1

Table 1.8-2 Detailed Address Space of CM33 (6/13)

Start Address	End Address	Size		Space	Remarks
5150_0000h	515F_FFFFh	1	Mbyte	Reserved	*1
5141_0000h	514F_FFFFh	960	Kbytes	Reserved	*1
5140_0000h	5140_FFFFh	64	Kbytes	DMAC0 (Non-Secure)	
5130_0000h	513F_FFFFh	1	Mbyte	Reserved	*1
5120_0000h	512F_FFFFh	1	Mbyte	Reserved	*1
5110_0000h	511F_FFFFh	1	Mbyte	Reserved	*1
5106_0000h	510F_FFFFh	640	Kbytes	Reserved	*1
5105_0000h	5105_FFFFh	64	Kbytes	PDM1 (Non-Secure)	
5104_0000h	5104_FFFFh	64	Kbytes	PDM0 (Non-Secure)	
5103_0000h	5103_FFFFh	64	Kbytes	xSPI (REG) (Non-Secure)	
5102_0000h	5102_FFFFh	64	Kbytes	Reserved	*1
5101_0000h	5101_FFFFh	64	Kbytes	SRAM0 (REG) (Non-Secure)	
5100_1000h	5100_FFFFh	60	Kbytes	Reserved	*1
5100_0000h	5100_0FFFh	4	Kbytes	Reserved	*1
50D0_0000h	50FF_FFFFh	3	Mbytes	Reserved	*1
50C0_0000h	50CF_FFFFh	1	Mbyte	Reserved	*1
5080_0000h	50BF_FFFFh	4	Mbytes	Reserved	*1
504D_0000h	507F_FFFFh	3264	Kbytes	Reserved	*1
504C_0000h	504C_FFFFh	64	Kbytes	Reserved	*1
504B_0000h	504B_FFFFh	64	Kbytes	Reserved	*1
504A_0000h	504A_FFFFh	64	Kbytes	Reserved	*1
5049_0000h	5049_FFFFh	64	Kbytes	Reserved	*1
5048_0000h	5048_FFFFh	64	Kbytes	MHU (Non-Secure)	
5047_0000h	5047_FFFFh	64	Kbytes	TZC400 (xSPI) (Non-Secure)	
5046_0000h	5046_FFFFh	64	Kbytes	TZC400 (SRAM0) (Non-Secure)	
5045_0000h	5045_FFFFh	64	Kbytes	OTP (Non-Secure)	
5044_0000h	5044_FFFFh	64	Kbytes	Reserved	*1
5043_0000h	5043_FFFFh	64	Kbytes	SYS (Non-Secure)	
5042_0000h	5042_FFFFh	64	Kbytes	CPG (Non-Secure)	
5041_0000h	5041_FFFFh	64	Kbytes	PFC (Non-Secure)	
5040_0000h	5040_FFFFh	64	Kbytes	ICU (Non-Secure)	
5000_0000h	503F_FFFFh	4	Mbytes	Reserved	*1
4F00_0000h	4FFF_FFFFh	16	Mbytes	CST (Secure)	
4E40_0000h	4EFF_FFFFh	12	Mbytes	Reserved	*1
4E20_0000h	4E3F_FFFFh	2	Mbytes	Reserved	*1
4E10_0000h	4E1F_FFFFh	1	Mbyte	Reserved	*1
4E02_0000h	4E0F_FFFFh	896	Kbytes	Reserved	*1
4E01_0000h	4E01_FFFFh	64	Kbytes	Reserved	*1
4E00_0000h	4E00_FFFFh	64	Kbytes	DDR0 (MEMC) (Secure)	
4C00_0000h	4DFF_FFFFh	32	Mbytes	Reserved	*1
4A00_0000h	4BFF_FFFFh	32	Mbytes	DDR0 (PHY) (Secure)	
49E0_0000h	49FF_FFFFh	2	Mbytes	Reserved	*1
49D0_0000h	49DF_FFFFh	1	Mbyte	Reserved	*1
49CA_0000h	49CF_FFFFh	384	Kbytes	Reserved	*1
49C9_0000h	49C9_FFFFh	64	Kbytes	Reserved	*1

Table 1.8-2 Detailed Address Space of CM33 (7/13)

Start Address	End Address	Size		Space	Remarks
49C8_0000h	49C8_FFFFh	64	Kbytes	NPU (Secure)	
49C7_0000h	49C7_FFFFh	64	Kbytes	Reserved	*1
49C6_0000h	49C6_FFFFh	64	Kbytes	Reserved	*1
49C5_0000h	49C5_FFFFh	64	Kbytes	Reserved	*1
49C4_0000h	49C4_FFFFh	64	Kbytes	Reserved	*1
49C3_0000h	49C3_FFFFh	64	Kbytes	Reserved	*1
49C2_0000h	49C2_FFFFh	64	Kbytes	Reserved	*1
49C1_0000h	49C1_FFFFh	64	Kbytes	Reserved	*1
49C0_0000h	49C0_FFFFh	64	Kbytes	SRAM1 (REG) (Secure)	
4920_0000h	49BF_FFFFh	10	Mbytes	Reserved	*1
4910_0000h	491F_FFFFh	1	Mbyte	Reserved	*1
4900_0000h	490F_FFFFh	1	Mbyte	Reserved	*1
4800_0000h	48FF_FFFFh	16	Mbytes	Reserved	*1
4700_0000h	47FF_FFFFh	16	Mbytes	Reserved	*1
46C0_0000h	46FF_FFFFh	4	Mbytes	Reserved	*1
4680_0000h	46BF_FFFFh	4	Mbytes	Reserved	*1
4670_0000h	467F_FFFFh	1	Mbyte	Reserved	*1
4660_0000h	466F_FFFFh	1	Mbyte	Reserved	*1
4650_0000h	465F_FFFFh	1	Mbyte	Reserved	*1
464D_0000h	464F_FFFFh	192	Kbytes	Reserved	*1
464C_0000h	464C_FFFFh	64	Kbytes	LVDS (Secure)	
464B_0000h	464B_FFFFh	64	Kbytes	LCDC1 (VSPD) (Secure)	
464A_0000h	464A_FFFFh	64	Kbytes	LCDC1 (FCPVD) (Secure)	
4649_0000h	4649_FFFFh	64	Kbytes	LCDC1 (DU) (Secure)	
4648_0000h	4648_FFFFh	64	Kbytes	LCDC0 (VSPD) (Secure)	
4647_0000h	4647_FFFFh	64	Kbytes	LCDC0 (FCPVD) (Secure)	
4646_0000h	4646_FFFFh	64	Kbytes	LCDC0 (DU) (Secure)	
4645_0000h	4645_FFFFh	64	Kbytes	Reserved	*1
4644_0000h	4644_FFFFh	64	Kbytes	DSI (DPHY) (Secure)	
4643_0000h	4643_FFFFh	64	Kbytes	DSI (LINK) (Secure)	
4642_0000h	4642_FFFFh	64	Kbytes	VCD (CE) (Secure)	
4641_0000h	4641_FFFFh	64	Kbytes	VCD (FCPC) (Secure)	
4640_0000h	4640_FFFFh	64	Kbytes	VCD (VLC) (Secure)	
4632_0000h	463F_FFFFh	896	Kbytes	Reserved	*1
4631_0000h	4631_FFFFh	64	Kbytes	FDP (FCPF) (Secure)	
4630_0000h	4630_FFFFh	64	Kbytes	FDP (Secure)	
4620_0000h	462F_FFFFh	1	Mbyte	Reserved	*1
4610_0000h	461F_FFFFh	1	Mbyte	Reserved	*1
4608_0000h	460F_FFFFh	512	Kbytes	Reserved	*1
4607_0000h	4607_FFFFh	64	Kbytes	Reserved	*1
4606_0000h	4606_FFFFh	64	Kbytes	VSP (FCPVI) (Secure)	
4605_0000h	4605_FFFFh	64	Kbytes	VSP (Secure)	
4604_0000h	4604_FFFFh	64	Kbytes	Reserved	*1
4603_0000h	4603_FFFFh	64	Kbytes	Reserved	*1
4602_0000h	4602_FFFFh	64	Kbytes	Reserved	*1

Table 1.8-2 Detailed Address Space of CM33 (8/13)

Start Address	End Address	Size		Space	Remarks
4601_0000h	4601_FFFFh	64	Kbytes	Reserved	*1
4600_0000h	4600_FFFFh	64	Kbytes	CRU0 (Secure)	
45F0_0000h	45FF_FFFFh	1	Mbyte	Reserved	*1
45E0_0000h	45EF_FFFFh	1	Mbyte	Reserved	*1
45D0_0000h	45DF_FFFFh	1	Mbyte	Reserved	*1
45C5_0000h	45CF_FFFFh	704	Kbytes	Reserved	*1
45C4_0000h	45C4_FFFFh	64	Kbytes	GBETH1 (Secure)	
45C3_0000h	45C3_FFFFh	64	Kbytes	GBETH0 (Secure)	
45C2_0000h	45C2_FFFFh	64	Kbytes	SD2 (Secure)	
45C1_0000h	45C1_FFFFh	64	Kbytes	SD1 (Secure)	
45C0_0000h	45C0_FFFFh	64	Kbytes	SD0 (Secure)	
45B0_0000h	45BF_FFFFh	1	Mbyte	Reserved	*1
45A0_0000h	45AF_FFFFh	1	Mbyte	Reserved	*1
4590_0000h	459F_FFFFh	1	Mbyte	Reserved	*1
458A_0000h	458F_FFFFh	384	Kbytes	Reserved	*1
4589_0000h	4589_FFFFh	64	Kbytes	Reserved	*1
4588_0000h	4588_FFFFh	64	Kbytes	Reserved	*1
4587_0000h	4587_FFFFh	64	Kbytes	USB30 (PHY) (Secure)	
4586_0000h	4586_FFFFh	64	Kbytes	Reserved	*1
4585_0000h	4585_FFFFh	64	Kbytes	USB30 (Host) (Secure)	
4584_0000h	4584_FFFFh	64	Kbytes	USB21 (PHY) (Secure)	
4583_0000h	4583_FFFFh	64	Kbytes	USB20 (PHY) (Secure)	
4582_0000h	4582_FFFFh	64	Kbytes	USB20 (Function) (Secure)	
4581_0000h	4581_FFFFh	64	Kbytes	USB21 (Host) (Secure)	
4580_0000h	4580_FFFFh	64	Kbytes	USB20 (Host) (Secure)	
4540_0000h	457F_FFFFh	4	Mbytes	Reserved	*1
44E0_0000h	453F_FFFFh	6	Mbytes	Reserved	*1
44D0_0000h	44DF_FFFFh	1	Mbyte	Reserved	*1
44C0_0000h	44CF_FFFFh	1	Mbyte	Reserved	*1
44B0_0000h	44BF_FFFFh	1	Mbyte	Reserved	*1
44A0_0000h	44AF_FFFFh	1	Mbyte	Reserved	*1
4490_0000h	449F_FFFFh	1	Mbyte	GIC (Secure)	
4486_0000h	448F_FFFFh	640	Kbytes	Reserved	*1
4485_0000h	4485_FFFFh	64	Kbytes	GE3D (Secure)	
4484_0000h	4484_FFFFh	64	Kbytes	DMAC2 (Secure)	
4483_0000h	4483_FFFFh	64	Kbytes	DMAC1 (Secure)	
4482_0000h	4482_FFFFh	64	Kbytes	Reserved	*1
4481_0000h	4481_FFFFh	64	Kbytes	Reserved	*1
4480_0000h	4480_FFFFh	64	Kbytes	Reserved	*1
4440_3000h	447F_FFFFh	4084	Kbytes	Reserved	*1
4440_2C00h	4440_2FFFh	1	Kbyte	SPDIF2 (Secure)	
4440_2800h	4440_2BFFh	1	Kbyte	SPDIF1 (Secure)	
4440_2400h	4440_27FFh	1	Kbyte	SPDIF0 (Secure)	
4440_2000h	4440_23FFh	1	Kbyte	RIIC7 (Secure)	
4440_1C00h	4440_1FFFh	1	Kbyte	RIIC6 (Secure)	

Table 1.8-2 Detailed Address Space of CM33 (9/13)

Start Address	End Address	Size		Space	Remarks
4440_1800h	4440_1BFFh	1	Kbyte	RIIC5 (Secure)	
4440_1400h	4440_17FFh	1	Kbyte	RIIC4 (Secure)	
4440_1000h	4440_13FFh	1	Kbyte	RIIC3 (Secure)	
4440_0C00h	4440_0FFFh	1	Kbyte	RIIC2 (Secure)	
4440_0800h	4440_0BFFh	1	Kbyte	RIIC1 (Secure)	
4440_0400h	4440_07FFh	1	Kbyte	RIIC0 (Secure)	
4440_0000h	4440_03FFh	1	Kbyte	WDT1 (Secure)	
4420_0000h	443F_FFFFh	2	Mbytes	Reserved	*1
4410_0000h	441F_FFFFh	1	Mbyte	Reserved	*1
4402_0000h	440F_FFFFh	896	Kbytes	Reserved	*1
4401_0000h	4401_FFFFh	64	Kbytes	SYC (Secure)	
4400_3000h	4400_FFFFh	52	Kbytes	Reserved	*1
4400_2000h	4400_2FFFh	4	Kbytes	TSU (Secure)	
4400_1000h	4400_1FFFh	4	Kbytes	GTM3 (Secure)	
4400_0000h	4400_0FFFh	4	Kbytes	GTM2 (Secure)	
43E0_0000h	43FF_FFFFh	2	Mbytes	Reserved	*1
43D0_0000h	43DF_FFFFh	1	Mbyte	Reserved	*1
43CD_0000h	43CF_FFFFh	192	Kbytes	Reserved	*1
43CC_0000h	43CC_FFFFh	64	Kbytes	TZC400 (PCIE0) (Secure)	
43CB_0000h	43CB_FFFFh	64	Kbytes	Reserved	*1
43CA_0000h	43CA_FFFFh	64	Kbytes	TZC400 (RCPU Bus) (Secure)	
43C9_0000h	43C9_FFFFh	64	Kbytes	Reserved	*1
43C8_0000h	43C8_FFFFh	64	Kbytes	Reserved	*1
43C7_0000h	43C7_FFFFh	64	Kbytes	TZC400 (DDR_1) (Secure)	
43C6_0000h	43C6_FFFFh	64	Kbytes	TZC400 (DDR_0) (Secure)	
43C5_0000h	43C5_FFFFh	64	Kbytes	ADMAC (Secure)	
43C4_0000h	43C4_FFFFh	64	Kbytes	SSIU (DMAC) (Secure)	
43C3_0000h	43C3_FFFFh	64	Kbytes	SSIU (Secure)	
43C2_0000h	43C2_FFFFh	64	Kbytes	ADG (Secure)	
43C1_0000h	43C1_FFFFh	64	Kbytes	SCU (DMAC) (Secure)	
43C0_0000h	43C0_FFFFh	64	Kbytes	SCU (Secure)	
4380_0000h	43BF_FFFFh	4	Mbytes	Reserved	*1
4360_0000h	437F_FFFFh	2	Mbytes	Reserved	*1
4350_0000h	435F_FFFFh	1	Mbyte	Reserved	*1
4342_0000h	434F_FFFFh	896	Kbytes	Reserved	*1
4341_0000h	4341_FFFFh	64	Kbytes	Reserved	*1
4340_0000h	4340_FFFFh	64	Kbytes	PCIE0 (Secure)	
4303_0000h	433F_FFFFh	3904	Kbytes	Reserved	*1
4302_0000h	4302_FFFFh	64	Kbytes	GPT1 (Secure)	
4301_0000h	4301_FFFFh	64	Kbytes	GPT0 (Secure)	
4300_3C00h	4300_FFFFh	49	Kbytes	Reserved	*1
4300_3800h	4300_3BFFh	1	Kbyte	POEG1D (Secure)	
4300_3400h	4300_37FFh	1	Kbyte	POEG1C (Secure)	
4300_3000h	4300_33FFh	1	Kbyte	POEG1B (Secure)	
4300_2C00h	4300_2FFFh	1	Kbyte	POEG1A (Secure)	

Table 1.8-2 Detailed Address Space of CM33 (10/13)

Start Address	End Address	Size		Space	Remarks
4300_2800h	4300_2BFFh	1	Kbyte	POEG0D (Secure)	
4300_2400h	4300_27FFh	1	Kbyte	POEG0C (Secure)	
4300_2000h	4300_23FFh	1	Kbyte	POEG0B (Secure)	
4300_1C00h	4300_1FFFh	1	Kbyte	POEG0A (Secure)	
4300_1800h	4300_1BFFh	1	Kbyte	CMTW7 (Secure)	
4300_1400h	4300_17FFh	1	Kbyte	CMTW6 (Secure)	
4300_1000h	4300_13FFh	1	Kbyte	CMTW5 (Secure)	
4300_0C00h	4300_0FFFh	1	Kbyte	CMTW4 (Secure)	
4300_0800h	4300_0BFFh	1	Kbyte	CRC (Secure)	
4300_0400h	4300_07FFh	1	Kbyte	WDT3 (Secure)	
4300_0000h	4300_03FFh	1	Kbyte	WDT2 (Secure)	
42E0_0000h	42FF_FFFFh	2	Mbytes	Reserved	*1
42D0_0000h	42DF_FFFFh	1	Mbyte	Reserved	*1
42C2_0000h	42CF_FFFFh	896	Kbytes	Reserved	*1
42C1_0000h	42C1_FFFFh	64	Kbytes	Reserved	*1
42C0_4000h	42C0_FFFFh	48	Kbytes	Reserved	*1
42C0_3000h	42C0_3FFFh	4	Kbytes	GTM7 (Secure)	
42C0_2000h	42C0_2FFFh	4	Kbytes	GTM6 (Secure)	
42C0_1000h	42C0_1FFFh	4	Kbytes	GTM5 (Secure)	
42C0_0000h	42C0_0FFFh	4	Kbytes	GTM4 (Secure)	
4280_3400h	42BF_FFFFh	4083	Kbytes	Reserved	*1
4280_3000h	4280_33FFh	1	Kbyte	RSCI9 (Secure)	
4280_2C00h	4280_2FFFh	1	Kbyte	RSCI8 (Secure)	
4280_2800h	4280_2BFFh	1	Kbyte	RSCI7 (Secure)	
4280_2400h	4280_27FFh	1	Kbyte	RSCI6 (Secure)	
4280_2000h	4280_23FFh	1	Kbyte	RSCI5 (Secure)	
4280_1C00h	4280_1FFFh	1	Kbyte	RSCI4 (Secure)	
4280_1800h	4280_1BFFh	1	Kbyte	RSCI3 (Secure)	
4280_1400h	4280_17FFh	1	Kbyte	RSCI2 (Secure)	
4280_1000h	4280_13FFh	1	Kbyte	RSCI1 (Secure)	
4280_0C00h	4280_0FFFh	1	Kbyte	RSCI0 (Secure)	
4280_0800h	4280_0BFFh	1	Kbyte	RSPI2 (Secure)	
4280_0400h	4280_07FFh	1	Kbyte	RSPI1 (Secure)	
4280_0000h	4280_03FFh	1	Kbyte	RSPI0 (Secure)	
4260_0000h	427F_FFFFh	2	Mbytes	Reserved	*1
4250_0000h	425F_FFFFh	1	Mbyte	Reserved	*1
4248_0000h	424F_FFFFh	512	Kbytes	Reserved	*1
4244_0000h	4247_FFFFh	256	Kbytes	CANFD (Secure)	
4243_0000h	4243_FFFFh	64	Kbytes	Reserved	*1
4242_0000h	4242_FFFFh	64	Kbytes	Reserved	*1
4241_0000h	4241_FFFFh	64	Kbytes	Reserved	*1
4240_0000h	4240_FFFFh	64	Kbytes	I3C (Secure)	
4220_0000h	423F_FFFFh	2	Mbytes	Reserved	*1
4210_0000h	421F_FFFFh	1	Mbyte	Reserved	*1
420C_0000h	420F_FFFFh	256	Kbytes	Reserved	*1

Table 1.8-2 Detailed Address Space of CM33 (11/13)

Start Address	End Address	Size		Space	Remarks
420A_0000h	420B_FFFFh	128	Kbytes	Reserved	*1
4208_0000h	4209_FFFFh	128	Kbytes	Reserved	*1
4206_0000h	4207_FFFFh	128	Kbytes	Reserved	*1
4204_0000h	4205_FFFFh	128	Kbytes	Reserved	*1
4202_0000h	4203_FFFFh	128	Kbytes	Reserved	*1
4201_0000h	4201_FFFFh	64	Kbytes	DMAC4 (Secure)	
4200_0000h	4200_FFFFh	64	Kbytes	DMAC3 (Secure)	
41C0_2800h	41FF_FFFFh	4086	Kbytes	Reserved	*1
41C0_2400h	41C0_27FFh	1	Kbyte	CMTW3 (Secure)	
41C0_2000h	41C0_23FFh	1	Kbyte	CMTW2 (Secure)	
41C0_1C00h	41C0_1FFFh	1	Kbyte	CMTW1 (Secure)	
41C0_1800h	41C0_1BFFh	1	Kbyte	CMTW0 (Secure)	
41C0_1400h	41C0_17FFh	1	Kbyte	SCIF (Secure)	
41C0_1000h	41C0_13FFh	1	Kbyte	RIIC8 (Secure)	
41C0_0C00h	41C0_0FFFh	1	Kbyte	RTC (Read Only) (Secure)	
41C0_0800h	41C0_0BFFh	1	Kbyte	RTC (Secure)	
41C0_0400h	41C0_07FFh	1	Kbyte	WDT0 (Secure)	
41C0_0000h	41C0_03FFh	1	Kbyte	ADC (Secure)	
41A0_0000h	41BF_FFFFh	2	Mbytes	Reserved	*1
4190_0000h	419F_FFFFh	1	Mbyte	Reserved	*1
4180_2000h	418F_FFFFh	1016	Kbytes	Reserved	*1
4180_1000h	4180_1FFFh	4	Kbytes	GTM1 (Secure)	
4180_0000h	4180_0FFFh	4	Kbytes	GTM0 (Secure)	
4160_0000h	417F_FFFFh	2	Mbytes	Reserved	*1
4150_0000h	415F_FFFFh	1	Mbyte	Reserved	*1
4141_0000h	414F_FFFFh	960	Kbytes	Reserved	*1
4140_0000h	4140_FFFFh	64	Kbytes	DMAC0 (Secure)	
4130_0000h	413F_FFFFh	1	Mbyte	Reserved	*1
4120_0000h	412F_FFFFh	1	Mbyte	Reserved	*1
4110_0000h	411F_FFFFh	1	Mbyte	Reserved	*1
4106_0000h	410F_FFFFh	640	Kbytes	Reserved	*1
4105_0000h	4105_FFFFh	64	Kbytes	PDM1 (Secure)	
4104_0000h	4104_FFFFh	64	Kbytes	PDM0 (Secure)	
4103_0000h	4103_FFFFh	64	Kbytes	xSPI (REG) (Secure)	
4102_0000h	4102_FFFFh	64	Kbytes	Reserved	*1
4101_0000h	4101_FFFFh	64	Kbytes	SRAM0 (REG) (Secure)	
4100_1000h	4100_FFFFh	60	Kbytes	Reserved	*1
4100_0000h	4100_0FFFh	4	Kbytes	Reserved	*1
40D0_0000h	40FF_FFFFh	3	Mbytes	Reserved	*1
40C0_0000h	40CF_FFFFh	1	Mbyte	Reserved	*1
4080_0000h	40BF_FFFFh	4	Mbytes	Reserved	*1
404D_0000h	407F_FFFFh	3264	Kbytes	Reserved	*1
404C_0000h	404C_FFFFh	64	Kbytes	Reserved	*1
404B_0000h	404B_FFFFh	64	Kbytes	Reserved	*1
404A_0000h	404A_FFFFh	64	Kbytes	Reserved	*1

Table 1.8-2 Detailed Address Space of CM33 (12/13)

Start Address	End Address	Size		Space	Remarks
4049_0000h	4049_FFFFh	64	Kbytes	Reserved	*1
4048_0000h	4048_FFFFh	64	Kbytes	MHU (Secure)	
4047_0000h	4047_FFFFh	64	Kbytes	TZC400 (xSPI) (Secure)	
4046_0000h	4046_FFFFh	64	Kbytes	TZC400 (SRAM0) (Secure)	
4045_0000h	4045_FFFFh	64	Kbytes	OTP (Secure)	
4044_0000h	4044_FFFFh	64	Kbytes	Reserved	*1
4043_0000h	4043_FFFFh	64	Kbytes	SYS (Secure)	
4042_0000h	4042_FFFFh	64	Kbytes	CPG (Secure)	
4041_0000h	4041_FFFFh	64	Kbytes	PFC (Secure)	
4040_0000h	4040_FFFFh	64	Kbytes	ICU (Secure)	
4000_0000h	403F_FFFFh	4	Mbytes	Reserved	*1
3860_0000h	3FFF_FFFFh	122	Mbytes	Reserved	*1
3820_0000h	385F_FFFFh	4	Mbytes	Reserved	*1
3818_0000h	381F_FFFFh	512	Kbytes	Reserved	*1
3810_0000h	3817_FFFFh	512	Kbytes	Reserved	*1
3808_0000h	380F_FFFFh	512	Kbytes	Reserved	*1
3804_0000h	3807_FFFFh	256	Kbytes	SRAM1 (Data, Non-Secure)	
3800_0000h	3803_FFFFh	256	Kbytes	SRAM0 (Data, Non-Secure)	
37F8_2000h	37FF_FFFFh	504	Kbytes	Reserved	*1
37F8_0000h	37F8_1FFFh	8	Kbytes	Reserved	*1
3002_0000h	37F7_FFFFh	130304	Kbytes	Reserved	*1
3000_0000h	3001_FFFFh	256	Kbytes	Reserved	*1
2860_0000h	2FFF_FFFFh	122	Mbytes	Reserved	*1
2820_0000h	285F_FFFFh	4	Mbytes	Reserved	*1
2818_0000h	281F_FFFFh	512	Kbytes	Reserved	*1
2810_0000h	2817_FFFFh	512	Kbytes	Reserved	*1
2808_0000h	280F_FFFFh	512	Kbytes	Reserved	*1
2804_0000h	2807_FFFFh	256	Kbytes	SRAM1 (Data, Secure)	
2800_0000h	2803_FFFFh	256	Kbytes	SRAM0 (Data, Secure)	
27F8_2000h	27FF_FFFFh	504	Kbytes	Reserved	*1
27F8_0000h	27F8_1FFFh	8	Kbytes	Reserved	*1
2002_0000h	27F7_FFFFh	130304	Kbytes	Reserved	*1
2000_0000h	2001_FFFFh	256	Kbytes	Reserved	*1
1860_0000h	1FFF_FFFFh	122	Mbytes	Reserved	*1
1820_0000h	185F_FFFFh	4	Mbytes	Reserved	*1
1818_0000h	181F_FFFFh	512	Kbytes	Reserved	*1
1810_0000h	1817_FFFFh	512	Kbytes	Reserved	*1
1808_0000h	180F_FFFFh	512	Kbytes	Reserved	*1
1804_0000h	1807_FFFFh	256	Kbytes	SRAM1 (Code, Non-Secure)	
1800_0000h	1803_FFFFh	256	Kbytes	SRAM0 (Code, Non-Secure)	
17F8_2000h	17FF_FFFFh	504	Kbytes	Reserved	*1
17F8_0000h	17F8_1FFFh	8	Kbytes	Reserved	*1
1002_0000h	17F7_FFFFh	130304	Kbytes	Reserved	*1
1000_0000h	1001_FFFFh	256	Kbytes	Reserved	*1
0860_0000h	0FFF_FFFFh	122	Mbytes	Reserved	*1

Table 1.8-2 Detailed Address Space of CM33 (13/13)

Start Address	End Address	Size		Space	Remarks
0820_0000h	085F_FFFFh	4	Mbytes	Reserved	*1
0818_0000h	081F_FFFFh	512	Kbytes	Reserved	*1
0810_0000h	0817_FFFFh	512	Kbytes	Reserved	*1
0808_0000h	080F_FFFFh	512	Kbytes	Reserved	*1
0804_0000h	0807_FFFFh	256	Kbytes	SRAM1 (Code, Secure)	
0800_0000h	0803_FFFFh	256	Kbytes	SRAM0 (Code, Secure)	
07F8_2000h	07FF_FFFFh	504	Kbytes	Reserved	*1
07F8_0000h	07F8_1FFFh	8	Kbytes	Reserved	*1
0002_0000h	07F7_FFFFh	130304	Kbytes	Reserved	*1
0000_0000h	0001_FFFFh	256	Kbytes	Reserved	*1

Note 1. Access to the reserved areas is prohibited. If access to the reserved area is attempted, incorrect operation may occur.

Note 2. For details on the PPB/Vendor_SYS area, refer to “Arm Cortex-M33 Processor Technical Reference Manual”.

SECTION 1 OVERVIEW

1.9 Boot Operation

This section describes the boot operation of this LSI.

Boot mode is determined by the settings on external pins or at the time of release from the reset state. This LSI has three types of boot mode: CPU (CA55 or CM33) normal execution mode, boundary scan mode, and debug mode.

1.9.1 CPU (CA55 or CM33) Normal Execution Mode

The LSI has two types of execution method: CM33 cold boot and CA55 cold boot. BOOTSELCPU controls the CPU from which a cold boot is initiated. **Table 1.9-1** describes BOOTSELCPU.

During a CM33 cold boot, booting from either the serial flash memory or through SCIF downloading is selectable. During a CA55 cold boot, booting from any among an embedded SD memory (eSD), eMMC, serial flash memory, through USB downloading, or through SCIF downloading is selectable. The MD_BOOT0, MD_BOOT1 and MD_BOOT2 pins control the device from which booting is initiated. **Table 1.9-2** describes MD_BOOT0, MD_BOOT1, and MD_BOOT2.

Table 1.9-1 Boot CPU Selection Pin

Signal Name	Function	Description
BOOTSELCPU	Boot CPU selection	Selects whether to initiate a cold boot from the CA55 or CM33 0: CM33 cold boot 1: CA55 cold boot

Table 1.9-2 Boot Device Selection Pins

Signal Name			Mode	Selected Module	Selected Device	CA55 Boot	CM33 Boot
MD_BOOT_2	MD_BOOT_1	MD_BOOT_0					
0	0	0	Boot mode 0	SD/eMMC ch0	eSD (3.3 V for execution)*2	Supported*1	Not supported
0	0	1	Boot mode 1	SD/eMMC ch0	eMMC (IO voltage is 3.3 V.)	Supported*1	Not supported
0	1	0	Boot mode 2	xSPI	Serial flash memory (IO voltage is 3.3 V.)	Supported*1	Supported*1
0	1	1	Boot mode 3	SCIF ch0	SCIF download	Supported	Supported
1	0	0	Boot mode 4	USB2.0 ch0	USB download	Supported*1	Not supported
1	0	1	Boot mode 5	SD/eMMC ch0	eMMC (IO voltage is 1.8 V.)	Supported*1	Not supported
1	1	0	Boot mode 6	xSPI	Serial flash memory (IO voltage is 1.8 V.)	Supported*1	Supported*1

Note 1. In the boot modes other than SCIF download mode, if reading or writing data from the opposite device fails, the mode shifts to SCIF download mode. If SCIF download mode processing fails, an infinite loop occurs within the built-in ROM and the boot processing ends. (In this case, SCIF ch0 outputs “SCIF Download mode due to parameter error”.)

Note 2. The standard that eSD Boot conforms to is defined in *SD Specification Part 1 eSD Addendum (Version 2.10)*. When using this mode, eSD should be mounted on a board. This mode can be used with an SD card only for evaluation purposes. Note that the error reset function cannot be used when operating with an SD card.

1.9.1.1 Operation of Booting

1.9.1.1.1 Boot Mode 0 (eSD)

Embedded SD Boot (hereafter eSD boot) performs boot processing from the loader program stored in the eSD device connected to channel 0 of the SD/MMC host interface (hereafter SD0).

In execution in this mode, the voltage is 3.3 V and the width of the data bus is always 4 bits. The controller (SD0) for the device to be used in booting is in the module standby state at the time of execution. Execution of the ROM program proceeds after release from module standby, and involves setting the CPG, selecting of the clock that produces the external clock signal, and initializing SD0.

When the LSI starts up in this mode, the following processing is performed.

1. Setting peripheral modules (SD0, PFC).
2. Connecting the eSD device.
3. Issuing a read command to the eSD device from SD0, getting the loader program size block from sector 1, and storing it at 0_0810_1E00h of the SRAM.
4. Getting the following information from the loader program size block stored in step 3.
   (A) Loader program size
   (B) Loader program load address
   (C) Loader program destination address
5. According to the information obtained in step 4, deploying the loader program for the loader program size (A) from the eSD device to the loader program destination address (C) in the SRAM.
6. Initializing SD0.
7. The device information and boot results are stored in the SRAM.
8. If the processing up to step 7 is completed normally, the program counter is moved to the loader program destination address and the loader program obtained from the eSD is executed.

(1) Information on related pins

Table 1.9-3 lists interface signals with the external device in boot mode 0 (eSD).

Table 1.9-3 External Interface Signals in Boot Mode 0

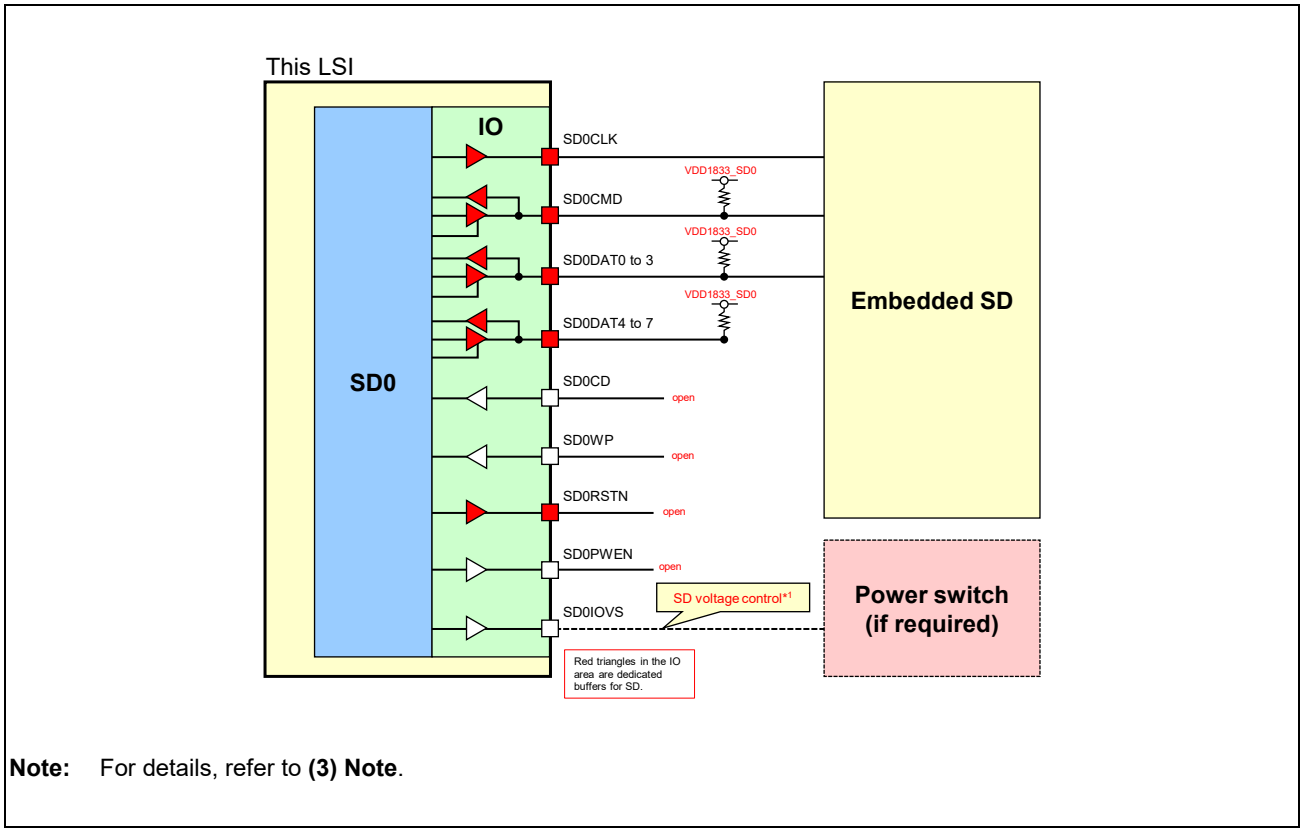
Pin Name	Input/Output	Function
SD0CLK	Output	SD clock
SD0CMD	I/O	SD command/response
SD0DAT0	I/O	SD data 0
SD0DAT1	I/O	SD data 1
SD0DAT2	I/O	SD data 2
SD0DAT3	I/O	SD data 3

Note: Sharing of the bus between the embedded SDIO and an 8-bit SD bus is not supported.

(2) External connection

Figure 1.9-1 is a schematic diagram of the connections of the IO pins of SD0 and the external device.
The boot program does not monitor the states of SD0CD and SD0WP.

SD0PWEN and SD0IOVS are not controlled by the boot program. Control the SD power supply enable with a signal different from SD0PWEN. Also, control the SD0IOVS pin as necessary in the program after boot operation.



Note: For details, refer to **(3) Note**.

Figure 1.9-1 Schematic Diagram of Connections for Boot Mode 0

(3) Note

Boot operation using SD card is available for evaluation purposes only.

To speed up the access after booting, handshaking with the other devices, changing to 1.8 V if required, and access between the LSI and the SD card is made by the user program. At this time, changing of voltage is handled by transfer among this device, the other device, and the power management IC or power switch. However, changing of speed is executed by the user program stored in the SD card because it is dependent on the user system.

Figure 1.9-2 is a schematic diagram of the connections when the voltage is to be changed after booting.

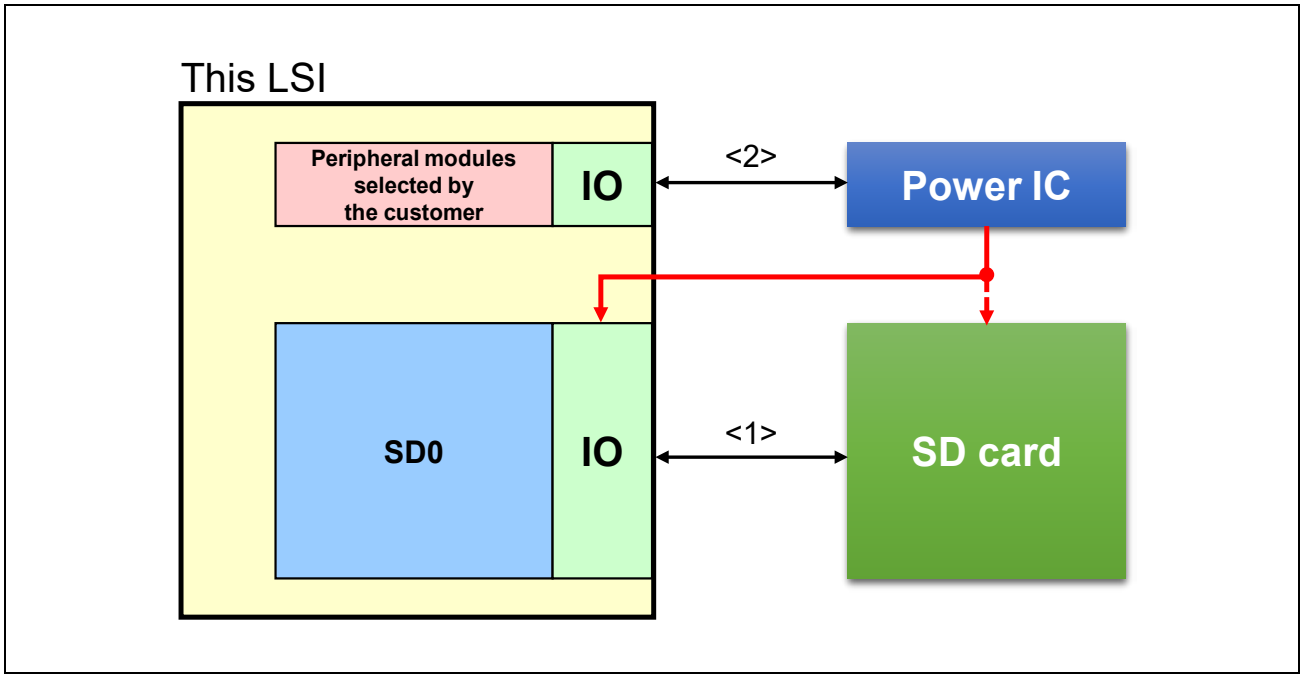


Figure 1.9-2 Schematic Diagram of SD Voltage Control in Boot Mode 0

1.9.1.1.2 Boot Mode 1 / Boot Mode 5 (eMMC 3.3V / 1.8V)

eMMC Boot performs boot processing from the loader program stored in the eMMC device connected to channel 0 of the SD/MMC host interface (hereafter SD0).

When eMMC is selected as the boot device, the width of the data bus is limited to 8 bits. The controller (SD0) for the device to be used in booting is in the module standby state at the time of execution. Execution of the ROM program proceeds after release from module standby, and involves setting the CPG, selecting of the clock that produces the external clock signal, and initializing SD0.

When the LSI starts up in this mode, the following processing is performed.

1. Setting peripheral modules (SD0, PFC, GTM0).
2. Setting the IO buffer (drive strength etc.) at 1.8 V or 3.3 V.
3. Starting the alternative boot operation mode of the eMMC device.
   Read access begins from sector 0 of the partition selected by the [179] field (PARTITION_CONFIG) of the extended CSD (EXT_CSD) register in the eMMC device. For details, refer to **(3) Alternative Boot Operation**.
4. Dummy reading from sector 0, getting the loader program size block from sector 1, and storing it at 0_0810_1E00h of the SRAM.
5. Getting the following information from the loader program size block stored in step 4.
   (A) Loader program size
   (B) Loader program load address
   (C) Loader program destination address
6. According to the information obtained in step 5, deploying the loader program for the sector 2 loader program size (A) from the eMMC device to the loader program destination address (C) in the SRAM.
7. Initializing SD0.
8. The device information and boot results are stored in the SRAM.
9. If the processing up to step 8 is completed normally, the program counter is moved to the loader program destination address and the loader program obtained from the eMMC is executed.

(1) Information on related pins

Table 1.9-4 lists interface signals with the external devices in boot mode 1 / boot mode 5 (eMMC).

Table 1.9-4 External Interface Signals in Boot Mode 1 / Boot Mode 5 (eMMC)

Pin Name	Input/Output	Function
SD0CLK	Output	eMMC clock
SD0CMD	I/O	eMMC command/response
SD0DAT0	I/O	eMMC data 0
SD0DAT1	I/O	eMMC data 1
SD0DAT2	I/O	eMMC data 2
SD0DAT3	I/O	eMMC data 3
SD0DAT4	I/O	eMMC data 4
SD0DAT5	I/O	eMMC data 5
SD0DAT6	I/O	eMMC data 6
SD0DAT7	I/O	eMMC data 7
SD0RSTN	Output	eMMC reset

Note: Booting from an MMC card is not supported.

Note: The only supported data width is 8 bits; that is, booting through a 1-bit or 4-bit interface is not supported.

(2) External connection

Figure 1.9-3 is a schematic diagram of the connection with eMMC.
The states of SD0CD and SD0WP are not used by the boot program.

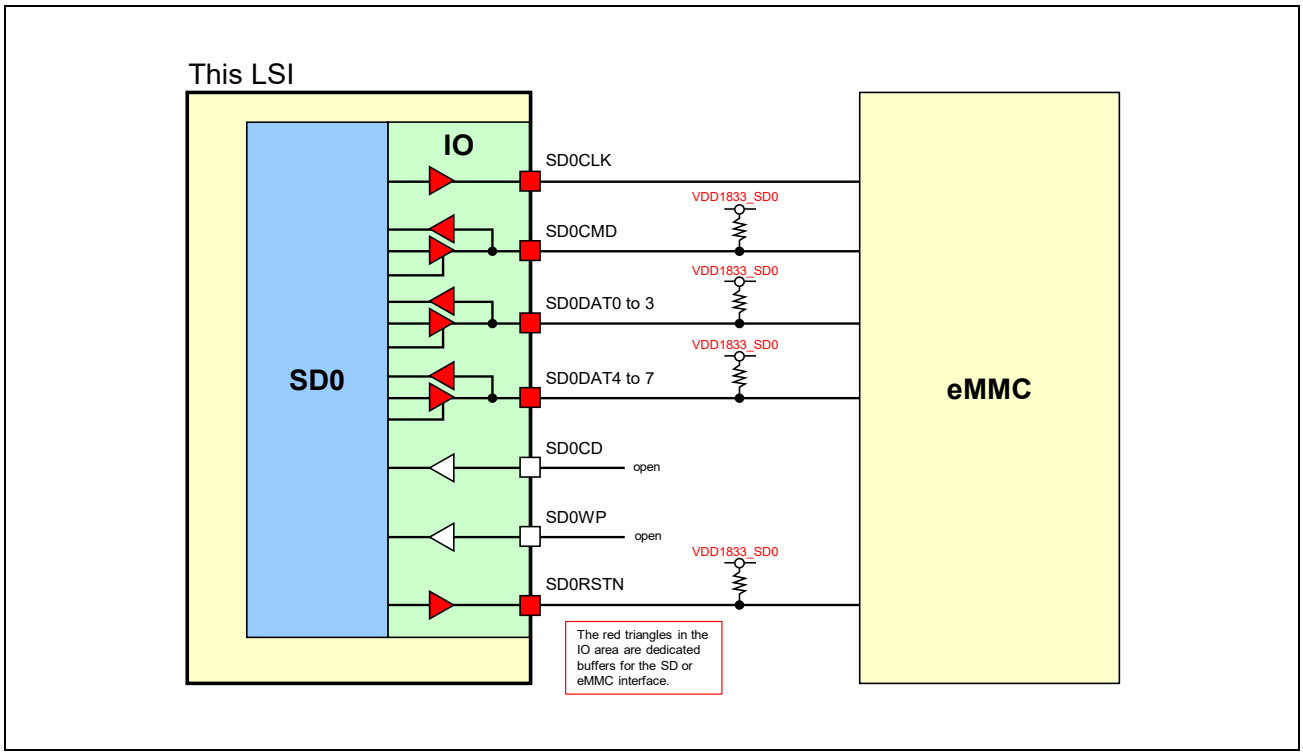
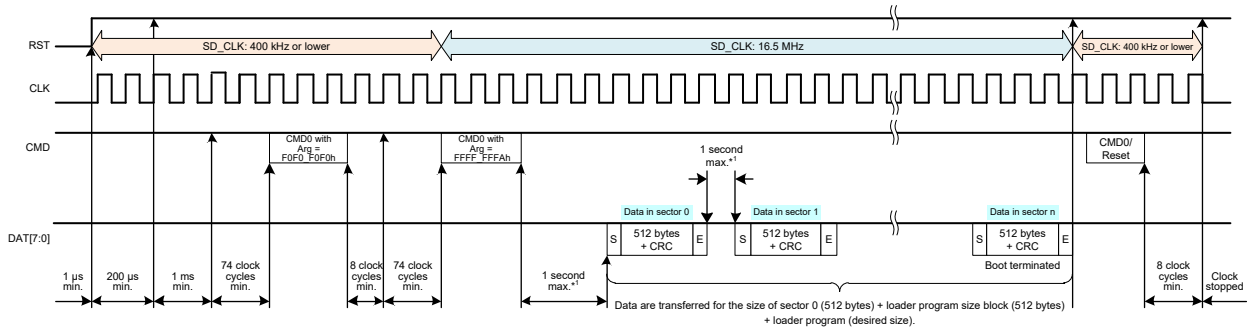


Figure 1.9-3 Schematic Diagram of Connections for Boot Mode 1 / Boot Mode 5

(3) Alternative Boot Operation

Figure 1.9-4 shows the alternative boot operation to read the loader program. This boot program sets up the registers in SD0 so that the SD0 operates with an 8-bit bus width when booting from the eMMC device. On the eMMC device side, the data bus width in the alternative boot operation is determined by the setting of the BOOT_BUS_WIDTH[177] field in the extended CSD register. To boot up from the eMMC device correctly, set up the extended CSD register in the eMMC device as shown in **Table 1.9-5** when writing a program to the eMMC device.



- This CPU accesses the eMMC device with a data bus width of 8 bits in the alternative boot operation. The data bus width of the eMMC device can be specified in the BOOT_BUS_WIDTH[177] field of the extended CSD register. When writing a loader program to the eMMC device, specify the 8-bit bus width in this field.
- The eMMC device sequentially outputs data from sector 0 of the partition selected in the PARTITION_CONFIG[179] field of the extended CSD register in the alternative boot operation.
- This CPU does not detect Boot Ack.
  The output of Boot Ack from the eMMC device can be specified in the PARTITION_CONFIG[179] field. Be sure to stop the output of Boot Ack by using this field.

Note 1. For a maximum of one second, the BRE, ERR3, and ERR1 bits of the SD_INFO2 register are polled to check whether data have been received, a timeout has occurred, and a CRC error has occurred, respectively.
<Normal operation>
SD_INFO2.BRE = 1
Data are transferred from SD_BUF to the SRAM.
<Error>
CRC error: SD_INFO2.ERR1 = 1
The loader program is read from a reserved area.
Timeout error: SD_INFO2.ERR3 = 1
Execution of the boot program for eMMC is terminated.

Figure 1.9-4 Alternative Boot Operation to Read the Loader Program

Table 1.9-5 Setting of the Data Bus Width in Alternative Boot Operation

Field in Extended CSD Register of eMMC Device	Setting
BOOT_BUS_WIDTH [177]	Bits 1 and 0: BOOT_BUS_WIDTH = 10b (The bus width is set to 8 bits in alternative boot operation.)

■ Partitions

An eMMC device has multiple partitions (**Figure 1.9-5**). The partition used for booting can be specified in the PARTITION_CONFIG[179] field of the extended CSD (EXT_CSD) register. To boot up correctly, specify the boot partition in the extended CSD (EXT_CSD) register of the eMMC device as shown in **Table 1.9-6** when writing a program to the eMMC device.

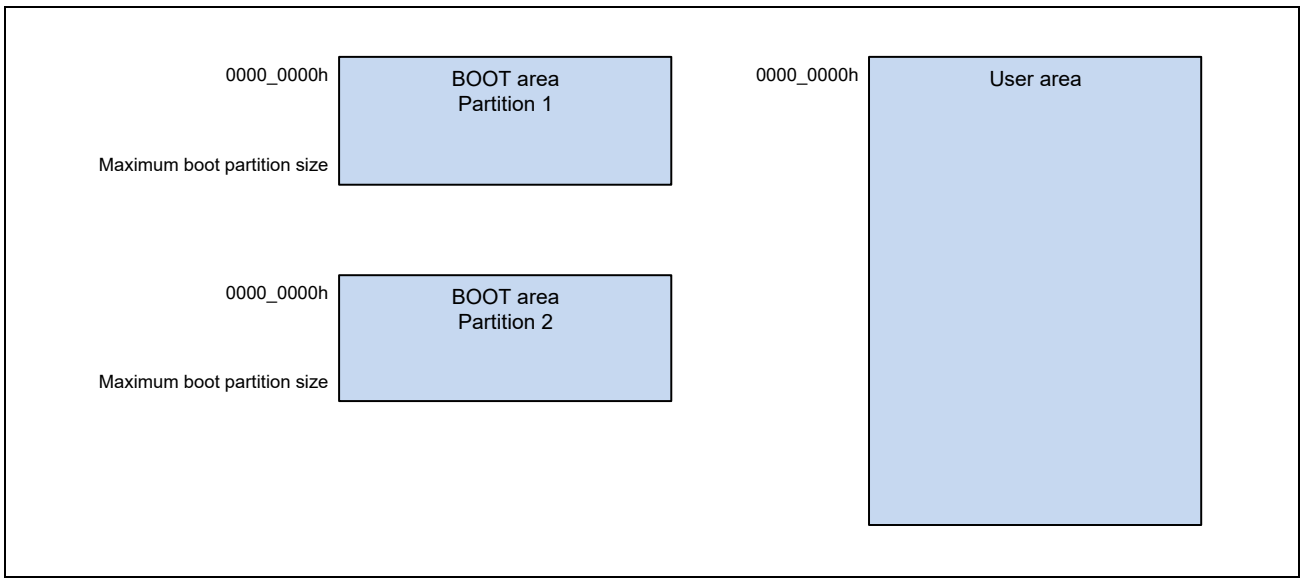


Figure 1.9-5 Partitions in the eMMC Device

Table 1.9-6 Boot Partition Setting in the Alternative Boot Operation

Field in Extended CSD Register of eMMC Device	Setting
PARTITION_CONFIG [179]	• Set bit 6 (BOOT_ACK) to 0b (the eMMC device does not output a boot acknowledge in the alternative boot operation). • Specify the boot partition in bits 5 to 3 (BOOT_PARTITION_ENABLE). 000b: Booting is disabled. 001b: Booting from sector 0 in boot partition 1 in the boot operation. 010b: Booting from sector 0 in boot partition 2 in the boot operation. 011b to 110b: Reserved 111b: Booting from sector 0 in the user area in the boot operation.

1.9.1.1.3 Boot Mode 2 / Boot Mode 6 (Serial Flash Memory 3.3V / 1.8V)

xSPI Boot performs boot processing from the loader program stored in the serial flash memory connected to channel 0 of the xSPI interface.

This mode is for execution from a user program stored in an externally connected single, quad, or octa flash device. Starting up is accompanied by handshaking through the SPI protocol with the single, quad, or octa flash device, so the ROM boot program controls the pins labeled ■ in the figures of **(2) External connection**, and subsequent access is controlled by the stored user program.

When the LSI starts up in this mode, the following processing is performed.

1. Release from xSPI module standby.
2. Setting peripheral modules (xSPI0, PFC, GTM0).
3. Setting the IO buffer (drive strength etc.) at 1.8 V or 3.3 V.
4. Getting the loader program size block stored at the start address (0_2000_0000h) of the external address space from xSPI, and storing it at 0_0810_1E00h of the SRAM.
5. Getting the following information from the loader program size block stored in step 4.
   (A) Loader program size
   (B) Loader program load address
   (C) Loader program destination address
6. According to the information in step 5, deploying the loader program for the loader program size (A) to the loader program destination address (C) in the SRAM.
7. The device information and boot results are stored in the SRAM.
8. If the processing up to step 7 is completed normally, the program counter is moved to the loader program destination address and the loader program obtained from the serial flash memory is executed.

(1) Information on related pins

Table 1.9-7 lists interface signals with the external devices in boot mode 2 / boot Mode 6 (serial flash memory).

Table 1.9-7 External Interface Signals in Boot Mode 2 / Boot Mode 6

Pin Name	Input/Output	Function
XSPI0_CKP	Output	xSPI clock (positive)
XSPI0_CKN	Output	xSPI clock (negative)
XSPI0_CS0N	Output	xSPI chip selection
XSPI0_DS	I/O	xSPI read data strobe/write data mask
XSPI0_IO0	I/O	xSPI data 0
XSPI0_IO1	I/O	xSPI data 1
XSPI0_IO2	I/O	xSPI data 2
XSPI0_IO3	I/O	xSPI data 3
XSPI0_IO4	I/O	xSPI data 4
XSPI0_IO5	I/O	xSPI data 5
XSPI0_IO6	I/O	xSPI data 6
XSPI0_IO7	I/O	xSPI data 7
XSPI0_RESET0N	Output	xSPI master reset
XSPI0_RSTO0N	Input	xSPI slave reset
XSPI0_INT0N	Input	xSPI interrupt
XSPI0_ECS0N	Input	xSPI error correction

(2) External connection

Figure 1.9-6 is a schematic diagram of the connection with a single SPI memory, **Figure 1.9-7** with a quad SPI memory, and **Figure 1.9-8** with a flash memory (octal).

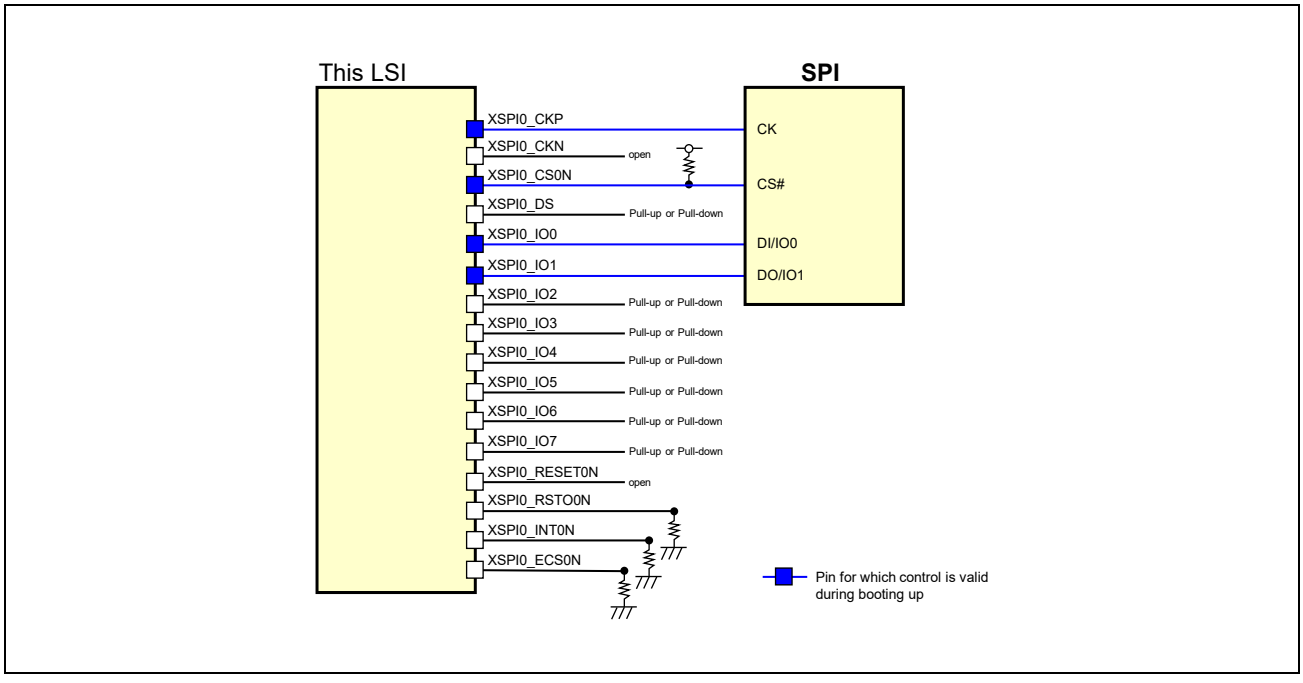


Figure 1.9-6 Schematic Diagram of Connection with a Single SPI Device in Boot Mode 2 / Boot Mode 6

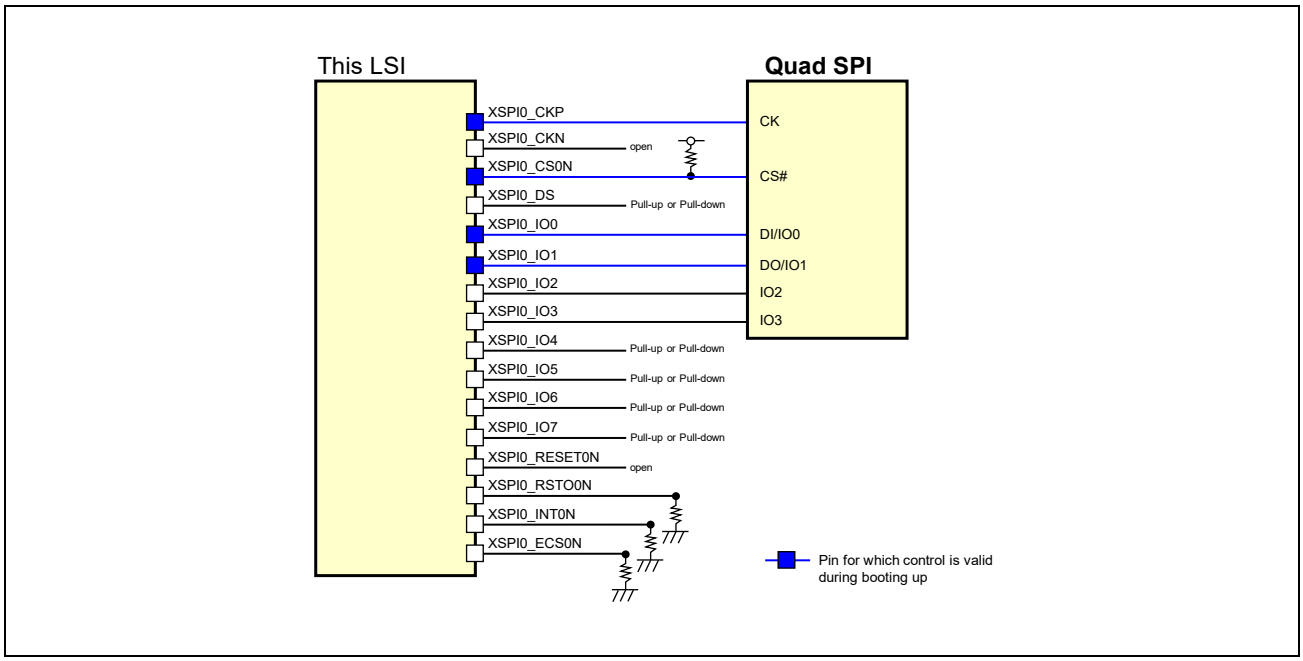


Figure 1.9-7 Schematic Diagram of Connection with a Quad SPI Device in Boot Mode 2 / Boot Mode 6

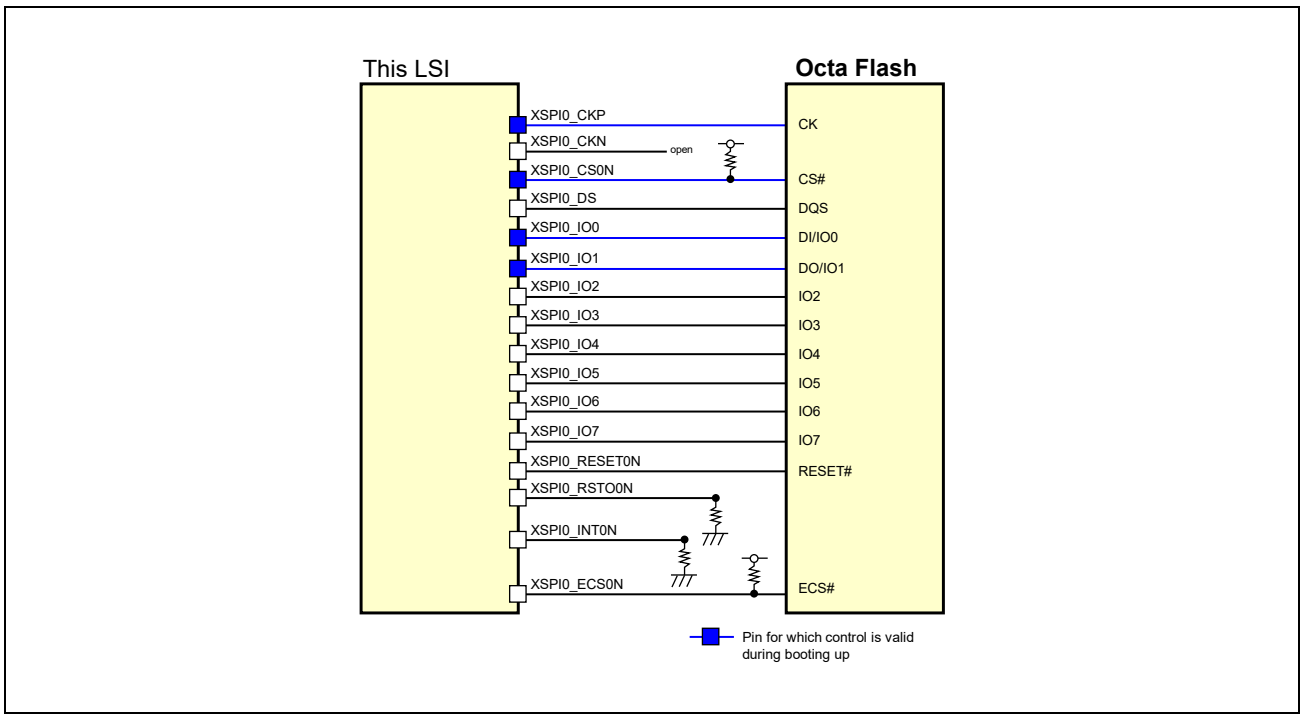


Figure 1.9-8 Schematic Diagram of Connection with an Octa Flash Device in Boot Mode 2 / Boot Mode 6

1.9.1.1.4 Boot Mode 3 (SCIF Download)

This mode connects the RS232C port of this LSI to an external PC and downloads data stored in the PC.

If an external PC does not have RS-232C ports, the connection is made via a serial-to-USB conversion chip mounted on a board or in the cable.

When the LSI starts up in this mode, the following processing is performed.

1. Setting up the necessary peripheral modules (SCIF channel 0 and PFC) and specifying the parameters for communications.

   When the LSI becomes ready for data reception, the boot program outputs the following message to the SCIF.
   "SCIF Download mode"

2. Upon receiving the Motorola S0 record sent from the external host PC through the SCIF, the boot program converts the S3 record to binary data and copies the data to the addresses 0_0001_2000h to 0_0002_EFFEh (up to 116 Kbytes) of the SRAM.

   The boot program outputs the following message to the SCIF.
   "-- Load Program to System RAM ----------"

3. Upon receiving the S7 record, the boot program outputs the following message to the SCIF and terminates copying to the SRAM.
   "-- Start Boot Program on System RAM ---------"

   After that, execution branches to the address 0_0001_2000h of the SRAM.

(1) Supported file format

Prepare the files used in this mode so that it meets the following specifications.

- SRECORD format
- The S0 record must always be generated
  (because the boot program requires an S0 record as a trigger for the reception processing)
- S0, S3, and S7 only
- The role of each format:
  S0, S7: Only use the byte count and checksum fields
  S3: Transfer boot parameters and boot loader in the record

(2) When the file format is out of specifications

If the data in the Motorola S records sent from the external host PC satisfies any of the following error conditions, the corresponding error message is output to the SCIF and copying to the SRAM is aborted.

— An S1 or S2 record is received: "Invalid Record Type Error!!! "

— An address outside the SRAM area is specified: "Address Error!!! "

— An illegal character code is found: "Invalid Character Error!!! "

— An illegal byte count is found: "Invalid Byte Count Error!!! "

The data transmission and reception operation in SCIF communications (asynchronous) is shown in **Figure 1.9-9**.

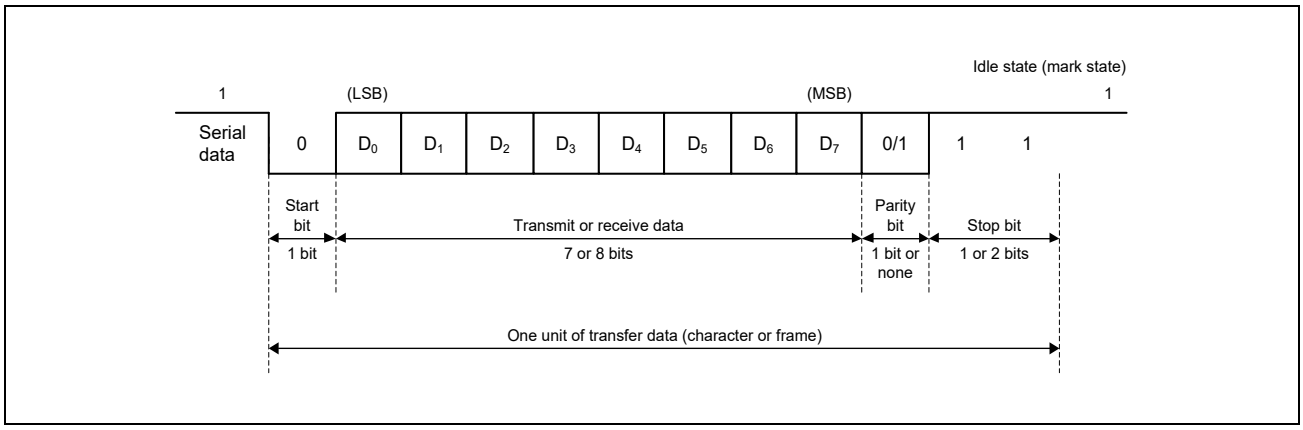


Figure 1.9-9 Data Transmission and Reception Operation in SCIF Communications

(1) Information on related pins

Table 1.9-8 lists interface signals with the external devices in boot mode 3 (SCIF download).

Table 1.9-8 External Interface Signals in Boot Mode 3

Pin Name	Input/Output	Function
SCIF_RXD	Input	SCIF received data
SCIF_TXD	Output	SCIF transmitted data

(2) External connection

Figure 1.9-10 is a schematic diagram of the connections for downloading through the SCIF.

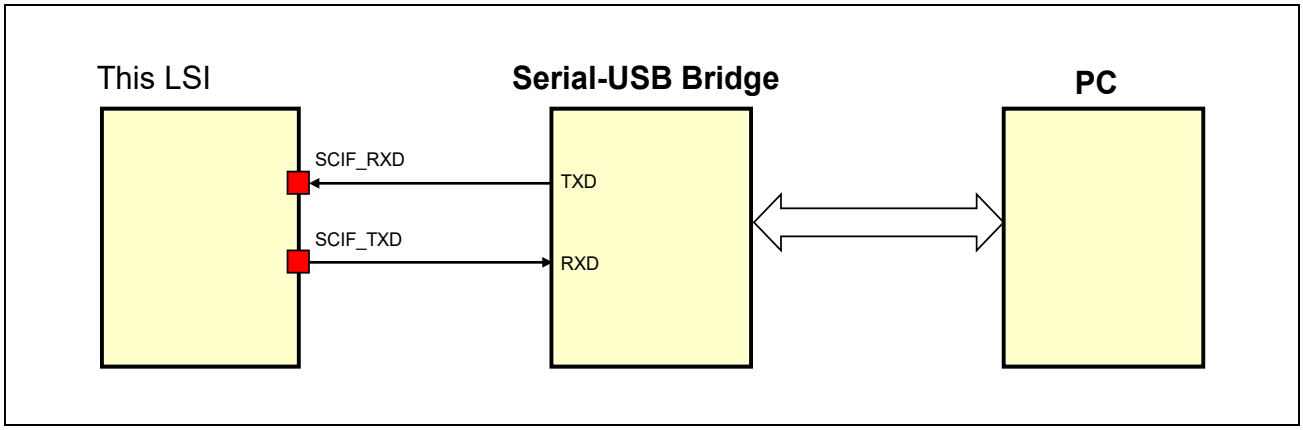


Figure 1.9-10 Schematic Diagram of Connections for Boot Mode 3

1.9.1.1.5 Boot Mode 4 (USB Download)

This mode supports downloads and uses the USB 2.0 interface to run the loader program.

This can be done directly in the internal memory. When the LSI receives an open request from the PC, the LSI sends a start message to the PC. The PC starts transmitting the loader program, which is transferred to the LSI internal memory. After the transmission is complete, the loader program process is executed.
If you start the boot process in USB boot mode after the reset is canceled, the following settings are used.

- USB mode: HS mode, Function mode
- USB function: USB Communication Device Class (CDC)

When the LSI starts up in this mode, the following processing is performed.

1. If the initial setting is successfully completed and the reception is available, input will be waited.
2. If a single character is entered, the following message is output.

   “USB Download mode (Normal USB boot)”

   From now on, it will be the same as the SCIF download.

(1) Information on related pins

Table 1.9-9 lists interface signals with the external devices in boot mode 4 (USB Download).

Table 1.9-9 External Interface Signals in Boot Mode 4

Signal Name	Pin Name	Input/Output	Function	Pin Classification
USB	USB20_QDP	I/O	USB2 DP Signal	Dedicated pins
	USB20_QDM	I/O	USB2 DM Signal	
	USB20_VBUSIN	I/O	USB2 VBUS	
	USB20_TXRTUNE	I/O	USB2 Transmitter Tune Pin	
	USB20_OTG_ID	I/O	OTG ID input	

Note: The signals that represent the interface with the other device are excerpted in the above table.

(2) External connection

Figure 1.9-11 is a schematic diagram of the connections for downloading through the USB.

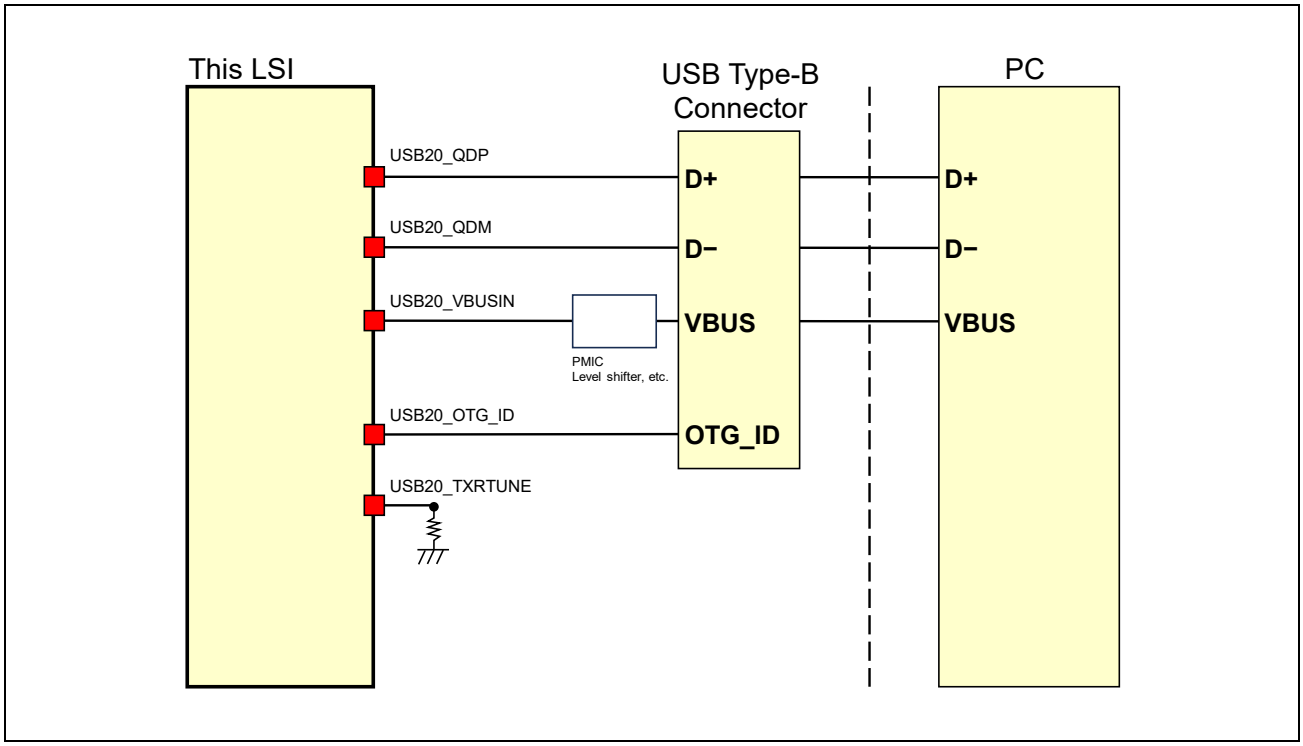


Figure 1.9-11 Schematic Diagram of Connections for Boot Mode 4

(3) Note

- After the USB cable is connected to the LSI, do not disconnect the USB cable until the loader program is executed.
- The USB20_TXRTUNE is connected to GND via a reference resistor. The resistor value must be connected to the default resistance value. Please follow the product guide.

1.9.1.2 Allocation of the Loader Program

1.9.1.2.1 Boot Mode 0 (eSD)

(1) Allocation in eSD V2.0 (Single Partition)

Figure 1.9-12 shows the allocation of the loader program size blocks and loader programs in the eSD device conforming to the eSD V2.0 standard. To prevent read-disturb errors, up to seven loader program size blocks and loader programs can be multiplexed and written to the eSD device.

The loader program size blocks are allocated to sectors 1 to 7. In a loader program size block, the loader program size is stored in the first four bytes and the signature AA55h is stored in the last two bytes. **Figure 1.9-13** shows the structure of the loader program size data. Be sure to store the loader program size in 4-byte little endian.

The following describes the operation of transferring the loader program.

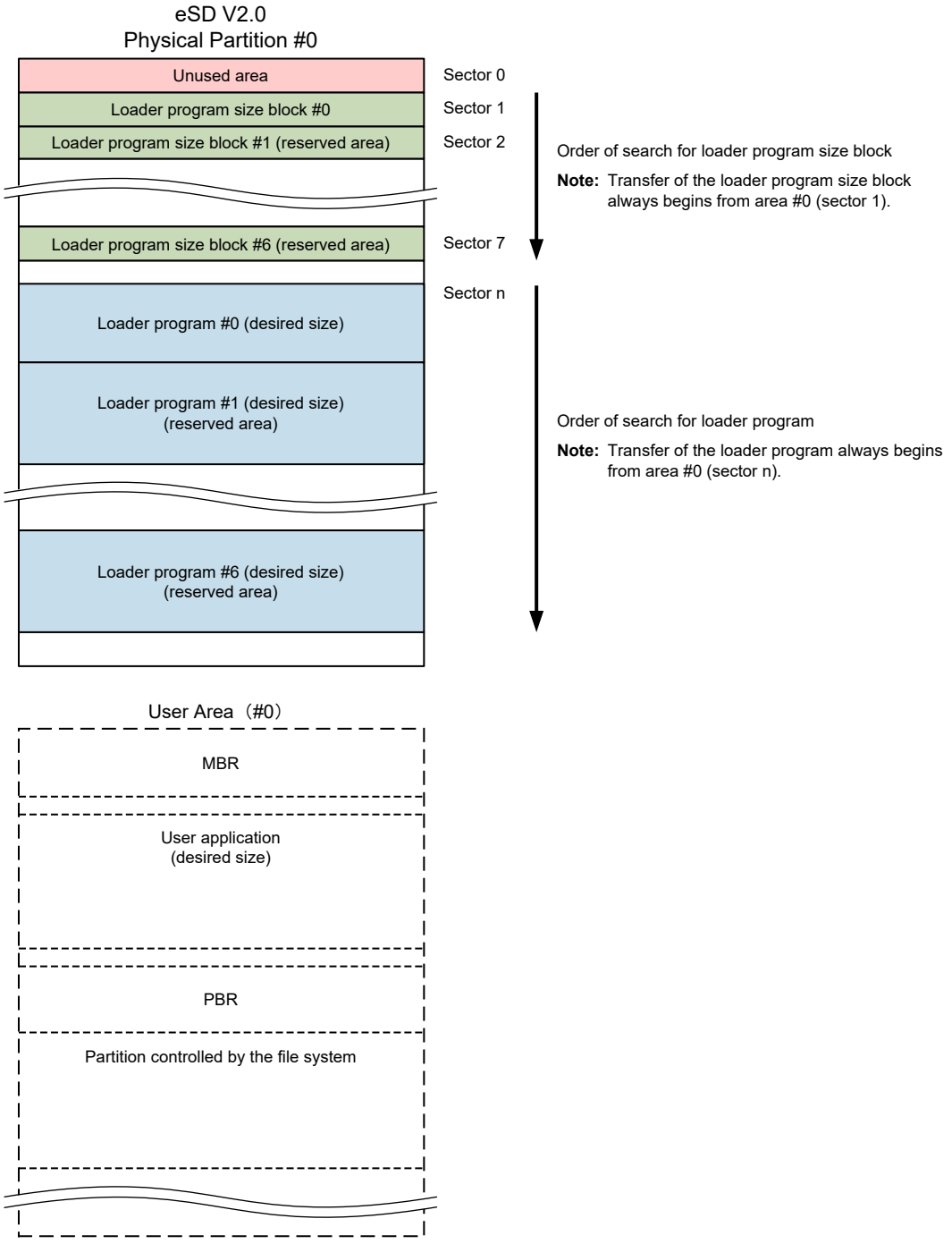
(1) The loader program size block is transferred from sector 1 in physical partition #0 of the eSD device to the SRAM.
If a communication error occurs during data transfer, read access is retried up to three times per area. If the retry processing is repeated three times during the transfer of the loader program size block in a single area, another loader program size block is transferred from the next multiplexed area.

(2) The signature in the transferred loader program size block is checked. When the signature matches the expected value (AA55h), the size of the loader program data is obtained from the loader program size block and the processing for transferring the loader program is executed.
If the signature does not match the expected value, another loader program size block is transferred from the next multiplexed area.

(3) The transfer of the loader program begins from the loader program load address in physical partition #0 of the eSD device. The loader program is transferred to the SRAM for the obtained loader program size.
If a communication error occurs during data transfer, read access is retried up to three times per area. If the retry processing is repeated three times during the loader program transfer in a single area, another loader program is transferred from the next multiplexed area.

NOTE

Even when the loader size block is transferred from a multiplexed (reserved) area, the transfer of the loader program always begins from the area for loader program #0.



Note 1. Write the same data to loader program size block areas #0 to #6.

Note 2. Write the same data to loader program areas #0 to #6.

Note 3. The start sector location of the loader program is calculated as follows.
Loader program load address + obtained size (number of sectors) × n

Figure 1.9-12 Allocation of Loader Programs in the eSD Device Conforming to the eSD V2.0 Standard

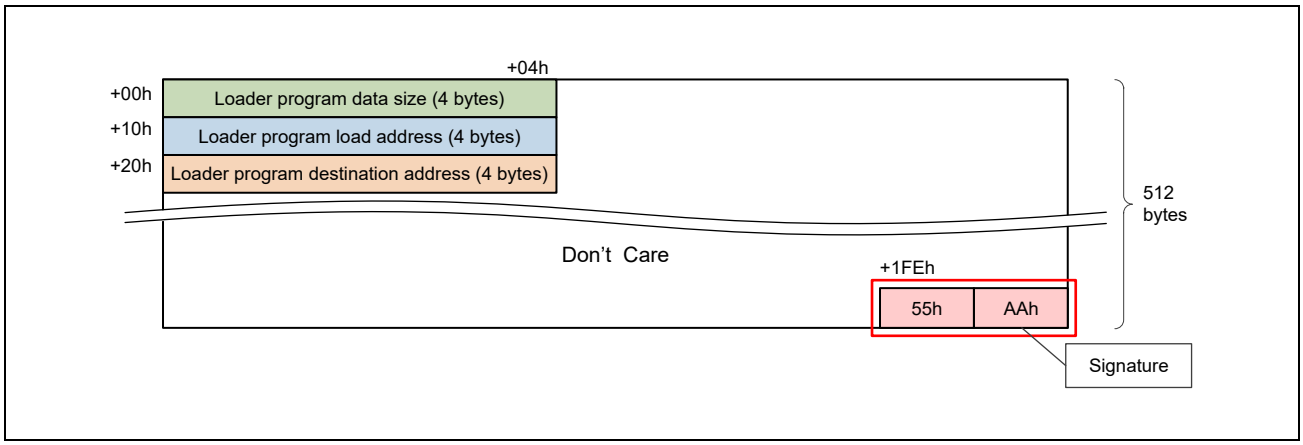


Figure 1.9-13 Structure of the Loader Program Size Block

NOTE

Store the loader program data size in 4-byte little endian.

(2) Allocation in eSD V2.1 (Multi Partitions)

Figure 1.9-14 shows the allocation of the loader program size blocks and loader programs in the eSD device conforming to the eSD V2.1 standard. To prevent read-disturb errors, up to seven loader program size blocks and loader programs can be multiplexed and written to the eSD device

The specifications and allocation of loader program size blocks are the same as those in eSD V2.0. For the structure of the loader program size block, see **Figure 1.9-13**.

The following describes the operation of transferring the loader program.

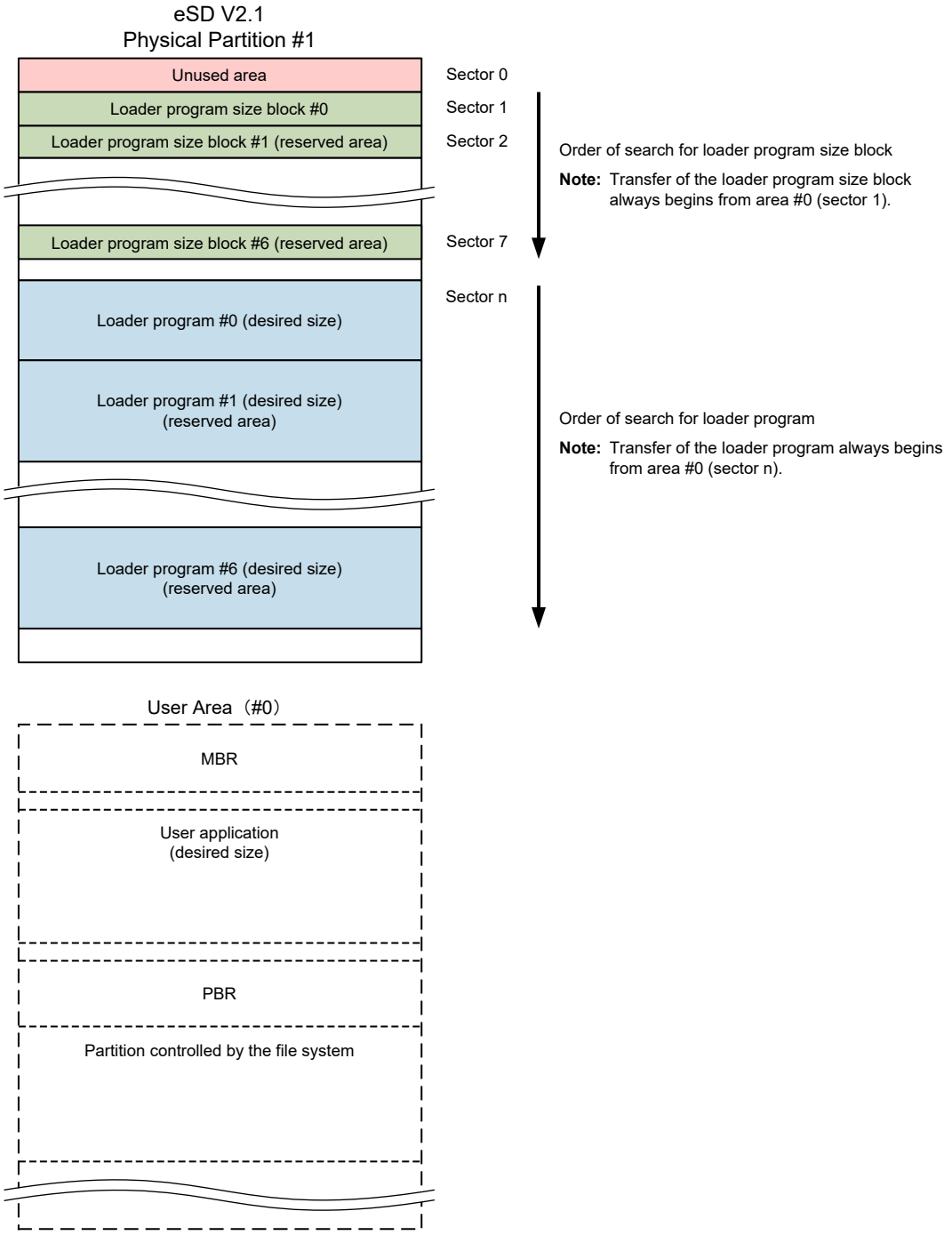
(1) The loader program size block is transferred from sector 1 in physical partition #1 of the eSD device to the SRAM.
If a communication error occurs during data transfer, read access is retried up to three times per area. If the retry processing is repeated three times during the transfer of the loader program size block in a single area, another loader program size block is transferred from the next multiplexed area.

(2) The signature in the transferred loader program size block is checked. When the signature matches the expected value (AA55h), the size of the loader program data is obtained from the loader program size block and the processing for transferring the loader program is executed. If the signature does not match the expected value, another loader program size block is transferred from the next multiplexed area.

(3) The transfer of the loader program begins from the loader program load address in physical partition #1 of the eSD device. The loader program is transferred to the SRAM for the obtained loader program size.
If a communication error occurs during data transfer, read access is retried up to three times per area. If the retry processing is repeated three times during the loader program transfer in a single area, another loader program is transferred from the next multiplexed area.

NOTE

Even when the loader size block is transferred from a multiplexed (reserved) area, the transfer of the loader program always begins from the area for loader program #0.



Note 1. Write the same data to loader program size block areas #0 to #6.

Note 2. Write the same data to loader program areas #0 to #6.

Note 3. The start sector location of the loader program is calculated as follows.
Loader program load address + obtained size (number of sectors) × n

Figure 1.9-14 Allocation of Loader Programs in the eSD Device Conforming to the eSD V2.1 Standard

(3) How to Distinguish between eSD V2.0 (Single Partition) and V2.1 (Multi Partitions)

This boot program first assumes that an eSD device supporting multi partitions is connected and begins loader program transfer from physical partition #1. If an error regarding a command for multi partitions shown in the following table occurs, the boot program assumes that an eSD device with a single partition is connected and transfers the loader program from physical partition #0.

Table 1.9-10 Errors Regarding Commands for Multi Partitions

No.	Issued Command	Command Function	Error	Remarks
1	CMD43 (SELECT_PARTITION)	Selects physical partition #1.	Multi partitions are not supported.	
2	CMD45 (QUERY_PARTITION)	Obtains the physical partition ID.	The specified partition does not exist.	

1.9.1.2.2 Boot Mode 1 / Boot Mode 5 (eMMC 3.3V / 1.8V)

Figure 1.9-15 shows the allocation of the loader program in the eMMC device conforming to the MMCA 4.4 standard.

In the loader program size block, the loader program size is stored in the first four bytes and the signature AA55h is stored in the last two bytes. **Figure 1.9-16** shows the structure of the loader program size data. Be sure to store the loader program size in 4-byte little endian.

The transfer of the loader program begins from the first address (sector 0) of the boot partition. A dummy read from sector 0 is done, and then the size of the loader program data is obtained from the data read from sector 1.

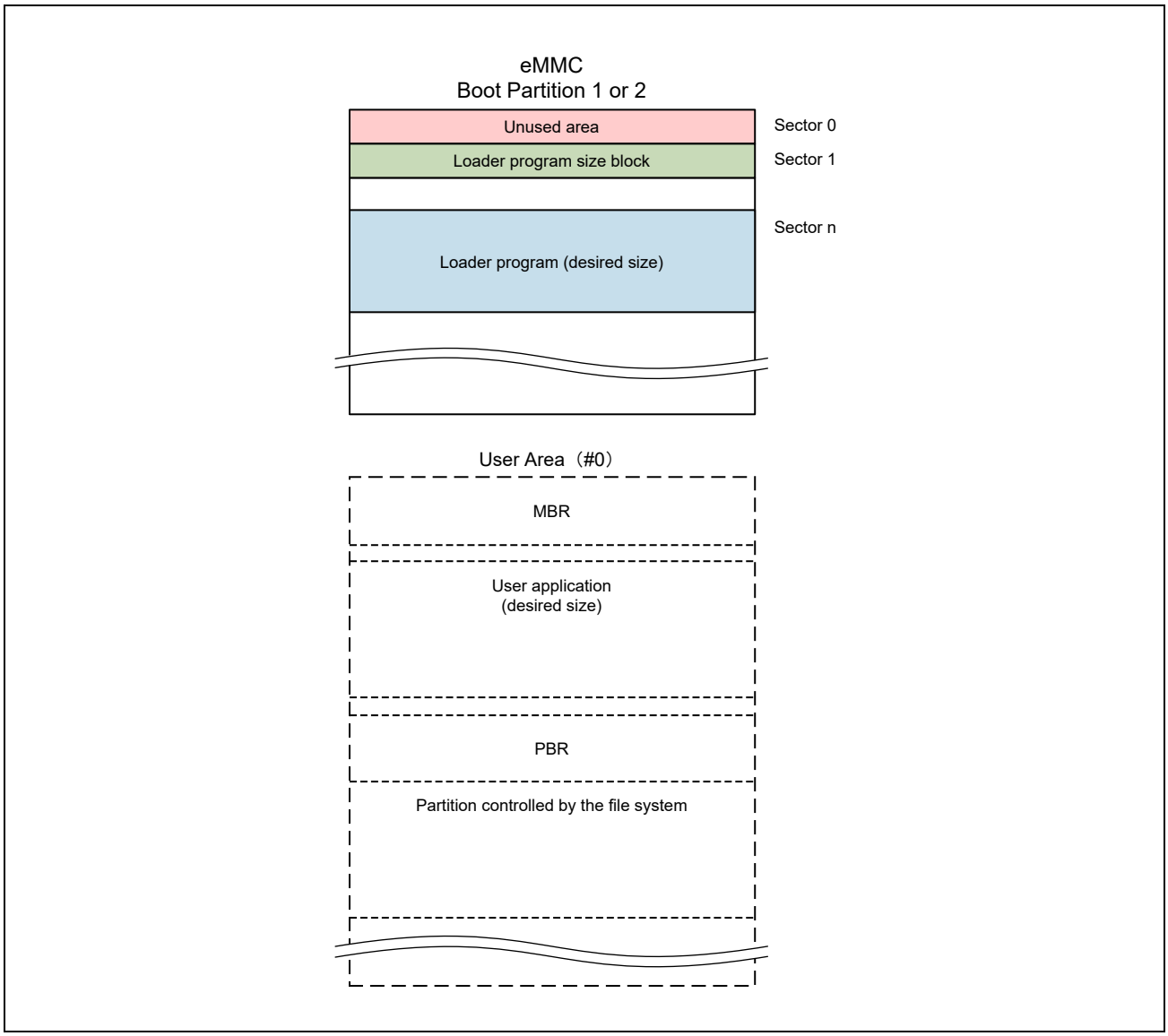


Figure 1.9-15 Allocation of Loader Program in the eMMC Device

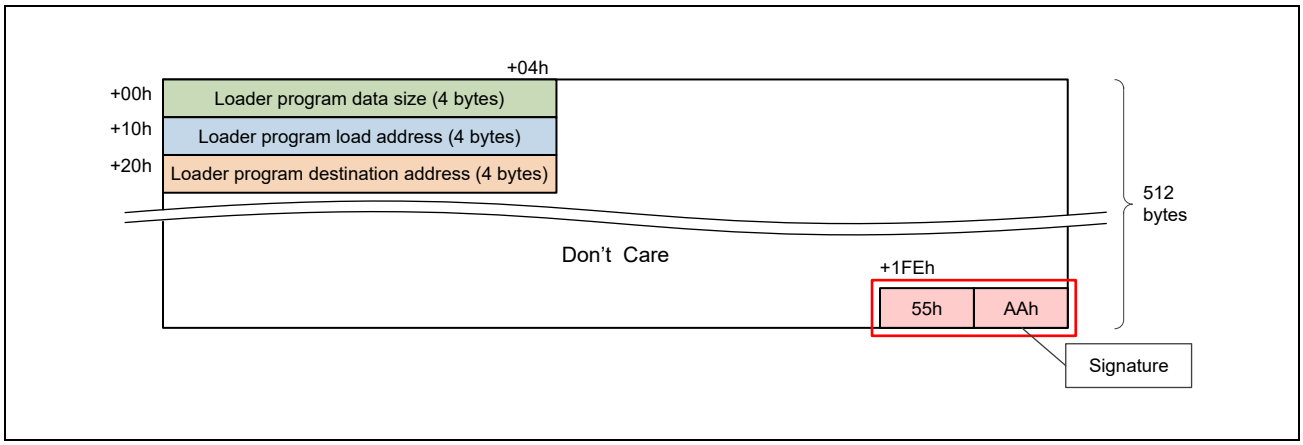


Figure 1.9-16 Structure of the Loader Program Size Block

In the eMMC device, the loader program is read from the partition specified by the BOOT_PARTITION_ENABLE bits in the PARTITION_CONFIG (ECSD[179]) field of the extended CSD register. Write the loader program to the partition selected by this register. **Table 1.9-11** shows the specifications of the extended CSD register.

Table 1.9-11 Specifications of the Extended CSD Register

No.	Field Name	Bit Name	Setting	Remarks
1	PARTITION_CONFIG (ECSD[179])	BOOT_PARTITION_ENABLE bits [5:3]	0h: Device not enabled for boot (default) 1h: Boot partition 1 enabled for boot 2h: Boot partition 2 enabled for boot 7h: User area enabled for boot	

Allocate partitions as follows.

- Store the loader program in boot partition 1.
- Store the application program in boot partition 2.
- Allocate the user area to a partition controlled by the file system.

1.9.1.2.3 Boot Mode 2 / Boot Mode 6 (Serial Flash Memory 3.3V / 1.8V)

Figure 1.9-17 shows the allocation of the loader program size block and loader program in the serial flash memory.

In the loader program size block, the loader program size is stored in the first four bytes and the signature AA55h is stored in the last two bytes. **Figure 1.9-18** shows the structure of the loader program size data. Be sure to store the loader program size in 4-byte little endian.

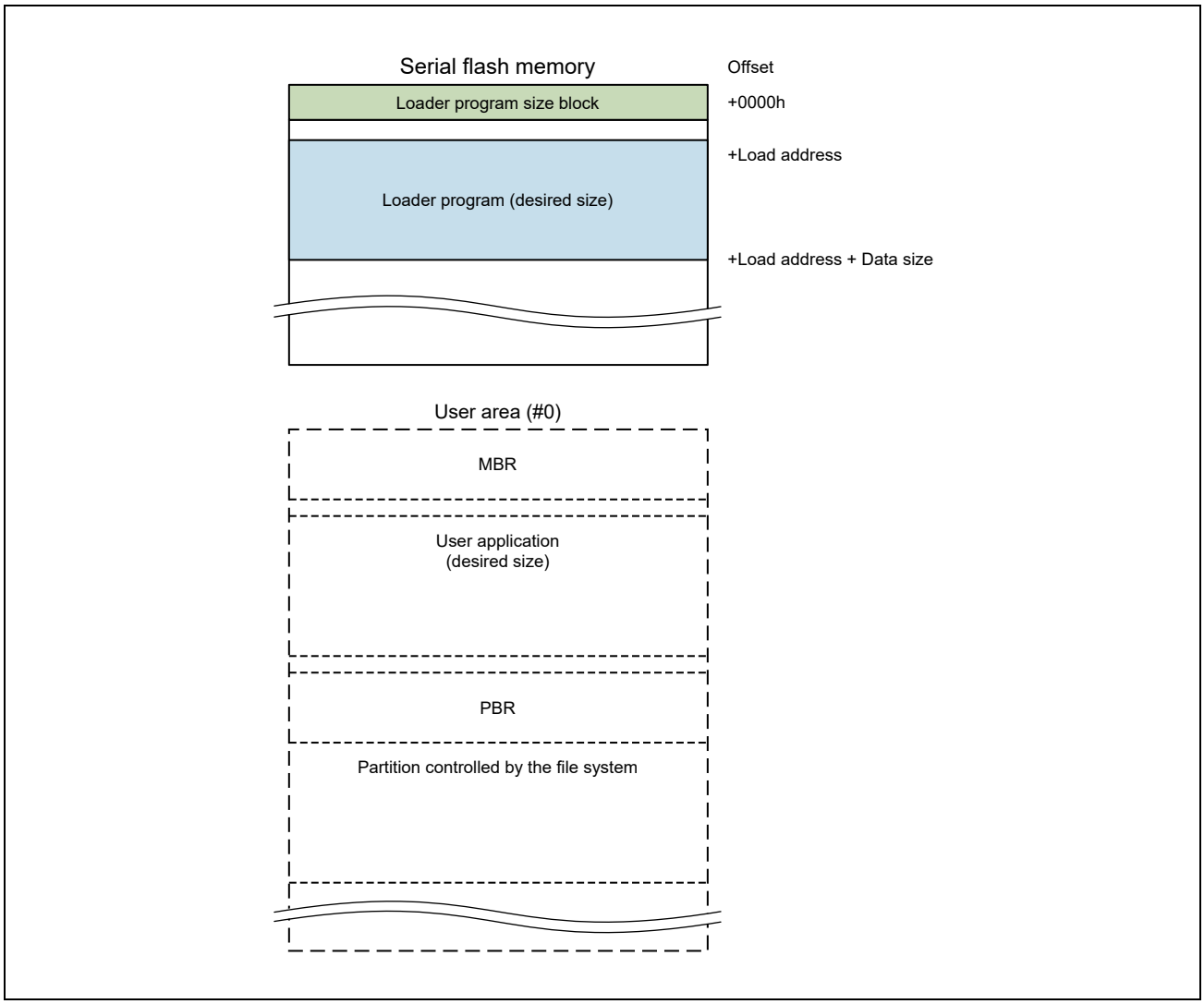


Figure 1.9-17 Allocation of Loader Program in the Serial Flash Memory Device

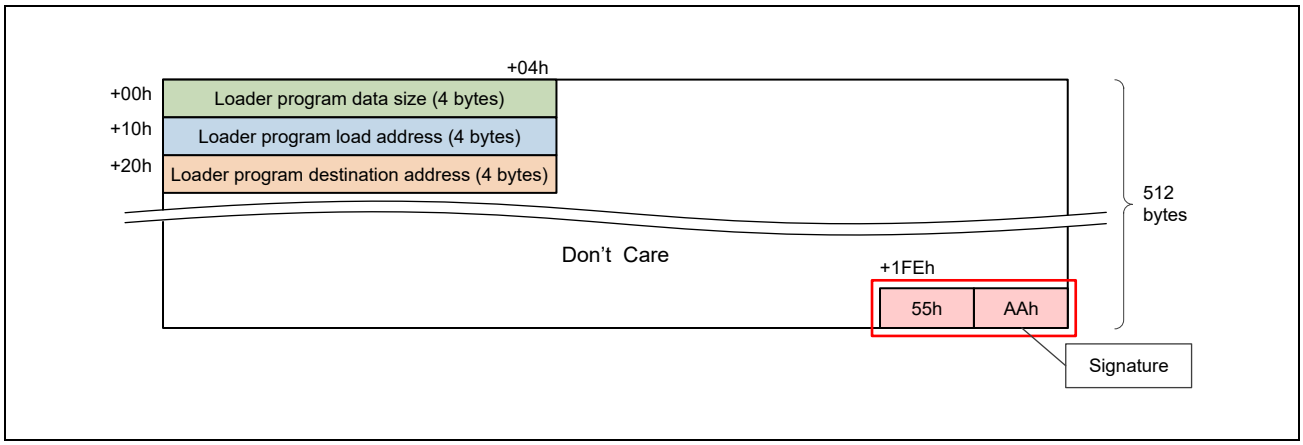


Figure 1.9-18 Structure of the Loader Program Size Block

1.9.1.2.4 Boot Mode 3 (SCIF Download)

In the loader program size block (512 bytes), the loader program size is stored in the first four bytes and the signature AA55h is stored in the last two bytes. Be sure to store the loader program size in 4-byte little endian.

The offset addresses in the SRAM for storing the file (the loader program) downloaded in this mode are as follows:

- CA55 cold boot: 0_0810_2000h
- CM33 cold boot: 0_2800_2000h

1.9.2 Boundary Scan Mode

Table 1.9-12 shows the pin that is related to the boundary scan.

Table 1.9-12 Boundary Scan Mode Pin

Pin Name	Function	Description
BSCANP	Boundary scan setting	0: Normal operation 1: Inter-LSI connection test on the user board (in the board)

The boundary scan mode is used to conduct connection tests between other ICs that connect to the external pins of this LSI, and it is enabled when BSCANP is high.

Five pins of the JTAG are used to control when this mode is enabled. This LSI has functionality to limit access for increased security when accessing from a debugger via the JTAG, however, no limitation is applied in the boundary scan mode.

1.9.3 Debug Mode

The LSI has pins for applying control that is different from that for the normal mode when debugging is to proceed. **Table 1.9-13** lists the relationships between the mode and the state of MD_BOOT3.

Table 1.9-13 Debug-Related Pins

MD_BOOT3	Mode	Overview of Operation
0	Normal operation	Normal operation mode
1	Debug mode	A mode for making a debugger connection and debugging the functions of the LSI.

Note 1. The level on the pin is captured by SYS on the rising edge of PRST_N. Setting of MD_BOOT3 to 1 is required to be in advance on the rising edge of PRST_N even if the JTAG connection is made in the midst of execution after booting up.

SECTION 2 PROCESSORS

2.1 Processors Overview

This LSI is equipped with 2 CPUs and 1 AI Accelerator.

- Cortex®-A55 Quad (CA55)
- Cortex®-M33 (CM33)
- Arm®Ethos™-U55 (NPU)

The CA55 is intended for application processing, the CM33 for system control.

For details on each CPU, see **2.2 CPU**.

In addition, this LSI is configured with the Ethos™-U55 (NPU). For details, see **2.3 Ethos-U55 (NPU)**.

SECTION 2 PROCESSORS

2.2 CPU

2.2.1 Overview

This LSI has following CPUs.

- Quad 64-bit Arm® Cortex®-A55 processors (Application processing)
- 32-bit Arm® Cortex®-M33 processor (System management)

2.2.2 Application Processors Cortex-A55 (CA55)

The CA55 is a core block equipped with quad Cortex-A55 cores. For details on the functions including interrupts of the CA55, see the Arm® Cortex®-A55 Core Technical Reference Manual and related documents in **2.2.2.5 Function Reference**. For details of interrupt type and control, see **4.6 Interrupt Controller**.

2.2.2.1 Features

The CA55 incorporates an extensively redesigned microarchitecture system that improves performance across the board while being very competitive in area and power efficiency.

Table 2.2-1 Function Summary

	Description
CPU	• Cortex-A55 r2p0 (ARMv8-A) quad cores
Cache memory	• L1$ (I/D) = 32 KB (Parity) / 32 KB (ECC) • L2$ = None • L3$ = 1024 KB (ECC)
Maximum operating frequency	• CPU — Core: Over Drive 1800 MHz*1 Normal Drive 1100 MHz*1 — Cluster: Over Drive 1260 MHz*1 Normal Drive 1100 MHz*1
Interrupt controller	• External interrupt controller*2

Note 1. For details of Over Drive, refer to **10.1 Electrical Characteristics**.

Note 2. Interrupts generated in the system of this LSI are conveyed to GIC-600 via the ICU. For the various settings and setting procedures for the GIC, refer to the *GIC-600 Generic Interrupt Controller Technical Reference Manual* and **4.6 Interrupt Controller**.

Table 2.2-2 Core Configuration

	Value	Description
L1 instruction cache size	32 KB	L1 instruction cache = 32 KB (Parity)
L1 data cache size	32 KB	L1 data cache = 32 KB (ECC)
L3 cache size	1024 KB	L3 cache = 1024 KB (ECC)
ECC or parity core cache protection	Included	Support for core cache ECC
Advanced SIMD and floating-point support (including dot product instruction support)	Included	Support for Neon™ FPU engine
Cryptographic extension	Included (optional)	Support for cryptography engine
AArch32	N/A	Not supported
Big endian	N/A	Not supported

Table 2.2-3 Cluster Configuration

Item	Initial Value	Description	Switching Timing
AA64nAA32[0]	1b	For core 0. Fixed to 1 (AArch64).	—
AA64nAA32[1]	1b	For core 1. Fixed to 1 (AArch64).	—
AA64nAA32[2]	1b	For core 2. Fixed to 1 (AArch64).	—
AA64nAA32[3]	1b	For core 3. Fixed to 1 (AArch64).	—
CFGEND[0]	0b	For core 0. Fixed to 0 (little endian).	—
CFGEND[1]	0b	For core 1. Fixed to 0 (little endian).	—
CFGEND[2]	0b	For core 2. Fixed to 0 (little endian).	—
CFGEND[3]	0b	For core 3. Fixed to 0 (little endian).	—
CLUSTERIDAFF2[7:0]	0h	For core 0. The value is fixed.	—
CLUSTERIDAFF3[7:0]	0h	For core 1. The value is fixed.	—
CRYPTODISABLE	—	Fixed to 1 and 0 respectively for secure and normal products.	—
RVBARADDR0[39:2]	OTP setting	For core 0. The value can be changed.	Cold, cluster, core 0 reset
RVBARADDR1[39:2]	00_0200_0000h	For core 1. The value can be changed. The default is the start address of SRAM0 (corresponding to 00_0800_0000h).	Cold, cluster, core 1 reset
RVBARADDR2[39:2]	00_0200_0000h	For core 2. The value can be changed. The default is the start address of SRAM0 (corresponding to 00_0800_0000h).	Cold, cluster, core 2 reset
RVBARADDR3[39:2]	00_0200_0000h	For core 3. The value can be changed. The default is the start address of SRAM0 (corresponding to 00_0800_0000h).	Cold, cluster, core 3 reset
GICCDISABLE	0b	Fixed to 0 (CPU interface enabled)	—
ASTARTMP[39:20]	0_0100h	Start address of PP IF*1. The value can be changed.	Cold, cluster reset
AENDMP[39:20]	0_01FFh	End address of PP IF*1. The value can be changed.	Cold, cluster reset

Note 1. PP IF stands for Peripheral Port Interface.
For details, see the Arm® DynamIQ™ Shared Unit Technical Reference Manual and related documents.

The settings by the configuration pins are reflected when a reset is applied to the target core. For the types of reset and the areas to be reset, see **Table 2.2-4**. The configuration pins are RVBARADDR0/1/2/3 for specifying the base addresses and ASTARTMP and AENDMP for specifying the areas of PP IF.

The procedure for changing RVBARADDR0/1/2/3 is given below.

1. Transfer the program to the changed base address.
2. Write the base address to the SYS_ACPU_CFG_RVAL0/1/2/3 and SYS_ACPU_CFG_RVAH0/1/2/3 registers.
3. Apply a reset for core 0/1/2/3.

The procedure for changing ASTARTMP and AENDMP is given below.

1. Write ASTARTMP to the SYS_ACPU_CFG_SMPH and SYS_ACPU_CFG_SMPL registers.
2. Write AENMP to the SYS_ACPU_CFG_EMPH and SYS_ACPU_CFG_EMPL registers.
3. Apply a reset for the cluster.

CAUTION

When ASTARTMP or AENDMP is changed, note the points below.

- When AENDMP <= ASTARTMP, the PP IF is not used.
- When ASTARTMP < AENDMP, the PP IF is used. The lower limit address of the peripheral area includes the setting value of ASTARTMP; the upper limit address of the peripheral area does not include the setting value of AENDMP. The upper limit address is set by subtracting 1 byte from AENDMP.

2.2.2.2 Resets

Table 2.2-4 shows the supported types of reset and the areas that are reset by the respective types. For the procedures of handling the individual resets, see **2.2.2.2.1 Cold Reset** and the subsequent parts. See the Arm® DynamIQ™ Shared Unit Technical Reference Manual and Arm® Cortex®-A55 Core Technical Reference Manual.

Table 2.2-4 Areas to be Reset

Reset Function	Areas to be Reset
Cold reset	Entire area
Cluster warm reset	Entire area of the cluster + core 0 (and cores 1, 2, and 3 in quad-core devices) except for the debug, RAS, and ETM registers
Core warm reset	Entire area of the target core except for the debug, RAS, and ETM registers
Debug reset	Debug block

The P-Channels can be used to control safety in resetting. For details on the P-Channels, see the AMBA® Low Power Interface Specification Arm® Q-Channel and P-Channel Interfaces. For details on registers for controlling the P-Channel and reset pins, see **4.4 Clock Pulse Generator (CPG)**. **Table 2.2-5** shows the supported power modes for each P-Channel. Transitions to power modes other than those listed in the table below are not supported.

Table 2.2-5 Supported Power Modes

PSTATE	Power Mode
CLUSTERPSTATE[6:0]	• 100_1000b (ON) • 000_0000b (OFF)
COREPSTATE0[5:0]	• 00_1000b (ON) • 00_0001b (OFF_EMU) • 00_0000b (OFF)
COREPSTATE1[5:0]	• 00_1000b (ON) • 00_0001b (OFF_EMU) • 00_0000b (OFF)
COREPSTATE2[5:0]*1	• 00_1000b (ON) • 00_0001b (OFF_EMU) • 00_0000b (OFF)
COREPSTATE3[5:0]*1	• 00_1000b (ON) • 00_0001b (OFF_EMU) • 00_0000b (OFF)

Note 1. Only OFF mode is supported in the Dual Core products.

2.2.2.2.1 Cold Reset

A cold reset is applied at any of the following cases.

(a) The sequence of a power-on reset by the CPG is executed (in CA55 boot mode).
(b) The WDT counter has underflowed.
(c) Software control by the CM33 is applied.

For details on (a) and (b) reset, see **4.4 Clock Pulse Generator (CPG)** and **5.4 Watchdog Timer (WDT)**, respectively. To apply a cold reset by case (c) reset, follow the procedure below. The CPU column shows the CPU which executes processing. At the time of reading in the Processing column, continue polling until the expected value is obtained.

Table 2.2-6 Procedure for Applying Cold Reset

Step	CPU	Register Name	Processing	Value
1	CA55	Execute a power-down sequence described in the Technical Reference Manual (TRM) on all powered-on cores.		
2	CM33	CPG_LP_CA55_CTL6	READ	****_****_****_****_****_****_****_0000b
3	CM33	CPG_LP_CA55_CTL2	WRITE	****_****_*000_0001_****_****_*000_0001b
		CPG_LP_CA55_CTL3	WRITE	****_****_*000_0001_****_****_*000_0001b
4	CM33	CPG_LP_CA55_CTL2	READ	****_***1_0000_0001_****_***1_0000_0001b
		CPG_LP_CA55_CTL3	READ	****_***1_0000_0001_****_***1_0000_0001b
5	CM33	CPG_LP_CA55_CTL2	WRITE	****_****_*000_0000_****_****_*000_0000b
		CPG_LP_CA55_CTL3	WRITE	****_****_*000_0000_****_****_*000_0000b
6	CM33	CPG_LP_CA55_CTL2	READ	****_***0_0000_0000_****_***0_0000_0000b
		CPG_LP_CA55_CTL3	READ	****_***0_0000_0000_****_***0_0000_0000b
7	CM33	CPG_LP_CA55_CTL1	WRITE	****_****_****_****_****_****_****_0000_0001b
8	CM33	CPG_LP_CA55_CTL1	READ	****_****_****_****_****_****_**10_0000_0001b
9	CM33	CPG_LP_CA55_CTL1	WRITE	****_****_****_****_****_****_****_0000_0000b
10	CM33	CPG_LP_CA55_CTL1	READ	****_****_****_****_****_****_**00_0000_0000b
11	Other	CPG_RST_0	WRITE	**11_1111_1111_1111_**00_0000_0000_0000b
12	Other	CPG_RST_1	WRITE	****_****_****_*111_****_****_****_*000b
13	Other	CPG_RSTMON_0	READ	****_****_****_***1_1111_1111_1111_1111b

Note: Processing from step 1 to 10 is a recommended procedure to minimize the impact on the system.

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

The control procedure for releasing the cold reset by (c) to start core 0 is as follows.

Table 2.2-7 Procedure for Releasing Cold Reset to Start Core 0

Step	CPU	Register Name	Processing	Value
1	Other	CPG_RSTMON_0	READ	****_****_****_***1_1111_1111_1111_1111b
2	Other	CPG_LP_CA55_CTL1	WRITE	****_****_****_****_****_****_1001_0001b
3	Other	CPG_LP_CA55_CTL2	WRITE	****_****_****_****_****_****_*001_0001b
4	Other	CPG_RST_1	WRITE	****_****_****_*111_****_****_****_*111b
5	Other	CPG_RSTMON_0	READ	****_****_****_***0_0011_1111_1111_1111b
6	Other	CPG_RST_0	WRITE	**11_1111_1111_1111_**11_1111_1111_1111b
7	Other	CPG_RSTMON_0	READ	****_****_****_***0_0000_0000_0000_0000b
8	Other	CPG_LP_CA55_CTL1	READ	****_****_****_****_****_**10_1001_0001b
9	Other	CPG_LP_CA55_CTL2	READ	****_****_****_****_****_***1_0001_0001b
10	Other	CPG_LP_CA55_CTL1	WRITE	****_****_****_****_****_**10_1001_0000b
11	Other	CPG_LP_CA55_CTL1	READ	****_****_****_****_****_**00_1001_0000b
12	Other	CPG_LP_CA55_CTL2	WRITE	****_****_****_****_****_***1_0001_0000b
13	Other	CPG_LP_CA55_CTL2	READ	****_****_****_****_****_***0_0001_0000b

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

The procedure for starting another core (core x) after the startup of core 0 is as follows.

Table 2.2-8 Procedure for Starting Core x

Step	CPU	Processing
1	Other	Release the cold reset to start the cluster according to **Table 2.2-7**.
2	Any	Set the COREPREQx bit of CPG_LP_CA55_CTLn to 1b and the COREPSTATEx bits to 00_1000b.
3	Any	Check that the COREPDENYx bit of CPG_LP_CA55_CTLn is 0b and the COREPACCEPTx bit is 1b.
4	Any	Set the COREPREQx bit of CPG_LP_CA55_CTLn to 0b.
5	Any	Check that the COREPDENYx bit of CPG_LP_CA55_CTLn is 0b and the COREPACCEPTx bit is 0b.

Note: CPG_LP_CA55_CTLn: CPG_LP_CA55_CTL2 for core 0/1; CPG_LP_CA55_CTL3 for core 2/3.

2.2.2.2.2 Core Warm Reset

There are the two types of core warm reset: internal and external resets. The former requires software control by the CA55 itself and the latter requires software control by the CA55 and CM33. Though the CA55 handles an internal reset in the same way as an external reset, if you intend to reset a single core, using an internal reset is recommended because handling this requires fewer steps than an external reset. An external reset is mainly used as part of a cluster warm reset, which is described in **2.2.2.2.3 Cluster Warm Reset**.

To apply an internal reset, follow the procedure below.

Table 2.2-9 Core Warm Reset (Internal)

Step	CPU	Register Name	Processing	Value
1	CA55	RMR_EL3.RR	Write	1b
2	CA55	—	WFI	—

Release from an internal reset is automatic following application of the reset. No specific processing is required. For details, see the Arm® DynamIQ™ Shared Unit Technical Reference Manual and Arm® Cortex®-A55 Core Technical Reference Manual.

Application of the reset requires transition of the core power mode to OFF or EMU_OFF by the P-channel pin. The following describes the reset sequence for each core using the sequence for setting the power mode to OFF. Read the power mode as EMU_OFF when placing each core into this state.

The control procedure for applying an external reset to core 0 is as follows.

Table 2.2-10 Core 0 Warm Reset (External Reset Procedure)

Step	CPU	Register Name	Processing	Value
1	CA55	Execute a power-down sequence*1 on core 0.		
2	Any	CPG_LP_CA55_CTL6	READ	****_****_****_****_****_****_****_***0b
3	Any	CPG_LP_CA55_CTL2	WRITE	****_****_****_****_****_****_****_*000_0001b
4	Any	CPG_LP_CA55_CTL2	READ	****_****_****_****_****_****_***1_0000_0001b
5	Any	CPG_LP_CA55_CTL2	WRITE	****_****_****_****_****_****_****_*000_0000b
6	Any	CPG_LP_CA55_CTL2	READ	****_****_****_****_****_****_***0_0000_0000b
7	Any	CPG_RST_0	WRITE	****_****_***1_****_****_****_***0_****b
8	Any	CPG_RSTMON_0	READ	****_****_****_****_****_****_****_***1_****b

Note: Steps 1 and 2 are a recommended procedure to minimize the impact on the system.

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

Note 1. For details, refer to the Arm® Cortex®-A55 Core Technical Reference Manual.

The control procedure for releasing the core warm reset to start core 0 is as follows.

Table 2.2-11 Core 0 Warm Reset (External Reset Release Procedure)

Step	CPU	Register Name	Processing	Value
1	Any	CPG_LP_CA55_CTL2	WRITE	****_****_****_****_****_****_*001_0001b
2	Any	CPG_RST_0	WRITE	***1_1111_1111_1111_**11_1111_1111_1111b
3	Any	CPG_RSTMON_0	READ	****_****_****_***0_0000_0000_0000_0000b
4	Any	CPG_LP_CA55_CTL2	READ	****_****_****_****_****_***1_0001_0001b
5	Any	CPG_LP_CA55_CTL2	WRITE	****_****_****_****_****_****_*001_0000b
6	Any	CPG_LP_CA55_CTL2	READ	****_****_****_****_****_***0_0001_0000b

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

The control procedure for applying an external reset to core 1 is as follows.

Table 2.2-12 Core 1 Warm Reset (External Reset Procedure)

Step	CPU	Register Name	Processing	Value
1	CA55	Execute a power-down sequence*1 on core 1.		
2	Any	CPG_LP_CA55_CTL6	READ	****_****_****_****_****_****_****_**0*b
3	Any	CPG_LP_CA55_CTL2	WRITE	****_****_*000_0001_****_****_****_****b
4	Any	CPG_LP_CA55_CTL2	READ	****_***1_0000_0001_****_****_****_****b
5	Any	CPG_LP_CA55_CTL2	WRITE	****_****_*000_0000_****_****_****_****b
6	Any	CPG_LP_CA55_CTL2	READ	****_***0_0000_0000_****_****_****_****b
7	Any	CPG_RST_0	WRITE	****_****_**1*_****_****_****_**0*_****b
8	Any	CPG_RSTMON_0	READ	****_****_****_****_****_****_**1*_****b

Note: Steps 1 and 2 are a recommended procedure to minimize the impact on the system.

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

Note 1. For details, refer to the Arm® Cortex®-A55 Core Technical Reference Manual.

The control procedure for releasing the core warm reset to start core 1 is as follows.

Table 2.2-13 Core 1 Warm Reset (External Reset Release Procedure)

Step	CPU	Register Name	Processing	Value
1	Any	CPG_LP_CA55_CTL2	WRITE	****_****_*001_0001_****_****_****_****b
2	Any	CPG_RST_0	WRITE	***1_1111_1111_1111_**11_1111_1111_1111b
3	Any	CPG_RSTMON_0	READ	****_****_****_***0_0000_0000_0000_0000b
4	Any	CPG_LP_CA55_CTL2	READ	****_***1_0001_0001_****_****_****_****b
5	Any	CPG_LP_CA55_CTL2	WRITE	****_****_*001_0000_****_****_****_****b
6	Any	CPG_LP_CA55_CTL2	READ	****_***0_0001_0000_****_****_****_****b

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

The control procedure for applying an external reset to core 2 is as follows.

This procedure cannot be performed on dual-core products. This procedure is only supported for Quad core products.

Table 2.2-14 Core 2 Warm Reset (External Reset Procedure)

Step	CPU	Register Name	Processing	Value
1	CA55	Execute a power-down sequence*1 on core 2.		
2	Any	CPG_LP_CA55_CTL6	READ	****_****_****_****_****_****_****_*0**b
3	Any	CPG_LP_CA55_CTL3	WRITE	****_****_****_****_****_****_*000_0001b
4	Any	CPG_LP_CA55_CTL3	READ	****_****_****_****_****_****_***1_0000_0001b
5	Any	CPG_LP_CA55_CTL3	WRITE	****_****_****_****_****_****_*000_0000b
6	Any	CPG_LP_CA55_CTL3	READ	****_****_****_****_****_****_***0_0000_0000b
7	Any	CPG_RST_0	WRITE	****_****_*1**_****_****_****_*0**_****b
8	Any	CPG_RSTMON_0	READ	****_****_****_****_****_****_*1**_****b

Note: Steps 1 and 2 are a recommended procedure to minimize the impact on the system.

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

Note 1. For details, refer to the Arm® Cortex®-A55 Core Technical Reference Manual.

The control procedure for releasing the core warm reset to start core 2 is as follows.

This procedure cannot be performed on dual-core products. This procedure is only supported for Quad core products.

Table 2.2-15 Core 2 Warm Reset (External Reset Release Procedure)

Step	CPU	Register Name	Processing	Value
1	Any	CPG_LP_CA55_CTL3	WRITE	****_****_****_****_****_****_*001_0001b
2	Any	CPG_RST_0	WRITE	***1_1111_1111_1111_**11_1111_1111_1111b
3	Any	CPG_RSTMON_0	READ	****_****_****_***0_0000_0000_0000_0000b
4	Any	CPG_LP_CA55_CTL3	READ	****_****_****_****_****_****_***1_0001_0001b
5	Any	CPG_LP_CA55_CTL3	WRITE	****_****_****_****_****_****_*001_0000b
6	Any	CPG_LP_CA55_CTL3	READ	****_****_****_****_****_****_***0_0001_0000b

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

The control procedure for applying an external reset to core 3 is as follows.

This procedure cannot be performed on dual-core products. This procedure is only supported for Quad core products.

Table 2.2-16 Core 3 Warm Reset (External Reset Procedure)

Step	CPU	Register Name	Processing	Value
1	CA55	Execute a power-down sequence*1 on core 3.		
2	Any	CPG_LP_CA55_CTL6	READ	****_****_****_****_****_****_****_0***b
3	Any	CPG_LP_CA55_CTL3	WRITE	****_****_*000_0001_****_****_****_****b
4	Any	CPG_LP_CA55_CTL3	READ	****_***1_0000_0001_****_****_****_****b
5	Any	CPG_LP_CA55_CTL3	WRITE	****_****_*000_0000_****_****_****_****b
6	Any	CPG_LP_CA55_CTL3	READ	****_***0_0000_0000_****_****_****_****b
7	Any	CPG_RST_0	WRITE	****_****_1***_****_****_****_0***_****b
8	Any	CPG_RSTMON_0	READ	****_****_****_****_****_****_1***_****b

Note: Steps 1 and 2 are a recommended procedure to minimize the impact on the system.

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

Note 1. For details, refer to the Arm® Cortex®-A55 Core Technical Reference Manual.

The control procedure for releasing the core warm reset to start core 3 is as follows.

This procedure cannot be performed on dual-core products. This procedure is only supported for Quad core products.

Table 2.2-17 Core 3 Warm Reset (External Reset Release Procedure)

Step	CPU	Register Name	Processing	Value
1	Any	CPG_LP_CA55_CTL2	WRITE	****_****_*001_0001_****_****_****_****b
2	Any	CPG_RST_0	WRITE	***1_1111_1111_1111_**11_1111_1111_1111b
3	Any	CPG_RSTMON_0	READ	****_****_****_***0_0000_0000_0000_0000b
4	Any	CPG_LP_CA55_CTL3	READ	****_***1_0001_0001_****_****_****_****b
5	Any	CPG_LP_CA55_CTL3	WRITE	****_****_*001_0000_****_****_****_****b
6	Any	CPG_LP_CA55_CTL3	READ	****_***0_0001_0000_****_****_****_****b

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

2.2.2.2.3 Cluster Warm Reset

There are the two types of cluster warm reset: normal and immediate resets. Both require software control by the CA55 and CM33. Though the CA55 handles an immediate reset in the same way as a normal reset, an immediate reset should only be used for debugging because handling this may cause a deadlock.

The control procedure for applying a normal reset while cores 0/1/2/3 are started up is as follows.

If there is any core that is not started up, do not perform steps 1 to 5 on this core.

Table 2.2-18 Cluster Warm Reset (Normal Reset Procedure)

Step	CPU	Register Name	Processing	Value
1	CA55	Execute a power-down sequence*1 on all powered-on cores 0/1/2/3.		
2	Other	CPG_LP_CA55_CTL6	READ	****_****_****_****_****_****_****_0000b
3	Other	CPG_LP_CA55_CTL2	WRITE	****_****_*000_0001_****_****_*000_0001b
		CPG_LP_CA55_CTL3	WRITE	****_****_*000_0001_****_****_*000_0001b
4	Other	CPG_LP_CA55_CTL2	READ	****_***1_0000_0001_****_***1_0000_0001b
		CPG_LP_CA55_CTL3	READ	****_***1_0000_0001_****_***1_0000_0001b
5	Other	CPG_LP_CA55_CTL2	WRITE	****_****_*000_0000_****_****_*000_0000b
		CPG_LP_CA55_CTL3	WRITE	****_****_*000_0000_****_****_*000_0000b
6	Other	CPG_LP_CA55_CTL2	READ	****_***0_0000_0000_****_***0_0000_0000b
		CPG_LP_CA55_CTL3	READ	****_***0_0000_0000_****_***0_0000_0000b
7	Other	CPG_LP_CA55_CTL1	WRITE	****_****_****_****_****_****_****_0000_0001b
8	Other	CPG_LP_CA55_CTL1	READ	****_****_****_****_****_****_**10_0000_0001b
9	Other	CPG_LP_CA55_CTL1	WRITE	****_****_****_****_****_****_****_0000_0000b
10	Other	CPG_LP_CA55_CTL1	READ	****_****_****_****_****_****_**00_0000_0000b
11	Other	CPG_RST_0	WRITE	**11_1111_1111_1111_**10_1101_0000_1111b
12	Other	CPG_RST_1	WRITE	****_****_****_*111_****_****_****_*001b
13	Other	CPG_RSTMON_0	READ	****_****_****_***1_1001_0010_1111_0000b

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

Note 1. For details, refer to the Arm® Cortex®-A55 Core Technical Reference Manual.

When applying an immediate reset, perform only steps 11 to 13 of the normal reset sequence.

The control procedure for releasing the cluster warm reset to start core 0 is as follows.

Table 2.2-19 Cluster Warm Reset (Normal Reset Release and Core 0 Startup Procedure)

Step	CPU	Register Name	Processing	Value
1	Other	CPG_LP_CA55_CTL1	WRITE	****_****_****_****_****_****_1001_0001b
2	Other	CPG_LP_CA55_CTL2	WRITE	****_****_****_****_****_****_*001_0001b
3	Other	CPG_RST_1	WRITE	****_****_****_*111_****_****_****_*111b
4	Other	CPG_RSTMON_0	READ	****_****_****_***0_0001_0010_1111_0000b
5	Other	CPG_RST_0	WRITE	**11_1111_1111_1111_**11_1111_1111_1111b
6	Other	CPG_RSTMON_0	READ	****_****_****_***0_0000_0000_0000_0000b
7	Other	CPG_LP_CA55_CTL1	READ	****_****_****_****_****_**10_1001_0001b
8	Other	CPG_LP_CA55_CTL2	READ	****_****_****_****_****_***1_0001_0001b
9	Other	CPG_LP_CA55_CTL1	WRITE	****_****_****_****_****_****_1001_0000b
10	Other	CPG_LP_CA55_CTL1	READ	****_****_****_****_****_**00_1001_0000b
11	Other	CPG_LP_CA55_CTL2	WRITE	****_****_****_****_****_****_*001_0000b
12	Other	CPG_LP_CA55_CTL2	READ	****_****_****_****_****_***0_0001_0000b

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

2.2.2.2.4 Debug Reset

Debug resets can only be applied under software control.

To apply a debug reset, follow the procedure below.

Table 2.2-20 Procedure for Applying Debug Reset

Step	CPU	Register Name	Processing	Value
1	Any	CPG_RST_1	WRITE	****_****_****_*001_****_****_****_*110b
2	Any	CPG_RSTMON_0	READ	****_****_****_***0_0010_0000_0000_0000b

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

To release from the debug reset state, follow the procedure below.

Table 2.2-21 Procedure for Releasing Debug Reset

Step	CPU	Register Name	Processing	Value
1	Any	CPG_RST_1	WRITE	****_****_****_*001_****_****_****_*111b
2	Any	CPG_RSTMON_0	READ	****_****_****_***0_0000_0000_0000_0000b

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

2.2.2.3 Interrupts

Table 2.2-22 lists the interrupt output signals.

Table 2.2-22 Interrupt Output Signals

Port name	Type*3	Pulse / Level	INT ID*2
nVCPUMNTIRQ[3:0]	PPI	Level	PPI : 25
nCNTHPIRQ[3:0]	PPI	Level	PPI : 26
nCNTPNSIRQ[3:0]	PPI	Level	PPI : 30
nCNTPSIRQ[3:0]	PPI	Level	PPI : 29
nCNTVIRQ[3:0]	PPI	Level	PPI : 27
nCNTHVIRQ[3:0]	PPI	Level	PPI : 28
nFAULTIRQ[4:1]	SPI	Level	—
nFAULTIRQ[0]	SPI	Level	—
nERRIRQ[4:1]	SPI	Level	—
nERRIRQ[0]	SPI	Level	—
CTIIRQ[3:0]	PPI	Level*1	PPI : 24
nCLUSTERPMUIRQ	SPI	Level	—
nPMUIRQ[3:0]	PPI	Level	PPI : 23
nCOMMIRQ[3:0]	PPI	Level	PPI : 22

Note 1. These signals are configured as level interrupt because this LSI is intended to use software control. To clear the interrupt, use the CTIAPPCLEAR register.

Note 2. For details of INT ID of SPI, see **4.6 Interrupt Controller**. The interrupts classified as PPI under the Type column can also be used as SPI.

Note 3. For details on PPI and SPI interrupts, see **4.6.2 Generic Interrupt Controller (GIC).**

2.2.2.4 Low Power Consumption Mode

The CA55 supports two standby modes.

- Core wait for interrupt (WFI)
- Core wait for event (WFE)

For details on each mode, see the Arm® Cortex®-A55 Core Technical Reference Manual and related documents.

The CA55 enters standby mode immediately after the execution of a WFI/WFE instruction.

2.2.2.4.1 Application Sleep Mode

This LSI defines the procedure for the CA55 to execute WFI/WFE instructions as the application sleep mode. This mode allows the CA55 to move to the low-power consumption mode safely without adversely affecting the system.

For the sequence for transition to this mode, see **4.4 Clock Pulse Generator (CPG)**.

2.2.2.5 Function Reference

For details on the functions of the DynamIQ™ Shared Unit and CA55, see the documents listed below.

- Arm® DynamIQ Shared Unit Technical Reference Manual
- Arm® Cortex®-A55 Core Technical Reference Manual
- Arm® Cortex®-A55 Core Advanced SIMD and Floating-point Support Technical Reference Manual
- Arm® Cortex®-A55 Core Cryptographic Extension Technical Reference Manual

For details on clock and the controls, see **4.4 Clock Pulse Generator (CPG)**.

2.2.2.6 Boot Procedure

The procedure for booting the CA55 is the same as that for releasing the cold reset.

Before releasing the cold reset, the following steps are required.

1. Set the configuration settings controllable by the SYS*[1].
2. Set the clock supply*[2].

Note 1. Related SYS registers
SYS_ACPU_CFG_SMPH / L
SYS_ACPU_CFG_EMPH / L
SYS_ACPU_CFG_RVAHx / Lx (x = 0 to 3)

Note 2. Related CPG registers (see **4.4 Clock Pulse Generator (CPG)** for the settings of PLLs and dividers.)
CPG_CLKON_0
CPG_CLKMON_0

When starting the CA55 with debug mode enabled, check that it has been released from reset (perform step 7 in **Table 2.2-7**) and then set the connection with CoreSight (CST) to the communication ready state.

2.2.2.7 CoreSight Connection

The CA55 performs local communications with CST using the ATB, APB, and timestamp interfaces. Though power-down is supported in this LSI, since the CA55 and CST belong to a different power domain, the two states are defined for the availability of communications.

- Communication ready state
- Communication interrupted state (default setting)

When powering down, place these interfaces in the communication interrupted state. When performing communications with the CST, place them in the communication ready state.

Since the ATB and timestamp interfaces have power-down registers, use software control to place these interfaces in the communication interrupted state.

The procedure for transition to the communication ready state is given below.

Table 2.2-23 Procedure for Transition to Communication Ready State

Step	CPU	Processing
1	Any	Write 1b to the CSYSREQ_ATx (x = 0/1/2/3) and CSYSREQ_TS bits of the CPG_LP_CA55_CTL7 register.
2	Any	Check that the CSYSACK_ATx and CSYSACK_TS bits of the CPG_LP_CA55_CTL7 register are 1b.
3	Any	Write 1b to the LPI_ATB_REQ_CA55x bit (x = 0/1/2/3) of the CPG_LP_CST_CTL2 registers.*1
4	Any	Check that the LPI_ATB_ACK_CA55x bit (x = 0/1/2/3) of the CPG_LP_CST_CTL2 register is 1b.

Note 1. Controlling the timestamp interface of the CST is not required because it is common to all CPUs.

The procedure for transition to the communication interrupted state is given below.

Table 2.2-24 Procedure for Transition to Communication Interrupted State

Step	CPU	Processing
1	Any	Write 0b to the CSYSREQ_ATx (x = 0/1/2/3) and CSYSREQ_TS bits of the CPG_LP_CA55_CTL7 register.
2	Any	Check that the CSYSREQ_ATx (x = 0/1/2/3) and CSYSREQ_TS bits of the CPG_LP_CA55_CTL7 register are 0b.
3	Any	Write 0b to the LPI_ATB_REQ_CA55x bit (x = 0/1/2/3) of the CPG_LP_CST_CTL2 register.*1
4	Any	Check that the LPI_ATB_ACK_CA55x bit (x = 0/1/2/3) of the CPG_LP_CST_CTL2 register is 0b.

Note 1. Controlling the timestamp interface of the CST is not required because it is common to all CPUs.

2.2.2.8 Usage Note for Dual Core Products

In dual-core products, Quad core information is read when the register indicating the number of cores is read.

(Example: CLUSTERCFR[2:0] = 011b)

2.2.3 System CPU Cortex-M33 (CM33)

The CM33 is a core block equipped with an Arm Cortex-M33 CPU.

For details on the functions of the CM33, see the Arm® Cortex®-M33 Processor Technical Reference Manual.

Refer to **1.8.2 CM33 Address Space** for CM33 address space.

2.2.3.1 Features

The CM33 processor is a highly energy-efficient processor that is intended for a microcontroller and deeply embedded applications. The processor is based on the Armv8-M architecture. For details on the functions of the CM33, see the Arm® Cortex®-M33 Processor Technical Reference Manual.

Table 2.2-25 Functional Overview

	Description
CPU	• Cortex-M33 r0p4 (ARMv8-M)
Maximum operating frequency	• 200 MHz
Interrupt controller	• NVIC (Nested Vectored Interrupt Controller)*1

Note 1. Interrupts generated in the system of this LSI are conveyed to the internal NVIC of Cortex-M33 via the ICU. For the various settings and setting procedures, refer to the *Arm® Cortex®-M33 Processor Technical Reference Manual* and **4.6.1 Interrupt Control Unit (ICU)**.

Table 2.2-26 CM33 Configuration

Description	Configuration
CPU	Cortex-M33 r0p4 (ARMv8-M)
Floating-point	Single-precision floating-point only.
DSP extension	Armv8-M DSP Extension supported
Security extension	Armv8-M security extension
Non-secure protected memory regions	16 regions
Secure protected memory regions	16 regions
Security attribution unit (SAU)	8 regions
Interrupts	480 interrupts
Number of bits of interrupt priority	8 bits are supported. 256 levels of priority are implemented.
Debug watchpoints and breakpoints	Full set 4 data watchpoint comparators and 8 breakpoint comparators
ITM and data watchpoint and trace (DWT) trace functionality	Complete ITM and DWT trace
Embedded trace macrocell (ETM)	ETM instruction execution trace
Micro trace buffer (MTB)	MTB is not supported.
Cross trigger interface (CTI)	CTI is included.
Wakeup interrupt controller (WIC)	WIC is included.
External coprocessor interface	Coprocessor hardware is not supported.

Table 2.2-27 Initial Settings of CM33 Configuration Pins

Item	Initial Value	Description	Switching Timing
CFGSSTCALIB [25:0]	0000_3D08h	Secure SysTick calibration. The value is fixed.	—
CFGNSSTCALIB [25:0]	0000_3D08h	Non-Secure SysTick calibration. The value is fixed.	—
INITSVTOR[31:7]	0000_0000h	Secure Reset Vector. The value can be changed.*1	Cold / warm reset
INITNSVTOR[31:7]	1800_0000h	Non-Secure Reset Vector. The value can be changed.*1	Cold / warm reset

Note 1. Change the setting in accord with the setting of IDAU.

2.2.3.2 Resets

2.2.3.2.1 Reset Functions and Areas

The table below lists the reset functions supported by the CM33 and the corresponding areas to be reset. Supply the clock before releasing the reset.

Table 2.2-28 Supported Reset Combinations

Reset Function	Areas to be Reset
Cold reset	Entire area
Warm reset	Cortex-M33 excluding the debug module

2.2.3.2.2 Cold Reset

A cold reset is applied at any of the following cases.

(a) The power-on reset sequence is executed (in CM33 boot mode).
(b) The WDT counter has underflowed.
(c) Software control by the other CPU is applied.

For details on (a) and (b), see **4.4 Clock Pulse Generator (CPG)** and **5.4 Watchdog Timer (WDT)**. The procedure for applying a cold reset by case (c) is given below. The CPU column shows the CPU which executes processing. The Value column shows the setting at the time of writing and the expected value at the time of reading. When reading, continue polling until the expected value is obtained.

Table 2.2-29 Procedure for Applying Cold Reset

Step	CPU	Register Name	Processing	Value
1	CM33	Execute system MPU sleep.*1		
2	Other	CPG_LP_CM33CTL0	Read	****_****_****_****_****_****_***1_***1b
3	Other	CPG_RST_1	Write	****_****_**11_1***_****_****_**00_0***b
4	Other	CPG_RSTMON_0	Read	****_****_****_111*_****_****_****_****b

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

Note 1. The CM33 cannot be controlled when the WDT count underflow interrupt or LOCKUP is detected. If this is the case, skip this step.

The procedure for releasing the cold reset is as follows. The booting procedure differs depending on the operating mode (normal mode or debug mode).

Table 2.2-30 Procedure for Releasing Cold Reset (Normal mode)

Step	CPU	Register Name	Processing	Value
1	Other	CPG_RSTMON_0	Read	****_****_****_111*_****_****_****_****b
2	Other	CPG_RST_1	Write	****_****_**11_1***_****_****_**00_1***b
3	Other	CPG_RSTMON_0	Read	****_****_****_110*_****_****_****_****b
4	Other	CPG_RST_1	Write	****_****_**11_1***_****_****_**11_1***b
5	Other	CPG_RSTMON_0	Read	****_****_****_000*_****_****_****_****b

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

Table 2.2-31 Procedure for Releasing Cold Reset (Debug mode)

Step	CPU	Register Name	Processing	Value
1	Other	CPG_LP_CM33CTL1	Write	****_****_****_****_**1*_****_****_****b
2	Other	CPG_CM33_CTL	Write	****_****_****_****_****_****_****_***1b
3	Other	CPG_RSTMON_0	Read	****_****_****_111*_****_****_****_****b
4	Other	CPG_RST_1	Write	****_****_**11_1***_****_****_**00_1***b
5	Other	CPG_RSTMON_0	Read	****_****_****_110*_****_****_****_****b
6	Other	CPG_RST_1	Write	****_****_**11_1***_****_****_**11_1***b
7	Other	CPG_RSTMON_0	Read	****_****_****_000*_****_****_****_****b
8	-	Sets registers regarding debug from CST via D-AHB		
9	Other	CPG_CM33_CTL	Write	****_****_****_****_****_****_****_***0b

Note: Writing to bits with * is prohibited (read-modify-write operations are required).

Before releasing the cold reset, the following steps are required.

1. Set the configuration settings controllable by the SYS*[1].
2. Set the clock supply*[2].

Note 1. Related SYS registers
SYS_MCPU_CFGx (x = 2 to 5)

Note 2. Related CPG registers (see **4.4 Clock Pulse Generator (CPG)** for the settings of PLLs and dividers.)
CPG_CLKON_1
CPG_CLKMON_0

2.2.3.2.3 Warm Reset

When applying a warm reset, the WFI instruction*[1] is used to guarantee that no transaction is issued. For the procedure for applying a warm reset, see “CM33 Warm Reset Operation Procedure” in **4.4 Clock Pulse Generator (CPG)**. In this LSI, hardware control is used to clear INT_MASK and release the warm reset. After the WFI has been issued, the CM33 is automatically restored.

Note 1. Correct operation is not guaranteed if the WFE instruction is executed.

2.2.3.3 Interrupts

2.2.3.3.1 Interrupt Pins

Table 2.2-32 lists the interrupt input pins and the interrupt output pins, respectively.

Table 2.2-32 Interrupt Output Pins

Port Name	Interrupt Content	Pulse / Level
CTIIRQ [1:0]	CTI interrupt output	Level
LOCKUP*1	Processor is in Lockup state	Level
FPIXC	Inexact	Level
FPIDC	Input denormal	Level
FPOFC	Overflow	Level
FPUFC	Underflow	Level
FPDZC	Divide-by-zero	Level
FPIOC	Invalid operation	Level
ERRINT_C*2	C-AHB bridge error interrupt	Pulse
ERRINT_S*2	S-AHB bridge error interrupt	Pulse

Note 1. LOCKUP is a signal that is asserted when the Cortex-M33 hangs. If LOCKUP is detected, executing a cold reset is recommended. For the conditions for asserting LOCKUP, refer to the ARM®v8-M Architecture Reference Manual.

Note 2. ERRINT_C and ERRINT_S exist as pins, but these pins are unavailable because they are not used within the IP.

2.2.3.4 Address Space

For details on the address space, refer to the Arm® Cortex®-M33 Processor Technical Reference Manual.

2.2.3.4.1 IDAU Setting

The CM33 includes an Implementation Defined Attribution Unit (IDAU). It supports two security maps. The security attribute settings are valid on release from the cold reset, and dynamic switching is not supported. In this LSI, CM33 Config Register 4 is used to select a security map. The initial setting of IDAUZERONS is 0. **Table 2.2-33** shows the security attributes assigned by the IDAU.

Table 2.2-33 IDAU Security Attributes

Area No.	Security Attribute IDAUZERONS = 0 (default)	Security Attribute IDAUZERONS = 1	Lower-limit Address	Upper-limit Address
0	Secure	Non-secure	0000_0000h	0FFF_FFFFh
1	Non-secure	Secure	1000_0000h	1FFF_FFFFh
2	Secure	Non-secure	2000_0000h	2FFF_FFFFh
3	Non-secure	Secure	3000_0000h	3FFF_FFFFh
4	Secure	Non-secure	4000_0000h	4FFF_FFFFh
5	Non-secure	Secure	5000_0000h	5FFF_FFFFh
6	Secure	Non-secure	6000_0000h	6FFF_FFFFh
7	Non-secure	Secure	7000_0000h	7FFF_FFFFh
8	Secure	Non-secure	8000_0000h	8FFF_FFFFh
9	Non-secure	Secure	9000_0000h	9FFF_FFFFh
10	Secure	Non-secure	A000_0000h	AFFF_FFFFh
11	Non-secure	Secure	B000_0000h	BFFF_FFFFh
12	Secure	Non-secure	C000_0000h	CFFF_FFFFh
13	Non-secure	Secure	D000_0000h	DFFF_FFFFh
14	Secure	Non-secure	E000_0000h	EFFF_FFFFh
15	Non-secure	Secure	F000_0000h	FFFF_FFFFh

2.2.3.5 Description of Functions

For the registers and functions of Cortex-M33, refer to the Arm® Cortex®-M33 Processor Technical Reference Manual and Arm®v8-M Architecture Reference Manual.

2.2.3.6 Low Power Consumption Mode

2.2.3.6.1 Supported Low Power Consumption Modes

The CM33 supports the core wait for interrupt (WFI) and core wait for event (WFE) instructions. For the details, refer to the Arm® Cortex®-M33 Processor Technical Reference Manual.

The CM33 also supports three sleep modes that are enabled by a combination of the internal register SCR.DEEPSLEEP and the CPG_LP_CM33CTL0 register (WIC interface). The combinations are shown below.

Table 2.2-34 Low Power Consumption Mode Settings

Name	SCR.DEEPSLEEP	CPG_LP_CM33CTL0 Register	Note
Standard standby	0b	CM33_LP_CTL24 = 0b CM33_LP_CTL25 = 0b	Standard low power consumption mode using WFI/WFE. This LSI uses this mode with a warm reset.
Deep standby	1b	CM33_LP_CTL24 = 0b CM33_LP_CTL25 = 0b	This LSI uses this mode in CM33 sleep mode.
WIC standby	1b	CM33_LP_CTL24 = 1b CM33_LP_CTL25 = 1b	This LSI uses this mode in software standby mode.
—	0b	CM33_LP_CTL24 = 1b CM33_LP_CTL25 = 1b	Not supported

2.2.3.6.2 Method for Executing System MPU Sleep Mode

System MPU sleep mode of this LSI is a low power consumption mode that uses the WFI or WFE instruction of the Cortex-M33. For control over transition to this mode, see **4.4 Clock Pulse Generator (CPG)**.

2.2.3.6.3 Method for Executing Software Standby Mode

Software standby mode of this LSI is an ultra-low power consumption mode that stops all modules other than the WIC of the CM33. For control over transition to this mode, see **4.4 Clock Pulse Generator (CPG)**. The WFE instruction cannot be used because return from this mode is interrupt-driven.

For the Q-channel specifications, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces. The specifications specific to the CM33 are given below.

Table 2.2-35 Q-channel Control Specifications

Domain	Item	Description
FPU	QACTIVE assertion condition	• The FPU is enabled by the CPACR register. • The FPU is processing instructions
	Power-up condition	• The CM33 is operating normally. (Q_EXIT is retained while the CM33 is not started)
	Power-down condition	• Q-REQUEST is issued while QACTIVE is negated.
Debug	QACTIVE assertion condition	• When PPB access to the debug module occurs (such access is not executed and is retained until the debug domain enters Q-RUN). • The DEMCR.TRACE bit is set. • The BPU is enabled. • The ETM is enabled. • The ITM, WDT, and ETM are active (trace data remains).
	Power-up condition	• The CM33 is operating normally. (Q_EXIT is retained while the CM33 is not started)
	Power-down condition	• Q-REQUEST is issued while QACTIVE is negated.

Note: The software standby sequence requires powering down each domain before issuing the WFI instruction.

(1) SysTick Timers

The SysTick timers in the CM33 are described below.

- There is one Secure SysTick timer and one Non-secure SysTick timer.
- The count cycle of SysTick timers is common to Secure and Non-secure.
- A reference clock for counting by a SysTick timer is implemented in this LSI.
- The configuration of Systick timers is a fixed value. The values are reflected in register SYST_CALIB*[1] of the Cortex-M33. For the settings, see **Table 2.2-27**. The settings of the 24 lower-order bits and the 25th bit are reflected in TENMS and SKEW, respectively. The setting of the 26th bit is reflected in NOREF, but set the bit to 0b because a reference clock is implemented in this LSI.

Note 1. For details on the SYST_CALIB register, refer to the Arm®v8-M Architecture Reference Manual.

(2) Usage of FPU

When using the FPU of the CM33, the FPU must be enabled by using CPACR of an internal register after setting the state of Q-channel of the FPU domain to Q-RUN (in this LSI, Q-channel of the FPU domain is set to the Q-STOPPED state by default).

The registers for controlling Q-channel of the FPU domain are included in the CPG. For details, refer to the list of the CPG registers.

For details on the Q-channel control specifications of the FPU domain, see **Table 2.2-35**.

(3) Usage of the Debug Function

Using the debug function of Cortex-M33 requires setting the state of Q-channel of the debug domain to Q-RUN (the initial state of Q-channel of the debug domain differs depending on the boot mode).

The registers for controlling Q-channel of the debug domain are included in the CPG. For details, see **4.4 Clock Pulse Generator (CPG)**.

For details on the Q-channel control specifications of the debug domain, see **Table 2.2-35**.

(4) Restrictions of Functions

The CM33 has some restrictions on the functions due to specification of this LSI. **Table 2.2-36** lists the restrictions on the functions.

Table 2.2-36 List of Restrictions of Functions

No.	Description
1	Non-support for EPPB In this LSI, the CST is accessible via the system bus. Therefore, operation for access to the CST by the EPPB is not guaranteed.
2	Non-support for WFE instruction in warm reset and software standby In this LSI, operation of the WFE instruction is not guaranteed while a warm reset is being executed. Return from software standby mode is interrupt-driven. Return by an event is not supported.
3	Non-support for NMI This LSI assumes that NMIs are processed not only by the CM33 but also by other CPUs. Therefore, the NMI function of the CM33 is not supported. The function is implemented as maskable interrupt as with other CPUs.
4	Systick timer accuracy In this LSI, dynamic switching of the frequency of CLK0 is possible, but the timer accuracy is not guaranteed when there is a dynamic frequency switchover during the measurement period.
5	Debugging-disabled period Since the internal debug module is stopped while in software standby mode during debug mode, debugging is disabled during this period.

SECTION 2 PROCESSORS

2.3 Ethos-U55 (NPU)

2.3.1 Overview

This LSI includes the Arm Ethos-U55 NPU as a Neural Processing Unit (NPU). The Arm Ethos-U55 NPU targets 8-bit and 16-bit integer quantized Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). The NPU supports 8-bit weights.

2.3.1.1 Features

The NPU has the following features.

- The number of 8x8 MACs that are performed per cycle is 256.
- Maximum inferential performance is 0.5TOPs.
- Including a DMA controller that can read and write to external memory.
- Two Arm® AMBA® 5 AXI Manager interfaces that provide the DMA controller with access to external memory.
  - A read/write manager, M0.
  - A read-only manager, M1.
- An Arm® AMBA® 4 APB completer interface that allows the application processor to program the NPU.

2.3.1.2 Block Diagram

Figure 2.3-1 shows the NPU system diagram and **Figure 2.3-2** shows the NPU block diagram.

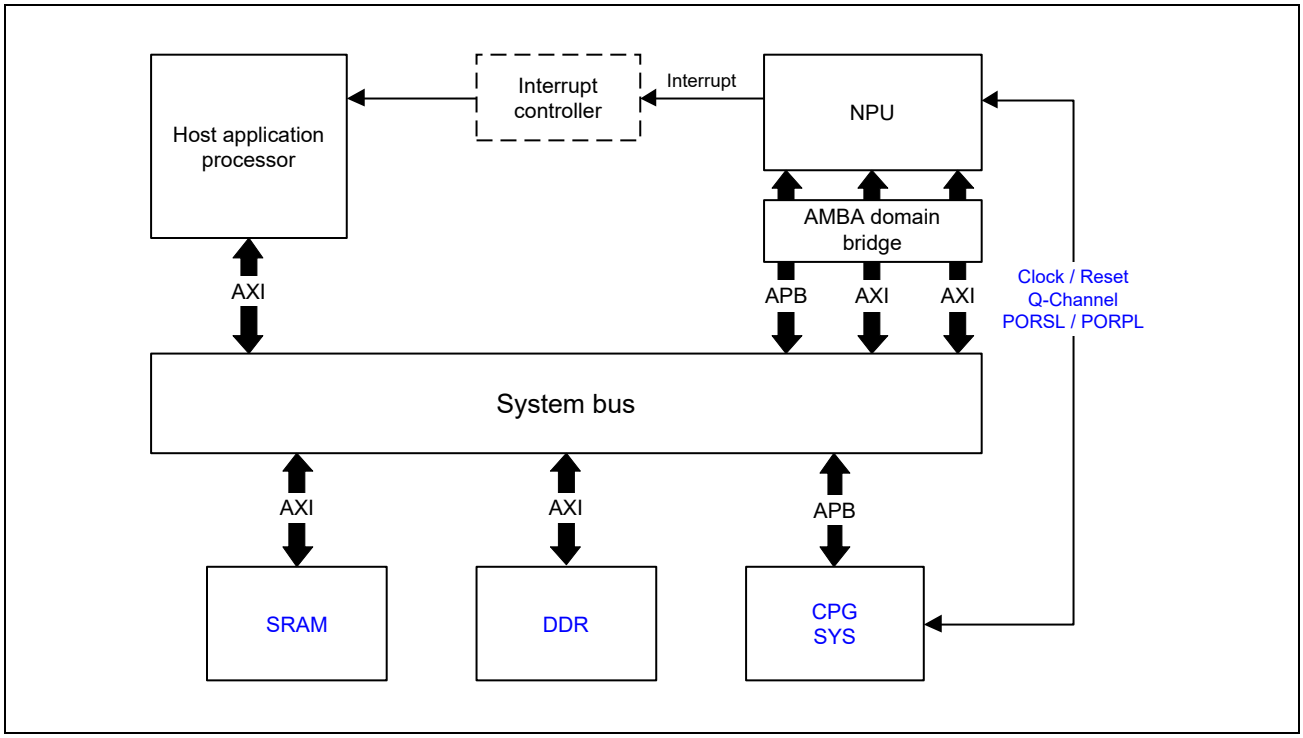


Figure 2.3-1 System Diagram

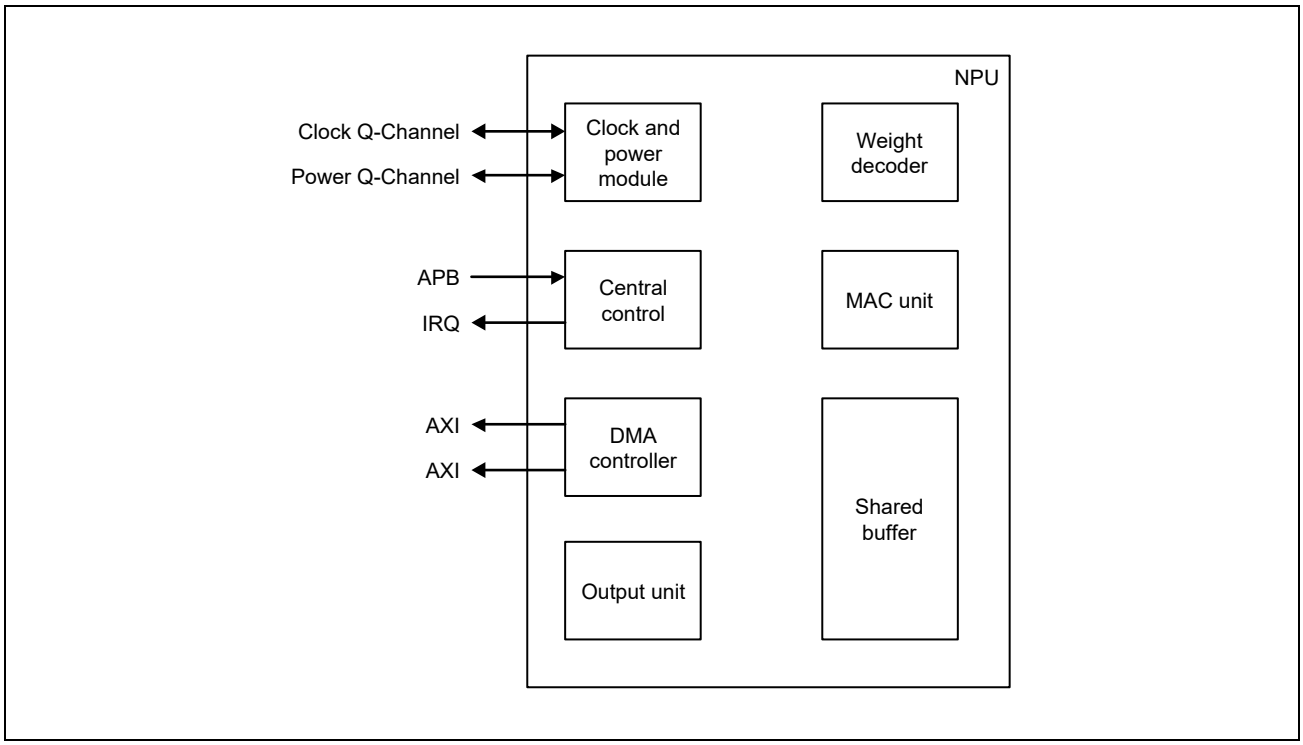


Figure 2.3-2 Block Diagram

2.3.1.3 Registers

Base address of NPU registers is allocated to 0_19C8_0000h. The address range is 64 KB (0_19C8_0000h - 0_19C8_FFFFh). For more information on NPU registers, refer Arm® Ethos™-U55 NPU Technical Reference Manual 4. Programmers model.

2.3.1.4 Usage Note

Renesas does not provide technical support for users to program NPU registers directly. Arm delivers the hardware Register Transfer Level (RTL) of the NPU with an open-source driver and compiler. A neural network must be compiled offline using the open-source compiler to produce a command stream. The application invokes the driver, which communicates with the NPU to tell it where the command stream is and initiates the network traversal. The command stream describes the steps necessary for the NPU to execute the operators compiled into the command stream autonomously. When complete, the NPU raises an IRQ to the driver.

SECTION 3 MEMORY

3.1 Memory Overview

This LSI has a 2-channel SRAM (SRAM0, SRAM1).

Each SRAM size is 256 KB for a total of 512 KB SRAM. See **3.2 SRAM** for details on SRAM.

The LPDDR4/4X controller (DDR) is used to connect external memory.

This unit conforms to the following standards.

- LPDDR4 (JEDEC STANDARD JESD209-4D)
- LPDDR4X (JEDEC STANDARD JESD209-4-1A)

See **3.4 LPDDR4/4X Controller (DDR)** for details.

This LSI is equipped with six TrustZone address space controllers (TZCs) to ensure safe memory access.

See **3.5 TrustZone Address Space Controller (TZC)** for details.

SECTION 3 MEMORY

3.2 SRAM

3.2.1 Overview

This LSI has two 256-Kbyte areas of on-chip RAM for use as the CM33, and Ethos-U55 working areas.

Each of on-chip RAM incorporates an ECC function, which allows the detection and correction of 1-bit errors and the detection of 2-bit errors. When a 1-bit or 2-bit error occurs, the address where the error occurred can be stored in an ECC error address register.

Table 3.2-1 shows the relationship between the unit number and the base address in RAM area.

Table 3.2-1 Relationship between Unit Number and Address <SRAMm_base> (m = 0 to 1)

Unit Number	Channel Number	RAM Area Base Address <SRAMm_base>	RAM Area Base Address (Code Area) <SRAMm_base>	RAM Area Base Address (Data Area) <SRAMm_base>
SRAM0	2	0_07F8_0000h	17F8_0000h*1, 07F8_0000h*2	37F8_0000h*1, 27F8_0000h*2
		0_0800_0000h	1800_0000h*1, 0800_0000h*2	3800_0000h*1, 2800_0000h*2
SRAM1	8	0_0804_0000h	1804_0000h*1, 0804_0000h*2	3804_0000h*1, 2804_0000h*2

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

3.2.1.1 SRAMm (m = 0)

The data bus width is 64 bits. This is divided into two channels as shown in **Figure 3.2-1**, with detection handled in each 32-bit channel.

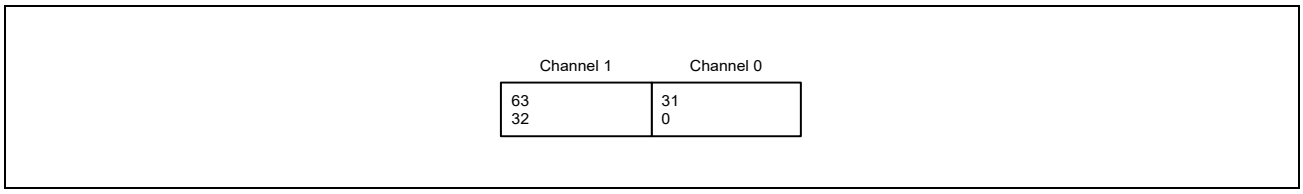


Figure 3.2-1 Division for ECC Processing for SRAMm (m = 0)

Table 3.2-2 Relationship between Each Channel and Address for SRAMm (m = 0)

Channel Number	Address
Ch. 0	<SRAMm_base> + 20h × k
Ch. 1	<SRAMm_base> + 20h × k + 04h

Note: k = 0, 1, 2, …, 32767

Note: <SRAMm_base>: See **Table 3.2-1**.

3.2.1.2 SRAMm (m = 1)

The data bus width is 256 bits. This is divided into eight channels as shown in **Figure 3.2-2**, with detection handled in each 32-bit channel.

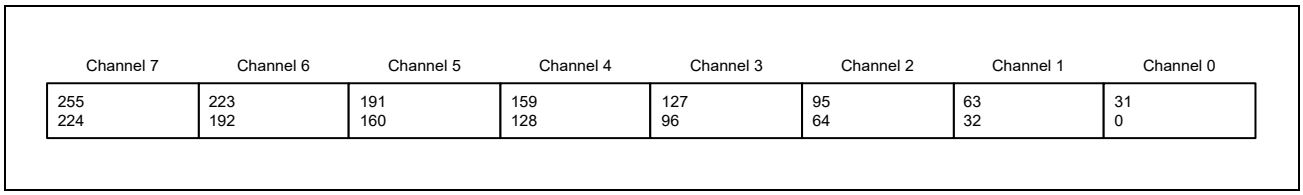


Figure 3.2-2 Division for ECC Processing for SRAMm (m = 1)

Table 3.2-3 Relationship between Each Channel and Address for SRAMm (m = 1)

Channel Number	Address
Ch. 0	<SRAMm_base> + 20h × k
Ch. 1	<SRAMm_base> + 20h × k + 04h
Ch. 2	<SRAMm_base> + 20h × k + 08h
Ch. 3	<SRAMm_base> + 20h × k + 0Ch
Ch. 4	<SRAMm_base> + 20h × k + 10h
Ch. 5	<SRAMm_base> + 20h × k + 14h
Ch. 6	<SRAMm_base> + 20h × k + 18h
Ch. 7	<SRAMm_base> + 20h × k + 1Ch

Note: k = 0, 1, 2, …, 8191

Note: <SRAMm_base>: See **Table 3.2-1**.

3.2.1.3 Features

- Error detection and correction
  - Detection and correction of 1-bit errors and detection of 2-bit errors are possible.
  - An error involving 3 or more bits cannot be correctly handled and may result in erroneous detection and correction.
- Control of error detection and correction
  - Error detection and correction, and correction of 1-bit errors can respectively be enabled or disabled.
- Capture of the addresses where errors occurred
  - When a 1-bit or 2-bit error is detected, the address where the error was encountered is stored in an ECC error address register. Each channel has eight ECC error address registers.
- Interrupt request
  - An interrupt request can be generated on the detection of a 1-bit error (selectable as enabled or disabled).
  - An interrupt request can be generated on the detection of a 2-bit error (selectable as enabled or disabled).
  - An overflow interrupt is generated from each channel on detection of an error when all ECC error address registers contain values.

3.2.2 RAM Registers

The base addresses for the respective channels are as follows.

Table 3.2-4 Register Base Addresses

Base Register Name	Unit Name	Base Address
<SRAM0_base>	SRAM0	0_1101_0000h (5101_0000h*1, 4101_0000h*2)
<SRAM1_base>	SRAM1	0_19C0_0000h (59C0_0000h*1, 49C0_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

3.2.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
ECC Control Registers_n	SRAMm_CTL_n	0000_0010h	0000h + n x 0040h	32
ECC Error Address Registers 0_n	SRAMm_EAD0_n	0000_0000h	0010h + n x 0040h	32
ECC Error Address Registers 1_n	SRAMm_EAD1_n	0000_0000h	0014h + n x 0040h	32
ECC Error Address Registers 2_n	SRAMm_EAD2_n	0000_0000h	0018h + n x 0040h	32
ECC Error Address Registers 3_n	SRAMm_EAD3_n	0000_0000h	001Ch + n x 0040h	32
ECC Error Address Registers 4_n	SRAMm_EAD4_n	0000_0000h	0020h + n x 0040h	32
ECC Error Address Registers 5_n	SRAMm_EAD5_n	0000_0000h	0024h + n x 0040h	32
ECC Error Address Registers 6_n	SRAMm_EAD6_n	0000_0000h	0028h + n x 0040h	32
ECC Error Address Registers 7_n	SRAMm_EAD7_n	0000_0000h	002Ch + n x 0040h	32

Note: n: SRAMm channel number (m = 0 to 1)

3.2.3 RAM Register Descriptions

3.2.3.1 ECC Control Registers_n (SRAMm_CTL_n) (n: SRAMm Channel Number)

Each ECC control register_n is used to control the ECC function in the corresponding channel n.

After writing the initial value to the entire usage area of the on-chip RAM, set the given ECERVF bits to 1b to enable error detection. Be sure to write 01b to the corresponding EMCA[1:0] bits to enable writing to an ECERVF bit.

For writing the initial value to the on-chip RAM areas, see **3.2.5 Initializing the Detection-Usage Areas of the On-Chip RAM**.

Access Size : 32 bits

Address : <SRAMm_base> + 0000h + n x 0040h

Initial Value : 0000_0010h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECDEDF7	ECSEDF7	ECDEDF6	ECSEDF6	ECDEDF5	ECSEDF5	ECDEDF4	ECSEDF4	ECDEDF3	ECSEDF3	ECDEDF2	ECSEDF2	ECDEDF1	ECSEDF1	ECDEDF0	ECSEDF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EMCA[1:0]		-	-	ECOVFF	ECER2C	ECER1C	-	-	ECERVF	EC1ECP	EC2EDIC	EC1EDIC	ECER2F	ECER1F	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	RW	RW	R	R	R	RW	RW	R	R	RW	RW	RW	RW	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	ECDEDF7	0h	R	These bits hold the address of the (x + 1)th error to have been detected. These bits are cleared to 0b under the following condition: Clearing ECER2F or ECER1F in the SRAMm_CTL_n register.
30	ECSEDF7	0h	R	
29	ECDEDF6	0h	R	
28	ECSEDF6	0h	R	
27	ECDEDF5	0h	R	
26	ECSEDF5	0h	R	
25	ECDEDF4	0h	R	
24	ECSEDF4	0h	R	
23	ECDEDF3	0h	R	
22	ECSEDF3	0h	R	
21	ECDEDF2	0h	R	
20	ECSEDF2	0h	R	
19	ECDEDF1	0h	R	
18	ECSEDF1	0h	R	
17	ECDEDF0	0h	R	
16	ECSEDF0	0h	R	
15, 14	EMCA[1:0]	0h	RW	Enable Writing to ECERVF 01b: Enabled Others: Disabled These bits are always read as 00b.
13, 12	-	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
11	ECOVFF	0h	R	ECC Overflow Detection Flag This bit is set to 1b and an overflow interrupt is output on detection of an error when the ECC error address registers (SRAMm_EADp_n (p = 0 to 7)) all contain data. An overflow interrupt is output even if the error is detected while the setting of this bit is already 1b. This bit is cleared to 0b under any of the following conditions. 1. Power-on reset 2. Writing 1b to the corresponding ECER2C bit or ECER1C bit 3. Setting the ECC error detection enable bit to “disabled” (ECERVF = 0b)

Bit	Bit Name	Initial Value	R/W	Description
10	ECER2C	0h	RW	2-Bit Error Detection Flag Clear 0b: The internal state does not change. 1b: The corresponding 2-bit error detection flag ECER2F is cleared. This bit is always read as 0b.
9	ECER1C	0h	RW	1-Bit Error Detection Flag Clear 0b: The internal state does not change. 1b: The corresponding 1-bit error detection flag ECER1F is cleared. This bit is always read as 0b.
8, 7	-	All 0	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
6	ECERVF	0h	RW	ECC Error Detection Enable 0b: Disables error detection. 1b: Enables error detection. 1-bit error correction (EC1ECP) and interrupt control (EC2EDIC and EC1EDIC) are only possible when error detection is enabled. Note: To write to this bit, the corresponding EMCA[1:0] bits must be set to 01b.
5	EC1ECP	0h	RW	1-Bit Error Correction Enable 0b: The error is corrected when a 1-bit error is detected. 1b: The error is not corrected when a 1-bit error is detected.
4	EC2EDIC	1h	RW	2-Bit Error Detection Interrupt Control 0b: An interrupt is not generated when a 2-bit error is detected. 1b: An interrupt is generated when a 2-bit error is detected. The interrupt is only generated when the corresponding 2-bit error detection flag (ECER2F) changes from 0 to 1.
3	EC1EDIC	0h	RW	1-Bit Error Detection Interrupt Control 0b: An interrupt is not generated when a 1-bit error is detected. 1b: An interrupt is generated when a 1-bit error is detected. The interrupt is only generated when the corresponding 1-bit error detection flag (ECER1F) changes from 0 to 1.
2	ECER2F	0h	R	2-Bit Error Detection Flag 0b: A 2-bit error has not occurred. 1b: A 2-bit error has occurred. This bit is cleared to 0b under any of the following conditions. 1. Power-on reset 2. Writing 1b to the corresponding ECER2C bit 3. Setting the ECC error detection enable bit to "disabled" (ECERVF = 0b)
1	ECER1F	0h	R	1-Bit Error Detection Flag 0b: A 1-bit error has not occurred. 1b: A 1-bit error has occurred. This bit is cleared to 0b under any of the following conditions. 1. Power-on reset 2. Writing 1b to the corresponding ECER1C bit 3. Setting the ECC error detection enable bit to "disabled" (ECERVF = 0b)
0	-	0h	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

3.2.3.2 ECC Error Address Registers p_n (SRAMm_EADp_n) (n: SRAMm Channel Number) (p = 0 to 7)

When a 1- or 2-bit error occurs, up to eight error addresses are sequentially stored in the registers from SRAMm_EAD0_n to SRAMm_EAD7_n.

When a 2-bit error is detected while only 1-bit error addresses were stored up to SRAMm_EAD7_n, SRAMm_EAD7_n is overwritten by the address of the 2-bit error. Otherwise, it will not be overwritten. For details, see **3.2.6.4 ECC Error Address Register Overwrite Conditions**.

Access Size : 32 bits

Address : <SRAMm_base> + 0010h + n x 0040h + p x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECEAD0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECEAD0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ECEAD0[31:0]	0h	R	These bits hold the address of the (x + 1)th error having been detected. These bits are cleared to 0b under the following condition: Clearing ECER2F or ECER1F in the SRAMm_CTL_n register.

3.2.4 Initializing the ECC Function

Initialize the ECC by following the procedure below to enable the ECC function.

1. Set the VECCEN bit in the ECCRAM setting register corresponding to the RAM area for which the ECC function is to be enabled to 1b.
2. Write the value to the entire usage area of the RAM for which the ECC function is to be enabled. The initial value can be any 32-bit value.
3. Set the ECERVF bits in the ECC control registers_n to 1b to enable error detection. Then, set the respective bits to control 1-bit error correction and interrupts.

For notes on the settings of registers of this module, see **3.2.6.1 Notes on Setting the Registers of This Module**.

ECC is initially enabled in SRAM0 by hardware control at booting.

3.2.5 Initializing the Detection-Usage Areas of the On-Chip RAM

Before using the ECC function, write the initial value to the entire usage area of the on-chip RAM for which the ECC function is to be enabled. The initial value can be any value. Since the detection of ECC errors is handled in 32-bit units in each channel, the initial value should also be written in 32-bit units.

Enabling error detection without initialization may cause an ECC error due to unintended reading.

Examples of Operations that May Lead to Errors

1. When writing is to proceed in 8- or 16-bit units, reading the 32-bit value from the given address and then writing the data
2. If the locations are not initialized, an ECC error may occur when the 32-bit value is read. For details on writing in 8- or 16-bit units, see **3.2.6.2 8- or 16-Bit Access**.
3. When the CPU cache is enabled, if a cache miss occurs in writing, the cache being refilled by a 1-line unit
   At this time if data are not initialized, an ECC error may occur because data are read in 1-line (32-byte) units.

3.2.6 Usage Notes

3.2.6.1 Notes on Setting the Registers of This Module

Ensure that none of the bus masters proceeds with access to the on-chip RAM areas while the registers listed in **3.2.2 RAM Registers** are being set up.

3.2.6.2 8- or 16-Bit Access

The ECC function for the on-chip RAM areas handles error detection in 32-bit units. Accordingly, when access is to be in 8- or 16-bit units, the operations proceed as follows.

3.2.6.2.1 8- or 16-bit reading

a) The 32 bits of data are read and the ECC is used to determine whether or not an error is present.

b) The 8 or 16 bits of data are then returned to the bus master that issued the read request.

3.2.6.2.2 8- or 16-bit writing

a) The 32 bits of data from the address that contains the location for writing are read and the ECC is used to determine whether or not an error is present.

b) The 32 bits of data that have been read are then overwritten with the 8- or 16-bit data.

c) The ECC value for the overwritten 32-bit data in (b) is added by writing it to the RAM.

3.2.6.3 Numbers of Cycles for Reading from or Writing to a RAM Area with the ECC Function Enabled

The numbers of cycles for reading from or writing to an on-chip RAM area with the ECC function enabled are as follows.

3.2.6.3.1 Reading

The number of cycles for reading in 8- or 16-bit and 32-bit units when the ECC function is enabled is one cycle of SRAM_0_ACLK or SRAM_1_ACLK longer than when the ECC function is disabled.

3.2.6.3.2 Writing

The number of cycles for writing in 8- or 16-bit units when the ECC function is enabled is two cycles of SRAM_0_ACLK or SRAM_1_ACLK longer than when the ECC function is disabled. The number of cycles for writing in 32-bit units when the ECC function is enabled is the same as when the ECC function is disabled.

3.2.6.4 ECC Error Address Register Overwrite Conditions

If a new ECC error occurs with the ECC error address stored up to SRAMm_EAD7_n, SRAMm_EAD7_n may or may not be overwritten. The conditions to be overwritten are shown below.

Error Address Storage Status Up to <REG_base> EAD7_n	New ECC Error	<REG_base>EAD7_n Overwrite	Overflow Interrupt
Only 1-bit error	1-bit error	Not overwritten	Occurred
Only 1-bit error	2-bit error	Overwritten	Occurred
Only 2-bit error	1-bit error	Not overwritten	Occurred
Only 2-bit error	2-bit error	Not overwritten	Occurred
1-bit error and 2-bit error	1-bit error	Not overwritten	Occurred
1-bit error and 2-bit error	2-bit error	Not overwritten	Occurred

SECTION 3 MEMORY

3.4 LPDDR4/4X Controller (DDR)

This manual is a simplified version. For more information, refer to the User's Manual Additional Document.

3.4.1 Overview

This section describes the features of the DDR unit of this LSI.

This unit is an external bus controller for LPDDR4 and LPDDR4X. This unit supports LPDDR4-3200 and LPDDR4X-3200. The interface bus width is 32 bits. It supports the in-line ECC feature.

It is composed of the DDR controller block (MC) and the DDR PHY block (PHY).

For the setup of this block, refer to the Linux software package of this LSI, which contains the DDR setup codes.

3.4.1.1 Features

This unit supports the following specifications.

- LPDDR4 (JEDEC STANDARD JESD209-4D)
- LPDDR4X (JEDEC STANDARD JESD209-4-1A)

Table 3.4-1 lists the features of this unit.

Table 3.4-1 LPDDR4/4X Controller Features

Feature	Description
DRAM I/F	• LPDDR4X-3200 Bus width: 32 bits Memory Size: Up to 4 GBytes, Single rank (15 mm PKG, 21 mm PKG) • LPDDR4-3200 Bus width: 32 bits Memory Size: Up to 8 GBytes, Dual rank (15 mm PKG) Memory Size: Up to 4 GBytes, Single rank (15 mm PKG, 21 mm PKG)
MC	• Fully pipelined command, read and write data interfaces to the controller. • Advanced bank look-ahead features for high memory throughput. • A programmable register interface to control memory parameters and protocols including auto pre-charge. • Full initialization of memory on controller reset. • Supports the Weighted Round-Robin arbitration schema for arbitrating request from the ports. • ECC function for single bit and double bit error reporting, single bit error correction, and programmable removal of ECC storage. • Built-in self-test (BIST) for external DRAM memories. • Programmable memory data path size of full memory data width or half memory data width.
PHY	For more information, refer to the User's Manual Additional Document.
Low power	• Multiple low power states (power-down, self-refresh, I/O retention) • Automatic or software interface

3.4.1.2 Block Diagram

Figure 3.4-1 shows a block diagram.

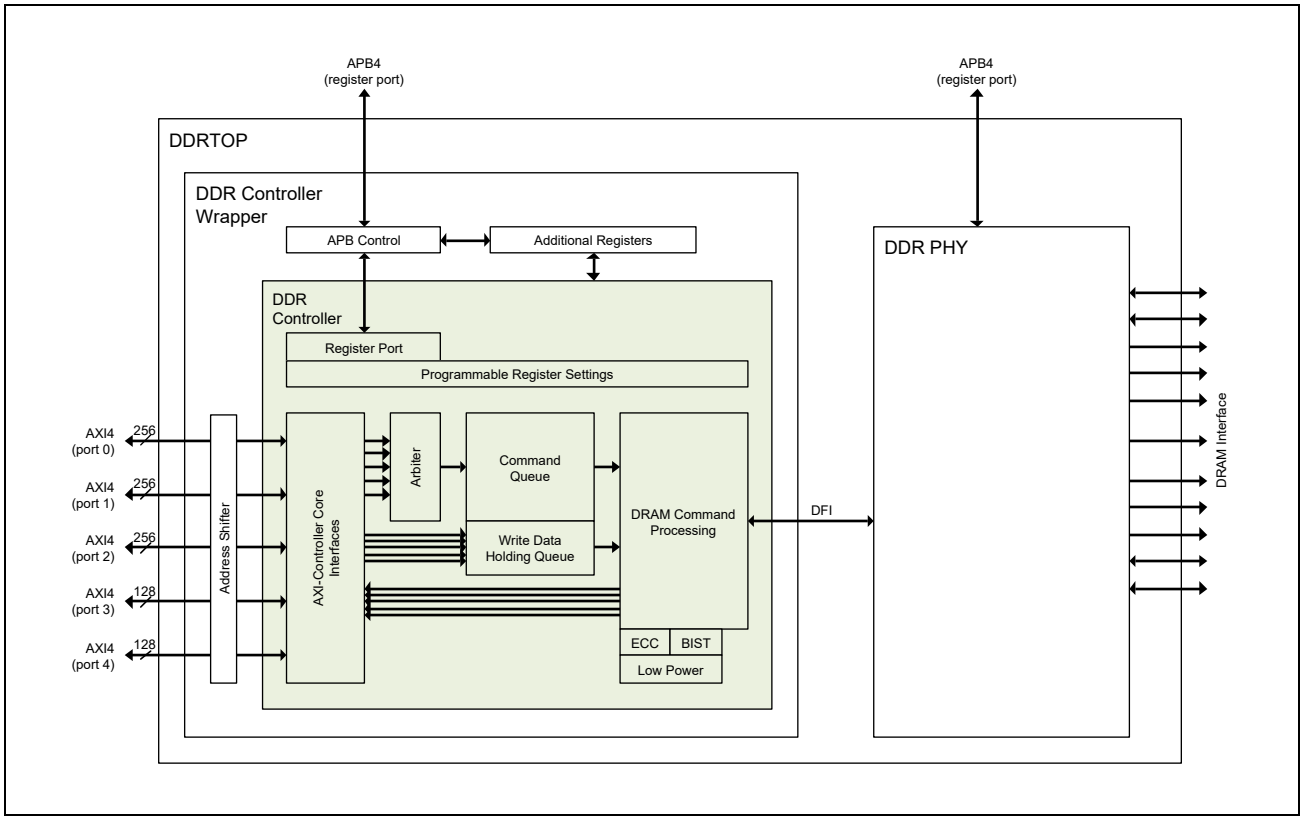


Figure 3.4-1 Block Diagram

3.4.1.3 External Pins

Table 3.4-2 lists the external pins of this unit.

Table 3.4-2 External Pins (1/2)

Pin Name	Input/Output	Description
DDRm_ZN	—	Calibration external reference resistor
DDRm_RESETN	Output	DRAM reset
DDRm_CKEA0	I/O	DRAM clock enable (rank 0)
DDRm_CKEA1	I/O	DRAM clock enable (rank 1)
DDRm_CAA0	I/O	DRAM command/address
DDRm_CAA1	I/O	DRAM command/address
DDRm_CKAT	I/O	DRAM clock (positive)
DDRm_CKAC	I/O	DRAM clock (negative)
DDRm_CSA0	I/O	DRAM chip select (rank 0)
DDRm_CSA1	I/O	DRAM chip select (rank 1)
DDRm_CAA2	I/O	DRAM command/address
DDRm_CAA3	I/O	DRAM command/address
DDRm_CAA4	I/O	DRAM command/address
DDRm_CAA5	I/O	DRAM command/address
DDRm_DQA0	I/O	DRAM data
DDRm_DQA1	I/O	DRAM data
DDRm_DQA2	I/O	DRAM data
DDRm_DQA3	I/O	DRAM data
DDRm_DQA4	I/O	DRAM data
DDRm_DQA5	I/O	DRAM data
DDRm_DQA6	I/O	DRAM data
DDRm_DQA7	I/O	DRAM data
DDRm_DMIA0	I/O	DRAM data mask inversion
DDRm_DQSAT0	I/O	DRAM data strobe (positive)
DDRm_DQSAC0	I/O	DRAM data strobe (negative)
DDRm_DQA8	I/O	DRAM data
DDRm_DQA9	I/O	DRAM data
DDRm_DQA10	I/O	DRAM data
DDRm_DQA11	I/O	DRAM data
DDRm_DQA12	I/O	DRAM data
DDRm_DQA13	I/O	DRAM data
DDRm_DQA14	I/O	DRAM data
DDRm_DQA15	I/O	DRAM data
DDRm_DMIA1	I/O	DRAM data mask inversion
DDRm_DQSAT1	I/O	DRAM data strobe (positive)
DDRm_DQSAC1	I/O	DRAM data strobe (negative)
DDRm_CKEB0	I/O	DRAM clock enable (rank 0)
DDRm_CKEB1	I/O	DRAM clock enable (rank 1)
DDRm_CAB0	I/O	DRAM command/address
DDRm_CAB1	I/O	DRAM command/address
DDRm_CKBT	I/O	DRAM clock (positive)

Table 3.4-2 External Pins (2/2)

Pin Name	Input/Output	Description
DDRm_CKBC	I/O	DRAM clock (negative)
DDRm_CSB0	I/O	DRAM chip select (rank 0)
DDRm_CSB1	I/O	DRAM chip select (rank 1)
DDRm_CAB2	I/O	DRAM command/address
DDRm_CAB3	I/O	DRAM command/address
DDRm_CAB4	I/O	DRAM command/address
DDRm_CAB5	I/O	DRAM command/address
DDRm_DQB0	I/O	DRAM data
DDRm_DQB1	I/O	DRAM data
DDRm_DQB2	I/O	DRAM data
DDRm_DQB3	I/O	DRAM data
DDRm_DQB4	I/O	DRAM data
DDRm_DQB5	I/O	DRAM data
DDRm_DQB6	I/O	DRAM data
DDRm_DQB7	I/O	DRAM data
DDRm_DMIB0	I/O	DRAM data mask inversion
DDRm_DQSBT0	I/O	DRAM data strobe (positive)
DDRm_DQSBC0	I/O	DRAM data strobe (negative)
DDRm_DQB8	I/O	DRAM data
DDRm_DQB9	I/O	DRAM data
DDRm_DQB10	I/O	DRAM data
DDRm_DQB11	I/O	DRAM data
DDRm_DQB12	I/O	DRAM data
DDRm_DQB13	I/O	DRAM data
DDRm_DQB14	I/O	DRAM data
DDRm_DQB15	I/O	DRAM data
DDRm_DMIB1	I/O	DRAM data mask inversion
DDRm_DQSBT1	I/O	DRAM data strobe (positive)
DDRm_DQSBC1	I/O	DRAM data strobe (negative)

Note: m = 0

3.4.2 Usage Notes

This unit has the following restrictions.

Table 3.4-3 Restrictions

Item	Restriction
Period of DRAM inaccessibility	It has the period of DRAM inaccessibility to execute the periodic training. It is 1,280 core clock cycles every 134,217,728 core clock cycles. So, it depends only on the Bit Rate (the frequency of the core clock). It is below: 3200 Mbps: 1.6 µs every 167 ms
Density when ECC function is enabled	When the ECC function is enabled, 1/8 is used for the ECC code, so the usable density is 7/8.
LPDDR4	Not support the followings; Refresh Management (RFM), Scaling Level (SCL).
LPDDR4X	Not support the followings; Single-ended mode for Clock and Strobe.

SECTION 3 MEMORY

3.5 TrustZone Address Space Controller (TZC)

3.5.1 Overview

This LSI has nine address space controllers (TZCs) to realize memory access in a safe area. It performs security checks on transactions to memory or peripherals. Transactions must meet security requirements to access memory or peripherals. This is an IP (“CoreLink™ TrustZone Address Space Controller TZC-400”) provided by Arm, for details on the functions of TZC-400, see the relevant Technical Reference Manual.

3.5.1.1 Features

The TZC works between TrustZone system ACE-Lite masters and ACE-Lite slaves to filter bus access from master to slave. It performs filtering based on the security requirements specified in the address space.

Each TZC is installed on the interface of the following modules of this LSI.

- xSPI
- SRAM0
- RCPU Bus
- DDR
- PCIE

For details on the internal bus, refer to **1.7 Internal Bus**.

The TZC provides the following key features:

- The ability to define up to eight address regions in the area map.
- A default base region to cover all remaining portions of the address map.
- Software programmable security access permissions for each address region through an APB interface. This includes the default base region, Region 0.
- Filter units only allow data transfer between an ACE-Lite master and an ACE-Lite slave if the security status of the ACE-Lite transaction and its identity match the security settings of the memory region it addresses.
- Common region configuration register settings shared between multiple filter units.
- Normal path for accesses with a much higher outstanding access support.
- Identity-based filtering of Non-secure accesses.
- Reporting and interrupt signaling that is configurable from software to manage failed permission checks. You can program the TZC to assert TZCINT_* (* = XSP, MSRM, ACRCB, DDR00, DDR01, PCI) when an access fails its security check.
- Gate keeper to allow or block accesses to each filter unit.
- Support for 256 outstanding transactions on the normal paths.

3.5.1.2 Block Diagram

Figure 3.5-1 shows a block diagram.

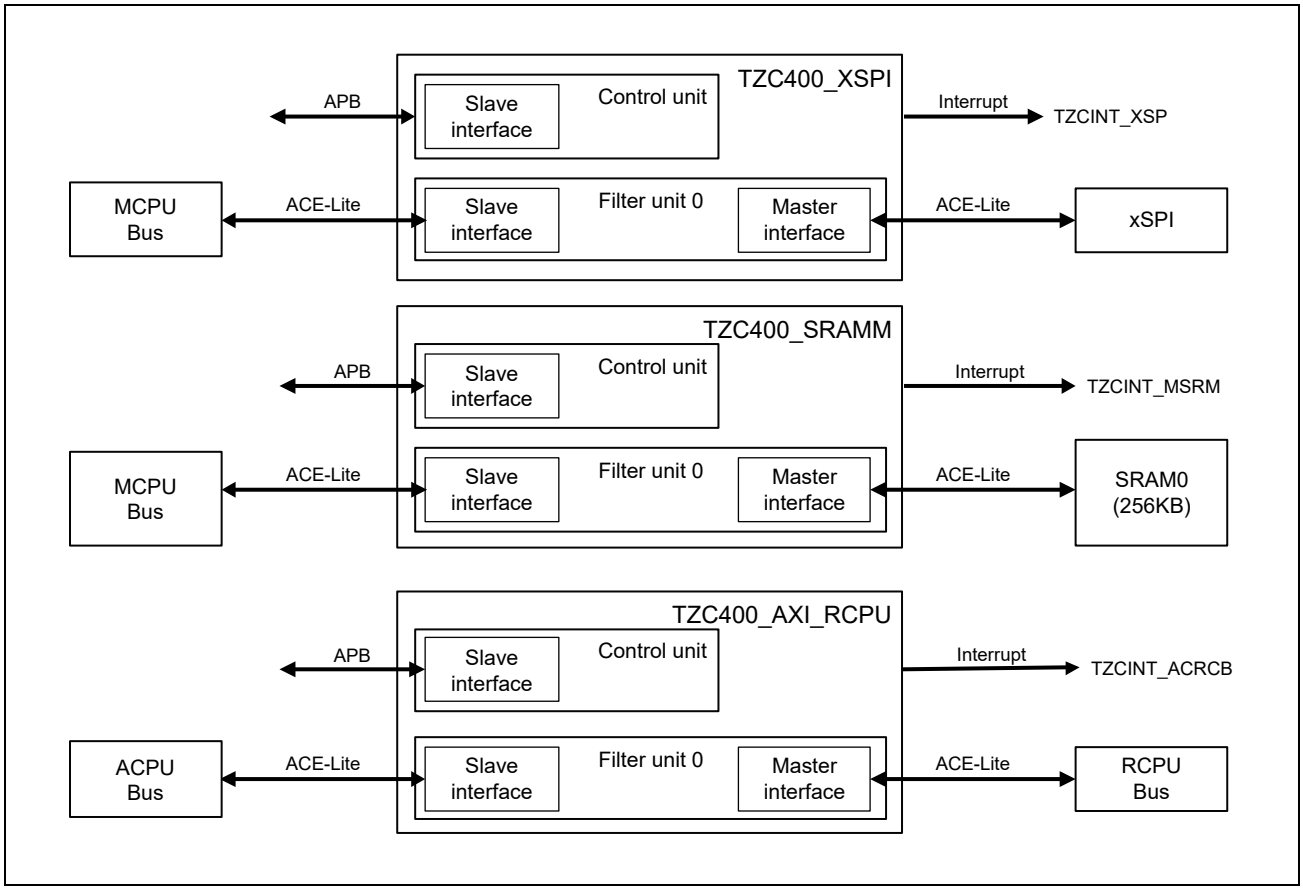


Figure 3.5-1 Block Diagram (1/3)

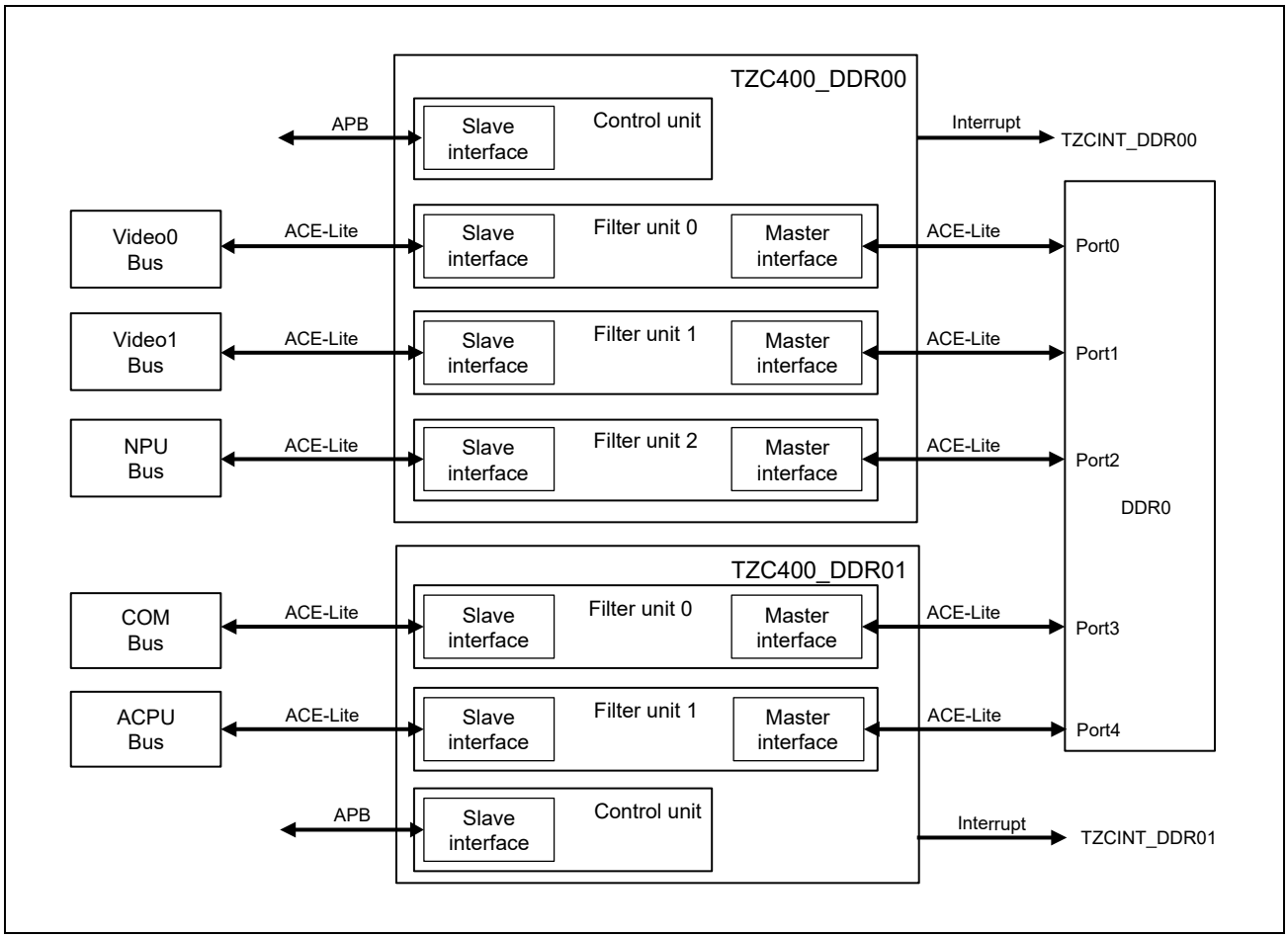


Figure 3.5-1 Block Diagram (2/3)

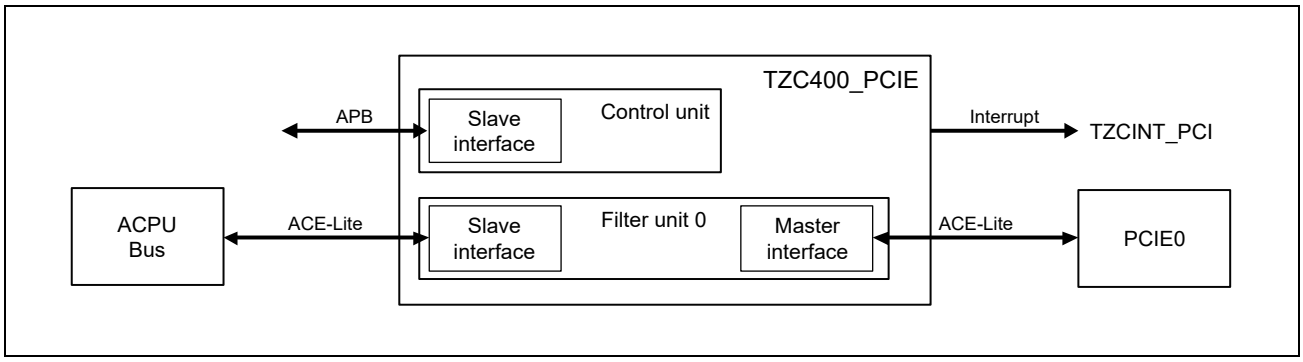


Figure 3.5-1 Block Diagram (3/3)

Filter units perform security checks. Each filter unit has an ACE-Lite slave interface and an ACE-Lite master interface. All filter units operate from one set of shared region configuration registers. This ensures consistency across all filter units.

The number of filter units of TZC installed on each interface is as follows.

- xSPI: 1 filter unit
- SRAM0: 1 filter unit
- RCPU bus: 1 filter unit
- DDR: 1 TZC, 3 filter units and 2 filter units
  - filter unit 0: Access from the Video0 Bus passes
  - filter unit 1: Access from the Video1 Bus passes
  - filter unit 2: Access from the NPU Bus passes
  - filter unit 0: Access from the COM Bus passes
  - filter unit 1: Access from the ACPU Bus passes
- PCIE: 1 filter unit

3.5.1.3 External Pins

In the TZC, there are no external pins.

3.5.2 Register Configuration

The base address of the TZC associated with each module is as follows:

- TZC400_XSPI: xSPI

  0_1047_0000h (CA55 Address Space)
  4047_0000h (CM33 Address Space Secure)
  5047_0000h (CM33 Address Space Non-Secure)

- TZC400_SRAMM: SRAM0 (256 KB)

  0_1046_0000h (CA55 Address Space)
  4046_0000h (CM33 Address Space Secure)
  5046_0000h (CM33 Address Space Non-Secure)

- TZC400_AXI_RCPU: RCPU Bus

  0_13CA_0000h (CA55 Address Space)
  43CA_0000h (CM33 Address Space Secure)
  53CA_0000h (CM33 Address Space Non-Secure)

- TZC400_DDR00: DDR0 (Video0, Video1, NPU Bus)

  0_13C6_0000h (CA55 Address Space)
  43C6_0000h (CM33 Address Space Secure)
  53C6_0000h (Cortex-M33 Address Space Non-Secure)

- TZC400_DDR01: DDR0 (COM, ACPU Bus)

  0_13C7_0000h (CA55 Address Space)
  43C7_0000h (CM33 Address Space Secure)
  53C7_0000h (CM33 Address Space Non-Secure)

- TZC400_PCIE: PCIE

  0_13CC_0000h (CA55 Address Space)
  43CC_0000h (CM33 Address Space Secure)
  53CC_0000h (CM33 Address Space Non-Secure)

For details on the functions of TZC-400, see the relevant Technical Reference Manual. (ARM® CoreLink™ TZC-400 TrustZone® Address Space Controller Technical Reference Manual, Revision: r0p1)

3.5.3 Register Descriptions

For details on the functions of TZC-400, see the relevant Technical Reference Manual. (ARM® CoreLink™ TZC-400 TrustZone® Address Space Controller Technical Reference Manual, Revision: r0p1)

The TZC has registers to specify the address area where access is controlled. Refer to Overall Address Space (**1.8 Address Map**) for the addresses specified in these registers.

3.5.4 Operation

For details on the functions of TZC-400, see the relevant Technical Reference Manual. (ARM® CoreLink™ TZC-400 TrustZone® Address Space Controller Technical Reference Manual, Revision: r0p1)

When accessing the bus using this unit, check the settings of SYS, CPG, and each TZC.

3.5.5 Usage Note

For details on the functions of TZC-400, see the relevant Technical Reference Manual. (ARM® CoreLink™ TZC-400 TrustZone® Address Space Controller Technical Reference Manual, Revision: r0p1)

SECTION 4 SYSTEM

4.1 System Overview

This section describes the units that control the overall system of this LSI.

■ Pin Function Controller (PFC) (See 4.2)

Controls the external pins (GPIO control, switching of multi-function pins, setting of drive strength, etc.) of this LSI.

■ System Controller (SYS) (See 4.3)

Controls each unit (TZC control, address extension, setting of each unit, etc.).

■ Clock Pulse Generator (CPG) (See 4.4)

Performs clock/reset control, boot control, and low power control for each unit.

■ Power Management Unit (PMU) (See 4.5)

The power supply of this LSI is roughly classified into two power supply domains (PD_AWO, PD_OTHERS).

Controls each power domain and enable signals of various power supplies.

■ Interrupt Controller (See 4.6)

The Interrupt Control Uniti (ICU) manages and controls interrupt signals.

In addition, the ICU is equipped with an Event Link Controller, and by outputting each event signal to an arbitrary unit, the load on the CPU is reduced.

In this section, GIC and MHU function details will also be described.

■ DMA Controller (DMAC) (See 4.7)

The LSI contains a total of 80 channels of DMAC. See the ICU description for DMAC allocation.

■ Renesas Secure IP (RSIP-E50B) (See 4.8)

Controls security of this LSI. This unit is an optional function.

■ Debug Interface (See 4.9)

This LSI equips Arm® CoreSight™. The CST unit controls the debugger connection.

SECTION 4 SYSTEM

4.2 Pin Function Controller (PFC)

This section describes the functions of the PFC (IO_TOP).

4.2.1 Overview

The PFC unit serves as the interface between the external pins of the LSI and the given units within the LSI via the IO block (1.2-V to 1.8-V/1.8-V/3.3-V I/O buffers and 1.2-V to 1.8-V OSC buffers) and also controls the mode of the IO block.

The external pins of the LSI are classified into dedicated pins and pins having multiplexed functions (multiplexed pins), and the PFC controls selection from among the multiplexed functions or of operation as general-purpose I/O (GPIO) port pins.

The multiplexed pins are assigned to the external pins of the LSI (P00 to PS3) as described in **1.2 Pin**. For the names of the multiplexed signals (functions) which can be selected for the respective multiplexed pins, refer to the list of multiplexed pins in **1.2 Pin**.

CAUTION

When making the settings, do not select the same multiplexed function for several different external pins.

Table 4.2-1 Overview

Function	Description
Pin function controller	• GPIO control • Switching between functions multiplexed on the pins • Switching the drive strength of the pins • Switching the slew rate of the pins • Pull-up/down control of the pins • Input enable control of the pins • Output enable control of the pins • N-ch. open drain control of the pins • Schmitt control of the pins • OSC mode switching • Reset latch function • Digital noise filter control
Event link controller	• ELC_GPIO control

4.2.1.1 GPIO Control and Switching between Functions Multiplexed on the Pins

The PFC supports general-purpose I/O (GPIO) port pins.

The external pins have the following functions.

- GPIO output value control (PFC_P_mn register)
- GPIO input/output enable (PFC_PM_mn register)
- GPIO input pin monitoring (PFC_PIN_mn register)

Additionally, the signals to be connected to an I/O buffer can be switched to GPIO operation or to any of the available settings from among mode 1 to mode 15.

- Mode switching (PFC_PMC_mn register)
- Function switching (PFC_PFC_mn register)

The table below shows the relationship between port mode (GPIO) and control mode (multiplexed function) and the PFC_P_mn, PFC_PM_mn, PFC_PMC_mn, and PFC_PFC_mn registers as port control registers. For details of each register, see **4.2.2 Register Description**.

Table 4.2-2 GPIO Control and Switching between Functions Multiplexed on the Pins

	PMC_mn = 0 (Initial Value)*1 (Port Mode)				PMC_mn = 1 (Control Mode)	
External Pin	PM_mn = 11b	PM_mn = 10b	PM_mn = 01b	PM_mn = 00b (Initial Value)	PFC_mn = 0000b (Initial Value)	PFC_mn = 0001b to 1111b
Multiplexed pins	P_mn is output. Input enabled	P_mn is output. Input disabled	External pin input	Hi-Z	Hi-Z	Mode 1 to 15

Note 1. Regarding PS*, PJ* and PM* ports, the initial value is 1.

4.2.1.2 Switching and Controlling the Pins

The PFC has functionality for switching and controlling the signals to be connected to the I/O buffers.

The functionality for switching and controlling the signals consists of the following items. For details of each register, see **4.2.2 Register Description**.

- Switching the drive strength of the pins (PFC_IOLH_mn register)
- Switching the slew rate of the pins (PFC_SR_mn register)
- Pull-up/down control of the pins (PFC_PUPD_mn register)
- Input enable control of the pins (PFC_IEN_mn register)
- Output enable control of the pins (PFC_OEN_mn register)
- N-ch. open drain control of the pins (PFC_NOD_mn register)
- Schmitt control of the pins (PFC_SMT_mn register)

4.2.1.3 OSC Mode Switching

The PFC has functionality for internally handling switching the mode of the main oscillator (between crystal oscillator mode and external clock reception mode). For details of each register, see **4.2.2 Register Description**.

- OSC mode switching register (PFC_OSCBYPS register)

The targets for control by the PFC_OSCBYPS register are the RTC oscillator and audio oscillator.

For the main oscillator, the mode is switched by the signal on the external QBYPASS pin, instead of the OSCBYPS register.

4.2.1.4 Reset Latch Function

The PFC has a reset latch function for the mode pins (configuration pins).

The latches retain the values of the mode pins at the time of a system reset (QRESN = low). For details, see **4.2.3.5 RST_LATCH**.

The targets of retention by the reset latch are as follows.

- MD_BOOT0
- MD_BOOT1
- MD_BOOT2
- MD_BOOT3
- MD_BOOT4
- BOOTSELCPU
- BOOTPLLCA0
- BOOTPLLCA1
- MD_CLKS
- QBYPASS
- QRESNSEL

4.2.1.5 Digital Noise Filter Control

The PFC has a digital noise filter function, which allows the blocking of transient pulses such as those which are input from an external pin due to noise. For details of each register, see **4.2.2 Register Description**.

The targets for digital noise filter control are as follows.

- Dedicated NMI pin
- Input direction line for GPIO
- External TINT interrupt
- Multiplexed IRQ function
- Multiplexed DREQ function
- Port event control pins (the targets for PFC_ELC_GPIO register control described in **4.2.1.6 Event Link Controller**)

The digital noise filter control has the following functions and these can be set by registers.

- Use or non-use of the filter (PFC_FILONOFF register)
- Number of stages of the filter (PFC_FILNUM register)
- Sampling interval of the filter (PFC_FILCLKSEL register)

Remark The above registers include those for NMI, GPIO, and TINT, for IRQ, and for DREQ.

The digital noise filter settings for the target pins for port event control are fixed to the following.

- Use or non-use of the filter: Use
- Number of stages of the filter: 3
- Sampling interval of the filter: 1/1 (no division)

4.2.1.6 Event Link Controller

The PFC has PFC_ELC_GPIO registers for port event control. For details of each register, see **4.2.2 Register Description**. For the targets for control by the PFC_ELC_GPIO registers, see **Table 4.2-5**.

The PFC_ECL_GPIO registers has the following functions.

- Port group specification (PFC_ELC_PGRg register)
- Port group control (PFC_ELC_PGCg register)
- Port buffer (PFC_ELC_PDBFg register)
- Event connection port specification (PFC_ELC_PELs register)
- Input edge detection control (PFC_ELC_DPTC register)
- Port event control (ELC_ELSR2 register)

4.2.1.7 Functional Block Diagram

Figure 4.2-1 is a functional block diagram of the PFC.

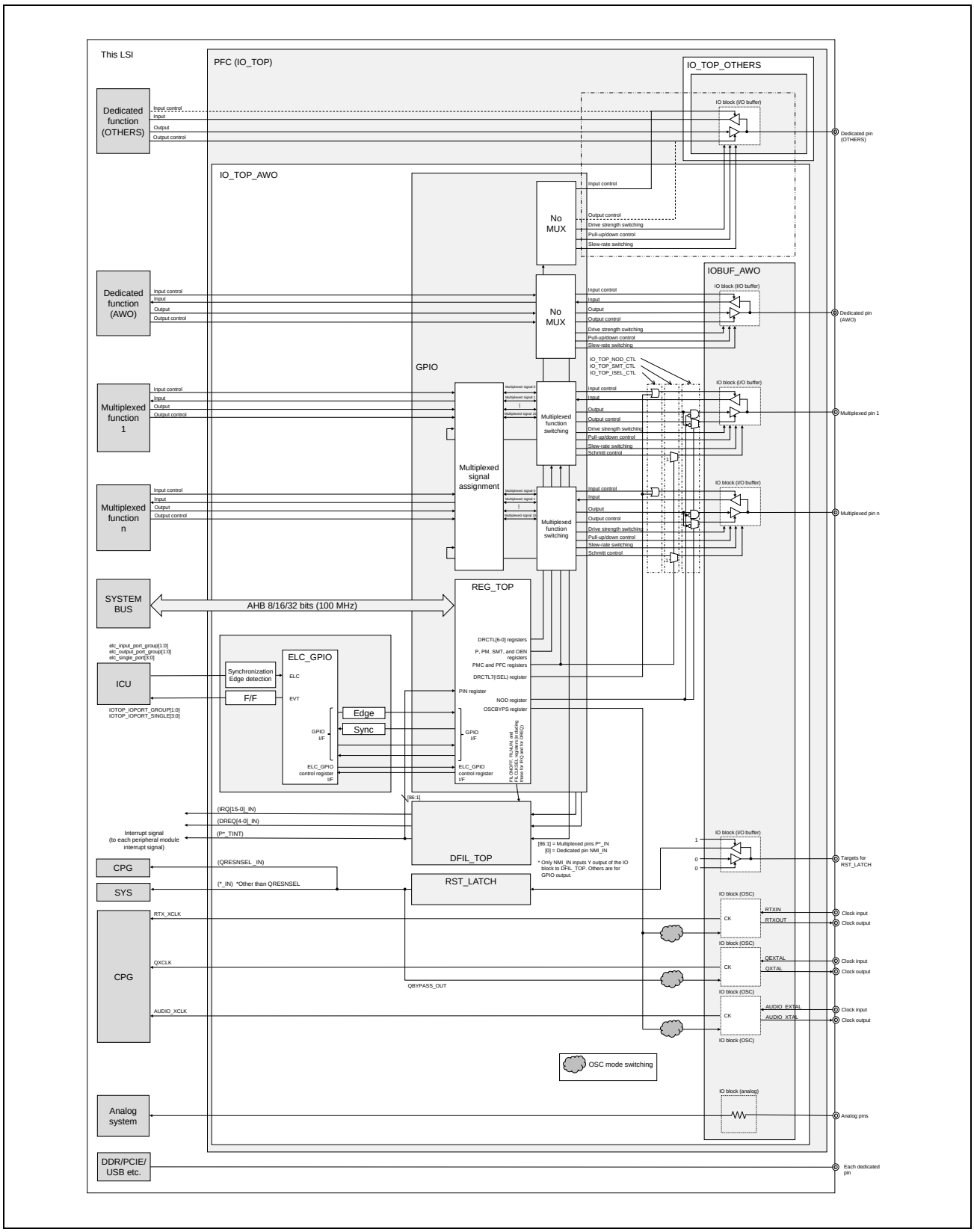


Figure 4.2-1 Functional Block Diagram

4.2.2 Register Description

The base address of the PFC register is shown below.

Table 4.2-3 Register Base Addresses

Base Address Name	Base Address
<PFC_base>	0_1041_0000h (5041_0000h*1, 4041_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

4.2.2.1 List of Registers

The table below lists the registers.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Port Registers	PFC_P_mn	00h	0020h	8
Port Mode Registers	PFC_PM_mn	0000h	0140h	8, 16
Port Mode Control Registers	PFC_PMC_mn	xxh	0220h	8
Port Function Control Registers	PFC_PFC_mn	0000_0000h	0480h	8, 16, 32
Port Pin Input Registers	PFC_PIN_mn	xxh	0820h	8
IOLH Switching Registers	PFC_IOLH_mn	0x0x_0x0xh	1018h	8, 16, 32
Slew-Rate Switching Registers	PFC_SR_mn	0x0x_0x0xh	1418h	8, 16, 32
IEN Switching Registers	PFC_IEN_mn	0x0x_0x0xh	1818h	8, 16, 32
PU/PD Switching Registers	PFC_PUPD_mn	0x0x_0x0xh	1C28h	8, 16, 32
Digital Noise Filter (FILONOFF) Registers	PFC_FILONOFF_mn	0000_0000h	2008h	8, 16, 32
Digital Noise Filter (FILNUM) Registers	PFC_FILNUM_mn	0000_0000h	2408h	8, 16, 32
Digital Noise Filter (FILCLKSEL) Registers	PFC_FILCLKSEL_mn	0000_0000h	2808h	8, 16, 32
ISEL Control Registers	PFC_ISEL_mn	0000_0000h	2D00h	8, 16, 32
N-ch Open Drain Control Registers	PFC_NOD_mn	0000_0x0xh	3028h	8, 16, 32
Schmitt Control Registers	PFC_SMT_mn	0000_0000h	3500h	8, 16, 32
Port Group Specification Registers	PFC_ELC_PGRg	00h	3800h	8
Port Group Control Registers	PFC_ELC_PGCg	88h	3802h	8
Port Buffer Registers	PFC_ELC_PDBFg	00h	3804h	8
Event Connection Port Specification Registers	PFC_ELC_PEL	80h	380Ch	8
Input Edge Detection Control Register	PFC_ELC_DPTC	00h	3810h	8
Port Event Control Register	PFC_ELC_ELSR2	00h	3811h	8
Reserve	-	-	3812h to 3BFFh	-
OSC Bypass Mode Switching Register	PFC_OSCBYPS	000C_0303h	3C00h	8, 16, 32
Write Protect Register	PFC_PWPR	0000_00A0h	3C04h	8, 16, 32
IRQ Digital Noise Filter (FILONOFF) Register	PFC_FILONOFF_IRQ	0000_0000h	3C08h	8, 16, 32
IRQ Digital Noise Filter (FILNUM) Register	PFC_FILNUM_IRQ	0000_0000h	3C0Ch	8, 16, 32
IRQ Digital Noise Filter (FILCLKSEL) Register	PFC_FILCLKSEL_IRQ	0000_0000h	3C10h	8, 16, 32
DMAC_REQ Digital Noise Filter (FILONOFF) Register	PFC_FILONOFF_DMAC_REQ	0000_0000h	3C14h	8, 16, 32
DMAC_REQ Digital Noise Filter (FILNUM) Register	PFC_FILNUM_DMAC_REQ	0000_0000h	3C18h	8, 16, 32
DMAC_REQ Digital Noise Filter (FILCLKSEL) Register	PFC_FILCLKSEL_DMAC_REQ	0000_0000h	3C1Ch	8, 16, 32
NMI Digital Noise Filter (FILONOFF) Register	PFC_FILONOFF_NMI	0000_0000h	3C20h	8, 16, 32
NMI Digital Noise Filter (FILNUM) Register	PFC_FILNUM_NMI	0000_0000h	3C24h	8, 16, 32
NMI Digital Noise Filter (FILCLKSEL) Register	PFC_FILCLKSEL_NMI	0000_0000h	3C28h	8, 16, 32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	3C2Ch to 3C3Fh	-
OEN Switching Register	PFC_OEN	0000_003Fh	3C40h	8, 16, 32

For the correspondence between ports and the respective control registers, see the table below.

Table 4.2-4 Correspondence between Ports and Registers (Dedicated Pins) (1/2)

External Pin	IOLH_mn	SR_mn	IEN_mn	PUPD_mn	NOD_mn
QRESN	—	—	—	—	—
QBYPASS	—	—	—	—	—
BSCANP	—	—	—	—	—
MD_BOOT3	—	—	—	—	—
QRESNSEL	—	—	—	—	—
PWEN0	—	—	—	—	—
PWEN1	—	—	—	—	—
PWEN2	—	—	—	—	—
MD_BOOT0	—	—	—	—	—
MD_BOOT1	—	—	—	—	—
MD_BOOT2	—	—	—	—	—
BOOTSELCPU	—	—	—	—	—
MD_CLKS	—	—	—	—	—
RTXIN	—	—	—	—	—
RTXOUT	—	—	—	—	—
QEXTAL	—	—	—	—	—
QXTAL	—	—	—	—	—
AUDIO_EXTAL	—	—	—	—	—
AUDIO_XTAL	—	—	—	—	—
WDTUDFCA	✓	✓	—	✓	✓
WDTUDFCM	✓	✓	—	✓	✓
SCIF_RXD	✓	✓	—	✓	—
SCIF_TXD	✓	✓	—	✓	—
ANI000	—	—	—	—	—
ANI001	—	—	—	—	—
ANI002	—	—	—	—	—
ANI003	—	—	—	—	—
ANI004	—	—	—	—	—
ANI005	—	—	—	—	—
ANI006	—	—	—	—	—
ANI007	—	—	—	—	—
SD0CLK	✓	✓	—	—	—
SD0CMD	✓	✓	✓	✓	—
SD0DAT0	✓	✓	✓	✓	—
SD0DAT1	✓	✓	✓	✓	—
SD0DAT2	✓	✓	✓	✓	—
SD0DAT3	✓	✓	✓	✓	—
SD0DAT4	✓	✓	✓	✓	—
SD0DAT5	✓	✓	✓	✓	—
SD0DAT6	✓	✓	✓	✓	—
SD0DAT7	✓	✓	✓	✓	—

Table 4.2-4 Correspondence between Ports and Registers (Dedicated Pins) (2/2)

External Pin	IOLH_mn	SR_mn	IEN_mn	PUPD_mn	NOD_mn
SD0PWEN	✓	✓	—	—	—
SD0IOVS	✓	✓	—	—	—
SD0RSTN	✓	✓	—	—	—

Table 4.2-5 Correspondence between Ports and Registers (Multiplexed Pins) (1/4)

External Pin	P_mn	PM_mn	PMC_mn	PFC_mn	PIN_mn	IOLH_mn	SR_mn	IEN_mn	PUPD_mn	FILONOFF_mn FILNUM_mn FILCLKSEL_mn	ISEL_mn	NOD_mn	SMT_mn	ELC_*
P00	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P01	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P02	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P03	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P04	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P05	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P06	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P07	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P10	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P11	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P12	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P13	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P14	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P15	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P16	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P17	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P20	✓	✓	✓	✓	✓	✓	—	—	✓	✓	✓	✓	—	—
P21	✓	✓	✓	✓	✓	✓	—	—	✓	✓	✓	✓	—	—
P30	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P31	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P32	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P33	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P34	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P35	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P36	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P37	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P40	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P41	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P42	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P43	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P44	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P45	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P50	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P51	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P52	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P53	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P54	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P55	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P56	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P60	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P61	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P62	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P63	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P64	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—

Table 4.2-5 Correspondence between Ports and Registers (Multiplexed Pins) (2/4)

External Pin	P_mn	PM_mn	PMC_mn	PFC_mn	PIN_mn	IOLH_mn	SR_mn	IEN_mn	PUPD_mn	FILONOFF_mn FILNUM_mn FILCLKSEL_mn	ISEL_mn	NOD_mn	SMT_mn	ELC_*
P65	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P66	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P70	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P71	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P72	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P73	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P74	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P75	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P76	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P77	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓
P80	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P81	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P82	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P83	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P84	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
P85	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PA0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PA1	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PA2	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PA3	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PA4	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PA5	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PA6	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PA7	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PB0	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PB1	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PB2	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PB3	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PB4	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PB5	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PB6	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PB7	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PC0	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PC1	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PC2	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PD0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PD1	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PD2	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PD3	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PD4	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PD5	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PD6	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PD7	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—

Table 4.2-5 Correspondence between Ports and Registers (Multiplexed Pins) (3/4)

External Pin	P_mn	PM_mn	PMC_mn	PFC_mn	PIN_mn	IOLH_mn	SR_mn	IEN_mn	PUPD_mn	FILONOFF_mn FILNUM_mn FILCLKSEL_mn	ISEL_mn	NOD_mn	SMT_mn	ELC_*
PE0	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PE1	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PE2	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PE3	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PE4	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PE5	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PE6	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PE7	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PF0	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PF1	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PF2	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PG0	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PG1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PG2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PG3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PG4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PG5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PG6	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PG7	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PH0	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PH1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PH2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PH3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PH4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PH5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PJ0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—
PJ1	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PJ2	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PJ3	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PJ4	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PK0	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PK1	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PK2	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PK3	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PL0	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PL1	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PL2	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PL3	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PL4	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PL5	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PL6	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PL7	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PM0	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PM1	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—

Table 4.2-5 Correspondence between Ports and Registers (Multiplexed Pins) (4/4)

External Pin	P_mn	PM_mn	PMC_mn	PFC_mn	PIN_mn	IOLH_mn	SR_mn	IEN_mn	PUPD_mn	FILONOFF_mn FILNUM_mn FILCLKSEL_mn	ISEL_mn	NOD_mn	SMT_mn	ELC_*
PM2	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PM3	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PM4	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PM5	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PM6	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PM7	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PS0	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PS1	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PS2	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—
PS3	✓	✓	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	—

4.2.2.2 Port Registers (PFC_P_mn)

These registers set output values on the GPIO pins. For the correspondence with the ports, see **Table 4.2-6**.

Access Size : 8 bits
Offset Address : <PFC_base> + 0020h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	P_m7	P_m6	P_m5	P_m4	P_m3	P_m2	P_m1	P_m0
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	P_m7	0h	RW	Sets the value of the GPIO pin. 0b: Outputs low 1b: Outputs high
6	P_m6	0h	RW	Sets the value of the GPIO pin. 0b: Outputs low 1b: Outputs high
5	P_m5	0h	RW	Sets the value of the GPIO pin. 0b: Outputs low 1b: Outputs high
4	P_m4	0h	RW	Sets the value of the GPIO pin. 0b: Outputs low 1b: Outputs high
3	P_m3	0h	RW	Sets the value of the GPIO pin. 0b: Outputs low 1b: Outputs high
2	P_m2	0h	RW	Sets the value of the GPIO pin. 0b: Outputs low 1b: Outputs high
1	P_m1	0h	RW	Sets the value of the GPIO pin. 0b: Outputs low 1b: Outputs high
0	P_m0	0h	RW	Sets the value of the GPIO pin. 0b: Outputs low 1b: Outputs high

Table 4.2-6 Corresponding Pins and Offset Addresses of PFC_P_mn

Offset Address	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0020h	PFC_P_20	P07	P06	P05	P04	P03	P02	P01	P00
0021h	PFC_P_21	P17	P16	P15	P14	P13	P12	P11	P10
0022h	PFC_P_22	—	—	—	—	—	—	P21	P20
0023h	PFC_P_23	P37	P36	P35	P34	P33	P32	P31	P30
0024h	PFC_P_24	—	—	P45	P44	P43	P42	P41	P40
0025h	PFC_P_25	—	P56	P55	P54	P53	P52	P51	P50
0026h	PFC_P_26	—	P66	P65	P64	P63	P62	P61	P60
0027h	PFC_P_27	P77	P76	P75	P74	P73	P72	P71	P70
0028h	PFC_P_28	—	—	P85	P84	P83	P82	P81	P80
002Ah	PFC_P_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
002Bh	PFC_P_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
002Ch	PFC_P_2C	—	—	—	—	—	PC2	PC1	PC0
002Dh	PFC_P_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
002Eh	PFC_P_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
002Fh	PFC_P_2F	—	—	—	—	—	PF2	PF1	PF0
0030h	PFC_P_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
0031h	PFC_P_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
0033h	PFC_P_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
0034h	PFC_P_34	—	—	—	—	PK3	PK2	PK1	PK0
0035h	PFC_P_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
0036h	PFC_P_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
003Ch	PFC_P_3C	—	—	—	—	PS3	PS2	PS1	PS0

4.2.2.3 Port Mode Registers (PFC_PM_mn)

These registers set the input/output of the GPIO pins. For the correspondence with the ports, see **Table 4.2-7**.

Access Size : 8, 16 bits
Offset Address : <PFC_base> + 0140h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PM_m7[1:0]		PM_m6[1:0]		PM_m5[1:0]		PM_m4[1:0]		PM_m3[1:0]		PM_m2[1:0]		PM_m1[1:0]		PM_m0[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 14	PM_m7[1:0]	0h	RW	Sets the input/output of the GPIO pin. 00b: Hi-Z 01b: Input mode 10b: Output mode (input disabled) 11b: Output mode (input enabled)
13 to 12	PM_m6[1:0]	0h	RW	Sets the input/output of the GPIO pin. 00b: Hi-Z 01b: Input mode 10b: Output mode (input disabled) 11b: Output mode (input enabled)
11 to 10	PM_m5[1:0]	0h	RW	Sets the input/output of the GPIO pin. 00b: Hi-Z 01b: Input mode 10b: Output mode (input disabled) 11b: Output mode (input enabled)
9 to 8	PM_m4[1:0]	0h	RW	Sets the input/output of the GPIO pin. 00b: Hi-Z 01b: Input mode 10b: Output mode (input disabled) 11b: Output mode (input enabled)
7 to 6	PM_m3[1:0]	0h	RW	Sets the input/output of the GPIO pin. 00b: Hi-Z 01b: Input mode 10b: Output mode (input disabled) 11b: Output mode (input enabled)
5 to 4	PM_m2[1:0]	0h	RW	Sets the input/output of the GPIO pin. 00b: Hi-Z 01b: Input mode 10b: Output mode (input disabled) 11b: Output mode (input enabled)
3 to 2	PM_m1[1:0]	0h	RW	Sets the input/output of the GPIO pin. 00b: Hi-Z 01b: Input mode 10b: Output mode (input disabled) 11b: Output mode (input enabled)
1 to 0	PM_m0[1:0]	0h	RW	Sets the input/output of the GPIO pin. 00b: Hi-Z 01b: Input mode 10b: Output mode (input disabled) 11b: Output mode (input enabled)

Table 4.2-7 Corresponding Pins and Offset Addresses of PFC_PM_mn

Offset Address	Register Name	Bits 15 and 14	Bits 13 and 12	Bits 11 and 10	Bits 9 and 8	Bits 7 and 6	Bits 5 and 4	Bits 3 and 2	Bits 1 and 0
0140h	PFC_PM_20	P07	P06	P05	P04	P03	P02	P01	P00
0142h	PFC_PM_21	P17	P16	P15	P14	P13	P12	P11	P10
0144h	PFC_PM_22	—	—	—	—	—	—	P21	P20
0146h	PFC_PM_23	P37	P36	P35	P34	P33	P32	P31	P30
0148h	PFC_PM_24	—	—	P45	P44	P43	P42	P41	P40
014Ah	PFC_PM_25	—	P56	P55	P54	P53	P52	P51	P50
014Ch	PFC_PM_26	—	P66	P65	P64	P63	P62	P61	P60
014Eh	PFC_PM_27	P77	P76	P75	P74	P73	P72	P71	P70
0150h	PFC_PM_28	—	—	P85	P84	P83	P82	P81	P80
0154h	PFC_PM_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
0156h	PFC_PM_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
0158h	PFC_PM_2C	—	—	—	—	—	PC2	PC1	PC0
015Ah	PFC_PM_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
015Ch	PFC_PM_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
015Eh	PFC_PM_2F	—	—	—	—	—	PF2	PF1	PF0
0160h	PFC_PM_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
0162h	PFC_PM_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
0166h	PFC_PM_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
0168h	PFC_PM_34	—	—	—	—	PK3	PK2	PK1	PK0
016Ah	PFC_PM_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
016Ch	PFC_PM_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
0178h	PFC_PM_3C	—	—	—	—	PS3	PS2	PS1	PS0

The PFC_PM_mn register settings when using the event link controller are as listed below. For details, refer to **4.6 Interrupt Controller** of the functional specification.

Table 4.2-8 Register Settings when ELC is Used

Operation	Relevant Interface between PFC and ICU	Direction of I/F	PFC_PM_mn Register Setting
(1) Single input port operation upon event generation	IOTOP_IOPORT_SINGLE[3:0]	Output	01b
(2) Single output port operation upon event input	elc_single_port[3:0]	Input	10b (or 11b*1)
(3) Input port group operation upon event generation	IOTOP_IOPORT_GROUP[1:0]	Output	01b
(4) Input port group operation upon event input	elc_input_port_group[1:0]	Input	01b
(5) Output port group operation upon event input	elc_output_port_group[1:0]	Input	10b (or 11b*2)

Note 1. 11b when the setting of ELC_PELs.PSM = 1xb (at the time of even input, toggled output).

Note 2. 11b when the setting of ELC_PGCg.PGCO = 010b (at the time of even input, toggled output).

4.2.2.4 Port Mode Control Registers (PFC_PMC_mn)

These registers set switching between the multiplexed pin functions and GPIO. The registers can be write-protected by the PFC_PWPR register. For the correspondence with the ports, see **Table 4.2-9**.

Access Size : 8 bits
Offset Address : <PFC_base> + 0220h
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	PMC_m7	PMC_m6	PMC_m5	PMC_m4	PMC_m3	PMC_m2	PMC_m1	PMC_m0
Initial Value	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	PMC_m7	x	RW	Sets switching between the multiplexed functions of the GPIO pins. 0b: Port mode (GPIO) 1b: Control mode (multiplexed function)
6	PMC_m6	x	RW	Sets switching between the multiplexed functions of the GPIO pins. 0b: Port mode (GPIO) 1b: Control mode (multiplexed function)
5	PMC_m5	x	RW	Sets switching between the multiplexed functions of the GPIO pins. 0b: Port mode (GPIO) 1b: Control mode (multiplexed function)
4	PMC_m4	x	RW	Sets switching between the multiplexed functions of the GPIO pins. 0b: Port mode (GPIO) 1b: Control mode (multiplexed function)
3	PMC_m3	x	RW	Sets switching between the multiplexed functions of the GPIO pins. 0b: Port mode (GPIO) 1b: Control mode (multiplexed function)
2	PMC_m2	x	RW	Sets switching between the multiplexed functions of the GPIO pins. 0b: Port mode (GPIO) 1b: Control mode (multiplexed function)
1	PMC_m1	x	RW	Sets switching between the multiplexed functions of the GPIO pins. 0b: Port mode (GPIO) 1b: Control mode (multiplexed function)
0	PMC_m0	x	RW	Sets switching between the multiplexed functions of the GPIO pins. 0b: Port mode (GPIO) 1b: Control mode (multiplexed function)

x: undefined value. It depends on the port number. Initial value of PS*, PJ*, PL* and PM* port is 1 and the value of the other ports is 0.

Note: When switching multiplexed pins in control mode or switching between control mode and port mode, start by setting PM_mn = 00b and PMC_mn = 0b to place the pins in the Hi-Z state and then set the other registers appropriately before switching the functions in order to avoid contention of the signals with those of other-party devices.

Table 4.2-9 Corresponding Pins and Offset Addresses of PFC_PMC_mn

Offset Address	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0220h	PFC_PMC_20	P07	P06	P05	P04	P03	P02	P01	P00
0221h	PFC_PMC_21	P17	P16	P15	P14	P13	P12	P11	P10
0222h	PFC_PMC_22	—	—	—	—	—	—	P21	P20
0223h	PFC_PMC_23	P37	P36	P35	P34	P33	P32	P31	P30
0224h	PFC_PMC_24	—	—	P45	P44	P43	P42	P41	P40
0225h	PFC_PMC_25	—	P56	P55	P54	P53	P52	P51	P50
0226h	PFC_PMC_26	—	P66	P65	P64	P63	P62	P61	P60
0227h	PFC_PMC_27	P77	P76	P75	P74	P73	P72	P71	P70
0228h	PFC_PMC_28	—	—	P85	P84	P83	P82	P81	P80
022Ah	PFC_PMC_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
022Bh	PFC_PMC_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
022Ch	PFC_PMC_2C	—	—	—	—	—	PC2	PC1	PC0
022Dh	PFC_PMC_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
022Eh	PFC_PMC_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
022Fh	PFC_PMC_2F	—	—	—	—	—	PF2	PF1	PF0
0230h	PFC_PMC_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
0231h	PFC_PMC_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
0233h	PFC_PMC_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
0234h	PFC_PMC_34	—	—	—	—	PK3	PK2	PK1	PK0
0235h	PFC_PMC_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
0236h	PFC_PMC_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
023Ch	PFC_PMC_3C	—	—	—	—	PS3	PS2	PS1	PS0

4.2.2.5 Port Function Control Registers (PFC_PFC_mn)

These registers set the multiplexed functions. The registers can be write-protected by the PFC_PWPR register. For the correspondence with the ports, see **Table 4.2-10**. For details of each mode, refer to **1.2 Pin**.

Access Size : 8, 16, 32 bits
Offset Address : <PFC_base> + 0480h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PFC_m7[3:0]				PFC_m6[3:0]				PFC_m5[3:0]				PFC_m4[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFC_m3[3:0]				PFC_m2[3:0]				PFC_m1[3:0]				PFC_m0[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	PFC_m7[3:0]	0h	RW	Sets the multiplexed function of the GPIO pin. 1111b to 0000b: Mode 15 to mode 0
27 to 24	PFC_m6[3:0]	0h	RW	Sets the multiplexed function of the GPIO pin. 1111b to 0000b: Mode 15 to mode 0
23 to 20	PFC_m5[3:0]	0h	RW	Sets the multiplexed function of the GPIO pin. 1111b to 0000b: Mode 15 to mode 0
19 to 16	PFC_m4[3:0]	0h	RW	Sets the multiplexed function of the GPIO pin. 1111b to 0000b: Mode 15 to mode 0
15 to 12	PFC_m3[3:0]	0h	RW	Sets the multiplexed function of the GPIO pin. 1111b to 0000b: Mode 15 to mode 0
11 to 8	PFC_m2[3:0]	0h	RW	Sets the multiplexed function of the GPIO pin. 1111b to 0000b: Mode 15 to mode 0
7 to 4	PFC_m1[3:0]	0h	RW	Sets the multiplexed function of the GPIO pin. 1111b to 0000b: Mode 15 to mode 0
3 to 0	PFC_m0[3:0]	0h	RW	Sets the multiplexed function of the GPIO pin. 1111b to 0000b: Mode 15 to mode 0

Table 4.2-10 Corresponding Pins and Offset Addresses of PFC_PFC_mn

Offset Address	Register Name	Bits 31 to 28	Bits 27 to 24	Bits 23 to 20	Bits 19 to 16	Bits 15 to 12	Bits 11 to 8	Bits 7 to 4	Bits 3 to 0
0480h	PFC_PFC_20	P07	P06	P05	P04	P03	P02	P01	P00
0484h	PFC_PFC_21	P17	P16	P15	P14	P13	P12	P11	P10
0488h	PFC_PFC_22	—	—	—	—	—	—	P21	P20
048Ch	PFC_PFC_23	P37	P36	P35	P34	P33	P32	P31	P30
0490h	PFC_PFC_24	—	—	P45	P44	P43	P42	P41	P40
0494h	PFC_PFC_25	—	P56	P55	P54	P53	P52	P51	P50
0498h	PFC_PFC_26	—	P66	P65	P64	P63	P62	P61	P60
049Ch	PFC_PFC_27	P77	P76	P75	P74	P73	P72	P71	P70
04A0h	PFC_PFC_28	—	—	P85	P84	P83	P82	P81	P80
04A8h	PFC_PFC_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
04ACh	PFC_PFC_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
04B0h	PFC_PFC_2C	—	—	—	—	—	PC2	PC1	PC0
04B4h	PFC_PFC_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
04B8h	PFC_PFC_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
04BCh	PFC_PFC_2F	—	—	—	—	—	PF2	PF1	PF0
04C0h	PFC_PFC_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
04C4h	PFC_PFC_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
04CCh	PFC_PFC_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
04D0h	PFC_PFC_34	—	—	—	—	PK3	PK2	PK1	PK0
04D4h	PFC_PFC_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
04D8h	PFC_PFC_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
04F0h	PFC_PFC_3C	—	—	—	—	PS3	PS2	PS1	PS0

4.2.2.6 Port Pin Input Registers (PFC_PIN_mn)

These are read-only registers that allow the monitoring of input values on the input pins. Writing to these registers has no effect. For the correspondence with the ports, see **Table 4.2-11**.

Access Size : 8 bits
Offset Address : <PFC_base> + 0820h
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	PIN_m7	PIN_m6	PIN_m5	PIN_m4	PIN_m3	PIN_m2	PIN_m1	PIN_m0
Initial Value	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	PIN_m7	x	R	The initial value "x" is a value of the Y pin of the IO block when the IE pin of the IO block = 0 (depends on the IO block type). Refer to **Figure 4.2-2**.
6	PIN_m6	x	R	The initial value "x" is a value of the Y pin of the IO block when the IE pin of the IO block = 0 (depends on the IO block type). Refer to **Figure 4.2-2**.
5	PIN_m5	x	R	The initial value "x" is a value of the Y pin of the IO block when the IE pin of the IO block = 0 (depends on the IO block type). Refer to **Figure 4.2-2**.
4	PIN_m4	x	R	The initial value "x" is a value of the Y pin of the IO block when the IE pin of the IO block = 0 (depends on the IO block type). Refer to **Figure 4.2-2**.
3	PIN_m3	x	R	The initial value "x" is a value of the Y pin of the IO block when the IE pin of the IO block = 0 (depends on the IO block type). Refer to **Figure 4.2-2**.
2	PIN_m2	x	R	The initial value "x" is a value of the Y pin of the IO block when the IE pin of the IO block = 0 (depends on the IO block type). Refer to **Figure 4.2-2**.
1	PIN_m1	x	R	The initial value "x" is a value of the Y pin of the IO block when the IE pin of the IO block = 0 (depends on the IO block type). Refer to **Figure 4.2-2**.
0	PIN_m0	x	R	The initial value "x" is a value of the Y pin of the IO block when the IE pin of the IO block = 0 (depends on the IO block type). Refer to **Figure 4.2-2**.

x: Undefined value

Table 4.2-11 Corresponding Pins and Offset Addresses of PFC_PIN_mn

Offset Address	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0820h	PFC_PIN_20	P07	P06	P05	P04	P03	P02	P01	P00
0821h	PFC_PIN_21	P17	P16	P15	P14	P13	P12	P11	P10
0822h	PFC_PIN_22	—	—	—	—	—	—	P21	P20
0823h	PFC_PIN_23	P37	P36	P35	P34	P33	P32	P31	P30
0824h	PFC_PIN_24	—	—	P45	P44	P43	P42	P41	P40
0825h	PFC_PIN_25	—	P56	P55	P54	P53	P52	P51	P50
0826h	PFC_PIN_26	—	P66	P65	P64	P63	P62	P61	P60
0827h	PFC_PIN_27	P77	P76	P75	P74	P73	P72	P71	P70
0828h	PFC_PIN_28	—	—	P85	P84	P83	P82	P81	P80
082Ah	PFC_PIN_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
082Bh	PFC_PIN_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
082Ch	PFC_PIN_2C	—	—	—	—	—	PC2	PC1	PC0
082Dh	PFC_PIN_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
082Eh	PFC_PIN_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
082Fh	PFC_PIN_2F	—	—	—	—	—	PF2	PF1	PF0
0830h	PFC_PIN_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
0831h	PFC_PIN_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
0833h	PFC_PIN_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
0834h	PFC_PIN_34	—	—	—	—	PK3	PK2	PK1	PK0
0835h	PFC_PIN_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
0836h	PFC_PIN_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
083Ch	PFC_PIN_3C	—	—	—	—	PS3	PS2	PS1	PS0

4.2.2.7 IOLH Switching Registers (PFC_IOLH_mn)

These registers set the drive strength of the IO block. For the correspondence with the ports, see **Table 4.2-12** and **Table 4.2-13**.

Access Size : 8, 16, 32 bits
Offset Address : <PFC_base> + 1018h
Initial Value : 0x0x_0x0xh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	IOLH_m3[1:0]		-	-	-	-	-	-	IOLH_m2[1:0]	
Initial Value	0	0	0	0	0	0	x	x	0	0	0	0	0	0	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	IOLH_m1[1:0]		-	-	-	-	-	-	IOLH_m0[1:0]	
Initial Value	0	0	0	0	0	0	x	x	0	0	0	0	0	0	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
25, 24	IOLH_m3[1:0]	x	RW	Sets the drive strength of the GPIO pin. The drive strength value varies depending on the I/O types.
23 to 18	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
17, 16	IOLH_m2[1:0]	x	RW	Sets the drive strength of the GPIO pin. The drive strength value varies depending on the I/O types.
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
9, 8	IOLH_m1[1:0]	x	RW	Sets the drive strength of the GPIO pin. The drive strength value varies depending on the I/O types.
7 to 2	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
1, 0	IOLH_m0[1:0]	x	RW	Sets the drive strength of the GPIO pin. The drive strength value varies depending on the I/O types.

x: Undefined value

“Higher-Order Address” and “Lower-Order Address” at the offset addresses correspond respectively to the left-side 32-bit units and right-side 32 bit-units of the map below.

Table 4.2-12 Corresponding Pins and Offset Addresses of PFC_IOLH_mn

Offset Address	Register Name	Higher-Order Address				Lower-Order Address			
		Bits 25 and 24	Bits 17 and 16	Bits 9 and 8	Bits 1 and 0	Bits 25 and 24	Bits 17 and 16	Bits 9 and 8	Bits 1 and 0
102Ch / 1028h	PFC_IOLH_05	—	—	—	—	—	—	WDTUDFCM	WDTUDFCA
1034h / 1030h	PFC_IOLH_06	—	—	—	—	—	—	SCIF_TXD	SCIF_RXD
104Ch / 1048h	PFC_IOLH_09	—	—	—	SD0IOVS	SD0PWEN	SD0RSTN	SD0CMD	SD0CLK
1054h / 1050h	PFC_IOLH_0A	SD0DAT7	SD0DAT6	SD0DAT5	SD0DAT4	SD0DAT3	SD0DAT2	SD0DAT1	SD0DAT0
1104h / 1100h	PFC_IOLH_20	P07	P06	P05	P04	P03	P02	P01	P00
110Ch / 1108h	PFC_IOLH_21	P17	P16	P15	P14	P13	P12	P11	P10
1114h / 1110h	PFC_IOLH_22	—	—	—	—	—	—	P21	P20
111Ch / 1118h	PFC_IOLH_23	P37	P36	P35	P34	P33	P32	P31	P30
1124h / 1120h	PFC_IOLH_24	—	—	P45	P44	P43	P42	P41	P40
112Ch / 1128h	PFC_IOLH_25	—	P56	P55	P54	P53	P52	P51	P50
1134h / 1130h	PFC_IOLH_26	—	P66	P65	P64	P63	P62	P61	P60
113Ch / 1138h	PFC_IOLH_27	P77	P76	P75	P74	P73	P72	P71	P70
1144h / 1140h	PFC_IOLH_28	—	—	P85	P84	P83	P82	P81	P80
1154h / 1150h	PFC_IOLH_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
115Ch / 1158h	PFC_IOLH_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
1164h / 1160h	PFC_IOLH_2C	—	—	—	—	—	PC2	PC1	PC0
116Ch / 1168h	PFC_IOLH_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
1174h / 1170h	PFC_IOLH_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
117Ch / 1178h	PFC_IOLH_2F	—	—	—	—	—	PF2	PF1	PF0
1184h / 1180h	PFC_IOLH_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
118Ch / 1188h	PFC_IOLH_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
119Ch / 1198h	PFC_IOLH_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
11A4h / 11A0h	PFC_IOLH_34	—	—	—	—	PK3	PK2	PK1	PK0
11ACh / 11A8h	PFC_IOLH_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
11B4h / 11B0h	PFC_IOLH_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
11E4h / 11E0h	PFC_IOLH_3C	—	—	—	—	PS3	PS2	PS1	PS0

For the values after a reset and the drive strength values, see the table below.

Table 4.2-13 Drive Strength Setting Values on Respective Pins and Values after Reset (1/4)

Pin Name	Drive Strength Setting Value IOLH_mn bits = 00b/10b/01b/11b	Value after Reset	Pin Group*2,*3,*4
WDTUDFCA	×1 / ×2 / ×4 / ×6	01b (×4)*1	A
WDTUDFCM	×1 / ×2 / ×4 / ×6	01b (×4)*1	A
SCIF_RXD	×1 / ×2 / ×4 / ×6	01b (×4)	A
SCIF_TXD	×1 / ×2 / ×4 / ×6	01b (×4)	A
SD0CLK	×1 / ×2 / ×4 / ×6	11b (×6)	B
SD0CMD	×1 / ×2 / ×4 / ×6	11b (×6)	B
SD0DAT0	×1 / ×2 / ×4 / ×6	11b (×6)	B
SD0DAT1	×1 / ×2 / ×4 / ×6	11b (×6)	B
SD0DAT2	×1 / ×2 / ×4 / ×6	11b (×6)	B
SD0DAT3	×1 / ×2 / ×4 / ×6	11b (×6)	B
SD0DAT4	×1 / ×2 / ×4 / ×6	11b (×6)	B
SD0DAT5	×1 / ×2 / ×4 / ×6	11b (×6)	B
SD0DAT6	×1 / ×2 / ×4 / ×6	11b (×6)	B
SD0DAT7	×1 / ×2 / ×4 / ×6	11b (×6)	B
SD0PWEN	×1 / ×2 / ×4 / ×6	11b (×6)	B
SD0IOVS	×1 / ×2 / ×4 / ×6	11b (×6)	B
SD0RSTN	×1 / ×2 / ×4 / ×6	11b (×6)	B
PS0	×1 / ×2 / ×4 / ×6	01b (×4)	A
PS1	×1 / ×2 / ×4 / ×6	01b (×4)	A
PS2	×1 / ×2 / ×4 / ×6	01b (×4)	A
PS3	×1 / ×2 / ×4 / ×6	01b (×4)	A
PJ0	×1 / ×2 / ×4 / ×6	01b (×4)	A
PJ1	×1 / ×2 / ×4 / ×6	01b (×4)	A
PJ2	×1 / ×2 / ×4 / ×6	01b (×4)	A
PJ3	×1 / ×2 / ×4 / ×6	01b (×4)	A
PJ4	×1 / ×2 / ×4 / ×6	01b (×4)	A
PL0	×1 / ×2 / ×4 / ×6	11b (×6)	B
PL1	×1 / ×2 / ×4 / ×6	11b (×6)	B
PL2	×1 / ×2 / ×4 / ×6	11b (×6)	B
PL3	×1 / ×2 / ×4 / ×6	11b (×6)	B
PL4	×1 / ×2 / ×4 / ×6	11b (×6)	B
PL5	×1 / ×2 / ×4 / ×6	11b (×6)	B
PL6	×1 / ×2 / ×4 / ×6	11b (×6)	B
PL7	×1 / ×2 / ×4 / ×6	11b (×6)	B
PM0	×1 / ×2 / ×4 / ×6	11b (×6)	B
PM1	×1 / ×2 / ×4 / ×6	11b (×6)	B
PM2	×1 / ×2 / ×4 / ×6	11b (×6)	B
PM3	×1 / ×2 / ×4 / ×6	11b (×6)	B
PM4	×1 / ×2 / ×4 / ×6	11b (×6)	B
PM5	×1 / ×2 / ×4 / ×6	11b (×6)	B
PM6	×1 / ×2 / ×4 / ×6	11b (×6)	B
PM7	×1 / ×2 / ×4 / ×6	11b (×6)	B

Table 4.2-13 Drive Strength Setting Values on Respective Pins and Values after Reset (2/4)

Pin Name	Drive Strength Setting Value IOLH_mn bits = 00b/10b/01b/11b	Value after Reset	Pin Group*2,*3,*4
PG0	×1 / ×2 / ×4 / ×6	01b (×4)	B
PG1	×1 / ×2 / ×4 / ×6	01b (×4)	B
PG2	×1 / ×2 / ×4 / ×6	01b (×4)	B
PG3	×1 / ×2 / ×4 / ×6	01b (×4)	B
PG4	×1 / ×2 / ×4 / ×6	01b (×4)	B
PG5	×1 / ×2 / ×4 / ×6	01b (×4)	B
PG6	×1 / ×2 / ×4 / ×6	01b (×4)	B
PG7	×1 / ×2 / ×4 / ×6	01b (×4)	B
PH0	×1 / ×2 / ×4 / ×6	01b (×4)	B
PH1	×1 / ×2 / ×4 / ×6	01b (×4)	B
PH2	×1 / ×2 / ×4 / ×6	01b (×4)	B
PH3	×1 / ×2 / ×4 / ×6	01b (×4)	B
PH4	×1 / ×2 / ×4 / ×6	01b (×4)	B
PH5	×1 / ×2 / ×4 / ×6	01b (×4)	B
PK0	×1 / ×2 / ×4 / ×6	01b (×4)	B
PK1	×1 / ×2 / ×4 / ×6	01b (×4)	B
PK2	×1 / ×2 / ×4 / ×6	01b (×4)	B
PK3	×1 / ×2 / ×4 / ×6	01b (×4)	B
PA0	×1 / ×2 / ×4 / ×6	01b (×4)	B
PA1	×1 / ×2 / ×4 / ×6	01b (×4)	B
PA2	×1 / ×2 / ×4 / ×6	01b (×4)	B
PA3	×1 / ×2 / ×4 / ×6	01b (×4)	B
PA4	×1 / ×2 / ×4 / ×6	01b (×4)	B
PA5	×1 / ×2 / ×4 / ×6	01b (×4)	B
PA6	×1 / ×2 / ×4 / ×6	01b (×4)	B
PA7	×1 / ×2 / ×4 / ×6	01b (×4)	B
PB0	×1 / ×2 / ×4 / ×6	01b (×4)	B
PB1	×1 / ×2 / ×4 / ×6	01b (×4)	B
PB2	×1 / ×2 / ×4 / ×6	01b (×4)	B
PB3	×1 / ×2 / ×4 / ×6	01b (×4)	B
PB4	×1 / ×2 / ×4 / ×6	01b (×4)	B
PB5	×1 / ×2 / ×4 / ×6	01b (×4)	B
PB6	×1 / ×2 / ×4 / ×6	01b (×4)	B
PB7	×1 / ×2 / ×4 / ×6	01b (×4)	B
PC0	×1 / ×2 / ×4 / ×6	01b (×4)	B
PC1	×1 / ×2 / ×4 / ×6	01b (×4)	B
PC2	×1 / ×2 / ×4 / ×6	01b (×4)	B
PD0	×1 / ×2 / ×4 / ×6	01b (×4)	B
PD1	×1 / ×2 / ×4 / ×6	01b (×4)	B
PD2	×1 / ×2 / ×4 / ×6	01b (×4)	B
PD3	×1 / ×2 / ×4 / ×6	01b (×4)	B
PD4	×1 / ×2 / ×4 / ×6	01b (×4)	B
PD5	×1 / ×2 / ×4 / ×6	01b (×4)	B
PD6	×1 / ×2 / ×4 / ×6	01b (×4)	B

Table 4.2-13 Drive Strength Setting Values on Respective Pins and Values after Reset (3/4)

Pin Name	Drive Strength Setting Value IOLH_mn bits = 00b/10b/01b/11b	Value after Reset	Pin Group*2,*3,*4
PD7	×1 / ×2 / ×4 / ×6	01b (×4)	B
PE0	×1 / ×2 / ×4 / ×6	01b (×4)	B
PE1	×1 / ×2 / ×4 / ×6	01b (×4)	B
PE2	×1 / ×2 / ×4 / ×6	01b (×4)	B
PE3	×1 / ×2 / ×4 / ×6	01b (×4)	B
PE4	×1 / ×2 / ×4 / ×6	01b (×4)	B
PE5	×1 / ×2 / ×4 / ×6	01b (×4)	B
PE6	×1 / ×2 / ×4 / ×6	01b (×4)	B
PE7	×1 / ×2 / ×4 / ×6	01b (×4)	B
PF0	×1 / ×2 / ×4 / ×6	01b (×4)	B
PF1	×1 / ×2 / ×4 / ×6	01b (×4)	B
PF2	×1 / ×2 / ×4 / ×6	01b (×4)	B
P00	×1 / ×2 / ×4 / ×6	01b (×4)	B
P01	×1 / ×2 / ×4 / ×6	01b (×4)	B
P02	×1 / ×2 / ×4 / ×6	01b (×4)	B
P03	×1 / ×2 / ×4 / ×6	01b (×4)	B
P04	×1 / ×2 / ×4 / ×6	01b (×4)	B
P05	×1 / ×2 / ×4 / ×6	01b (×4)	B
P06	×1 / ×2 / ×4 / ×6	01b (×4)	B
P07	×1 / ×2 / ×4 / ×6	01b (×4)	B
P10	×1 / ×2 / ×4 / ×6	01b (×4)	B
P11	×1 / ×2 / ×4 / ×6	01b (×4)	B
P12	×1 / ×2 / ×4 / ×6	01b (×4)	B
P13	×1 / ×2 / ×4 / ×6	01b (×4)	B
P14	×1 / ×2 / ×4 / ×6	01b (×4)	B
P15	×1 / ×2 / ×4 / ×6	01b (×4)	B
P16	×1 / ×2 / ×4 / ×6	01b (×4)	B
P17	×1 / ×2 / ×4 / ×6	01b (×4)	B
P40	×1 / ×2 / ×4 / ×6	01b (×4)	B
P41	×1 / ×2 / ×4 / ×6	01b (×4)	B
P42	×1 / ×2 / ×4 / ×6	01b (×4)	B
P43	×1 / ×2 / ×4 / ×6	01b (×4)	B
P44	×1 / ×2 / ×4 / ×6	01b (×4)	B
P45	×1 / ×2 / ×4 / ×6	01b (×4)	B
P20	×1 / ×2 / ×3 / ×4	01b (×4)	D
P21	×1 / ×2 / ×3 / ×4	01b (×4)	D
P50	×1 / ×2 / ×4 / ×6	01b (×4)	B
P51	×1 / ×2 / ×4 / ×6	01b (×4)	B
P52	×1 / ×2 / ×4 / ×6	01b (×4)	B
P53	×1 / ×2 / ×4 / ×6	01b (×4)	B
P54	×1 / ×2 / ×4 / ×6	01b (×4)	B
P55	×1 / ×2 / ×4 / ×6	01b (×4)	B
P56	×1 / ×2 / ×4 / ×6	01b (×4)	B
P60	×1 / ×2 / ×4 / ×6	01b (×4)	B

Table 4.2-13 Drive Strength Setting Values on Respective Pins and Values after Reset (4/4)

Pin Name	Drive Strength Setting Value IOLH_mn bits = 00b/10b/01b/11b	Value after Reset	Pin Group*2,*3,*4
P61	×1 / ×2 / ×4 / ×6	01b (×4)	B
P62	×1 / ×2 / ×4 / ×6	01b (×4)	B
P63	×1 / ×2 / ×4 / ×6	01b (×4)	B
P64	×1 / ×2 / ×4 / ×6	01b (×4)	B
P65	×1 / ×2 / ×4 / ×6	01b (×4)	B
P66	×1 / ×2 / ×4 / ×6	01b (×4)	B
P70	×1 / ×2 / ×4 / ×6	01b (×4)	B
P71	×1 / ×2 / ×4 / ×6	01b (×4)	B
P72	×1 / ×2 / ×4 / ×6	01b (×4)	B
P73	×1 / ×2 / ×4 / ×6	01b (×4)	B
P74	×1 / ×2 / ×4 / ×6	01b (×4)	B
P75	×1 / ×2 / ×4 / ×6	01b (×4)	B
P76	×1 / ×2 / ×4 / ×6	01b (×4)	B
P77	×1 / ×2 / ×4 / ×6	01b (×4)	B
P80	×1 / ×2 / ×4 / ×6	01b (×4)	B
P81	×1 / ×2 / ×4 / ×6	01b (×4)	B
P82	×1 / ×2 / ×4 / ×6	01b (×4)	B
P83	×1 / ×2 / ×4 / ×6	01b (×4)	B
P84	×1 / ×2 / ×4 / ×6	01b (×4)	B
P85	×1 / ×2 / ×4 / ×6	01b (×4)	B
P30	×1 / ×2 / ×4 / ×6	01b (×4)	B
P31	×1 / ×2 / ×4 / ×6	01b (×4)	B
P32	×1 / ×2 / ×4 / ×6	01b (×4)	B
P33	×1 / ×2 / ×4 / ×6	01b (×4)	B
P34	×1 / ×2 / ×4 / ×6	01b (×4)	B
P35	×1 / ×2 / ×4 / ×6	01b (×4)	B
P36	×1 / ×2 / ×4 / ×6	01b (×4)	B
P37	×1 / ×2 / ×4 / ×6	01b (×4)	B

Note 1. WDTUDFCA and WDTUDFCM are reset in response to the assertion of the ERROR_RESETn signal from the CPG. They are not reset in response to the assertion of the external QRESN pin.

Note 2. Group A output impedance

I/O voltage	00b	10b	01b	11b
3.3 V	150Ω	75Ω	38Ω	25Ω
1.8 V	130Ω	65Ω	33Ω	22Ω

Note 3. Group B output impedance

I/O voltage	00b	10b	01b	11b
3.3 V	65Ω	55Ω	44Ω	33Ω
1.8 V	50Ω	40Ω	33Ω	25Ω

Note 4. Group D output impedance

I/O voltage	00b	10b	01b	11b
1.8 V	110Ω	55Ω	30Ω	20Ω
1.2 V	150Ω	75Ω	38Ω	25Ω

4.2.2.8 Slew-Rate Switching Registers (PFC_SR_mn)

These registers set the slew rate of the IO block. For the correspondence with the ports, see **Table 4.2-14** and **Table 4.2-15**.

Access Size : 8, 16, 32 bits
Offset Address : <PFC_base> + 1418h
Initial Value : 0x0x_0x0xh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	SR_m3	-	-	-	-	-	-	-	SR_m2
Initial Value	0	0	0	0	0	0	0	x	0	0	0	0	0	0	0	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	SR_m1	-	-	-	-	-	-	-	SR_m0
Initial Value	0	0	0	0	0	0	0	x	0	0	0	0	0	0	0	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
24	SR_m3	x	RW	Sets the slew rate of the GPIO pin. 0b: Fast 1b: Slow
23 to 17	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
16	SR_m2	x	RW	Sets the slew rate of the GPIO pin. 0b: Fast 1b: Slow
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
8	SR_m1	x	RW	Sets the slew rate of the GPIO pin. 0b: Fast 1b: Slow
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
0	SR_m0	x	RW	Sets the slew rate of the GPIO pin. 0b: Fast 1b: Slow

x: Undefined value

“Higher-Order Address” and “Lower-Order Address” at the offset addresses correspond respectively to the left-side 32-bit units and right-side 32 bit-units of the map below.

Table 4.2-14 Corresponding Pins and Offset Addresses of PFC_SR_mn

Offset Address	Register Name	Higher-Order Address				Lower-Order Address			
		Bit 24	Bit 16	Bit 8	Bit 0	Bit 24	Bit 16	Bit 8	Bit 0
142Ch / 1428h	PFC_SR_05	—	—	—	—	—	—	WDTUDFCM	WDTUDFCA
1434h / 1430h	PFC_SR_06	—	—	—	—	—	—	SCIF_TXD	SCIF_RXD
144Ch / 1448h	PFC_SR_09	—	—	—	SD0IOVS	SD0PWEN	SD0RSTN	SD0CMD	SD0CLK
1454h / 1450h	PFC_SR_0A	SD0DAT7	SD0DAT6	SD0DAT5	SD0DAT4	SD0DAT3	SD0DAT2	SD0DAT1	SD0DAT0
1504h / 1500h	PFC_SR_20	P07	P06	P05	P04	P03	P02	P01	P00
150Ch / 1508h	PFC_SR_21	P17	P16	P15	P14	P13	P12	P11	P10
151Ch / 1518h	PFC_SR_23	P37	P36	P35	P34	P33	P32	P31	P30
1524h / 1520h	PFC_SR_24	—	—	P45	P44	P43	P42	P41	P40
152Ch / 1528h	PFC_SR_25	—	P56	P55	P54	P53	P52	P51	P50
1534h / 1530h	PFC_SR_26	—	P66	P65	P64	P63	P62	P61	P60
153Ch / 1538h	PFC_SR_27	P77	P76	P75	P74	P73	P72	P71	P70
1544h / 1540h	PFC_SR_28	—	—	P85	P84	P83	P82	P81	P80
1554h / 1550h	PFC_SR_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
155Ch / 1558h	PFC_SR_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
1564h / 1560h	PFC_SR_2C	—	—	—	—	—	PC2	PC1	PC0
156Ch / 1568h	PFC_SR_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
1574h / 1570h	PFC_SR_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
157Ch / 1578h	PFC_SR_2F	—	—	—	—	—	PF2	PF1	PF0
1584h / 1580h	PFC_SR_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
158Ch / 1588h	PFC_SR_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
159Ch / 1598h	PFC_SR_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
15A4h / 15A0h	PFC_SR_34	—	—	—	—	PK3	PK2	PK1	PK0
15ACh / 15A8h	PFC_SR_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
15B4h / 15B0h	PFC_SR_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
15E4h / 15E0h	PFC_SR_3C	—	—	—	—	PS3	PS2	PS1	PS0

For the values after a reset, see the table below.

Table 4.2-15 Values on Respective Pins after Reset (1/4)

Pin Name	Value after Reset
WDTUDFCA	0b (Fast)*1
WDTUDFCM	0b (Fast)*1
SCIF_RXD	0b (Fast)
SCIF_TXD	0b (Fast)
SD0CLK	0b (Fast)
SD0CMD	0b (Fast)
SD0DAT0	0b (Fast)
SD0DAT1	0b (Fast)
SD0DAT2	0b (Fast)
SD0DAT3	0b (Fast)
SD0DAT4	0b (Fast)
SD0DAT5	0b (Fast)
SD0DAT6	0b (Fast)
SD0DAT7	0b (Fast)
SD0PWEN	0b (Fast)
SD0IOVS	0b (Fast)
SD0RSTN	0b (Fast)
PS0	0b (Fast)
PS1	0b (Fast)
PS2	0b (Fast)
PS3	0b (Fast)
PJ0	0b (Fast)
PJ1	0b (Fast)
PJ2	0b (Fast)
PJ3	0b (Fast)
PJ4	0b (Fast)
PL0	0b (Fast)
PL1	0b (Fast)
PL2	0b (Fast)
PL3	0b (Fast)
PL4	0b (Fast)
PL5	0b (Fast)
PL6	0b (Fast)
PL7	0b (Fast)
PM0	0b (Fast)
PM1	0b (Fast)
PM2	0b (Fast)
PM3	0b (Fast)
PM4	0b (Fast)
PM5	0b (Fast)
PM6	0b (Fast)
PM7	0b (Fast)
PG0	0b (Fast)

Table 4.2-15 Values on Respective Pins after Reset (2/4)

Pin Name	Value after Reset
PG1	0b (Fast)
PG2	0b (Fast)
PG3	0b (Fast)
PG4	0b (Fast)
PG5	0b (Fast)
PG6	0b (Fast)
PG7	0b (Fast)
PH0	0b (Fast)
PH1	0b (Fast)
PH2	0b (Fast)
PH3	0b (Fast)
PH4	0b (Fast)
PH5	0b (Fast)
PK0	0b (Fast)
PK1	0b (Fast)
PK2	0b (Fast)
PK3	0b (Fast)
PA0	0b (Fast)
PA1	0b (Fast)
PA2	0b (Fast)
PA3	0b (Fast)
PA4	0b (Fast)
PA5	0b (Fast)
PA6	0b (Fast)
PA7	0b (Fast)
PB0	0b (Fast)
PB1	0b (Fast)
PB2	0b (Fast)
PB3	0b (Fast)
PB4	0b (Fast)
PB5	0b (Fast)
PB6	0b (Fast)
PB7	0b (Fast)
PC0	0b (Fast)
PC1	0b (Fast)
PC2	0b (Fast)
PD0	0b (Fast)
PD1	0b (Fast)
PD2	0b (Fast)
PD3	0b (Fast)
PD4	0b (Fast)
PD5	0b (Fast)
PD6	0b (Fast)
PD7	0b (Fast)
PE0	0b (Fast)

Table 4.2-15 Values on Respective Pins after Reset (3/4)

Pin Name	Value after Reset
PE1	0b (Fast)
PE2	0b (Fast)
PE3	0b (Fast)
PE4	0b (Fast)
PE5	0b (Fast)
PE6	0b (Fast)
PE7	0b (Fast)
PF0	0b (Fast)
PF1	0b (Fast)
PF2	0b (Fast)
P00	0b (Fast)
P01	0b (Fast)
P02	0b (Fast)
P03	0b (Fast)
P04	0b (Fast)
P05	0b (Fast)
P06	0b (Fast)
P07	0b (Fast)
P10	0b (Fast)
P11	0b (Fast)
P12	0b (Fast)
P13	0b (Fast)
P14	0b (Fast)
P15	0b (Fast)
P16	0b (Fast)
P17	0b (Fast)
P40	0b (Fast)
P41	0b (Fast)
P42	0b (Fast)
P43	0b (Fast)
P44	0b (Fast)
P45	0b (Fast)
P50	0b (Fast)
P51	0b (Fast)
P52	0b (Fast)
P53	0b (Fast)
P54	0b (Fast)
P55	0b (Fast)
P56	0b (Fast)
P60	0b (Fast)
P61	0b (Fast)
P62	0b (Fast)
P63	0b (Fast)
P64	0b (Fast)
P65	0b (Fast)

Table 4.2-15 Values on Respective Pins after Reset (4/4)

Pin Name	Value after Reset
PA1	0b (Fast)
PA2	0b (Fast)
PA3	0b (Fast)
PA4	0b (Fast)
PA5	0b (Fast)
PA6	0b (Fast)
PA7	0b (Fast)
PB0	0b (Fast)
PB1	0b (Fast)
PB2	0b (Fast)
PB3	0b (Fast)
PB4	0b (Fast)
PB5	0b (Fast)
P66	0b (Fast)
P70	0b (Fast)
P71	0b (Fast)
P72	0b (Fast)
P73	0b (Fast)
P74	0b (Fast)
P75	0b (Fast)
P76	0b (Fast)
P77	0b (Fast)
P80	0b (Fast)

Note 1. WDTUDFCA and WDTUDFCM are reset in response to the assertion of the ERROR_RESETn signal from the CPG. They are not reset in response to the assertion of the external QRESN pin.

4.2.2.9 IEN Switching Registers (PFC_IEN_mn)

These registers control inputs on the multiplexed pins other than those in use for GPIO. These registers also control the dedicated input pins. For the correspondence with the ports, see **Table 4.2-16** and **Table 4.2-17**.

Access Size : 8, 16, 32 bits
Offset Address : <PFC_base> + 1818h
Initial Value : 0x0x_0x0xh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	IEN_m3	-	-	-	-	-	-	-	IEN_m2
Initial Value	0	0	0	0	0	0	0	x	0	0	0	0	0	0	0	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	IEN_m1	-	-	-	-	-	-	-	IEN_m0
Initial Value	0	0	0	0	0	0	0	x	0	0	0	0	0	0	0	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
24	IEN_m3	x	RW	Controls the input of pins for multiplexed functions other than GPIO and dedicated pins. 0b: Input disabled 1b: Input enabled
23 to 17	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
16	IEN_m2	x	RW	Controls the input of pins for multiplexed functions other than GPIO and dedicated pins. 0b: Input disabled 1b: Input enabled
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
8	IEN_m1	x	RW	Controls the input of pins for multiplexed functions other than GPIO and dedicated pins. 0b: Input disabled 1b: Input enabled
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
0	IEN_m0	x	RW	Controls the input of pins for multiplexed functions other than GPIO and dedicated pins. 0b: Input disabled 1b: Input enabled

x: Undefined value

“Higher-Order Address” and “Lower-Order Address” at the offset addresses correspond respectively to the left-side 32-bit units and right-side 32 bit-units of the map below.

Table 4.2-16 Corresponding Pins and Offset Addresses of PFC_IEN_mn

Offset Address	Register Name	Higher-Order Address				Lower-Order Address			
		Bit 24	Bit 16	Bit 8	Bit 0	Bit 24	Bit 16	Bit 8	Bit 0
184Ch / 1848h	PFC_IEN_09	—	—	—	—	—	—	SD0CMD	—
1854h / 1850h	PFC_IEN_0A	SD0DAT7	SD0DAT6	SD0DAT5	SD0DAT4	SD0DAT3	SD0DAT2	SD0DAT1	SD0DAT0
1954h / 1950h	PFC_IEN_2A	—	—	—	—	—	—	—	PA0
196Ch / 1968h	PFC_IEN_2D	—	—	—	—	—	—	—	PD0
1984h / 1980h	PFC_IEN_30	—	—	PG5	PG4	PG3	PG2	PG1	—
198Ch / 1988h	PFC_IEN_31	—	—	PH5	PH4	PH3	PH2	PH1	—
199Ch / 1998h	PFC_IEN_33	—	—	—	—	—	—	—	PJ0

For the values after a reset, see the table below.

Table 4.2-17 Values on Respective Pins after Reset

Pin Name	Value after Reset
SD0CMD	1b (input enabled)
SD0DAT0	1b (input enabled)
SD0DAT1	1b (input enabled)
SD0DAT2	1b (input enabled)
SD0DAT3	1b (input enabled)
SD0DAT4	1b (input enabled)
SD0DAT5	1b (input enabled)
SD0DAT6	1b (input enabled)
SD0DAT7	1b (input enabled)
PA0	1b (input enabled)
PD0	1b (input enabled)
PG1	1b (input enabled)
PG2	1b (input enabled)
PG3	1b (input enabled)
PG4	1b (input enabled)
PG5	1b (input enabled)
PH1	1b (input enabled)
PH2	1b (input enabled)
PH3	1b (input enabled)
PH4	1b (input enabled)
PH5	1b (input enabled)
PJ0	1b (input enabled)

4.2.2.10 PU/PD Switching Registers (PFC_PUPD_mn)

These registers set pulling up and down by the IO block of signals on pins. For the correspondence with the ports, see **Table 4.2-18** and **Table 4.2-19**.

Access Size : 8, 16, 32 bits
Offset Address : <PFC_base> + 1C28h
Initial Value : 0x0x_0x0xh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	PUPD_m3[1:0]		-	-	-	-	-	-	PUPD_m2[1:0]	
Initial Value	0	0	0	0	0	0	x	x	0	0	0	0	0	0	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PUPD_m1[1:0]		-	-	-	-	-	-	PUPD_m0[1:0]	
Initial Value	0	0	0	0	0	0	x	x	0	0	0	0	0	0	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
25, 24	PUPD_m3[1:0]	x	RW	Sets the pull-up/down control of the GPIO pin. 0xb: Pull-up/down not selected 10b: Pull-down selected 11b: Pull-up selected
23 to 18	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
17 ,16	PUPD_m2[1:0]	x	RW	Sets the pull-up/down control of the GPIO pin. 0xb: Pull-up/down not selected 10b: Pull-down selected 11b: Pull-up selected
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
9, 8	PUPD_m1[1:0]	x	RW	Sets the pull-up/down control of the GPIO pin. 0xb: Pull-up/down not selected 10b: Pull-down selected 11b: Pull-up selected
7 to 2	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
1, 0	PUPD_m0[1:0]	x	RW	Sets the pull-up/down control of the GPIO pin. 0xb: Pull-up/down not selected 10b: Pull-down selected 11b: Pull-up selected

“Higher-Order Address” and “Lower-Order Address” at the offset addresses correspond respectively to the left-side 32-bit units and right-side 32 bit-units of the map below.

Table 4.2-18 Corresponding Pins and Offset Addresses of PFC_PUPD_mn

Offset Address	Register Name	Higher-Order Address				Lower-Order Address			
		Bits 25 and 24	Bits 17 and 16	Bits 9 and 8	Bits 1 and 0	Bits 25 and 24	Bits 17 and 16	Bits 9 and 8	Bits 1 and 0
1C2Ch / 1C28h	PFC_PUPD_05	—	—	—	—	—	—	WDTUDF_CM	WDTUDF_CA
1C34h / 1C30h	PFC_PUPD_06	—	—	—	—	—	—	SCIF_TXD	SCIF_RXD
1C4Ch / 1C48h	PFC_PUPD_09	—	—	—	—	—	—	SD0CMD	-
1C54h / 1C50h	PFC_PUPD_0A	SD0DAT7	SD0DAT6	SD0DAT5	SD0DAT4	SD0DAT3	SD0DAT2	SD0DAT1	SD0DAT0
1D04h / 1D00h	PFC_PUPD_20	P07	P06	P05	P04	P03	P02	P01	P00
1D0Ch / 1D08h	PFC_PUPD_21	P17	P16	P15	P14	P13	P12	P11	P10
1D14h / 1D10h	PFC_PUPD_22	—	—	—	—	—	—	P21	P20
1D1Ch / 1D18h	PFC_PUPD_23	P37	P36	P35	P34	P33	P32	P31	P30
1D24h / 1D20h	PFC_PUPD_24	—	—	P45	P44	P43	P42	P41	P40
1D2Ch / 1D28h	PFC_PUPD_25	—	P56	P55	P54	P53	P52	P51	P50
1D34h / 1D30h	PFC_PUPD_26	—	P66	P65	P64	P63	P62	P61	P60
1D3Ch / 1D38h	PFC_PUPD_27	P77	P76	P75	P74	P73	P72	P71	P70
1D44h / 1D40h	PFC_PUPD_28	—	—	P85	P84	P83	P82	P81	P80
1D54h / 1D50h	PFC_PUPD_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
1D5Ch / 1D58h	PFC_PUPD_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
1D64h / 1D60h	PFC_PUPD_2C	—	—	—	—	—	PC2	PC1	PC0
1D6Ch / 1D68h	PFC_PUPD_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
1D74h / 1D70h	PFC_PUPD_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
1D7Ch / 1D78h	PFC_PUPD_2F	—	—	—	—	—	PF2	PF1	PF0
1D84h / 1D80h	PFC_PUPD_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
1D8Ch / 1D88h	PFC_PUPD_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
1D9Ch / 1D98h	PFC_PUPD_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
1DA4h / 1DA0h	PFC_PUPD_34	—	—	—	—	PK3	PK2	PK1	PK0
1DACh / 1DA8h	PFC_PUPD_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
1DB4h / 1DB0h	PFC_PUPD_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
1DE4h / 1DE0h	PFC_PUPD_3C	—	—	—	—	PS3	PS2	PS1	PS0

For the values after a reset, see the table below.

Table 4.2-19 Values on Respective Pins after Reset (1/4)

Pin Name	Value after Reset
WDTUDFCA	00b (open)*1
WDTUDFCM	00b (open)*1
SCIF_RXD	00b (open)
SCIF_TXD	00b (open)
SD0CMD	00b (open)
SD0DAT0	00b (open)
SD0DAT1	00b (open)
SD0DAT2	00b (open)
SD0DAT3	00b (open)
SD0DAT4	00b (open)
SD0DAT5	00b (open)
SD0DAT6	00b (open)
SD0DAT7	00b (open)
PS0	00b (open)
PS1	10b (Pull-down)
PS2	10b (Pull-down)
PS3	11b (Pull-up)
PJ0	00b (open)
PJ1	00b (open)
PJ2	00b (open)
PJ3	00b (open)
PJ4	00b (open)
PL0	00b (open)
PL1	00b (open)
PL2	00b (open)
PL3	00b (open)
PL4	00b (open)
PL5	00b (open)
PL6	00b (open)
PL7	00b (open)
PM0	00b (open)
PM1	00b (open)
PM2	00b (open)
PM3	00b (open)
PM4	00b (open)
PM5	00b (open)
PM6	00b (open)
PM7	00b (open)
PG0	00b (open)
PG1	00b (open)
PG2	00b (open)
PG3	00b (open)
PG4	00b (open)

Table 4.2-19 Values on Respective Pins after Reset (2/4)

Pin Name	Value after Reset
PG5	00b (open)
PG6	00b (open)
PG7	00b (open)
PH0	00b (open)
PH1	00b (open)
PH2	00b (open)
PH3	00b (open)
PH4	00b (open)
PH5	00b (open)
PK0	00b (open)
PK1	00b (open)
PK2	00b (open)
PK3	00b (open)
PA0	00b (open)
PA1	00b (open)
PA2	00b (open)
PA3	00b (open)
PA4	00b (open)
PA5	00b (open)
PA6	00b (open)
PA7	00b (open)
PB0	00b (open)
PB1	00b (open)
PB2	00b (open)
PB3	00b (open)
PB4	00b (open)
PB5	00b (open)
PB6	00b (open)
PB7	00b (open)
PC0	00b (open)
PC1	00b (open)
PC2	00b (open)
PD0	00b (open)
PD1	00b (open)
PD2	00b (open)
PD3	00b (open)
PD4	00b (open)
PD5	00b (open)
PD6	00b (open)
PD7	00b (open)
PE0	00b (open)
PE1	00b (open)
PE2	00b (open)
PE3	00b (open)
PE4	00b (open)

Table 4.2-19 Values on Respective Pins after Reset (3/4)

Pin Name	Value after Reset
PE5	00b (open)
PE6	00b (open)
PE7	00b (open)
PF0	00b (open)
PF1	00b (open)
PF2	00b (open)
P00	00b (open)
P01	00b (open)
P02	00b (open)
P03	00b (open)
P04	00b (open)
P05	00b (open)
P06	00b (open)
P07	00b (open)
P10	00b (open)
P11	00b (open)
P12	00b (open)
P13	00b (open)
P14	00b (open)
P15	00b (open)
P16	00b (open)
P17	00b (open)
P40	00b (open)
P41	00b (open)
P42	00b (open)
P43	00b (open)
P44	00b (open)
P45	00b (open)
P20	00b (open)
P21	00b (open)
P50	00b (open)
P51	00b (open)
P52	00b (open)
P53	00b (open)
P54	00b (open)
P55	00b (open)
P56	00b (open)
P60	00b (open)
P61	00b (open)
P62	00b (open)
P63	00b (open)
P64	00b (open)
P65	00b (open)
P66	00b (open)
P70	00b (open)

Table 4.2-19 Values on Respective Pins after Reset (4/4)

Pin Name	Value after Reset
P71	00b (open)
P72	00b (open)
P73	00b (open)
P74	00b (open)
P75	00b (open)
P76	00b (open)
P77	00b (open)
P80	00b (open)
P81	00b (open)
P82	00b (open)
P83	00b (open)
P84	00b (open)
P85	00b (open)
P30	00b (open)
P31	00b (open)
P32	00b (open)
P33	00b (open)
P34	00b (open)
P35	00b (open)
P36	00b (open)
P37	00b (open)

Note 1. WDTUDFCA and WDTUDFCM are reset in response to the assertion of the ERROR_RESETn signal from the CPG. They are not reset in response to the assertion of the external QRESN pin.

4.2.2.11 Digital Noise Filter (FILONOFF) Registers (PFC_FILONOFF_mn)

These registers set whether or not to use the digital noise filter. For the correspondence with the ports, see **Table 4.2-20**.

Access Size : 8, 16, 32 bits
Offset Address : <PFC_base> + 2008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	FILONOFF_m3	-	-	-	-	-	-	-	FILONOFF_m2
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	FILONOFF_m1	-	-	-	-	-	-	-	FILONOFF_m0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
24	FILONOFF_m3	0h	RW	Sets whether or not to use the digital noise filter for the GPIO pin. 0b: No filter 1b: With filter
23 to 17	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
16	FILONOFF_m2	0h	RW	Sets whether or not to use the digital noise filter for the GPIO pin. 0b: No filter 1b: With filter
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
8	FILONOFF_m1	0h	RW	Sets whether or not to use the digital noise filter for the GPIO pin. 0b: No filter 1b: With filter
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
0	FILONOFF_m0	0h	RW	Sets whether or not to use the digital noise filter for the GPIO pin. 0b: No filter 1b: With filter

NOTE

Before setting these registers, disable interrupt detection handled by the ICU.

“Higher-Order Address” and “Lower-Order Address” at the offset addresses correspond respectively to the left-side 32-bit units and right-side 32 bit-units of the map below.

Table 4.2-20 Corresponding Pins and Offset Addresses of PFC_FILONOFF_mn

		Higher-Order Address				Lower-Order Address			
Offset Address	Register Name	Bit 24	Bit 16	Bit 8	Bit 0	Bit 24	Bit 16	Bit 8	Bit 0
2104h / 2100h	PFC_FILONOFF_20	P07	P06	P05	P04	P03	P02	P01	P00
210Ch / 2108h	PFC_FILONOFF_21	P17	P16	P15	P14	P13	P12	P11	P10
2114h / 2110h	PFC_FILONOFF_22	—	—	—	—	—	—	P21	P20
211Ch / 2118h	PFC_FILONOFF_23	P37	P36	P35	P34	P33	P32	P31	P30
2124h / 2120h	PFC_FILONOFF_24	—	—	P45	P44	P43	P42	P41	P40
212Ch / 2128h	PFC_FILONOFF_25	—	P56	P55	P54	P53	P52	P51	P50
2134h / 2130h	PFC_FILONOFF_26	—	P66	P65	P64	P63	P62	P61	P60
213Ch / 2138h	PFC_FILONOFF_27	P77	P76	P75	P74	P73	P72	P71	P70
2144h / 2140h	PFC_FILONOFF_28	—	—	P85	P84	P83	P82	P81	P80
2154h / 2150h	PFC_FILONOFF_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
215Ch / 2158h	PFC_FILONOFF_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
2164h / 2160h	PFC_FILONOFF_2C	—	—	—	—	—	PC2	PC1	PC0
216Ch / 2168h	PFC_FILONOFF_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
2174h / 2170h	PFC_FILONOFF_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
217Ch / 2178h	PFC_FILONOFF_2F	—	—	—	—	—	PF2	PF1	PF0
2184h / 2180h	PFC_FILONOFF_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
218Ch / 2188h	PFC_FILONOFF_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
219Ch / 2198h	PFC_FILONOFF_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
21A4h / 21A0h	PFC_FILONOFF_34	—	—	—	—	PK3	PK2	PK1	PK0
21ACh / 21A8h	PFC_FILONOFF_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
21B4h / 21B0h	PFC_FILONOFF_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
21E4h / 21E0h	PFC_FILONOFF_3C	—	—	—	—	PS3	PS2	PS1	PS0

4.2.2.12 Digital Noise Filter (FILNUM) Registers (PFC_FILNUM_mn)

These registers set the number of stages of the digital noise filter. For the correspondence with the ports, see **Table 4.2-21**.

Access Size : 8, 16, 32 bits
Offset Address : <PFC_base> + 2408h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	FILNUM_m3 [1:0]		-	-	-	-	-	-	FILNUM_m2 [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	FILNUM_m1 [1:0]		-	-	-	-	-	-	FILNUM_m0 [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
25, 24	FILNUM_m3 [1:0]	0h	RW	Sets the number of stages of the digital noise filter for the GPIO pin. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
23 to 18	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
17, 16	FILNUM_m2 [1:0]	0h	RW	Sets the number of stages of the digital noise filter for the GPIO pin. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
9, 8	FILNUM_m1 [1:0]	0h	RW	Sets the number of stages of the digital noise filter for the GPIO pin. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
7 to 2	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
1, 0	FILNUM_m0 [1:0]	0h	RW	Sets the number of stages of the digital noise filter for the GPIO pin. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages

NOTE

Before setting these registers, disable interrupt detection handled by the ICU.

“Higher-Order Address” and “Lower-Order Address” at the offset addresses correspond respectively to the left-side 32-bit units and right-side 32 bit-units of the map below.

Table 4.2-21 Corresponding Pins and Offset Addresses of PFC_FILNUM_mn

Offset Address	Register Name	Higher-Order Address				Lower-Order Address			
		Bit 24	Bit 16	Bit 8	Bit 0	Bit 24	Bit 16	Bit 8	Bit 0
2504h / 2500h	PFC_FILNUM_20	P07	P06	P05	P04	P03	P02	P01	P00
250Ch / 2508h	PFC_FILNUM_21	P17	P16	P15	P14	P13	P12	P11	P10
2514h / 2510h	PFC_FILNUM_22	—	—	—	—	—	—	P21	P20
251Ch / 2518h	PFC_FILNUM_23	P37	P36	P35	P34	P33	P32	P31	P30
2524h / 2520h	PFC_FILNUM_24	—	—	P45	P44	P43	P42	P41	P40
252Ch / 2528h	PFC_FILNUM_25	—	P56	P55	P54	P53	P52	P51	P50
2534h / 2530h	PFC_FILNUM_26	—	P66	P65	P64	P63	P62	P61	P60
253Ch / 2538h	PFC_FILNUM_27	P77	P76	P75	P74	P73	P72	P71	P70
2544h / 2540h	PFC_FILNUM_28	—	—	P85	P84	P83	P82	P81	P80
2554h / 2550h	PFC_FILNUM_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
255Ch / 2558h	PFC_FILNUM_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
2564h / 2560h	PFC_FILNUM_2C	—	—	—	—	—	PC2	PC1	PC0
256Ch / 2568h	PFC_FILNUM_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
2574h / 2570h	PFC_FILNUM_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
257Ch / 2578h	PFC_FILNUM_2F	—	—	—	—	—	PF2	PF1	PF0
2584h / 2580h	PFC_FILNUM_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
258Ch / 2588h	PFC_FILNUM_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
259Ch / 2598h	PFC_FILNUM_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
25A4h / 25A0h	PFC_FILNUM_34	—	—	—	—	PK3	PK2	PK1	PK0
25ACh / 25A8h	PFC_FILNUM_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
25B4h / 25B0h	PFC_FILNUM_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
25E4h / 25E0h	PFC_FILNUM_3C	—	—	—	—	PS3	PS2	PS1	PS0

4.2.2.13 Digital Noise Filter (FILCLKSEL) Registers (PFC_FILCLKSEL_mn)

These registers set the sampling interval to be input to the digital noise filter. The sampling interval is the time obtained by frequency division for the SHCLK (IOTOP_0_SHCLK) input clock. For the correspondence with the ports, see **Table 4.2-22**.

Access Size : 8, 16, 32 bits
Offset Address : <PFC_base> + 2808h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	FILCLKSEL_m3 [1:0]		-	-	-	-	-	-	FILCLKSEL_m2 [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	FILCLKSEL_m1 [1:0]		-	-	-	-	-	-	FILCLKSEL_m0 [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
25, 24	FILCLKSEL_m3 [1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the GPIO pin. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
23 to 18	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
17, 16	FILCLKSEL_m2 [1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the GPIO pin. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
9, 8	FILCLKSEL_m1 [1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the GPIO pin. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
7 to 2	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
1, 0	FILCLKSEL_m0 [1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the GPIO pin. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256

NOTE

Before setting these registers, disable interrupt detection handled by the ICU.

“Higher-Order Address” and “Lower-Order Address” at the offset addresses correspond respectively to the left-side 32-bit units and right-side 32 bit-units of the map below.

Table 4.2-22 Corresponding Pins and Offset Addresses of PFC_FILCLKSEL_mn

		Higher-Order Address				Lower-Order Address			
Offset Address	Register Name	Bit 24	Bit 16	Bit 8	Bit 0	Bit 24	Bit 16	Bit 8	Bit 0
2904h / 2900h	PFC_FILCLKSEL_20	P07	P06	P05	P04	P03	P02	P01	P00
290Ch / 2908h	PFC_FILCLKSEL_21	P17	P16	P15	P14	P13	P12	P11	P10
2914h / 2910h	PFC_FILCLKSEL_22	—	—	—	—	—	—	P21	P20
291Ch / 2918h	PFC_FILCLKSEL_23	P37	P36	P35	P34	P33	P32	P31	P30
2924h / 2920h	PFC_FILCLKSEL_24	—	—	P45	P44	P43	P42	P41	P40
292Ch / 2928h	PFC_FILCLKSEL_25	—	P56	P55	P54	P53	P52	P51	P50
2934h / 2930h	PFC_FILCLKSEL_26	—	P66	P65	P64	P63	P62	P61	P60
293Ch / 2938h	PFC_FILCLKSEL_27	P77	P76	P75	P74	P73	P72	P71	P70
2944h / 2940h	PFC_FILCLKSEL_28	—	—	P85	P84	P83	P82	P81	P80
2954h / 2950h	PFC_FILCLKSEL_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
295Ch / 2958h	PFC_FILCLKSEL_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
2964h / 2960h	PFC_FILCLKSEL_2C	—	—	—	—	—	PC2	PC1	PC0
296Ch / 2968h	PFC_FILCLKSEL_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
2974h / 2970h	PFC_FILCLKSEL_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
297Ch / 2978h	PFC_FILCLKSEL_2F	—	—	—	—	—	PF2	PF1	PF0
2984h / 2980h	PFC_FILCLKSEL_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
298Ch / 2988h	PFC_FILCLKSEL_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
299Ch / 2998h	PFC_FILCLKSEL_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
29A4h / 29A0h	PFC_FILCLKSEL_34	—	—	—	—	PK3	PK2	PK1	PK0
29ACh / 29A8h	PFC_FILCLKSEL_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
29B4h / 29B0h	PFC_FILCLKSEL_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
29E4h / 29E0h	PFC_FILCLKSEL_3C	—	—	—	—	PS3	PS2	PS1	PS0

4.2.2.14 ISEL Control Registers (PFC_ISEL_mn)

These registers control the IE pin of the IO block (**Figure 4.2-2**). For the correspondence with the ports, see **Table 4.2-23**.

Access Size : 8, 16, 32 bits
Offset Address : <PFC_base> + 2D00h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	ISEL_m3	-	-	-	-	-	-	-	ISEL_m2
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	ISEL_m1	-	-	-	-	-	-	-	ISEL_m0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
24	ISEL_m3	0h	RW	Controls the IE pin of IO block. 0b: The IE pin of IO block is controlled by the PM_mn register when PMC_mn = 0 (port mode) and by the multiplexed function when PMC_mn = 1 (control mode). 1b: The IE pin of IO block is forcibly enabled (IE ON).
23 to 17	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
16	ISEL_m2	0h	RW	Controls the IE pin of IO block. 0b: The IE pin of IO block is controlled by the PM_mn register when PMC_mn = 0 (port mode) and by the multiplexed function when PMC_mn = 1 (control mode). 1b: The IE pin of IO block is forcibly enabled (IE ON).
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
8	ISEL_m1	0h	RW	Controls the IE pin of IO block. 0b: The IE pin of IO block is controlled by the PM_mn register when PMC_mn = 0 (port mode) and by the multiplexed function when PMC_mn = 1 (control mode). 1b: The IE pin of IO block is forcibly enabled (IE ON).
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
0	ISEL_m0	0h	RW	Controls the IE pin of IO block. 0b: The IE pin of IO block is controlled by the PM_mn register when PMC_mn = 0 (port mode) and by the multiplexed function when PMC_mn = 1 (control mode). 1b: The IE pin of IO block is forcibly enabled (IE ON).

NOTE

When PMC_mn = 1b or the event link controller is to be used, setting ISEL_mn = 1b is prohibited.

“Higher-Order Address” and “Lower-Order Address” at the offset addresses correspond respectively to the left-side 32-bit units and right-side 32 bit-units of the map below.

Table 4.2-23 Corresponding Pins and Offset Addresses of PFC_ISEL_mn

		Higher-Order Address				Lower-Order Address			
Offset Address	Register Name	Bit 24	Bit 16	Bit 8	Bit 0	Bit 24	Bit 16	Bit 8	Bit 0
2D04h / 2D00h	PFC_ISEL_20	P07	P06	P05	P04	P03	P02	P01	P00
2D0Ch / 2D08h	PFC_ISEL_21	P17	P16	P15	P14	P13	P12	P11	P10
2D14h / 2D10h	PFC_ISEL_22	—	—	—	—	—	—	P21	P20
2D1Ch / 2D18h	PFC_ISEL_23	P37	P36	P35	P34	P33	P32	P31	P30
2D24h / 2D20h	PFC_ISEL_24	—	—	P45	P44	P43	P42	P41	P40
2D2Ch / 2D28h	PFC_ISEL_25	—	P56	P55	P54	P53	P52	P51	P50
2D34h / 2D30h	PFC_ISEL_26	—	P66	P65	P64	P63	P62	P61	P60
2D3Ch / 2D38h	PFC_ISEL_27	P77	P76	P75	P74	P73	P72	P71	P70
2D44h / 2D40h	PFC_ISEL_28	—	—	P85	P84	P83	P82	P81	P80
2D54h / 2D50h	PFC_ISEL_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
2D5Ch / 2D58h	PFC_ISEL_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
2D64h / 2D60h	PFC_ISEL_2C	—	—	—	—	—	PC2	PC1	PC0
2D6Ch / 2D68h	PFC_ISEL_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
2D74h / 2D70h	PFC_ISEL_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
2D7Ch / 2D78h	PFC_ISEL_2F	—	—	—	—	—	PF2	PF1	PF0
2D84h / 2D80h	PFC_ISEL_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
2D8Ch / 2D88h	PFC_ISEL_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
2D9Ch / 2D98h	PFC_ISEL_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
2DA4h / 2DA0h	PFC_ISEL_34	—	—	—	—	PK3	PK2	PK1	PK0
2DACh / 2DA8h	PFC_ISEL_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
2DB4h / 2DB0h	PFC_ISEL_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
2DE4h / 2DE0h	PFC_ISEL_3C	—	—	—	—	PS3	PS2	PS1	PS0

4.2.2.15 N-ch Open Drain Control Registers (PFC_NOD_mn)

These registers control the N-ch. open drain outputs for the multiplexed pins. For the correspondence with the ports, see **Table 4.2-24** and **Table 4.2-25**.

Access Size : 8, 16, 32 bits
Offset Address : <PFC_base> + 3028h
Initial Value : 0000_0x0xh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	NOD_m3	-	-	-	-	-	-	-	NOD_m2
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	NOD_m1	-	-	-	-	-	-	-	NOD_m0
Initial Value	0	0	0	0	0	0	0	x	0	0	0	0	0	0	0	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
24	NOD_m3	0h	RW	Controls the N-ch. open drain of the GPIO pin. 0b: N-ch. open drain OFF 1b: N-ch. open drain ON
23 to 17	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
16	NOD_m2	0h	RW	Controls the N-ch. open drain of the GPIO pin. 0b: N-ch. open drain OFF 1b: N-ch. open drain ON
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
8	NOD_m1	x	RW	Controls the N-ch. open drain of the GPIO pin. 0b: N-ch. open drain OFF 1b: N-ch. open drain ON
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
0	NOD_m0	x	RW	Controls the N-ch. open drain of the GPIO pin. 0b: N-ch. open drain OFF 1b: N-ch. open drain ON

“Higher-Order Address” and “Lower-Order Address” at the offset addresses correspond respectively to the left-side 32-bit units and right-side 32 bit-units of the map below.

Table 4.2-24 Corresponding Pins and Offset Addresses of PFC_NOD_mn

Offset Address	Register Name	Higher-Order Address				Lower-Order Address			
		Bit 24	Bit 16	Bit 8	Bit 0	Bit 24	Bit 16	Bit 8	Bit 0
302Ch / 3028h	PFC_NOD_05	-	-	-	-	-	-	WDTUDFCM	WDTUDFCA
3104h / 3100h	PFC_NOD_20	P07	P06	P05	P04	P03	P02	P01	P00
310Ch / 3108h	PFC_NOD_21	P17	P16	P15	P14	P13	P12	P11	P10
3114h / 3110h	PFC_NOD_22	-	-	-	-	-	-	P21	P20
311Ch / 3118h	PFC_NOD_23	P37	P36	P35	P34	P33	P32	P31	P30
3124h / 3120h	PFC_NOD_24	-	-	P45	P44	P43	P42	P41	P40
312Ch / 3128h	PFC_NOD_25	-	P56	P55	P54	P53	P52	P51	P50
3134h / 3130h	PFC_NOD_26	-	P66	P65	P64	P63	P62	P61	P60
313Ch / 3138h	PFC_NOD_27	P77	P76	P75	P74	P73	P72	P71	P70
3144h / 3140h	PFC_NOD_28	-	-	P85	P84	P83	P82	P81	P80
3154h / 3150h	PFC_NOD_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
315Ch / 3158h	PFC_NOD_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
3164h / 3160h	PFC_NOD_2C	-	-	-	-	-	PC2	PC1	PC0
316Ch / 3168h	PFC_NOD_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
3174h / 3170h	PFC_NOD_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
317Ch / 3178h	PFC_NOD_2F	-	-	-	-	-	PF2	PF1	PF0
3184h / 3180h	PFC_NOD_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
318Ch / 3188h	PFC_NOD_31	-	-	PH5	PH4	PH3	PH2	PH1	PH0
319Ch / 3198h	PFC_NOD_33	-	-	-	PJ4	PJ3	PJ2	PJ1	PJ0
31A4h / 31A0h	PFC_NOD_34	-	-	-	-	PK3	PK2	PK1	PK0
31ACh / 31A8h	PFC_NOD_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
31B4h / 31B0h	PFC_NOD_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
31E4h / 31E0h	PFC_NOD_3C	-	-	-	-	PS3	PS2	PS1	PS0

Table 4.2-25 Values on Pins after Reset (1/4)

Pin Name	Value after Reset
WDTUDFCA	1b (ON)*1
WDTUDFCM	1b (ON)*1
PS0	0b (OFF)
PS1	0b (OFF)
PS2	0b (OFF)
PS3	0b (OFF)
PJ0	0b (OFF)
PJ1	0b (OFF)
PJ2	0b (OFF)
PJ3	0b (OFF)
PJ4	0b (OFF)
PL0	0b (OFF)
PL1	0b (OFF)
PL2	0b (OFF)
PL3	0b (OFF)
PL4	0b (OFF)
PL5	0b (OFF)
PL6	0b (OFF)
PL7	0b (OFF)
PM0	0b (OFF)
PM1	0b (OFF)
PM2	0b (OFF)
PM3	0b (OFF)
PM4	0b (OFF)
PM5	0b (OFF)
PM6	0b (OFF)
PM7	0b (OFF)
PG0	0b (OFF)
PG1	0b (OFF)
PG2	0b (OFF)
PG3	0b (OFF)
PG4	0b (OFF)
PG5	0b (OFF)
PG6	0b (OFF)
PG7	0b (OFF)
PH0	0b (OFF)
PH1	0b (OFF)
PH2	0b (OFF)
PH3	0b (OFF)
PH4	0b (OFF)
PH5	0b (OFF)
PK0	0b (OFF)
PK1	0b (OFF)
PK2	0b (OFF)
PK3	0b (OFF)

Table 4.2-25 Values on Pins after Reset (2/4)

Pin Name	Value after Reset
PA0	0b (OFF)
PA1	0b (OFF)
PA2	0b (OFF)
PA3	0b (OFF)
PA4	0b (OFF)
PA5	0b (OFF)
PA6	0b (OFF)
PA7	0b (OFF)
PB0	0b (OFF)
PB1	0b (OFF)
PB2	0b (OFF)
PB3	0b (OFF)
PB4	0b (OFF)
PB5	0b (OFF)
PB6	0b (OFF)
PB7	0b (OFF)
PC0	0b (OFF)
PC1	0b (OFF)
PC2	0b (OFF)
PD0	0b (OFF)
PD1	0b (OFF)
PD2	0b (OFF)
PD3	0b (OFF)
PD4	0b (OFF)
PD5	0b (OFF)
PD6	0b (OFF)
PD7	0b (OFF)
PE0	0b (OFF)
PE1	0b (OFF)
PE2	0b (OFF)
PE3	0b (OFF)
PE4	0b (OFF)
PE5	0b (OFF)
PE6	0b (OFF)
PE7	0b (OFF)
PF0	0b (OFF)
PF1	0b (OFF)
PF2	0b (OFF)
P00	0b (OFF)
P01	0b (OFF)
P02	0b (OFF)
P03	0b (OFF)
P04	0b (OFF)
P05	0b (OFF)
P06	0b (OFF)

Table 4.2-25 Values on Pins after Reset (3/4)

Pin Name	Value after Reset
P07	0b (OFF)
P10	0b (OFF)
P11	0b (OFF)
P12	0b (OFF)
P13	0b (OFF)
P14	0b (OFF)
P15	0b (OFF)
P16	0b (OFF)
P17	0b (OFF)
P40	0b (OFF)
P41	0b (OFF)
P42	0b (OFF)
P43	0b (OFF)
P44	0b (OFF)
P45	0b (OFF)
P20	0b (OFF)
P21	0b (OFF)
P50	0b (OFF)
P51	0b (OFF)
P52	0b (OFF)
P53	0b (OFF)
P54	0b (OFF)
P55	0b (OFF)
P56	0b (OFF)
P60	0b (OFF)
P61	0b (OFF)
P62	0b (OFF)
P63	0b (OFF)
P64	0b (OFF)
P65	0b (OFF)
P66	0b (OFF)
P70	0b (OFF)
P71	0b (OFF)
P72	0b (OFF)
P73	0b (OFF)
P74	0b (OFF)
P75	0b (OFF)
P76	0b (OFF)
P77	0b (OFF)
P80	0b (OFF)
P81	0b (OFF)
P82	0b (OFF)
P83	0b (OFF)
P84	0b (OFF)
P85	0b (OFF)

Table 4.2-25 Values on Pins after Reset (4/4)

Pin Name	Value after Reset
P30	0b (OFF)
P31	0b (OFF)
P32	0b (OFF)
P33	0b (OFF)
P34	0b (OFF)
P35	0b (OFF)
P36	0b (OFF)
P37	0b (OFF)

Note 1. WDTUDFCA and WDTUDFCM are reset in response to the assertion of the ERROR_RESETn signal from the CPG. They are not reset in response to the assertion of the external QRESN pin.

NOTE

Use WDTUDFCA and WDTUDFCM with N-ch. open drain outputs (NOD_05[0] = 1b, NOD_05[8] = 1b) and include an external pull-up resistor.

4.2.2.16 Schmitt Control Registers (PFC_SMT_mn)

These registers control the Schmitt-trigger inputs for the multiplexed pins. For the correspondence with the ports, see **Table 4.2-26**.

Access Size : 8, 16, 32 bits
Offset Address : <PFC_base> + 3500h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	SMT_m3	-	-	-	-	-	-	-	SMT_m2
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	SMT_m1	-	-	-	-	-	-	-	SMT_m0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
24	SMT_m3	0h	RW	Controls the Schmitt pin of the GPIO pins. 0b: Sets the Schmitt pin to low (Schmitt OFF). 1b: Sets the Schmitt pin to high (Schmitt ON). When the RIIC0 to RIIC8 functions are selected, the Schmitt pin is high, regardless of the value set in this register.
23 to 17	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
16	SMT_m2	0h	RW	Controls the Schmitt pin of the GPIO pins. 0b: Sets the Schmitt pin to low (Schmitt OFF). 1b: Sets the Schmitt pin to high (Schmitt ON). When the RIIC0 to RIIC8 functions are selected, the Schmitt pin is high, regardless of the value set in this register.
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
8	SMT_m1	0h	RW	Controls the Schmitt pin of the GPIO pins. 0b: Sets the Schmitt pin to low (Schmitt OFF). 1b: Sets the Schmitt pin to high (Schmitt ON). When the RIIC0 to RIIC8 functions are selected, the Schmitt pin is high, regardless of the value set in this register.
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
0	SMT_m0	0h	RW	Controls the Schmitt pin of the GPIO pins. 0b: Sets the Schmitt pin to low (Schmitt OFF). 1b: Sets the Schmitt pin to high (Schmitt ON). When the RIIC0 to RIIC8 functions are selected, the Schmitt pin is high, regardless of the value set in this register.

“Higher-Order Address” and “Lower-Order Address” at the offset addresses correspond respectively to the left-side 32-bit units and right-side 32 bit-units of the map below.

Table 4.2-26 Corresponding Pins and Offset Addresses of PFC_SMT_mn

		Higher-Order Address				Lower-Order Address			
Offset Address	Register Name	Bit 24	Bit 16	Bit 8	Bit 0	Bit 24	Bit 16	Bit 8	Bit 0
3504h / 3500h	PFC_SMT_20	P07	P06	P05	P04	P03	P02	P01	P00
350Ch / 3508h	PFC_SMT_21	P17	P16	P15	P14	P13	P12	P11	P10
351Ch / 3518h	PFC_SMT_23	P37	P36	P35	P34	P33	P32	P31	P30
3524h / 3520h	PFC_SMT_24	—	—	P45	P44	P43	P42	P41	P40
352Ch / 3528h	PFC_SMT_25	—	P56	P55	P54	P53	P52	P51	P50
3534h / 3530h	PFC_SMT_26	—	P66	P65	P64	P63	P62	P61	P60
353Ch / 3538h	PFC_SMT_27	P77	P76	P75	P74	P73	P72	P71	P70
3544h / 3540h	PFC_SMT_28	—	—	P85	P84	P83	P82	P81	P80
3554h / 3550h	PFC_SMT_2A	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
355Ch / 3558h	PFC_SMT_2B	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
3564h / 3560h	PFC_SMT_2C	—	—	—	—	—	PC2	PC1	PC0
356Ch / 3568h	PFC_SMT_2D	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
3574h / 3570h	PFC_SMT_2E	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
357Ch / 3578h	PFC_SMT_2F	—	—	—	—	—	PF2	PF1	PF0
3584h / 3580h	PFC_SMT_30	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
358Ch / 3588h	PFC_SMT_31	—	—	PH5	PH4	PH3	PH2	PH1	PH0
359Ch / 3598h	PFC_SMT_33	—	—	—	PJ4	PJ3	PJ2	PJ1	PJ0
35A4h / 35A0h	PFC_SMT_34	—	—	—	—	PK3	PK2	PK1	PK0
35ACh / 35A8h	PFC_SMT_35	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0
35B4h / 35B0h	PFC_SMT_36	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0
35E4h / 35E0h	PFC_SMT_3C	—	—	—	—	PS3	PS2	PS1	PS0

4.2.2.17 Port Group Specification Registers (PFC_ELC_PGRg)

Remark Index g = 1 is for the multiplexed pins P1n and g = 2 is for the multiplexed pins P7n (n = 0 to 7).

These registers set the group for I/O ports.

NOTE

This group is hereafter called a “port group”. Each port (corresponding 1 bit) is called a “single port”.

Access Size : 8 bits
Offset Address : <PFC_base> + 3800h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	PGR7	PGR6	PGR5	PGR4	PGR3	PGR2	PGR1	PGR0
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	PGR7	0h	RW	0b: Port group is not specified. 1b: Port group is specified.
6	PGR6	0h	RW	0b: Port group is not specified. 1b: Port group is specified.
5	PGR5	0h	RW	0b: Port group is not specified. 1b: Port group is specified.
4	PGR4	0h	RW	0b: Port group is not specified. 1b: Port group is specified.
3	PGR3	0h	RW	0b: Port group is not specified. 1b: Port group is specified.
2	PGR2	0h	RW	0b: Port group is not specified. 1b: Port group is specified.
1	PGR1	0h	RW	0b: Port group is not specified. 1b: Port group is specified.
0	PGR0	0h	RW	0b: Port group is not specified. 1b: Port group is specified.

Table 4.2-27 PFC_ELC_PGRg Offset Addresses

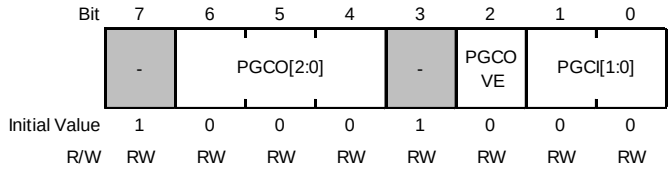
Offset Address	Register Name
3800h	PFC_ELC_PGR1
3801h	PFC_ELC_PGR2

4.2.2.18 Port Group Control Registers (PFC_ELC_PGCg)

Remark Index g = 1 is for the multiplexed pins P1n and g = 2 is for the multiplexed pins P7n (n = 0 to 7).

These registers specify the output format of the signals which are externally output from the ports at the time of input of the event signal for the port group which is set for output. For an input port group, these registers enable or disable overwriting of the PFC_PDBFg register and set the condition for event generation (a change in the signal which is externally input).

Access Size : 8 bits
Offset Address : <PFC_base> + 3802h
Initial Value : 88h



Bit	Bit Name	Initial Value	R/W	Description
7	-	1h	RW	Reserved Whenever it is read, 1 is read. The written value should always be 1b.
6 to 4	PGCO[2:0]	0h	RW	000b: Outputs 0 when an event is input 001b: Outputs 1 when an event is input 010b: Toggle (inverted) output when an event is input (PM_mn[1:0] of the corresponding pin should be set to 11b) 011b: Outputs the buffer value (PDBFg register) when an event is input 1xxb: Setting prohibited
3	-	1h	RW	Reserved Whenever it is read, 1 is read. The written value should always be 1b.
2	PGCOVE	0h	RW	0b: Disables overwriting to the PDBFg register After the CPU read of the PDBFg register is executed, the PDBFg register is updated only by the occurrence of the first event. The PDBFg register is not updated by the occurrence of an event until the next CPU read is executed. 1b: Enables overwrite to the PDBFg register Regardless of the CPU read of the PDBFg register, the PDBFg register is updated by the occurrence of an event.
1, 0	PGCI[1:0]	0h	RW	00b: Detects the rising edge of the external input signal and generates an event. 01b: Detects the falling edge of the external input signal and generates an event. 1xb: Detects both the rising and falling edges of the external input signal and generates an event.

Table 4.2-28 PFC_ELC_PGCg Offset Addresses

Offset Address	Register Name
3802h	PFC_ELC_PGC1
3803h	PFC_ELC_PGC2

4.2.2.19 Port Buffer Registers (PFC_ELC_PDBFg)

Remark Index g = 1 is for the multiplexed pins P1n and g = 2 is for the multiplexed pins P7n (n = 0 to 7).

These 8-bit registers are paired with the PFC_PGRg registers.

Access Size : 8 bits
Offset Address : <PFC_base> + 3804h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	PDBF7	PDBF6	PDBF5	PDBF4	PDBF3	PDBF2	PDBF1	PDBF0
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	PDBF7	0h	RW	The following operations are performed depending on the input/output of the port. - When the port is an output port: The value written to the PDBFg register is transferred to the P_mn register. - When the port is an input port: The signal value of the external pin is transferred to the PDBFg register. Writing to the bits specified for the input port group is disabled.
6	PDBF6	0h	RW	The following operations are performed depending on the input/output of the port. - When the port is an output port: The value written to the PDBFg register is transferred to the P_mn register. - When the port is an input port: The signal value of the external pin is transferred to the PDBFg register. Writing to the bits specified for the input port group is disabled.
5	PDBF5	0h	RW	The following operations are performed depending on the input/output of the port. - When the port is an output port: The value written to the PDBFg register is transferred to the P_mn register. - When the port is an input port: The signal value of the external pin is transferred to the PDBFg register. Writing to the bits specified for the input port group is disabled.
4	PDBF4	0h	RW	The following operations are performed depending on the input/output of the port. - When the port is an output port: The value written to the PDBFg register is transferred to the P_mn register. - When the port is an input port: The signal value of the external pin is transferred to the PDBFg register. Writing to the bits specified for the input port group is disabled.
3	PDBF3	0h	RW	The following operations are performed depending on the input/output of the port. - When the port is an output port: The value written to the PDBFg register is transferred to the P_mn register. - When the port is an input port: The signal value of the external pin is transferred to the PDBFg register. Writing to the bits specified for the input port group is disabled.
2	PDBF2	0h	RW	The following operations are performed depending on the input/output of the port. - When the port is an output port: The value written to the PDBFg register is transferred to the P_mn register. - When the port is an input port: The signal value of the external pin is transferred to the PDBFg register. Writing to the bits specified for the input port group is disabled.
1	PDBF1	0h	RW	The following operations are performed depending on the input/output of the port. - When the port is an output port: The value written to the PDBFg register is transferred to the P_mn register. - When the port is an input port: The signal value of the external pin is transferred to the PDBFg register. Writing to the bits specified for the input port group is disabled.
0	PDBF0	0h	RW	The following operations are performed depending on the input/output of the port. - When the port is an output port: The value written to the PDBFg register is transferred to the P_mn register. - When the port is an input port: The signal value of the external pin is transferred to the PDBFg register. Writing to the bits specified for the input port group is disabled.

Table 4.2-29 PFC_ELC_PDBFg Offset Addresses

Offset Address	Register Name
3804h	PFC_ELC_PDBF1
3808h	PFC_ELC_PDBF2

4.2.2.20 Event Connection Port Specification Registers (PFC_ELC_PELs)

Remark Index s = 0 to 3 is for four single ports.

These registers specify the single ports to which events are linked and set the operation at the time of event input and the condition for event generation. In this LSI, a total of four single ports can be set for any bit of the multiplexed pins P1n and P7n (n = 0 to 7).

Access Size : 8 bits
Offset Address : <PFC_base> + 380Ch
Initial Value : 80h

Bit	7	6	5	4	3	2	1	0
	-	PSM[1:0]		PSP[1:0]		PSB[2:0]		
Initial Value	1	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	-	1h	RW	Reserved These bits are read as 1b. The write value should be 1b.
6 to 5	PSM[1:0]	0h	RW	- When the port is set for output: Specifies the port output data. 00b: Outputs 0b when an event is input 01b: Outputs 1b when an event is input 1xb: Toggle (inverted) output when an event is input (PM_mn[1:0] of the corresponding pin should be set to 11b) - When the port is set for input: Selects the event output edge. 00b: Detects the rising edge and outputs an event. 01b: Detects the falling edge and outputs an event. 1xb: Detects both the rising and falling edges and outputs an event.
4 to 3	PSP[1:0]	0h	RW	00b: Setting prohibited 01b: Multiplexed pin P1n (n = 0 to 7) (Corresponds to the PGR1 register). 10b: Multiplexed pin P7n (n = 0 to 7) (Corresponds to the PGR2 register). 11b: Setting prohibited
2 to 0	PSB[2:0]	0h	RW	Specifies the bit number of the eight I/O ports.

Table 4.2-30 PFC_ELC_PELs Offset Addresses

Offset Address	Register Name
380Ch	PFC_ELC_PEL0
380Dh	PFC_ELC_PEL1
380Eh	PFC_ELC_PEL2
380Fh	PFC_ELC_PEL3

4.2.2.21 Input Edge Detection Control Register (PFC_ELC_DPTC)

Remark Index s = 0 to 3 is for four single ports.

This register specifies enabling or disabling of input edge detection in the single ports to which events are linked.

When single ports s (s = 0 to 3: for four single ports) are specified by the PFC_ELSR2 register, input edge detection must be enabled for the corresponding single ports.

Access Size : 8 bits
Address : <PFC_base> + 3810h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	PTC3	PTC2	PTC1	PTC0
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
3	PTC3	0h	RW	0b: Single input ports, input edge detection disabled 1b: Single input ports, input edge detection enabled
2	PTC2	0h	RW	0b: Single input ports, input edge detection disabled 1b: Single input ports, input edge detection enabled
1	PTC1	0h	RW	0b: Single input ports, input edge detection disabled 1b: Single input ports, input edge detection enabled
0	PTC0	0h	RW	0b: Single input ports, input edge detection disabled 1b: Single input ports, input edge detection enabled

NOTE

The PFC_ELC_DPTC register controls edge detection at the time of single input port operation upon event generation. This has no effect at the time of single output port operation upon event input.

In single input port operation upon event generation,

- When ELC_DPTC.PTCs = 0, the setting has no effect even if an effective edge is input from an external pin.
- When ELC_DPTC.PTCs = 1, the event signal is output to the ICU at the time of event output edge detection specified by the ELC_PELs.PSM bit, if an effective edge is input from an external pin.

4.2.2.22 Port Event Control Register (PFC_ELC_ELSR2)

This register specifies enabling or disabling of each function of the port event controller to which events are linked. When the following peripheral functions are specified by the PFC_ELSR2 register, the corresponding functions must be enabled.

- Input port groups 1 and 2

  g = 1: for the multiplexed pins P1n; g = 2: for the multiplexed pins P7n (n = 0 to 7)

- Single ports 0 to 3

  s = 0 to 3: for four single ports

Access Size : 8 bits
Address : <PFC_base> + 3811h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	PES3	PES2	PES1	PES0	PEG2	PEG1	-	-
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	PES3	0h	RW	0b: Single ports, event link function disabled 1b: Single ports, event link function enabled
6	PES2	0h	RW	0b: Single ports, event link function disabled 1b: Single ports, event link function enabled
5	PES1	0h	RW	0b: Single ports, event link function disabled 1b: Single ports, event link function enabled
4	PES0	0h	RW	0b: Single ports, event link function disabled 1b: Single ports, event link function enabled
3	PEG2	0h	RW	0b: Input port group g, event link function disabled 1b: Input port group g, event link function enabled
2	PEG1	0h	RW	0b: Input port group g, event link function disabled 1b: Input port group g, event link function enabled
1, 0	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.

4.2.2.23 OSC Bypass Mode Switching Register (PFC_OSCBYPS)

This register switches the bypass mode for the RTC oscillator and audio oscillator.

Access Size : 8, 16, 32 bits

Offset Address : <PFC_base> + 3C00h

Initial Value : 000C_0303h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	OSCSF1[1:0]		-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	OSCPW1	OSCPW0	-	-	-	-	-	-	OSCBYPS1	OSCBYPS0
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
19 to 18	OSCSF1[1:0]	3h	RW	Controls the SF[1:0] pins (select frequency pins) of the OSC and the OSCSF1 bit corresponds to the Audio OSC. Frequency Range 00b: (SF0, SF1) = (0, 0) 32 KHz to 1 MHz 10b: (SF0, SF1) = (0, 1) 1.1 MHz to 12 MHz 01b: (SF0, SF1) = (1, 0) 12.1 MHz to 24 MHz 11b: (SF0, SF1) = (1, 1) 24.1 MHz to 48 MHz
17 to 10	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
9	OSCPW1	1h	RW	Controls the power down mode of the Audio OSC. 0b: Power down mode OFF (either the crystal oscillator mode or the external clock receive mode is selected depending on the OSCBYPS1 bit). 1b: Power down mode ON (the power down mode is selected regardless of the OSCBYPS1 bit)
8	OSCPW0	1h	RW	Controls the power down mode of the RTC OSC. 0b: Power down mode OFF (either the crystal oscillator mode or the external clock receive mode is selected depending on the OSCBYPS0 bit). 1b: Power down mode ON (the power down mode is selected regardless of the OSCBYPS0 bit)
7 to 2	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
1	OSCBYPS1	1h	RW	Controls the bypass mode of the Audio OSC. 0b: Bypass mode OFF (the crystal oscillator mode) 1b: Bypass mode ON (the external clock receive mode)
0	OSCBYPS0	1h	RW	Controls the bypass mode of the RTC OSC. 0b: Bypass mode OFF (the crystal oscillator mode) 1b: Bypass mode ON (the external clock receive mode)

The table below lists switching of the mode for the RTC oscillator and audio oscillator.

Table 4.2-31 Bypass Modes for RTC Oscillator and Audio Oscillator

OSCBYPS	OSCPW	Mode
0b	0b	Crystal oscillator mode
1b	0b	External clock receive mode
X	1b	Power-down mode

The main oscillator is controlled by the reset latch output of the external QBYPASS pin, instead of the PFC_OSCBYPS register.

Table 4.2-32 Bypass Modes for Main Oscillator

QBYPASS	Mode
Low	Crystal oscillator mode
High	External clock receive mode

4.2.2.24 Write Protect Register (PFC_PWPR)

This register enables or disables writing to the PFC_PFC or PFC_PMC registers and the PFC_OEN registers.

Access Size : 8, 16, 32 bits
Address : <PFC_base> + 3C04h
Initial Value : 0000_00A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	REGWE_A	REGWE_B	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
7	-	1h	RW	Reserved Whenever it is read, 1 is read. The written value should always be 1b.
6	REGWE_A	0h	RW	Controls writing to PFC and PMC, which are write-protected in the initial state. 0b: Disables writing to the PFC and PMC registers. 1b: Enables writing to the PFC and PMC registers.
5	REGWE_B	1h	RW	Controls writing to the OEN register, which is initially write-enabled. 0b: Disables writing to the OEN register. 1b: Enables writing to the OEN register.
4 to 0	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.

4.2.2.25 IRQ Digital Noise Filter (FILONOFF) Register (PFC_FILONOFF_IRQ)

This register sets whether or not to use the digital noise filter for the IRQ function.

The register function is the same as that of the PFC_FILONOFF register.

Access Size : 8, 16, 32 bits

Address : <PFC_base> + 3C08h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	IRQ15	-	IRQ14	-	IRQ13	-	IRQ12	-	IRQ11	-	IRQ10	-	IRQ9	-	IRQ8
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	IRQ7	-	IRQ6	-	IRQ5	-	IRQ4	-	IRQ3	-	IRQ2	-	IRQ1	-	IRQ0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
30	IRQ15	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
29	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
28	IRQ14	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
27	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
26	IRQ13	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
25	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
24	IRQ12	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
23	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
22	IRQ11	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
21	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
20	IRQ10	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
19	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
18	IRQ9	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
17	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
16	IRQ8	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter

Bit	Bit Name	Initial Value	R/W	Description
15	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
14	IRQ7	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
13	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
12	IRQ6	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
11	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
10	IRQ5	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
9	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
8	IRQ4	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
7	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
6	IRQ3	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
5	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
4	IRQ2	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
3	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
2	IRQ1	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter
1	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
0	IRQ0	0h	RW	Sets whether or not to use the digital noise filter for the IRQ function. 0b: No filter 1b: With filter

NOTE

The IRQ function can be used as an interrupt or an event for the ELC. Before setting this register, disable the IRQ function depending on the purpose for which it is to be used.

- If the IRQ function is to be used as an interrupt, disable the function by using the interrupt controller of the CPU.
- If the IRQ function is to be used as an event for the ELC, disable the function by using the event output source select register (EVTSELx) of the ICU.

4.2.2.26 IRQ Digital Noise Filter (FILNUM) Register (PFC_FILNUM_IRQ)

This register sets the number of stages of the digital noise filter for the IRQ function.

The register function is the same as that of the PFC_FILNUM register.

Access Size : 8, 16, 32 bits
Address : <PFC_base> + 3C0Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IRQ15[1:0]		IRQ14[1:0]		IRQ13[1:0]		IRQ12[1:0]		IRQ11[1:0]		IRQ10[1:0]		IRQ9[1:0]		IRQ8[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ7[1:0]		IRQ6[1:0]		IRQ5[1:0]		IRQ4[1:0]		IRQ3[1:0]		IRQ2[1:0]		IRQ1[1:0]		IRQ0[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	IRQ15[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
29, 28	IRQ14[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
27, 26	IRQ13[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
25, 24	IRQ12[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
23, 22	IRQ11[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
21, 20	IRQ10[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
19, 18	IRQ9[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
17, 16	IRQ8[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages

Bit	Bit Name	Initial Value	R/W	Description
15, 14	IRQ7[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
13, 12	IRQ6[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
11, 10	IRQ5[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
9, 8	IRQ4[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
7, 6	IRQ3[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
5, 4	IRQ2[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
3, 2	IRQ1[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
1, 0	IRQ0[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the IRQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages

NOTE

The IRQ function can be used as an interrupt or an event for the ELC. Before setting this register, disable the IRQ function depending on the purpose for which it is to be used.

- If the IRQ function is to be used as an interrupt, disable the function by using the interrupt controller of the CPU.
- If the IRQ function is to be used as an event for the ELC, disable the function by using the event output source select register (EVTSELx) of the ICU.

4.2.2.27 IRQ Digital Noise Filter (FILCLKSEL) Register (PFC_FILCLKSEL_IRQ)

This register sets the sampling interval to be input to the digital noise filter for the IRQ function. The sampling interval is the time obtained by frequency division for the SHCLK input clock.

The register function is the same as that of the PFC_FILCLKSEL register.

Access Size : 8, 16, 32 bits
Address : <PFC_base> + 3C10h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IRQ15[1:0]		IRQ14[1:0]		IRQ13[1:0]		IRQ12[1:0]		IRQ11[1:0]		IRQ10[1:0]		IRQ9[1:0]		IRQ8[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ7[1:0]		IRQ6[1:0]		IRQ5[1:0]		IRQ4[1:0]		IRQ3[1:0]		IRQ2[1:0]		IRQ1[1:0]		IRQ0[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	IRQ15[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
29, 28	IRQ14[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
27, 26	IRQ13[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
25, 24	IRQ12[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
23, 22	IRQ11[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
21, 20	IRQ10[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
19, 18	IRQ9[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
17, 16	IRQ8[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256

Bit	Bit Name	Initial Value	R/W	Description
15, 14	IRQ7[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
13, 12	IRQ6[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
11, 10	IRQ5[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
9, 8	IRQ4[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
7, 6	IRQ3[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
5, 4	IRQ2[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
3, 2	IRQ1[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
1, 0	IRQ0[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter for the IRQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256

NOTE

The IRQ function can be used as an interrupt or an event for the ELC. Before setting this register, disable the IRQ function depending on the purpose for which it is to be used.

- If the IRQ function is to be used as an interrupt, disable the function by using the interrupt controller of the CPU.
- If the IRQ function is to be used as an event for the ELC, disable the function by using the event output source select register (EVTSELx) of the ICU.

4.2.2.28 DMAC_REQ Digital Noise Filter (FILONOFF) Register (PFC_FILONOFF_DMAC_REQ)

This register sets whether or not to use the digital noise filter for the DREQ function.

The register function is the same as that of the PFC_FILONOFF register.

Access Size : 8, 16, 32 bits
Address : <PFC_base> + 3C14h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	DMAC_REQ4	-	DMAC_REQ3	-	DMAC_REQ2	-	DMAC_REQ1	-	DMAC_REQ0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
8	DMAC_REQ4	0h	RW	Sets whether or not to use the digital noise filter for the DREQ function. 0b: No filter 1b: With filter
7	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
6	DMAC_REQ3	0h	RW	Sets whether or not to use the digital noise filter for the DREQ function. 0b: No filter 1b: With filter
5	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
4	DMAC_REQ2	0h	RW	Sets whether or not to use the digital noise filter for the DREQ function. 0b: No filter 1b: With filter
3	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
2	DMAC_REQ1	0h	RW	Sets whether or not to use the digital noise filter for the DREQ function. 0b: No filter 1b: With filter
1	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
0	DMAC_REQ0	0h	RW	Sets whether or not to use the digital noise filter for the DREQ function. 0b: No filter 1b: With filter

NOTE

Before setting this register, disable the function by using the DMAx source select register y (DMxSELy) of the ICU.

4.2.2.29 DMAC_REQ Digital Noise Filter (FILNUM) Register (PFC_FILNUM_DMAC_REQ)

This register sets the number of stages of the digital noise filter for the DREQ function.

The register function is the same as that of the PFC_FILNUM register.

Access Size : 8, 16, 32 bits

Address : <PFC_base> + 3C18h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	DMAC_REQ4 [1:0]		DMAC_REQ3 [1:0]		DMAC_REQ2 [1:0]		DMAC_REQ1 [1:0]		DMAC_REQ0 [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
9, 8	DMAC_REQ4 [1:0]	0h	RW	Sets the number of stages of the digital noise filter for the DREQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
7, 6	DMAC_REQ3 [1:0]	0h	RW	Sets the number of stages of the digital noise filter for the DREQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
5, 4	DMAC_REQ2 [1:0]	0h	RW	Sets the number of stages of the digital noise filter for the DREQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
3, 2	DMAC_REQ1 [1:0]	0h	RW	Sets the number of stages of the digital noise filter for the DREQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages
1, 0	DMAC_REQ0 [1:0]	0h	RW	Sets the number of stages of the digital noise filter for the DREQ function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages

NOTE

Before setting this register, disable the function by using the DMAx source select register y (DMxSELy) of the ICU.

4.2.2.30 DMAC_REQ Digital Noise Filter (FILCLKSEL) Register (PFC_FILCLKSEL_DMAC_REQ)

This register sets the sampling interval to be input to the digital noise filter for the DREQ function. The sampling interval is the time obtained by frequency division for the SHCLK input clock.

The register function is the same as that of the PFC_FILCLKSEL register.

Access Size : 8, 16, 32 bits

Address : <PFC_base> + 3C1Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	DMAC_REQ4 [1:0]		DMAC_REQ3 [1:0]		DMAC_REQ2 [1:0]		DMAC_REQ1 [1:0]		DMAC_REQ0 [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
9, 8	DMAC_REQ4 [1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter of the DREQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
7, 6	DMAC_REQ3 [1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter of the DREQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
5, 4	DMAC_REQ2 [1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter of the DREQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
3, 2	DMAC_REQ1 [1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter of the DREQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256
1, 0	DMAC_REQ0 [1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter of the DREQ function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256

NOTE

Before setting this register, disable the function by using the DMAx source select register y (DMxSELy) of the ICU.

4.2.2.31 NMI Digital Noise Filter (FILONOFF) Register (PFC_FILONOFF_NMI)

This register sets whether or not to use the digital noise filter for the NMI function.

Access Size : 8, 16, 32 bits
Address : <PFC_base> + 3C20h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	NMI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
0	NMI	0h	RW	Sets whether or not to use the digital noise filter for the NMI function. 0b: No filter 1b: With filter

4.2.2.32 NMI Digital Noise Filter (FILNUM) Register (PFC_FILNUM_NMI)

This register sets the number of stages of the digital noise filter for the NMI function.

Access Size : 8, 16, 32 bits

Address : <PFC_base> + 3C24h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	NMI[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
1 to 0	NMI[1:0]	0h	RW	Sets the number of stages of the digital noise filter for the NMI function. 00b: 4 stages 01b: 8 stages 10b: Setting prohibited 11b: 16 stages

4.2.2.33 NMI Digital Noise Filter (FILCLKSEL) Register (PFC_FILCLKSEL_NMI)

This register sets the sampling interval to be input to the digital noise filter for the NMI function. The sampling interval is the time obtained by frequency division for the SHCLK input clock.

Access Size : 8, 16, 32 bits
Address : <PFC_base> + 3C28h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	NMI[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
1 to 0	NMI[1:0]	0h	RW	Sets the sampling interval to be input to the digital noise filter of the NMI function. 00b: Divided by 4 01b: Divided by 32 10b: Divided by 128 11b: Divided by 256

4.2.2.34 OEN Switching Register (PFC_OEN)

This register sets the direction (input or output) of the IO block.

Access Size : 8, 16, 32 bits
Address : <PFC_base> + 3C40h
Initial Value : 0000_003Fh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	OEN5	OEN4	OEN3	OEN2	OEN1	OEN0
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
5	OEN5	1h	RW	Controls the OE pin of IO block for the following dedicated pins (3ST attribute). The values in parentheses indicate the corresponding bits. XSPI0_CKP (OEN5 bit) XSPI0_CKN (OEN4 bit) XSPI0_CS0N (OEN3 bit) XSPI0_RESET0N (OEN2 bit) 0b: OE of IO block is ON (enabled). 1b: OE of IO block is OFF (disabled).
4	OEN4	1h	RW	Controls the OE pin of IO block for the following dedicated pins (3ST attribute). The values in parentheses indicate the corresponding bits. XSPI0_CKP (OEN5 bit) XSPI0_CKN (OEN4 bit) XSPI0_CS0N (OEN3 bit) XSPI0_RESET0N (OEN2 bit) 0b: OE of IO block is ON (enabled). 1b: OE of IO block is OFF (disabled).
3	OEN3	1h	RW	Controls the OE pin of IO block for the following dedicated pins (3ST attribute). The values in parentheses indicate the corresponding bits. XSPI0_CKP (OEN5 bit) XSPI0_CKN (OEN4 bit) XSPI0_CS0N (OEN3 bit) XSPI0_RESET0N (OEN2 bit) 0b: OE of IO block is ON (enabled). 1b: OE of IO block is OFF (disabled).
2	OEN2	1h	RW	Controls the OE pin of IO block for the following dedicated pins (3ST attribute). The values in parentheses indicate the corresponding bits. XSPI0_CKP (OEN5 bit) XSPI0_CKN (OEN4 bit) XSPI0_CS0N (OEN3 bit) XSPI0_RESET0N (OEN2 bit) 0b: OE of IO block is ON (enabled). 1b: OE of IO block is OFF (disabled).

Bit	Bit Name	Initial Value	R/W	Description
1	OEN1	1h	RW	Controls the direction (MII/RGMII) of IO block for ET0/1_TXC_TXCLK (IO attribute). The values in parentheses indicate the corresponding bits. ET1_TXC_TXCLK (OEN1 bit) ET0_TXC_TXCLK (OEN0 bit) 0b: The direction of IO block is OUT (output); RGMII mode. 1b: The direction of IO block is IN (input), MII mode.
0	OEN0	1h	RW	Controls the direction (MII/RGMII) of IO block for ET0/1_TXC_TXCLK (IO attribute). The values in parentheses indicate the corresponding bits. ET1_TXC_TXCLK (OEN1 bit) ET0_TXC_TXCLK (OEN0 bit) 0b: The direction of IO block is OUT (output); RGMII mode. 1b: The direction of IO block is IN (input), MII mode.

NOTE

- OEN0 and OEN1
  When switching between the MII and RGMII of GBETH0 and GBETH1, the register setting on the IP side of GBETH0 and GBETH1 is used, but this does not control the direction of the I/O buffer. Use this register to switch the direction of the IO block. For the setting procedure, see **6.3 Gigabit Ethernet (GBETH)**.
- When setting the OEN0 and OEN1 bits to 0 (to set the direction of the IO block as an output), set the clock switching register bit of the CPG listed in the table below to 0 (PLLETH+OSC path side).

Table 4.2-33 CPG Register for OEN0 and OEN1 Bits

OEN Bit	OSCPW	Clock Switching Register (CPG)
OEN1	ET1_TXC_TXCLK	CPG_SSEL1.SELCTL0 bit
OEN0	ET0_TXC_TXCLK	CPG_SSEL0.SELCTL2 bit

4.2.3 Functional Description

This section describes the functional details of the PFC.

4.2.3.1 IO Block

4.2.3.1.1 List of IO blocks

The table below lists the IO blocks. For the pins and the corresponding registers, see **4.2.2 Register Description**.

Table 4.2-34 IO Blocks

Classification	Drive Strength Switching	Pull-Up/Down Switching	Power Supply Voltage Switching	CMOS/ Schmitt Switching	Slew-Rate Switching	OSC Mode Switching	Input Frequency Switching
1.8-V/3.3-V bidirectional buffer*1	✓	✓	✓	✓	✓	—	—
1.8-V dedicated buffer (bidirectional)	✓	✓	—	—	—	—	—
1.8-V dedicated buffer (OSC)	—	—	—	—	—	✓	✓*2
1.8-V analog buffer	—	—	—	—	—	—	—
Power supply control (3.3-V system)	—	—	—	—	—	—	—
Power supply control (1.8-V system)	—	—	—	—	—	—	—

Note 1. All Pxx pins have a 3.3-V tolerance. Even when 1.8 V is being supplied for the power group, 3.3-V signals can be captured. When N-ch. open drain outputs are specified by the PFC_NOD_mn register, 3.3-V signals can be output by pulling them up to 3.3 V.

Note 2. The RTC oscillator is excluded.

4.2.3.2 IO_TOP_ISEL_CTL

In port mode (PMC_mn = 0b), input control of the IO block is controlled by the PFC_PM_mn register. The input mode can also forcibly be set by the PFC_ISEL_mn register.

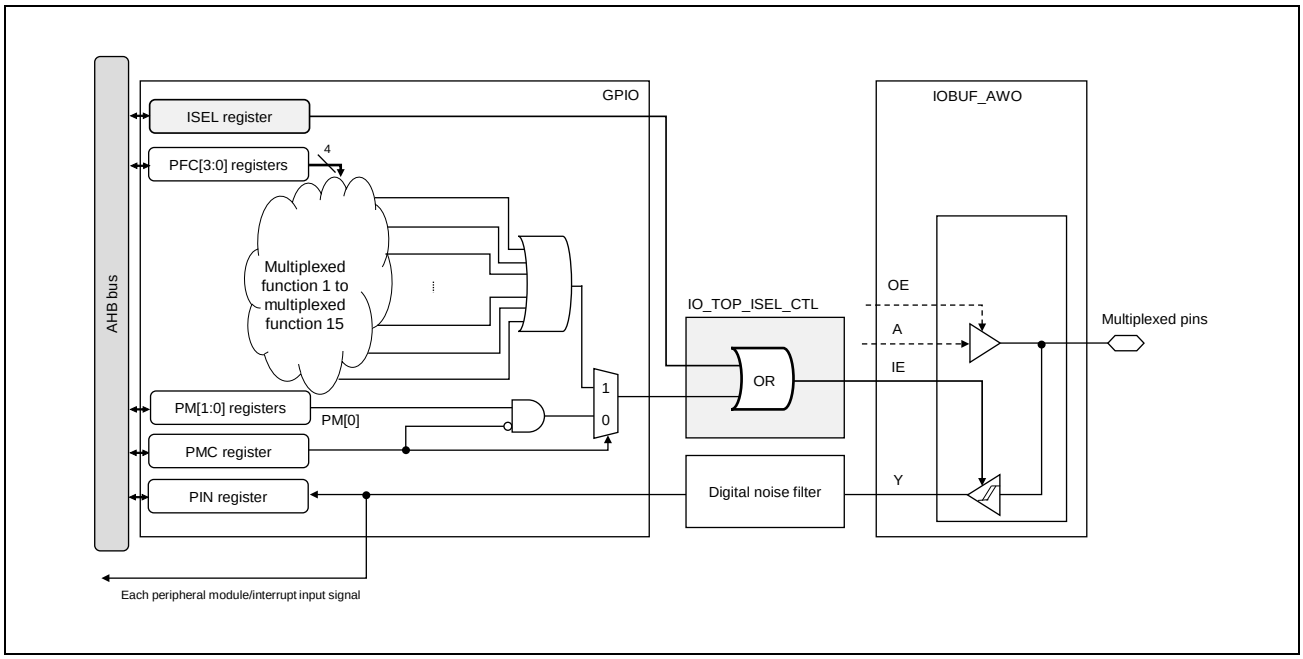


Figure 4.2-2 IO_TOP_ISEL_CTL Block Diagram

NOTE

When PMC_mn = 1b or the event link controller is to be used, setting ISEL_mn = 1b is prohibited.

4.2.3.3 IO_TOP_SMT_CTL

In control mode (PMC_mn = 1b), the Schmitt input signal (IS) of the IO block can be switched by using the PFC_SMT_mn register.

- The multiplexed pin functions are the targets for switching.
- When the function from RIIC0 to RIIC8 is selected, set the Schmitt input signal (IS) of the IO block to the high level to enable the Schmitt function, regardless of the setting of the PFC_SMT_mn register.
- The I3C0 function does not apply. The I3C0 function is determined by the setting of the PFC_SMT_mn register.

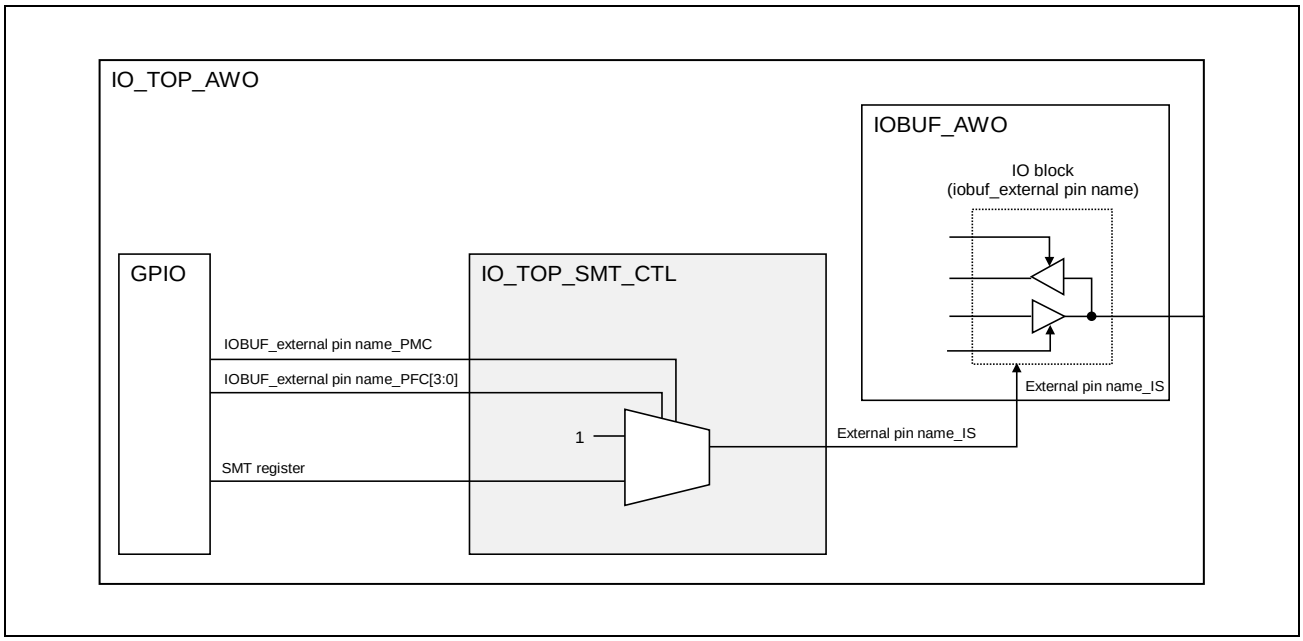


Figure 4.2-3 IO_TOP_SMT_CTL Block Diagram

4.2.3.4 IO_TOP_NOD_CTL

The PFC_NOD_mn register can control the switching to N-ch open-drain output.

When the value of the PFC_NOD_mn register is 1b, it controls the data input signal (A) of the IO block as well as the output data enable signal (OE). The control circuit is allocated to a later stage than the multiplexed function switching circuit of GPIO so that N-ch. open drain outputs can also be controlled in port mode.

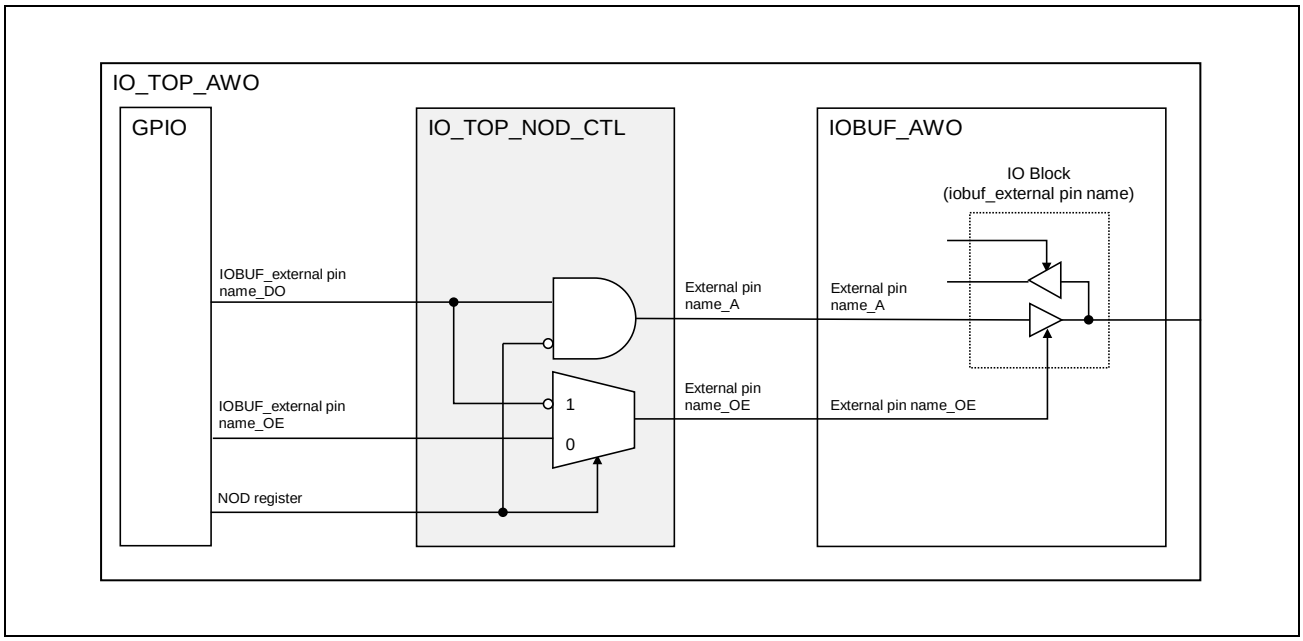


Figure 4.2-4 IO_TOP_NOD_CTL Block Diagram

4.2.3.5 RST_LATCH

RST_LATCH captures data on the corresponding pins during a system reset (QRESN = low) and passes the data to the SYS or CPG. The data are retained through release from the system reset (QRESN = high).

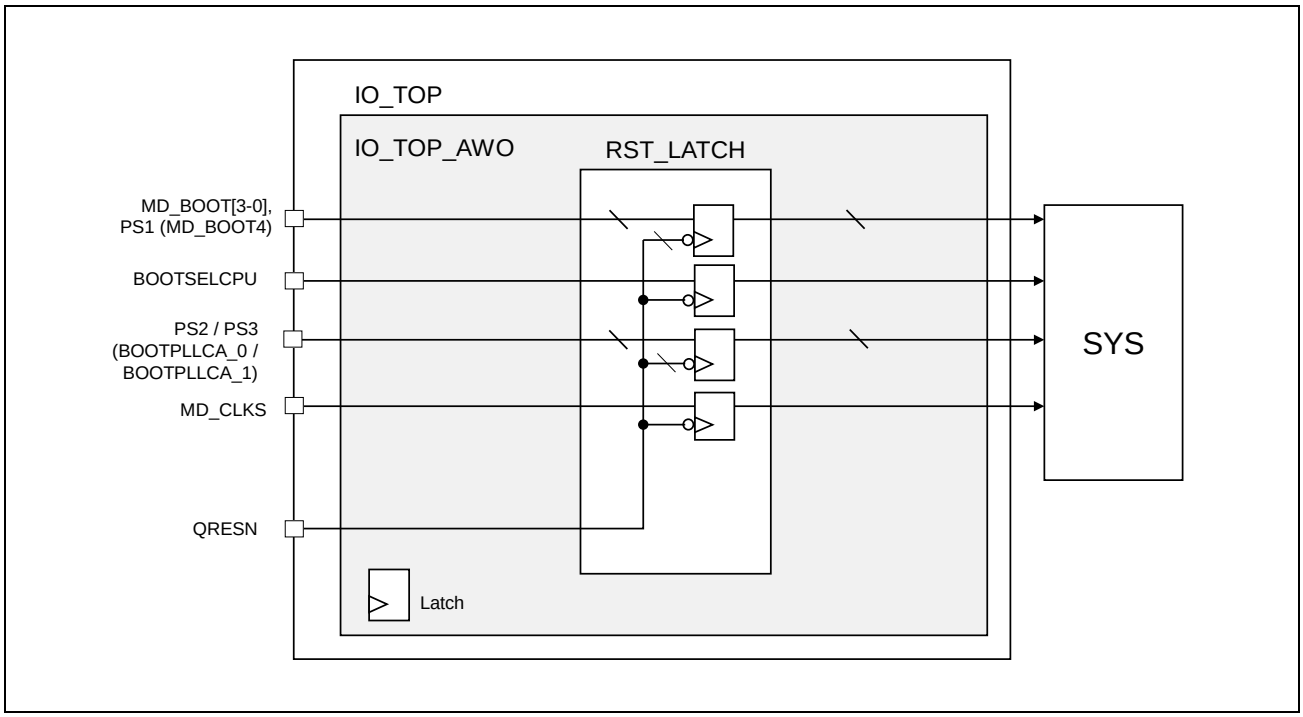


Figure 4.2-5 Schematic Diagram of RST_LATCH

The figure below shows the timing chart.

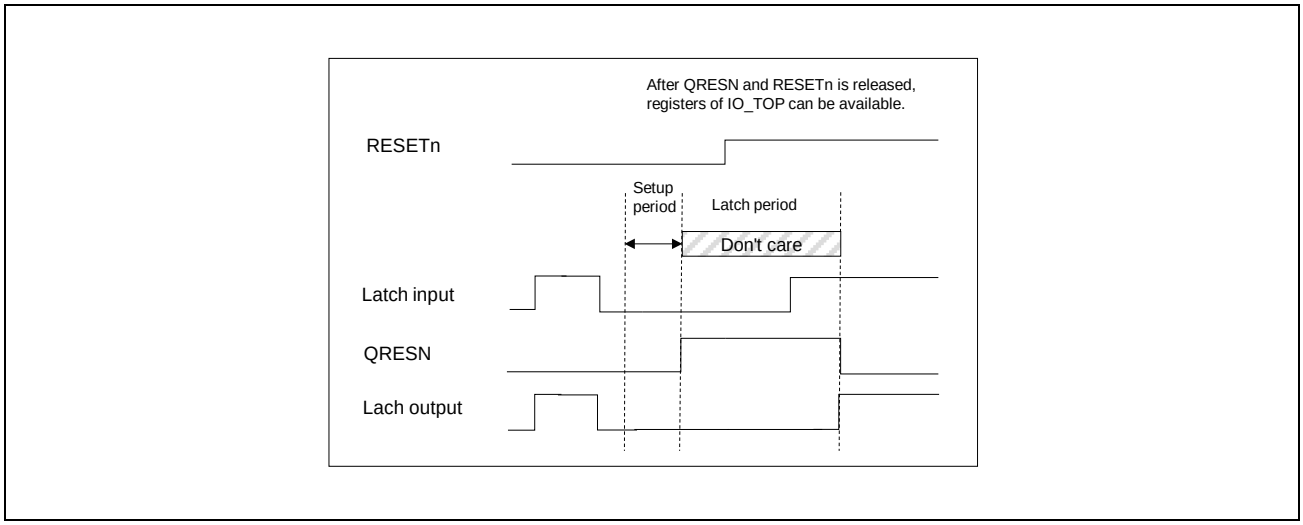


Figure 4.2-6 RST_LATCH Timing Chart

4.2.3.6 DFIL_TOP

The PFC has a digital noise filter function.

The PFC_FILONOFF_mn register sets whether or not to use the digital noise filter, the PFC_FILNUM_mn register sets the number of stages of the filter, and the PFC_FILCLKSEL_mn register sets the sampling interval of the filter.

Among the targets for control listed in **4.2.1.5 Digital Noise Filter Control**, the targets for digital noise filter control by DFIL_TOP are as follows (that is, all except for the port event control pin functions).

- Multiplexed NMI pin
- Input direction line for GPIO
- External TINT interrupt
- Multiplexed IRQ function
- Multiplexed DREQ function

4.2.4 Operation Procedure

4.2.4.1 Control of Dedicated and Multiplexed Pins

1. Set the PFC_IOLH_mn, PFC_SR_mn, PFC_IEN_mn, PFC_PUPD_mn, PFC_NOD_mn, and PFC_OEN registers for the desired functions.

Remark Changing the setting of the PFC_OEN register requires manipulation of the PWPR.REGWE_B bit.
- Before changing the setting of the PFC_OEN register: Set the PWPR.REGWE_B bit to 1b.
- After changing the setting of the PFC_OEN register: Set the PWPR.REGWE_B bit to 0b.

4.2.4.2 Multiplexed Pins (in Port Mode): GPIO Control and Control of the Pins

1. Check that the value of the PFC_PMC_mn register is 0b.

Remark Changing the setting of the PFC_PMC_mn register requires manipulation of the PWPR.REGWE_A bit.
- Before changing the setting of the PFC_PMC_mn register: Set the PWPR.REGWE_A bit to 1b.
- After changing the setting of the PFC_PMC_mn register: Set the PWPR.REGWE_A bit to 0b.

2. Set the input/output of GPIO by using the PFC_PM_mn register.
3. Set output values on GPIO by using the PFC_P_mn register.
4. Input values on the input pins can be monitored by using the PFC_PIN_mn register.
5. Set the PFC_IOLH_mn, PFC_SR_mn, PFC_ISEL_mn, PFC_PUPD_mn, PFC_NOD_mn, and PFC_SMT_mn registers for the desired functions.

4.2.4.3 Multiplexed Pins (in Control Mode): Switching between Functions Multiplexed on the Pins and Control of the Pins

1. Check that the value of the PFC_PMC_mn register is 1b.

Remark Changing the setting of the PFC_PMC_mn register requires manipulation of the PWPR.REGWE_A bit.
- Before changing the setting of the PFC_PMC_mn register: Set the PWPR.REGWE_A bit to 1b.

2. Set the multiplexed function mode by using the PFC_PFC_mn register.

Remark Changing the setting of the PFC_PFC_mn register requires manipulation of the PWPR.REGWE_A bit.
- Before changing the setting of the PFC_PFC_mn register: Set the PWPR.REGWE_A bit to 1b.
  If the PWPR.REGWE_A bit has been set to 1b in step 1, re-setting is not required.
- After changing the setting of the PFC_PFC_mn register: Set the PWPR.REGWE_A bit to 0b.

3. Set the PFC_IOLH_mn, PFC_SR_mn, PFC_IEN_mn, PFC_PUPD_mn, PFC_NOD_mn, and PFC_SMT_mn registers for the desired functions.

Remark When making the settings, do not select the same multiplexed function for several different external pins.

The table below lists the assignment of the multiplexed IRQ and DREQ functions.

For example, the IRQ0 multiplexed function can be assigned to the external P13, P33, P81, PA5, PC0, PG7, PL0, or PS0 pin. However, do not assign it to two or more external pins.

Similarly, the DREQ0 multiplexed function can be assigned to P44, PC1, or PJ3, but do not assign it to two or more external pins.

Table 4.2-35 Assignment of IRQ and DMAC_REQ (1/2)

Pin	IRQ (Mode 15)	DREQ (Mode 13)	Pin	IRQ (Mode 15)	DREQ (Mode 13)	Pin	IRQ (Mode 15)	DREQ (Mode 13)
P00	IRQ5	—	P53	IRQ7	—	PB2	IRQ5	—
P01	IRQ6	—	P54	IRQ8	—	PB3	IRQ6	—
P02	IRQ7	—	P55	IRQ9	—	PB4	IRQ7	—
P03	IRQ8	DREQ1	P56	IRQ10	—	PB5	IRQ8	—
P04	IRQ9	—	P60	IRQ11	—	PB6	IRQ9	—
P05	IRQ10	—	P61	IRQ12	—	PB7	IRQ10	—
P06	IRQ11	DREQ2	P62	IRQ13	—	PC0	IRQ0	—
P07	IRQ12	—	P63	IRQ14	—	PC1	IRQ1	DREQ0
P10	IRQ13	—	P64	IRQ15	—	PC2	IRQ2	—
P11	IRQ14	DREQ3	P65	IRQ5	—	PD0	IRQ3	—
P12	IRQ15	—	P66	IRQ6	—	PD1	IRQ4	DREQ1
P13	IRQ0	—	P70	IRQ7	—	PD2	IRQ5	—
P14	IRQ1	DREQ4	P71	IRQ8	—	PD3	IRQ6	—
P15	IRQ2	—	P72	IRQ9	—	PD4	IRQ7	DREQ2
P16	IRQ3	—	P73	IRQ10	—	PD5	IRQ8	—
P17	IRQ4	—	P74	IRQ11	—	PD6	IRQ9	—
P20	IRQ11	—	P75	IRQ12	—	PD7	IRQ10	DREQ3
P21	IRQ12	—	P76	IRQ13	—	PE0	IRQ11	—
P30	IRQ13	—	P77	IRQ14	—	PE1	IRQ12	—
P31	IRQ14	—	P80	IRQ15	—	PE2	IRQ13	DREQ4
P32	IRQ15	—	P81	IRQ0	—	PE3	IRQ14	—
P33	IRQ0	—	P82	IRQ1	—	PE4	IRQ15	—
P34	IRQ1	—	P83	IRQ2	—	PE5	IRQ10	—
P35	IRQ2	—	P84	IRQ3	—	PE6	IRQ11	—
P36	IRQ3	—	P85	IRQ4	—	PE7	IRQ12	—
P37	IRQ4	—	PA0	IRQ11	—	PF0	IRQ13	—
P40	IRQ5	—	PA1	IRQ12	—	PF1	IRQ14	—
P41	IRQ6	—	PA2	IRQ13	—	PF2	IRQ15	—
P42	IRQ7	—	PA3	IRQ14	—	PG0	IRQ9	—
P43	IRQ8	—	PA4	IRQ15	—	PG1	IRQ10	—
P44	IRQ9	DREQ0	PA5	IRQ0	—	PG2	IRQ11	—
P45	IRQ10	—	PA6	IRQ1	—	PG3	IRQ12	—
P50	IRQ4	—	PA7	IRQ2	—	PG4	IRQ13	—
P51	IRQ5	—	PB0	IRQ3	—	PG5	IRQ14	—
P52	IRQ6	—	PB1	IRQ4	—	PG6	IRQ15	—

Table 4.2-35 Assignment of IRQ and DMAC_REQ (2/2)

Pin	IRQ (Mode 15)	DREQ (Mode 13)
PG7	IRQ0	—
PH0	IRQ1	—
PH1	IRQ2	—
PH2	IRQ3	—
PH3	IRQ4	—
PH4	IRQ5	—
PH5	IRQ6	—
PJ0	IRQ4	—
PJ1	IRQ5	—
PJ2	IRQ6	—
PJ3	IRQ7	DREQ0
PJ4	IRQ8	—
PK0	IRQ7	—
PK1	IRQ8	—
PK2	IRQ9	—
PK3	IRQ10	—
PL0	IRQ0	—
PL1	IRQ1	—
PL2	IRQ2	—
PL3	IRQ3	—
PL4	IRQ12	—
PL5	IRQ13	—
PL6	IRQ14	—
PL7	IRQ15	—
PM0	IRQ4	—
PM1	IRQ5	—
PM2	IRQ6	—
PM3	IRQ7	—
PM4	IRQ8	—
PM5	IRQ9	—
PM6	IRQ10	—
PM7	IRQ11	—
PS0	IRQ0	—
PS1	IRQ1	—
PS2	IRQ2	—
PS3	IRQ3	—

4.2.4.4 OSC Mode Switching

1. Each register of OSCBYPS handles switching the mode between crystal oscillator mode and external clock receive mode for the RTC oscillator and audio oscillator.

Remark Since the main oscillator requires determination of the mode before the CPU starts up (before the PLL starts up), it is controlled by the external QBYPASS pin.

4.2.4.5 Reset Latch Function

No procedure is required for this function. Data on the corresponding pins are captured during a system reset (QRESN = low) and retained through release from the system reset (QRESN = high).

4.2.4.6 Digital Noise Filter Control

1. Set the PFC_FILONOFF_mn, PFC_FILNUM_mn, PFC_FILCLKSEL_mn, PFC_FILONOFF_IRQ, PFC_FILNUM_IRQ, PFC_FILCLKSEL_IRQ, PFC_FILONOFF_DMAC_REQ, PFC_FILNUM_DMAC_REQ, and PFC_FILCLKSEL_DMAC_REQ, FILONOFF_NMI, FILNUM_NMI and FILCLKSEL_NMI registers for the desired functions.

4.2.4.7 Event Link Controller — Port Event Control (PFC_ELC_GPIO)

The I/O port pins that can be assigned to event linking are the multiplexed pins P1n and P7n (n = 0 to 7).

1. Set the following PFC_ELC_GPIO registers for the desired functions.
   PFC_ELC_PGRg, PFC_ELC_PGCg, PFC_ELC_PDBFg, PFC_ELC_PELs, PFC_ELC_DPTC, and PFC_ELC_ELSR2 registers

Remark The event link controller can only be used while PFC_PMC_mn = 0b (in port mode).

There are two types of event link: an event link to a single port and an event link to a port group.

- Single Port: An event can be connected to eight I/O ports.
  Set the bits of the I/O ports to which events are linked by specification with the use of the PFC_ELC_PELs register.
- Port Group: An event can be connected to a port group consisting of the desired bits in the same eight I/O ports.
  Set the bits of the I/O ports to which events are linked by specification with the use of the PFC_ELC_PGRg register.

(1) to (6) below describe operations of a single port and a port group and (7) describes an example of operation.

(1) Single Input Port Operation upon Event Generation

For a single input port, an event is generated when a change in the signal as the condition for event generation is detected on the external pin that is connected (the event output is a pulse for one SHCLK cycle). The condition for event generation is set by the PSM bit of the PFC_ELC_PELs register. **Figure 4.2-7** shows an example of event link operation by a single input port.

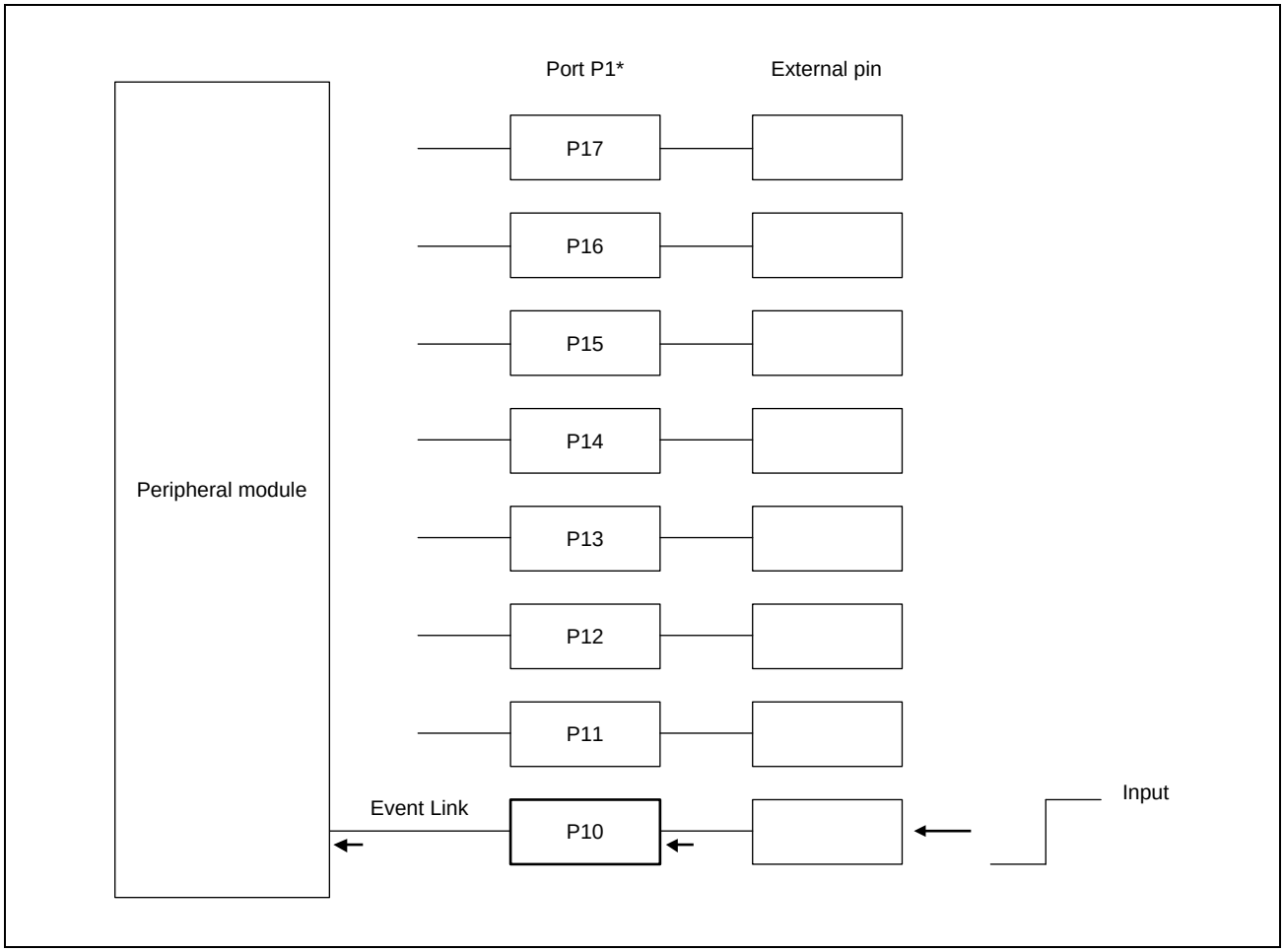


Figure 4.2-7 Single Input Port Operation

(2) Single Output Port Operation upon Event Input

For a single output port, the external pin output changes in the assigned way in response to the input of an event signal (the rising edge condition). The change in the external pin output is set by the PSM bit of the PFC_ELC_PELs register. **Figure 4.2-8** shows an example of event link operation by a single output port.

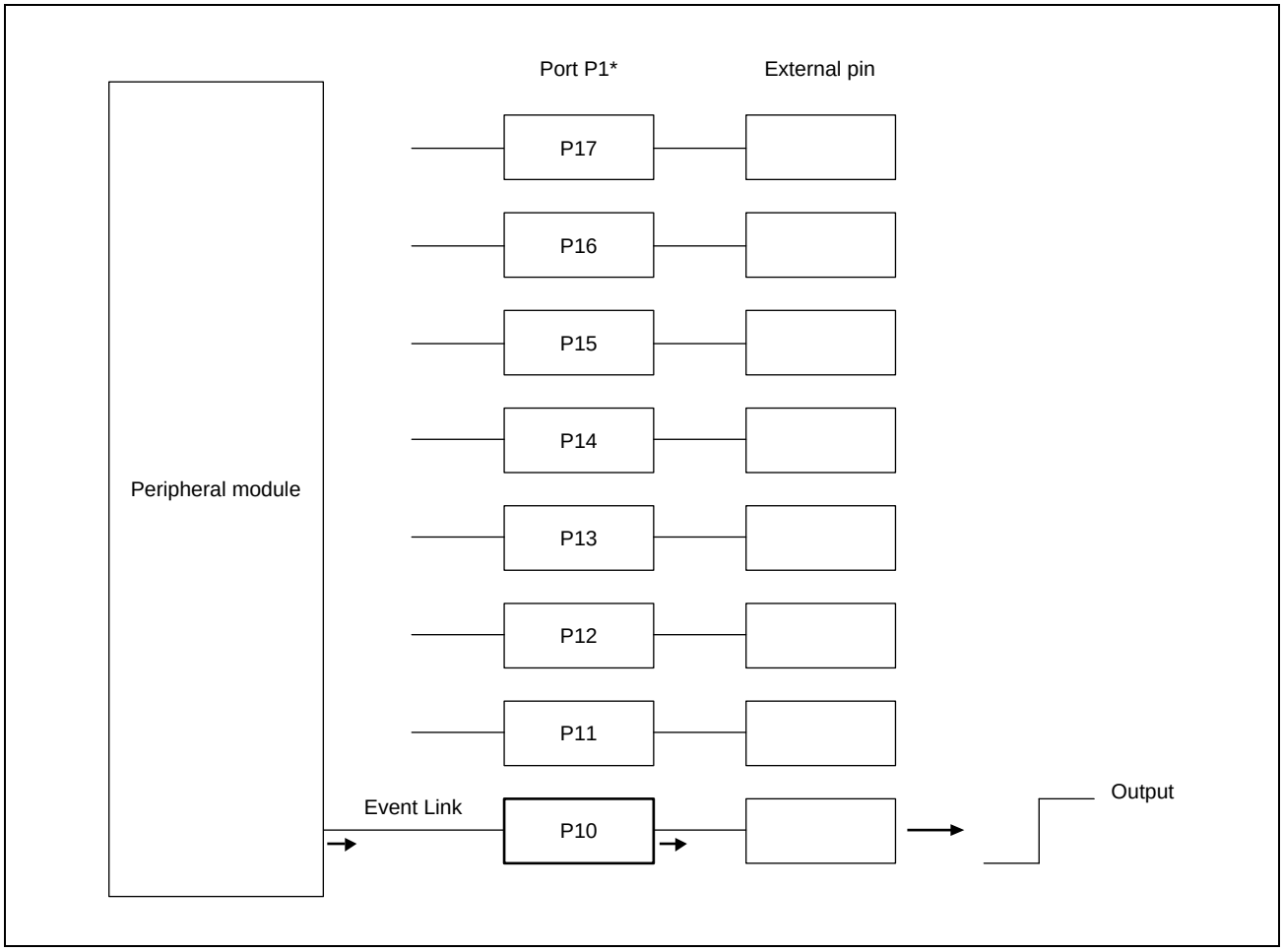


Figure 4.2-8 Single Output Port Operation

NOTE

In single output port operation, if the same port is specified by the ELC_PELs.PSB bit for four single ports and different port output data are specified by the ELC_PELs.PSM bits for the single ports for which the same port has been specified, contention among port outputs may arise depending on the timing of event input.

(3) Input Port Group Operation upon Event Generation

In an input port group, an event is generated when the change in the signal that has been selected as the condition for event operation is detected on one of the external pins that is connected to the port group. The condition for event generation is set by the PGCI bit of the PFC_ELC_PGCg register.

(4) Input Port Group Operation upon Event Input

When an event signal (the rising edge condition) is input to an input port group, the signal value on the external pin at the time of event input is transferred to the PFC_ELC_PDBFg register. Furthermore, if another event is input to the input port group, operation proceeds in accord with the setting of the PGCOVE bit of the PFC_ELC_PGCg register. **Figure 4.2-9** shows an example of operation.

- ELC_PGCg.PGCOVE = 0b (overwriting of the PFC_ELC_PDBFg register is disabled)
  If the value which has been transferred to the PFC_ELC_PDBFg register at the time of the previous event input has already been read by the CPU, the signal value on the external pin will be transferred to the PFC_ELC_PDBFg register. If it has not been read, the value on the external pin will not be transferred and the input event will be invalid.
- ELC_PGCg.PGCOVE = 1b (overwriting of the PFC_ELC_PDBFg register is enabled)
  If another event is input to the input port group, the signal value on the external pin will be transferred to the PFC_ELC_PDBFg register.

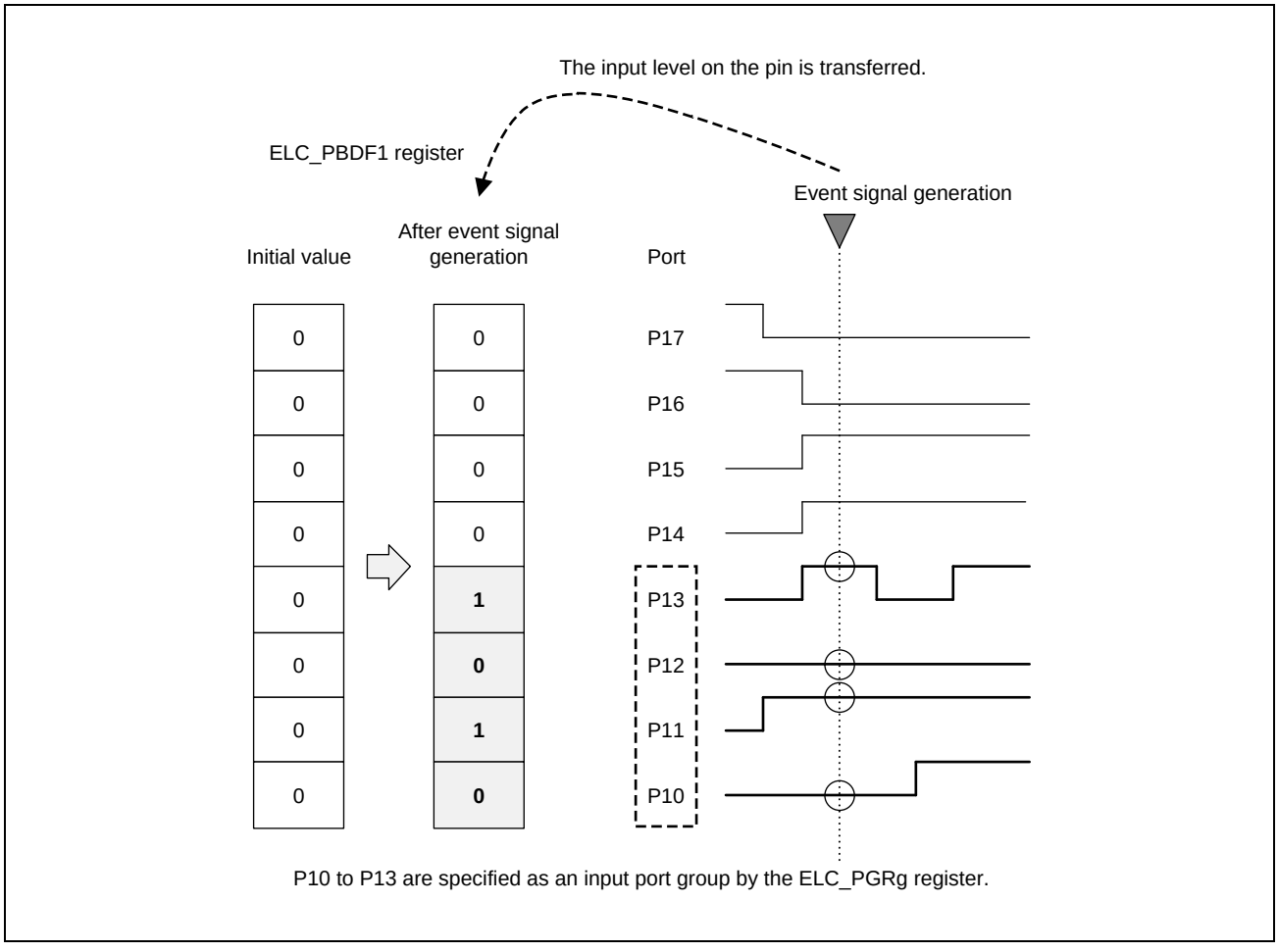


Figure 4.2-9 Input Port Group Operation

(5) Output Port Group Operation upon Event Input

When an event signal (the rising edge condition) is input to an output port group while the setting of the PGCO bits of the PFC_ELC_PGCg register is 000b, 001b, or 010b, the PFC_P_mn register for the bits specified as a group by the ELC_PGRg register is changed to the value set by the PGCO bits.

When an event signal (the rising edge condition) is input to the output port group while the setting of the PGCO bits of the PFC_ELC_PGCg register is 011b, the PFC_ELC_PDBFg register value will be transferred to the PFC_P_mn register for the bits specified as a group by the PFC_ELC_PGRg register. **Figure 4.2-10** shows an example of output port group operation at the time of event input when ELC_PGCg.PGCO = 011b.

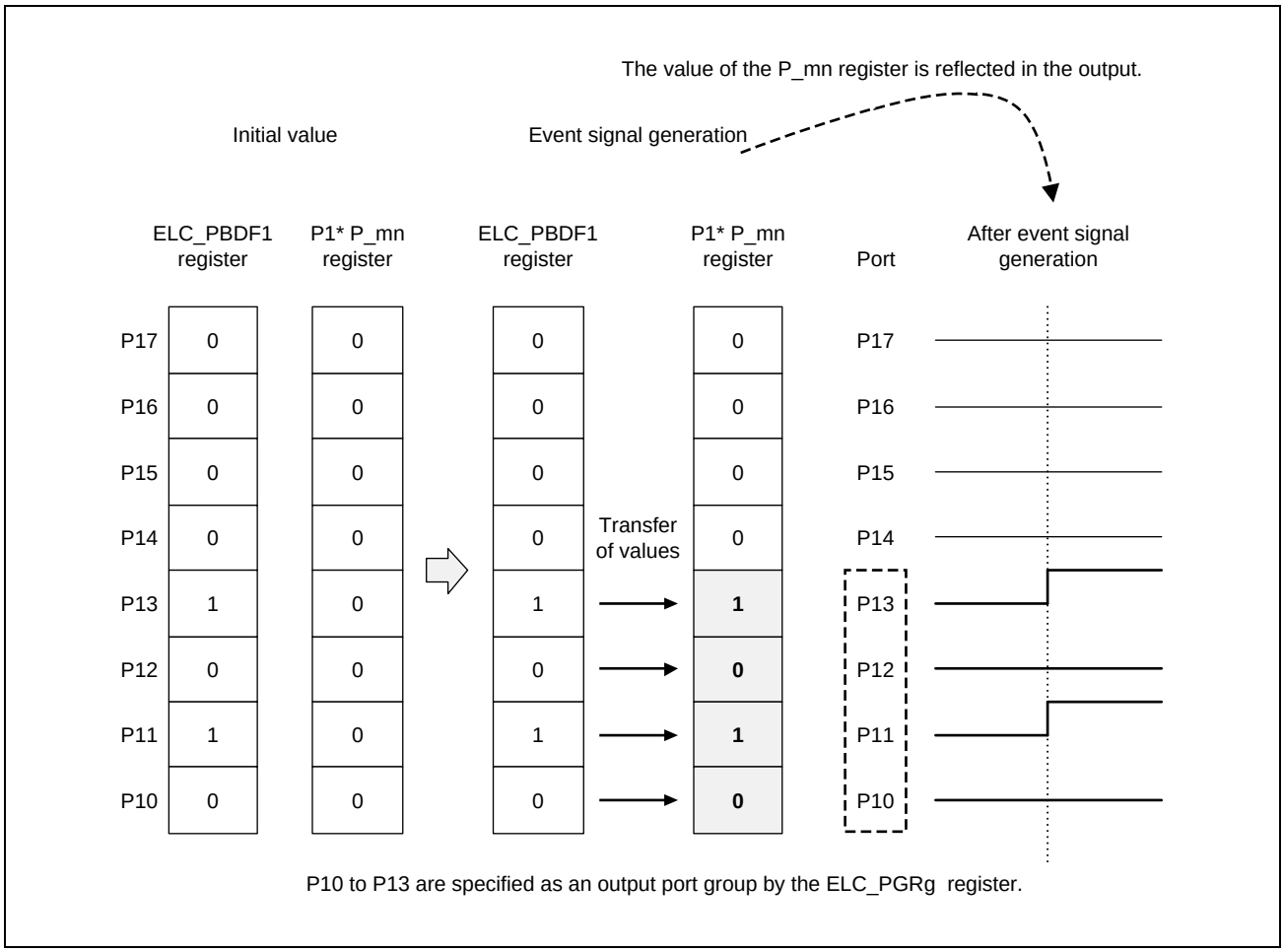


Figure 4.2-10 Output Port Group Operation

(6) Restrictions on Writing to PFC_P_mn and PFC_ELC_PDBFg Registers by a CPU

Event linking is subject to the restrictions on the CPU writing to the PFC_P_mn and PFC_ELC_PDBFg registers.

Remark 1. For the bits of a port specified as members of an output port group, writing to the associated bits of the PFC_P_mn register has no effect.

Remark 2. For the bits of a port specified as members of an input port group, writing to the associated bits of the PFC_ELC_PDBFg register has no effect.

(7) Example of operation

a) Set up operation of the module to which events are linked. To link events, the ELC and relevant modules must be enabled. If the ELC and relevant modules are stopped due, for example, to a low-power mode setting, the module cannot operate.

b) If an event is linked to a port, set the registers associated with the port.

- I/O port settings
  - PFC_P_mn register: Sets the initial value of the output port.
  - PFC_PM_mn register: Sets the I/O direction of the port.
- ELC settings
  - PFC_ELC_PGRg register: When ports are to be used as a port group, set the ports to be in the group in bit units.
  - PFC_ELC_PGCg register: Sets up operation of the port group.
  - PFC_ELC_PELs register: When ports are to be used as single ports, set the bit numbers of the ports, port operation at the time of event input, and the condition for event generation.
  - PFC_ELC_DPTC register: Sets input edge detection for single ports.
  - PFC_ELC_ELSR2 register: Sets the functions of the port event controller to which events are linked.

*** Setting the PFC_ELC_PDBFg register**

- Output port group
  Before setting the PFC_ELC_PDBFg register, set the PFC_ELC_PGCg register. The PFC_ELC_PGCg register must be changed before changing the PFC_P_mn register.
- Input port group
  Setting the PFC_ELC_PDBFg register is not required. However, its value after a reset is 0b, so if you want to check a change from high to low level in response to event input, set the PFC_ELC_PDBFg register for the target port to 1b.

4.2.4.8 Switching Power in the OTHERS Area Power Domain from ON to OFF

Switching power from ON to OFF

Due to this operation, the output levels to external devices are retained even after the OTHERS area is turned off.

1. Access the registers to make preparations for setting port mode (GPIO).
   - When the direction of the I/O buffer is to be “input”:
     Set the PFC_PM_mn register so that the GPIO setting is for input mode.
   - When the direction of the I/O buffer is to be “output”:
     Set the PFC_PM_mn register so that the GPIO setting is for output mode.
     Set the bits of the PFC_P_mn register to the values you want to have output.

2. Set the PFC_PMC_mn register to 0 to set port mode (GPIO).

Remark Changing the setting of the PFC_PMC_mn register requires manipulation of the PWPR.REGWE_A bit.
- Before changing the setting of the PFC_PMC_mn register: Set the PWPR.REGWE_A bit to 1b.
- After changing the setting of the PFC_PMC_mn register: Set the PWPR.REGWE_A bit to 0b.

Switching power from OFF to ON

Same as the normal power-on sequence.

SECTION 4 SYSTEM

4.3 System Controller (SYS)

This section describes the functions of the system controller (SYS) of this LSI.

4.3.1 Functional Overview

4.3.1.1 Functions

- This module provides product information on the LSI chip, descriptions of the states of the external pins, and the states of essential informaation at the time of booting up from OTP memory.
- TrustZone control
  - Control over privileged or non-privileged (normal) access by all bus masters
  - Control over secure/non-secure management by all bus masters
  - Setting the security levels for each of the slaves (access control)
  - Control over security within SYS itself for the setting registers covered by the above rows
  - Control by registers within SYS itself over each of the slaves and associated access control
  - Control by registers within SYS itself of access to units that do not currently have bus access
- Extended access by specific masters to addresses beyond the 4-GB space
  - Units to access DDR:
  - Address[31:30] (up to 4 GB) → Address[34:30] (up to 32 GB)/in 1-GB units
    - VCD
    - MIPI DSI
    - LCDC
    - VSP
    - SD0/SD1/SD2
    - USB20/USB21
    - USB30
    - GBETH0/GBETH1
    - NPU (Ethos-U55)
  - Units to access DDR and PCIE:
  - Address[31:30] (up to 4 GB) → Address[35:30] (up to 64 GB)/in 1-GB units
    - DMAC0/DMAC1/DMAC2/DMAC3/DMAC4/FDP
  - Address[31:28] (up to 4 GB) → Address[35:28] (up to 64 GB)/in 256-MB units
    - CM33

- Control of settings and states for some units
  - Target units
    - SRAM, GBETH, PCIE, CM33, CA55, xSPI, ADC, TSU
- WDT control
  - Control of stopping the WDT from the CoreSight
  - Control of stopping a specified channel of the WDT from the CPU

4.3.2 Connection Schematic Diagram

Figure 4.3-1 is a schematic view of the external connections of SYS with other blocks.

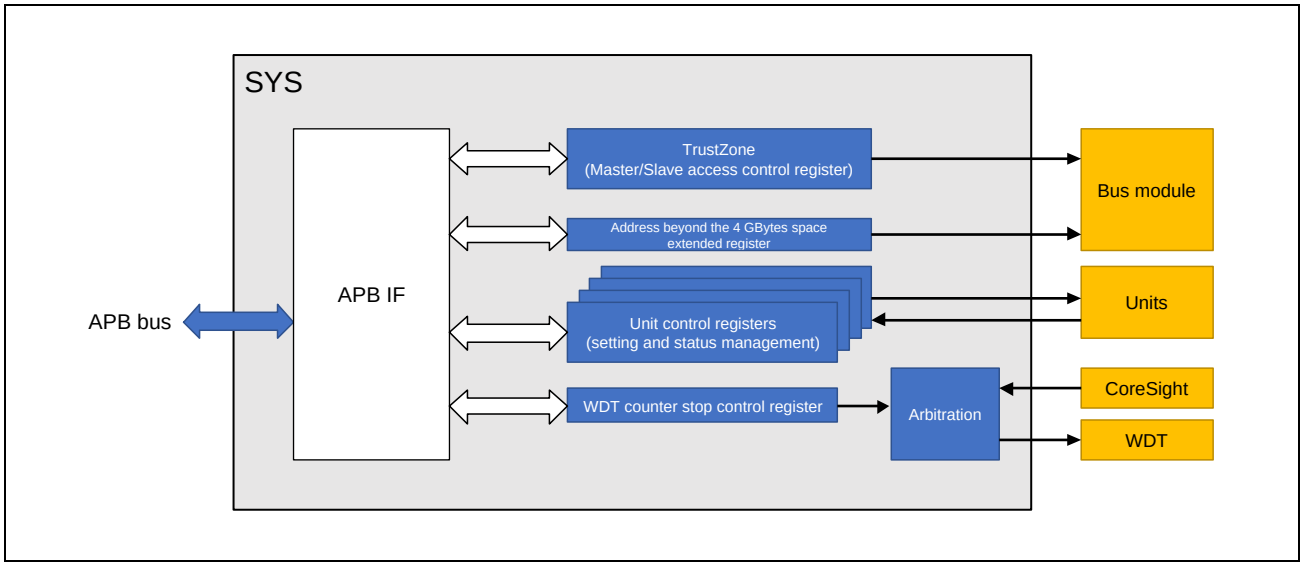


Figure 4.3-1 SYS Connection Schematic Diagram

4.3.3 Registers

The base address of the SYS register is shown below.

Table 4.3-1 Register Base Addresses

Base Address Name	Base Address
<SYS_base>	0_1043_0000h (5043_0000h*1, 4043_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

4.3.3.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Master Access Control Register 0	SYS_MSTACCCTL0	0000_0000h	0000h	32
Master Access Control Register 1	SYS_MSTACCCTL1	0000_0000h	0004h	32
Reserve	-	-	0008h to 000Bh	-
Master Access Control Register 3	SYS_MSTACCCTL3	00AA_AAAAh	000Ch	32
Master Access Control Register 4	SYS_MSTACCCTL4	0000_00AAh	0010h	32
Master Access Control Register 5	SYS_MSTACCCTL5	00AA_AAAAh	0014h	32
Master Access Control Register 6	SYS_MSTACCCTL6	0000_AAAAh	0018h	32
Master Access Control Register 7	SYS_MSTACCCTL7	0000_00AAh	001Ch	32
Master Access Control Register 8	SYS_MSTACCCTL8	0000_AAAAh	0020h	32
Reserve	-	-	0024h to 002Fh	-
Master Access Control Register 12	SYS_MSTACCCTL12	0000_00AAh	0030h	32
Master Access Control Register 13	SYS_MSTACCCTL13	0000_AAAAh	0034h	32
Master Access Control Register 14	SYS_MSTACCCTL14	0000_00AAh	0038h	32
Master Access Control Register 15	SYS_MSTACCCTL15	0000_00AAh	003Ch	32
Reserve	-	-	0040h to 004Bh	-
Master Access Control Register 19	SYS_MSTACCCTL19	0000_00AAh	004Ch	32
Reserve	-	-	0050h to 0053h	-
Master Access Control Register 21	SYS_MSTACCCTL21	0000_00AAh	0054h	32
Reserve	-	-	0058h to 00FFh	-
Slave Access Control Register 0	SYS_SLVACCCTL0	0000_0000h	0100h	32
Slave Access Control Register 1	SYS_SLVACCCTL1	0000_0003h	0104h	32
Slave Access Control Register 2	SYS_SLVACCCTL2	0000_0000h	0108h	32
Slave Access Control Register 3	SYS_SLVACCCTL3	0000_0000h	010Ch	32
Slave Access Control Register 4	SYS_SLVACCCTL4	0000_0000h	0110h	32
Slave Access Control Register 5	SYS_SLVACCCTL5	0000_0000h	0114h	32
Slave Access Control Register 6	SYS_SLVACCCTL6	0000_0000h	0118h	32
Slave Access Control Register 7	SYS_SLVACCCTL7	0000_0000h	011Ch	32
Slave Access Control Register 8	SYS_SLVACCCTL8	0000_0000h	0120h	32
Slave Access Control Register 9	SYS_SLVACCCTL9	0000_0000h	0124h	32
Slave Access Control Register 10	SYS_SLVACCCTL10	0000_0000h	0128h	32
Slave Access Control Register 11	SYS_SLVACCCTL11	0000_0000h	012Ch	32
Slave Access Control Register 12	SYS_SLVACCCTL12	0000_0000h	0130h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Slave Access Control Register 13	SYS_SLVACCCTL13	0000_0000h	0134h	32
Slave Access Control Register 14	SYS_SLVACCCTL14	0000_0000h	0138h	32
Reserve	-	-	013Ch to 013Fh	-
Slave Access Control Register 16	SYS_SLVACCCTL16	0000_0000h	0140h	32
Reserve	-	-	0144h to 014Bh	-
Slave Access Control Register 19	SYS_SLVACCCTL19	0000_0000h	014Ch	32
Slave Access Control Register 20	SYS_SLVACCCTL20	0000_0000h	0150h	32
Slave Access Control Register 21	SYS_SLVACCCTL21	0000_000Ah	0154h	32
Reserve	-	-	0158h to 017Fh	-
Slave Access Control Register 32	SYS_SLVACCCTL32	0000_0000h	0180h	32
Reserve	-	-	0184h to 018Bh	-
Slave Access Control Register 35	SYS_SLVACCCTL35	0000_0000h	018Ch	32
Slave Access Control Register 36	SYS_SLVACCCTL36	0000_0000h	0190h	32
Slave Access Control Register 37	SYS_SLVACCCTL37	0000_0000h	0194h	32
Slave Access Control Register 38	SYS_SLVACCCTL38	0000_0000h	0198h	32
Slave Access Control Register 39	SYS_SLVACCCTL39	0000_0000h	019Ch	32
Slave Access Control Register 40	SYS_SLVACCCTL40	0000_0000h	01A0h	32
Slave Access Control Register 41	SYS_SLVACCCTL41	0000_0000h	01A4h	32
Reserve	-	-	01A8h to 01ABh	-
Slave Access Control Register 43	SYS_SLVACCCTL43	0000_0000h	01ACh	32
Slave Access Control Register 44	SYS_SLVACCCTL44	0000_0000h	01B0h	32
Reserve	-	-	01B4h to 01BBh	-
Slave Access Control Register 47	SYS_SLVACCCTL47	0000_0000h	01BCh	32
Slave Access Control Register 48	SYS_SLVACCCTL48	0000_0000h	01C0h	32
Slave Access Control Register 49	SYS_SLVACCCTL49	0000_0000h	01C4h	32
Slave Access Control Register 50	SYS_SLVACCCTL50	0000_0000h	01C8h	32
Slave Access Control Register 51	SYS_SLVACCCTL51	0000_0000h	01CCh	32
Slave Access Control Register 52	SYS_SLVACCCTL52	0000_0000h	01D0h	32
Slave Access Control Register 53	SYS_SLVACCCTL53	0000_0000h	01D4h	32
Reserve	-	-	01D8h to 01DBh	-
Slave Access Control Register 55	SYS_SLVACCCTL55	0000_0000h	01DCh	32
Slave Access Control Register 56	SYS_SLVACCCTL56	0000_0000h	01E0h	32
Slave Access Control Register 57	SYS_SLVACCCTL57	0000_20A8h	01E4h	32
Reserve	-	-	01E8h to 01FFh	-
Slave Access Control Register 64	SYS_SLVACCCTL64	0000_0000h	0200h	32
Slave Access Control Register 65	SYS_SLVACCCTL65	0000_0000h	0204h	32
Slave Access Control Register 66	SYS_SLVACCCTL66	0000_0000h	0208h	32
Slave Access Control Register 67	SYS_SLVACCCTL67	0000_0000h	020Ch	32
Slave Access Control Register 68	SYS_SLVACCCTL68	0000_0000h	0210h	32
Reserve	-	-	0214h to 023Fh	-
Slave Access Control Register 80	SYS_SLVACCCTL80	0000_0000h	0240h	32
Slave Access Control Register 81	SYS_SLVACCCTL81	0000_0000h	0244h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	0248h to 025Bh	-
Slave Access Control Register 87	SYS_SLVACCCTL87	0000_0000h	025Ch	32
Reserve	-	-	0260h to 0263h	-
Slave Access Control Register 89	SYS_SLVACCCTL89	0000_0000h	0264h	32
Reserve	-	-	0268h to 026Bh	-
Slave Access Control Register 91	SYS_SLVACCCTL91	0000_0000h	026Ch	32
Slave Access Control Register 92	SYS_SLVACCCTL92	0000_0000h	0270h	32
Slave Access Control Register 93	SYS_SLVACCCTL93	0000_0000h	0274h	32
Reserve	-	-	0278h to 02FFh	-
LSI Mode Signal Register	SYS_LSI_MODE	000x_xx0xh	0300h	32
LSI Device ID Register	SYS_LSI_DEVID	x867_9447h	0304h	32
LSI Product Register	SYS_LSI_PRR	0000_0x0xh	0308h	32
Reserve	-	-	030Ch to 032Fh	-
TSU1 Trimming Register0	SYS_LSI_OTPTSU1TRMVAL0	xxxx_xxxxh	0330h	32
TSU1 Trimming Register1	SYS_LSI_OTPTSU1TRMVAL1	xxxx_xxxxh	0334h	32
Reserve	-	-	0338h to 04FFh	-
Address Offset Register 0	SYS_AOF0	0302_0100h	0500h	32
Address Offset Register 1	SYS_AOF1	0302_0100h	0504h	32
Address Offset Register 2	SYS_AOF2	0302_0100h	0508h	32
Reserve	-	-	050Ch to 0513h	-
Address Offset Register 5	SYS_AOF5	0302_0100h	0514h	32
Address Offset Register 6	SYS_AOF6	0302_0100h	0518h	32
Reserve	-	-	051Ch to 051Fh	-
Address Offset Register 8	SYS_AOF8	0302_0100h	0520h	32
Address Offset Register 9	SYS_AOF9	0302_0100h	0524h	32
Reserve	-	-	0528h to 053Fh	-
Address Offset Register 16	SYS_AOF16	0302_0100h	0540h	32
Address Offset Register 17	SYS_AOF17	0302_0100h	0544h	32
Address Offset Register 18	SYS_AOF18	0302_0100h	0548h	32
Address Offset Register 19	SYS_AOF19	0302_0100h	054Ch	32
Address Offset Register 20	SYS_AOF20	0302_0100h	0550h	32
Address Offset Register 21	SYS_AOF21	0302_0100h	0554h	32
Address Offset Register 22	SYS_AOF22	0302_0100h	0558h	32
Reserve	-	-	055Ch to 055Fh	-
Address Offset Register 24	SYS_AOF24	0302_0100h	0560h	32
Address Offset Register 25	SYS_AOF25	0302_0100h	0564h	32
Reserve	-	-	0568h to 057Fh	-
Address Offset Register32	SYS_AOF32	0000_0403h	0580h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Address Offset Register 33	SYS_AOF33	0302_0100h	0584h	32
Reserve	-	-	0588h to 05A7h	-
Address Offset Register 42	SYS_AOF42	0302_0100h	05A8h	32
Address Offset Register 43	SYS_AOF43	0302_0100h	05ACh	32
Reserve	-	-	05B0h to 05BFh	-
Address Offset Register 48	SYS_AOF48	0302_0100h	05C0h	32
Address Offset Register 49	SYS_AOF49	0302_0100h	05C4h	32
Address Offset Register 50	SYS_AOF50	0302_0100h	05C8h	32
Reserve	-	-	05CCh to 05FFh	-
CA55 peripheral port region start address Low register	SYS_ACPU_CFG_SMPL	1000_0000h	0600h	32
CA55 peripheral port region start address High register	SYS_ACPU_CFG_SMPH	0000_0000h	0604h	32
CA55 peripheral port region end address Low register	SYS_ACPU_CFG_EMPL	1FF0_0000h	0608h	32
CA55 peripheral port region end address High register	SYS_ACPU_CFG_EMPH	0000_0000h	060Ch	32
Reserve	-	-	0610h to 0623h	-
CA55 Core0 Reset Vector Address Low Configuration Register	SYS_ACPU_CFG_RVAL0	0000_0000h	0624h	32
CA55 Core0 Reset Vector Address High Configuration Register	SYS_ACPU_CFG_RVAH0	0000_0000h	0628h	32
CA55 Core1 Reset Vector Address Low Configuration Register	SYS_ACPU_CFG_RVAL1	0810_0000h	062Ch	32
CA55 Core1 Reset Vector Address High Configuration Register	SYS_ACPU_CFG_RVAH1	0000_0000h	0630h	32
CA55 Core2 Reset Vector Address Low Configuration Register	SYS_ACPU_CFG_RVAL2	0810_0000h	0634h	32
CA55 Core2 Reset Vector Address High Configuration Register	SYS_ACPU_CFG_RVAH2	0000_0000h	0638h	32
CA55 Core3 Reset Vector Address Low Configuration Register	SYS_ACPU_CFG_RVAL3	0810_0000h	063Ch	32
CA55 Core3 Reset Vector Address High Configuration Register	SYS_ACPU_CFG_RVAH3	0000_0000h	0640h	32
Reserve	-	-	0644h to 0807h	-
CM33 Config Register2	SYS_MCPU_CFG2	0000_0000h	080Ch	32
CM33 Config Register3	SYS_MCPU_CFG3	1800_0000h	0810h	32
CM33 Config Register4	SYS_MCPU_CFG4	0000_0000h	0814h	32
CM33 Lock Register	SYS_MCPU_CFG5	0007_0000h	0818h	32
Reserve	-	-	081Ch to 08FFh	-
xSPI CS0 start address	SYS_SPI_STAADDCS0	2000_0000h	0900h	32
xSPI CS0 end address	SYS_SPI_ENDADDCS0	27FF_FFFFh	0904h	32
xSPI CS1 start address	SYS_SPI_STAADDCS1	2800_0000h	0908h	32
xSPI CS1 end address	SYS_SPI_ENDADDCS1	2FFF_FFFFh	090Ch	32
Reserve	-	-	0A04h to 0AFFh	-
SRAM0 ECC Setting Register	SYS_SRAM0_ECC	0000_0000h	0B00h	32
SRAM0 Access Control Register	SYS_SRAM0_EN	0000_0003h	0B04h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	0B08h to 0B4Fh	-
SRAM1 ECC Setting Register	SYS_SRAM1_ECC	0000_0000h	0B50h	32
SRAM1 Access Control Register	SYS_SRAM1_EN	0000_0003h	0B54h	32
Reserve	-	-	0B58h to 0BFFh	-
WDT0 Control Register	SYS_WDT0_CTRL	0000_0000h	0C00h	32
WDT2 Control Register	SYS_WDT2_CTRL	0000_0000h	0C04h	32
WDT3 Control Register	SYS_WDT3_CTRL	0000_0000h	0C08h	32
WDT1 Control Register	SYS_WDT1_CTRL	0000_0000h	0C0Ch	32
Reserve	-	-	0C10h to 0DFFh	-
VSP Control Register	SYS_VSP_CLK	0000_00xxh	0E00h	32
Reserve	-	-	0E04h to 0EFFh	-
GBETH0 Config Register	SYS_GBETH0_CFG	0001_0000h	0F00h	32
GBETH1 Config Register	SYS_GBETH1_CFG	0001_0000h	0F04h	32
Reserve	-	-	0F08h to 0FFFh	-
PCIE INTX register	SYS_PCIE_INTX_CH0	0000_0000h	1000h	32
PCIE MSI register1	SYS_PCIE_MSI1_CH0	0000_0000h	1004h	32
PCIE MSI register2	SYS_PCIE_MSI2_CH0	0000_0000h	1008h	32
PCIE MSI register3	SYS_PCIE_MSI3_CH0	0000_0000h	100Ch	32
PCIE MSI register4	SYS_PCIE_MSI4_CH0	0000_0000h	1010h	32
PCIE MSI register5	SYS_PCIE_MSI5_CH0	0000_0000h	1014h	32
PCIE PME register	SYS_PCIE_PME_CH0	0000_0000h	1018h	32
PCIE ACK register	SYS_PCIE_ACK_CH0	0000_0000h	101Ch	32
PCIE Misc function register	SYS_PCIE_MISC_CH0	000x_0000h	1020h	32
PCIE MODE register	SYS_PCIE_MODE_CH0	0000_0000h	1024h	32
Reserve	-	-	1028h to 15FFh	-
ADC Config Register	SYS_ADC_CFG	0000_0001h	1600h	32
Reserve	-	-	1604h to 16FFh	-
General Register 0	SYS_GPREG_0	0000_0000h	1700h	32
General Register 1	SYS_GPREG_1	0000_0000h	1704h	32
General Register 2	SYS_GPREG_2	0000_0000h	1708h	32
General Register 3	SYS_GPREG_3	0000_0000h	170Ch	32

4.3.3.2 Register Description

4.3.3.2.1 Master Access Control Register 0 (SYS_MSTACCCTL0)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits

Address : <SYS_base> + 0000h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SXRDMC0_ARSEL	-	SXRDMC0_ARNS	SXRDMC0_ARPU	SXRDMC0_AWSEL	-	SXRDMC0_AWNS	SXRDMC0_AWPU	SXADMC1_ARSEL	-	SXADMC1_ARNS	SXADMC1_ARPU	SXADMC1_AWSEL	-	SXADMC1_AWNS	SXADMC1_AWPU
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SXADMC00_ARSEL	-	SXADMC00_ARNS	SXADMC00_ARPU	SXADMC00_AWSEL	-	SXADMC00_AWNS	SXADMC00_AWPU	SXMDMC_ARSEL	-	SXMDMC_ARNS	SXMDMC_ARPU	SXMDMC_AWSEL	-	SXMDMC_AWNS	SXMDMC_AWPU
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	SXRDMC0_ARSEL	0h	RW	DMAC3 read access security attribute source selection 0b: The setting of the master DMAC3 is selected for ARPROT[1:0]. 1b: SXRDMC0_ARPU is selected for ARPROT[0] and SXRDMC0_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
30	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29	SXRDMC0_ARNS	0h	RW	DMAC3 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
28	SXRDMC0_ARPU	0h	RW	DMAC3 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
27	SXRDMC0_AWSEL	0h	RW	DMAC3 write access security attribute source selection 0b: The setting of the master DMAC4 (RCPU) is selected for AWPROT[1:0]. 1b: SXRDMC0_AWPU is selected for AWPROT[0] and SXRDMC0_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
26	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25	SXRDMC0_AWNS	0h	RW	DMAC3 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
24	SXRDMC0_AWPU	0h	RW	DMAC3 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

Bit	Bit Name	Initial Value	R/W	Description
23	SXADMC1_ARSEL	0h	RW	DMAC2 read access security attribute source selection 0b: The setting of the master DMAC2 is selected for ARPROT[1:0]. 1b: SXADMC1_ARPU is selected for ARPROT[0] and SXADMC1_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
22	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21	SXADMC1_ARNS	0h	RW	DMAC2 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
20	SXADMC1_ARPU	0h	RW	DMAC2 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
19	SXADMC1_AWSEL	0h	RW	DMAC2 write access security attribute source selection 0b: The setting of the master DMAC2 is selected for AWPROT[1:0]. 1b: SXADMC1_AWPU is selected for AWPROT[0] and SXADMC1_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
18	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	SXADMC1_AWNS	0h	RW	DMAC2 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
16	SXADMC1_AWPU	0h	RW	DMAC2 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
15	SXADMC00_ARSEL	0h	RW	DMAC1 read access security attribute source selection 0b: The setting of the master DMAC1 is selected for ARPROT[1:0]. 1b: SXADMC00_ARPU is selected for ARPROT[0] and SXADMC00_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
14	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	SXADMC00_ARNS	0h	RW	DMAC1 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
12	SXADMC00_ARPU	0h	RW	DMAC1 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
11	SXADMC00_AWSEL	0h	RW	DMAC1 write access security attribute source selection 0b: The setting of the master DMAC1 is selected for AWPROT[1:0]. 1b: SXADMC00_AWPU is selected for AWPROT[0] and SXADMC00_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
10	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
9	SXADMC00_AWNS	0h	RW	DMAC1 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
8	SXADMC00_AWPU	0h	RW	DMAC1 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
7	SXMDMC_ARSEL	0h	RW	DMAC0 read access security attribute source selection 0b: The setting of the master DMAC0 is selected for ARPROT[1:0]. 1b: SXMDMC_ARPU is selected for ARPROT[0] and SXMDMC_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SXMDMC_ARNS	0h	RW	DMAC0 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SXMDMC_ARPU	0h	RW	DMAC0 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SXMDMC_AWSEL	0h	RW	DMAC0 write access security attribute source selection 0b: The setting of the master DMAC0 is selected for AWPROT[1:0]. 1b: SXMDMC_AWPU is selected for AWPROT[0] and SXMDMC_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXMDMC_AWNS	0h	RW	DMAC0 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXMDMC_AWPU	0h	RW	DMAC0 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.2 Master Access Control Register 1 (SYS_MSTACCCTL1)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits
Address : <SYS_base> + 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SXRDMC1_ARSEL	-	SXRDMC1_ARNS	SXRDMC1_ARPU	SXRDMC1_AWSEL	-	SXRDMC1_AWNS	SXRDMC1_AWPU
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	SXRDMC1_ARSEL	0h	RW	DMAC4 read access security attribute source selection 0b: The setting of the master DMAC4 is selected for ARPROT[1:0]. 1b: SXRDMC1_ARPU is selected for ARPROT[0] and SXRDMC1_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SXRDMC1_ARNS	0h	RW	DMAC4 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SXRDMC1_ARPU	0h	RW	DMAC4 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SXRDMC1_AWSEL	0h	RW	DMAC4 write access security attribute source selection 0b: The setting of the master DMAC4 is selected for AWPROT[1:0]. 1b: SXRDMC1_AWPU is selected for AWPROT[0] and SXRDMC1_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXRDMC1_AWNS	0h	RW	DMAC4 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXRDMC1_AWPU	0h	RW	DMAC4 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.3 Master Access Control Register 3 (SYS_MSTACCCTL3)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits

Address : <SYS_base> + 000Ch

Initial Value : 00AA_AAAAh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	SXSD2_ARSEL	-	SXSD2_ARNS	SXSD2_ARPU	SXSD2_AWSEL	-	SXSD2_AWNS	SXSD2_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SXSD1_ARSEL	-	SXSD1_ARNS	SXSD1_ARPU	SXSD1_AWSEL	-	SXSD1_AWNS	SXSD1_AWPU	SXSD0_ARSEL	-	SXSD0_ARNS	SXSD0_ARPU	SXSD0_AWSEL	-	SXSD0_AWNS	SXSD0_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23	SXSD2_ARSEL	1h	RW	SD2 read access security attribute source selection 0b: ARPROT[1:0] always becomes 10b. 1b: SXSD2_ARPU is selected for ARPROT[0] and SXSD2_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
22	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21	SXSD2_ARNS	1h	RW	SD2 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
20	SXSD2_ARPU	0h	RW	SD2 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
19	SXSD2_AWSEL	1h	RW	SD2 write access security attribute source selection 0b: AWPROT[1:0] always becomes 10b. 1b: SXSD2_AWPU is selected for AWPROT[0] and SXSD2_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
18	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	SXSD2_AWNS	1h	RW	SD2 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
16	SXSD2_AWPU	0h	RW	SD2 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
15	SXSD1_ARSEL	1h	RW	SD1 read access security attribute source selection 0b: ARPROT[1:0] always becomes 10b. 1b: SXSD1_ARPU is selected for ARPROT[0] and SXSD1_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

Bit	Bit Name	Initial Value	R/W	Description
14	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	SXSD1_ARNS	1h	RW	SD1 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
12	SXSD1_ARPU	0h	RW	SD1 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
11	SXSD1_AWSEL	1h	RW	SD1 write access security attribute source selection 0b: AWPROT[1:0] always becomes 10b. 1b: SXSD1_AWPU is selected for AWPROT[0] and SXSD1_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
10	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	SXSD1_AWNS	1h	RW	SD1 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
8	SXSD1_AWPU	0h	RW	SD1 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
7	SXSD0_ARSEL	1h	RW	SD0 read access security attribute source selection 0b: ARPROT[1:0] always becomes 10b. 1b: SXSD0_ARPU is selected for ARPROT[0] and SXSD0_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SXSD0_ARNS	1h	RW	SD0 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SXSD0_ARPU	0h	RW	SD0 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SXSD0_AWSEL	1h	RW	SD0 write access security attribute source selection 0b: AWPROT[1:0] always becomes 10b. 1b: SXSD0_AWPU is selected for AWPROT[0] and SXSD0_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXSD0_AWNS	1h	RW	SD0 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXSD0_AWPU	0h	RW	SD0 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.4 Master Access Control Register 4 (SYS_MSTACCCTL4)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits
Address : <SYS_base> + 0010h
Initial Value : 0000_00AAh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SXU3H0_ARSEL	-	SXU3H0_ARNS	SXU3H0_ARPU	SXU3H0_AWSEL	-	SXU3H0_AWNS	SXU3H0_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	SXU3H0_ARSEL	1h	RW	USB30 host read access security attribute source selection 0b: ARPROT[1:0] always becomes 10b. 1b: SXU3H0_ARPU is selected for ARPROT[0] and SXU3H0_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SXU3H0_ARNS	1h	RW	USB30 host read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SXU3H0_ARPU	0h	RW	USB30 host read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SXU3H0_AWSEL	1h	RW	USB30 host write access security attribute source selection 0b: AWPROT[1:0] always becomes 10b. 1b: SXU3H0_AWPU is selected for AWPROT[0] and SXU3H0_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXU3H0_AWNS	1h	RW	USB30 host write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXU3H0_AWPU	0h	RW	USB30 host write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.5 Master Access Control Register 5 (SYS_MSTACCCTL5)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits

Address : <SYS_base> + 0014h

Initial Value : 00AA_AAAAh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	SHU2F_ARSEL	-	SHU2F_ARNS	SHU2F_ARPU	SHU2F_AWSEL	-	SHU2F_AWNS	SHU2F_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHU2H1_P1_ARSEL	-	SHU2H1_P1_ARNS	SHU2H1_P1_ARPU	SHU2H1_P1_AWSEL	-	SHU2H1_P1_AWNS	SHU2H1_P1_AWPU	SHU2H0_P0_ARSEL	-	SHU2H0_P0_ARNS	SHU2H0_P0_ARPU	SHU2H0_P0_AWSEL	-	SHU2H0_P0_AWNS	SHU2H0_P0_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23	SHU2F_ARSEL	1h	RW	USB20 function read access security attribute source selection 0b: HPROT[1] of the master USB20 function is selected for ARPROT[0], and ARPROT[1] always becomes 1b. 1b: SHU2F_ARPU is selected for ARPROT[0] and SHU2F_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
22	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21	SHU2F_ARNS	1h	RW	USB20 function read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
20	SHU2F_ARPU	0h	RW	USB20 function read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
19	SHU2F_AWSEL	1h	RW	USB20 function write access security attribute source selection 0b: HPROT[1] of the master USB20 function is selected for AWPROT[0], and AWPROT[1] always becomes 1b. 1b: SHU2F_AWPU is selected for AWPROT[0] and SHU2F_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
18	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	SHU2F_AWNS	1h	RW	USB20 function write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
16	SHU2F_AWPU	0h	RW	USB20 function write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

Bit	Bit Name	Initial Value	R/W	Description
15	SHU2H1_P1_ARSEL	1h	RW	USB21 host read access security attribute source selection 0b: HPROT[1] of the master USB21 host is selected for ARPROT[0], and ARPROT[1] always becomes 1b. 1b: SHU2H1_ARPU is selected for ARPROT[0] and SHU2H1_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
14	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	SHU2H1_P1_ARNS	1h	RW	USB21 host read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
12	SHU2H1_P1_ARPU	0h	RW	USB21 host read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
11	SHU2H1_P1_AWSEL	1h	RW	USB21 host write access security attribute source selection 0b: HPROT[1] of the master USB21 host is selected for AWPROT[0], and AWPROT[1] always becomes 1b. 1b: SHU2H1_AWPU is selected for AWPROT[0] and SHU2H1_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
10	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	SHU2H1_P1_AWNS	1h	RW	USB21 host write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
8	SHU2H1_P1_AWPU	0h	RW	USB21 host write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
7	SHU2H0_P0_ARSEL	1h	RW	USB20 host read access security attribute source selection 0b: HPROT[1] of the master USB20 host is selected for ARPROT[0], and ARPROT[1] always becomes 1b. 1b: SHU2H0_ARPU is selected for ARPROT[0] and SHU2H0_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SHU2H0_P0_ARNS	1h	RW	USB20 host read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SHU2H0_P0_ARPU	0h	RW	USB20 host read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SHU2H0_P0_AWSEL	1h	RW	USB20 host write access security attribute source selection 0b: HPROT[1] of the master USB20 host is selected for AWPROT[0], and AWPROT[1] always becomes 1b. 1b: SHU2H0_AWPU is selected for AWPROT[0] and SHU2H0_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
1	SHU2H0_P0_AWNS	1h	RW	USB20 host write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SHU2H0_P0_AWPU	0h	RW	USB20 host write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.6 Master Access Control Register 6 (SYS_MSTACCCTL6)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits
Address : <SYS_base> + 0018h
Initial Value : 0000_AAAAh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SXGBE1_ARSEL	-	SXGBE1_ARNS	SXGBE1_ARPU	SXGBE1_AWSEL	-	SXGBE1_AWNS	SXGBE1_AWPU	SXGBE0_ARSEL	-	SXGBE0_ARNS	SXGBE0_ARPU	SXGBE0_AWSEL	-	SXGBE0_AWNS	SXGBE0_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15	SXGBE1_ARSEL	1h	RW	GBETH1 read access security attribute source selection 0b: ARPROT[1:0] always becomes 10b. 1b: SXGBE1_ARPU is selected for ARPROT[0] and SXGBE1_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
14	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	SXGBE1_ARNS	1h	RW	GBETH1 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
12	SXGBE1_ARPU	0h	RW	GBETH1 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
11	SXGBE1_AWSEL	1h	RW	GBETH1 write access security attribute source selection 0b: AWPROT[1:0] always becomes 10b. 1b: SXGBE1_AWPU is selected for AWPROT[0] and SXGBE1_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
10	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	SXGBE1_AWNS	1h	RW	GBETH1 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
8	SXGBE1_AWPU	0h	RW	GBETH1 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
7	SXGBE0_ARSEL	1h	RW	GBETH0 read access security attribute source selection 0b: ARPROT[1:0] always becomes 10b. 1b: SXGBE0_ARPU is selected for ARPROT[0] and SXGBE0_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

Bit	Bit Name	Initial Value	R/W	Description
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SXGBE0_ARNS	1h	RW	GBETH0 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SXGBE0_ARPU	0h	RW	GBETH0 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SXGBE0_AWSEL	1h	RW	GBETH0 write access security attribute source selection 0b: AWPROT[1:0] always becomes 10b. 1b: SXGBE0_AWPU is selected for AWPROT[0] and SXGBE0_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXGBE0_AWNS	1h	RW	GBETH0 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXGBE0_AWPU	0h	RW	GBETH0 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.7 Master Access Control Register 7 (SYS_MSTACCCTL7)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits
Address : <SYS_base> + 001Ch
Initial Value : 0000_00AAh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SXPCI0_ARSEL	-	SXPCI0_ARNS	SXPCI0_ARPU	SXPCI0_AWSEL	-	SXPCI0_AWNS	SXPCI0_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	SXPCI0_ARSEL	1h	RW	PCIE0 read access security attribute source selection 0b: The setting of the master PCIE0 is selected for ARPROT[1:0]. 1b: SXPCI0_ARPU is selected for ARPROT[0] and SXPCI0_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SXPCI0_ARNS	1h	RW	PCIE0 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SXPCI0_ARPU	0h	RW	PCIE0 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SXPCI0_AWSEL	1h	RW	PCIE0 write access security attribute source selection 0b: The setting of the master PCIE0 is selected for AWPROT[1:0]. 1b: SXPCI0_AWPU is selected for AWPROT[0] and SXPCI0_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXPCI0_AWNS	1h	RW	PCIE0 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXPCI0_AWPU	0h	RW	PCIE0 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.8 Master Access Control Register 8 (SYS_MSTACCCTL8)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits
Address : <SYS_base> + 0020h
Initial Value : 0000_AAAAh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	SXCRS0_AWNS	SXCRS0_AWPU	-	-	-	-	-	-	SXCRV0_AWNS	SXCRV0_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15	-	1h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
14	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	-	1h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
12	-	0h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
11	-	1h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
10	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	SXCRS0_AWNS	1h	RW	CRU0 (Statistics) write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
8	SXCRS0_AWPU	0h	RW	CRU0 (Statistics) write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
7	-	1h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	-	1h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
4	-	0h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

Bit	Bit Name	Initial Value	R/W	Description
3	-	1h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXCRV0_AWNS	1h	RW	CRU0 (Video) write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXCRV0_AWPU	0h	RW	CRU0 (Video) write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.9 Master Access Control Register 12 (SYS_MSTACCCTL12)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits

Address : <SYS_base> + 0030h

Initial Value : 0000_00AAh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SXDSI_ARSEL	-	SXDSI_ARNS	SXDSI_ARPU	SXDSI_AWSEL	-	SXDSI_AWNS	SXDSI_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	SXDSI_ARSEL	1h	RW	DSI read access security attribute source selection 0b: The setting of the master DSI is selected for ARPROT[1:0]. ARPROT[1:0] is fixed to 10b. 1b: SXDSI_ARPU is selected for ARPROT[0] and SXDSI_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SXDSI_ARNS	1h	RW	DSI read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SXDSI_ARPU	0h	RW	DSI read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SXDSI_AWSEL	1h	RW	DSI write access security attribute source selection 0b: The setting of the master DSI is selected for AWPROT[1:0]. AWPROT[1:0] is fixed to 10b. 1b: SXDSI_AWPU is selected for AWPROT[0] and SXDSI_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXDSI_AWNS	1h	RW	DSI write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXDSI_AWPU	0h	RW	DSI write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.10 Master Access Control Register 13 (SYS_MSTACCCTL13)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits

Address : <SYS_base> + 0034h

Initial Value : 0000_AAAAh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SXLCD1_ARSEL	-	SXLCD1_ARNS	SXLCD1_ARPU	SXLCD1_AWSEL	-	SXLCD1_AWNS	SXLCD1_AWPU	SXLCD0_ARSEL	-	SXLCD0_ARNS	SXLCD0_ARPU	SXLCD0_AWSEL	-	SXLCD0_AWNS	SXLCD0_AWPU
Initial Value	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15	SXLCD1_ARSEL	1h	RW	LCDC ch1 read access security attribute source selection 0b: ARPROT[1:0] always becomes 10b. 1b: SXLCD1_ARPU is selected for ARPROT[0] and SXLCD1_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
14	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	SXLCD1_ARNS	1h	RW	LCDC ch1 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
12	SXLCD1_ARPU	0h	RW	LCDC ch1 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
11	SXLCD1_AWSEL	1h	RW	LCDC ch1 write access security attribute source selection 0b: AWPROT[1:0] always becomes 10b. 1b: SXLCD1_AWPU is selected for AWPROT[0] and SXLCD1_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
10	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	SXLCD1_AWNS	1h	RW	LCDC ch1 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
8	SXLCD1_AWPU	0h	RW	LCDC ch1 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
7	SXLCD0_ARSEL	1h	RW	LCDC ch0 read access security attribute source selection 0b: ARPROT[1:0] always becomes 10b. 1b: SXLCD0_ARPU is selected for ARPROT[0] and SXLCD0_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

Bit	Bit Name	Initial Value	R/W	Description
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SXLCD0_ARNS	1h	RW	LCDC ch0 read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SXLCD0_ARPU	0h	RW	LCDC ch0 read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SXLCD0_AWSEL	1h	RW	LCDC ch0 write access security attribute source selection 0b: AWPROT[1:0] always becomes 10b. 1b: SXLCD0_AWPU is selected for AWPROT[0] and SXLCD0_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXLCD0_AWNS	1h	RW	LCDC ch0 write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXLCD0_AWPU	0h	RW	LCDC ch0 write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.11 Master Access Control Register 14 (SYS_MSTACCCTL14)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits
Address : <SYS_base> + 0038h
Initial Value : 0000_00AAh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SXGPU_ARSEL	-	SXGPU_ARNS	SXGPU_ARPU	SXGPU_AWSEL	-	SXGPU_AWNS	SXGPU_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	SXGPU_ARSEL	1h	RW	GE3D read access security attribute source selection 0b: ARPROT[1:0] always becomes 10b. 1b: SXGPU_ARPU is selected for ARPROT[0] and SXGPU_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SXGPU_ARNS	1h	RW	GE3D read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SXGPU_ARPU	0h	RW	GE3D read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SXGPU_AWSEL	1h	RW	GE3D write access security attribute source selection 0b: AWPROT[1:0] always becomes 10b. 1b: SXGPU_AWPU is selected for AWPROT[0] and SXGPU_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXGPU_AWNS	1h	RW	GE3D write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXGPU_AWPU	0h	RW	GE3D write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.12 Master Access Control Register 15 (SYS_MSTACCCTL15)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits
Address : <SYS_base> + 003Ch
Initial Value : 0000_00AAh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SXVCP_ARSEL	-	SXVCP_ARNS	SXVCP_ARPU	SXVCP_AWSEL	-	SXVCP_AWNS	SXVCP_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	SXVCP_ARSEL	1h	RW	VCD read access security attribute source selection 0b: The setting of the master VCD is selected for ARPROT[1:0]. ARPROT[1:0] is fixed to 10b. 1b: SXVCP_ARPU is selected for ARPROT[0] and SXVCP_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SXVCP_ARNS	1h	RW	VCD read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SXVCP_ARPU	0h	RW	VCD read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SXVCP_AWSEL	1h	RW	VCD write access security attribute source selection 0b: The setting of the master VCP is selected for AWPROT[1:0]. AWPROT[1:0] is fixed to 10b. 1b: SXVCP_AWPU is selected for AWPROT[0] and SXVCP_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXVCP_AWNS	1h	RW	VCD write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXVCP_AWPU	0h	RW	VCD write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.13 Master Access Control Register 19 (SYS_MSTACCCTL19)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits
Address : <SYS_base> + 004Ch
Initial Value : 0000_00AAh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SXVSP_ARSEL	-	SXVSP_ARNS	SXVSP_ARPU	SXVSP_AWSEL	-	SXVSP_AWNS	SXVSP_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	SXVSP_ARSEL	1h	RW	VSP read access security attribute source selection 0b: The setting of the master VSP is selected for ARPROT[1:0]. ARPROT[1:0] is fixed to 10b. 1b: SXVSP_ARPU is selected for ARPROT[0] and SXVSP_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SXVSP_ARNS	1h	RW	VSP read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SXVSP_ARPU	0h	RW	VSP read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SXVSP_AWSEL	1h	RW	VSP write access security attribute source selection 0b: The setting of the master VSP is selected for AWPROT[1:0]. AWPROT[1:0] is fixed to 10b. 1b: SXVSP_AWPU is selected for AWPROT[0] and SXVSP_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXVSP_AWNS	1h	RW	VSP write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXVSP_AWPU	0h	RW	VSP write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.14 Master Access Control Register 21 (SYS_MSTACCCTL21)

This register controls the privileged/normal access and secure/non-secure access for each of the bus masters.

Access Size : 32 bits

Address : <SYS_base> + 0054h

Initial Value : 0000_00AAh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SXFDP_ARSEL	-	SXFDP_ARNS	SXFDP_ARPU	SXFDP_AWSEL	-	SXFDP_AWNS	SXFDP_AWPU
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	SXFDP_ARSEL	1h	RW	FDP read access security attribute source selection 0b: The setting of the master FDP is selected for ARPROT[1:0]. 1b: SXFDP_ARPU is selected for ARPROT[0] and SXFDP_ARNS is selected for ARPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	SXFDP_ARNS	1h	RW	FDP read access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
4	SXFDP_ARPU	0h	RW	FDP read access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
3	SXFDP_AWSEL	1h	RW	FDP write access security attribute source selection 0b: The setting of the master FDP is selected for AWPROT[1:0]. 1b: SXFDP_AWPU is selected for AWPROT[0] and SXFDP_AWNS is selected for AWPROT[1]. Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SXFDP_AWNS	1h	RW	FDP write access secure attribute 0b: Secure 1b: Non-secure Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.
0	SXFDP_AWPU	0h	RW	FDP write access privilege attribute 0b: Non-privileged access 1b: Privileged access Changing the setting of this register while the target master is operating is prohibited. Change the register setting before starting access on the master side.

4.3.3.2.15 Slave Access Control Register 0 (SYS_SLVACCCTL0)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0100h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MPCPG_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MPCPG_SL [1:0]	0h	RW	Security control is applied to the slave CPG.

4.3.3.2.16 Slave Access Control Register 1 (SYS_SLVACCCTL1)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0104h
Initial Value : 0000_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MPSYS_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MPSYS_SL [1:0]	3h	R	Security control is applied to the slave SYS.

4.3.3.2.17 Slave Access Control Register 2 (SYS_SLVACCCTL2)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0108h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	MPICU1_SL [1:0]		MPICU0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	MPICU1_SL [1:0]	0h	RW	Security control is applied to the slave ICU.
1, 0	MPICU0_SL [1:0]	0h	RW	Security control is applied to the slave ICU.

4.3.3.2.18 Slave Access Control Register 3 (SYS_SLVACCCTL3)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 010Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MXMDMC_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MXMDMC_SL [1:0]	0h	RW	Security control is applied to the slave DMAC0.

4.3.3.2.19 Slave Access Control Register 4 (SYS_SLVACCCTL4)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0110h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MPMSRM0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MPMSRM0_SL [1:0]	0h	RW	Security control is applied to the slave SRAM0 control register.

4.3.3.2.20 Slave Access Control Register 5 (SYS_SLVACCCTL5)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0114h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MPCST_SL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MPCST_SL[1:0]	0h	RW	Security control is applied to the slave CST.

4.3.3.2.21 Slave Access Control Register 6 (SYS_SLVACCCTL6)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0118h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MPMHU_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MPMHU_SL [1:0]	0h	RW	Security control is applied to the slave MHU.

4.3.3.2.22 Slave Access Control Register 7 (SYS_SLVACCCTL7)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 011Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	MPMOST1_SL [1:0]		MPMOST0_SL [1:0]		MOMCMT3_SL [1:0]		MOMCMT2_SL [1:0]		MOMCMT1_SL [1:0]		MOMCMT0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11, 10	MPMOST1_SL [1:0]	0h	RW	Security control is applied to the slave GTM1.
9, 8	MPMOST0_SL [1:0]	0h	RW	Security control is applied to the slave GTM0.
7, 6	MOMCMT3_SL [1:0]	0h	RW	Security control is applied to the slave CMTW3.
5, 4	MOMCMT2_SL [1:0]	0h	RW	Security control is applied to the slave CMTW2.
3, 2	MOMCMT1_SL [1:0]	0h	RW	Security control is applied to the slave CMTW1.
1, 0	MOMCMT0_SL [1:0]	0h	RW	Security control is applied to the slave CMTW0.

4.3.3.2.23 Slave Access Control Register 8 (SYS_SLVACCCTL8)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0120h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MOMWDT_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MOMWDT_SL [1:0]	0h	RW	Security control is applied to the slave WDT0.

4.3.3.2.24 Slave Access Control Register 9 (SYS_SLVACCCTL9)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0124h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MORTC_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MORTC_SL [1:0]	0h	RW	Security control is applied to the slave RTC.

4.3.3.2.25 Slave Access Control Register 10 (SYS_SLVACCCTL10)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0128h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MORTR_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MORTR_SL [1:0]	0h	RW	Security control (R-only) is applied to the slave RTC.

4.3.3.2.26 Slave Access Control Register 11 (SYS_SLVACCCTL11)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 012Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SPI_REG_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	SPI_REG_SL [1:0]	0h	RW	Security control is applied to the register of the slave xSPI.

4.3.3.2.27 Slave Access Control Register 12 (SYS_SLVACCCTL12)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0130h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MHGPO_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MHGPO_SL [1:0]	0h	RW	Security control is applied to the slave PFC.

4.3.3.2.28 Slave Access Control Register 13 (SYS_SLVACCCTL13)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0134h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	MPPDM1_SL [1:0]		MPPDM0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	MPPDM1_SL [1:0]	0h	RW	Security control is applied to the slave PDM1.
1, 0	MPPDM0_SL [1:0]	0h	RW	Security control is applied to the slave PDM0.

4.3.3.2.29 Slave Access Control Register 14 (SYS_SLVACCCTL14)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0138h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MOADC_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MOADC_SL [1:0]	0h	RW	Security control is applied to the slave ADC.

4.3.3.2.30 Slave Access Control Register 16 (SYS_SLVACCCTL16)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0140h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	MPTSG_SL [1:0]		-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	MPTSG_SL [1:0]	0h	RW	Security control is applied to the slave OTP.
1, 0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

4.3.3.2.31 Slave Access Control Register 19 (SYS_SLVACCCTL19)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 014Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MOSCF_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MOSCF_SL [1:0]	0h	RW	Security control is applied to the slave SCIF.

4.3.3.2.32 Slave Access Control Register 20 (SYS_SLVACCCTL20)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0150h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MOMI2C_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MOMI2C_SL [1:0]	0h	RW	Security control is applied to the slave RIIC8.

4.3.3.2.33 Slave Access Control Register 21 (SYS_SLVACCCTL21)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Only 10b and 11b can be set. Settings of 00b and 01b are prohibited.

Access Size : 32 bits
Address : <SYS_base> + 0154h
Initial Value : 0000_000Ah

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	MPTM33_SL [1:0]		MPTSPI_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	MPTM33_SL [1:0]	2h	RW	Security control is applied to the slave TZC(SRAM0).
1, 0	MPTSPI_SL [1:0]	2h	RW	Security control is applied to the slave TZC(xSPI).

4.3.3.2.34 Slave Access Control Register 32 (SYS_SLVACCCTL32)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0180h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	MXADMC1_SL [1:0]		MXADMC0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	MXADMC1_SL [1:0]	0h	RW	Security control is applied to the slave DMAC2.
1, 0	MXADMC0_SL [1:0]	0h	RW	Security control is applied to the slave DMAC1.

4.3.3.2.35 Slave Access Control Register 35 (SYS_SLVACCCTL35)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 018Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MPSYC_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MPSYC_SL [1:0]	0h	RW	Security control is applied to the slave SYC.

4.3.3.2.36 Slave Access Control Register 36 (SYS_SLVACCCTL36)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0190h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	MPAOST1_SL [1:0]		MPAOST0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	MPAOST1_SL [1:0]	0h	RW	Security control is applied to the slave GTM3.
1, 0	MPAOST0_SL [1:0]	0h	RW	Security control is applied to the slave GTM2.

4.3.3.2.37 Slave Access Control Register 37 (SYS_SLVACCCTL37)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0194h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MOAWDT_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MOAWDT_SL [1:0]	0h	RW	Security control is applied to the slaves WDT1.

4.3.3.2.38 Slave Access Control Register 38 (SYS_SLVACCCTL38)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0198h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MOAI2C7_SL [1:0]		MOAI2C6_SL [1:0]		MOAI2C5_SL [1:0]		MOAI2C4_SL [1:0]		MOAI2C3_SL [1:0]		MOAI2C2_SL [1:0]		MOAI2C1_SL [1:0]		MOAI2C0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15, 14	MOAI2C7_SL [1:0]	0h	RW	Security control is applied to the slave RIIC7.
13, 12	MOAI2C6_SL [1:0]	0h	RW	Security control is applied to the slave RIIC6.
11, 10	MOAI2C5_SL [1:0]	0h	RW	Security control is applied to the slave RIIC5.
9, 8	MOAI2C4_SL [1:0]	0h	RW	Security control is applied to the slave RIIC4.
7, 6	MOAI2C3_SL [1:0]	0h	RW	Security control is applied to the slave RIIC3.
5, 4	MOAI2C2_SL [1:0]	0h	RW	Security control is applied to the slave RIIC2.
3, 2	MOAI2C1_SL [1:0]	0h	RW	Security control is applied to the slave RIIC1.
1, 0	MOAI2C0_SL [1:0]	0h	RW	Security control is applied to the slave RIIC0.

4.3.3.2.39 Slave Access Control Register 39 (SYS_SLVACCCTL39)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 019Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	MXSD2_SL[1:0]		MXSD1_SL[1:0]		MXSD0_SL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5, 4	MXSD2_SL[1:0]	0h	RW	Security control is applied to the slave SD2.
3, 2	MXSD1_SL[1:0]	0h	RW	Security control is applied to the slave SD1.
1, 0	MXSD0_SL[1:0]	0h	RW	Security control is applied to the slave SD0.

4.3.3.2.40 Slave Access Control Register 40 (SYS_SLVACCCTL40)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01A0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MPU2P1_SL [1:0]		MPU2P0_SL [1:0]		MHU2H1_SL [1:0]		MHU2H0_SL [1:0]		-	-	MXU3H0_SL [1:0]		-	-	MPU3P0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15, 14	MPU2P1_SL [1:0]	0h	RW	Security control is applied to the PHY1 slave USB2.0 interface.
13, 12	MPU2P0_SL [1:0]	0h	RW	Security control is applied to the PHY0 slave USB2.0 interface.
11, 10	MHU2H1_SL [1:0]	0h	RW	Security control is applied to the channel 1 slave USB2.0 Host interface.
9, 8	MHU2H0_SL [1:0]	0h	RW	Security control is applied to the channel 0 slave USB2.0 Host/Function interface.
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5, 4	MXU3H0_SL [1:0]	0h	RW	Security control is applied to the channel 0 slave USB 3.2 interface.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MPU3P0_SL [1:0]	0h	RW	Security control is applied to the PHY0 slave USB3.2 interface.

4.3.3.2.41 Slave Access Control Register 41 (SYS_SLVACCCTL41)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01A4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	MXGBE1_SL [1:0]		MXGBE0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	MXGBE1_SL [1:0]	0h	RW	Security control is applied to the channel 1 slave GBETH.
1, 0	MXGBE0_SL [1:0]	0h	RW	Security control is applied to the channel 0 slave GBETH.

4.3.3.2.42 Slave Access Control Register 43 (SYS_SLVACCCTL43)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01ACh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	MPDDP0_SL [1:0]		-	-	MPDDM0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5, 4	MPDDP0_SL [1:0]	0h	RW	Security control is applied to the slave DDR PHY0.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MPDDM0_SL [1:0]	0h	RW	Security control is applied to the slave DDR0.

4.3.3.2.43 Slave Access Control Register 44 (SYS_SLVACCCTL44)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01B0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MPCRU0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MPCRU0_SL [1:0]	0h	RW	Security control is applied to the slave CRU0.

4.3.3.2.44 Slave Access Control Register 47 (SYS_SLVACCCTL47)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01BCh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	MPDSL_SL [1:0]		MPDSD_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	MPDSL_SL [1:0]	0h	RW	Security control is applied to the slave DSI (PHY).
1, 0	MPDSD_SL [1:0]	0h	RW	Security control is applied to the slave DSI (LINK).

4.3.3.2.45 Slave Access Control Register 48 (SYS_SLVACCCTL48)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01C0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MPLCD0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	-	0h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1, 0	MPLCD0_SL [1:0]	0h	RW	Security control is applied to the slave LCDC0

4.3.3.2.46 Slave Access Control Register 49 (SYS_SLVACCCTL49)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01C4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MXGPU_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MXGPU_SL [1:0]	0h	RW	Security control is applied to the slave GE3D

4.3.3.2.47 Slave Access Control Register 50 (SYS_SLVACCCTL50)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01C8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MPVCC_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MPVCC_SL [1:0]	0h	RW	Security control is applied to the slave VCD

4.3.3.2.48 Slave Access Control Register 51 (SYS_SLVACCCTL51)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01CCh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	MPADMC_SL [1:0]		MPSSIUDMC_SL[1:0]		MPSSIU_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5, 4	MPADMC_SL [1:0]	0h	RW	Security control is applied to the slave ADMAC
3, 2	MPSSIUDMC_SL[1:0]	0h	RW	Security control is applied to the slave SSIU (DMAC)
1, 0	MPSSIU_SL [1:0]	0h	RW	Security control is applied to the slave SSIU

4.3.3.2.49 Slave Access Control Register 52 (SYS_SLVACCCTL52)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01D0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	MOSPD2_SL [1:0]		MOSPD1_SL [1:0]		MOSPD0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5, 4	MOSPD2_SL [1:0]	0h	RW	Security control is applied to the slave SPDIF2
3, 2	MOSPD1_SL [1:0]	0h	RW	Security control is applied to the slave SPDIF1
1, 0	MOSPD0_SL [1:0]	0h	RW	Security control is applied to the slave SPDIF0

4.3.3.2.50 Slave Access Control Register 53 (SYS_SLVACCCTL53)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01D4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	MPADG_SL [1:0]		MPSCUDMC_SL[1:0]		MPSCU_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5, 4	MPADG_SL [1:0]	0h	RW	Security control is applied to the slave ADG
3, 2	MPSCUDMC_SL[1:0]	0h	RW	Security control is applied to the slave SCU (DMAC)
1, 0	MPSCU_SL [1:0]	0h	RW	Security control is applied to the slave SCU

4.3.3.2.51 Slave Access Control Register 55 (SYS_SLVACCCTL55)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01DCh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MXGIC_SL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MXGIC_SL[1:0]	0h	RW	Security control is applied to the slave GIC

4.3.3.2.52 Slave Access Control Register 56 (SYS_SLVACCCTL56)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 01E0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MPMTSU1_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MPMTSU1_SL [1:0]	0h	RW	Security control is applied to the slave TSU1

4.3.3.2.53 Slave Access Control Register 57 (SYS_SLVACCCTL57)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Only 10b and 11b can be set. Settings of 00b and 01b are prohibited.

Access Size : 32 bits

Address : <SYS_base> + 01E4h

Initial Value : 0000_20A8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	MPTPCI_SL [1:0]		-	-	-	-	MPTDD00_SL [1:0]		MPTDD01_SL [1:0]		MPTR8_SL [1:0]		-	-
Initial Value	0	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0
R/W	R	R	RW	RW	R	R	R	R	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13, 12	MPTPCI_SL [1:0]	2h	RW	Security control is applied to the slave TZC(PCIe).
11 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7, 6	MPTDD00_SL [1:0]	2h	RW	Security control is applied to the slave TZC(DDR00).
5, 4	MPTDD01_SL [1:0]	2h	RW	Security control is applied to the slave TZC(DDR01).
3, 2	MPTR8_SL [1:0]	2h	RW	Security control is applied to the slave TZC(RCPU Bus).
1, 0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

4.3.3.2.54 Slave Access Control Register 64 (SYS_SLVACCCTL64)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0200h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MOCRC_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MOCRC_SL [1:0]	0h	RW	Security control is applied to the slave CRC control register.

4.3.3.2.55 Slave Access Control Register 65 (SYS_SLVACCCTL65)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0204h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	MOGPT1_SL [1:0]		MOGPT0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	MOGPT1_SL [1:0]	0h	RW	Security control is applied to the slave GPT1 control register.
1, 0	MOGPT0_SL [1:0]	0h	RW	Security control is applied to the slave GPT0 control register.

4.3.3.2.56 Slave Access Control Register 66 (SYS_SLVACCCTL66)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0208h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MOPOE7_SL [1:0]		MOPOE6_SL [1:0]		MOPOE5_SL [1:0]		MOPOE4_SL [1:0]		MOPOE3_SL [1:0]		MOPOE2_SL [1:0]		MOPOE1_SL [1:0]		MOPOE0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15, 14	MOPOE7_SL [1:0]	0h	RW	Security control is applied to the slave POEG1D control register.
13, 12	MOPOE6_SL [1:0]	0h	RW	Security control is applied to the slave POEG1C control register.
11, 10	MOPOE5_SL [1:0]	0h	RW	Security control is applied to the slave POEG1B control register.
9, 8	MOPOE4_SL [1:0]	0h	RW	Security control is applied to the slave POEG1A control register.
7, 6	MOPOE3_SL [1:0]	0h	RW	Security control is applied to the slave POEG0D control register.
5, 4	MOPOE2_SL [1:0]	0h	RW	Security control is applied to the slave POEG0C control register.
3, 2	MOPOE1_SL [1:0]	0h	RW	Security control is applied to the slave POEG0B control register.
1, 0	MOPOE0_SL [1:0]	0h	RW	Security control is applied to the slave POEG0A control register.

4.3.3.2.57 Slave Access Control Register 67 (SYS_SLVACCCTL67)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 020Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	MORCMT3_SL [1:0]		MORCMT2_SL [1:0]		MORCMT1_SL [1:0]		MORCMT0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7, 6	MORCMT3_SL [1:0]	0h	RW	Security control is applied to the slave CMTW7 control register.
5, 4	MORCMT2_SL [1:0]	0h	RW	Security control is applied to the slave CMTW6 control register.
3, 2	MORCMT1_SL [1:0]	0h	RW	Security control is applied to the slave CMTW5 control register.
1, 0	MORCMT0_SL [1:0]	0h	RW	Security control is applied to the slave CMTW4 control register.

4.3.3.2.58 Slave Access Control Register 68 (SYS_SLVACCCTL68)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0210h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	MORWDT1_SL [1:0]		MORWDT0_SL [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	MORWDT1_SL [1:0]	0h	RW	Security control is applied to the control registers of WDT3 within the SYS unit.
1, 0	MORWDT0_SL [1:0]	0h	RW	Security control is applied to the control registers of WDT2 within the SYS unit.

4.3.3.2.59 Slave Access Control Register 80 (SYS_SLVACCCTL80)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0240h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	I3C_SL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	I3C_SL[1:0]	0h	RW	Security control is applied to the control registers of I3C0 within the SYS unit.

4.3.3.2.60 Slave Access Control Register 81 (SYS_SLVACCCTL81)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0244h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CANFD_SL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	CANFD_SL[1:0]	0h	RW	Security control is applied to the control registers of CANFD within the SYS unit.

4.3.3.2.61 Slave Access Control Register 87 (SYS_SLVACCCTL87)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 025Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ACPU_SL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	ACPU_SL[1:0]	0h	RW	Security control is applied to the control registers of CA55 within the SYS unit.

4.3.3.2.62 Slave Access Control Register 89 (SYS_SLVACCCTL89)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0264h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MCPU_SL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MCPU_SL[1:0]	0h	RW	Security control is applied to the control registers of CM33 within the SYS unit.

4.3.3.2.63 Slave Access Control Register 91 (SYS_SLVACCCTL91)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 026Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LSI_SL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	LSI_SL[1:0]	0h	RW	Security control is applied to the control registers of LSI within the SYS unit.

4.3.3.2.64 Slave Access Control Register 92 (SYS_SLVACCCTL92)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0270h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	AOF_SL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	AOF_SL[1:0]	0h	RW	Security control is applied to the control registers of AOF within the SYS unit.

4.3.3.2.65 Slave Access Control Register 93 (SYS_SLVACCCTL93)

This register sets the security level for each of the slaves. For details, see **4.3.4.1 Security Level Setting**.

Access Size : 32 bits
Address : <SYS_base> + 0274h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	GPREG_SL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	GPREG_SL[1:0]	0h	RW	Security control is applied to the control registers of GPREG within the SYS unit.

4.3.3.2.66 LSI Mode Signal Register (SYS_LSI_MODE)

This register indicates the states of external pins of the LSI chip.

Access Size : 32 bits

Address : <SYS_base> + 0300h

Initial Value : 000x_xx0xh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SEC_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	STAT_MD_CLKS	STAT_BOOTPLLCA55[1:0]		STAT_BOOTSELECTER	STAT_DEBUGEN	-	-	-	-	-	-	STAT_MD_BOOT[2:0]		
Initial Value	0	0	x	x	x	x	x	x	0	0	0	0	0	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	SEC_EN	x	R	This register indicates whether this LSI support the security function or not. 0b: Not supported 1b: Available
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	STAT_MD_CLKS	x	R	This register indicates the states of the external MD_CLKS pin. The values are refreshed on the rising edge of PRST_N. However, this register is not updated when a reset has been generated within the LSI chip. Although SYS is temporarily reset, latched data from the states of pins in IOTOP are not updated, with the result that updating does not proceed after release of the reset. This bit controls switching of SSCG on or off for PLLDTY, PLLCA55, PLLVDO, and PLLDRP. 0b: SSCG off 1b: SSCG on
12, 11	STAT_BOOTPLLCA55[1:0]	x	R	This register indicates the states of the external BOOTPLLCA_0 and BOOTPLLCA_1 pins. The values are refreshed on the rising edge of PRST_N. However, this register is not updated when a reset has been generated within the LSI chip. Although SYS is temporarily reset, latched data from the states of pins in IOTOP are not updated, with the result that updating does not proceed after release of the reset. Bit 0: BOOTPLLCA_0 Bit 1: BOOTPLLCA_1
10	STAT_BOOTSELECTER	x	R	This register indicates the states of the external BOOTSELCPU pin. The values are refreshed on the rising edge of PRST_N. However, this register is not updated when a reset has been generated within the LSI chip. Although SYS is temporarily reset, latched data from the states of pins in IOTOP are not updated, with the result that updating does not proceed after release of the reset. 0b: CM33 cold boot 1b: CM55 cold boot
9	STAT_DEBUGEN	x	R	This register indicates the states of the external MD_BOOT3 pin. The values are refreshed on the rising edge of PRST_N. However, this register is not updated when a reset has been generated within the LSI chip. Although SYS is temporarily reset, latched data from the states of pins in IOTOP are not updated, with the result that updating does not proceed after release of the reset. 0b: Normal operation 1b: Debugging is by an ARM debugger connected through the JTAG pins.
8	-	x	R	Reserved Whenever it is read, the undefined value is read.

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	STAT_MD_BOOT[2:0]	x	R	This register indicates the states of the external MD_BOOT2, MD_BOOT1, and MD_BOOT0 pins. The values are refreshed on the rising edge of PRST_N. However, this register is not updated when a reset has been generated within the LSI chip. Although SYS is temporarily reset, latched data from the states of pins in IOTOP are not updated, with the result that updating does not proceed after release of the reset. Bit 0: MD_BOOT0 Bit 1: MD_BOOT1 Bit 2: MD_BOOT2

Note: x: Undefined value

4.3.3.2.67 LSI Device ID Register (SYS_LSI_DEVID)

This register indicates the device ID of the LSI.

Access Size : 32 bits
Address : <SYS_base> + 0304h
Initial Value : x867_9447h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DEV_ID[31:28]				DEV_ID[27:0]											
Initial Value	x	x	x	x	1	0	0	0	0	1	1	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEV_ID[27:0]															
Initial Value	1	0	0	1	0	1	0	0	0	1	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DEV_ID[31:28]	x	R	These bits are for the products individual number (product revision)
27 to 0	DEV_ID[27:0]	867_9447h	R	These bits are for the products individual number (28-bit fixed value) 867_9447h

Note: X: Undefined value

4.3.3.2.68 LSI Product Register (SYS_LSI_PRR)

This register indicates the product information of LSI.

Access Size : 32 bits
Address : <SYS_base> + 0308h
Initial Value : 0000_0x0xh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CA55_DIS	-	-	-	-	-	-	NPU_DIS	-
Initial Value	0	0	0	0	0	0	0	x	0	0	0	0	0	0	x	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	CA55_DIS	x	R	This bit indicates CA55 core number. 1b: CA55 Dual 0b: CA55 Quad
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	-	0h	R	Reserved This bit is read as 0b. The written value will be ignored.
1	NPU_DIS	x	R	This bit indicates whether the NPU exists. 1b: The NPU does not exist. 0b: The NPU exists.
0	-	0h	R	Reserved This bit is read as 0b. The written value will be ignored.

Note: x: Undefined value

4.3.3.2.69 TSU1 Trimming Register 0 (SYS_LSI_OTPTSU1TRMVAL0)

This register indicates the trimming information for TSU1.

Access Size : 32 bits
Address : <SYS_base> + 0330h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSU1_TRMVAL0[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSU1_TRMVAL0[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TSU1_TRMVAL0[31:0]	x	R	TSU1 trimming information 0 monitor register (LT) This register is for reading trimming information for use in measurement at low temperatures by TSU1; the trimming information is stored in OTP memory. The value read is used in a calculation for correction following temperature measurement. Bit 31 to 28: Reserved Bit 27 to 16: Measurement temperature code in measurement at low temperatures Bit 15 to 12: Reserved Bit 11 to 0: TSU1 measurement data in measurement at low temperatures (output code)*1

Note: x: Undefined value

Note 1. These bits indicate a value for 19Bh to E43h.

4.3.3.2.70 TSU1 Trimming Register 1 (SYS_LSI_OTPTSU1TRMVAL1)

This register indicates the trimming information for TSU1.

Access Size : 32 bits
Address : <SYS_base> + 0334h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSU1_TRMVAL1[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSU1_TRMVAL1[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TSU1_TRMVAL1[31:0]	x	R	TSU1 trimming information 1 monitor register (HT) This register is for reading trimming information for use in measurement at high temperatures by TSU1; the trimming information is stored in OTP memory. The value read is used in a calculation for correction following temperature measurement. Bit 31 to 28: Reserved Bit 27 to 16: Measurement temperature code in measurement at high temperatures Bit 15 to 12: Reserved Bit 11 to 0: TSU1 measurement data in measurement at high temperatures (output code)*1

Note: x: Undefined value

Note 1. These bits indicate a value for 19Bh to E43h.

4.3.3.2.71 Address Offset Register n (SYS_AOFn) (n = 0 to 2, 5, 6, 8, 9, 16 to 22, 24, 25)

For details on respective registers, see **Table 4.3-2**.

Access Size : 32 bits

Address : <SYS_base> + 0500h + n * 0004h

Initial Value : 0302_0100h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	OFS11_<unit>[5:0]						-	-	OFS10_<unit>[5:0]					
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	OFS01_<unit>[5:0]						-	-	OFS00_<unit>[5:0]					
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29 to 24	OFS11_<unit>[5:0]	3h	RW	Address value converted to bits [34:30] when the address [31:30] = 11b The master unit is a 32-bit address (4-GB space) master. Corresponding to bits [31:30] of the address on the master side (indicating 1-GB units), access to an address that has been converted to an address space indicated by bits [34:30] is possible. Accordingly, access to an address space beyond 4 GB is possible. Changing the setting of this register while the target master is operating is prohibited. Change the register setting while access on the master side is stopped.
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21 to 16	OFS10_<unit>[5:0]	2h	RW	Address value converted to bits [34:30] when the address [31:30] = 10b The master unit is a 32-bit address (4-GB space) master. Corresponding to bits [31:30] of the address on the master side (indicating 1-GB units), access to an address that has been converted to an address space indicated by bits [34:30] is possible. Accordingly, access to an address space beyond 4 GB is possible. Changing the setting of this register while the target master is operating is prohibited. Change the register setting while access on the master side is stopped.
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 8	OFS01_<unit>[5:0]	1h	RW	Address value converted to bits [34:30] when the address [31:30] = 01b The master unit is a 32-bit address (4-GB space) master. Corresponding to bits [31:30] of the address on the master side (indicating 1-GB units), access to an address that has been converted to an address space indicated by bits [34:30] is possible. Accordingly, access to an address space beyond 4 GB is possible. Changing the setting of this register while the target master is operating is prohibited. Change the register setting while access on the master side is stopped.
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5 to 0	OFS00_<unit>[5:0]	0h	RW	Address value converted to bits [34:30] when the address [31:30] = 00b The master unit is a 32-bit address (4-GB space) master. Corresponding to bits [31:30] of the address on the master side (indicating 1-GB units), access to an address that has been converted to an address space indicated by bits [34:30] is possible. Accordingly, access to an address space beyond 4 GB is possible. Changing the setting of this register while the target master is operating is prohibited. Change the register setting while access on the master side is stopped.

Table 4.3-2 Relationship between SYS_AOFn and Supported Units

Offset Address	SYS_AOFn	Bit Name <unit>	Supported units
0500h	SYS_AOF0	VCD	VCD
0504h	SYS_AOF1	DSI	DSI
0508h	SYS_AOF2	LCDC0	LCDC0
0514h	SYS_AOF5	LCDC1	LCDC1
0518h	SYS_AOF6	VSP	VSP
0520h	SYS_AOF8	NPU	NPU
0524h	SYS_AOF9	NPUR	NPUR
0540h	SYS_AOF16	SDHI_0	SD0
0544h	SYS_AOF17	SDHI_1	SD1
0548h	SYS_AOF18	SDHI_2	SD2
054Ch	SYS_AOF19	USB2H_0	USB20 (Host)
0550h	SYS_AOF20	USB2H_1	USB21 (Host)
0554h	SYS_AOF21	USB2F	USB20 (Func)
0558h	SYS_AOF22	USB3_0	USB30
0560h	SYS_AOF24	GBETH_0	GBETH0
0564h	SYS_AOF25	GBETH_1	GBETH1

4.3.3.2.72 Address Offset Register n (SYS_AOFn) (n = 32)

This is the address-extension register for the CM33 core. It only controls extension in 256-MB units for the CM33 core. For details, refer to **1.7.3.2.2 36-Bit Address Space Access for CM33**.

Access Size : 32 bits
Address : <SYS_base> + 0580h
Initial Value : 0000_0403h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFS0100_MCPU_S[7:0]								OFS0011_MCPU_S[7:0]							
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 8	OFS0100_MCPU_S[7:0]	4h	RW	Address value converted to bits [35:28] when the address [31:28] of CM33 = 0100b The master CM33 is a 32-bit address (4-GB space) master. Corresponding to bits [31:28] of the address on the master side (indicating 256-MB units), access to an address that has been converted to an address space indicated by bits [35:28] is possible. Accordingly, access to an address space beyond 4 GB is possible. Changing the setting of this register while the target master is operating is prohibited. Change the register setting while access on the master side is stopped.
7 to 0	OFS0011_MCPU_S[7:0]	3h	RW	Address value converted to bits [35:28] when the address [31:28] of CM33 = 0011b The master CM33 is a 32-bit address (4-GB space) master. Corresponding to bits [31:28] of the address on the master side (indicating 256-MB units), access to an address that has been converted to an address space indicated by bits [35:28] is possible. Accordingly, access to an address space beyond 4 GB is possible. Changing the setting of this register while the target master is operating is prohibited. Change the register setting while access on the master side is stopped.

4.3.3.2.73 Address Offset Register n (SYS_AOFn) (n = 33, 42, 43, 48 to 50)

For details on respective registers, see **Table 4.3-3**.

Access Size : 32 bits

Address : <SYS_base> + 0584h + (m-33) * 0004h

Initial Value : 0302_0100h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	OFS11_<unit>[5:0]						-	-	OFS10_<unit>[5:0]					
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	OFS01_<unit>[5:0]						-	-	OFS00_<unit>[5:0]					
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29 to 24	OFS11_<unit>[5:0]	3h	RW	Address value converted to bits [35:30] when the address [31:30] = 11b The master unit is a 32-bit address (4-GB space) master. Corresponding to bits [31:30] of the address on the master side (indicating 1-GB units), access to an address that has been converted to an address space indicated by bits [35:30] is possible. Accordingly, access to an address space beyond 4 GB is possible. Changing the setting of this register while the target master is operating is prohibited. Change the register setting while access on the master side is stopped.
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21 to 16	OFS10_<unit>[5:0]	2h	RW	Address value converted to bits [35:30] when the address [31:30] = 10b The master unit is a 32-bit address (4-GB space) master. Corresponding to bits [31:30] of the address on the master side (indicating 1-GB units), access to an address that has been converted to an address space indicated by bits [35:30] is possible. Accordingly, access to an address space beyond 4 GB is possible. Changing the setting of this register while the target master is operating is prohibited. Change the register setting while access on the master side is stopped.
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 8	OFS01_<unit>[5:0]	1h	RW	Address value converted to bits [35:30] when the address [31:30] = 01b The master unit is a 32-bit address (4-GB space) master. Corresponding to bits [31:30] of the address on the master side (indicating 1-GB units), access to an address that has been converted to an address space indicated by bits [35:30] is possible. Accordingly, access to an address space beyond 4 GB is possible. Changing the setting of this register while the target master is operating is prohibited. Change the register setting while access on the master side is stopped.
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5 to 0	OFS00_<unit>[5:0]	0h	RW	Address value converted to bits [35:30] when the address [31:30] = 00b The master unit is a 32-bit address (4-GB space) master. Corresponding to bits [31:30] of the address on the master side (indicating 1-GB units), access to an address that has been converted to an address space indicated by bits [35:30] is possible. Accordingly, access to an address space beyond 4 GB is possible. Changing the setting of this register while the target master is operating is prohibited. Change the register setting while access on the master side is stopped.

Table 4.3-3 Relationship between SYS_AOFm and Supported Units

Offset Address	SYS_AOFm	Bit Name <unit2>	Supported units
0584h	SYS_AOF33	DMAC0	DMAC0
05A8h	SYS_AOF42	DMAC3	DMAC3
05ACh	SYS_AOF43	DMAC4	DMAC4
05C0h	SYS_AOF48	DMAC1	DMAC1
05C4h	SYS_AOF49	DMAC2	DMAC2
05C8h	SYS_AOF50	FDP	FDP

4.3.3.2.74 CA55 Peripheral Port Region Start Address Low Register (SYS_ACPU_CFG_SMPL)

This register is used to identify the start address of the peripheral port area.

Access Size : 32 bits
Address : <SYS_base> + 0600h
Initial Value : 1000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ASTARTMP[11:0]												-	-	-	-
Initial Value	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	ASTARTMP [11:0]	100h	RW	Lower 12 bits of start address of the peripheral port area (ASTARTMP [11:0])
19 to 0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

4.3.3.2.75 CA55 Peripheral Port Region Start Address High Register (SYS_ACPU_CFG_SMPH)

This register is used to identify the start address of the peripheral port area.

Access Size : 32 bits
Address : <SYS_base> + 0604h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ASTARTMP[19:12]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	ASTARTMP [19:12]	0h	RW	Higher 8 bits of start address of the peripheral port area (ASTARTMP [19:12])

4.3.3.2.76 CA55 Peripheral Port Region End Address Low Register (SYS_ACPU_CFG_EMPL)

This register is used to identify the end address of the peripheral port area.

Access Size : 32 bits
Address : <SYS_base> + 0608h
Initial Value : 1FF0_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AENDMP[11:0]												-	-	-	-
Initial Value	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	AENDMP[11:0]	1FFh	RW	Lower 12 bits of end address of the peripheral port area (AENDMP [11:0])
19 to 0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

4.3.3.2.77 CA55 Peripheral Port Region End Address High Register (SYS_ACPU_CFG_EMPH)

This register is used to identify the end address of the peripheral port area.

Access Size : 32 bits
Address : <SYS_base> + 060Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	AENDMP[19:12]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	AENDMP[19:12]	0h	RW	Higher 8 bits of end address of the peripheral port area (AENDMP [19:12])

4.3.3.2.78 CA55 Core 0 Reset Vector Address Low Configuration Register (SYS_ACPU_CFG_RVAL0)

This register sets the reset vector address of CA55 core 0.

Access Size : 32 bits
Address : <SYS_base> + 0624h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RVBARADDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RVBARADDR[15:2]														-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	RVBARADDR [31:2]	0h	RW	This indicates bits 31 to 2 of the lower-order part of the reset vector address for CA55 core 0. Apply a reset (cold, cluster warm, or core warm reset) following every change to the vector address setting. For a sequence of each reset, see **2.2 CPU**.
1, 0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

4.3.3.2.79 CA55 Core 0 Reset Vector Address High Configuration Register (SYS_ACPU_CFG_RVAH0)

This register sets the reset vector address of CA55 core 0.

Access Size : 32 bits
Address : <SYS_base> + 0628h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RVBARADDR[39:32]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	RVBARADDR [39:32]	0h	RW	This indicates bits 39 to 32 of the upper-order part of the reset vector address for CA55 core 0. Apply a reset (cold, cluster warm, or core warm reset) following every change to the vector address setting. For a sequence of each reset, see **2.2 CPU**.

4.3.3.2.80 CA55 Core 1 Reset Vector Address Low Configuration Register (SYS_ACPU_CFG_RVAL1)

This register sets the reset vector address of CA55 core 1.

Access Size : 32 bits
Address : <SYS_base> + 062Ch
Initial Value : 0810_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RVBARADDR[31:16]															
Initial Value	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RVBARADDR[15:2]														-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	RVBARADDR [31:2]	2040000h	RW	This indicates bits 31 to 2 of the lower-order part of the reset vector address for CA55 core 1. Apply a reset (cold, cluster warm, or core warm reset) following every change to the vector address setting. For a sequence of each reset, see **2.2 CPU**.
1, 0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

4.3.3.2.81 CA55 Core 1 Reset Vector Address High Configuration Register (SYS_ACPU_CFG_RVAH1)

This register sets the reset vector address of CA55 core 1.

Access Size : 32 bits
Address : <SYS_base> + 0630h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RVBARADDR[39:32]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	RVBARADDR [39:32]	0h	RW	This indicates bits 39 to 32 of the upper-order part of the reset vector address for CA55 core 1. Apply a reset (cold, cluster warm, or core warm reset) following every change to the vector address setting. For a sequence of each reset, see **2.2 CPU**.

4.3.3.2.82 CA55 Core 2 Reset Vector Address Low Configuration Register (SYS_ACPU_CFG_RVAL2)

This register sets the reset vector address of CA55 core 2.

Access Size : 32 bits
Address : <SYS_base> + 0634h
Initial Value : 0810_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RVBARADDR[31:16]															
Initial Value	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RVBARADDR[15:2]														-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	RVBARADDR [31:2]	2040000h	RW	This indicates bits 31 to 2 of the lower-order part of the reset vector address for CA55 core 2. Apply a reset (cold, cluster warm, or core warm reset) following every change to the vector address setting. For a sequence of each reset, see **2.2 CPU**.
1, 0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

4.3.3.2.83 CA55 Core 2 Reset Vector Address High Configuration Register (SYS_ACPU_CFG_RVAH2)

This register sets the reset vector address of CA55 core 2.

Access Size : 32 bits
Address : <SYS_base> + 0638h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RVBARADDR[39:32]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	RVBARADDR [39:32]	0h	RW	This indicates bits 39 to 32 of the upper-order part of the reset vector address for CA55 core 2. Apply a reset (cold, cluster warm, or core warm reset) following every change to the vector address setting. For a sequence of each reset, see **2.2 CPU**.

4.3.3.2.84 CA55 Core 3 Reset Vector Address Low Configuration Register (SYS_ACPU_CFG_RVAL3)

This register sets the reset vector address of CA55 core 3.

Access Size : 32 bits
Address : <SYS_base> + 063Ch
Initial Value : 0810_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RVBARADDR[31:16]															
Initial Value	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RVBARADDR[15:2]														-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	RVBARADDR [31:2]	2040000h	RW	This indicates bits 31 to 2 of the lower-order part of the reset vector address for CA55 core 3. Apply a reset (cold, cluster warm, or core warm reset) following every change to the vector address setting. For a sequence of each reset, see **2.2 CPU**.
1, 0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

4.3.3.2.85 CA55 Core 3 Reset Vector Address High Configuration Register (SYS_ACPU_CFG_RVAH3)

This register sets the reset vector address of CA55 core 3.

Access Size : 32 bits
Address : <SYS_base> + 0640h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

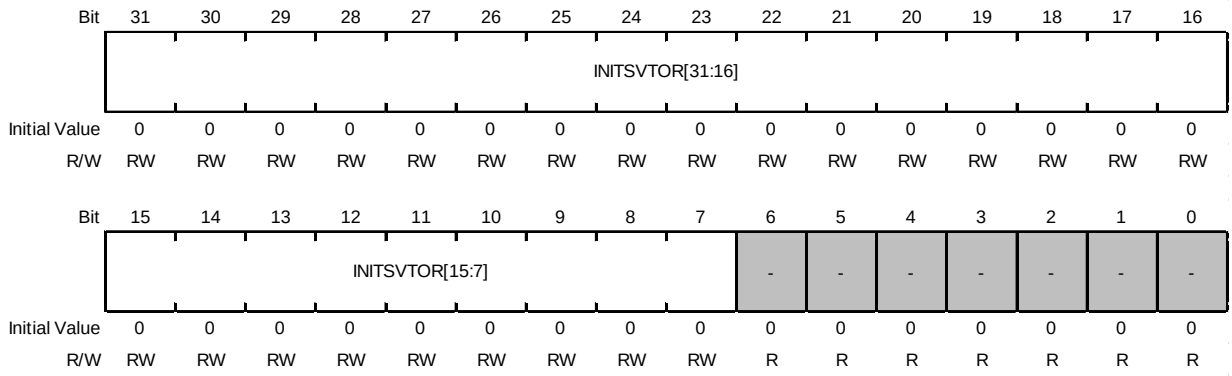
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RVBARADDR[39:32]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	RVBARADDR [39:32]	0h	RW	This indicates bits 39 to 32 of the upper-order part of the reset vector address for CA55 core 3. Apply a reset (cold, cluster warm, or core warm reset) following every change to the vector address setting. For a sequence of each reset, see **2.2 CPU**.

4.3.3.2.86 CM33 Config Register 2 (SYS_MCPU_CFG2)

This register indicates the secure vector address.

Access Size : 32 bits
Address : <SYS_base> + 080Ch
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 7	INITSVTOR [31:7]	0h	RW	This register indicates a secure vector address. The address value before a cold reset of the CM33 must be confirmed.
6 to 0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

4.3.3.2.87 CM33 Config Register 3 (SYS_MCPU_CFG3)

This register indicates the non-secure vector address.

Access Size : 32 bits
Address : <SYS_base> + 0810h
Initial Value : 1800_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INITNSVTOR[31:16]															
Initial Value	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INITNSVTOR[15:7]									-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	INITNSVTOR [31:7]	30_0000h	RW	This register indicates a non-secure vector address. The address value before a cold reset of the CM33 must be confirmed.
6 to 0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

4.3.3.2.88 CM33 Config Register 4 (SYS_MCPU_CFG4)

This register switches the IDAU security map.

Access Size : 32 bits
Address : <SYS_base> + 0814h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	IDAUZERONS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	IDAUZERONS	0h	RW	Security map switching signal for IDAU 0b: Area 0 is secured. 1b: Area 0 is not secured.

4.3.3.2.89 CM33 Lock Register (SYS_MCPU_CFG5)

This register controls authorization to change vector addresses.

Access Size : 32 bits

Address : <SYS_base> + 0818h

Initial Value : 0007_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LOCKNSVTOR	LOCKSVTAIRCR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	-	7h	RW	Reserved The written value should always be 7h.
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	-	0h	RW	Reserved. The written value should always be 0b.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4 to 2	-	0h	RW	Reserved. The written value should always be 0b.
1	LOCKNSVTOR	0h	RW	This bit controls authorization to change non-secure vector addresses. 0b: Change is possible. 1b: Change is not possible.
0	LOCKSVTAIRCR	0h	RW	This bit controls authorization to change secure vector addresses. 0b: Change is possible. 1b: Change is not possible.

4.3.3.2.90 xSPI CS0 Start Address Register (SYS_SPI_STAADDCS0)

This register sets the start address of xSPI CS0.

Access Size : 32 bits
Address : <SYS_base> + 0900h
Initial Value : 2000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	XSPI_STARTADDRCS0[31:16]															
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	XSPI_STARTADDRCS0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	XSPI_STARTADDRCS0[31:0]	2000_0000h	RW	Start address of xSPI CS0. The initial value is for a 128 MB dual device. For a 256 MB single device, the value should be set to 2000_0000h.

4.3.3.2.91 xSPI CS0 End Address Register (SYS_SPI_ENDADDCS0)

This register sets the end address of xSPI CS0.

Access Size : 32 bits
Address : <SYS_base> + 0904h
Initial Value : 27FF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	XSPI_ENDADDRCS0[31:16]															
Initial Value	0	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	XSPI_ENDADDRCS0[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	XSPI_ENDADDRCS0[31:0]	27FF_FFFFh	RW	End address of xSPI CS0. The initial value is for a 128 MB dual device. For a 256 MB single device, the value should be set to 2FFF_FFFFh.

4.3.3.2.92 xSPI CS1 Start Address Register (SYS_SPI_STAADDCS1)

This register sets the start address of xSPI CS1.

Access Size : 32 bits
Address : <SYS_base> + 0908h
Initial Value : 2800_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	XSPI_STARTADDRCS1[31:16]															
Initial Value	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	XSPI_STARTADDRCS1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	XSPI_STARTADDRCS1[31:0]	2800_0000h	RW	Start address of xSPI CS1. The initial value is for a 128 MB dual device. For a 256 MB single device, the value should be set to 0000_0000h (CS1 is not used).

4.3.3.2.93 xSPI CS1 End Address Register (SYS_SPI_ENDADDCS1)

This register sets the end address of xSPI CS1.

Access Size : 32 bits
Address : <SYS_base> + 090Ch
Initial Value : 2FFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	XSPI_ENDADDRCS1[31:16]															
Initial Value	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	XSPI_ENDADDRCS1[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	XSPI_ENDADDRCS1[31:0]	2FFF_FFFFh	RW	End address of xSPI CS1. The initial value is for a 128 MB dual device. For a 256 MB single device, the value should be set to 0000_0000h (CS1 is not used).

4.3.3.2.94 SRAM0 ECC Setting Register (SYS_SRAM0_ECC)

This register selects enabling or disabling correction by ECC of errors in the SRAM0.

Access Size : 32 bits
Address : <SYS_base> + 0B00h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	VECCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	VECCEN	0h	RW	This bit selects enabling or disabling correction by ECC of errors in the SRAM0. 0b: ECC function in the internal RAM is disabled. 1b: ECC function in the internal RAM is enabled.

4.3.3.2.95 SRAM0 Access Control Register (SYS_SRAM0_EN)

This register is used to control access to the SRAM0.

Access Size : 32 bits
Address : <SYS_base> + 0B04h
Initial Value : 0000_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	VLWEN	VCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	VLWEN	1h	RW	This bit selects enabling or disabling writing to the SRAM0. 0b: Writing to the internal RAM is disabled. 1b: Writing to the internal RAM is enabled. The setting at the time of actual usage is 1b.
0	VCEN	1h	RW	This bit selects enabling or disabling access to the SRAM0. 0b: Access to the internal RAM is disabled. 1b: Access to the internal RAM is enabled. The setting at the time of actual usage is 1b.

4.3.3.2.96 SRAM1 ECC Setting Register (SYS_SRAM1_ECC)

This register selects enabling or disabling correction by ECC of errors in the SRAM1.

Access Size : 32 bits
Address : <SYS_base> + 0B50h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	VECCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	VECCEN	0h	RW	This bit selects enabling or disabling correction by ECC of errors in the SRAM1. 0b: ECC function in the internal RAM is disabled. 1b: ECC function in the internal RAM is enabled.

4.3.3.2.97 SRAM1 Access Control Register (SYS_SRAM1_EN)

This register is used to control access to the SRAM1.

Access Size : 32 bits
Address : <SYS_base> + 0B54h
Initial Value : 0000_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	VLWEN	VCEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	VLWEN	1h	RW	This bit selects enabling or disabling writing to the SRAM1. 0b: Writing to the internal RAM is disabled. 1b: Writing to the internal RAM is enabled. The setting at the time of actual usage is 1b.
0	VCEN	1h	RW	This bit selects enabling or disabling access to the SRAM1. 0b: Access to the internal RAM is disabled. 1b: Access to the internal RAM is enabled. The setting at the time of actual usage is 1b.

4.3.3.2.98 WDT0 Control Register (SYS_WDT0_CTRL)

This register controls WDT0.

Access Size : 32 bits

Address : <SYS_base> + 0C00h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WDTSTOPMASK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	bp_halted
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved The written value should always be 0b.
30 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	WDTSTOPMASK	0h	RW	Control over inhibition of stopping WDT0 from the CoreSight module during debugging 0b: Counting for WDT0 is stopped when debugging The CSTDBG signal is transmitted from CoreSight to WDT0. 1b: Counting for WDT0 is not stopped when debugging. The CSTDBG signal is not transmitted from CoreSight to WDT0.
15 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	bp_halted	0h	RW	This bit selects validation or invalidation of forcible stopping of the counter for WDT0. 0b: Counter is operated. 1b: Counter is stopped and its counter value is held.

4.3.3.2.99 WDT1 Control Register (SYS_WDT1_CTRL)

This register controls WDT1.

Access Size : 32 bits
Address : <SYS_base> + 0C0Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WDTSTOPMASK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	bp_halted
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved The written value should always be 0b.
30 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	WDTSTOPMASK	0h	RW	Control over inhibition of stopping WDT1 from the CoreSight module during debugging 0b: Counting for WDT1 is stopped when debugging The CSTDBG signal is transmitted from CoreSight to WDT1. 1b: Counting for WDT1 is not stopped when debugging. The CSTDBG signal is not transmitted from CoreSight to WDT1.
15 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	bp_halted	0h	RW	This bit selects validation or invalidation of forcible stopping of the counter for WDT1. 0b: Counter is operated. 1b: Counter is stopped and its counter value is held.

4.3.3.2.100 WDT2 Control Register (SYS_WDT2_CTRL)

This register controls WDT2.

Access Size : 32 bits
Address : <SYS_base> + 0C04h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WDTSTOPMASK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	bp_halted
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved The written value should always be 0b.
30 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	WDTSTOPMASK	0h	RW	Control over inhibition of stopping WDT2 from the CoreSight module during debugging 0b: Counting for WDT2 is stopped when debugging The CSTDBG signal is transmitted from CoreSight to WDT2. 1b: Counting for WDT2 is not stopped when debugging. The CSTDBG signal is not transmitted from CoreSight to WDT2.
15 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	bp_halted	0h	RW	This bit selects validation or invalidation of forcible stopping of the counter for WDT2. 0b: Counter is operated. 1b: Counter is stopped and its counter value is held.

4.3.3.2.101 WDT3 Control Register (SYS_WDT3_CTRL)

This register controls WDT3.

Access Size : 32 bits
Address : <SYS_base> + 0C08h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WDTSTOPMASK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	bp_halted
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved The written value should always be 0b.
30 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	WDTSTOPMASK	0h	RW	Control over inhibition of stopping WDT3 from the CoreSight module during debugging 0b: Counting for WDT3 is stopped when debugging The CSTDBG signal is transmitted from CoreSight to WDT3. 1b: Counting for WDT3 is not stopped when debugging. The CSTDBG signal is not transmitted from CoreSight to WDT3.
15 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	bp_halted	0h	RW	This bit selects validation or invalidation of forcible stopping of the counter for WDT3. 0b: Counter is operated. 1b: Counter is stopped and its counter value is held.

4.3.3.2.102 VSP Control Register (SYS_VSP_CLK)

This register controls the clock of VSP.

Access Size : 32 bits
Address : <SYS_base> + 0E00h
Initial Value : 0000_00xxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	x_mstp_vsp_ck_n	o_vspi_clk_mon
Initial Value	0	0	0	0	0	0	0	0	0	0	x	0	0	0	1	x
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	-	x	R	Reserved These bits are read as undefined value. The written value will be ignored.
4	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	x_mstp_vsp_ck_n	1h	RW	Module stop signal (for clock divider) 0b: Clock divider is disabled 1b: Clock divider is enabled
0	o_vspi_clk_mon	x	R	VSPI clock monitor 0b: Clock is not supplied 1b: Clock is supplied

Note: x: Undefined value

4.3.3.2.103 GBETH0 Config Register (SYS_GBETH0_CFG)

This register controls the configuration of GBETH0.

Access Size : 32 bits
Address : <SYS_base> + 0F00h
Initial Value : 0001_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	PHY_INTF_TYPE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MAC_SPEED [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	PHY_INTF_TYPE[2:0]	1h	RW	PHY interface mode setting of GBETH0 000b: MII 001b: RGMII Others: Setting is prohibited
15 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MAC_SPEED [1:0]	0h	R	Indeicates the transfer speed of GBETH0 00b: 1 Gbps 01b: Setting is prohibited 10b: 10 Mbps 11b: 100 Mbps

4.3.3.2.104 GBETH1 Config Register (SYS_GBETH1_CFG)

This register controls the configuration of GBETH1.

Access Size : 32 bits
Address : <SYS_base> + 0F04h
Initial Value : 0001_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	PHY_INTF_TYPE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MAC_SPEED [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	PHY_INTF_TYPE[2:0]	1h	RW	PHY interface mode setting of GBETH1 000b: MII 001b: RGMII Others: Setting is prohibited
15 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	MAC_SPEED [1:0]	0h	R	Indeicates the transfer speed of GBETH1 00b: 1 Gbps 01b: Setting is prohibited 10b: 10 Mbps 11b: 100 Mbps

4.3.3.2.105 PCIE INTX Ch 0 Register (SYS_PCIE_INTX_CH0)

This register controls the INTX-transmitting triggers for PCIE channel 0 when it is operating as an endpoint (EP).

Access Size : 32 bits
Address : <SYS_base> + 1000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INTX_EP_F1	INTX_EP_F0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	INTX_EP_F1	0h	RW	INTx transmission In operation as an endpoint (EP), setting the INTX transmission trigger for function 1 to 1 sends asserted INTX and setting it to 0 de-asserts INTX.
0	INTX_EP_F0	0h	RW	INTx transmission In operation as an endpoint (EP), setting the INTX transmission trigger for function 0 to 1 sends asserted INTX and setting it to 0 de-asserts INTX.

4.3.3.2.106 PCIE MSI Ch 0 Register 1 (SYS_PCIE_MSI1_CH0)

This register controls the MSI-issuing triggers for PCIE channel 0 when it is operating as an endpoint (EP).

Access Size : 32 bits
Address : <SYS_base> + 1004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	UI_EXTMSI_VAL4	UI_EXTMSI_VAL3	UI_EXTMSI_VAL2	UI_EXTMSI_VAL1	UI_EXTMSI_VAL0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	UI_EXTMSI_VAL4	0h	RW	MSI-issuing 4 In EP operation, setting the MSI-issuing trigger to 1 leads to MSI transmission.
3	UI_EXTMSI_VAL3	0h	RW	MSI-issuing 3 In EP operation, setting the MSI-issuing trigger to 1 leads to MSI transmission.
2	UI_EXTMSI_VAL2	0h	RW	MSI-issuing 2 In EP operation, setting the MSI-issuing trigger to 1 leads to MSI transmission.
1	UI_EXTMSI_VAL1	0h	RW	MSI-issuing 1 In EP operation, setting the MSI-issuing trigger to 1 leads to MSI transmission.
0	UI_EXTMSI_VAL0	0h	RW	MSI-issuing 0 In EP operation, setting the MSI-issuing trigger to 1 leads to MSI transmission.

4.3.3.2.107 PCIE MSI Ch 0 Register 2 (SYS_PCIE_MSI2_CH0)

This register sets the PCIE channel 0 MSI vector.

Access Size : 32 bits
Address : <SYS_base> + 1008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	UI_EXTMSI_VEC3[4:0]					-	-	-	UI_EXTMSI_VEC2[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	UI_EXTMSI_VEC1[4:0]					-	-	-	UI_EXTMSI_VEC0[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 24	UI_EXTMSI_VEC3[4:0]	0h	RW	MSI vector setting Setting of the vector value for the MSI issued by MSI-issuing 3
23 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20 to 16	UI_EXTMSI_VEC2[4:0]	0h	RW	MSI vector setting Setting of the vector value for the MSI issued by MSI-issuing 2
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 8	UI_EXTMSI_VEC1[4:0]	0h	RW	MSI vector setting Setting of the vector value for the MSI issued by MSI-issuing 1
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4 to 0	UI_EXTMSI_VEC0[4:0]	0h	RW	MSI vector setting Setting of the vector value for the MSI issued by MSI-issuing 0

4.3.3.2.108 PCIE MSI Ch 0 Register 3 (SYS_PCIE_MSI3_CH0)

This register sets the PCIE channel 0 MSI vector.

Access Size : 32 bits
Address : <SYS_base> + 100Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	UI_EXTMSI_VEC4[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4 to 0	UI_EXTMSI_VEC4[4:0]	0h	RW	MSI vector setting Setting of the vector value for the MSI issued by MSI-issuing 4

4.3.3.2.109 PCIE MSI Ch 0 Register 4 (SYS_PCIE_MSI4_CH0)

This register sets the function number of PCIE channel 0 MSI.

Access Size : 32 bits

Address : <SYS_base> + 1010h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	UI_EXTMSI_FUNC3[2:0]			-	-	-	-	-	UI_EXTMSI_FUNC2[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	RW	RW	R	R	R	R	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	UI_EXTMSI_FUNC1[2:0]			-	-	-	-	-	UI_EXTMSI_FUNC0[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	RW	RW	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
26 to 24	UI_EXTMSI_FUNC3[2:0]	0h	RW	MSI function number setting Setting of the function number for MSI issued by MSI-issuing 3
23 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	UI_EXTMSI_FUNC2[2:0]	0h	RW	MSI function number setting Setting of the function number for MSI issued by MSI-issuing 2
15 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10 to 8	UI_EXTMSI_FUNC1[2:0]	0h	RW	MSI function number setting Setting of the function number for MSI issued by MSI-issuing 1
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	UI_EXTMSI_FUNC0[2:0]	0h	RW	MSI function number setting Setting of the function number for MSI issued by MSI-issuing 0

4.3.3.2.110 PCIE MSI Ch 0 Register 5 (SYS_PCIE_MSI5_CH0)

This register sets the function number of PCIE channel 0 MSI.

Access Size : 32 bits
Address : <SYS_base> + 1014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	UI_EXTMSI_FUNC4[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	UI_EXTMSI_FUNC4[2:0]	0h	RW	MSI function number setting Setting of the function number for MSI issued by MSI-issuing 4

4.3.3.2.111 PCIE PME Ch 0 Register (SYS_PCIE_PME_CH0)

This register controls PME messages for PCIE channel 0.

Access Size : 32 bits
Address : <SYS_base> + 1018h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	CFG_PMCSR_PME_STATUS_F1	CFG_PMCSR_PME_STATUS_F0	-	-	-	-	-	-	-	PME_TIM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	CFG_PMCSR_PME_STATUS_F1	0h	RW	PME_STATUS setting Setting the PM_PME message issuing trigger to 1 leads to the message being issued. This is for use with function 1.
8	CFG_PMCSR_PME_STATUS_F0	0h	RW	PME_STATUS setting Setting the PM_PME message issuing trigger to 1 leads to the message being issued. This is for use with function 0.
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	PME_TIM	0h	RW	Clock for a PM_PME messages This is used at the time of sending PM_PME messages. Setting of 0 or 1 is reflected in the clock after an interval of from 100 to 150 ms.

4.3.3.2.112 PCIE ACK Ch 0 Register (SYS_PCIE_ACK_CH0)

This register controls ACK for PCIE channel 0.

Access Size : 32 bits
Address : <SYS_base> + 101Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	D3_EVENT_ACK_F1	D3_EVENT_ACK_F0	-	-	-	-	-	-	-	TURN_OFF_EVENT_ACK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	D3_EVENT_ACK_F1	0h	RW	This handles reply to a request for a transition to a non-D0 state (L1) in response to the ACK for a D3 transition. Setting this bit to 1 sends PM_Request_Ack. This is for use with function 1.
8	D3_EVENT_ACK_F0	0h	RW	This handles reply to a request for a transition to a non-D0 state (L1) in response to the ACK for a D3 transition. Setting this bit to 1 sends PM_Request_Ack. This is for use with function 0.
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	TURN_OFF_EVENT_ACK	0h	RW	TURN_OFF_EVENT message reception ACK When a PME_Turn_off message is received, setting the PME_TO_Ack message transmission trigger to 1 sends the message.

4.3.3.2.113 PCIE Misc Function Ch 0 Register (SYS_PCIE_MISC_CH0)

This register controls ASPM and FLR for PCIE channel 0.

Access Size : 32 bits
Address : <SYS_base> + 1020h
Initial Value : 000x_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	FLR_REQ[1:0]		FLR_RESET[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	x	x	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ALLOW_ENTER_L1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19, 18	FLR_REQ[1:0]	x	R	Function level request This indicates a function level reset request from RC. Bit 0: function 0, bit 1: function 1
17, 16	FLR_RESET[1:0]	0h	RW	Function level reset This resets functions (clears the configuration register). Bit 0: function 0, bit 1: function 1
15 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	ALLOW_ENTER_L1	0h	RW	ASPM L1 transition enable 0b: Transition inhibited 1b: Transition enabled

Note: x: Undefined value

4.3.3.2.114 PCIE MODE Ch 0 Register (SYS_PCIE_MODE_CH0)

This register controls switching of PCIE channel 0 between the root complex (RC) and EP modes.

Access Size : 32 bits
Address : <SYS_base> + 1024h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MODE_PORT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	MODE_PORT	0h	RW	RC/EP switching 0b: EP 1b: RC

4.3.3.2.115 ADC Config Register (SYS_ADC_CFG)

This register controls the power of ADC.

Access Size : 32 bits
Address : <SYS_base> + 1600h
Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	sy_mstp_ada
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	sy_mstp_ada	1h	RW	A/D converter power control 0b: The A/D converter is active. 1b: The A/D converter power is turned off (default).

4.3.3.2.116 General Register 0 (SYS_GPREG_0)

These are general registers 0.

Access Size : 32 bits
Address : <SYS_base> + 1700h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GPREG0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPREG0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GPREG0[31:0]	0h	RW	General-purpose register 0 It is a 32-bit register. It is possible to write any value, read the written value.

4.3.3.2.117 General Register 1 (SYS_GPREG_1)

These are general registers 1.

Access Size : 32 bits
Address : <SYS_base> + 1704h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GPREG1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPREG1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GPREG1[31:0]	0h	RW	General-purpose register 1 It is a 32-bit register. It is possible to write any value, read the written value.

4.3.3.2.118 General Register 2 (SYS_GPREG_2)

These are general registers 2.

Access Size : 32 bits
Address : <SYS_base> + 1708h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GPREG2[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPREG2[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GPREG2[31:0]	0h	RW	General-purpose register 2 It is a 32-bit register. It is possible to write any value, read the written value.

4.3.3.2.119 General Register 3 (SYS_GPREG_3)

These are general registers 3.

Access Size : 32 bits
Address : <SYS_base> + 170Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GPREG3[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GPREG3[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GPREG3[31:0]	0h	RW	General-purpose register 3 It is a 32-bit register. It is possible to write any value, read the written value.

4.3.4 Functional Description

4.3.4.1 Security Level Setting

The SYS_SLVACCCTLx registers handle setting of the level of access security for all slaves.

Four types of access are possible.

(1) Non-secure & normal

(2) Non-secure & privileged

(3) Secure & normal

(4) Secure & privileged

- When the *_SL bits are 00b, all four types of access (1), (2), (3), and (4) are allowed.
- When the *_SL bit is 01b, only (1) is rejected, and (2), (3), and (4) are allowed.
- When the *_SL bit is 10b, (1) and (2) are rejected, and (3) and (4) are allowed.
- When the *_SL bit is 11b, (1), (2), and (3) are rejected, and only (4) is allowed.

Set the privileged/normal access and secure/non-secure access for each of the bus masters according to the security level of each of the slaves. The SYS_MSTACCCTLx register is used to control access to respective masters.

For details, refer to **1.7 Internal Bus**.

Table 4.3-4 shows the relation between the registers and bits for use in re-setting the security attributes and the target masters for control.

Table 4.3-4 List of Register Bits for Use in Re-setting of Security Attributes (1/2)

Target Master for Control*1	Control Register	Write Access Control Bits*2			Read Access Control Bits*3		
		Bit for use in re-setting the security attributes immediately below		Bit for use in selecting the security attribute source	Bit for use in re-setting the security attributes immediately below		Bit for use in selecting the security attribute source
		AWPROT[0]	AWPROT[1]		ARPROT[0]	ARPROT[1]	
DMAC0	SYS_MSTACCCTL0	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
DMAC1	SYS_MSTACCCTL0	Bit 8: AWPU	Bit 9: AWNS	Bit 11: AWSEL	Bit 12: ARPU	Bit 13: ARNS	Bit 15: ARSEL
DMAC2	SYS_MSTACCCTL0	Bit 16: AWPU	Bit 17: AWNS	Bit 19: AWSEL	Bit 20: ARPU	Bit 21: ARNS	Bit 23: ARSEL
DMAC3	SYS_MSTACCCTL0	Bit 24: AWPU	Bit 25: AWNS	Bit 27: AWSEL	Bit 28: ARPU	Bit 29: ARNS	Bit 31: ARSEL
DMAC4	SYS_MSTACCCTL1	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
SD0	SYS_MSTACCCTL3	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
SD1	SYS_MSTACCCTL3	Bit 8: AWPU	Bit 9: AWNS	Bit 11: AWSEL	Bit 12: ARPU	Bit 13: ARNS	Bit 15: ARSEL
SD2	SYS_MSTACCCTL3	Bit 16: AWPU	Bit 17: AWNS	Bit 19: AWSEL	Bit 20: ARPU	Bit 21: ARNS	Bit 23: ARSEL
USB30	SYS_MSTACCCTL4	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
USB20 (Host)*4	SYS_MSTACCCTL5	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
USB21 (Host)*4	SYS_MSTACCCTL5	Bit 8: AWPU	Bit 9: AWNS	Bit 11: AWSEL	Bit 12: ARPU	Bit 13: ARNS	Bit 15: ARSEL
USB20 (Function)*4	SYS_MSTACCCTL5	Bit 16: AWPU	Bit 17: AWNS	Bit 19: AWSEL	Bit 20: ARPU	Bit 21: ARNS	Bit 23: ARSEL
GBETH0	SYS_MSTACCCTL6	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
GBETH1	SYS_MSTACCCTL6	Bit 8: AWPU	Bit 9: AWNS	Bit 11: AWSEL	Bit 12: ARPU	Bit 13: ARNS	Bit 15: ARSEL
PCIE0	SYS_MSTACCCTL7	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
CRU0 (Video)*5	SYS_MSTACCCTL8	Bit 0: AWPU	Bit 1: AWNS	—	—	—	—
CRU0 (Statistics)*5	SYS_MSTACCCTL8	Bit 8: AWPU	Bit 9: AWNS	—	—	—	—
DSI	SYS_MSTACCCTL12	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
LCDC0	SYS_MSTACCCTL13	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
LCDC1	SYS_MSTACCCTL13	Bit 8: AWPU	Bit 9: AWNS	Bit 11: AWSEL	Bit 12: ARPU	Bit 13: ARNS	Bit 15: ARSEL
3DGE	SYS_MSTACCCTL14	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL

Table 4.3-4 List of Register Bits for Use in Re-setting of Security Attributes (2/2)

Target Master for Control*1	Control Register	Write Access Control Bits*2			Read Access Control Bits*3		
		Bit for use in re-setting the security attributes immediately below		Bit for use in selecting the security attribute source	Bit for use in re-setting the security attributes immediately below		Bit for use in selecting the security attribute source
		AWPROT[0]	AWPROT[1]		ARPROT[0]	ARPROT[1]	
VCD	SYS_MSTACCCTL15	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
VSP	SYS_MSTACCCTL19	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL
FDP	SYS_MSTACCCTL21	Bit 0: AWPU	Bit 1: AWNS	Bit 3: AWSEL	Bit 4: ARPU	Bit 5: ARNS	Bit 7: ARSEL

Note 1. Re-setting of security attributes for the Cortex-A55, Cortex-M33, and CoreSight bus masters is not possible.

Note 2. The values set in AWPU and AWNS are effective when the setting of the corresponding AWSEL bit is 1b.

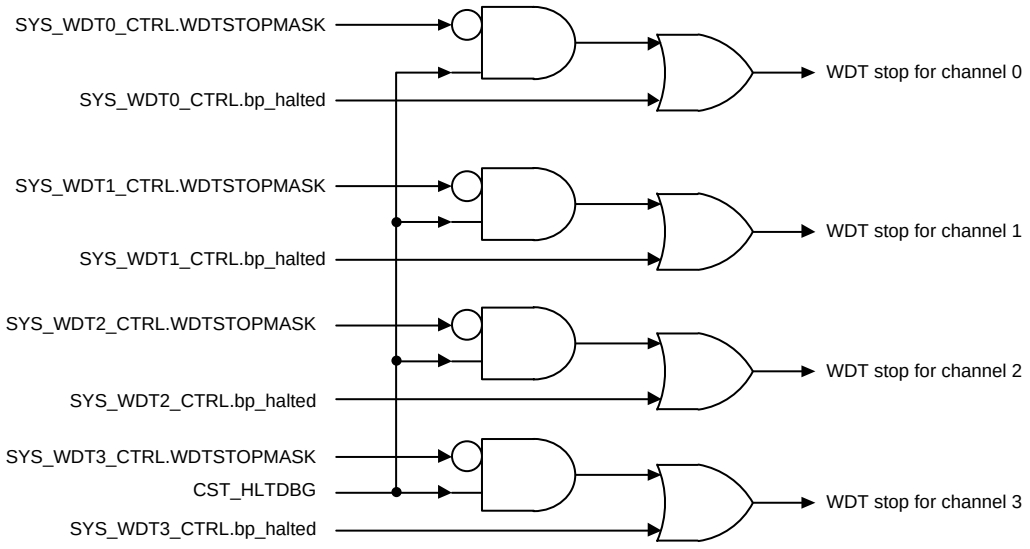
Note 3. The values set in ARPU and ARNS are effective when the setting of the corresponding ARSEL bit is 1b.

Note 4. When the setting of AWSEL for use in control of the USB2.0 module is 0b, the values in AWPROT[0] and ARPROT[0] are the same as the value of HPROT[1] output from the USB2.0 module, and the values of AWPROT[1] and ARPROT[1] are always 1b. For control of HPROT[1] in the USB2.0 module, see **6.5 USB 2.0 Interface**. The HPROT signal is prescribed in the AHB-Lite protocol. For details, see the *AMBA 3 AHB-Lite Protocol Specification* published by Arm Ltd.

Note 5. AWSEL and ARSEL for use in control of the ISU and CRU should be set to 1b. Set AWPU, AWNS, ARPU, and ARNS for the desired security attributes. Operation is not guaranteed if AWSEL and ARSEL are set to 0b.

4.3.4.2 Stopping of Counting by WDTm

Each SYS_WDTm_CTRL register can control the stopping of counting by WDTm (m: 0 to 3). The configuration of the circuit for this function is shown below.



Note: CST_HLTDBG: Debugging enable input from CST

Figure 4.3-2 Circuit Diagram for WDT Logic

When CST_HLTDBG is asserted, the WDTSTOPMASK bits do not stop WDT.

bp_halted bits can stop WDT specified by the register setting regardless of CST_HLTDBG.

SECTION 4 SYSTEM

4.4 Clock Pulse Generator (CPG)

This section describes the clock pulse generator (CPG) of this LSI chip.

This manual is a simplified version. For more information, refer to the User's Manual Additional Document.

4.4.1 Overview

The CPG mainly handles the generation and control of clock signals for the IP modules, the generation and control of resets, and control over booting, low power consumption, and the power supply domains.

Table 4.4-1 Overview

Function	Description
Generation and control of clock signals	Clock control
Generation and control of resets	Reset control
Control over booting	Control of booting
Control over the power supply domains	Control over the sequence of turning power off (Refer to **4.5 Power Management Unit (PMU)** for details.)
Control over low power consumption	Control over handshaking with the CPUs and the stopping of clock signals

4.4.1.1 Functions for the Generation and Control of Clock Signals

- Generation of clock signals to be supplied to the various units from an externally input clock signal or PLL output clock signal
- Frequency selection by register settings
- Selection of clock supply routes by register settings that set up the use of the clock selectors
- Control of turning clock supplies on and off by register settings
- Control of the frequencies of the clock signals to be supplied to the CA55 in boot modes
- Control by register settings of the spread spectrum clock generator (SSCG), the multiplication factor settings for the PLL clock signals, and turning the clock signals on and off

- The CPG has eleven PLLs. Their respective roles are as follows.
  - PLLCM33 (default SSCG setting: fixed to off):
    For the system bus in the PD_AWO domain and the various units in that domain
  - PLLCLN (default SSCG setting: fixed to off):
    For the system bus and various units for which the SSCG function is not available
  - PLLDTY (the level on the external MD_CLKS pin determines the default SSCG setting):
    For the system bus and various units for which the SSCG function is available
  - PLLCA55 (the level on the external MD_CLKS pin determines the default SSCG setting):
    For the CA55
  - PLLVDO (the level on the external MD_CLKS pin determines the default SSCG setting):
    For the CRU, CA55, and GE3D
  - PLLETH (default SSCG setting: fixed to off):
    For the GBETH, NPU, and DSI
  - PLLDSI0 (default SSCG setting: off):
    For the DSI, LCDC0, and LVDS0
  - PLLDDR0 (default SSCG setting: fixed to off):
    For DDR channel 0
  - PLLDSI1 (default SSCG setting: off):
    For the DSI, LCDC1, and LVDS1

4.4.1.2 Function for the Generation and Control of Resets

The CPG generates reset signals in response to each of the reset source factors. The types of reset are described below.

- System reset (external pin)
- Debug reset (software reset by CoreSight)
- Software reset (control over the resetting of individual units)
- Error reset
- CM33 warm reset control
- Control over switching the resetting of individual units off and on

4.4.1.3 Function for the Control of Booting

The CPG controls booting of the system. The following four boot modes are available.

- CM33 boot (normal mode)
- CM33 boot (debug mode)
- CA55 boot (normal mode)
- CA55 boot (debug mode)

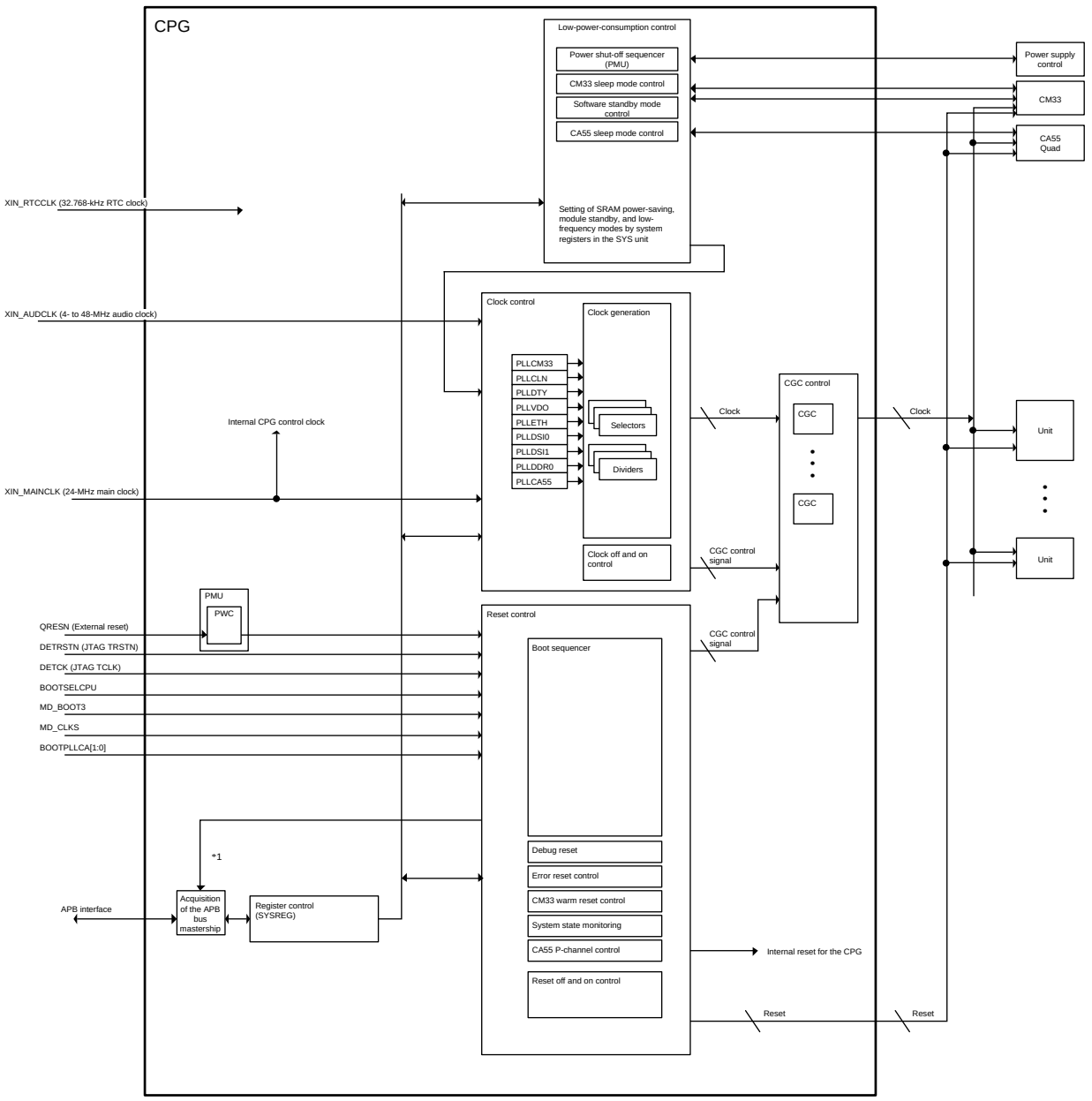
4.4.1.4 Function for the Control over the Power Supply Domains

The CPG uses its power management unit (PMU) to control the sequence of turning power off in the PD_OTHERS domain. Refer to **4.5 Power Management Unit (PMU)** for details.

4.4.1.5 Function for the Control over Low Power Consumption

- The CPG controls low power consumption for the system.
  The CPG controls handshaking with the CPUs in some low-power-consumption modes.
- The types of low-power-consumption mode are described below.
  - Control over power supplies in the PD_OTHERS domain
  - Control over the SRAM power-saving mode
  - Module standby mode
  - Low-frequency mode
  - CA55 sleep mode
  - CM33 sleep mode
  - Software standby mode

4.4.1.6 Functional Block Diagram



Note 1. This is for forcibly modifying the initial setting information such as units to be booted, PLLs to be booted, control over the frequencies for the CA55, settings of the PLLCA55, setting for PLL SSCGs, and control over CM33 waiting.

Figure 4.4-1 CPG Functional Block Diagram

4.4.2 List of Clock Signals

See the separate Excel file for the list of clock signals.

Clock Source	PLLCA55	PLLCA55	PLLCA55	PLLCA55	PLLCA55	PLLDTY	PLLVDO
PLL Oscillation Frequency	1100	1500	1600	1700	1800	1600	1260
Clock gearing division ratio: 1	1100	1500	1600	1700	1800	1600	1260
Clock gearing division ratio: 2	550	750	800	850	900	800	630
Clock gearing division ratio: 4	275	375	400	425	450	400	315
Clock gearing division ratio: 8	138	188	200	213	225	200	158
Clock gearing division ratio: 16	69	94	100	106	112	100	79
Clock gearing division ratio: 32	34	47	50	53	56	50	39
Variable CORECLK	1100, 550, 275, 138	1500, 750, 375, 188	1600, 800, 400, 200	1700, 850, 425, 213	1800, 900, 450, 225	—	—
Variable PERIPHCLK	275, 138, 69, 34	375, 188, 94, 47	400, 200, 100, 50	425, 213, 106, 53	450, 225, 112, 56	—	—
Variable SCLK	1100, 550, 275, 138	750, 375, 188	800, 400, 200	850, 425, 213	900, 450, 225	800, 400, 200	1260, 630, 315, 158

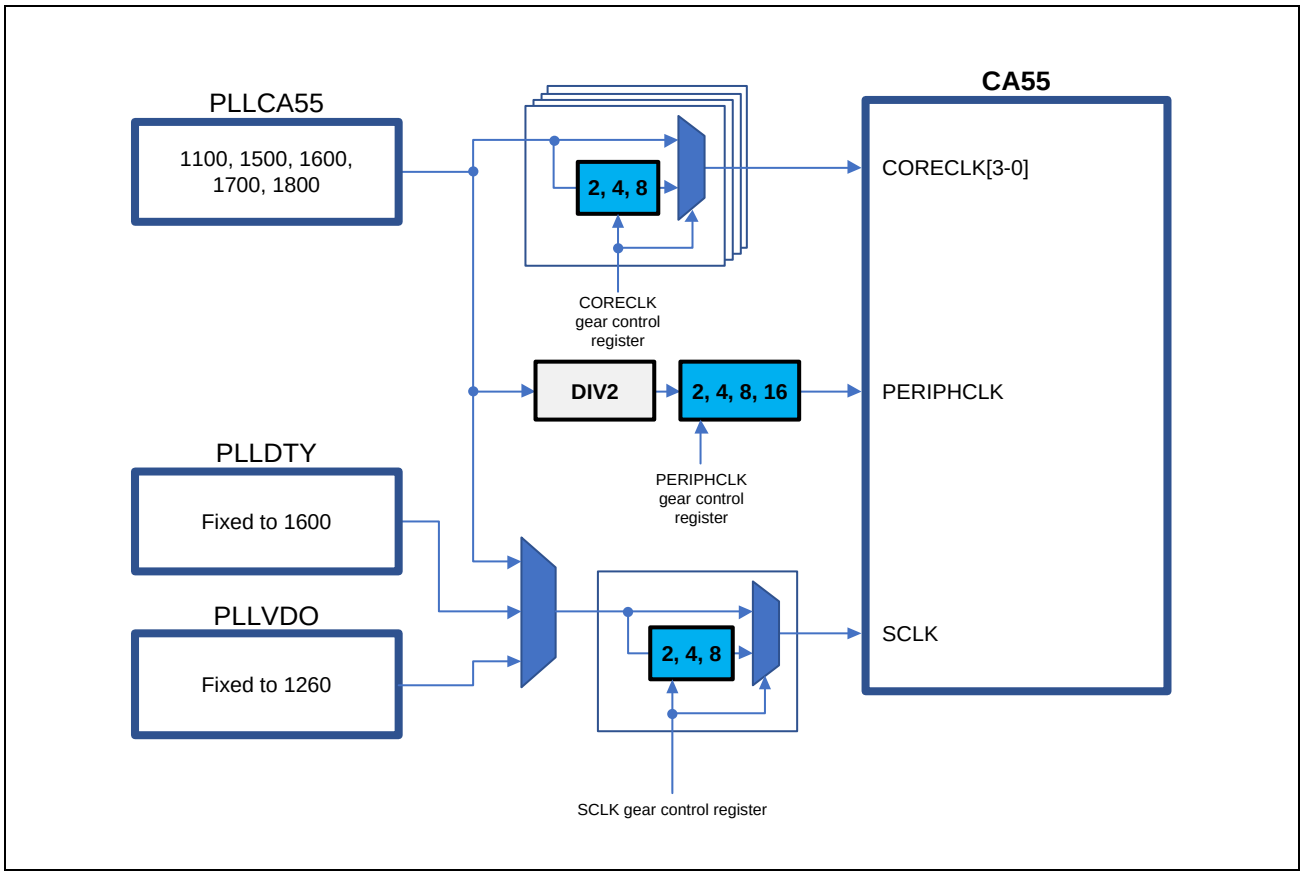


Figure 4.4-2 Variable CA55 Clock Signals

4.4.3 List of Reset Signals

Table 4.4-2 List of Reset Signals (1/6)

Unit	Signal Name	Reset Control Circuit Type*1	Remarks
SYS	SYS_0_PRESETN	Type B	APB reset
PMU	No signal is allocated.	Type B	PMU reset
DMAC0	DMAC_0_ARESETN	Type B	Reset input to the AXI
DMAC1	DMAC_1_ARESETN	Type B	Reset input to the AXI
DMAC2	DMAC_2_ARESETN	Type B	Reset input to the AXI
DMAC3	DMAC_3_ARESETN	Type B	Reset input to the AXI
DMAC4	DMAC_4_ARESETN	Type B	Reset input to the AXI
ICU	ICU_0_PRESETN_I	Type B	APB reset
CRC	CRC_0_RST	Type B	Reset
CA55	CA55_RESET0	CA55 reset	Core cold reset
CA55	CA55_RESET1	CA55 reset	Core cold reset
CA55	CA55_RESET2	CA55 reset	Core cold reset
CA55	CA55_RESET3	CA55 reset	Core cold reset
CA55	CA55_RESET4	CA55 reset	Core warm reset excluding debug registers, RAS registers, and ETM registers
CA55	CA55_RESET5	CA55 reset	Core warm reset excluding debug registers, RAS registers, and ETM registers
CA55	CA55_RESET6	CA55 reset	Core warm reset excluding debug registers, RAS registers, and ETM registers
CA55	CA55_RESET7	CA55 reset	Core warm reset excluding debug registers, RAS registers, and ETM registers
CA55	CA55_RESET8	CA55 reset	Cold reset for all registers in the SCLK domain
CA55	CA55_RESET9	CA55 reset	Warm reset for all registers in the SCLK domain, excluding RAS registers
CA55	CA55_RESET10	CA55 reset	Reset for all registers in the PCLK domain
CA55	CA55_RESET11	CA55 reset	Reset for all registers in the ATCLK domain
CA55	CA55_RESET12	CA55 reset	Reset for all registers in the GICCLK domain
CA55	CA55_RESET13	CA55 reset	Reset for all registers in the PERIPHCLK domain
CA55	CA55_RESET14	CA55 reset	Asynchronous reset for the debug block and time stamp interface
CA55	CA55_RESET15	CA55 reset	Asynchronous reset
CA55	CA55_RESET16	CA55 reset	Asynchronous reset for the generic timer interface
CM33	CM33_RESET0	CM33 reset	Power-up reset. This resets the entire processor.
CM33	CM33_RESET1	CM33 reset	The RESET signal resets the processor.
CM33	CM33_RESET2	CM33 reset	MISC reset
GIC	GIC_0_GICRESET_N	Type B	GIC reset
GIC	GIC_0_DBG_GICRESET_N	Type B	Reset for the PMU and active-low setting in error recording
SRAM0	SRAM_0_ARESETN	Type B	AXI reset
SRAM1	SRAM_1_ARESETN	Type B	AXI reset
ROM	BTROM_0_ARESETN	Type B	AXI reset
CST	CST_0_CS_RESETN	CST reset	Reset signal for the cs_clk components
CST	CST_0_TS_RESETN	CST reset	Reset signal for the ts_clk components
CST	CST_0_APB_SB_RESETN	CST reset	Reset signal for the apb_sb_clk components
CST	CST_0_APB_CA55_RESETN	CST reset	Reset signal for the apb_ca55_clk components
CST	CST_0_APB_CM33_RESETN	CST reset	Reset signal for the apb_cm33_clk components
CST	CST_0_AHB_CM33_RESETN	CST reset	Reset signal for the ahb_cm33_clk components

Table 4.4-2 List of Reset Signals (2/6)

Unit	Signal Name	Reset Control Circuit Type*1	Remarks
CST	CST_0_AHB_ATH_RESETN	CST reset	Reset signal for the ahb_ath_clk components
CST	CST_0_ATB_CA55_RESETN	CST reset	Reset signal for the atb_ca55_clk components
CST	CST_0_ATB_CM33_RESETN	CST reset	Reset signal for the atb_cm33_clk components
CST	CST_0_ATB_CST_RESETN	CST reset	Reset signal for the atb_cst_clk components
CST	CST_0_AXI_SB_RESETN	CST reset	Reset signal for the axi_sb_clk components
CST	CST_0_AXI_ETR_RESETN	CST reset	Reset signal for the axi_etr_clk components
CST	CST_0_NTRST	CST reset	Asynchronous reset for the TAP Optional reset signal for the swclktck components
CST	CST_0_NPOTRST	CST reset	JTAG power-up reset and serial wiring power-up reset Reset signal for the swclktck components
SYC	SYC_0_RESETN	Type B	Global reset
MHU	MHU_0_PRESETN	Type B	APB interface reset
TZC400_DDR00/01	AXI_TZCDDR_0_PRESET0N	Type B	APB interface reset
TZC400_DDR00/01	AXI_TZCDDR_0_PRESET1N	Type B	APB interface reset
TZC400_DDR00/01	AXI_TZCDDR_0_ARESET0N	Type B	ACE-Lite reset for filter 0
TZC400_DDR00/01	AXI_TZCDDR_0_ARESET1N	Type B	ACE-Lite reset for filter 1
TZC400_DDR00/01	AXI_TZCDDR_0_ARESET2N	Type B	ACE-Lite reset for filter 2
TZC400_DDR00/01	AXI_TZCDDR_0_ARESET3N	Type B	ACE-Lite reset for filter 3
TZC400_DDR00/01	AXI_TZCDDR_0_ARESET4N	Type B	ACE-Lite reset for filter 4
ACPU bus	ACPU_AXI_RESETN	Type B	Unit reset
ACPU bus	ACPU_Peri_DDR_AXI_RESETN	Type B	Unit reset
MCPU bus	MCPU_AXI_RESETN	Type B	Unit reset
RCPU bus	RCPU_AXI_RESETN	Type B	Unit reset
Video 0 bus	VIDEO0_AXI_RESETN	Type B	Unit reset
Video 0 bus	ACPU_Peri_VIDEO0_AXI_RESETN	Type B	Unit reset
Video 1 bus	VIDEO1_AXI_RESETN	Type B	Unit reset
Video 1 bus	ACPU_Peri_VIDEO1_AXI_RESETN	Type B	Unit reset
COM bus	COM_AXI_RESETN	Type B	Unit reset
COM bus	ACPU_Peri_COM0_AXI_RESETN	Type B	Unit reset
COM bus	ACPU_Peri_COM1_AXI_RESETN	Type B	Unit reset
GPT0	GPT_0_RST_P_REG	Type B	Unit reset (PCLKA)
GPT0	GPT_0_RST_S_REG	Type B	Unit reset (PCLKD)
GPT1	GPT_1_RST_P_REG	Type B	Unit reset (PCLKA)
GPT1	GPT_1_RST_S_REG	Type B	Unit reset (PCLKD)
POEG0A	POEGA_0_RST	Type B	Unit reset
POEG0B	POEGB_0_RST	Type B	Unit reset
POEG0C	POEGC_0_RST	Type B	Unit reset
POEG0D	POEGD_0_RST	Type B	Unit reset
POEG1A	POEGA_1_RST	Type B	Unit reset
POEG1B	POEGB_1_RST	Type B	Unit reset

Table 4.4-2 List of Reset Signals (3/6)

Unit	Signal Name	Reset Control Circuit Type*1	Remarks
POEG1C	POEGC_1_RST	Type B	Unit reset
POEG1D	POEGD_1_RST	Type B	Unit reset
MCPU_CMTW0	MCPU_CMTW_0_RST_M	Type B	Unit reset
MCPU_CMTW1	MCPU_CMTW_1_RST_M	Type B	Unit reset
MCPU_CMTW2	MCPU_CMTW_2_RST_M	Type B	Unit reset
MCPU_CMTW3	MCPU_CMTW_3_RST_M	Type B	Unit reset
RCPU_CMTW0	RCPU_CMTW_0_RST_M	Type B	Unit reset
RCPU_CMTW1	RCPU_CMTW_1_RST_M	Type B	Unit reset
RCPU_CMTW2	RCPU_CMTW_2_RST_M	Type B	Unit reset
RCPU_CMTW3	RCPU_CMTW_3_RST_M	Type B	Unit reset
MCPU_GTM0	MCPU_GTM_0_PRESETZ	Type B	APB interface reset
MCPU_GTM1	MCPU_GTM_1_PRESETZ	Type B	APB interface reset
ACPU_GTM0	ACPU_GTM_0_PRESETZ	Type B	APB interface reset
ACPU_GTM1	ACPU_GTM_1_PRESETZ	Type B	APB interface reset
RCPU_GTM0	RCPU_GTM_0_PRESETZ	Type B	APB interface reset
RCPU_GTM1	RCPU_GTM_1_PRESETZ	Type B	APB interface reset
RCPU_GTM2	RCPU_GTM_2_PRESETZ	Type B	APB interface reset
RCPU_GTM3	RCPU_GTM_3_PRESETZ	Type B	APB interface reset
WDT0	WDT_0_RESET	Type B	Unit reset
WDT1	WDT_1_RESET	Type B	Unit reset
WDT2	WDT_2_RESET	Type B	Unit reset
WDT3	WDT_3_RESET	Type B	Unit reset
RTC	RTC_0_RST_RTC	Type B	Reset
RTC	RTC_0_RST_RTC_V	Type B	Reset for the RTEST registers
RSPI0	RSPI_0_PRESETN	Type B	APB interface reset control
RSPI0	RSPI_0_TRESETN	Type B	Communications reset control
RSPI1	RSPI_1_PRESETN	Type B	APB interface reset control
RSPI1	RSPI_1_TRESETN	Type B	Communications reset control
RSPI2	RSPI_2_PRESETN	Type B	APB interface reset control
RSPI2	RSPI_2_TRESETN	Type B	Communications reset control
RSCI0	RSCI_0_PRESETN	Type B	Input for resetting the APB clock
RSCI0	RSCI_0_TRESETN	Type B	Input for resetting TCLK
RSCI1	RSCI_1_PRESETN	Type B	Input for resetting the APB clock
RSCI1	RSCI_1_TRESETN	Type B	Input for resetting TCLK
RSCI2	RSCI_2_PRESETN	Type B	Input for resetting the APB clock
RSCI2	RSCI_2_TRESETN	Type B	Input for resetting TCLK
RSCI3	RSCI_3_PRESETN	Type B	Input for resetting the APB clock
RSCI3	RSCI_3_TRESETN	Type B	Input for resetting TCLK
RSCI4	RSCI_4_PRESETN	Type B	Input for resetting the APB clock
RSCI4	RSCI_4_TRESETN	Type B	Input for resetting TCLK
RSCI5	RSCI_5_PRESETN	Type B	Input for resetting the APB clock
RSCI5	RSCI_5_TRESETN	Type B	Input for resetting TCLK
RSCI6	RSCI_6_PRESETN	Type B	Input for resetting the APB clock
RSCI6	RSCI_6_TRESETN	Type B	Input for resetting TCLK
RSCI7	RSCI_7_PRESETN	Type B	Input for resetting the APB clock

Table 4.4-2 List of Reset Signals (4/6)

Unit	Signal Name	Reset Control Circuit Type*1	Remarks
RSCI7	RSCI_7_TRESETN	Type B	Input for resetting TCLK
RSCI8	RSCI_8_PRESETN	Type B	Input for resetting the APB clock
RSCI8	RSCI_8_TRESETN	Type B	Input for resetting TCLK
RSCI9	RSCI_9_PRESETN	Type B	Input for resetting the APB clock
RSCI9	RSCI_9_TRESETN	Type B	Input for resetting TCLK
SCIF0	SCIF_0_RST_SYSTEM_N	Type B	System reset (reset by input of a low level)
I3C0	I3C_0_PRESETN	Type B	APB interface reset signal/SCAN reset signal
I3C0	I3C_0_TRESETN	Type B	Reset signal
RIIC8	RIIC_8_MRST	Type B	Reset
RIIC0	RIIC_0_MRST	Type B	Reset
RIIC1	RIIC_1_MRST	Type B	Reset
RIIC2	RIIC_2_MRST	Type B	Reset
RIIC3	RIIC_3_MRST	Type B	Reset
RIIC4	RIIC_4_MRST	Type B	Reset
RIIC5	RIIC_5_MRST	Type B	Reset
RIIC6	RIIC_6_MRST	Type B	Reset
RIIC7	RIIC_7_MRST	Type B	Reset
CANFD	CANFD_0_RSTP_N	Type B	Asynchronous reset (active low) (pclk clock domain)
CANFD	CANFD_0_RSTC_N	Type B	Asynchronous reset (active low) (clkc clock domain)
xSPI	SPI_HRESETN	Type B	Hardware reset
xSPI	SPI_ARESETN	Type B	AXI reset
PFC	IOTOP_0_RESETN	Type B	PFC main reset
PFC	IOTOP_0_ERROR_RESETN	Type B	Reset for the control register related to the WDTUDFCA and WDTUDFCM pins
SD0	SDHI_0_IXRST	Type B	Reset (ch0)
SD1	SDHI_1_IXRST	Type B	Reset (ch1)
SD2	SDHI_2_IXRST	Type B	Reset (ch2)
USB30	—	—	—
USB20	—	—	—
USB21	—	—	—
USB20 USB21	—	—	—
USB20 USB21	—	—	—
GBETH0	—	—	—
GBETH1	—	—	—
PCIE	PCIE_0_ARESETN	Type B	PCIe reset
DDR0	—	—	—
DDR0	—	—	—
DDR0	—	—	—
DDR0	—	—	—
DDR0	—	—	—
DDR0	—	—	—
DDR0	—	—	—
DDR0	—	—	—

Table 4.4-2 List of Reset Signals (5/6)

Unit	Signal Name	Reset Control Circuit Type*1	Remarks
DDR0	—	—	—
DDR0	—	—	—
CRU0	CRU_0_PRESETN	Type B	Reset for the APB system and LINK module
CRU0	CRU_0_ARESETN	Type B	AXI reset
CRU0	CRU_0_S_RESETN	Type B	D-PHY module-related reset
DSI	DSI_0_PRESETN	Type B	APB interface reset
DSI	DSI_0_ARESETN	Type B	AXI reset
LCDC	LCDC_0_RESET_N	Type B	Reset
LCDC	LCDC_1_RESET_N	Type B	Reset
GE3D	GE3D_RESETN	Type B	Reset for GE3D
GE3D	GE3D_AXI_RESETN	Type B	Reset on the AXI slave side
GE3D	GE3D_ACE_RESETN	Type B	Reset on the ACE master side
VCD	VCD_0_RESETN	Type B	Asynchronous
SSIU	SSIF_0_ASYNC_RESET_SSI	Type B	Reset for SSIU common parts
SSIU	SSIF_0_SYNC_RESET_SSI0	Type B	SSI0 module reset
SSIU	SSIF_0_SYNC_RESET_SSI1	Type B	SSI1 module reset
SSIU	SSIF_0_SYNC_RESET_SSI2	Type B	SSI2 module reset
SSIU	SSIF_0_SYNC_RESET_SSI3	Type B	SSI3 module reset
SSIU	SSIF_0_SYNC_RESET_SSI4	Type B	SSI4 module reset
SSIU	SSIF_0_SYNC_RESET_SSI5	Type B	SSI5 module reset
SSIU	SSIF_0_SYNC_RESET_SSI6	Type B	SSI6 module reset
SSIU	SSIF_0_SYNC_RESET_SSI7	Type B	SSI7 module reset
SSIU	SSIF_0_SYNC_RESET_SSI8	Type B	SSI8 module reset
SSIU	SSIF_0_SYNC_RESET_SSI9	Type B	SSI9 module reset
ADMAC	DMACpp_0_ARST	Type B	Unit reset
SCU	SCU_0_RESET_SRU	Type B	Unit reset
ADG	ADG_0_RST_RESET_ADG	Type B	Unit reset
SPDIF0	SPDIF_0_RST	Type B	Unit reset
SPDIF1	SPDIF_1_RST	Type B	Unit reset
SPDIF2	SPDIF_2_RST	Type B	Unit reset
PDM0	PDM_0_PRESETN	Type B	PDM interface reset signal synchronized with the bus clock
PDM0	PDM_0_CRESETN	Type B	PDM interface reset signal synchronized with the core clock
PDM1	PDM_1_PRESETN	Type B	PDM interface reset signal synchronized with the bus clock
PDM1	PDM_1_CRESETN	Type B	PDM interface reset signal synchronized with the core clock
ADC0	ADC_0_ADRST_N	Type B	Unit reset
TSU1	TSU_1_PRESETN	Type B	Unit reset
NPU Bus	ACPU_Peri_NPU0_AXI_RESETN	TYPE-B	UNIT
NPU Bus	ACPU_Peri_NPU1_AXI_RESETN	TYPE-B	UNIT
NPU Bus	NPU_AXI_RESETN	TYPE-B	UNIT
LVDS	LVDS_TOP_RESET_N	TYPE-B	Reset for LVDS
VSPI	VSPI_x_smreset_vspiss_n	TYPE-B	Reset for SubSsystem
VSPI	VSPI_x_sreset_vsp1z	TYPE-B	Reset for VSPI
VSPI	VSPI_x_spreset_fcpvi0	TYPE-B	Reset for FCPVI
NPU	NPU_nRESET	TYPE-B	Reset for NPU
NPU	NPU_nARESET	TYPE-B	AXI Reset

Table 4.4-2 List of Reset Signals (6/6)

Unit	Signal Name	Reset Control Circuit Type*1	Remarks
NPU	NPU_nPRESET	TYPE-B	APB Reset
FDP1	FDP1_reset_brdg_n	TYPE-B	Reset for SubSsystem
FDP1	FDP1_reset_fdp1_p	TYPE-B	Reset for FDP1
FDP1	FDP1_reset_fcpf_p	TYPE-B	Reset for FCPF
OTP	OTPC_0_RESET_N	Type B	Unit reset

Note: For more information, refer to the User's Manual Additional Document.

Note 1. Reset Control Circuit Type
<Type A>
Clock operation is not stopped after release from the reset state. This does not apply to this LSI.
<Type B>
To avoid contention between clock and reset signals, the unit clock of the domain to which a given reset signal belongs is stopped after release from the reset state. (Note that clock signals for some units are not stopped.)
<CA55 reset>
This is a reset control circuit exclusively for use with the CA55. The circuit proceeds with control compliant with the reset constraints during CA55 cold reset by hardware.
<CM33 reset>
This is a reset control circuit exclusively for use with the CM33. The circuit proceeds with control compliant with the reset constraints during CM33 cold reset by hardware.
<CST reset>
This is a reset control circuit exclusively for use with the CST. The circuit proceeds with control by following the conditions for resetting the debug system.

4.4.4 Registers

The base addresses for the CPG registers (SYSREG) are given below.

Table 4.4-3 Register Base Addresses

Base Address Name	Base Address
<CPG_base>	0_1042_0000h (5042_0000h*1, 4042_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

NOTE

- The initial values of these registers are those before forcibly modifying them by hardware (initial values following a reset). For the bits to be forcibly modified by hardware, the note “The setting is to be forcibly modified by hardware,” is added in the following sections.
- Write the values read from reserved bits to them by using a read–modify–write operation.
- When writing to a bit with a write enable bit, set the corresponding xxxx_WEN bit to 1 at the same time.

The CPG registers are listed in the table below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	0000h to 000Fh	-
PLLCM33 Monitor Registers	CPG_PLLCM33_MON	0000_0000h	0010h	32
Reserve	-	-	0014h to 001Fh	-
PLLCLN Standby Control Registers	CPG_PLLCLN_STBY	0000_0000h	0020h	32
Reserve	-	-	0024h to 002Fh	-
PLLCLN Monitor Registers	CPG_PLLCLN_MON	0000_0000h	0030h	32
Reserve	-	-	0034h to 003Fh	-
PLLDTY Standby Control Registers	CPG_PLLDTY_STBY	0000_0004h	0040h	32
Reserve	-	-	0044h to 004Fh	-
PLLDTY Monitor Registers	CPG_PLLDTY_MON	0000_0000h	0050h	32
Reserve	-	-	0054h to 005Fh	-
PLLCA55 Standby Control Registers	CPG_PLLCA55_STBY	0000_0004h	0060h	32
PLLCA55 Output Clock Setting Register 1	CPG_PLLCA55_CLK1	0000_3E82h	0064h	32
PLLCA55 Output Clock Setting Register 2	CPG_PLLCA55_CLK2	000C_1E01h	0068h	32
Reserve	-	-	006Ch to 006Fh	-
PLLCA55 Monitor Registers	CPG_PLLCA55_MON	0000_0000h	0070h	32
Reserve	-	-	0074h to 007Fh	-
PLLVDO Standby Control Registers	CPG_PLLVDO_STBY	0000_0004h	0080h	32
Reserve	-	-	0084h to 008Fh	-
PLLVDO Monitor Registers	CPG_PLLVDO_MON	0000_0000h	0090h	32
Reserve	-	-	0094h to 009Fh	-
PLLETH Standby Control Registers	CPG_PLLETH_STBY	0000_0000h	00A0h	32
Reserve	-	-	00A4h to 00AFh	-
PLLETH Monitor Registers	CPG_PLLETH_MON	0000_0000h	00B0h	32
Reserve	-	-	00B4h to 00BFh	-
PLLDSI0 Standby Control Registers	CPG_PLLDSI0_STBY	0000_0000h	00C0h	32
PLLDSI0 Output Clock Setting Register 1	CPG_PLLDSI0_CLK1	0000_3182h	00C4h	32
PLLDSI0 Output Clock Setting Register 2	CPG_PLLDSI0_CLK2	000C_1803h	00C8h	32
Reserve	-	-	00CCh to 00CFh	-
PLLDSI0 Monitor Registers	CPG_PLLDSI0_MON	0000_0000h	00D0h	32
Reserve	-	-	00D4h to 00DFh	-
PLLDDR0 Standby Control Registers	CPG_PLLDDR0_STBY	0000_0000h	00E0h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	00E4h to 00EFh	-
PLLDDR0 Monitor Registers	CPG_PLLDDR0_MON	0000_0000h	00F0h	32
Reserve	-	-	00F4h to 015Fh	-
PLLDSI1 Standby Control Registers	CPG_PLLDSI1_STBY	0000_0000h	0160h	32
PLLDSI1 Output Clock Setting Register 1	CPG_PLLDSI1_CLK1	0000_3182h	0164h	32
PLLDSI1 Output Clock Setting Register 2	CPG_PLLDSI1_CLK2	000C_1803h	0168h	32
PLLDSI1 Monitor Registers	CPG_PLLDSI1_MON	0000_0000h	0170h	32
Reserve	-	-	0174h to 02FFh	-
Static Mux Control Register 0	CPG_SSEL0	0000_1000h	0300h	32
Static Mux Control Register 1	CPG_SSEL1	0000_0010h	0304h	32
Reserve	-	-	0308h	-
Static Mux Control Register 3	CPG_SSEL3	0000_0000h	030Ch	32
Reserve	-	-	030Ch to 03FFh	-
Dynamic Gear Control Register 0	CPG_CDDIV0	0000_0000h	0400h	32
Dynamic Gear Control Register 1	CPG_CDDIV1	0000_0000h	0404h	32
Dynamic Gear Control Register 2	CPG_CDDIV2	0000_0000h	0408h	32
Dynamic Gear Control Register 3	CPG_CDDIV3	0000_0000h	040Ch	32
Dynamic Gear Control Register 4	CPG_CDDIV4	0000_0000h	0410h	32
Dynamic Gear Control Register 5	CPG_CDDIV5	0000_0000h	0414h	32
Reserve	-	-	0418h to 04FFh	-
Static Gear Control Register 0	CPG_CSDIV0	0000_0000h	0500h	32
Static Gear Control Register 1	CPG_CSDIV1	0000_0000h	0504h	32
Reserve	-	-	0508h to 05FFh	-
CGC Control Register 0	CPG_CLKON_0	0000_0021h	0600h	32
CGC Control Register 1	CPG_CLKON_1	0000_0024h	0604h	32
CGC Control Register 2	CPG_CLKON_2	0000_7BDFh	0608h	32
CGC Control Register 3	CPG_CLKON_3	0000_0000h	060Ch	32
CGC Control Register 4	CPG_CLKON_4	0000_1800h	0610h	32
CGC Control Register 5	CPG_CLKON_5	0000_0000h	0614h	32
CGC Control Register 6	CPG_CLKON_6	0000_0000h	0618h	32
CGC Control Register 7	CPG_CLKON_7	0000_0000h	061Ch	32
CGC Control Register 8	CPG_CLKON_8	0000_0000h	0620h	32
CGC Control Register 9	CPG_CLKON_9	0000_0000h	0624h	32
CGC Control Register 10	CPG_CLKON_10	0000_0000h	0628h	32
CGC Control Register 11	CPG_CLKON_11	0000_0000h	062Ch	32
CGC Control Register 12	CPG_CLKON_12	0000_0000h	0630h	32
CGC Control Register 13	CPG_CLKON_13	0000_0000h	0634h	32
CGC Control Register 14	CPG_CLKON_14	0000_0000h	0638h	32
CGC Control Register 15	CPG_CLKON_15	0000_0000h	063Ch	32
CGC Control Register 16	CPG_CLKON_16	0000_E000h	0640h	32
CGC Control Register 17	CPG_CLKON_17	0000_F000h	0644h	32
CGC Control Register 18	CPG_CLKON_18	0000_0003h	0648h	32
CGC Control Register 19	CPG_CLKON_19	0000_0000h	064Ch	32
CGC Control Register 20	CPG_CLKON_20	0000_0000h	0650h	32
CGC Control Register 21	CPG_CLKON_21	0000_0000h	0654h	32
CGC Control Register 22	CPG_CLKON_22	0000_0000h	0658h	32
CGC Control Register 23	CPG_CLKON_23	0000_0000h	065Ch	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
CGC Control Register 24	CPG_CLKON_24	0000_07FFh	0660h	32
CGC Control Register 25	CPG_CLKON_25	0000_0000h	0664h	32
CGC Control Register 26	CPG_CLKON_26	0000_0000h	0668h	32
CGC Control Register 27	CPG_CLKON_27	0000_0000h	066Ch	32
Reserve	-	-	0670h to 06FFh	-
Dynamic Gear/Mux Status Monitor Register	CPG_CLKSTATUS0	000x_xxxxh	0700h	32
Reserve	-	-	0704h to 07FFh	-
CGC Monitor Register 0	CPG_CLKMON_0	xxxx_xxxxh	0800h	32
CGC Monitor Register 1	CPG_CLKMON_1	xxxx_xxxxh	0804h	32
CGC Monitor Register 2	CPG_CLKMON_2	xxxx_xxxxh	0808h	32
CGC Monitor Register 3	CPG_CLKMON_3	xxxx_xxxxh	080Ch	32
CGC Monitor Register 4	CPG_CLKMON_4	xxxx_xxxxh	0810h	32
CGC Monitor Register 5	CPG_CLKMON_5	xxxx_xxxxh	0814h	32
CGC Monitor Register 6	CPG_CLKMON_6	xxxx_xxxxh	0818h	32
CGC Monitor Register 7	CPG_CLKMON_7	xxxx_xxxxh	081Ch	32
CGC Monitor Register 8	CPG_CLKMON_8	xxxx_xxxxh	0820h	32
CGC Monitor Register 9	CPG_CLKMON_9	xxxx_xxxxh	0824h	32
CGC Monitor Register 10	CPG_CLKMON_10	xxxx_xxxxh	0828h	32
CGC Monitor Register 11	CPG_CLKMON_11	xxxx_xxxxh	082Ch	32
Reserve	-	-	0830h to 08FFh	-
Reset Control Register 0	CPG_RST_0	0000_0000h	0900h	32
Reset Control Register 1	CPG_RST_1	0000_0038h	0904h	32
Reserve	-	-	0908h	-
Reset Control Register 3	CPG_RST_3	0000_1843h	090Ch	32
Reset Control Register 4	CPG_RST_4	0000_FFC0h	0910h	32
Reset Control Register 5	CPG_RST_5	0000_007Fh	0914h	32
Reset Control Register 6	CPG_RST_6	0000_0000h	0918h	32
Reset Control Register 7	CPG_RST_7	0000_0020h	091Ch	32
Reset Control Register 8	CPG_RST_8	0000_0000h	0920h	32
Reset Control Register 9	CPG_RST_9	0000_0000h	0924h	32
Reset Control Register 10	CPG_RST_10	0000_0060h	0928h	32
Reset Control Register 11	CPG_RST_11	0000_0000h	092Ch	32
Reset Control Register 12	CPG_RST_12	0000_0000h	0930h	32
Reset Control Register 13	CPG_RST_13	0000_0000h	0934h	32
Reset Control Register 14	CPG_RST_14	0000_0000h	0938h	32
Reset Control Register 15	CPG_RST_15	0000_8C00h	093Ch	32
Reset Control Register 16	CPG_RST_16	0000_0000h	0940h	32
Reset Control Register 17	CPG_RST_17	0000_0000h	0944h	32
Reset Control Register 18	CPG_RST_18	0000_0000h	0948h	32
Reserve	-	-	094Ch to 09FFh	-
Reset Monitor Register 0	CPG_RSTMON_0	xxxx_xxxxh	0A00h	32
Reset Monitor Register 1	CPG_RSTMON_1	xxxx_xxxxh	0A04h	32
Reset Monitor Register 2	CPG_RSTMON_2	xxxx_xxxxh	0A08h	32
Reset Monitor Register 3	CPG_RSTMON_3	xxxx_xxxxh	0A0Ch	32
Reset Monitor Register 4	CPG_RSTMON_4	xxxx_xxxxh	0A10h	32
Reset Monitor Register 5	CPG_RSTMON_5	xxxx_xxxxh	0A14h	32
Reset Monitor Register 6	CPG_RSTMON_6	xxxx_xxxxh	0A18h	32
Reset Monitor Register 7	CPG_RSTMON_7	xxxx_xxxxh	0A1Ch	32
Reset Monitor Register 8	CPG_RSTMON_8	xxxx_xxxxh	0A20h	32
Reset Monitor Register 9	CPG_RSTMON_9	xxxx_xxxxh	0A24h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	0A28h to 0AFFh	-
Error Reset Select Register 1	CPG_ERRORRST_SEL1	0000_0000h	0B00h	32
Error Reset Select Register 2	CPG_ERRORRST_SEL2	0000_0000h	0B04h	32
Error Reset Select Register 3	CPG_ERRORRST_SEL3	0000_0000h	0B08h	32
Error Reset Select Register 4	CPG_ERRORRST_SEL4	0000_0000h	0B0Ch	32
Error Reset Select Register 5	CPG_ERRORRST_SEL5	0000_0000h	0B10h	32
Error Reset Select Register 6	CPG_ERRORRST_SEL6	0000_0000h	0B14h	32
Error Reset Select Register 7	CPG_ERRORRST_SEL7	0000_0000h	0B18h	32
Error Reset Select Register 8	CPG_ERRORRST_SEL8	0000_0000h	0B1Ch	32
Reserve	-	-	0B20h to 0B3Fh	-
Error Reset Register 2	CPG_ERROR_RST2	0000_0000h	0B40h	32
Error Reset Register 3	CPG_ERROR_RST3	0000_0000h	0B44h	32
Error Reset Register 4	CPG_ERROR_RST4	0000_0000h	0B48h	32
Error Reset Register 5	CPG_ERROR_RST5	0000_0000h	0B4Ch	32
Error Reset Register 6	CPG_ERROR_RST6	0000_0000h	0B50h	32
Error Reset Register 7	CPG_ERROR_RST7	0000_0000h	0B54h	32
Error Reset Register 8	CPG_ERROR_RST8	0000_0000h	0B58h	32
Reserve	-	-	0B5Ch to 0BFFh	-
Low-Power-Consumption Sequence Control Register 1	CPG_LP_CTL1	0000_0000h	0C00h	32
Low-Power-Consumption Sequence Control Register 2	CPG_LP_CTL2	0000_0000h	0C04h	32
GE3D Low-Power-Consumption Sequence Control Register	CPG_LP_GE3D_CTL	xxxx_1Fxxh	0C08h	32
CM33 Control register	CPG_CM33_CTL	0000_0000h	0C0Ch	32
Reserve	-	-	0C10h to 0C17h	-
Low-Power Sequence Cortex-M33 Control Register 0	CPG_LP_CM33CTL0	0x0x_0xxxh	0C18h	32
Low-Power Sequence Cortex-M33 Control Register 1	CPG_LP_CM33CTL1	xxxx_31xxh	0C1Ch	32
Cortex-A55 Clock Control Register 1	CPG_LP_CA55_CTL1	0000_0x90h	0C20h	32
Cortex-A55 Clock Control Register 2	CPG_LP_CA55_CTL2	0xx0_0xx0h	0C24h	32
Cortex-A55 Clock Control Register 3	CPG_LP_CA55_CTL3	0xx0_0xx0h	0C28h	32
Reserve	-	-	0C2Ch to 0C33h	-
Cortex-A55 Clock Control Register 6	CPG_LP_CA55_CTL6	0000_000xh	0C34h	32
Cortex-A55 Clock Control Register 7	CPG_LP_CA55_CTL7	0000_xx00h	0C38h	32
Reserve	-	-	0C3Ch to 0C48h	-
Low-Power Sequence Control Register	CPG_LP_PMU_CTL1	0000_00xxh	0C4Ch	32
SRAM Standby Control Register 1	CPG_LP_SRAM_STBY_CTL1	0000_0000h	0C50h	32
SRAM Standby Control Register 2	CPG_LP_SRAM_STBY_CTL2	0000_0000h	0C54h	32
SRAM Standby Control Register 3	CPG_LP_SRAM_STBY_CTL3	0000_0000h	0C58h	32
Reserve	-	-	0C5Ch to 0C6Fh	-
CST Control Register 2	CPG_LP_CST_CTL2	0x0x_0x0xh	0C70h	32
Reserve	-	-	0C78h to 0C7Fh	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	0C80h to 0C87h	-
OTP Handshaking Monitor Register	CPG_OTP_HANDSHAKE_MON	0000_000xh	0C88h	32
Reset Control Register for the PD_OTHERS Domain	CPG_OTHERS_INI	0000_000xh	0C8Ch	32
Reserve	-	-	0C90h	-
NPU Control Register 1	CPG_LP_NPU_CTL1	xxxx_xxxxh	0C94h	32
NPU Control Register 2	CPG_LP_NPU_CTL2	0000_000xh	0C98h	32
VSPI Control Register 1	CPG_LP_VSPI_CTL1	0xxx_xxxxh	0C9Ch	32
VSPI Control Register 2	CPG_LP_VSPI_CTL2	0000_000xh	0CA0h	32
FDP1 Control Register	CPG_LP_FDP1_CTL	xxxx_xxxxh	0CA4h	32
Reserve	-	-	0CA8h to 0CFFh	-
MSTOP Register 1	CPG_BUS_1_MSTOP	0000_0FFFh	0D00h	32
MSTOP Register 2	CPG_BUS_2_MSTOP	0000_E03Fh	0D04h	32
MSTOP Register 3	CPG_BUS_3_MSTOP	0000_FE1Ch	0D08h	32
MSTOP Register 4	CPG_BUS_4_MSTOP	0000_C027h	0D0Ch	32
MSTOP Register 5	CPG_BUS_5_MSTOP	0000_FEE1h	0D10h	32
MSTOP Register 6	CPG_BUS_6_MSTOP	0000_FFFFh	0D14h	32
MSTOP Register 7	CPG_BUS_7_MSTOP	0000_5F83h	0D18h	32
MSTOP Register 8	CPG_BUS_8_MSTOP	0000_007Ch	0D1Ch	32
MSTOP Register 9	CPG_BUS_9_MSTOP	0000_F810h	0D20h	32
MSTOP Register 10	CPG_BUS_10_MSTOP	0000_B8AEh	0D24h	32
MSTOP Register 11	CPG_BUS_11_MSTOP	0000_FFFFh	0D28h	32
MSTOP Register 12	CPG_BUS_12_MSTOP	0000_0201h	0D2Ch	32
MSTOP Register 13	CPG_BUS_13_MSTOP	0000_07FFh	0D30h	32

4.4.4.1 pllname Standby Control Registers (CPG_pllname_STBY)

[pllname]

PLLCM33, PLLCLN, PLLDTY, PLLCA55, PLLVDO, PLLETH, PLLDSI0/1, PLLDDR0

These registers control the standby states of the individual PLLs.

Remark Except for PLL_DSI0 and PLLDSI1, use SSC_MODE = 00b.
In the case of MIPI-DSI, SSC is disabled. (set SSC_EN=0b)
For LVDS, the upper limit of the clock rate must not exceed 87 MHz.

Access Size : 32 bits
Address : <CPG_base> + 0020h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	SSC_MODE_WEN	-	SSC_EN_WEN	-	RESETB_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	W	R	W	R	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	SSC_MODE[1:0]		-	SSC_EN	-	RESETB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	x	0	x
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	R	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
20	SSC_MODE_WEN	0h	W	pllname write enable for SSC_MODE
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
18	SSC_EN_WEN	0h	W	pllname write enable for SSC_EN
17	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
16	RESETB_WEN	0h	W	pllname write enable for RESETB
15 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
5 to 4	SSC_MODE[1:0]	0h	RW	SSC Center/Down Spread setting 00b : Down spread 10b : Center spread Others: Setting prohibited
3	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
2	SSC_EN	x	RW	pllname SSCG on/off setting 0b: SSCG off 1b: SSCG on
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
0	RESETB	x	RW	pllname reset setting 0b: Reset state 1b: Active (reset released state))

x: Undefined value

Note: Regarding the initial value, refer to **Table 4.4-4**.

For details on the registers such as offset addresses, refer to the table below.

Table 4.4-4 Details on the CPG_pllname_STBY Registers such as Offset Addresses

Register Name	Offset Address	Initial Value	SSC_EN Bit
CPG_PLLCM33_STBY	Reserved	Reserved	Reserved
CPG_PLLCLN_STBY	020h	0000_0000h*1	Reserved
CPG_PLLDTY_STBY	040h	0000_0004h*1	Enable
CPG_PLLCA55_STBY	060h	0000_0004h*1	Enable
CPG_PLLVDO_STBY	080h	0000_0004h*1	Enable
CPG_PLLETH_STBY	0A0h	0000_0000h	Reserved
CPG_PLLDSI0_STBY	0C0h	0000_0000h	Enable
CPG_PLLDDR0_STBY	0E0h	0000_0000h	Enable
CPG_PLLDSI1_STBY	160h	0000_0000h	Enable

Note 1. The setting is to be forcibly modified by hardware.
The settings may differ depending on the boot mode.

4.4.4.2 pllname Output Clock Setting Registers 1 (CPG_pllname_CLK1)

[pllname]

PLLCM33, PLLCLN, PLLDTY, PLLCA55, PLLVDO, PLLETH, PLLDSI0/1, PLLDDR0

These registers set the clocks output by the individual PLLs.

Access Size : 32 bits
Address : <CPG_base> + 0064h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DIV_K[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DIV_M[9:0]										DIV_P[5:0]					
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	DIV_K[15:0]	0h	RW	pllname delta-sigma modulator (DSM) setting
15 to 6	DIV_M[9:0]	x	RW	pllname main-divider setting
5 to 0	DIV_P[5:0]	x	RW	pllname pre-divider setting

x: Undefined value

Note: Regarding the initial value, refer to **Table 4.4-5**.

For details on the registers such as offset addresses, refer to the table below.

Table 4.4-5 Details on the CPG_pllname_CLK1 Registers such as Offset Addresses

Register Name	Offset Address	Initial Value
CPG_PLLCM33_CLK1	Reserved	Reserved
CPG_PLLCLN_CLK1	Reserved	Reserved
CPG_PLLDTY_CLK1	Reserved	Reserved
CPG_PLLCA55_CLK1	064h	0000_3E82h*1
CPG_PLLVDO_CLK1	Reserved	Reserved
CPG_PLLETH_CLK1	Reserved	Reserved
CPG_PLLDSI0_CLK1	0C4h	0000_3182h
CPG_PLLDDR0_CLK1	Reserved	Reserved
CPG_PLLDSI1_CLK1	164h	0000_3182h

Note 1. The setting is to be forcibly modified by hardware.
The settings may differ depending on the boot mode.

Note 2. When this clock is to be used as the source clock for CA55 SCLK, the frequency is fixed to 1260 MHz so making the setting is prohibited. When the clock is not to be used as the source clock, making the setting is possible (for up to 1260 MHz).

For the register settings, refer to the table below.

Table 4.4-6 CPG_pllname_CLK1 Register Settings

Bit	Bit Name	Description
31 to 16	DIV_K	For setting the delta-sigma modulator (DSM)*1 The values in the range from −32,768 to 32,767 (decimal numbers) that satisfy the condition in Note 2 are specifiable as two's complements. 1000_0000_0000_0000b: k = −32,768 1000_0000_0000_0001b: k = −32,767 1000_0000_0000_0010b: k = −32,766 ⋮ 0111_1111_1111_1101b: k = +32,765 0111_1111_1111_1110b: k = +32,766 0111_1111_1111_1111b: k = +32,767
15 to 6	DIV_M	For setting the main divider*1 The values in the range from 64 to 533 that satisfy the conditions in Notes 2 and 3 are specifiable as unsigned binary numbers. 11_1111_1111b to 10_0001_0110b ("m" = 534): Setting prohibited 10_0001_0101b: Division value "m" = 533 10_0001_0100b: Division value "m" = 532 ⋮ 00_0100_0001b: Division value "m" = 65 00_0100_0000b: Division value "m" = 64 00_0011_1111b to 00_0000_0000b: Setting prohibited
5 to 0	DIV_P	For setting the pre-divider*1 The values in the range from 1 to 4 that satisfy the conditions in Notes 2 and 3 are specifiable. 11_1111b to 00_00_0101b: Setting prohibited 00_0100b: Division value "p" = 4 (FFREF = 6 MHz) 00_0011b: Division value "p" = 3 (FFREF = 8 MHz) 00_0010b: Division value "p" = 2 (FFREF = 12 MHz) 00_0001b: Division value "p" = 1 (FFREF = 24 MHz) 00_0000b: Setting prohibited

Note 1. Changing a setting requires resetting the corresponding PLL. For details on the procedure for doing so, refer to **4.4.9.4 Procedure for Setting PLLs**.

Note 2. Refer to the expressions to calculate the PLL oscillation frequency (F_{FVCO}) and output frequency (F_{FOUT}) in **4.4.5.3.5 PLL settings**.

Note 3. Refer to the table of the allowed range of PLL frequencies in **4.4.5.3.5 PLL settings**.

4.4.4.3 pllname Output Clock Setting Registers 2 (CPG_pllname_CLK2)

[pllname]

PLLCM33, PLLCLN, PLLDTY, PLLCA55, PLLVDO, PLLETH, PLLDSI0/1, PLLDDR0

These registers set the clocks output by the individual PLLs.

Remark MFR bit and MRR bit other than PLL_DSI0 and PLLDSI1 should not be changed from the read values.

Access Size : 32 bits
Address : <CPG_base> + 0068h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	MFR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	MRR[5:0]						-	-	-	-	-	DIV_S[2:0]		
Initial Value	0	0	x	x	x	x	x	x	0	0	0	0	0	x	x	x
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
23 to 16	MFR[7:0]	x	RW	Setting the Modulation Frequency
15 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
13 to 8	MRR[5:0]	x	RW	Setting the Modulation Rate
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
2 to 0	DIV_S[2:0]	x	RW	pllname divider S setting

x: Undefined value

Note: Regarding the initial value, refer to **Table 4.4-7**.

For details on the registers such as offset addresses, refer to the table below.

Table 4.4-7 Details on the CPG_pllname_CLK2 Registers such as Offset Addresses

Register Name	Offset Address	Initial Value
CPG_PLLCM33_CLK2	Reserved	Reserved
CPG_PLLCLN_CLK2	Reserved	Reserved
CPG_PLLDTY_CLK2	Reserved	Reserved
CPG_PLLCA55_CLK2	068h	000C_1E01h*1
CPG_PLLVDO_CLK2	Reserved	Reserved
CPG_PLLETH_CLK2	Reserved	Reserved
CPG_PLLDSI0_CLK2	0C8h	000C_1803h
CPG_PLLDDR0_CLK2	Reserved	Reserved
CPG_PLLDSI1_CLK2	168h	000C_1803h

Note 1. The setting is to be forcibly modified by hardware.
The settings may differ depending on the boot mode.

Note 2. When this clock is to be used as the source clock for CA55 SCLK, the frequency is fixed to 1260 MHz so making the setting is prohibited. When the clock is not to be used as the source clock, making the setting is possible (for up to 1260 MHz).

Table 4.4-8 CPG_pllname_CLK2 Register Settings

Bit	Bit Name	Description
2 to 0	DIV_S	For setting divider S*1 The values in the range from 0 to 6 (decimal numbers) are specifiable. 111b: Setting prohibited 110b: Division value “s” = 6 (Division ratio = 64) 101b: Division value “s” = 5 (Division ratio = 32) 100b: Division value “s” = 4 (Division ratio = 16) 011b: Division value “s” = 3 (Division ratio = 8) 010b: Division value “s” = 2 (Division ratio = 4) 001b: Division value “s” = 1 (Division ratio = 2) 000b: Division value “s” = 0 (Division ratio = 1)

Note 1. For details on DIV_M (“m”) and DIV_P (“p”), refer to PLL output clock setting registers 1 (CPG_pllname_CLK1).

4.4.4.4 pllname Monitor Registers (CPG_pllname_MON)

[pllname]

PLLCM33, PLLCLN, PLLDTY, PLLCA55, PLLVDO, PLLETH, PLLDSI0/1, PLLDDR0

These registers are used to read the states of the individual PLLs

Access Size : 32 bits
Address : <CPG_base> + 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	PLLn_LOCK	-	-	-	PLLn_RESETB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
4	PLLn_LOCK	0h	R	pllname locked-state monitor 0b: PLL not locked in 1b: PLL is locked in
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
0	PLLn_RESETB	0h	R	pllname operating mode monitor 0b: Reset state (standby mode) 1b: Normal mode

For details on the offset addresses, refer to the table below.

Table 4.4-9 CPG_pllname_MON Offset Addresses

Register Name	Offset Address
CPG_PLLCM33_MON	010h
CPG_PLLCLN_MON	030h
CPG_PLLDTY_MON	050h
CPG_PLLCA55_MON	070h
CPG_PLLVDO_MON	090h
CPG_PLLETH_MON	0B0h
CPG_PLLDSI0_MON	0D0h
CPG_PLLDDR0_MON	0F0h
CPG_PLLDSI1_MON	170h

4.4.4.5 Static Mux Control Registers (CPG_SSELm) (m = 0 to 3)

These registers control the static multiplexors. For details on the registers, refer to **Table 4.4-10** and block diagrams of the clock system.

Access Size : 32 bits
Address : <CPG_base> + 0300h + m x 0004h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	SELCTL3WEN	-	-	-	SELCTL2WEN	-	-	-	SELCTL1WEN	-	-	-	SELCTL0WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	W	R	R	R	W	R	R	R	W	R	R	R	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SELCTL3	-	-	-	SELCTL2	-	-	-	SELCTL1	-	-	-	SELCTL0
Initial Value	0	0	0	x	0	0	0	x	0	0	0	x	0	0	0	x
R/W	R	R	R	RW	R	R	R	RW	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
28	SELCTL3WEN	0h	W	Write enable for SELCTL3
27 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
24	SELCTL2WEN	0h	W	Write enable for SELCTL2
23 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
20	SELCTL1WEN	0h	W	Write enable for SELCTL1
19 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
16	SELCTL0WEN	0h	W	Write enable for SELCTL0
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
12	SELCTL3	x	RW	Refer to **Table 4.4-10.**
11 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
8	SELCTL2	x	RW	Refer to **Table 4.4-10.**
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
4	SELCTL1	x	RW	Refer to **Table 4.4-10.**
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
0	SELCTL0	x	RW	Refer to **Table 4.4-10.**

x: Undefined value

Note: Regarding the initial value, refer to **Table 4.4-10**.

Table 4.4-10 Specifications of the CPG_SSELm Registers

Register Name	SELCTL0	SELCTL1	SELCTL2	SELCTL3
CPG_SSEL0	SMUX2_CA55_SCLK1*1 0b: SMUX2_CA55_SCLK0 (default) 1b: PLLCA55	SMUX2_CA55_SCLK0*1 0b: PLLVDO (default) 1b: PLLDTY	SMUX2_GBE0_TXCLK 0b: CSDIV_2to100_PLLETH_GBE0 (default) 1b: ET0_TXC_TXCLK	SMUX2_GBE0_RXCLK 0b: CSDIV_2to100_PLLETH_GBE0 1b: ET0_RXC_RXCLK (default)
CPG_SSEL1	SMUX2_GBE1_TXCLK 0b: CSDIV_2to100_PLLETH_GBE1 (default) 1b: ET1_TXC_TXCLK	SMUX2_GBE1_RXCLK 0b: CSDIV_2to100_PLLETH_GBE1 1b: ET1_RXC_RXCLK (default)	SMUX2_XSPI_CLK0 0b: CDIV3_CLK533_PLLCM33_FIX (default) 1b: CDIV4_CLK400_PLLCM33_FIX	SMUX2_XSPI_CLK1 0b: SMUX2_XSPI_CLK0 (default) 1b: CDIV5_CLK320_PLLCM33_FIX
CPG_SSEL3	SMUX2_DSI0_CLK*2 0b: CDIV7_DSI0_CLK (default) 1b: CSDIV_2to16_PLLDSI0	SMUX2_DSI1_CLK*2 0b: CDIV7_DSI1_CLK (default) 1b: CSDIV_2to16_PLLDSI1	Reserved	Reserved

Note 1. The setting is to be forcibly modified by hardware.
The settings may differ depending on the boot mode.

Note 2. If LVDS0 / LVDS1 is used, be sure to set 0b.

4.4.4.6 Dynamic Gear Control Registers (CPG_CDDIVm) (m = 0 to 5)

These registers control dynamic gears (counter type). For details on the registers, refer to **Table 4.4-11** and block diagrams of the clock system.

Access Size : 32 bits
Address : <CPG_base> + 0400h + m x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	DIVCTL 3WEN	-	-	-	DIVCTL 2WEN	-	-	-	DIVCTL 1WEN	-	-	-	DIVCTL 0WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	W	R	R	R	W	R	R	R	W	R	R	R	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	DIVCTL3[2:0]			-	DIVCTL2[2:0]			-	DIVCTL1[2:0]			-	DIVCTL0[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
28	DIVCTL3WEN	0h	W	Write enable for DIVCTL3
27 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
24	DIVCTL2WEN	0h	W	Write enable for DIVCTL2
23 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
20	DIVCTL1WEN	0h	W	Write enable for DIVCTL1
19 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
16	DIVCTL0WEN	0h	W	Write enable for DIVCTL0
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
14 to 12	DIVCTL3[2:0]	0h	RW	Refer to **Table 4.4-11**.
11	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
10 to 8	DIVCTL2[2:0]	0h	RW	Refer to **Table 4.4-11**.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
6 to 4	DIVCTL1[2:0]	0h	RW	Refer to **Table 4.4-11**.
3	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
2 to 0	DIVCTL0[2:0]	0h	RW	Refer to **Table 4.4-11**.

Table 4.4-11 Specifications of the CPG_CDDIVm Registers

Register Name	DIVCTL0	DIVCTL1	DIVCTL2	DIVCTL3
CPG_CDDIV0	CDDIV_2to64_PLLCM33_CST400 000b: 1/2 001b: 1/4 010b: 1/8 011b: 1/16 100b: 1/64 Others: Setting prohibited	CDDIV_2to64_PLLCM33 000b: 1/2 001b: 1/4 010b: 1/8 011b: 1/16 100b: 1/64 Others: Setting prohibited	CDDIV_2to64_PLLDTY_ACPU*1 000b: 1/2 001b: 1/4 010b: 1/8 011b: 1/16 100b: 1/64 Others: Setting prohibited	CDDIV_1to8_PLLCA55_SCLK*1,*2 00b: 1/1 01b: 1/2 10b: 1/4 11b: 1/8 DIVCTL3[2] is reserved. When writing, read modify write is needed.
CPG_CDDIV1	CDDIV_1to8_CLK1800_PLLCA55_CA55_0*1,*2 00b: 1/1 01b: 1/2 10b: 1/4 11b: 1/8 DIVCTL0[2] is reserved. When writing, read modify write is needed.	CDDIV_1to8_CLK1800_PLLCA55_CA55_1*1,*2 00b: 1/1 01b: 1/2 10b: 1/4 11b: 1/8 DIVCTL1[2] is reserved. When writing, read modify write is needed.	CDDIV_1to8_CLK1800_PLLCA55_CA55_2*1,*2 00b: 1/1 01b: 1/2 10b: 1/4 11b: 1/8 DIVCTL2[2] is reserved. When writing, read modify write is needed.	CDDIV_1to8_CLK1800_PLLCA55_CA55_3*1,*2 00b: 1/1 01b: 1/2 10b: 1/4 11b: 1/8 DIVCTL3[2] is reserved. When writing, read modify write is needed.
CPG_CDDIV2	CDDIV_2to16_CA55PERI*1 00b: 1/2 01b: 1/4 10b: 1/8 11b: 1/16 DIVCTL0[2] is reserved. When writing, read modify write is needed.	Reserved	Reserved	Reserved
CPG_CDDIV3	Reserved	CDDIV_2to64_PLLVDO_GPU*1 000b: 1/2 001b: 1/4 010b: 1/8 011b: 1/16 100b: 1/64 Others: Setting prohibited	CDDIV_2to64_PLLDTY_RCPU*1 000b: 1/2 001b: 1/4 010b: 1/8 011b: 1/16 100b: 1/64 Others: Setting prohibited	CDDIV_2to4_PLLVDO_CRU0*1 0b: 1/2 1b: 1/4 DIVCTL3[2:1] is reserved. When writing, read modify write is needed.
CPG_CDDIV4	Reserved	Reserved	Reserved	CDDIV_1to4_PLLETH_NPU*1, 00b: 1/1 01b: 1/2 10b: 1/4 11b: Setting prohibited DIVCTL3[2] is reserved. When writing, read modify write is needed.

Table 4.4-11 Specifications of the CPG_CDDIVm Registers

Register Name	DIVCTL0	DIVCTL1	DIVCTL2	DIVCTL3
CPG_CDDIV5	CDDIV_2to64_PLLDTY_NPU*1 000b: 1/2 001b: 1/4 010b: 1/8 011b: 1/16 100b: 1/64 Others: Setting prohibited	Reserved	Reserved	Reserved

Note 1. The setting is initialized by setting the CPG_OTHERS_INI.OTHERS_RST bit to 1.

Note 2. The setting is to be forcibly modified by hardware.
The settings may differ depending on the boot mode.

4.4.4.7 Static Gear Control Registers (CPG_CSDIVm) (m = 0, 1)

These registers control the static gears (counter type). For details on the registers, refer to **Table 4.4-12** and block diagrams of the clock system.

Access Size : 32 bits
Address : <CPG_base> + 0500h + m x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	DIVCTL 3WEN	-	-	-	DIVCTL 2WEN	-	-	-	DIVCTL 1WEN	-	-	-	DIVCTL 0WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	W	R	R	R	W	R	R	R	W	R	R	R	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DIVCTL3[3:0]				DIVCTL2[3:0]				-	-	DIVCTL1[1:0]		-	-	DIVCTL0[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
28	DIVCTL3WEN	0h	W	Write enable for DIVCTL3
27 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
24	DIVCTL2WEN	0h	W	Write enable for DIVCTL2
23 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
20	DIVCTL1WEN	0h	W	Write enable for DIVCTL1
19 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
16	DIVCTL0WEN	0h	W	Write enable for DIVCTL0
15 to 12	DIVCTL3[3:0]	0h	RW	Refer to **Table 4.4-12**.
11 to 8	DIVCTL2[3:0]	0h	RW	Refer to **Table 4.4-12**.
7 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
5 to 4	DIVCTL1[1:0]	0h	RW	Refer to **Table 4.4-12**.
3 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
1 to 0	DIVCTL0[1:0]	0h	RW	Refer to **Table 4.4-12**.

Table 4.4-12 Specifications of the CPG_CSDIVm Registers

Register Name	DIVCTL0	DIVCTL1	DIVCTL2	DIVCTL3
CPG_CSDIV0	CSDIV_2to100_PLLETH_GBE0 00b: 1/2 01b: 1/10 10b: 1/100 Others: Setting prohibited	CSDIV_2to100_PLLETH_GBE1 00b: 1/2 01b: 1/10 10b: 1/100 Others: Setting prohibited	CSDIV_16to128_PLLETH_LPCLK 00b: 1/16 01b: 1/32 10b: 1/64 11b: 1/128 DIVCTL2[3:2] is reserved. When writing, read modify write is needed.	CSDIV_2to16_PLLCM33_XSPI 00b: 1/2 01b: 1/4 10b: 1/8 11b: 1/16
CPG_CSDIV1	CSDIV_8to10_PLLCM33_ADC_PCLK*1 0b: 1/8 1b: 1/10 DIVCTL0[1] is reserved. When writing, read modify write is needed.	CSDIV_2to16_PLLCM33_ADC_ADCLK*1 00b: 1/2 01b: 1/4 10b: 1/8 11b: 1/16	CSDIV_2to16_PLLDSI0 0000b: 1/2 0001b: 1/4 0010b: 1/6 0011b: 1/8 ⋮ 0111b: 1/16	CSDIV_2to16_PLLDSI1 0000b: 1/2 0001b: 1/4 0010b: 1/6 0011b: 1/8 ⋮ 0111b: 1/16

Note 1. ADC_0_ADCLK frequency follows the table below.

ADC_0_ADCLK frequency	CPG_CSDIV1.DIVCTL0	CPG_CSDIV1.DIVCTL1
50 MHz	0b	00b
40 MHz	1b	00b
20 MHz	1b	01b
10 MHz	1b	10b
5 MHz	1b	11b

CAUTION

These registers do not cover dynamic gearing.

4.4.4.8 CGC Control Registers (CPG_CLKON_m) (m = 0 to 27)

These registers control CGCs to switch clock signals on and off. For details on the registers, refer to **Table 4.4-13** to **Table 4.4-18** and block diagrams of the clock system.

Access Size : 32 bits
Address : <CPG_base> + 0600h + m x 0004h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CLK15_ONWEN	CLK14_ONWEN	CLK13_ONWEN	CLK12_ONWEN	CLK11_ONWEN	CLK10_ONWEN	CLK9_ONWEN	CLK8_ONWEN	CLK7_ONWEN	CLK6_ONWEN	CLK5_ONWEN	CLK4_ONWEN	CLK3_ONWEN	CLK2_ONWEN	CLK1_ONWEN	CLK0_ONWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLK15_ON	CLK14_ON	CLK13_ON	CLK12_ON	CLK11_ON	CLK10_ON	CLK9_ON	CLK8_ON	CLK7_ON	CLK6_ON	CLK5_ON	CLK4_ON	CLK3_ON	CLK2_ON	CLK1_ON	CLK0_ON
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CLK15_ONWEN	0h	W	Write enable for CLK15_ON
30	CLK14_ONWEN	0h	W	Write enable for CLK14_ON
29	CLK13_ONWEN	0h	W	Write enable for CLK13_ON
28	CLK12_ONWEN	0h	W	Write enable for CLK12_ON
27	CLK11_ONWEN	0h	W	Write enable for CLK11_ON
26	CLK10_ONWEN	0h	W	Write enable for CLK10_ON
25	CLK9_ONWEN	0h	W	Write enable for CLK9_ON
24	CLK8_ONWEN	0h	W	Write enable for CLK8_ON
23	CLK7_ONWEN	0h	W	Write enable for CLK7_ON
22	CLK6_ONWEN	0h	W	Write enable for CLK6_ON
21	CLK5_ONWEN	0h	W	Write enable for CLK5_ON
20	CLK4_ONWEN	0h	W	Write enable for CLK4_ON
19	CLK3_ONWEN	0h	W	Write enable for CLK3_ON
18	CLK2_ONWEN	0h	W	Write enable for CLK2_ON
17	CLK1_ONWEN	0h	W	Write enable for CLK1_ON
16	CLK0_ONWEN	0h	W	Write enable for CLK0_ON
15	CLK15_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
14	CLK14_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
13	CLK13_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
12	CLK12_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
11	CLK11_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
10	CLK10_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
9	CLK9_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON

Bit	Bit Name	Initial Value	R/W	Description
8	CLK8_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
7	CLK7_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
6	CLK6_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
5	CLK5_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
4	CLK4_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
3	CLK3_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
2	CLK2_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
1	CLK1_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON
0	CLK0_ON	x	RW	For the target signal, refer to **Table 4.4-13** to **Table 4.4-18**. 0b: Clock OFF 1b: Clock ON

x: Undefined value

Note: Regarding the initial values, refer to the tables.

Table 4.4-13 Specifications of the CPG_CLKON_m Registers (m = 0 to 4)

Spec.	CPG_CLKON_0	CPG_CLKON_1	CPG_CLKON_2	CPG_CLKON_3	CPG_CLKON_4
Offset address	0600h	0604h	0608h	060Ch	0610h
Initial value	0000_0021h	0000_0024h	0000_7BDFh	0000_0000h	0000_1800h
CLK0_ON	CGC_DMAC_0_ACLK	Reserved	CGC_BTROM_0_ACLK	CGC_MHU_0_pclk	CGC_RCPU_CMTW_5_clkm
CLK1_ON	CGC_DMAC_1_ACLK	Reserved	CGC_CST_0_cs_clk	CGC_GPT_0_pclk_sfr	CGC_RCPU_CMTW_6_clkm
CLK2_ON	CGC_DMAC_2_ACLK	CGC_CM33_CLK1*1	CGC_CST_0_ts_clk	CGC_GPT_1_pclk_sfr	CGC_RCPU_CMTW_7_clkm
CLK3_ON	CGC_DMAC_3_ACLK	CGC_GIC_0_gicclk*1	CGC_CST_0_apb_sb_clk	CGC_POEGA_0_PCLK	CGC_MCPU_GTM_0_PCLK
CLK4_ON	CGC_DMAC_4_ACLK	Reserved	CGC_CST_0_apb_ca55_clk	CGC_POEGB_0_PCLK	CGC_MCPU_GTM_1_PCLK
CLK5_ON	CGC_ICU_0_PCLK_I	CGC_SRAM_0_ACLK	Reserved	CGC_POEGC_0_PCLK	CGC_ACPU_GTM_0_PCLK
CLK6_ON	CGC_CRC_0_clk_crc	Reserved	CGC_CST_0_apb_cm33_clk	CGC_POEGD_0_PCLK	CGC_ACPU_GTM_1_PCLK
CLK7_ON	CGC_CA55_0_SCLK*1	Reserved	CGC_CST_0_ahb_cm33_clk	CGC_POEGA_1_PCLK	CGC_RCPU_GTM_0_PCLK
CLK8_ON	CGC_CA55_0_PCLK*1	CGC_SRAM_1_ACLK*1	CGC_CST_0_ahb_ath_clk	CGC_POEGB_1_PCLK	CGC_RCPU_GTM_1_PCLK
CLK9_ON	CGC_CA55_0_ATCLK*1	Reserved	CGC_CST_0_atb_ca55_clk	CGC_POEGC_1_PCLK	CGC_RCPU_GTM_2_PCLK
CLK10_ON	CGC_CA55_0_GICCLK*1	Reserved	Reserved	CGC_POEGD_1_PCLK	CGC_RCPU_GTM_3_PCLK
CLK11_ON	CGC_CA55_0_ACLK*1	Reserved	CGC_CST_0_atb_cm33_clk	CGC_MCPU_CMTW_0_clkm	CGC_WDT_0_clkp*1
CLK12_ON	CGC_CA55_0_TSCLK*1	Reserved	CGC_CST_0_atb_cst_clk	CGC_MCPU_CMTW_1_clkm	CGC_WDT_0_clk_loco*1
CLK13_ON	Reserved	Reserved	CGC_CST_0_axi_etr_clk	CGC_MCPU_CMTW_2_clkm	CGC_WDT_1_clkp
CLK14_ON	Reserved	Reserved	CGC_CST_0_axi_sb_clk	CGC_MCPU_CMTW_3_clkm	CGC_WDT_1_clk_loco
CLK15_ON	Reserved	Reserved	CGC_SYC_0_CNT_CLK*1	CGC_RCPU_CMTW_4_clkm	CGC_WDT_2_clkp

Note 1. The setting is to be forcibly modified by hardware.
The settings may differ depending on the boot mode.

Table 4.4-14 Specifications of the CPG_CLKON_m Registers (m = 5 to 9)

Spec.	CPG_CLKON_5	CPG_CLKON_6	CPG_CLKON_7	CPG_CLKON_8	CPG_CLKON_9
Offset address	0614h	0618h	061Ch	0620h	0624h
Initial value	0000_0000h	0000_0000h	0000_0000h	0000_0000h	0000_0000h
CLK0_ON	CGC_WDT_2_clk_loco	CGC_RSCI_0_ps_ps2_n	CGC_RSCI_3_ps_ps1_n	CGC_RSCI_7_PCLK	CGC_I3C_0_PCLKRW
CLK1_ON	CGC_WDT_3_clkp	CGC_RSCI_0_ps_ps1_n	CGC_RSCI_4_PCLK	CGC_RSCI_7_TCLK	CGC_I3C_0_PCLK
CLK2_ON	CGC_WDT_3_clk_loco	CGC_RSCI_1_PCLK	CGC_RSCI_4_TCLK	CGC_RSCI_7_ps_ps3_n	CGC_I3C_0_TCLK
CLK3_ON	CGC_RTC_0_clk_rtc	CGC_RSCI_1_TCLK	CGC_RSCI_4_ps_ps3_n	CGC_RSCI_7_ps_ps2_n	CGC_RIIC_8_ckm
CLK4_ON	CGC_RSPI_0_PCLK	CGC_RSCI_1_ps_ps3_n	CGC_RSCI_4_ps_ps2_n	CGC_RSCI_7_ps_ps1_n	CGC_RIIC_0_ckm
CLK5_ON	CGC_RSPI_0_pclk_sfr	CGC_RSCI_1_ps_ps2_n	CGC_RSCI_4_ps_ps1_n	CGC_RSCI_8_PCLK	CGC_RIIC_1_ckm
CLK6_ON	CGC_RSPI_0_TCLK	CGC_RSCI_1_ps_ps1_n	CGC_RSCI_5_PCLK	CGC_RSCI_8_TCLK	CGC_RIIC_2_ckm
CLK7_ON	CGC_RSPI_1_PCLK	CGC_RSCI_2_PCLK	CGC_RSCI_5_TCLK	CGC_RSCI_8_ps_ps3_n	CGC_RIIC_3_ckm
CLK8_ON	CGC_RSPI_1_pclk_sfr	CGC_RSCI_2_TCLK	CGC_RSCI_5_ps_ps3_n	CGC_RSCI_8_ps_ps2_n	CGC_RIIC_4_ckm
CLK9_ON	CGC_RSPI_1_TCLK	CGC_RSCI_2_ps_ps3_n	CGC_RSCI_5_ps_ps2_n	CGC_RSCI_8_ps_ps1_n	CGC_RIIC_5_ckm
CLK10_ON	CGC_RSPI_2_PCLK	CGC_RSCI_2_ps_ps2_n	CGC_RSCI_5_ps_ps1_n	CGC_RSCI_9_PCLK	CGC_RIIC_6_ckm
CLK11_ON	CGC_RSPI_2_pclk_sfr	CGC_RSCI_2_ps_ps1_n	CGC_RSCI_6_PCLK	CGC_RSCI_9_TCLK	CGC_RIIC_7_ckm
CLK12_ON	CGC_RSPI_2_TCLK	CGC_RSCI_3_PCLK	CGC_RSCI_6_TCLK	CGC_RSCI_9_ps_ps3_n	CGC_CANFD_0_pclk
CLK13_ON	CGC_RSCI_0_PCLK	CGC_RSCI_3_TCLK	CGC_RSCI_6_ps_ps3_n	CGC_RSCI_9_ps_ps2_n	CGC_CANFD_0_clk_ram
CLK14_ON	CGC_RSCI_0_TCLK	CGC_RSCI_3_ps_ps3_n	CGC_RSCI_6_ps_ps2_n	CGC_RSCI_9_ps_ps1_n	CGC_CANFD_0_clkc
CLK15_ON	CGC_RSCI_0_ps_ps3_n	CGC_RSCI_3_ps_ps2_n	CGC_RSCI_6_ps_ps1_n	CGC_SCIF_0_clk_pck	CGC_SPI_HCLK

Table 4.4-15 Specifications of the CPG_CLKON_m Registers (m = 10 to 14)

Spec.	CPG_CLKON_10	CPG_CLKON_11	CPG_CLKON_12	CPG_CLKON_13	CPG_CLKON_14
Offset address	0628h	062Ch	0630h	0634h	0638h
Initial value	0000_0000h	0000_0000h	0000_0000h	0000_0000h	0000_0000h
CLK0_ON	CGC_SPI_ACLK	—	—	Reserved	Reserved
CLK1_ON	CGC_SPI_clk_spi / CGC_SPI_clk_spix2	Reserved	—	Reserved	Reserved
CLK2_ON	Reserved	Reserved	—	CGC_CRU_0_aclk	Reserved
CLK3_ON	CGC_SDHI_0_IMCLK	—	—	CGC_CRU_0_vclk	Reserved
CLK4_ON	CGC_SDHI_0_IMCLK2	—	CGC_PCIE_0_ACLK	CGC_CRU_0_pclk	Reserved
CLK5_ON	CGC_SDHI_0_clk_hs	—	CGC_PCIE_0_CLK_PMU	Reserved	Reserved
CLK6_ON	CGC_SDHI_0_ACLK	—	—	Reserved	Reserved
CLK7_ON	CGC_SDHI_1_IMCLK	—	—	Reserved	Reserved
CLK8_ON	CGC_SDHI_1_IMCLK2	—	—	Reserved	CGC_DSI_0_pclk
CLK9_ON	CGC_SDHI_1_clk_hs	—	—	Reserved	CGC_DSI_0_aclk
CLK10_ON	CGC_SDHI_1_ACLK	—	—	Reserved	CGC_DSI_0_vclk1
CLK11_ON	CGC_SDHI_2_IMCLK	—	—	Reserved	CGC_DSI_0_lpclk
CLK12_ON	CGC_SDHI_2_IMCLK2	—	Reserved	Reserved	CGC_DSI_0_PLLREFCLK
CLK13_ON	CGC_SDHI_2_clk_hs	—	Reserved	Reserved	CGC_LCDC_0_clk_a
CLK14_ON	CGC_SDHI_2_ACLK	—	Reserved	Reserved	CGC_LCDC_0_clk_p
CLK15_ON	—	—	Reserved	Reserved	CGC_LCDC_0_clk_d

Note: For more information, refer to the User's Manual Additional Document.

Table 4.4-16 Specifications of the CPG_CLKON_m Registers (m = 15 to 19)

Spec.	CPG_CLKON_15	CPG_CLKON_16	CPG_CLKON_17	CPG_CLKON_18	CPG_CLKON_19
Offset address	063Ch	0640h	0644h	0648h	064Ch
Initial value	0000_0000h	0000_E000h	0000_F000h	0000_0003h	0000_0000h
CLK0_ON	CGC_GE3D_CLK	CGC_SPDIF_2_clkp	Reserved	CGC_MCPU_AXI_CLK50CF_MCP	CGC_ACPU_Peri_Video1_AXI_CLK100DF*1
CLK1_ON	CGC_GE3D_AXI_CLK	CGC_PDM_0_PCLK	R.eserved	CGC_MCPU_AXI_CLK24	CGC_ACPU_Peri_NPU0_AXI_CLK400DG_DRP*1
CLK2_ON	CGC_GE3D_ACE_CLK	CGC_PDM_0_PCLK_SFR	Reserved	CGC_ACPU_AXI_CLK400DG_ACP*1	CGC_ACPU_Peri_NPU1_AXI_CLK400DG_DRP*1
CLK3_ON	CGC_VCD_0_ACLK	CGC_PDM_0_CCLK	Reserved	CGC_ACPU_AXI_CLK200DG_ACP*1	CGC_ACPU_Peri_COM0_AXI_CLK400DG_ACP*1
CLK4_ON	CGC_VCD_0_PCLK	CGC_PDM_1_PCLK	Reserved	CGC_ACPU_AXI_CLK400DG_DRP*1	CGC_ACPU_Peri_COM0_AXI_CLK200DG_ACP*1
CLK5_ON	CGC_SSIF_0_clk	CGC_PDM_1_PCLK_SFR	Reserved	CGC_ACPU_AXI_CLK200CG_MCP*1	CGC_ACPU_Peri_COM0_AXI_CLK200DF*1
CLK6_ON	CGC_SCU_0_clk	CGC_PDM_1_CCLK	Reserved	CGC_ACPU_AXI_CLK200DF*1	CGC_ACPU_Peri_COM1_AXI_CLK200DG_ACP*1
CLK7_ON	CGC_SCU_0_clkx2	CGC_ADC_0_PCLK	CGC_RCPU_AXI_CLK200DG_RCP*1	CGC_ACPU_AXI_CLK200CF*1	CGC_ACPU_Peri_COM1_AXI_CLK200DF*1
CLK8_ON	CGC_DMACpp_0_clk	CGC_ADC_0_ADCLK	CGC_RCPU_AXI_CLK400DG_ACP*1	CGC_ACPU_AXI_CLK100DG_ACP*1	CGC_ACPU_Peri_DDR_AXI_CLK100DG_ACP*1
CLK9_ON	CGC_ADG_0_clks1	Reserved	CGC_RCPU_AXI_CLK200CF*1	CGC_ACPU_AXI_CLK100DF*1	CGC_VIDEO0_AXI_CLK400DG_ACP*1
CLK10_ON	CGC_ADG_0_clk_195m	CGC_TSU_1_PCLK	CGC_RCPU_AXI_CLK100CF*1	CGC_ACPU_AXI_CLK100CF*1	Reserved
CLK11_ON	CGC_ADG_0_audio_clka	Reserved (0b fix)	CGC_RCPU_AXI_CLK50CF*1	CGC_ACPU_AXI_CLK24*1	Reserved
CLK12_ON	CGC_ADG_0_audio_clkb	Reserved (0b fix)	CGC_MCPU_AXI_CLK200CG_MCP	CGC_ACPU_Peri_Video0_AXI_CLK400DG_ACP*1	CGC_VIDEO1_AXI_CLK400DG_ACP*1
CLK13_ON	CGC_ADG_0_audio_clkc	Reserved (1b fix)	CGC_MCPU_AXI_CLK100CF_MCP	CGC_ACPU_Peri_Video0_AXI_CLK100DF*1	CGC_VIDEO1_AXI_CLK100DG_ACP*1
CLK14_ON	CGC_SPDIF_0_clkp	CGC_OTPC_0_PCLK	CGC_MCPU_AXI_CLK100CG_MCP	CGC_ACPU_Peri_Video1_AXI_CLK100DG_ACP*1	Reserved
CLK15_ON	CGC_SPDIF_1_clkp	CGC_OTPC_0_SCLK	CGC_MCPU_AXI_CLK50CG_ADC	CGC_ACPU_Peri_Video1_AXI_CLK200DF*1	CGC_VIDEO1_AXI_CLK400DF*1

Note: For more information, refer to the User's Manual Additional Document.

Note 1. The setting is to be forcibly modified by hardware.
The settings may differ depending on the boot mode.

Table 4.4-17 Specifications of the CPG_CLKON_m Registers (m = 20 to 24)

Spec.	CPG_CLKON_20	CPG_CLKON_21	CPG_CLKON_22	CPG_CLKON_23	CPG_CLKON_24
Offset address	0650h	0654h	0658h	065Ch	0660h
Initial value	0000_0000h	0000_0000h	0000_0000h	0000_0000h	0000_07FFh
CLK0_ON	Reserved	Reserved	Audio master clock output by ADG for SSI ch0*2	System clock supplied to the DVC ch0 module of SCU*2	System clock supplied to the SSIU modules other than the below*2
CLK1_ON	CGC_NPU_AXI_CLK400DG_DRP*1	Reserved	Audio master clock output by ADG for SSI ch1*2	System clock supplied to the DVC ch1 module of SCU*2	System clock supplied to the SSI ch0 module of SSIU*2
CLK2_ON	CGC_COM_AXI_CLK400DG_ACP*1	Reserved	Audio master clock output by ADG for SSI ch2*2	System clock supplied to the CTU/MIX ch0 module of SCU*2	System clock supplied to the SSI ch1 module of SSIU*2
CLK3_ON	CGC_COM_AXI_CLK200DG_ACP*1	Reserved	Audio master clock output by ADG for SSI ch3*2	System clock supplied to the CTU/MIX ch1 module of SCU*2	System clock supplied to the SSI ch2 module of SSIU*2
CLK4_ON	CGC_COM_AXI_CLK400DG_DRP	Reserved	Audio master clock output by ADG for SSI ch4*2	System clock supplied to the SRC ch0 module of SCU*2	System clock supplied to the SSI ch3 module of SSIU*2
CLK5_ON	CGC_COM_AXI_CLK200DF*1	Reserved	Audio master clock output by ADG for SSI ch5*2	System clock supplied to the SRC ch1 module of SCU*2	System clock supplied to the SSI ch4 module of SSIU*2
CLK6_ON	CGC_AXI_TZCDDR_0_CLK100DG_ACP_PCLK0*1	Reserved	Audio master clock output by ADG for SSI ch6*2	System clock supplied to the SRC ch2 module of SCU*2	System clock supplied to the SSI ch5 module of SSIU *2
CLK7_ON	CGC_AXI_TZCDDR_0_CLK100DG_ACP_PCLK1*1	Reserved	Audio master clock output by ADG for SSI ch7*2	System clock supplied to the SRC ch3 module of SCU*2	System clock supplied to the SSI ch6 module of SSIU*2
CLK8_ON	CGC_AXI_TZCDDR_0_CLK400DG_ACP_ACLK0*1	Reserved	Audio master clock output by ADG for SSI ch8*2	System clock supplied to the SRC ch4 module of SCU*2	System clock supplied to the SSI ch7 module of SSIU*2
CLK9_ON	CGC_AXI_TZCDDR_0_CLK400DG_ACP_ACLK1*1	Reserved	Audio master clock output by ADG for SSI ch9*2	System clock supplied to the SRC ch5 module of SCU*2	System clock supplied to the SSI ch8 module of SSIU*2
CLK10_ON	CGC_AXI_TZCDDR_0_CLK400DG_DRP_ACLK2*1	Reserved	Audio master clock output by ADG for SPDIF ch0*2	System clock supplied to the SRC ch6 module of SCU*2	System clock supplied to the SSI ch9 module of SSIU*2
CLK11_ON	CGC_AXI_TZCDDR_0_CLK400DG_ACP_ACLK3*1	Reserved	Audio master clock output by ADG for SPDIF ch1*2	System clock supplied to the SRC ch7 module of SCU*2	Reserved
CLK12_ON	CGC_AXI_TZCDDR_0_CLK400DG_ACP_ACLK4*1	Reserved	Audio master clock output by ADG for SPDIF ch2*2	System clock supplied to the SRC ch8 module of SCU*2	Reserved
CLK13_ON	Reserved	Reserved	Reserved	System clock supplied to the SRC ch9 module of SCU*2	Reserved
CLK14_ON	Reserved	Reserved	Reserved	System clock supplied to the SCU modules other than the above*2	Reserved
CLK15_ON	Reserved	Reserved	Reserved	Reserved	Reserved

Note 1. The setting is to be forcibly modified by hardware.
The settings may differ depending on the boot mode.

Note 2. For details, refer to **8.1 Audio Overview**.

Table 4.4-18 Specifications of the CPG_CLKON_m Registers (m = 25 to 27)

Spec.	CPG_CLKON_25	CPG_CLKON_26	CPG_CLKON_27
Offset address	0664h	0668h	066Ch
Initial value	0000_0000h	0000_0000h	0000_0000h
CLK0_ON	CGC_DSI_0_vclk2*1	CGC_LVDS_TOP_CLK_CH0	CGC_PCIE_CORECLKIN
CLK1_ON	Reserved	CGC_LVDS_TOP_CLK_CH1	CGC_FDP1_clk_m
CLK2_ON	Reserved	CGC_LVDS_TOP_CLK_DOT_CH0	CGC_FDP1_clk_a_as
CLK3_ON	Reserved	CGC_LVDS_TOP_CLK_DOT_CH1	CGC_FDP1_clk_p_as
CLK4_ON	Reserved	CGC_LVDS_TOP_PCLK	CGC_VIDEO0_AXI_CLK400DF
CLK5_ON	Reserved	CGC_VSPI_clk_m	CGC_ACPU_Peri_Video0_AXI_CLK200DF
CLK6_ON	Reserved	CGC_VSPI_clk_a_as	Reserved
CLK7_ON	Reserved	CGC_VSPI_clk_p_as	Reserved
CLK8_ON	Reserved	CGC_LCDC_1_clk_a	Reserved
CLK9_ON	Reserved	CGC_LCDC_1_clk_p	Reserved
CLK10_ON	Reserved	CGC_LCDC_1_clk_d	Reserved
CLK11_ON	Reserved	CGC_NPU_ACLK	Reserved
CLK12_ON	Reserved	CGC_NPU_PCLK	Reserved
CLK13_ON	Reserved	CGC_NPU_MCLK	Reserved
CLK14_ON	Reserved	Reserved	Reserved
CLK15_ON	Reserved	CGC_PCIE_REFCLK_IN	Reserved

Note 1. The setting is to be forcibly modified by hardware.
For hardware operation of booting, refer to **4.4.7 Control of Booting**.

4.4.4.9 Dynamic Gear/Mux Status Monitor Register (CPG_CLKSTATUS0)

This register monitors the states of the dynamic gears/multiplexors.

Access Size : 32 bits

Address : <CPG_base> + 0700h

Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	DDIV5CTL0_STS	DDIV4CTL3_STS	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	x	x	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDIV3CTL3_STS	DDIV3CTL2_STS	DDIV3CTL1_STS	-	-	-	-	DDIV2CTL0_STS	DDIV1CTL3_STS	DDIV1CTL2_STS	DDIV1CTL1_STS	DDIV1CTL0_STS	DDIV0CTL3_STS	DDIV0CTL2_STS	DDIV0CTL1_STS	DDIV0CTL0_STS
Initial Value	x	x	x	0	0	0	0	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
20	DDIV5CTL0_STS	x	R	CDDIV_2to64_PLLDTY_NPU State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).
19	DDIV4CTL3_STS	x	R	CDDIV_1to4_PLLETH_NPU State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).
18 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
15	DDIV3CTL3_STS	x	R	CDDIV_2to4_PLLVDO_CRU0 State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).
14	DDIV3CTL2_STS	x	R	CDDIV_2to64_PLLDTY_RCPU State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).
13	DDIV3CTL1_STS	x	R	CDDIV_2to64_PLLVDO_GPU State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).
12 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
8	DDIV2CTL0_STS	x	R	CDDIV_2to16_CA55PERI State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).
7	DDIV1CTL3_STS	x	R	CDDIV_1to8_CLK1800_PLLCA55_CA55_3 State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).
6	DDIV1CTL2_STS	x	R	CDDIV_1to8_CLK1800_PLLCA55_CA55_2 State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).

Bit	Bit Name	Initial Value	R/W	Description
5	DDIV1CTL1_STS	x	R	CDDIV_1to8_CLK1800_PLLCA55_CA55_1 State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).
4	DDIV1CTL0_STS	x	R	CDDIV_1to8_CLK1800_PLLCA55_CA55_0 State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).
3	DDIV0CTL3_STS	x	R	CDDIV_1to8_PLLCA55_SCLK State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).
2	DDIV0CTL2_STS	x	R	CDDIV_2to64_PLLDTY_ACPU State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).
1	DDIV0CTL1_STS	x	R	CDDIV_2to64_PLLCM33 State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).
0	DDIV0CTL0_STS	x	R	CDDIV_2to64_PLLCM33_CST400 State monitor for completion of switching 0b: Switching has been completed. 1b: Switching has not been completed (in the busy state).

x: Undefined value

4.4.4.10 CGC Monitor Registers (CPG_CLKMON_m) (m = 0 to 11)

These registers monitor the states of the individual clock signals in terms of whether they are on or off. For details on the registers, refer to **Table 4.4-19** to **Table 4.4-22** and block diagrams of the clock system.

Remark The initial values of the CPG_CLKON_m registers are reflected in these registers. When the initial value of a CPG_CLKON_m register bit is 1 (the clock signal is on and supplied), the initial value of the corresponding CPG_CLKMON_m register bit is also 1 (the clock signal is on).

Access Size : 32 bits

Address : <CPG_base> + 0800h + m x 0004h

Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CLK31_MON	CLK30_MON	CLK29_MON	CLK28_MON	CLK27_MON	CLK26_MON	CLK25_MON	CLK24_MON	CLK23_MON	CLK22_MON	CLK21_MON	CLK20_MON	CLK19_MON	CLK18_MON	CLK17_MON	CLK16_MON
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLK15_MON	CLK14_MON	CLK13_MON	CLK12_MON	CLK11_MON	CLK10_MON	CLK9_MON	CLK8_MON	CLK7_MON	CLK6_MON	CLK5_MON	CLK4_MON	CLK3_MON	CLK2_MON	CLK1_MON	CLK0_MON
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	CLK31_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
30	CLK30_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
29	CLK29_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
28	CLK28_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
27	CLK27_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
26	CLK26_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
25	CLK25_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
24	CLK24_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
23	CLK23_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
22	CLK22_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
21	CLK21_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On

Bit	Bit Name	Initial Value	R/W	Description
20	CLK20_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
19	CLK19_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
18	CLK18_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
17	CLK17_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
16	CLK16_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
15	CLK15_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
14	CLK14_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
13	CLK13_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
12	CLK12_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
11	CLK11_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
10	CLK10_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
9	CLK9_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
8	CLK8_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
7	CLK7_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
6	CLK6_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
5	CLK5_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
4	CLK4_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
3	CLK3_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
2	CLK2_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
1	CLK1_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On
0	CLK0_MON	x	R	Clock monitor. For the target signal, refer to **Table 4.4-19** to **Table 4.4-22**. 0b: Off 1b: On

x: Undefined value

Table 4.4-19 Specifications of the CPG_CLKMON_m Registers (m = 0 to 3)

Spec.	CPG_CLKMON_0	CPG_CLKMON_1	CPG_CLKMON_2	CPG_CLKMON_3
Offset address	0800h	0804h	0808h	080Ch
CLK0_MON	CGC_DMAC_0_ACLK	CGC_BTROM_0_ACLK	CGC_RCPU_CMTW_1_clkm	CGC_RSCI_0_ps_ps2_n
CLK1_MON	CGC_DMAC_1_ACLK	CGC_CST_0_cs_clk	CGC_RCPU_CMTW_2_clkm	CGC_RSCI_0_ps_ps1_n
CLK2_MON	CGC_DMAC_2_ACLK	CGC_CST_0_ts_clk	CGC_RCPU_CMTW_3_clkm	CGC_RSCI_1_PCLK
CLK3_MON	CGC_DMAC_3_ACLK	CGC_CST_0_apb_sb_clk	CGC_MCPU_GTM_0_PCLK	CGC_RSCI_1_TCLK
CLK4_MON	CGC_DMAC_4_ACLK	CGC_CST_0_apb_ca55_clk	CGC_MCPU_GTM_1_PCLK	CGC_RSCI_1_ps_ps3_n
CLK5_MON	CGC_ICU_0_PCLK_I	Reserved	CGC_ACPU_GTM_0_PCLK	CGC_RSCI_1_ps_ps2_n
CLK6_MON	CGC_CRC_0_clk_crc	CGC_CST_0_apb_cm33_clk	CGC_ACPU_GTM_1_PCLK	CGC_RSCI_1_ps_ps1_n
CLK7_MON	CGC_CA55_0_SCLK	CGC_CST_0_ahb_cm33_clk	CGC_RCPU_GTM_0_PCLK	CGC_RSCI_2_PCLK
CLK8_MON	CGC_CA55_0_PCLK	CGC_CST_0_ahb_ath_clk	CGC_RCPU_GTM_1_PCLK	CGC_RSCI_2_TCLK
CLK9_MON	CGC_CA55_0_ATCLK	CGC_CST_0_atb_ca55_clk	CGC_RCPU_GTM_2_PCLK	CGC_RSCI_2_ps_ps3_n
CLK10_MON	CGC_CA55_0_GICCLK	Reserved	CGC_RCPU_GTM_3_PCLK	CGC_RSCI_2_ps_ps2_n
CLK11_MON	CGC_CA55_0_ACLK	CGC_CST_0_atb_cm33_clk	CGC_WDT_0_clkp	CGC_RSCI_2_ps_ps1_n
CLK12_MON	CGC_CA55_0_TSCLK	CGC_CST_0_atb_cst_clk	CGC_WDT_0_clk_loco	CGC_RSCI_3_PCLK
CLK13_MON	Reserved	CGC_CST_0_axi_etr_clk	CGC_WDT_1_clkp	CGC_RSCI_3_TCLK
CLK14_MON	Reserved	CGC_CST_0_axi_sb_clk	CGC_WDT_1_clk_loco	CGC_RSCI_3_ps_ps3_n
CLK15_MON	Reserved	CGC_SYC_0_CNT_CLK	CGC_WDT_2_clkp	CGC_RSCI_3_ps_ps2_n
CLK16_MON	Reserved	CGC_MHU_0_pclk	CGC_WDT_2_clk_loco	CGC_RSCI_3_ps_ps1_n
CLK17_MON	Reserved	CGC_GPT_0_pclk_sfr	CGC_WDT_3_clkp	CGC_RSCI_4_PCLK
CLK18_MON	CGC_CM33_CLK1	CGC_GPT_1_pclk_sfr	CGC_WDT_3_clk_loco	CGC_RSCI_4_TCLK
CLK19_MON	CGC_GIC_0_gicclk	CGC_POEGA_0_PCLK	CGC_RTC_0_clk_rtc	CGC_RSCI_4_ps_ps3_n
CLK20_MON	Reserved	CGC_POEGB_0_PCLK	CGC_RSPI_0_PCLK	CGC_RSCI_4_ps_ps2_n
CLK21_MON	CGC_SRAM_0_ACLK	CGC_POEGC_0_PCLK	CGC_RSPI_0_pclk_sfr	CGC_RSCI_4_ps_ps1_n
CLK22_MON	Reserved	CGC_POEGD_0_PCLK	CGC_RSPI_0_TCLK	CGC_RSCI_5_PCLK
CLK23_MON	Reserved	CGC_POEGA_1_PCLK	CGC_RSPI_1_PCLK	CGC_RSCI_5_TCLK
CLK24_MON	CGC_SRAM_1_ACLK	CGC_POEGB_1_PCLK	CGC_RSPI_1_pclk_sfr	CGC_RSCI_5_ps_ps3_n
CLK25_MON	Reserved	CGC_POEGC_1_PCLK	CGC_RSPI_1_TCLK	CGC_RSCI_5_ps_ps2_n
CLK26_MON	Reserved	CGC_POEGD_1_PCLK	CGC_RSPI_2_PCLK	CGC_RSCI_5_ps_ps1_n
CLK27_MON	Reserved	CGC_MCPU_CMTW_0_clkm	CGC_RSPI_2_pclk_sfr	CGC_RSCI_6_PCLK
CLK28_MON	Reserved	CGC_MCPU_CMTW_1_clkm	CGC_RSPI_2_TCLK	CGC_RSCI_6_TCLK
CLK29_MON	Reserved	CGC_MCPU_CMTW_2_clkm	CGC_RSCI_0_PCLK	CGC_RSCI_6_ps_ps3_n
CLK30_MON	Reserved	CGC_MCPU_CMTW_3_clkm	CGC_RSCI_0_TCLK	CGC_RSCI_6_ps_ps2_n
CLK31_MON	Reserved	CGC_RCPU_CMTW_1_clkm	CGC_RSCI_0_ps_ps3_n	CGC_RSCI_6_ps_ps1_n

Table 4.4-20 Specifications of the CPG_CLKMON_m Registers (m = 4 to 7)

Spec.	CPG_CLKMON_4	CPG_CLKMON_5	CPG_CLKMON_6	CPG_CLKMON_7
Offset address	0810h	0814h	0818h	081Ch
CLK0_MON	CGC_RSCI_7_PCLK	CGC_SPI_ACLK	—	Reserved
CLK1_MON	CGC_RSCI_7_TCLK	CGC_SPI_clk_spi	—	Reserved
CLK2_MON	CGC_RSCI_7_ps_ps3_n	CGC_SPI_clk_spix2	—	Reserved
CLK3_MON	CGC_RSCI_7_ps_ps2_n	CGC_SDHI_0_IMCLK	—	Reserved
CLK4_MON	CGC_RSCI_7_ps_ps1_n	CGC_SDHI_0_IMCLK2	CGC_PCIE_0_ACLK	Reserved
CLK5_MON	CGC_RSCI_8_PCLK	CGC_SDHI_0_clk_hs	CGC_PCIE_0_CLK_PMU	Reserved
CLK6_MON	CGC_RSCI_8_TCLK	CGC_SDHI_0_ACLK	—	Reserved
CLK7_MON	CGC_RSCI_8_ps_ps3_n	CGC_SDHI_1_IMCLK	—	Reserved
CLK8_MON	CGC_RSCI_8_ps_ps2_n	CGC_SDHI_1_IMCLK2	—	CGC_DSI_0_pclk
CLK9_MON	CGC_RSCI_8_ps_ps1_n	CGC_SDHI_1_clk_hs	—	CGC_DSI_0_aclk
CLK10_MON	CGC_RSCI_9_PCLK	CGC_SDHI_1_ACLK	—	CGC_DSI_0_vclk1
CLK11_MON	CGC_RSCI_9_TCLK	CGC_SDHI_2_IMCLK	—	CGC_DSI_0_lpclk
CLK12_MON	CGC_RSCI_9_ps_ps3_n	CGC_SDHI_2_IMCLK2	Reserved	CGC_DSI_0_PLLREFCLK
CLK13_MON	CGC_RSCI_9_ps_ps2_n	CGC_SDHI_2_clk_hs	Reserved	CGC_LCDC_0_clk_a
CLK14_MON	CGC_RSCI_9_ps_ps1_n	CGC_SDHI_2_ACLK	Reserved	CGC_LCDC_0_clk_p
CLK15_MON	CGC_SCIF_0_clk_pck	—	Reserved	CGC_LCDC_0_clk_d
CLK16_MON	CGC_I3C_0_PCLKRW	—	Reserved	CGC_GE3D_CLK
CLK17_MON	CGC_I3C_0_PCLK	Reserved	Reserved	CGC_GE3D_AXI_CLK
CLK18_MON	CGC_I3C_0_TCLK	Reserved	CGC_CRU_0_aclk	CGC_GE3D_ACE_CLK
CLK19_MON	CGC_RIIC_8_ckm	—	CGC_CRU_0_vclk	CGC_VCD_0_ACLK
CLK20_MON	CGC_RIIC_0_ckm	—	CGC_CRU_0_pclk	CGC_VCD_0_PCLK
CLK21_MON	CGC_RIIC_1_ckm	—	Reserved	CGC_SSIF_0_clk
CLK22_MON	CGC_RIIC_2_ckm	—	Reserved	CGC_SCU_0_clk
CLK23_MON	CGC_RIIC_3_ckm	—	Reserved	CGC_SCU_0_clkx2
CLK24_MON	CGC_RIIC_4_ckm	—	Reserved	CGC_DMACpp_0_clk
CLK25_MON	CGC_RIIC_5_ckm	—	Reserved	CGC_ADG_0_clks1
CLK26_MON	CGC_RIIC_6_ckm	—	Reserved	CGC_ADG_0_clk_195m
CLK27_MON	CGC_RIIC_7_ckm	—	Reserved	CGC_ADG_0_audio_clka
CLK28_MON	CGC_CANFD_0_pclk	—	Reserved	CGC_ADG_0_audio_clkb
CLK29_MON	CGC_CANFD_0_clk_ram	—	Reserved	CGC_ADG_0_audio_clkc
CLK30_MON	CGC_CANFD_0_clkc	—	Reserved	CGC_SPDIF_0_clkp
CLK31_MON	CGC_SPI_HCLK	—	Reserved	CGC_SPDIF_1_clkp

Note: For more information, refer to the User's Manual Additional Document.

Table 4.4-21 Specifications of the CPG_CLKMON_m Registers (m = 8 to 10)

Spec.	CPG_CLKMON_8	CPG_CLKMON_9	CPG_CLKMON_10
Offset address	0820h	0824h	0828h
CLK0_MON	CGC_SPDIF_2_clkp	CGC_MCPU_AXI_CLK50CF_MCP	Reserved
CLK1_MON	CGC_PDM_0_PCLK	CGC_MCPU_AXI_CLK24	CGC_NPU_AXI_CLK400DG_DRP
CLK2_MON	CGC_PDM_0_PCLK_SFR	CGC_ACPU_AXI_CLK400DG_ACP	CGC_COM_AXI_CLK400DG_ACP
CLK3_MON	CGC_PDM_0_CCLK	CGC_ACPU_AXI_CLK200DG_ACP	CGC_COM_AXI_CLK200DG_ACP
CLK4_MON	CGC_PDM_1_PCLK	CGC_ACPU_AXI_CLK400DG_DRP	CGC_COM_AXI_CLK400DG_DRP
CLK5_MON	CGC_PDM_1_PCLK_SFR	CGC_ACPU_AXI_CLK200CG_MCP	CGC_COM_AXI_CLK200DF
CLK6_MON	CGC_PDM_1_CCLK	CGC_ACPU_AXI_CLK200DF	CGC_AXI_TZCDDR_0_CLK100DG_ACP_PCLK0
CLK7_MON	CGC_ADC_0_PCLK	CGC_ACPU_AXI_CLK200CF	CGC_AXI_TZCDDR_0_CLK100DG_ACP_PCLK1
CLK8_MON	CGC_ADC_0_ADCLK	CGC_ACPU_AXI_CLK100DG_ACP	CGC_AXI_TZCDDR_0_CLK400DG_ACP_ACLK0
CLK9_MON	Reserved	CGC_ACPU_AXI_CLK100DF	CGC_AXI_TZCDDR_0_CLK400DG_ACP_ACLK1
CLK10_MON	CGC_TSU_1_PCLK	CGC_ACPU_AXI_CLK100CF	CGC_AXI_TZCDDR_0_CLK400DG_DRP_ACLK2
CLK11_MON	Reserved	CGC_ACPU_AXI_CLK24	CGC_AXI_TZCDDR_0_CLK400DG_ACP_ACLK3
CLK12_MON	Reserved	CGC_ACPU_Peri_Video0_AXI_CLK400DG_ACP	CGC_AXI_TZCDDR_0_CLK400DG_ACP_ACLK4
CLK13_MON	Reserved	CGC_ACPU_Peri_Video0_AXI_CLK100DF	Reserved
CLK14_MON	CGC_OTPC_0_PCLK	CGC_ACPU_Peri_Video1_AXI_CLK100DG_ACP	Reserved
CLK15_MON	CGC_OTPC_0_SCLK	CGC_ACPU_Peri_Video1_AXI_CLK200DF	Reserved
CLK16_MON	Reserved	CGC_ACPU_Peri_Video1_AXI_CLK100DF	Reserved
CLK17_MON	Reserved	CGC_ACPU_Peri_NPU0_AXI_CLK400DG_DRP	Reserved
CLK18_MON	Reserved	CGC_ACPU_Peri_NPU1_AXI_CLK400DG_DRP	Reserved
CLK19_MON	Reserved	CGC_ACPU_Peri_COM0_AXI_CLK400DG_ACP	Reserved
CLK20_MON	Reserved	CGC_ACPU_Peri_COM0_AXI_CLK200DG_ACP	Reserved
CLK21_MON	Reserved	CGC_ACPU_Peri_COM0_AXI_CLK200DF	CGC_DSI_0_vclk2
CLK22_MON	Reserved	CGC_ACPU_Peri_COM1_AXI_CLK200DG_ACP	CGC_LVDS_TOP_CLK_CH0
CLK23_MON	CGC_RCPU_AXI_CLK200DG_RCP	CGC_ACPU_Peri_COM1_AXI_CLK200DF	CGC_LVDS_TOP_CLK_CH1
CLK24_MON	CGC_RCPU_AXI_CLK400DG_ACP	CGC_ACPU_Peri_DDR_AXI_CLK100DG_ACP	CGC_LVDS_TOP_CLK_DOT_CH0
CLK25_MON	CGC_RCPU_AXI_CLK200CF	CGC_VIDEO0_AXI_CLK400DG_ACP	CGC_LVDS_TOP_CLK_DOT_CH1
CLK26_MON	CGC_RCPU_AXI_CLK100CF	Reserved	CGC_LVDS_TOP_PCLK
CLK27_MON	CGC_RCPU_AXI_CLK50CF	Reserved	CGC_VSPI_clk_m
CLK28_MON	CGC_MCPU_AXI_CLK200CG_MCP	CGC_VIDEO1_AXI_CLK400DG_ACP	CGC_VSPI_clk_a_as
CLK29_MON	CGC_MCPU_AXI_CLK100CF_MCP	CGC_VIDEO1_AXI_CLK100DG_ACP	CGC_VSPI_clk_p_as
CLK30_MON	CGC_MCPU_AXI_CLK100CG_MCP	Reserved	CGC_LCDC_1_clk_a
CLK31_MON	CGC_MCPU_AXI_CLK50CG_ADC	CGC_VIDEO1_AXI_CLK400DF	CGC_LCDC_1_clk_p

Table 4.4-22 Specifications of the CPG_CLKMON_m Registers (m = 11)

Spec.	CPG_CLKMON_11
Offset address	082Ch
CLK0_MON	CGC_LCDC_1_clk_d
CLK1_MON	CGC_NPU_ACLK
CLK2_MON	CGC_NPU_PCLK
CLK3_MON	CGC_NPU_MCLK
CLK4_MON	CGC_SHIPG_clkb
CLK5_MON	CGC_PCIE_REFCLK_IN
CLK6_MON	CGC_PCIE_CORECLKIN
CLK7_MON	CGC_FDP1_clk_m
CLK8_MON	CGC_FDP1_clk_a_as
CLK9_MON	CGC_FDP1_clk_p_as
CLK10_MON	CGC_VIDEO0_AXI_CLK400DF
CLK11_MON	CGC_ACPU_Peri_Video0_AXI_CLK200DF
CLK12_MON	Reserved
CLK13_MON	Reserved
CLK14_MON	Reserved
CLK15_MON	Reserved
CLK16_MON	Reserved
CLK17_MON	Reserved
CLK18_MON	Reserved
CLK19_MON	Reserved
CLK20_MON	Reserved
CLK21_MON	Reserved
CLK22_MON	Reserved
CLK23_MON	Reserved
CLK24_MON	Reserved
CLK25_MON	Reserved
CLK26_MON	Reserved
CLK27_MON	Reserved
CLK28_MON	Reserved
CLK29_MON	Reserved
CLK30_MON	Reserved
CLK31_MON	Reserved

4.4.4.11 Reset Control Registers (CPG_RST_m) (m = 0 to 18)

These registers control resetting the individual units. (0b: Reset on, 1b: Reset off)

For details on the registers, refer to **Table 4.4-23** to **Table 4.4-26**.

Access Size : 32 bits

Address : <CPG_base> + 0900h + m x 0004h

Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSTB15_WEN	RSTB14_WEN	RSTB13_WEN	RSTB12_WEN	RSTB11_WEN	RSTB10_WEN	RSTB9_WEN	RSTB8_WEN	RSTB7_WEN	RSTB6_WEN	RSTB5_WEN	RSTB4_WEN	RSTB3_WEN	RSTB2_WEN	RSTB1_WEN	RSTB0_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSTB15	RSTB14	RSTB13	RSTB12	RSTB11	RSTB10	RSTB9	RSTB8	RSTB7	RSTB6	RSTB5	RSTB4	RSTB3	RSTB2	RSTB1	RSTB0
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	RSTB15_WEN	0h	W	Write enable for RSTB15
30	RSTB14_WEN	0h	W	Write enable for RSTB14
29	RSTB13_WEN	0h	W	Write enable for RSTB13
28	RSTB12_WEN	0h	W	Write enable for RSTB12
27	RSTB11_WEN	0h	W	Write enable for RSTB11
26	RSTB10_WEN	0h	W	Write enable for RSTB10
25	RSTB9_WEN	0h	W	Write enable for RSTB9
24	RSTB8_WEN	0h	W	Write enable for RSTB8
23	RSTB7_WEN	0h	W	Write enable for RSTB7
22	RSTB6_WEN	0h	W	Write enable for RSTB6
21	RSTB5_WEN	0h	W	Write enable for RSTB5
20	RSTB4_WEN	0h	W	Write enable for RSTB4
19	RSTB3_WEN	0h	W	Write enable for RSTB3
18	RSTB2_WEN	0h	W	Write enable for RSTB2
17	RSTB1_WEN	0h	W	Write enable for RSTB1
16	RSTB0_WEN	0h	W	Write enable for RSTB0
15	RSTB15	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
14	RSTB14	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
13	RSTB13	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
12	RSTB12	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
11	RSTB11	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
10	RSTB10	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
9	RSTB9	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off

Bit	Bit Name	Initial Value	R/W	Description
8	RSTB8	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
7	RSTB7	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
6	RSTB6	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
5	RSTB5	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
4	RSTB4	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
3	RSTB3	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
2	RSTB2	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
1	RSTB1	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off
0	RSTB0	x	RW	For the target signal, refer to **Table 4.4-23** to **Table 4.4-26**. 0b: On 1b: Off

x: Undefined value

Note: Regarding the initial values, refer to the table.

Table 4.4-23 Specifications of the CPG_RST_m Registers (m = 0 to 4)

Spec.	CPG_RST_0	CPG_RST_1	CPG_RST_2	CPG_RST_3	CPG_RST_4
Offset address	0900h	0904h	0908h	090Ch	0910h
Initial value	0000_0000h	0000_0038h	0000_0000h	0000_1843h	0000_FFC0h
RSTB0	CA55_RESET0 (B group)*1	CA55_RESET14 (A group)*1	Reserved	SYS_0_PRESETN	Reserved
RSTB1	CA55_RESET1 (B group)*1	CA55_RESET15 (A group)*1	Reserved	DMAC_0_ARESETN*1	Reserved
RSTB2	CA55_RESET2 (B group)*1	CA55_RESET16 (A group)*1	Reserved	DMAC_1_ARESETN	Reserved
RSTB3	CA55_RESET3 (B group)*1	CM33_RESET2 (A group)*1	Reserved	DMAC_2_ARESETN	Reserved
RSTB4	CA55_RESET4 (B group)*1	CM33_RESET0 (B group)*1	Reserved	DMAC_3_ARESETN	Reserved
RSTB5	CA55_RESET5 (B group)*1	CM33_RESET1 (B group)*1	Reserved	DMAC_4_ARESETN	Reserved
RSTB6	CA55_RESET6 (B group)*1	Reserved	Reserved	ICU_0_PRESETN_I	BTROM_0_ARESETN
RSTB7	CA55_RESET7 (B group)*1	Reserved	Reserved	CRC_0_RST	CST_0_CS_RESETN
RSTB8	CA55_RESET8 (B group)*1	Reserved	Reserved	GIC_0_GICRESET_N*1	CST_0_TS_RESETN
RSTB9	CA55_RESET9 (B group)*1	Reserved	Reserved	GIC_0_DBG_GICRESET_N*1	CST_0_APB_SB_RESETN
RSTB10	CA55_RESET10 (B group)*1	Reserved	Reserved	Reserved	CST_0_APB_CA55_RESETN
RSTB11	CA55_RESET11 (B group)*1	Reserved	Reserved	SRAM_0_ARESETN	CST_0_APB_CM33_RESETN
RSTB12	CA55_RESET12 (B group)*1	Reserved	Reserved	Reserved	Reserved
RSTB13	CA55_RESET13 (B group)*1	Reserved	Reserved	Reserved	CST_0_AHB_CM33_RESETN
RSTB14	Reserved	Reserved	Reserved	SRAM_1_ARESETN*1	CST_0_AHB_ATH_RESETN
RSTB15	Reserved	Reserved	Reserved	Reserved	CST_0_ATB_CA55_RESETN

Note 1. The setting is to be forcibly modified by hardware.
The settings may differ depending on the boot mode.

Table 4.4-24 Specifications of the CPG_RST_m Registers (m = 5 to 9)

Spec.	CPG_RST_5	CPG_RST_6	CPG_RST_7	CPG_RST_8	CPG_RST_9
Offset address	0914h	0918h	091Ch	0920h	0924h
Initial value	0000_007Fh	0000_0000h	0000_0020h	0000_0000h	0000_0000h
RSTB0	CST_0_ATB_CM33_RESETN	POEGD_0_RST	ACPU_GTM_1_PRESETZ	RSPI_2_TRESETN	RSCI_7_TRESETN
RSTB1	Reserved	POEGA_1_RST	RCPU_GTM_0_PRESETZ	RSCI_0_PRESETN	RSCI_8_PRESETN
RSTB2	CST_0_ATB_CST_RESETN	POEGB_1_RST	RCPU_GTM_1_PRESETZ	RSCI_0_TRESETN	RSCI_8_TRESETN
RSTB3	CST_0_AXI_SB_RESETN	POEGC_1_RST	RCPU_GTM_2_PRESETZ	RSCI_1_PRESETN	RSCI_9_PRESETN
RSTB4	CST_0_AXI_ETR_RESETN	POEGD_1_RST	RCPU_GTM_3_PRESETZ	RSCI_1_TRESETN	RSCI_9_TRESETN
RSTB5	CST_0_NTRST	MCPU_CMTW_0_RST_M	WDT_0_RESET*1	RSCI_2_PRESETN	SCIF_0_RST_SYSTEM_N
RSTB6	CST_0_NPOTRST	MCPU_CMTW_1_RST_M	WDT_1_RESET	RSCI_2_TRESETN	I3C_0_PRESETN
RSTB7	SYC_0_RESETN*1	MCPU_CMTW_2_RST_M	WDT_2_RESET	RSCI_3_PRESETN	I3C_0_TRESETN
RSTB8	MHU_0_PRESETN	MCPU_CMTW_3_RST_M	WDT_3_RESET	RSCI_3_TRESETN	RIIC_0_MRST
RSTB9	GPT_0_RST_P_REG	RCPU_CMTW_0_RST_M	RTC_0_RST_RTC	RSCI_4_PRESETN	RIIC_1_MRST
RSTB10	GPT_0_RST_S_REG	RCPU_CMTW_1_RST_M	RTC_0_RST_RTC_V	RSCI_4_TRESETN	RIIC_2_MRST
RSTB11	GPT_1_RST_P_REG	RCPU_CMTW_2_RST_M	RSPI_0_PRESETN	RSCI_5_PRESETN	RIIC_3_MRST
RSTB12	GPT_1_RST_S_REG	RCPU_CMTW_3_RST_M	RSPI_0_TRESETN	RSCI_5_TRESETN	RIIC_4_MRST
RSTB13	POEGA_0_RST	MCPU_GTM_0_PRESETZ	RSPI_1_PRESETN	RSCI_6_PRESETN	RIIC_5_MRST
RSTB14	POEGB_0_RST	MCPU_GTM_1_PRESETZ	RSPI_1_TRESETN	RSCI_6_TRESETN	RIIC_6_MRST
RSTB15	POEGC_0_RST	ACPU_GTM_0_PRESETZ	RSPI_2_PRESETN	RSCI_7_PRESETN	RIIC_7_MRST

Note 1. The setting is to be forcibly modified by hardware.
The settings may differ depending on the boot mode.

Table 4.4-25 Specifications of the CPG_RST_m Registers (m = 10 to 14)

Spec.	CPG_RST_10	CPG_RST_11	CPG_RST_12	CPG_RST_13	CPG_RST_14
Offset address	0928h	092Ch	0930h	0934h	0938h
Initial value	0000_0060h	0000_0000h	0000_0000h	0000_0000h	0000_0000h
RSTB0	RIIC_8_MRST	—	Reserved	Reserved	VCD_0_RESETN
RSTB1	CANFD_0_RSTP_N	—	Reserved	Reserved	SSIF_0_ASYNC_RESET_SSI
RSTB2	CANFD_0_RSTC_N	PCIE_0_ARESETN	Reserved	Reserved	SSIF_0_SYNC_RESET_SSI0
RSTB3	SPI_HRESETN	—	Reserved	Reserved	SSIF_0_SYNC_RESET_SSI1
RSTB4	SPI_ARESETN	—	Reserved	Reserved	SSIF_0_SYNC_RESET_SSI2
RSTB5	IOTOP_0_RESETN	—	CRU_0_PRESETN	Reserved	SSIF_0_SYNC_RESET_SSI3
RSTB6	IOTOP_0_ERROR_RESETN	—	CRU_0_ARESETN	Reserved	SSIF_0_SYNC_RESET_SSI4
RSTB7	SDHI_0_IXRST	—	CRU_0_S_RESETN	DSI_0_PRESETN	SSIF_0_SYNC_RESET_SSI5
RSTB8	SDHI_1_IXRST	—	Reserved	DSI_0_ARESETN	SSIF_0_SYNC_RESET_SSI6
RSTB9	SDHI_2_IXRST	—	Reserved	Reserved	SSIF_0_SYNC_RESET_SSI7
RSTB10	—	—	Reserved	Reserved	SSIF_0_SYNC_RESET_SSI8
RSTB11	Reserved	—	Reserved	Reserved	SSIF_0_SYNC_RESET_SSI9
RSTB12	—	Reserved	Reserved	LCDC_0_RESET_N	SCU_0_RESET_SRU
RSTB13	—	Reserved	Reserved	GE3D_RESETN	DMACpp_0_ARST
RSTB14	—	Reserved	Reserved	GE3D_AXI_RESETN	ADG_0_RST_RESET_ADG
RSTB15	—	Reserved	Reserved	GE3D_ACE_RESETN	SPDIF_0_RST

Note: For more information, refer to the User's Manual Additional Document.

Table 4.4-26 Specifications of the CPG_RST_m Registers (m = 15 to 17)

Spec.	CPG_RST_15	CPG_RST_16	CPG_RST_17	CPG_RST_18
Offset address	093Ch	0940h	0944h	0948h
Initial value	0000_8C00h	0000_0000h	0000_0000h	0000_0000h
RSTB0	SPDIF_1_RST	ACPU_AXI_RESETN*1	AXI_TZCDDR_0_ARESET2N*1	NPU_nRESET
RSTB1	SPDIF_2_RST	ACPU_Peri_VIDEO0_AXI_RESETN*1	AXI_TZCDDR_0_ARESET3N*1	NPU_nARESET
RSTB2	PDM_0_PRESETN	ACPU_Peri_VIDEO1_AXI_RESETN*1	AXI_TZCDDR_0_ARESET4N*1	NPU_nPRESET
RSTB3	PDM_0_CRESET	ACPU_Peri_NPU0_AXI_RESETN*1	Reserved	SHIPG_rst_n
RSTB4	PDM_1_PRESETN	ACPU_Peri_NPU1_AXI_RESETN*1	Reserved	FDP1_reset_brdg_n
RSTB5	PDM_1_CRESETN	ACPU_Peri_COM0_AXI_RESETN*1	Reserved	FDP1_reset_fdp1_p
RSTB6	ADC_0_ADRST_N	ACPU_Peri_COM1_AXI_RESETN*1	Reserved	FDP1_reset_fcpf_p
RSTB7	Reserved	ACPU_Peri_DDR_AXI_RESETN*1	Reserved	Reserved
RSTB8	TSU_1_PRESETN	VIDEO0_AXI_RESETN*1	Reserved	Reserved
RSTB9	Reserved (0b fix)	VIDEO1_AXI_RESETN*1	Reserved	Reserved
RSTB10	Reserved (1b fix)	NPU_AXI_RESETN*1	LVDS_TOP_RESET_N	Reserved
RSTB11	OTPC_0_RESET_N	COM_AXI_RESETN*1	VSPI_x_smreset_vspiss_n	Reserved
RSTB12	Reserved	AXI_TZCDDR_0_PRESET0N*1	VSPI_x_sreset_vsp1z	Reserved
RSTB13	Reserved	AXI_TZCDDR_0_PRESET1N*1	VSPI_x_spreset_fcpvi0	Reserved
RSTB14	RCPU_AXI_RESETN*1	AXI_TZCDDR_0_ARESET0N*1	LCDC_1_RESET_N	Reserved
RSTB15	MCPU_AXI_RESETN	AXI_TZCDDR_0_ARESET1N*1	Reserved	Reserved

Note: For more information, refer to the User's Manual Additional Document.

Note 1. The setting is to be forcibly modified by hardware.
The settings may differ depending on the boot mode.

4.4.4.12 Reset Monitor Registers (CPG_RSTMON_m) (m = 0 to 9)

These registers monitor the individual reset signals. For details on the registers, refer to **Table 4.4-27** to **Table 4.4-30**.

Remark The initial values of the CPG_RST_m registers are reflected in these registers. When the initial value of a CPG_RST_m register bit is 1 (the reset is on and the target unit is in the reset state), the initial value of the corresponding CPG_RSTMON_m register bit is also 1 (the target unit is in the reset state).

Access Size : 32 bits

Address : <CPG_base> + 0A00h + m x 0004h

Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RST31_MON	RST30_MON	RST29_MON	RST28_MON	RST27_MON	RST26_MON	RST25_MON	RST24_MON	RST23_MON	RST22_MON	RST21_MON	RST20_MON	RST19_MON	RST18_MON	RST17_MON	RST16_MON
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RST15_MON	RST14_MON	RST13_MON	RST12_MON	RST11_MON	RST10_MON	RST9_MON	RST8_MON	RST7_MON	RST6_MON	RST5_MON	RST4_MON	RST3_MON	RST2_MON	RST1_MON	RST0_MON
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RST31_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
30	RST30_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
29	RST29_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
28	RST28_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
27	RST27_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
26	RST26_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
25	RST25_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
24	RST24_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
23	RST23_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
22	RST22_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
21	RST21_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
20	RST20_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.

Bit	Bit Name	Initial Value	R/W	Description
19	RST19_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
18	RST18_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
17	RST17_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
16	RST16_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
15	RST15_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
14	RST14_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
13	RST13_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
12	RST12_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
11	RST11_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
10	RST10_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
9	RST9_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
8	RST8_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
7	RST7_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
6	RST6_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
5	RST5_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
4	RST4_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
3	RST3_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
2	RST2_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
1	RST1_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.
0	RST0_MON	x	R	Reset monitor. For the target signal, refer to **Table 4.4-27** to **Table 4.4-30.** 0b: The target unit is released from the reset state. 1b: The target unit is in the reset state.

x: Undefined value

Table 4.4-27 Specifications of the CPG_RSTMON_m Registers (m = 0 to 2)

Spec.	CPG_RSTMON_0	CPG_RSTMON_1	CPG_RSTMON_2
Offset address	0A00h	0A04h	0A08h
RST0_MON	CA55_RESET0	Reserved	CST_0_ATB_CA55_RESETN
RST1_MON	CA55_RESET1	SYS_0_PRESETN	CST_0_ATB_CM33_RESETN
RST2_MON	CA55_RESET2	DMAC_0_ARESETN	Reserved
RST3_MON	CA55_RESET3	DMAC_1_ARESETN	CST_0_ATB_CST_RESETN
RST4_MON	CA55_RESET4	DMAC_2_ARESETN	CST_0_AXI_SB_RESETN
RST5_MON	CA55_RESET5	DMAC_3_ARESETN	CST_0_AXI_ETR_RESETN
RST6_MON	CA55_RESET6	DMAC_4_ARESETN	CST_0_NTRST
RST7_MON	CA55_RESET7	ICU_0_PRESETN_I	CST_0_NPOTRST
RST8_MON	CA55_RESET8	CRC_0_RST	SYC_0_RESETN
RST9_MON	CA55_RESET9	GIC_0_GICRESET_N	MHU_0_PRESETN
RST10_MON	CA55_RESET10	GIC_0_DBG_GICRESET_N	GPT_0_RST_P_REG
RST11_MON	CA55_RESET11	Reserved	GPT_0_RST_S_REG
RST12_MON	CA55_RESET12	SRAM_0_ARESETN	GPT_1_RST_P_REG
RST13_MON	CA55_RESET13	Reserved	GPT_1_RST_S_REG
RST14_MON	CA55_RESET14	Reserved	POEGA_0_RST
RST15_MON	CA55_RESET15	SRAM_1_ARESETN	POEGB_0_RST
RST16_MON	CA55_RESET16	Reserved	POEGC_0_RST
RST17_MON	CM33_RESET2	Reserved	POEGD_0_RST
RST18_MON	CM33_RESET0	Reserved	POEGA_1_RST
RST19_MON	CM33_RESET1	Reserved	POEGB_1_RST
RST20_MON	Reserved	Reserved	POEGC_1_RST
RST21_MON	Reserved	Reserved	POEGD_1_RST
RST22_MON	Reserved	Reserved	MCPU_CMTW_0_RST_M
RST23_MON	Reserved	BTROM_0_ARESETN	MCPU_CMTW_1_RST_M
RST24_MON	Reserved	CST_0_CS_RESETN	MCPU_CMTW_2_RST_M
RST25_MON	Reserved	CST_0_TS_RESETN	MCPU_CMTW_3_RST_M
RST26_MON	Reserved	CST_0_APB_SB_RESETN	RCPU_CMTW_0_RST_M
RST27_MON	Reserved	CST_0_APB_CA55_RESETN	RCPU_CMTW_1_RST_M
RST28_MON	Reserved	CST_0_APB_CM33_RESETN	RCPU_CMTW_2_RST_M
RST29_MON	Reserved	Reserved	RCPU_CMTW_3_RST_M
RST30_MON	Reserved	CST_0_AHB_CM33_RESETN	MCPU_GTM_0_PRESETZ
RST31_MON	Reserved	CST_0_AHB_ATH_RESETN	MCPU_GTM_1_PRESETZ

Table 4.4-28 Specifications of the CPG_RSTMON_m Registers (m = 3 to 5)

Spec.	CPG_RSTMON_3	CPG_RSTMON_4	CPG_RSTMON_5
Offset address	0A0Ch	0A10h	0A14h
RST0_MON	ACPU_GTM_0_PRESETZ	RSCI_7_PRESETN	—
RST1_MON	ACPU_GTM_1_PRESETZ	RSCI_7_TRESETN	—
RST2_MON	RCPU_GTM_0_PRESETZ	RSCI_8_PRESETN	—
RST3_MON	RCPU_GTM_1_PRESETZ	RSCI_8_TRESETN	PCIE_0_ARESETN
RST4_MON	RCPU_GTM_2_PRESETZ	RSCI_9_PRESETN	—
RST5_MON	RCPU_GTM_3_PRESETZ	RSCI_9_TRESETN	—
RST6_MON	WDT_0_RESET	SCIF_0_RST_SYSTEM_N	—
RST7_MON	WDT_1_RESET	I3C_0_PRESETN	—
RST8_MON	WDT_2_RESET	I3C_0_TRESETN	—
RST9_MON	WDT_3_RESET	RIIC_0_MRST	—
RST10_MON	RTC_0_RST_RTC	RIIC_1_MRST	—
RST11_MON	RTC_0_RST_RTC_V	RIIC_2_MRST	—
RST12_MON	RSPI_0_PRESETN	RIIC_3_MRST	—
RST13_MON	RSPI_0_TRESETN	RIIC_4_MRST	Reserved
RST14_MON	RSPI_1_PRESETN	RIIC_5_MRST	Reserved
RST15_MON	RSPI_1_TRESETN	RIIC_6_MRST	Reserved
RST16_MON	RSPI_2_PRESETN	RIIC_7_MRST	Reserved
RST17_MON	RSPI_2_TRESETN	RIIC_8_MRST	Reserved
RST18_MON	RSCI_0_PRESETN	CANFD_0_RSTP_N	Reserved
RST19_MON	RSCI_0_TRESETN	CANFD_0_RSTC_N	Reserved
RST20_MON	RSCI_1_PRESETN	SPI_HRESETN	Reserved
RST21_MON	RSCI_1_TRESETN	SPI_ARESETN	Reserved
RST22_MON	RSCI_2_PRESETN	IOTOP_0_RESETN	CRU_0_PRESETN
RST23_MON	RSCI_2_TRESETN	IOTOP_0_ERROR_RESETN	CRU_0_ARESETN
RST24_MON	RSCI_3_PRESETN	SDHI_0_IXRST	CRU_0_S_RESETN
RST25_MON	RSCI_3_TRESETN	SDHI_1_IXRST	Reserved
RST26_MON	RSCI_4_PRESETN	SDHI_2_IXRST	Reserved
RST27_MON	RSCI_4_TRESETN	—	Reserved
RST28_MON	RSCI_5_PRESETN	Reserved	Reserved
RST29_MON	RSCI_5_TRESETN	—	Reserved
RST30_MON	RSCI_6_PRESETN	—	Reserved
RST31_MON	RSCI_6_TRESETN	—	Reserved

Note: For more information, refer to the User's Manual Additional Document.

Table 4.4-29 Specifications of the CPG_RSTMON_m Registers (m = 6 to 8)

Spec.	CPG_RSTMON_6	CPG_RSTMON_7	CPG_RSTMON_8
Offset address	0A18h	0A1Ch	0A20h
RST0_MON	Reserved	SPDIF_0_RST	AXI_TZCDDR_0_ARESET1N
RST1_MON	Reserved	SPDIF_1_RST	AXI_TZCDDR_0_ARESET2N
RST2_MON	Reserved	SPDIF_2_RST	AXI_TZCDDR_0_ARESET3N
RST3_MON	Reserved	PDM_0_PRESETN	AXI_TZCDDR_0_ARESET4N
RST4_MON	Reserved	PDM_0_CRESETN	Reserved
RST5_MON	Reserved	PDM_1_PRESETN	Reserved
RST6_MON	Reserved	PDM_1_CRESETN	Reserved
RST7_MON	Reserved	ADC_0_ADRST_N	Reserved
RST8_MON	DSI_0_PRESETN	Reserved	Reserved
RST9_MON	DSI_0_ARESETN	TSU_1_PRESETN	Reserved
RST10_MON	Reserved	Reserved	Reserved
RST11_MON	Reserved	Reserved	LVDS_TOP_RESET_N
RST12_MON	Reserved	OTPC_0_RESET_N	VSPI_x_smreset_vspiss_n
RST13_MON	LCDC_0_RESET_N	Reserved	VSPI_x_sreset_vsp1z
RST14_MON	GE3D_RESETN	Reserved	VSPI_x_spreset_fcpvi0
RST15_MON	GE3D_AXI_RESETN	RCPU_AXI_RESETN	LCDC_1_RESET_N
RST16_MON	GE3D_ACE_RESETN	MCPU_AXI_RESETN	Reserved
RST17_MON	VCD_0_RESETN	ACPU_AXI_RESETN	Reserved
RST18_MON	SSIF_0_ASYNC_RESET_SSI	ACPU_Peri_VIDEO0_AXI_RESETN	Reserved
RST19_MON	SSIF_0_SYNC_RESET_SSI0	ACPU_Peri_VIDEO1_AXI_RESETN	Reserved
RST20_MON	SSIF_0_SYNC_RESET_SSI1	ACPU_Peri_NPU0_AXI_RESETN	Reserved
RST21_MON	SSIF_0_SYNC_RESET_SSI2	ACPU_Peri_NPU1_AXI_RESETN	Reserved
RST22_MON	SSIF_0_SYNC_RESET_SSI3	ACPU_Peri_COM0_AXI_RESETN	Reserved
RST23_MON	SSIF_0_SYNC_RESET_SSI4	ACPU_Peri_COM1_AXI_RESETN	Reserved
RST24_MON	SSIF_0_SYNC_RESET_SSI5	ACPU_Peri_DDR_AXI_RESETN	Reserved
RST25_MON	SSIF_0_SYNC_RESET_SSI6	VIDEO0_AXI_RESETN	Reserved
RST26_MON	SSIF_0_SYNC_RESET_SSI7	VIDEO1_AXI_RESETN	Reserved
RST27_MON	SSIF_0_SYNC_RESET_SSI8	NPU_AXI_RESETN	Reserved
RST28_MON	SSIF_0_SYNC_RESET_SSI9	COM_AXI_RESETN	Reserved
RST29_MON	SCU_0_RESET_SRU	AXI_TZCDDR_0_PRESET0N	Reserved
RST30_MON	DMACpp_0_ARST	AXI_TZCDDR_0_PRESET1N	Reserved
RST31_MON	ADG_0_RST_RESET_ADG	AXI_TZCDDR_0_ARESET0N	Reserved

Note: For more information, refer to the User's Manual Additional Document.

Table 4.4-30 Specifications of the CPG_RSTMON_m Registers (m = 9)

Spec.	CPG_RSTMON_9
Offset address	0A24h
RST0_MON	NPU_nRESET
RST1_MON	NPU_nARESET
RST2_MON	NPU_nPRESET
RST3_MON	Reserved
RST4_MON	FDP1_reset_brdg_n
RST5_MON	FDP1_reset_fdp1_p
RST6_MON	FDP1_reset_fcpf_p
RST7_MON	Reserved
RST8_MON	Reserved
RST9_MON	Reserved
RST10_MON	Reserved
RST11_MON	Reserved
RST12_MON	Reserved
RST13_MON	Reserved
RST14_MON	Reserved
RST15_MON	Reserved
RST16_MON	Reserved
RST17_MON	Reserved
RST18_MON	Reserved
RST19_MON	Reserved
RST20_MON	Reserved
RST21_MON	Reserved
RST22_MON	Reserved
RST23_MON	Reserved
RST24_MON	Reserved
RST25_MON	Reserved
RST26_MON	Reserved
RST27_MON	Reserved
RST28_MON	Reserved
RST29_MON	Reserved
RST30_MON	Reserved
RST31_MON	Reserved

Note: For more information, refer to the User's Manual Additional Document.

4.4.4.13 Error Reset Select Register 1 (CPG_ERRORRST_SEL1)

This register selects whether a cold reset is or is not applied and whether an external pin is or is not asserted in response to a reset request generated by the WDT.

Access Size : 32 bits
Address : <CPG_base> + 0B00h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	ERRRSTSEL3_WEN	ERRRSTSEL2_WEN	ERRRSTSEL1_WEN	ERRRSTSEL0_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	ERRRSTSEL3	ERRRSTSEL2	ERRRSTSEL1	ERRRSTSEL0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
19	ERRRSTSEL3_WEN	0h	W	Write enable for ERRRSTSEL3
18	ERRRSTSEL2_WEN	0h	W	Write enable for ERRRSTSEL2
17	ERRRSTSEL1_WEN	0h	W	Write enable for ERRRSTSEL1
16	ERRRSTSEL0_WEN	0h	W	Write enable for ERRRSTSEL0
15 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
3	ERRRSTSEL3	0h	RW	When the WDT generates a CA55 cold reset request 0b: A CA55 cold reset is masked. 1b: A CA55 cold reset is applied.
2	ERRRSTSEL2	0h	RW	When the WDT generates a CM33 cold reset request 0b: A CM33 cold reset is masked. 1b: A CM33 cold reset is applied.
1	ERRRSTSEL1	0h	RW	When the WDT generates a CA55 cold reset request 0b: Assertion of the WDTUDFCA signal is masked. 1b: The WDTUDFCA signal is asserted.
0	ERRRSTSEL0	0h	RW	When the WDT generates a CM33 cold reset request 0b: Assertion of the WDTUDFCM signal is masked. 1b: The WDTUDFCM signal is asserted.

4.4.4.14 Error Reset Select Registers (CPG_ERRORRST_SELm) (m = 2 to 8)

These registers select whether a system reset is or is not applied in response to generation of an error. For the target error interrupt signals of the individual bits, refer to **Table 4.4-31** and **Table 4.4-32**. For details on interrupt signals, refer to **4.6 Interrupt Controller**.

Access Size : 32 bits

Address : <CPG_base> + 0B04h + (m-2) x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ERRRSTSEL15_WEN	ERRRSTSEL14_WEN	ERRRSTSEL13_WEN	ERRRSTSEL12_WEN	ERRRSTSEL11_WEN	ERRRSTSEL10_WEN	ERRRSTSEL9_WEN	ERRRSTSEL8_WEN	ERRRSTSEL7_WEN	ERRRSTSEL6_WEN	ERRRSTSEL5_WEN	ERRRSTSEL4_WEN	ERRRSTSEL3_WEN	ERRRSTSEL2_WEN	ERRRSTSEL1_WEN	ERRRSTSEL0_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRRSTSEL15	ERRRSTSEL14	ERRRSTSEL13	ERRRSTSEL12	ERRRSTSEL11	ERRRSTSEL10	ERRRSTSEL9	ERRRSTSEL8	ERRRSTSEL7	ERRRSTSEL6	ERRRSTSEL5	ERRRSTSEL4	ERRRSTSEL3	ERRRSTSEL2	ERRRSTSEL1	ERRRSTSEL0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	ERRRSTSEL15_WEN	0h	W	Write enable for ERRRSTSEL15
30	ERRRSTSEL14_WEN	0h	W	Write enable for ERRRSTSEL14
29	ERRRSTSEL13_WEN	0h	W	Write enable for ERRRSTSEL13
28	ERRRSTSEL12_WEN	0h	W	Write enable for ERRRSTSEL12
27	ERRRSTSEL11_WEN	0h	W	Write enable for ERRRSTSEL11
26	ERRRSTSEL10_WEN	0h	W	Write enable for ERRRSTSEL10
25	ERRRSTSEL9_WEN	0h	W	Write enable for ERRRSTSEL9
24	ERRRSTSEL8_WEN	0h	W	Write enable for ERRRSTSEL8
23	ERRRSTSEL7_WEN	0h	W	Write enable for ERRRSTSEL7
22	ERRRSTSEL6_WEN	0h	W	Write enable for ERRRSTSEL6
21	ERRRSTSEL5_WEN	0h	W	Write enable for ERRRSTSEL5
20	ERRRSTSEL4_WEN	0h	W	Write enable for ERRRSTSEL4
19	ERRRSTSEL3_WEN	0h	W	Write enable for ERRRSTSEL3
18	ERRRSTSEL2_WEN	0h	W	Write enable for ERRRSTSEL2
17	ERRRSTSEL1_WEN	0h	W	Write enable for ERRRSTSEL1
16	ERRRSTSEL0_WEN	0h	W	Write enable for ERRRSTSEL0
15	ERRRSTSEL15	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.

Bit	Bit Name	Initial Value	R/W	Description
14	ERRRSTSEL14	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
13	ERRRSTSEL13	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
12	ERRRSTSEL12	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
11	ERRRSTSEL11	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
10	ERRRSTSEL10	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
9	ERRRSTSEL9	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
8	ERRRSTSEL8	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
7	ERRRSTSEL7	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
6	ERRRSTSEL6	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
5	ERRRSTSEL5	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
4	ERRRSTSEL4	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
3	ERRRSTSEL3	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
2	ERRRSTSEL2	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
1	ERRRSTSEL1	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.
0	ERRRSTSEL0	0h	RW	When a reset request is generated by the target error interrupt. For details, refer to **Table 4.4-31** and **Table 4.4-32**. 0b: The error system reset is masked. 1b: The error system reset is executed.

Table 4.4-31 Target Error Interrupts of the CPG_ERRORRST_SELm Registers (m = 2 to 5)

Spec.	CPG_ERRORRST_SEL2	CPG_ERRORRST_SEL3	CPG_ERRORRST_SEL4	CPG_ERRORRST_SEL5
Offset address	0B04h	0B08h	0B0Ch	0B10h
ERRRSTSEL0	WDT_CM33_iwdt_nmiundf_n	Reserved	Reserved	Reserved
ERRRSTSEL1	WDT_CA55_iwdt_nmiundf_n	Reserved	Reserved	GPT_U0_gpt_gtciv_n_0
ERRRSTSEL2	WDT_Other_0_iwdt_nmiundf_n	Reserved	Reserved	GPT_U0_gpt_gtciv_n_1
ERRRSTSEL3	WDT_Other_1_iwdt_nmiundf_n	Reserved	Reserved	GPT_U0_gpt_gtciv_n_2
ERRRSTSEL4	BUS_ERR_INT	Reserved	Reserved	GPT_U0_gpt_gtciv_n_3
ERRRSTSEL5	RAM_ERR_INT	Reserved	Reserved	GPT_U0_gpt_gtciv_n_4
ERRRSTSEL6	ACPU_nFAULTIRQ_0	Reserved	Reserved	GPT_U0_gpt_gtciv_n_5
ERRRSTSEL7	ACPU_nFAULTIRQ_1	Reserved	Reserved	GPT_U0_gpt_gtciv_n_6
ERRRSTSEL8	ACPU_nFAULTIRQ_2	Reserved	Reserved	GPT_U0_gpt_gtciv_n_7
ERRRSTSEL9	ACPU_nFAULTIRQ_3	Reserved	Reserved	GPT_U0_gpt_gtciu_n_0
ERRRSTSEL10	ACPU_nFAULTIRQ_4	Reserved	Reserved	GPT_U0_gpt_gtciu_n_1
ERRRSTSEL11	ACPU_nERRIRQ_0	Reserved	Reserved	GPT_U0_gpt_gtciu_n_2
ERRRSTSEL12	ACPU_nERRIRQ_1	Reserved	Reserved	GPT_U0_gpt_gtciu_n_3
ERRRSTSEL13	ACPU_nERRIRQ_2	MCPU_LOCKUP	Reserved	GPT_U0_gpt_gtciu_n_4
ERRRSTSEL14	ACPU_nERRIRQ_3	—	Reserved	GPT_U0_gpt_gtciu_n_5
ERRRSTSEL15	ACPU_nERRIRQ_4	—	Reserved	GPT_U0_gpt_gtciu_n_6

Note: For more information, refer to the User's Manual Additional Document.

Table 4.4-32 Target Error Interrupts of the CPG_ERRORRST_SELm Registers (m = 6 to 8)

Spec.	CPG_ERRORRST_SEL6	CPG_ERRORRST_SEL7	CPG_ERRORRST_SEL8
Offset address	0B14h	0B18h	0B1Ch
ERRRSTSEL0	GPT_U0_gpt_gtciu_n_7	GPT_U1_gpt_gtciv_n_7	GPT_U1_gpt_gtdei_n_7
ERRRSTSEL1	GPT_U0_gpt_gtdei_n_0	GPT_U1_gpt_gtciu_n_0	ADC_ada_adereq_n
ERRRSTSEL2	GPT_U0_gpt_gtdei_n_1	GPT_U1_gpt_gtciu_n_1	Reserved
ERRRSTSEL3	GPT_U0_gpt_gtdei_n_2	GPT_U1_gpt_gtciu_n_2	Reserved
ERRRSTSEL4	GPT_U0_gpt_gtdei_n_3	GPT_U1_gpt_gtciu_n_3	Reserved
ERRRSTSEL5	GPT_U0_gpt_gtdei_n_4	GPT_U1_gpt_gtciu_n_4	Reserved
ERRRSTSEL6	GPT_U0_gpt_gtdei_n_5	GPT_U1_gpt_gtciu_n_5	Reserved
ERRRSTSEL7	GPT_U0_gpt_gtdei_n_6	GPT_U1_gpt_gtciu_n_6	Reserved
ERRRSTSEL8	GPT_U0_gpt_gtdei_n_7	GPT_U1_gpt_gtciu_n_7	Reserved
ERRRSTSEL9	GPT_U1_gpt_gtciv_n_0	GPT_U1_gpt_gtdei_n_0	Reserved
ERRRSTSEL10	GPT_U1_gpt_gtciv_n_1	GPT_U1_gpt_gtdei_n_1	Reserved
ERRRSTSEL11	GPT_U1_gpt_gtciv_n_2	GPT_U1_gpt_gtdei_n_2	Reserved
ERRRSTSEL12	GPT_U1_gpt_gtciv_n_3	GPT_U1_gpt_gtdei_n_3	Reserved
ERRRSTSEL13	GPT_U1_gpt_gtciv_n_4	GPT_U1_gpt_gtdei_n_4	Reserved
ERRRSTSEL14	GPT_U1_gpt_gtciv_n_5	GPT_U1_gpt_gtdei_n_5	Reserved
ERRRSTSEL15	GPT_U1_gpt_gtciv_n_6	GPT_U1_gpt_gtdei_n_6	Reserved

4.4.4.15 Error Reset Registers (CPG_ERROR_RSTm) (m = 2 to 8)

Applying a reset in response to an error interrupt factor sets the corresponding bit of these registers to 1. The state flag is cleared by writing 1 to it. For the target error interrupt signals of the individual bits, refer to **Table 4.4-33** and **Table 4.4-34**. For details on interrupt signals, refer to **4.6 Interrupt Controller**.

Access Size : 32 bits

Address : <CPG_base> + 0B40h + (m-2) x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ERROR_RST15_WEN	ERROR_RST14_WEN	ERROR_RST13_WEN	ERROR_RST12_WEN	ERROR_RST11_WEN	ERROR_RST10_WEN	ERROR_RST9_WEN	ERROR_RST8_WEN	ERROR_RST7_WEN	ERROR_RST6_WEN	ERROR_RST5_WEN	ERROR_RST4_WEN	ERROR_RST3_WEN	ERROR_RST2_WEN	ERROR_RST1_WEN	ERROR_RST0_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERROR_RST15	ERROR_RST14	ERROR_RST13	ERROR_RST12	ERROR_RST11	ERROR_RST10	ERROR_RST9	ERROR_RST8	ERROR_RST7	ERROR_RST6	ERROR_RST5	ERROR_RST4	ERROR_RST3	ERROR_RST2	ERROR_RST1	ERROR_RST0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	ERROR_RST15_WEN	0h	W	Write enable for ERROR_RST15
30	ERROR_RST14_WEN	0h	W	Write enable for ERROR_RST14
29	ERROR_RST13_WEN	0h	W	Write enable for ERROR_RST13
28	ERROR_RST12_WEN	0h	W	Write enable for ERROR_RST12
27	ERROR_RST11_WEN	0h	W	Write enable for ERROR_RST11
26	ERROR_RST10_WEN	0h	W	Write enable for ERROR_RST10
25	ERROR_RST9_WEN	0h	W	Write enable for ERROR_RST9
24	ERROR_RST8_WEN	0h	W	Write enable for ERROR_RST8
23	ERROR_RST7_WEN	0h	W	Write enable for ERROR_RST7
22	ERROR_RST6_WEN	0h	W	Write enable for ERROR_RST6
21	ERROR_RST5_WEN	0h	W	Write enable for ERROR_RST5
20	ERROR_RST4_WEN	0h	W	Write enable for ERROR_RST4
19	ERROR_RST3_WEN	0h	W	Write enable for ERROR_RST3
18	ERROR_RST2_WEN	0h	W	Write enable for ERROR_RST2
17	ERROR_RST1_WEN	0h	W	Write enable for ERROR_RST1
16	ERROR_RST0_WEN	0h	W	Write enable for ERROR_RST0
15	ERROR_RST15	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.

Bit	Bit Name	Initial Value	R/W	Description
14	ERROR_RST14	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
13	ERROR_RST13	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
12	ERROR_RST12	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
11	ERROR_RST11	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
10	ERROR_RST10	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
9	ERROR_RST9	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
8	ERROR_RST8	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
7	ERROR_RST7	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
6	ERROR_RST6	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
5	ERROR_RST5	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
4	ERROR_RST4	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
3	ERROR_RST3	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
2	ERROR_RST2	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
1	ERROR_RST1	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.
0	ERROR_RST0	0h	RW	Applying a reset in response to the target signal sets this bit to 1. For details, refer to **Table 4.4-33** and **Table 4.4-34**. The bit is cleared by writing 1 to it.

Table 4.4-33 Target Error Interrupts of the CPG_ERROR_RSTm Registers (m = 2 to 5)

Spec.	CPG_ERROR_RST2	CPG_ERROR_RST3	CPG_ERROR_RST4	CPG_ERROR_RST5
Offset address	0B40h	0B44h	0B48h	0B4Ch
ERROR_RST0	WDT_CM33_iwdt_nmiundf_n	Reserved	Reserved	Reserved
ERROR_RST1	WDT_CA55_iwdt_nmiundf_n	Reserved	Reserved	GPT_U0_gpt_gtciv_n_0
ERROR_RST2	WDT_Other_0_iwdt_nmiundf_n	Reserved	Reserved	GPT_U0_gpt_gtciv_n_1
ERROR_RST3	WDT_Other_1_iwdt_nmiundf_n	Reserved	Reserved	GPT_U0_gpt_gtciv_n_2
ERROR_RST4	BUS_ERR_INT	Reserved	Reserved	GPT_U0_gpt_gtciv_n_3
ERROR_RST5	RAM_ERR_INT	Reserved	Reserved	GPT_U0_gpt_gtciv_n_4
ERROR_RST6	ACPU_nFAULTIRQ_0	Reserved	Reserved	GPT_U0_gpt_gtciv_n_5
ERROR_RST7	ACPU_nFAULTIRQ_1	Reserved	Reserved	GPT_U0_gpt_gtciv_n_6
ERROR_RST8	ACPU_nFAULTIRQ_2	Reserved	Reserved	GPT_U0_gpt_gtciv_n_7
ERROR_RST9	ACPU_nFAULTIRQ_3	Reserved	Reserved	GPT_U0_gpt_gtciu_n_0
ERROR_RST10	ACPU_nFAULTIRQ_4	Reserved	Reserved	GPT_U0_gpt_gtciu_n_1
ERROR_RST11	ACPU_nERRIRQ_0	Reserved	Reserved	GPT_U0_gpt_gtciu_n_2
ERROR_RST12	ACPU_nERRIRQ_1	Reserved	Reserved	GPT_U0_gpt_gtciu_n_3
ERROR_RST13	ACPU_nERRIRQ_2	MCPU_LOCKUP	Reserved	GPT_U0_gpt_gtciu_n_4
ERROR_RST14	ACPU_nERRIRQ_3	—	Reserved	GPT_U0_gpt_gtciu_n_5
ERROR_RST15	ACPU_nERRIRQ_4	—	Reserved	GPT_U0_gpt_gtciu_n_6

Note: For more information, refer to the User's Manual Additional Document.

Table 4.4-34 Target Error Interrupts of the CPG_ERROR_RSTm Registers (m = 6 to 8)

Spec.	CPG_ERROR_RST6	CPG_ERROR_RST7	CPG_ERROR_RST8
Offset address	0B50h	0B54h	0B58h
ERROR_RST0	GPT_U0_gpt_gtciu_n_7	GPT_U1_gpt_gtciv_n_7	GPT_U1_gpt_gtdei_n_7
ERROR_RST1	GPT_U0_gpt_gtdei_n_0	GPT_U1_gpt_gtciu_n_0	ADC_ada_adereq_n
ERROR_RST2	GPT_U0_gpt_gtdei_n_1	GPT_U1_gpt_gtciu_n_1	Reserved
ERROR_RST3	GPT_U0_gpt_gtdei_n_2	GPT_U1_gpt_gtciu_n_2	Reserved
ERROR_RST4	GPT_U0_gpt_gtdei_n_3	GPT_U1_gpt_gtciu_n_3	Reserved
ERROR_RST5	GPT_U0_gpt_gtdei_n_4	GPT_U1_gpt_gtciu_n_4	Reserved
ERROR_RST6	GPT_U0_gpt_gtdei_n_5	GPT_U1_gpt_gtciu_n_5	Reserved
ERROR_RST7	GPT_U0_gpt_gtdei_n_6	GPT_U1_gpt_gtciu_n_6	Reserved
ERROR_RST8	GPT_U0_gpt_gtdei_n_7	GPT_U1_gpt_gtciu_n_7	Reserved
ERROR_RST9	GPT_U1_gpt_gtciv_n_0	GPT_U1_gpt_gtdei_n_0	Reserved
ERROR_RST10	GPT_U1_gpt_gtciv_n_1	GPT_U1_gpt_gtdei_n_1	Reserved
ERROR_RST11	GPT_U1_gpt_gtciv_n_2	GPT_U1_gpt_gtdei_n_2	Reserved
ERROR_RST12	GPT_U1_gpt_gtciv_n_3	GPT_U1_gpt_gtdei_n_3	Reserved
ERROR_RST13	GPT_U1_gpt_gtciv_n_4	GPT_U1_gpt_gtdei_n_4	Reserved
ERROR_RST14	GPT_U1_gpt_gtciv_n_5	GPT_U1_gpt_gtdei_n_5	Reserved
ERROR_RST15	GPT_U1_gpt_gtciv_n_6	GPT_U1_gpt_gtdei_n_6	Reserved

4.4.4.16 Low-Power-Consumption Sequence Control Register 1 (CPG_LP_CTL1)

This register controls lower power consumption for the CPUs.

Access Size : 32 bits
Address : <CPG_base> + 0C00h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	CM33SLEEP_ACK	CA55SLEEP_ACK[3:0]				-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CM33WRESET_REQ	-	-	CM33SLEEP_REQ	CA55SLEEP_REQ[3:0]				-	-	-	-	-	-	-	STBY
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
28	CM33SLEEP_ACK	0h	RW	Response to transition of Cortex-M33 to sleep mode 0b: There was no request for the transition. 1b: There was a request for the transition.
27 to 24	CA55SLEEP_ACK[3:0]	0h	RW	Response to transition of Cortex-A55 cores to sleep mode 0b: There was no response to the transition. 1b: There was a response to the transition. bit[24]: For core 0 bit[25]: For core 1 bit[26]: For core 2 bit[27]: For core 3
23 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
15	CM33WRESET_REQ	0h	RW	Request for a Cortex-M33 warm reset 0b: There was no request for Cortex-M33 warm reset. 1b: There was a request for Cortex-M33 warm reset.
14 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
12	CM33SLEEP_REQ	0h	RW	Request for transition of Cortex-M33 to sleep mode 0b: There was no request for the transition. 1b: There was a request for the transition.
11 to 8	CA55SLEEP_REQ[3:0]	0h	RW	Request for transition of Cortex-A55 cores to sleep mode 0b: There was no request for the transition. 1b: There was a request for the transition. bit[8]: For core 0 bit[9]: For core 1 bit[10]: For core 2 bit[11]: For core 3
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
0	STBY	0h	RW	Request for transition to software standby mode 0b: There was no request for the transition. 1b: There was a request for the transition.

4.4.4.17 Low-Power-Consumption Sequence Control Register 2 (CPG_LP_CTL2)

This register controls the masking of interrupts to the CPU (CM33).

Access Size : 32 bits
Address : <CPG_base> + 0C04h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INTMASK_CM33_I
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
0	INTMASK_CM33_I	0h	RW	Setting of the signal to mask interrupts to Cortex-M33 from the ICU 0b: Interrupts are not masked. 1b: Interrupts are masked. This signal is asserted before transitions of Cortex-M33 to sleep mode (deep standby), software standby mode (WIC standby), or warm reset state. The signal is automatically negated by hardware.

4.4.4.18 GE3D Low-Power-Consumption Sequence Control Register (CPG_LP_GE3D_CTL)

This register controls lower power consumption for the GE3D unit.

Access Size : 32 bits
Address : <CPG_base> + 0C08h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	QDENY_ACE_MST	QDENY_ACE_SLV	QDENY_AXI_MST	QDENY_AXI_SLV	QDENY_GE3D	-	-	-	QACCEPTn_ACE_MST	QACCEPTn_ACE_SLV	QACCEPTn_AXI_MST	QACCEPTn_AXI_SLV	QACCEPTn_GE3D
Initial Value	0	0	0	x	x	x	x	x	0	0	0	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	QREQn_ACE_MST	QREQn_ACE_SLV	QREQn_AXI_MST	QREQn_AXI_SLV	QREQn_GE3D	-	-	-	QACTIVE_ACE_MST	QACTIVE_ACE_SLV	QACTIVE_AXI_MST	QACTIVE_AXI_SLV	QACTIVE_GE3D
Initial Value	0	0	0	1	1	1	1	1	0	0	0	x	x	x	x	x
R/W	R	R	R	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
28	QDENY_ACE_MST	x	R	ACE-lite MST Q-Channel request denial status This bit indicates the state of the QDENY signal from the ACE-lite asynchronous bridge master interface of the GE3D. The bit indicates the Q-channel state in combination with the QREQn_ACE_MST and QACCEPTn_ACE_MST bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
27	QDENY_ACE_SLV	x	R	ACE-lite SLV Q-Channel request denial status This bit indicates the state of the QDENY signal from the ACE-lite asynchronous bridge slave interface of the GE3D. The bit indicates the Q-channel state in combination with the QREQn_ACE_SLV and QACCEPTn_ACE_SLV bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
26	QDENY_AXI_MST	x	R	AXI4 MST Q-Channel request denial status This bit indicates the state of the QDENY signal from the AXI4 asynchronous bridge master interface of the GE3D. The bit indicates the Q-channel state in combination with the QREQn_AXI_MST and QACCEPTn_AXI_MST bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
25	QDENY_AXI_SLV	x	R	AXI4 SLV Q-Channel request denial status This bit indicates the state of the QDENY signal from the AXI4 asynchronous bridge slave interface of the GE3D. The bit indicates the Q-channel state in combination with the QREQn_AXI_SLV and QACCEPTn_AXI_SLV bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
24	QDENY_GE3D	x	R	GE3D Q-Channel request denial status This bit indicates the state of the QDENY signal from the GE3D. The bit indicates the Q-channel state in combination with the QREQn_GE3D and QACCEPTn_GE3D bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
23 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
20	QACCEPTn_ACE_MST	x	R	ACE-lite MST Q-Channel request accept status This bit indicates the state of the QACCEPTn signal from the ACE-lite asynchronous bridge master interface of the GE3D. The bit indicates the Q-channel state in combination with the QREQn_ACE_MST and QDENY_ACE_MST bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.

Bit	Bit Name	Initial Value	R/W	Description
19	QACCEPTn_ACE_SLV	x	R	ACE-lite SLV Q-Channel request accept status This bit indicates the state of the QACCEPTn signal from the ACE-lite asynchronous bridge slave interface of the GE3D. The bit indicates the Q-channel state in combination with the QREQn_ACE_SLV and QDENY_ACE_SLV bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
18	QACCEPTn_AXI_MST	x	R	AXI4 MST Q-Channel request accept status This bit indicates the state of the QACCEPTn signal from the AXI4 asynchronous bridge master interface of the GE3D. The bit indicates the Q-channel state in combination with the QREQn_AXI_MST and QDENY_AXI_MST bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
17	QACCEPTn_AXI_SLV	x	R	AXI4 SLV Q-Channel request accept status This bit indicates the state of the QACCEPTn signal from the AXI4 asynchronous bridge slave interface of the GE3D. The bit indicates the Q-channel state in combination with the QREQn_AXI_SLV and QDENY_AXI_SLV bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
16	QACCEPTn_GE3D	x	R	GE3D Q-Channel request accept status This bit indicates the state of the QACCEPTn signal from the GE3D. The bit indicates the Q-channel state in combination with the QREQn_GE3D and QDENY_GE3D bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
12	QREQn_ACE_MST	1h	RW	ACE-lite MST Q-Channel request status This bit sets the level of the QREQn signal to the ACE-lite asynchronous bridge master interface of the GE3D. When the bit is read, the value written will be read. The bit indicates the Q-channel state in combination with the QACCEPTn_ACE_MST and QDENY_ACE_MST bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
11	QREQn_ACE_SLV	1h	RW	ACE-lite SLV Q-Channel request status This bit sets the level of the QREQn signal to the ACE-lite asynchronous bridge slave interface of the GE3D. When the bit is read, the value written will be read. The bit indicates the Q-channel state in combination with the QACCEPTn_ACE_SLV and QDENY_ACE_SLV bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
10	QREQn_AXI_MST	1h	RW	AXI4 MST Q-Channel request status This bit sets the level of the QREQn signal to the AXI4 asynchronous bridge master interface of the GE3D. When the bit is read, the value written will be read. The bit indicates the Q-channel state in combination with the QACCEPTn_AXI_MST and QDENY_AXI_MST bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
9	QREQn_AXI_SLV	1h	RW	AXI4 SLV Q-Channel request status This bit sets the level of the QREQn signal to the AXI4 asynchronous bridge slave interface of the GE3D. When the bit is read, the value written will be read. The bit indicates the Q-channel state in combination with the QACCEPTn_AXI_SLV and QDENY_AXI_SLV bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
8	QREQn_GE3D	1h	RW	GE3D Q-Channel request status This bit sets the level of the QREQn signal to the GE3D. When the bit is read, the value written will be read. The bit indicates the Q-channel state in combination with the QACCEPTn_GE3D and QDENY_GE3D bits. For details on the combinations, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.

Bit	Bit Name	Initial Value	R/W	Description
4	QACTIVE_ACE_MST	x	R	ACE-lite MST Q-Channel active status This bit indicates the state of the QACTIVE signal from the ACE-lite asynchronous bridge master interface of the GE3D. The state indicated by the bit is used to determine whether to start the stopped clock operation under control of the Q-channel. For details on the specifications of the QACTIVE signal, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
3	QACTIVE_ACE_SLV	x	R	ACE-lite SLV Q-Channel active status This bit indicates the state of the QACTIVE signal from the ACE-lite asynchronous bridge slave interface of the GE3D. The state indicated by the bit is used to determine whether to start the stopped clock operation under control of the Q-channel. For details on the specifications of the QACTIVE signal, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
2	QACTIVE_AXI_MST	x	R	AXI4 MST Q-Channel active status This bit indicates the state of the QACTIVE signal from the AXI4 asynchronous bridge master interface of the GE3D. The state indicated by the bit is used to determine whether to start the stopped clock operation under control of the Q-channel. For details on the specifications of the QACTIVE signal, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
1	QACTIVE_AXI_SLV	x	R	AXI4 SLV Q-Channel active status This bit indicates the state of the QACTIVE signal from the AXI4 asynchronous bridge slave interface of the GE3D. The state indicated by the bit is used to determine whether to start the stopped clock operation under control of the Q-channel. For details on the specifications of the QACTIVE signal, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.
0	QACTIVE_GE3D	x	R	GE3D Q-Channel active status This bit indicates the state of the QACTIVE signal from the GE3D. The state indicated by the bit is used to determine whether to start the stopped clock operation under control of the Q-channel. For details on the specifications of the QACTIVE signal, refer to the AMBA® Low Power Interface Specification ARM® Q-Channel and P-Channel Interfaces.

x: Undefined value

4.4.4.19 CM33 Control Register (CPG_CM33_CTL)

This register controls the program fetch when releasing the CM33 cold reset.

Access Size : 32 bits
Address : <CPG_base> + 0C0Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CM33_BOOTCTL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
0	CM33_BOOTCTL	0h	RW	This bit controls the program fetch when releasing the CM33 cold reset. 0b: Fetch enable 1b: Fetch disable This bit is used in the debug mode.

4.4.4.20 Low-Power Sequence Cortex-M33 Control Register 0 (CPG_LP_CM33_CTL0)

This register controls lower power consumption for the CM33.

Access Size : 32 bits
Address : <CPG_base> + 0C18h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	CM33_LP_CTL25	CM33_LP_CTL24	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	x	0	0	0	0	0	0	0	x	1
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	CM33_WARMRESETREQ	-	-	-	-	CM33_LP_CTL4	-	-	-	CM33_LP_CTL0
Initial Value	0	0	0	0	0	0	x	x	0	0	0	x	0	0	0	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
25	CM33_LP_CTL25	x	R	WIC startup request response from the Cortex-M33 0: No request response 1: Request response received
24	CM33_LP_CTL24	0h	RW	WIC startup request status to the Cortex-M33 0: No request 1: Requested
23 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
17	-	x	R	Reserved. The read value is undefined. The written value should always be the read value.
16	-	1h	RW	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
9	CM33_WARMRESETREQ	x	R	Warm reset request from the CM33 0b: No request 1b: Requested
8	-	x	R	Reserved. The read value is undefined. The written value should always be the read value.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
4	CM33_LP_CTL4	x	R	CM33 deep standby status 0b: Neither the Cortex-M33 sleep mode nor software standby mode has been entered. 1b: Either the Cortex-M33 sleep mode or software standby mode has been entered.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
0	CM33_LP_CTL0	x	R	CM33 sleep mode status 0b: Not in sleep mode 1b: In sleep mode Asserted in multiple modes.

x: Undefined value

4.4.4.21 Low-Power Sequence Cortex-M33 Control Register 1 (CPG_LP_CM33_CTL1)

This register controls lower power consumption for the CM33.

Access Size : 32 bits

Address : <CPG_base> + 0C1Ch

Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	CM33_LP_QCH29	CM33_LP_QCH28	-	-	-	-	-	-	CM33_LP_QCH21	CM33_LP_QCH20	-	-	-	-
Initial Value	0	0	x	x	0	0	0	x	0	0	x	x	0	0	0	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	CM33_LP_QCH13	CM33_LP_QCH12	-	-	-	-	-	-	-	CM33_LP_QCH4	CM33_LP_QCH3	-	-	-
Initial Value	0	0	1	1	0	0	0	1	0	0	0	x	x	0	0	x
R/W	R	R	RW	RW	R	R	R	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
29	CM33_LP_QCH29	x	R	This bit indicates the state of the QDEBY signal from CM33. For details, refer to **2.2 CPU**.
28	CM33_LP_QCH28	x	R	This bit is used for CM33 software standby mode operation. For details, refer to **4.4.8.8 Software Standby Mode (only for the CM33 Cold Boot)**.
27 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
24	-	x	R	Reserved. The read value is undefined. The written value should always be the read value.
23 to 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
21	CM33_LP_QCH21	x	R	This bit indicates the state of the QACCEPTn signal from CM33. For details, refer to **2.2 CPU**.
20	CM33_LP_QCH20	x	R	This bit is used for CM33 software standby mode operation. For details, refer to **4.4.8.8 Software Standby Mode (only for the CM33 Cold Boot)**.
19 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
16	-	x	R	Reserved. The read value is undefined. The written value should always be the read value.
15 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
13	CM33_LP_QCH13	1h	RW	This bit sets the level of the QREQn signal to the CM33 debug domain. For details, refer to **2.2 CPU**.
12	CM33_LP_QCH12	1h	RW	This bit is used for CM33 software standby mode operation. For details, refer to **4.4.8.8 Software Standby Mode (only for the CM33 Cold Boot)**.
11 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
8	-	1h	RW	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
4	CM33_LP_QCH4	x	R	QACTIVE signal state from CM33 For details, refer to **2.2 CPU**.
3	CM33_LP_QCH3	x	R	FPU domain activation request 0b: No request 1b: Requested
2 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.

Bit	Bit Name	Initial Value	R/W	Description
0	-	x	R	Reserved. The read value is undefined. The written value should always be the read value.

x: Undefined value

4.4.4.22 Cortex-A55 Clock Control Register 1 (CPG_LP_CA55_CTL1)

This register controls the power modes for the CA55 cluster.

Access Size : 32 bits
Address : <CPG_base> + 0C20h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	CLUSTERPACCEPT	CLUSTERPDENY	CLUSTERPSTATE[6:0]							CLUSTERPREQ
Initial Value	0	0	0	0	0	0	x	x	1	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
9	CLUSTERPACCEPT	x	R	Power mode transition acceptance for Cluster 0b: Transition rejected 1b: Transition accepted
8	CLUSTERPDENY	x	R	Power mode transition rejected for Cluster 0b: Transition accepted 1b: Transition rejected
7 to 1	CLUSTERPSTATE[6:0]	48h	RW	Transition destination power mode for Cluster 100_1000b: ON 000_0000b: OFF Others: Setting prohibited
0	CLUSTERPREQ	0h	RW	Power mode transition request control for Cluster 0b: Request stopped 1b: Request output

x: Undefined value

4.4.4.23 Cortex-A55 Clock Control Register 2 (CPG_LP_CA55_CTL2)

This register controls the power modes of CA55 cores 0 and 1.

Access Size : 32 bits
Address : <CPG_base> + 0C24h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	COREPACCEPT1	COREPDENY1	COREPSTATE1[5:0]						COREPREQ1
Initial Value	0	0	0	0	0	0	0	x	x	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	COREPACCEPT0	COREPDENY0	COREPSTATE0[5:0]						COREPREQ0
Initial Value	0	0	0	0	0	0	0	x	x	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
24	COREPACCEPT1	x	R	Power mode transition acceptance for core 1 0b: Transition rejected 1b: Transition accepted
23	COREPDENY1	x	R	Power mode transition rejected for core 1 0b: Transition accepted 1b: Transition rejected
22 to 17	COREPSTATE1[5:0]	0h	RW	Transition destination power mode for core 1 00_1000b: ON 00_0001b: OFF_EMU 00_0000b: OFF Others: Setting prohibited
16	COREPREQ1	0h	RW	Power mode transition request control for core 1 0b: Request stopped 1b: Request output
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
8	COREPACCEPT0	x	R	Power mode transition acceptance for core 0 0b: Transition rejected 1b: Transition accepted
7	COREPDENY0	x	R	Power mode transition rejected for core 0 0b: Transition accepted 1b: Transition rejected
6 to 1	COREPSTATE0[5:0]	8h	RW	Transition destination power mode for core 0 00_1000b: ON 00_0001b: OFF_EMU 00_0000b: OFF Others: Setting prohibited
0	COREPREQ0	0h	RW	Power mode transition request control for core 0 0b: Request stopped 1b: Request output

x: Undefined value

4.4.4.24 Cortex-A55 Clock Control Register 3 (CPG_LP_CA55_CTL3)

This register controls the power modes of CA55 cores 2 and 3.

Access Size : 32 bits
Address : <CPG_base> + 0C28h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	COREPACCEPT3	COREPDENY3	COREPSTATE3[5:0]						COREPREQ3
Initial Value	0	0	0	0	0	0	0	x	x	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	COREPACCEPT2	COREPDENY2	COREPSTATE2[5:0]						COREPREQ2
Initial Value	0	0	0	0	0	0	0	x	x	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
24	COREPACCEPT3	x	R	Power mode transition acceptance for core 3 0b: Transition rejected 1b: Transition accepted
23	COREPDENY3	x	R	Power mode transition rejected for core 3 0b: Transition accepted 1b: Transition rejected
22 to 17	COREPSTATE3[5:0]	0h	RW	Transition destination power mode for core 3 00_1000b: ON 00_0001b: OFF_EMU 00_0000b: OFF Others: Setting prohibited
16	COREPREQ3	0h	RW	Power mode transition request control for core 3 0b: Request stopped 1b: Request output
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
8	COREPACCEPT2	x	R	Power mode transition acceptance for core 2 0b: Transition rejected 1b: Transition accepted
7	COREPDENY2	x	R	Power mode transition rejected for core 2 0b: Transition accepted 1b: Transition rejected
6 to 1	COREPSTATE2[5:0]	0h	RW	Transition destination power mode for core 2 00_1000b: ON 00_0001b: OFF_EMU 00_0000b: OFF Others: Setting prohibited
0	COREPREQ2	0h	RW	Power mode transition request control for core 2 0b: Request stopped 1b: Request output

x: Undefined value

4.4.4.25 Cortex-A55 Clock Control Register 6 (CPG_LP_CA55_CTL6)

This register is used to read the states of the individual cores of the CA55.

Access Size : 32 bits
Address : <CPG_base> + 0C34h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	COREINSTRRUN[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
3 to 0	COREINSTRRUN[3:0]	x	R	The operating state of the core 0b: Power off or WFE/WFI state 1b: Running state

x: Undefined value

4.4.4.26 Cortex-A55 Clock Control Register 7 (CPG_LP_CA55_CTL7)

This register controls the state where communications are shut off of the CA55.

Access Size : 32 bits

Address : <CPG_base> + 0C38h

Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	CSYSREQ_TS	CSYSREQ_AT3	CSYSREQ_AT2	CSYSREQ_AT1	CSYSREQ_AT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	CSYSACK_TS	CSYSACK_AT3	CSYSACK_AT2	CSYSACK_AT1	CSYSACK_AT0	-	-	-	-	-	-	-	-
Initial Value	0	0	0	x	x	x	x	x	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
20	CSYSREQ_TS	0h	RW	Clock powerdown request.
19	CSYSREQ_AT3	0h	RW	System exit low-power state request.
18	CSYSREQ_AT2	0h	RW	System exit low-power state request.
17	CSYSREQ_AT1	0h	RW	System exit low-power state request.
16	CSYSREQ_AT0	0h	RW	System exit low-power state request.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
12	CSYSACK_TS	x	R	Clock powerdown acknowledge.
11	CSYSACK_AT3	x	R	Exit low-power state acknowledgement.
10	CSYSACK_AT2	x	R	Exit low-power state acknowledgement.
9	CSYSACK_AT1	x	R	Exit low-power state acknowledgement.
8	CSYSACK_AT0	x	R	Exit low-power state acknowledgement.
7 to 0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.

x: Undefined value

4.4.4.27 Low-Power Sequence Control Register (CPG_LP_PMU_CTL1)

This register controls the PMU.

Access Size : 32 bits
Address : <CPG_base> + 0C4Ch
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	PD_OTHERS_ACK	CTRLACK	PD_OTHERS_REQ	CTRLREQ
Initial Value	0	0	0	0	0	0	0	0	0	0	0	x	x	x	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
4	-	x	R	Reserved. The read value is undefined. The written value should always be the read value.
3	PD_OTHERS_ACK	x	R	PMU control (power domain control status notification) 0b: Off control completed 1b: On control completed
2	CTRLACK	x	R	PMU control (power domain control completion response) 0b: Control completed 1b: Control in progress
1	PD_OTHERS_REQ	0h	RW	PMU control (power domain on/off request) 0b: Off request 1b: On request
0	CTRLREQ	0h	RW	PMU control (power domain control request) 0b: Request stopped 1b: Request output

x: Undefined value

4.4.4.28 SRAM Standby Control Registers (CPG_LP_SRAM_STBY_CTLm) (m = 1 to 3)

These registers control the SRAM standby mode for each unit. For details, refer to **Table 4.4-35** to **Table 4.4-37** and **4.4.8.5 SRAM Power-Saving Mode**.

Access Size : 32 bits
Address : <CPG_base> + 0C50h + (m-1) x 0004h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRAM_STBY_CTL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRAM_STBY_CTL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRAM_STBY_CTL[31:0]	0h	*	For the register specification, refer to **Table 4.4-35** to **Table 4.4-37**.

Note: Regarding the attribute, refer to **Table 4.4-35**.

Table 4.4-35 Specifications of the CPG_LP_SRAM_STBY_CTL1 Register

Bit Number	Bit Name	Bit Attribute	Target Unit
0	Reserved	R	—
1	Reserved	R	—
2	Reserved	R	—
3	Reserved	R	—
4	Reserved	R	—
5	Reserved	R	—
6	Reserved	R	—
7	Reserved	R	—
8	Reserved	R	—
9	VCD_0_SDM	RW	VCD
10	VCD_0_RS	RW	VCD
11	VCD_0_RSO	R	VCD
12	CANFD_0_SDM	RW	CANFD
13	CANFD_0_RS	RW	CANFD
14	CANFD_0_RSO	R	CANFD
15	Reserved	R	—
16	Reserved	R	—
17	Reserved	R	—
18	Reserved	R	—
19	Reserved	R	—
20	Reserved	R	—
21	Reserved	R	—
22	Reserved	R	—
23	Reserved	R	—
24	Reserved	R	—
25	Reserved	R	—
26	Reserved	R	—
27	Reserved	R	—
28	Reserved	R	—
29	Reserved	R	—
30	Reserved	R	—
31	Reserved	R	—

Table 4.4-36 Specifications of the CPG_LP_SRAM_STBY_CTL2 Register

Bit Number	Bit Name	Bit Attribute	Target Unit
0	SRAM_MCPU_0_CENMASK	RW	SRAM0
1	SRAM_MCPU_0_SDM	RW	SRAM0
2	SRAM_MCPU_0_RS	RW	SRAM0
3	SRAM_MCPU_0_RSO	R	SRAM0
4	Reserved	R	—
5	Reserved	R	—
6	Reserved	R	—
7	Reserved	R	—
8	Reserved	R	—
9	Reserved	R	—
10	Reserved	R	—
11	Reserved	R	—
12	SRAM_NPU_0_CENMASK	RW	SRAM1
13	SRAM_NPU_0_SDM	RW	SRAM1
14	SRAM_NPU_0_RS	RW	SRAM1
15	SRAM_NPU_0_RSO	R	SRAM1
16	Reserved	R	—
17	Reserved	R	—
18	Reserved	R	—
19	Reserved	R	—
20	Reserved	R	—
21	Reserved	R	—
22	Reserved	R	—
23	Reserved	R	—
24	Reserved	R	—
25	Reserved	R	—
26	Reserved	R	—
27	Reserved	R	—
28	Reserved	R	—
29	Reserved	R	—
30	Reserved	R	—
31	Reserved	R	—

Table 4.4-37 Specifications of the CPG_LP_SRAM_STBY_CTL3 Register

Bit Number	Bit Name	Bit Attribute	Target Unit
0	Reserved	R	—
1	Reserved	R	—
2	Reserved	R	—
3	Reserved	R	—
4	Reserved	R	—
5	Reserved	R	—
6	Reserved	R	—
7	Reserved	R	—
8	Reserved	R	—
9	Reserved	R	—
10	Reserved	R	—
11	Reserved	R	—
12	Reserved	R	—
13	Reserved	R	—
14	Reserved	R	—
15	Reserved	R	—
16	VSPI_SDM	RW	VSPI
17	VSPI_RS	RW	VSPI
18	VSPI_RSO	R	VSPI
19	NPU_SDM	RW	NPU
20	NPU_RS	RW	NPU
21	NPU_RSO	R	NPU
22	GE3D_SDM	RW	GE3D
23	GE3D_RS	RW	GE3D
24	GE3D_RSO	R	GE3D
25	FDP1_SDM	RW	FDP1
26	FDP1_RS	RW	FDP1
27	FDP1_RSO	R	FDP1
28	Reserved	R	—
29	Reserved	R	—
30	Reserved	R	—
31	Reserved	R	—

4.4.4.29 CST Control Register 2 (CPG_LP_CST_CTL2)

This register controls shutting off communications between the CA55 and CST.

Access Size : 32 bits

Address : <CPG_base> + 0C70h

Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	LPI_ATB_ACK_CA553	LPI_ATB_REQ_CA553	-	-	-	-	-	-	LPI_ATB_ACK_CA552	LPI_ATB_REQ_CA552
Initial Value	0	0	0	0	0	x	x	1	0	0	0	0	0	x	x	1
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	LPI_ATB_ACK_CA551	LPI_ATB_REQ_CA551	-	-	-	-	-	-	LPI_ATB_ACK_CA550	LPI_ATB_REQ_CA550
Initial Value	0	0	0	0	0	x	x	1	0	0	0	0	0	x	x	1
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
26	-	x	R	Reserved. The read value is undefined. The written value should always be the read value.
25	LPI_ATB_ACK_CA553	x	R	LPI Control ATB for CA55. Clock is required when driven HIGH.
24	LPI_ATB_REQ_CA553	1h	RW	LPI Control ATB for CA55. Clock powerdown request. (Low(0b) active)
23 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
18	-	x	R	Reserved. The read value is undefined. The written value should always be the read value.
17	LPI_ATB_ACK_CA552	x	R	LPI Control ATB for CA55. Clock is required when driven HIGH.
16	LPI_ATB_REQ_CA552	1h	RW	LPI Control ATB for CA55. Clock powerdown request. (Low(0b) active)
15 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
10	-	x	R	Reserved. The read value is undefined. The written value should always be the read value.
9	LPI_ATB_ACK_CA551	x	R	LPI Control ATB for CA55. Clock is required when driven HIGH.
8	LPI_ATB_REQ_CA551	1h	RW	LPI Control ATB for CA55. Clock powerdown request. (Low(0b) active)
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
2	-	x	R	Reserved. The read value is undefined. The written value should always be the read value.
1	LPI_ATB_ACK_CA550	x	R	LPI Control ATB for CA55. Clock is required when driven HIGH.
0	LPI_ATB_REQ_CA550	1h	RW	LPI Control ATB for CA55. Clock powerdown request. (Low(0b) active)

x: Undefined value

4.4.4.30 OTP Handshaking Monitor Register (CPG_OTP_HANDSHAKE_MON)

This register monitors the flag for completion of transferring the SRAM recovery information.

Access Size : 32 bits
Address : <CPG_base> + 0C88h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DFT_DONE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
0	DFT_DONE	x	R	Flag for completion of transferring the SRAM recovery information 0b: The information has not been transferred. 1b: The information has been transferred.

x: Undefined value

4.4.4.31 Reset Control for the PD_OTHERS Domain (CPG_OTHERS_INI)

This register controls resets for the PD_OTHERS domain in the CPG.

Access Size : 32 bits
Address : <CPG_base> + 0C8Ch
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OTHERS_RST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
0	OTHERS_RST	x	RW	Reset control for the PD_OTHERS domain in the CPG 0b: Normal operation 1b: A reset is applied.

x: Undefined value

4.4.4.32 NPU Low-Power-Consumption Sequence Control Register 1 (CPG_LP_NPU_CTL1)

This register controls lower power consumption for the NPU unit.

Access Size : 32 bits

Address : <CPG_base> + 0C94h

Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PWRQACTIVE_EU55	PWRQDENY_EU55	PWRQACCEPTn_EU55	PWRQREQn_EU55	CLKQACTIVE_M_APB	CLKQDENY_M_APB	CLKQACCEPTn_M_APB	CLKQREQn_M_APB	CLKQACTIVE_S_APB	CLKQDENY_S_APB	CLKQACCEPTn_S_APB	CLKQREQn_S_APB	CLKQACTIVE_M_ADB_M1	CLKQDENY_M_ADB_M1	CLKQACCEPTn_M_ADB_M1	CLKQREQn_M_ADB_M1
Initial Value	x	x	x	1	x	x	x	1	x	x	x	1	x	x	x	1
R/W	R	R	R	RW	R	R	R	RW	R	R	R	RW	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLKQACTIVE_S_ADB_M1	CLKQDENY_S_ADB_M1	CLKQACCEPTn_S_ADB_M1	CLKQREQn_S_ADB_M1	CLKQACTIVE_M_ADB_M0	CLKQDENY_M_ADB_M0	CLKQACCEPTn_M_ADB_M0	CLKQREQn_M_ADB_M0	CLKQACTIVE_S_ADB_M0	CLKQDENY_S_ADB_M0	CLKQACCEPTn_S_ADB_M0	CLKQREQn_S_ADB_M0	CLKQACTIVE_EU55	CLKQDENY_EU55	CLKQACCEPTn_EU55	CLKQREQn_EU55
Initial Value	x	x	x	1	x	x	x	1	x	x	x	1	x	x	x	1
R/W	R	R	R	RW	R	R	R	RW	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	PWRQACTIVE_EU55	x	R	Power Activation State for NPU EU55
30	PWRQDENY_EU55	x	R	Power Off Rejected for NPU EU55
29	PWRQACCEPTn_EU55	x	R	Power Off Permission of NPU EU55
28	PWRQREQn_EU55	1h	RW	Power Outage Requirements for NPU EU55
27	CLKQACTIVE_M_APB	x	R	Master Clock Activation State for NPU APB
26	CLKQDENY_M_APB	x	R	Master Clock Off Rejected for NPU APB
25	CLKQACCEPTn_M_APB	x	R	Master Clock Off Permission of NPU APB
24	CLKQREQn_M_APB	1h	RW	Master Clock Outage Requirements for NPU APB
23	CLKQACTIVE_S_APB	x	R	Slave Clock Activation State for NPU APB
22	CLKQDENY_S_APB	x	R	Slave Clock Off Rejected for NPU APB
21	CLKQACCEPTn_S_APB	x	R	Slave Clock Off Permission of NPU APB
20	CLKQREQn_S_APB	1h	RW	Slave Clock Outage Requirements for NPU APB
19	CLKQACTIVE_M_ADB_M1	x	R	Master Clock Activation State for NPU AXI M1
18	CLKQDENY_M_ADB_M1	x	R	Master Clock Off Rejected for NPU AXI M1
17	CLKQACCEPTn_M_ADB_M1	x	R	Master Clock Off Permission of NPU AXI M1
16	CLKQREQn_M_ADB_M1	1h	RW	Master Clock Outage Requirements for NPU AXI M1
15	CLKQACTIVE_S_ADB_M1	x	R	Slave Clock Activation State for NPU AXI M1
14	CLKQDENY_S_ADB_M1	x	R	Slave Clock Off Rejected for NPU AXI M1

Bit	Bit Name	Initial Value	R/W	Description
13	CLKQACCEPTn_S_ADB_M1	x	R	Slave Clock Off Permission of NPU AXI M1
12	CLKQREQn_S_ADB_M1	1h	RW	Slave Clock Outage Requirements for NPU AXI M1
11	CLKQACTIVE_M_ADB_M0	x	R	Master Clock Activation State for NPU AXI M0
10	CLKQDENY_M_ADB_M0	x	R	Master Clock Off Rejected for NPU AXI M0
9	CLKQACCEPTn_M_ADB_M0	x	R	Master Clock Off Permission of NPU AXI M0
8	CLKQREQn_M_ADB_M0	1h	RW	Master Clock Outage Requirements for NPU AXI M0
7	CLKQACTIVE_S_ADB_M0	x	R	Slave Clock Activation State for NPU AXI M0
6	CLKQDENY_S_ADB_M0	x	R	Slave Clock Off Rejected for NPU AXI M0
5	CLKQACCEPTn_S_ADB_M0	x	R	Slave Clock Off Permission of NPU AXI M0
4	CLKQREQn_S_ADB_M0	1h	RW	Slave Clock Outage Requirements for NPU AXI M0
3	CLKQACTIVE_EU55	x	R	Clock Activation State for NPU EU55
2	CLKQDENY_EU55	x	R	Clock Off Rejected for NPU EU55
1	CLKQACCEPTn_EU55	x	R	Clock Off Permission of NPU EU55
0	CLKQREQn_EU55	1h	RW	Clock Outage Requirements for NPU EU55

x: Undefined value

4.4.4.33 NPU Low-Power-Consumption Sequence Control Register 2 (CPG_LP_NPU_CTL2)

This register controls lower power consumption for the NPU unit.

Access Size : 32 bits
Address : <CPG_base> + 0C98h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	PWRQ ACTIVE _APB	PWRQ DENY_ APB	PWRQ ACCEP Tn_AP B	PWRQ REQn_ APB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	x	x	x	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
3	PWRQACTIVE_ APB	x	R	Power Activation State for NPU APB
2	PWRQDENY_A PB	x	R	Power Off Rejected for NPU APB
1	PWRQACCEPT n_APB	x	R	Power Off Permission of NPU APB
0	PWRQREQn_A PB	1h	RW	Power Outage Requirements for NPU APB

x: Undefined value

4.4.4.34 VSPI Low-Power-Consumption Sequence Control Register 1 (CPG_LP_VSPI_CTL1)

This register controls lower power consumption for the VSPI unit.

Access Size : 32 bits
Address : <CPG_base> + 0C9Ch
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	vso_apb_clk_s_qdeny	vso_apb_clk_s_qactive	vso_apb_clk_s_qaccept_n	vsi_apb_clk_s_qreq_n	vso_apb_clk_m_qdeny	vso_apb_clk_m_qactive	vso_apb_clk_m_qaccept_n	vsi_apb_clk_m_qreq_n	fvo_apb_pwr_qdeny	fvo_apb_pwr_qactive	fvo_apb_pwr_qaccept_n	fvi_apb_pwr_qreq_n
Initial Value	0	0	0	0	x	x	x	0	x	x	x	0	x	x	x	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	RW	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	fvo_apb_clk_s_qdeny	fvo_apb_clk_s_qactive	fvo_apb_clk_s_qaccept_n	fvi_apb_clk_s_qreq_n	fvo_apb_clk_m_qdeny	fvo_apb_clk_m_qactive	fvo_apb_clk_m_qaccept_n	fvi_apb_clk_m_qreq_n	fvo_axi_clkqactivem	fvo_axi_clkqdenym	fvo_axi_clkqacceptnm	fvi_axi_clkqreqnm	fvo_axi_clkqactives	fvo_axi_clkqdenys	fvo_axi_clkqacceptns	fvi_axi_clkqreqns
Initial Value	x	x	x	0	x	x	x	0	x	x	x	0	x	x	x	0
R/W	R	R	R	RW	R	R	R	RW	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
27	vso_apb_clk_s_qdeny	x	R	LPI deny to clock gating for master
26	vso_apb_clk_s_qactive	x	R	LPI request for clock for master
25	vso_apb_clk_s_qaccept_n	x	R	LPI accept to clock gating for master
24	vsi_apb_clk_s_qreq_n	0h	RW	LPI quiescence request for clock gating for master
23	vso_apb_clk_m_qdeny	x	R	LPI deny to clock gating for slave
22	vso_apb_clk_m_qactive	x	R	LPI request for clock for slave
21	vso_apb_clk_m_qaccept_n	x	R	LPI accept to clock gating for slave
20	vsi_apb_clk_m_qreq_n	0h	RW	LPI quiescence request for clock gating for slave
19	fvo_apb_pwr_qdeny	x	R	LPI deny for power down
18	fvo_apb_pwr_qactive	x	R	LPI request for power
17	fvo_apb_pwr_qaccept_n	x	R	LPI accept for power down
16	fvi_apb_pwr_qreq_n	0h	RW	LPI quiescence request for power down
15	fvo_apb_clk_s_qdeny	x	R	LPI deny to clock gating for master
14	fvo_apb_clk_s_qactive	x	R	LPI request for clock for master
13	fvo_apb_clk_s_qaccept_n	x	R	LPI accept to clock gating for master
12	fvi_apb_clk_s_qreq_n	0h	RW	LPI quiescence request for clock gating for master
11	fvo_apb_clk_m_qdeny	x	R	LPI deny to clock gating for slave

Bit	Bit Name	Initial Value	R/W	Description
10	fvo_apb_clk_m_qactive	x	R	LPI request for clock for slave
9	fvo_apb_clk_m_qaccept_n	x	R	LPI accept to clock gating for slave
8	fvi_apb_clk_m_qreq_n	0h	RW	LPI quiescence request for clock gating for slave
7	fvo_axi_clkqactivem	x	R	LPI deny to clock gating for master
6	fvo_axi_clkqdenym	x	R	LPI request for clock for master
5	fvo_axi_clkqacceptnm	x	R	LPI accept to clock gating for master
4	fvi_axi_clkqreqnm	0h	RW	LPI quiescence request for clock gating for master
3	fvo_axi_clkqactives	x	R	LPI deny to clock gating for slave
2	fvo_axi_clkqdenys	x	R	LPI request for clock for slave
1	fvo_axi_clkqacceptns	x	R	LPI accept to clock gating for slave
0	fvi_axi_clkqreqns	0h	RW	LPI quiescence request for clock gating for slave

x: Undefined value

4.4.4.35 VSPI Low-Power-Consumption Sequence Control Register 2 (CPG_LP_VSPI_CTL2)

This register controls lower power consumption for the VSPI unit.

Access Size : 32 bits
Address : <CPG_base> + 0CA0h
Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	vso_apb_pwr_qdeny	vso_apb_pwr_qactive	vso_apb_pwr_qaccept_n	vsi_apb_pwr_qreq_n
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	x	x	x	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be the read value.
3	vso_apb_pwr_qdeny	x	R	LPI deny for power down
2	vso_apb_pwr_qactive	x	R	LPI request for power
1	vso_apb_pwr_qaccept_n	x	R	LPI accept for power down
0	vsi_apb_pwr_qreq_n	0h	RW	LPI quiescence request for power down

x: Undefined value

4.4.4.36 FDP1 Low-Power-Consumption Sequence Control Register (CPG_LP_FDP1_CTL)

This register controls resets for the PD_OTHERS domain in the CPG.

Access Size : 32 bits

Address : <CPG_base> + 0CA4h

Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	fd1o_apb_pwrqactive	fd1o_apb_pwrqdeny	fd1o_apb_pwrqacceptn	fd1i_apb_pwrqreqn	fd1o_apb_clkmqactive	fd1o_apb_clkmqdeny	fd1o_apb_clkmqacceptn	fd1i_apb_clkmqreqn	fd1o_apb_clksqactive	fd1o_apb_clksqdeny	fd1o_apb_clksqacceptn	fd1i_apb_clksqreqn	ffo_apb_pwrqactive	ffo_apb_pwrqdeny	ffo_apb_pwrqacceptn	ffi_apb_pwrqreqn
Initial Value	x	x	x	1	x	x	x	1	x	x	x	1	x	x	x	1
R/W	R	R	R	RW	R	R	R	RW	R	R	R	RW	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ffo_apb_clkmqactive	ffo_apb_clkmqdeny	ffo_apb_clkmqacceptn	ffi_apb_clkmqreqn	ffo_apb_clksqactive	ffo_apb_clksqdeny	ffo_apb_clksqacceptn	ffi_apb_clksqreqn	ffo_axi_clkmqactive	ffo_axi_clkmqdeny	ffo_axi_clkmqacceptn	ffi_axi_clkmqreqn	ffo_axi_clksqactive	ffo_axi_clksqdeny	ffo_axi_clksqacceptn	ffi_axi_clksqreqn
Initial Value	x	x	x	1	x	x	x	1	x	x	x	1	x	x	x	1
R/W	R	R	R	RW	R	R	R	RW	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	fd1o_apb_pwrqactive	x	R	LPI deny for power down
30	fd1o_apb_pwrqdeny	x	R	LPI request for power
29	fd1o_apb_pwrqacceptn	x	R	LPI accept for power down
28	fd1i_apb_pwrqreqn	1h	RW	LPI quiescence request for power down
27	fd1o_apb_clkmqactive	x	R	LPI deny to clock gating for master
26	fd1o_apb_clkmqdeny	x	R	LPI request for clock for master
25	fd1o_apb_clkmqacceptn	x	R	LPI accept to clock gating for master
24	fd1i_apb_clkmqreqn	1h	RW	LPI quiescence request for clock gating for master
23	fd1o_apb_clksqactive	x	R	LPI deny to clock gating for slave
22	fd1o_apb_clksqdeny	x	R	LPI request for clock for slave
21	fd1o_apb_clksqacceptn	x	R	LPI accept to clock gating for slave
20	fd1i_apb_clksqreqn	1h	RW	LPI quiescence request for clock gating for slave
19	ffo_apb_pwrqactive	x	R	LPI deny for power down
18	ffo_apb_pwrqdeny	x	R	LPI request for power
17	ffo_apb_pwrqacceptn	x	R	LPI accept for power down
16	ffi_apb_pwrqreqn	1h	RW	LPI quiescence request for power down
15	ffo_apb_clkmqactive	x	R	LPI deny to clock gating for master
14	ffo_apb_clkmqdeny	x	R	LPI request for clock for master
13	ffo_apb_clkmqacceptn	x	R	LPI accept to clock gating for master

Bit	Bit Name	Initial Value	R/W	Description
12	ffi_apb_clkmqreqn	1h	RW	LPI quiescence request for clock gating for master
11	ffo_apb_clksqactive	x	R	LPI deny to clock gating for slave
10	ffo_apb_clksqdeny	x	R	LPI request for clock for slave
9	ffo_apb_clksqacceptn	x	R	LPI accept to clock gating for slave
8	ffi_apb_clksqreqn	1h	RW	LPI quiescence request for clock gating for slave
7	ffo_axi_clkmqactive	x	R	LPI deny to clock gating for master
6	ffo_axi_clkmqdeny	x	R	LPI request for clock for master
5	ffo_axi_clkmqacceptn	x	R	LPI accept to clock gating for master
4	ffi_axi_clkmqreqn	1h	RW	LPI quiescence request for clock gating for master
3	ffo_axi_clksqactive	x	R	LPI deny to clock gating for slave
2	ffo_axi_clksqdeny	x	R	LPI request for clock for slave
1	ffo_axi_clksqacceptn	x	R	LPI accept to clock gating for slave
0	ffi_axi_clksqreqn	1h	RW	LPI quiescence request for clock gating for slave

x: Undefined value

4.4.4.37 MSTOP Registers (CPG_BUS_m_MSTOP) (m = 1 to 13)

These registers control buses for the individual units. For details on the registers, refer to **Table 4.4-38** to **Table 4.4-40**.

Access Size : 32 bits

Address : <CPG_base> + 0D00h + (m-1) x 0004h

Initial Value : Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSTOP15_ON_WEN	MSTOP14_ON_WEN	MSTOP13_ON_WEN	MSTOP12_ON_WEN	MSTOP11_ON_WEN	MSTOP10_ON_WEN	MSTOP9_ON_WEN	MSTOP8_ON_WEN	MSTOP7_ON_WEN	MSTOP6_ON_WEN	MSTOP5_ON_WEN	MSTOP4_ON_WEN	MSTOP3_ON_WEN	MSTOP2_ON_WEN	MSTOP1_ON_WEN	MSTOP0_ON_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTOP15_ON	MSTOP14_ON	MSTOP13_ON	MSTOP12_ON	MSTOP11_ON	MSTOP10_ON	MSTOP9_ON	MSTOP8_ON	MSTOP7_ON	MSTOP6_ON	MSTOP5_ON	MSTOP4_ON	MSTOP3_ON	MSTOP2_ON	MSTOP1_ON	MSTOP0_ON
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	MSTOP15_ON_WEN	0h	W	Write enable for MSTOP15_ON
30	MSTOP14_ON_WEN	0h	W	Write enable for MSTOP14_ON
29	MSTOP13_ON_WEN	0h	W	Write enable for MSTOP13_ON
28	MSTOP12_ON_WEN	0h	W	Write enable for MSTOP12_ON
27	MSTOP11_ON_WEN	0h	W	Write enable for MSTOP11_ON
26	MSTOP10_ON_WEN	0h	W	Write enable for MSTOP10_ON
25	MSTOP9_ON_WEN	0h	W	Write enable for MSTOP9_ON
24	MSTOP8_ON_WEN	0h	W	Write enable for MSTOP8_ON
23	MSTOP7_ON_WEN	0h	W	Write enable for MSTOP7_ON
22	MSTOP6_ON_WEN	0h	W	Write enable for MSTOP6_ON
21	MSTOP5_ON_WEN	0h	W	Write enable for MSTOP5_ON
20	MSTOP4_ON_WEN	0h	W	Write enable for MSTOP4_ON
19	MSTOP3_ON_WEN	0h	W	Write enable for MSTOP3_ON
18	MSTOP2_ON_WEN	0h	W	Write enable for MSTOP2_ON
17	MSTOP1_ON_WEN	0h	W	Write enable for MSTOP1_ON
16	MSTOP0_ON_WEN	0h	W	Write enable for MSTOP0_ON
15	MSTOP15_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
14	MSTOP14_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)

Bit	Bit Name	Initial Value	R/W	Description
13	MSTOP13_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
12	MSTOP12_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
11	MSTOP11_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
10	MSTOP10_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
9	MSTOP9_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
8	MSTOP8_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
7	MSTOP7_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
6	MSTOP6_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
5	MSTOP5_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
4	MSTOP4_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
3	MSTOP3_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
2	MSTOP2_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
1	MSTOP1_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)
0	MSTOP0_ON	x	RW	For the target unit, refer to **Table 4.4-38** to **Table 4.4-40**. 0b: Normal operation 1b: Module stop state (MSTP)

x: Undefined value

Table 4.4-38 Target Units for Control by the CPG_BUS_m_MSTOP Registers (m = 1 to 4)

Spec.	CPG_BUS_1_MSTOP	CPG_BUS_2_MSTOP	CPG_BUS_3_MSTOP	CPG_BUS_4_MSTOP
Offset address	0D00h	0D04h	0D08h	0D0Ch
Initial value	0000_AFFFh	0000_E03Fh	0000_FE1Ch	0000_C427h
MSTOP0_ON	WDT1	SCU	SYC	MCPU_CMTW1
MSTOP1_ON	RIIC0	SCU (DMAC)	Reserved	MCPU_CMTW2
MSTOP2_ON	RIIC1	ADG	DMAC1	MCPU_CMTW3
MSTOP3_ON	RIIC2	SSIU	DMAC2	SRAM0
MSTOP4_ON	RIIC3	SSIU (DMAC)	GE3D	Reserved
MSTOP5_ON	RIIC4	ADMAC	GIC	xSPI
MSTOP6_ON	RIIC5	TZC400_DDR00	Reserved	Reserved
MSTOP7_ON	RIIC6	TZC400_DDR01	Reserved	PFC
MSTOP8_ON	RIIC7	Reserved	Reserved	Reserved
MSTOP9_ON	SPDIF0	Reserved	ADC	Reserved
MSTOP10_ON	SPDIF1	TZC400_AXI_RCPU	WDT0	Reserved
MSTOP11_ON	SPDIF2	TZC400_SRAMA	RTC	OTP
MSTOP12_ON	Reserved	TZC400_PCIE	RTC	TZC400_SRAMM
MSTOP13_ON	Reserved	ACPU_GTM0	RIIC8	TZC400_XSPI
MSTOP14_ON	ACPU bus to RCPU bus	ACPU_GTM1	SCIF	MHU
MSTOP15_ON	PCIE	TSU1	MCPU_CMTW0	Reserved

Table 4.4-39 Target Units for Control by the CPG_BUS_m_MSTOP Registers (m = 5 to 8)

Spec.	CPG_BUS_5_MSTOP	CPG_BUS_6_MSTOP	CPG_BUS_7_MSTOP	CPG_BUS_8_MSTOP
Offset address	0D10h	0D14h	0D18h	0D1Ch
Initial value	0000_FEE5h	0000_FFFFh	0000_FFFFh	0000_077Dh
MSTOP0_ON	Reserved	RCPU_CMTW1	DDR0	Reserved
MSTOP1_ON	CST	RCPU_CMTW2	DDR0	Reserved
MSTOP2_ON	Reserved	RCPU_CMTW3	Reserved	SD0
MSTOP3_ON	SRAM0 (Register)	POEG0A	Reserved	SD1
MSTOP4_ON	Reserved	POEG0B	Reserved	SD2
MSTOP5_ON	xSPI (Register)	POEG0C	Reserved	GBETH0
MSTOP6_ON	PDM0	POEG0D	Reserved	GBETH1
MSTOP7_ON	PDM1	POEG1A	USB20 (HOST)	Reserved
MSTOP8_ON	Reserved	POEG1B	USB21 (HOST)	Reserved
MSTOP9_ON	DMAC0	POEG1C	USB2 (FUNC)	Reserved
MSTOP10_ON	MCPU_GTM0	POEG1D	USB20 (PHY)	Reserved
MSTOP11_ON	MCPU_GTM1	GPT0	USB21 (PHY)	Reserved
MSTOP12_ON	WDT2	GPT1	USB30 (HOST)	SRAM1 (Register)
MSTOP13_ON	WDT3	DDR0	Reserved	Reserved
MSTOP14_ON	CRC	DDR0	USB30 (PHY)	Reserved
MSTOP15_ON	RCPU_CMTW0	DDR0	Reserved	Reserved

Table 4.4-40 Target Units for Control by the CPG_BUS_m_MSTOP Registers (m = 9 to 13)

Spec.	CPG_BUS_9_MSTOP	CPG_BUS_10_MSTOP	CPG_BUS_11_MSTOP	CPG_BUS_12_MSTOP	CPG_BUS_13_MSTOP
Offset address	0D20h	0D24h	0D28h	0D2Ch	0D30h
Initial value	0000_FBF0h	0000_DDEFh	0000_FFFFh	0000_0201h	0000_07FFh
MSTOP0_ON	Reserved	Reserved	RSPI0	RCPU_GTM3	Module stop signal for LVDS
MSTOP1_ON	Reserved	LCDC_0 (DU)	RSPI1	SRAM1	Module stop signal for VSPI
MSTOP2_ON	Reserved	LCDC_0 (FCPVD)	RSPI2	Reserved	Module stop signal for VSPI
MSTOP3_ON	Reserved	LCDC_0 (VSPD)	RSCI0	Reserved	Module stop signal for LCDC_1(DU)
MSTOP4_ON	CRU0	Reserved	RSCI1	Reserved	Module stop signal for LCDC_1(FCPVD)
MSTOP5_ON	Reserved	DDR0 (PHY)	RSCI2	Reserved	Module stop signal for LCDC_1(VSPD)
MSTOP6_ON	Reserved	Reserved	RSCI3	Reserved	Module stop signal for NPU
MSTOP7_ON	Reserved	DDR0 (Controller)	RSCI4	Reserved	Module stop signal for SHIPG
MSTOP8_ON	Reserved	Reserved	RSCI5	Reserved	Module stop signal for SHIP
MSTOP9_ON	Reserved	Reserved	RSCI6	MCPU bus to ACPU bus	Module stop signal for FDP1
MSTOP10_ON	Reserved	Reserved	RSCI7	ACPU bus to MCPU bus	Module stop signal for FDP1(FCPF)
MSTOP11_ON	VCD	DMAC3	RSCI8	Reserved	Reserved
MSTOP12_ON	VCD	DMAC4	RSCI9	Reserved	Reserved
MSTOP13_ON	VCD	Reserved	RCPU_GTM0	Reserved	Reserved
MSTOP14_ON	DSI (LINK)	CANFD	RCPU_GTM1	Reserved	Reserved
MSTOP15_ON	DSI (PHY)	I3C0	RCPU_GTM2	Reserved	Reserved

CAUTION

Access to a bus while the module is stopped is prohibited.

4.4.5 Functions for the Generation and Control of Clock Signals

4.4.5.1 Block Diagram of the Clock System

For more information, refer to the User's Manual Additional Document.

4.4.5.2 Control of Turning Unit Clock Signals On and Off

Control over turning unit clock signals on and off is realized by switching clock gating cells (CGCs) on and off under various conditions for enabling. The factors of control over turning clocks on and off are listed below.

- Control over turning clocks on and off by register settings
- Control over turning clocks on at the time of forcibly supplying them in the boot sequence.
- Control over turning clocks off at the time of release from the reset state
- Control over turning debug circuits (CSTs) off in normal mode (other than debug mode)

In addition, whether the CGCs are on or off can be confirmed by monitoring the clock signals output from the CGC control circuits with the *_CLK_MON registers.

4.4.5.3 Functions for Controlling, Setting, and Monitoring PLLs

This section describes functions for controlling, setting, and monitoring PLLs. The PLLs are controlled by internal registers of the CPG. The PLL states can be monitored by the internal registers of the CPG.

The functions for controlling, setting, and monitoring PLLs are listed below. Details of each of the functions are given in the following sections.

- PLL operating mode control
- Setting output clock signals
- Monitoring the PLL states (reset, locked)

Table 4.4-41 shows the implemented PLLs with or without SSCG.

Table 4.4-41 List of PLLs with or without SSCG

PLL Name	With or Without SSCG
PLLCM33	Without SSCG (fixed to off)
PLLCLN	Without SSCG (fixed to off)
PLLDTY	With SSCG (switched by a register setting)*1
PLLCA55	With SSCG (switched by a register setting)*1
PLLVDO	With SSCG (switched by a register setting)*1
PLLETH	Without SSCG (fixed to off)
PLLDSI0	With SSCG (switched by a register setting)
PLLDDR0	Without SSCG (default: fixed to off and switched by a register setting)
PLLDSI1	With SSCG (switched by a register setting)

Note 1. The default settings are defined by the state of MD_CLKS pin.

Figure 4.4-3 is a block diagram for controlling a PLL for which the SSCG function is available.

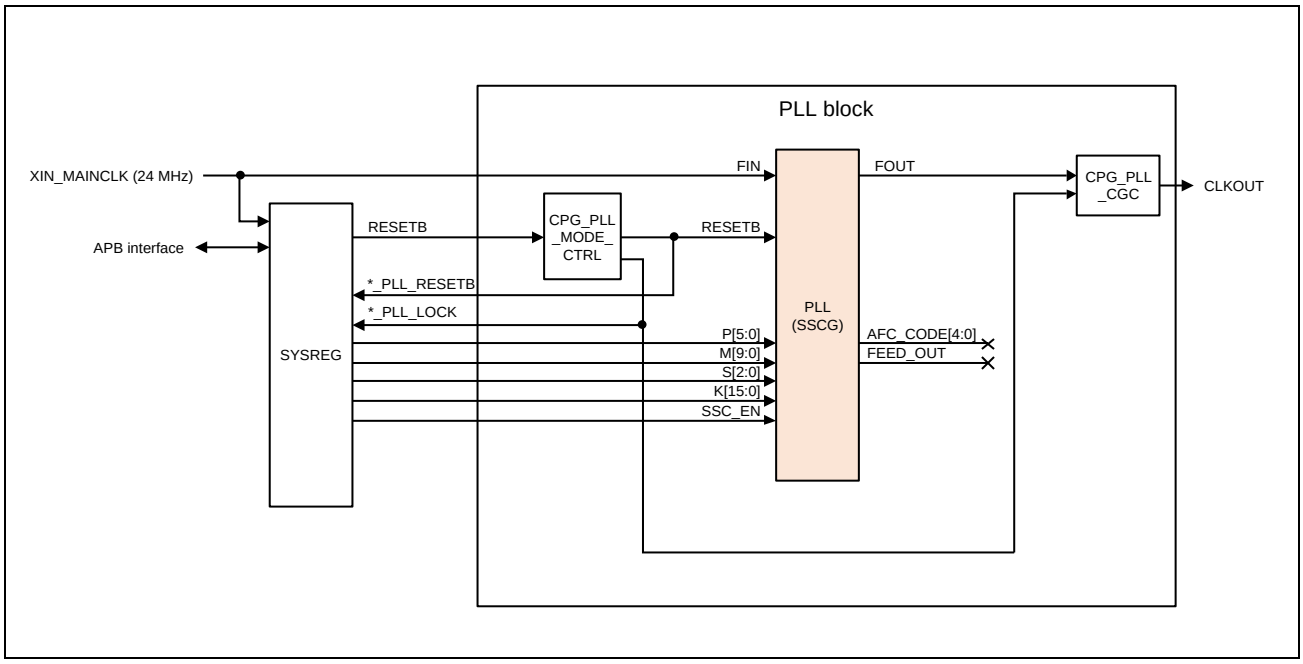


Figure 4.4-3 Block Diagram for Controlling a PLL for which the SSCG Function is Available

4.4.5.3.1 PLL operating mode control

The internal registers of the CPG are used to set the PLL operating mode and turn the SSCG on or off per PLL.

Either of the normal and standby modes is selectable as the operating mode.

The following items can be set with the internal registers of the CPG.

- Setting the operating mode (normal or standby mode)
- Turning the SSCG on or off (the setting is enabled in normal mode)

The related registers are listed in **Table 4.4-42**. For details on the settings, refer to the description of the given register.

For PLLCM33, PLLCLN, PLLETH, PLLDSI0, and PLLDSI1, the SSCG setting is fixed to off so making a setting to turn the SSCG on is prohibited.

Table 4.4-42 List of PLL Standby Control Registers

Register Name	Abbreviation	Function
PLLDTY standby control register	CPG_PLLDTY_STBY	Setting the standby mode (reset state) Turning the SSCG on or off
PLLCA55 standby control register	CPG_PLLCA55_STBY	Setting the standby mode (reset state) Turning the SSCG on or off
PLLVDO standby control register	CPG_PLLVDO_STBY	Setting the standby mode (reset state) Turning the SSCG on or off

4.4.5.3.2 Setting output clock signals

The internal registers of the CPG can be used to set the PLL output clock signal per PLL. The following item can be set.

- Setting the frequency of the output clock signal

The related registers are listed in **Table 4.4-43**.

Table 4.4-43 List of PLL Output Clock Setting Registers

Register Name	Abbreviation	Function
PLLCA55 output clock setting register 1	CPG_PLLCA55_CLK1	Setting the frequency
PLLCA55 output clock setting register 2	CPG_PLLCA55_CLK2	Setting the frequency
PLLDSI0 output clock setting register 1	CPG_PLLDSI0_CLK1	Setting the frequency
PLLDSI0 output clock setting register 2	CPG_PLLDSI0_CLK2	Setting the frequency
PLLDSI1 output clock setting register 1	CPG_PLLDSI1_CLK1	Setting the frequency
PLLDSI1 output clock setting register 2	CPG_PLLDSI1_CLK2	Setting the frequency

4.4.5.3.3 Monitoring the PLL states

The internal registers of the CPG can be used to monitor the PLL state (reset or locked) per PLL.

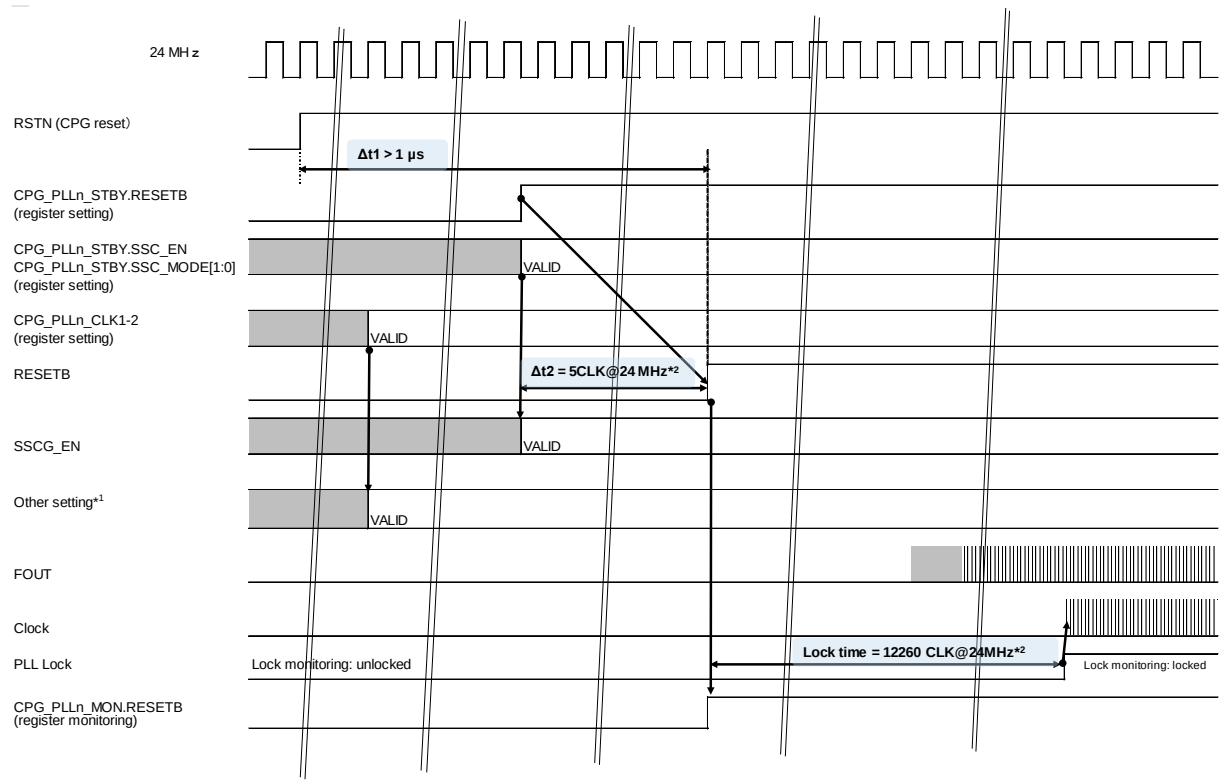
The related registers are listed in **Table 4.4-44**.

Table 4.4-44 List of PLL Monitor Registers

Register Name	Abbreviation	Function
PLLCM33 monitor register	CPG_PLLCM33_MON	Monitoring the PLL state
PLLCLN monitor register	CPG_PLLCLN_MON	Monitoring the PLL state
PLLDTY monitor register	CPG_PLLDTY_MON	Monitoring the PLL state
PLLCA55 monitor register	CPG_PLLCA55_MON	Monitoring the PLL state
PLLVDO monitor register	CPG_PLLVDO_MON	Monitoring the PLL state
PLLETH monitor register	CPG_PLLETH_MON	Monitoring the PLL state
PLLDSI0 monitor register	CPG_PLLDSI0_MON	Monitoring the PLL state
PLLDDR0 monitor register	CPG_PLLDDR0_MON	Monitoring the PLL state
PLLDSI1 monitor register	CPG_PLLDSI1_MON	Monitoring the PLL state

4.4.5.3.4 Timing setting

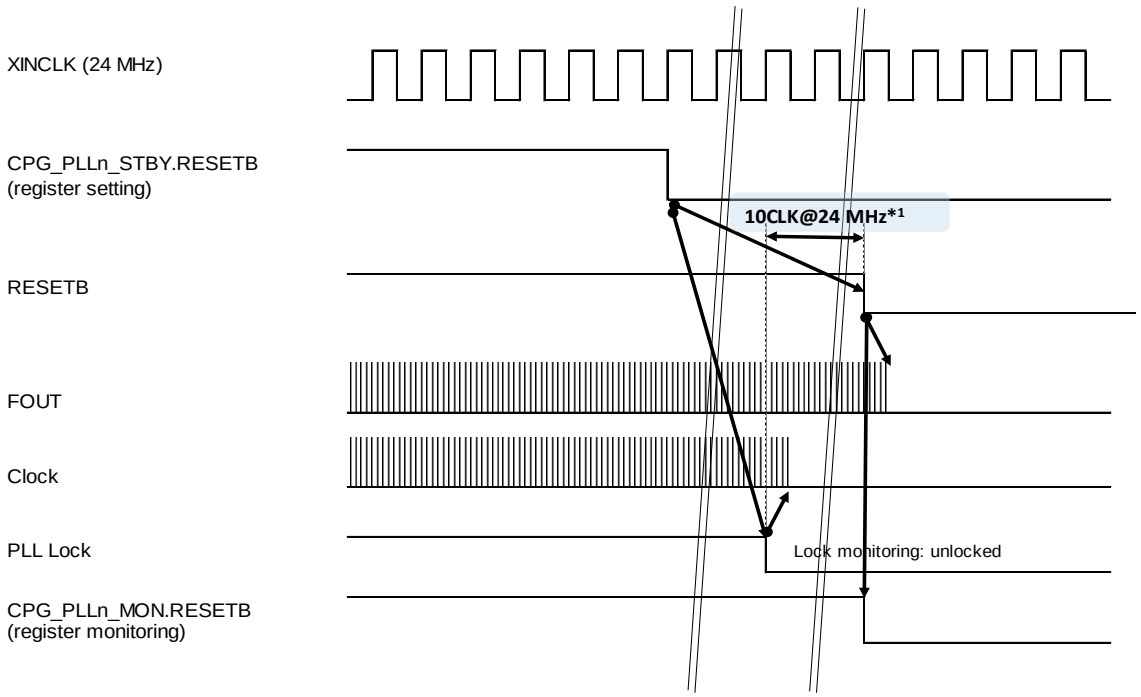
Figure 4.4-4 and **Figure 4.4-5** are timing charts in state transitions.



Note 1. Other setting: P, M, S, K

Note 2. As latency is incurred in processing by the synchronizer and sequencer in the actual circuit, the value is larger by one to five clock cycles.

Figure 4.4-4 Timing Chart in Transitions from Standby Mode to Normal Operating Mode of PLLs



Note 1. As latency is incurred in processing by the synchronizer and sequencer in the actual circuit, the value is larger by one to five clock cycles.

Figure 4.4-5 Timing Chart in Transitions from Normal Operating Mode to Standby Mode of PLLs

4.4.5.3.5 PLL settings

The initial settings of the individual PLLs are listed in the table below.

Table 4.4-45 PLL Initial Settings

PLL Name	K	m	p	s	Frequency (F_{Fout}) [MHz]	SSCG
PLLCM33	0	200	3	0	1600	Off*7
PLLCLN	0	200	3	0	1600	Off*7
PLLDTY	0	200	3	0	1600	On (off)*8
PLLCA55*1,*2	0	150	2	0	1800	On (off)*8
PLLCA55*1,*3	0	200	3	0	1600	On (off)*8
PLLCA55*1,*4	0	275	3	1	1100	On (off)*8
PLLCA55*1,*5 (These settings are the same as SYSREG initial values.)	0	250	2	1	1500	On (off)*8
PLLCA55*1,*6	−32768	213	3	0	1700	On (off)*8
PLLVDO	0	210	2	1	1260	On (off)*8
PLLETH	0	250	3	1	1000	Off*7
PLLDSI0*9	0	198	2	3	297	Off*7
PLLDSI0*10	0	203	2	1	1218	Off*7
PLLDDR0	0	200	3	1	800	Off
PLLDSI1*9	0	198	2	3	297	Off*7
PLLDSI1*11	0	203	2	2	609	Off*7

Note 1. The frequency setting for the PLLCA55 depends on the type of booting. For CM33 boot, the setting is not modified and the SYSREG initial value is retained.

Note 2. These settings are those when the PLLCA55 operates at 1.8 GHz after CM33 boot.

Note 3. When CA55 boot is selected and the levels of the BOOTPLLCA1 and BOOTPLLCA0 pins are high (1) and low (0), respectively,
or when CM33 boot is selected (the SYSREG initial value)

Note 4. When CA55 boot is selected and the levels of both BOOTPLLCA1 and BOOTPLLCA0 pins are low

Note 5. When CA55 boot is selected and the levels of the BOOTPLLCA1 and BOOTPLLCA0 pins are low and high, respectively

Note 6. When CA55 boot is selected and the levels of both BOOTPLLCA1 and BOOTPLLCA0 pins are high

Note 7. For PLLCM33, PLLCLN, PLLETH, PLLDSI0, PLLDSI1, SSCGs are fixed to off.

Note 8. The level on the external MD_CLKS pin in booting up determines the SSCG on/off settings for PLLDTY, PLLCA55, PLLVDO. In normal situations, the SSCGs are on.
— The level on MD_CLKS is high: SSCG is on.
— The level on MD_CLKS is low: SSCG is off.

Note 9. These are the SYSREG initial values for the PLLDSI0 and PLLDSI1.

Note 10. This is the setting value when the PLLDSI0 is operated at 1218 MHz.

Note 11. This is the setting value when the PLLDSI1 is operated at 609 MHz.

The table below lists the specifiable ranges of frequencies of the individual PLLs.

Table 4.4-46 Allowed range of PLL frequencies

PLL Name	Minimum Frequency (F_{Fout}) [MHz]	Maximum Frequency (F_{Fout}) [MHz]
PLLCM33 (fixed)	1600	1600
PLLCLN (fixed)	1600	1600
PLLDTY (fixed)	1600	1600
PLLCA55 (variable)	1100	1800
PLLVDO (fixed)	1260	1260
PLLETH (fixed)	1000	1000
PLLDSI0 (variable)	25	1218
PLLDDR0 (fixed)	800	800
PLLDSI1 (variable)	25	609

Note 1. When this clock is to be used as the source clock for CA55 SCLK, the frequency is fixed to 1260 MHz so making the setting is prohibited. When the clock is not to be used as the source clock, making the setting is possible (for up to 1260 MHz).

The PLL oscillation frequency (F_{FVCO}) and output frequency (F_{FOUT}) can be calculated by using the expressions below.

[Expressions]

$$F_{FVCO} = \frac{((m + k/65536) \times F_{Fin})}{p}$$

$$F_{Fout} = \frac{((m + k/65536) \times F_{Fin})}{(p \times 2^{s})}$$

[Limitations]

F_{Fin}: 24 MHz

6 MHz ≤ F_{FREF} = F_{FIN}/p ≤ 25 MHz

1,600 MHz ≤ F_{FVCO} ≤ 3,200 MHz

The modulation frequency (MF) and modulation ratio (pk-pk) (MR) can be calculated by using the expressions below.

[Expressions]

$$MF = \frac{(F_{Fin})}{p \times mfr \times (2^5)}$$

$$MR = \frac{mfr \times mrr}{m \times (2^6)} \times 100[\%]$$

[Limitations]

F_{Fin}: 24 MHz

$0 \leq mfr \leq 255$

$1 \leq mrr \leq 63$

$0 \leq mrr \times mfr \leq 512$

The following is an example of setting the modulation frequency (MF) and modulation ratio (pk-pk) (MR) of PLLDSI0/1.

Table 4.4-47 Configuration Example for Modulation Frequency (MF)

MF (kHz)	F_{fin} (MHz)	p	mfr
31.25	24	2	0Ch
62.50			06h
125.00			03h

Table 4.4-48 Configuration Example for Modulation Ratio (pk-pk) (MR)

MR (%)	mfr	m (MHz)	mrr
0.03	0Ch	609	01h
0.74			18h
1.29			2Ah
1.96		297	1Fh

4.4.5.4 Control Over Frequencies for the CA55

The CA55 initial states and how to control CORECLK, PERIPHCLK, and SCLK in each boot mode are described in **Table 4.4-49**.

In addition, the registers for use in control over frequencies for the CA55 are described in **Table 4.4-50**.

Table 4.4-49 Control over Frequencies for the CA55

Clock	Mode	PLL Output Frequency	Setting for Gearing in Stages after PLL
CORECLK	Initial state	In CA55 cold boot mode, the value set by hard coding according to the values of the BOOTPLLCA pins. In CM33 cold boot mode, the value following a reset.	In CA55 cold boot mode, the value determined by a setting in OTP memory. In CM33 cold boot mode, the value following a reset.
	CA55 cold boot	Can be set from CM33 if CA55 is in reset state. See **4.5.3.1.5 PD_CA55 area voltage change** for the procedure.	The setting is dynamically switchable.
	CM33 cold boot		The setting is dynamically switchable.
PERIPHCLK	Initial state	Same as the frequency of CORECLK (because the sources for the both clocks are the same PLL).	In CA55 cold boot mode, the value determined by a setting in OTP memory. In CM33 cold boot mode, the value following a reset.
	CA55 cold boot	Same as the frequency of CORECLK	The setting is dynamically switchable.
	CM33 cold boot	Same as the frequency of CORECLK	The setting is dynamically switchable.
SCLK	Initial state	In CA55 cold boot mode, any of the PLLCA55, PLLDTY, and PLLVDO is set as the clock source by using a setting in OTP memory. In CM33 cold boot mode, the value following a reset.	In CA55 cold boot mode, the value determined by a setting in OTP memory. In CM33 cold boot mode, the value following a reset.
	CA55 cold boot	The frequency can be modified under CM33 software control after the CM33 is booted up. In this case, if the PLLVDO is booted up under software control, changing the source to the PLLVDO is also possible. Note that doing so after booting the CA55 requires entry to the WFI state and Q-channel control.	The setting is dynamically switchable.
	CM33 cold boot	The frequency can be modified under CM33 software control both before and after the CA55 is booted up. In this case, if the PLLVDO is booted up under software control, changing the source to the PLLVDO is also possible. Note that doing so after booting the CA55 requires entry to the WFI state and Q-channel control.	The setting Is dynamically switchable.

Table 4.4-50 Registers for Use in Control over Frequencies for the CA55

Register	Description	Initial Value in CA55 Cold Boot Mode	Initial Value in CM33 Cold Boot Mode
PLLCA55 setting register 1	This register sets the parameters for the PLLCA55. 1500 MHz is set by the initial setting.	In CA55 cold boot mode, the PLL frequency setting corresponding to the levels on the BOOTPLLCA pins is captured as the initial value in the boot sequence.	In CM33 cold boot mode, the initial value following a reset is retained without updating it in the boot sequence.
PLLVDO setting register	This register sets the parameters for the PLLVDO. 1260 MHz is set by the initial setting.	The initial setting for 1260 MHz is retained.	The initial setting for 1260 MHz is retained.
SCLK source selection register • CPG_SSEL0[0] • CPG_SSEL0[4]	This register selects a source PLL for SCLK. The PLLCA55 (01b), PLLDTY (10b), or PLLVDO (00b) is selected. The value following a reset is for the PLLVDO (00b). This initial value is retained.	In CA55 cold boot mode, a setting in OTP memory is captured as the initial value in the boot sequence.*1	In CM33 cold boot mode, the initial value following a reset is retained without updating it in the boot sequence.
CORECLK gear control register • CPG_CDDIV1[1:0] (CORE0CLK) • CPG_CDDIV1[5:4] (CORE1CLK) • CPG_CDDIV1[9:8] (CORE2CLK) • CPG_CDDIV1[13:12] (CORE3CLK)	This register sets the gear division ratio of CORECLK. DIV1 (00b), DIV2 (01b), DIV4 (10b), or DIV8 (11b) is selected. The value following a reset is for DIV1 (00b).	In CA55 cold boot mode, a setting in OTP memory is captured as the initial value in the boot sequence.	In CM33 cold boot mode, the initial value following a reset is retained without updating it in the boot sequence.
PERIPHCLK gear control register • CPG_CDDIV2[1:0]	This register sets the gear division ratio of PERIPHCLK. DIV2 (00b), DIV4 (01b), DIV8 (10b), or DIV16 (11b) is selected. The value following a reset is for DIV2 (00b).	In CA55 cold boot mode, a setting in OTP memory is captured as the initial value in the boot sequence.	In CM33 cold boot mode, the initial value following a reset is retained without updating it in the boot sequence.
SCLK gear control register • CPG_CDDIV0[13:12]	This register sets the gear division ratio of SCLK. DIV1 (00b), DIV2 (01b), DIV4 (10b), or DIV8 (11b) is selected. The value following a reset is for DIV1 (00b).	In CA55 cold boot mode, a setting in OTP memory is captured as the initial value in the boot sequence.	In CM33 cold boot mode, the initial value following a reset is retained without updating it in the boot sequence.

Note 1. If no data are written to the OTP memory, the value corresponding to the state of the BOOTPLLCA pin is set. For details, see the section below.

4.4.5.4.1 Setting PLLCA55 by BOOTPLLCA pins

For details on the relationship between the BOOTPLLCA pins and PLLCA55 frequency settings, refer to **Table 4.4-51**. These pins must be finalized before unresetting. After the reset is released, these pins do not affect operation.

After releasing the reset, it is recommended to turn off the built-in pull-up and pull-down resistors with software. For external pull-up and pull-down resistors, 1 kΩ is recommended.

Table 4.4-51 Relationship between the BOOTPLLCA Pins and PLLCA55 Frequency Settings

BOOTPLLCA1 (PS2)	BOOTPLLCA0 (PS3)	CA55 CORECLK Frequency	CA55 SCLK Frequency
0	0	PLLCA55 1.1 GHz	PLLCA55 1.1 GHz
0	1	PLLCA55 1.5 GHz	PLLVDO 1.26 GHz
1	0	PLLCA55 1.6 GHz	PLLVDO 1.26 GHz
1	1	PLLCA55 1.7 GHz	PLLVDO 1.26 GHz

4.4.5.5 Clock Dividers and Selectors

The CPG uses several types of divider and selector to supply clock signals to suit the specifications of the individual units. The dividers and selectors are classified into two types in terms of the switching method: dynamic and static switching dividers and selectors. Clock switching by the former is glitch-free so dynamic switching is possible. On the other hand, as clock switching by the latter may lead to the generation of glitches, it proceeds after halting the clock supply to modules (static switching).

The types of divider and selector are described below.

- Fixed divider
- Dynamic switching selector
- Dynamic switching variable divider (output with 50% duty cycle at the time of frequency division by an even number)
- Static switching selector
- Static switching variable divider (output with 50% duty cycle at the time of frequency division by an even number)

For details on the specifications for connections between the dividers, selectors, and PLLs, refer to **4.4.5.1 Block Diagram of the Clock System**.

There are procedures for clock switching by dividers and selectors. Refer to **4.4.9.5 Procedure for Setting Frequency Division Ratios of Dynamic Switching Dividers** and **4.4.9.6 Procedure for Setting Static Switching Dividers and Selectors**.

4.4.5.5.1 Dynamic switching variable divider

Dynamic switching variable dividers can switch clock signals without the generation of glitches when the frequency division ratio settings are changed.

They can switch clock signals without halting the clock supply to units.

To avoid the generation of glitches in switching clock signals, clock switching takes certain time.

Figure 4.4-6 shows timing of clock switching by a dynamic switching variable divider.

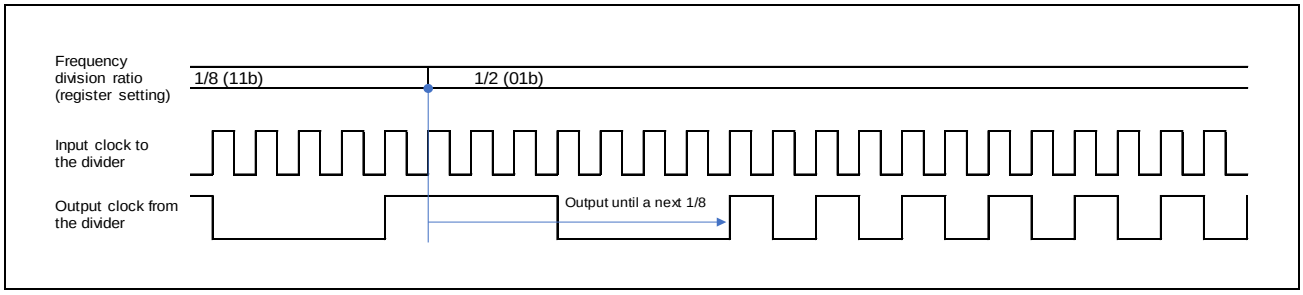


Figure 4.4-6 Timing of Clock Switching by the Dynamic Switching Variable Divider

4.4.5.5.2 Static switching selector

Clock output immediately reflecting the changed clock selection setting leads to the generation of glitches.

Therefore, to change the setting, halt the clock supply to units before doing so.

Figure 4.4-7 shows timing of clock switching by the selector.

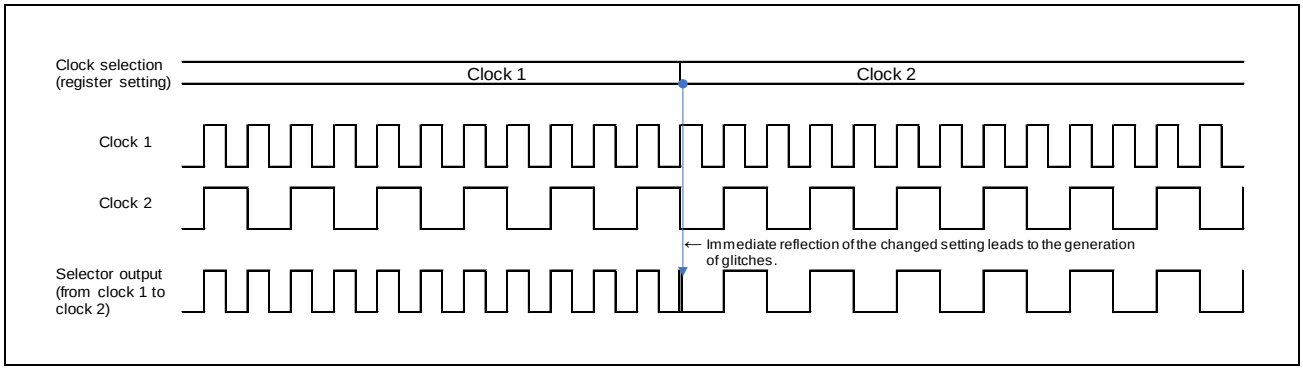


Figure 4.4-7 Timing of Clock Switching by the Static Switching Selector

4.4.5.5.3 Static switching variable divider

Static switching variable dividers immediately switch clock output in response to changing the frequency division ratio setting. This leads to the generation of a clock with the unintended pulse width (for frequency division by an odd number, the high-level pulse width is shorter by one cycle of the source clock than the low-level pulse width).

Therefore, to change the frequency division ratio setting, halt the units to which the given clock is being supplied before doing so.

Figure 4.4-8 shows timing of clock switching by the divider.

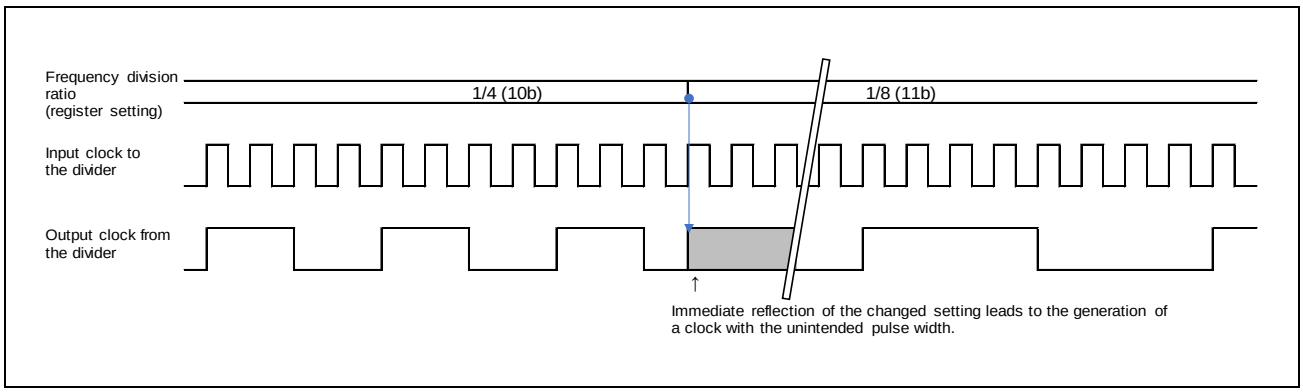


Figure 4.4-8 Timing of Clock Switching by the Static Switching Variable Divider

4.4.5.6 Temporarily Halting Clock Signals After Reset Signals are De-asserted

The CPG has a function for temporarily halting signals on the corresponding clock output pins after de-assertion of signals on the reset output pins.

For details on the correspondence between reset and clock pins, refer to **4.4.3 List of Reset Signals**.

The period over which the clock signals are temporarily halted is 64 clock cycles at 24 MHz in all (32 clock cycles at 24 MHz before and after the de-assertion of the reset signals).

4.4.6 Functions for the Generation and Control of Resets

4.4.6.1 Block Diagram of the Reset System

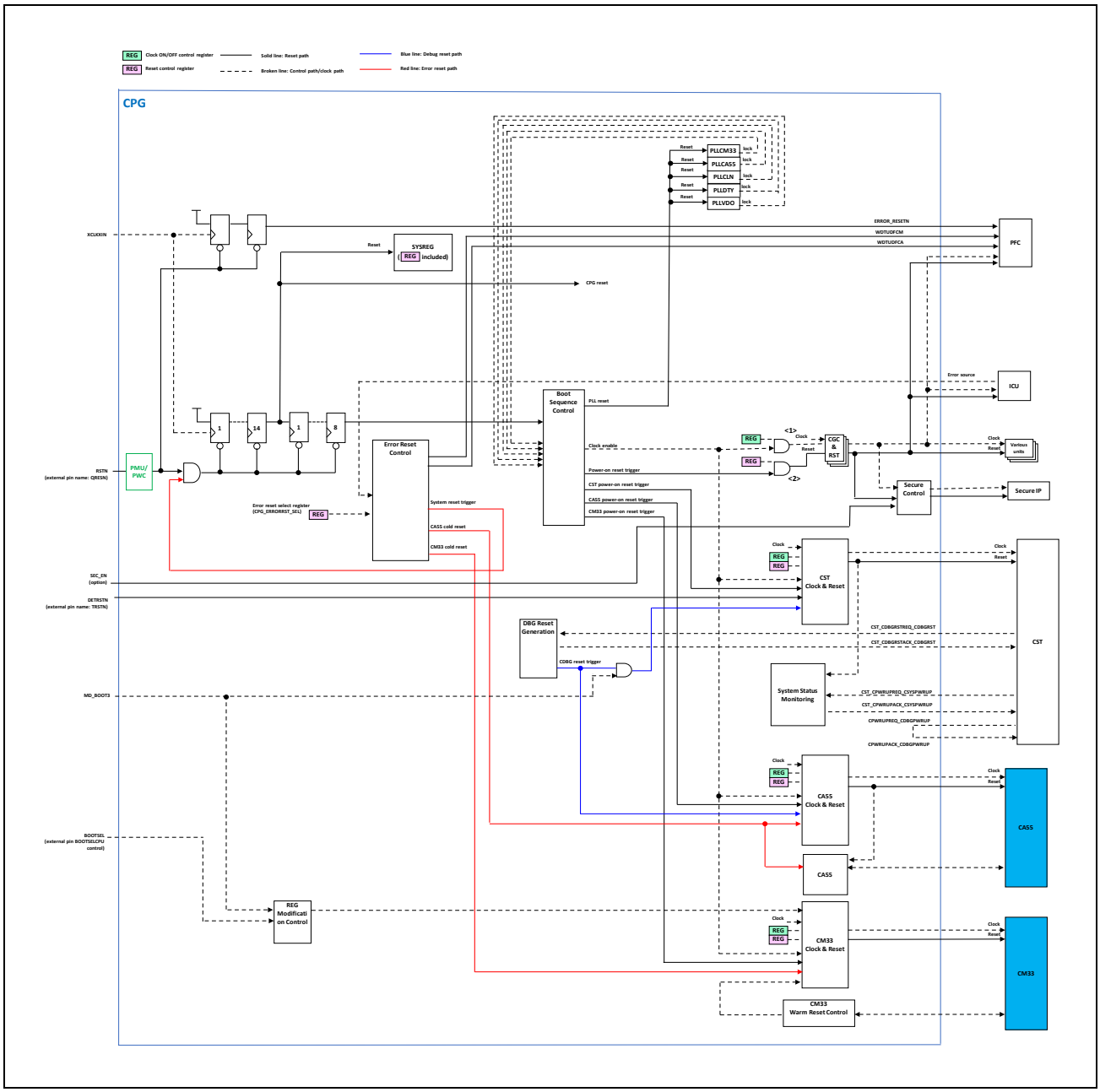


Figure 4.4-9 Block Diagram of the Reset System

4.4.6.2 Reset Application List

The scopes of application (targets) of generated resets according to the reset sources are listed below.

1) Hardware sources (QRESN, TRSTN, and MD_BOOT3)
2) Error reset sources
3) Debugger reset sources by the CST

Table 4.4-52 Reset Application List

No.	Operating Mode	Reset Initiation Source				Reset Applicability				Reset Content
		Debug Reset		System Reset						
		Hardware	Software	Hardware	Software					
	MD_BOOT3 pin	TRSTN pin	DBGRST_N*4	QRESN pin	Various error sources	CA55 (debugging related)	CST	Error Register*1	All Other Units (system reset)	
1	0	0	X	0	X	Reset	Reset	Reset	Reset	Hardware system reset
2	0	0	X	1	0	Reset*3	Reset	Outside the scope of reset	Reset*2	WDT system reset
3	0	0	X	1	1	Control by the corresponding register*5	Reset	Outside the scope of reset	Control by the corresponding register*5	Normal operating mode
4	0	1	X	0	X	Reset	Reset	Reset	Reset	Setting DEBUG = low and TRSTN = high is prohibited.
5	0	1	X	1	0	Reset*3	Reset	Outside the scope of reset	Reset*2	Setting DEBUG = low and TRSTN = high is prohibited.
6	0	1	X	1	1	Control by the corresponding register*5	Reset	Outside the scope of reset	Control by the corresponding register*5	Setting DEBUG = low and TRSTN = high is prohibited.
7	1	0	X	0	X	Reset	Reset	Reset	Reset	Hardware system reset
8	1	0	X	1	0	Reset*3	Reset*2	Outside the scope of reset	Reset*2	WDT system reset
9	1	0	X	1	1	Control by the corresponding register*5	Control by the corresponding register*5	Outside the scope of reset	Control by the corresponding register*5	Debug operating mode
10	1	1	X	0	X	Reset	Reset	Reset	Reset	Hardware system reset
11	1	1	0	1	0	Reset*3	Reset*2	Outside the scope of reset	Reset*2	WDT system reset
12	1	1	0	1	1	Reset	Reset	Outside the scope of reset	Control by the corresponding register*5	Software debugger reset
13	1	1	1	1	0	Reset*3	Reset*2	Outside the scope of reset	Reset*2	WDT system reset
14	1	1	1	1	1	Control by the corresponding register*5	Control by the corresponding register*5	Outside the scope of reset	Control by the corresponding register*5	Debug operating mode

Note: X: Don't care

Note: Setting reset combinations not listed in the table is prohibited.

Note 1. For details, see **4.4.4.15 Error Reset Registers (CPG_ERROR_RSTm) (m = 2 to 8)**.

Note 2. Applied when a system reset is enabled by the ERRORRST_SELm register. When it is disabled, "Control by the corresponding register*5" applies.

Note 3. Applied when a CA55 reset or system reset is enabled by the ERRORRST_SELm register. When it is disabled, "Control by the corresponding register*5" applies.

Note 4. For the CST software reset (DBGRST_N), see **4.4.6.6 Debug Reset (Software Reset by CoreSight)**.

Note 5. The reset is controlled by the “CPG_RST_m” register setting.

4.4.6.3 System Reset (External Pin)

Asserting the reset signal on the external QRESN pin initializes the whole LSI.

4.4.6.4 Switching Unit Reset On and Off

Switching reset signals for individual units on and off can be realized by switching a reset pin for the unit on and off according to enable conditions.

Sources of control for switching reset sources on and off are listed below.

- On and off control by a register setting
- Reset control in a boot sequence
- Debug reset control
- CA55 cold reset control
- CM33 cold reset control

4.4.6.5 Error Reset

This LSI has four channels of WDT. In the CPG, a reset request is issued in response to WDT sources as well as several error sources.

For the error sources and the hardware sequences which are initiated in response to the corresponding error source, see **Table 4.4-53**.

For details on the error sources, see **4.6 Interrupt Controller**.

Table 4.4-53 Error Sources and Hardware Sequences

Error Source and Hardware Sequence	CM33 Cold Reset*1	CA55 Cold Reset*1	Error System Reset*2	WDTUDFCM Assertion	WDTUDFCA Assertion
WDT source for CM33	✓		✓	✓	
WDT source for CA55 (all cores)*3		✓	✓		✓
WDT source for Other 0			✓		
WDT source for Other 1			✓		
Error source other than WDT			✓		

Note: Items containing the symbol "✓" can be masked individually.

Note 1. When an error system reset is executed, a cold reset is also applied to the CM33 and CA55.

Note 2. The CPG_ERROR_RSTm register is outside the scope of reset.

Note 3. For CA55, the whole CA55 is reset (resetting of individual cores is not possible).

4.4.6.5.1 Error reset circuit

A reset request from various error sources including those from the WDT can be arbitrarily assigned by the mask function through the CPG_ERRORRST_SELm register setting.

Setting the corresponding bit of CPG_ERRORRST_SEL to 1 asserts a reset request from various error sources including those from the WDT.

For details on the CPG_ERRORRST_SEL register, see **4.4.4.13 Error Reset Select Register 1 (CPG_ERRORRST_SEL1)** and **4.4.4.14 Error Reset Select Registers (CPG_ERRORRST_SELm) (m = 2 to 8)**.

4.4.6.5.2 WDTUDFCM pin

This pin provides external notification of the occurrence of an underflow or refresh error by the WDT (CM33).

Whether or not to provide this notification can be selected by using bit 0 of the ERRORRST_SEL1 register.

The initial level on this pin is high and the active level is low.

Assertion of the signal from another error system reset source is ignored until completion of the output of this pulse signal.

The reset sequence by the WDT (CM33) is as follows.

1. A reset request from the WDT corresponding to bit 0 of ERRORRST_SEL1 is generated.
2. The CPG issues an error system reset.
3. Wait for 64 cycles of the 24-MHz clock.
4. A low-level pulse waveform equivalent to 64 cycles of the 24-MHz clock is output from WDTUDFCM.
5. Wait for 64 cycles of the 24-MHz clock.
6. De-assert the error system reset signal and execute a boot sequence.

For the error system reset timing, see **Figure 4.4-10**.

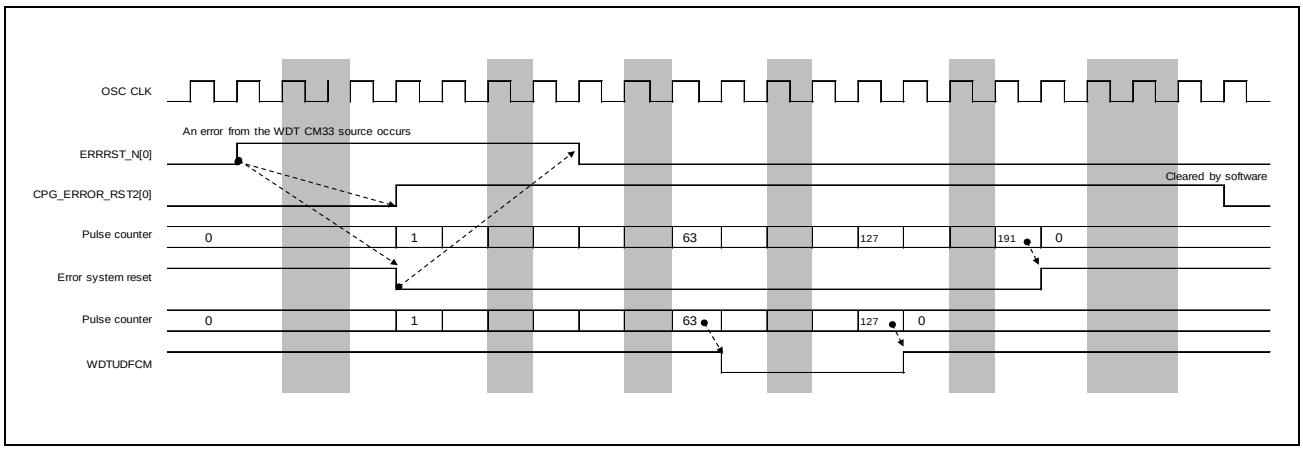


Figure 4.4-10 WDT CM33 Source Error System Reset Timing

4.4.6.5.3 WDTUDFCA pin

This pin provides external notification of the occurrence of an underflow or refresh error by the WDT (CA55).

Whether or not to provide this notification can be selected by using bit 1 of the ERRORRST_SEL1 register.

The initial level on this pin is high and the active level is low.

Assertion of the signal from another error system reset source is ignored until completion of the output of this pulse signal.

The reset sequence by the WDT (CA55) is as follows.

1. A reset request from the WDT corresponding to bit 1 of ERRORRST_SEL1 is generated.
2. The CPG issues an error system reset.
3. Wait for 64 cycles of the 24-MHz clock.
4. A low-level pulse waveform equivalent to 64 cycles of the 24-MHz clock is output from WDTUDFCA.
5. Wait for 64 cycles of the 24-MHz clock.
6. De-assert the error system reset and execute a boot sequence.

For the error system reset timing, see **Figure 4.4-11**.

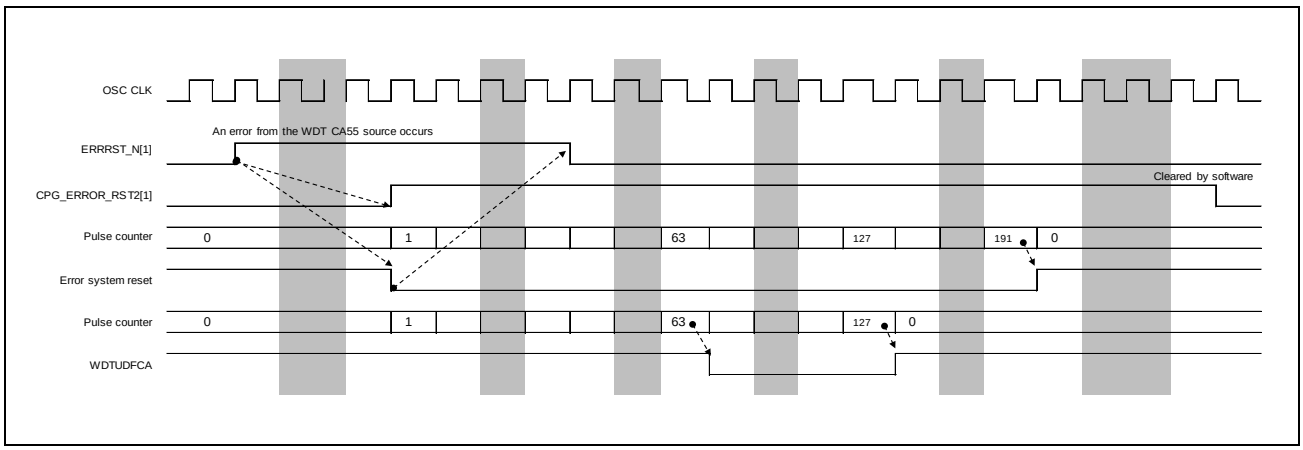


Figure 4.4-11 WDT CA55 Source Error System Reset Timing

4.4.6.5.4 Error reset registers (CPG_ERROR_RSTm)

These registers are for checking the type of error reset request which was a reset source after return from the reset, in the case of an error system reset having been applied by the error reset circuit in response to an error reset request, including that from the WDT.

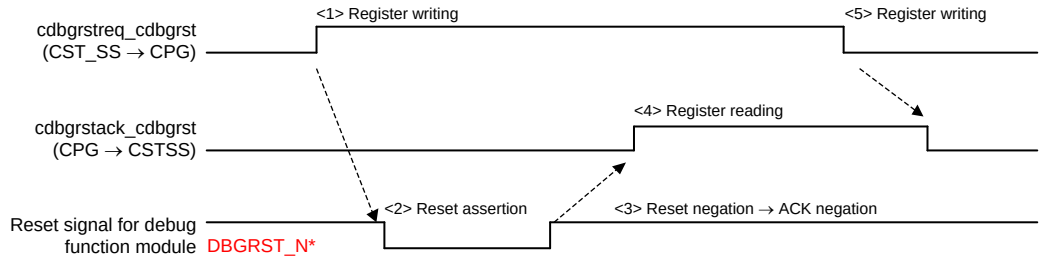
These registers are not reset by an error system reset generated by the error reset circuit.

Furthermore, the error source is not reset if the ERRRST_SELm register does not select application of a system reset, so the source can be checked by the interrupt from the corresponding error source (and the status register in each unit).

For details on these registers, see **4.4.4.15 Error Reset Registers (CPG_ERROR_RSTm) (m = 2 to 8)**.

4.4.6.6 Debug Reset (Software Reset by CoreSight)

An interface circuit with CoreSight (CST) is implemented in the CPG and a software reset of the debugging circuit of this LSI is applied from the debugger by executing the following handshake. This is only valid while QRESN = high, TRSTN = high, and MD_BOOT3 = high. The reset targets are the circuits for a debugger reset of the CST and CA55 (CM33 is outside the scope).



<1> Control the register in the CoreSight DAP from the debugger to assert the REQ signal (cdbgrstreq_cdbgrst).

<2> The CPG asserts the reset of the debug function module.

<3> The CPG negates the reset of the debug function module and asserts the ACK signal (cdbgrstack_cdbgrst). The CoreSight DAP captures the ACK assertion in the register.

<4> The debugger reads the register in the CoreSight DAP to detect the assertion of the ACK signal (that is, detect the completion of reset execution).

<5> The debugger controls the register in the CoreSight DAP to negate REQ.

<6> The CPG negates ACK and the CoreSight DAP captures the ACK negation in the register.

Figure 4.4-12 CoreSight Register Controlled Reset

4.4.6.7 CA55 Reset

The CA55 has several types of reset, differing in terms of hardware or software control and the procedure, and the CA55 reset signal through which control is applied.

Table 4.4-54 Reset Types of CA55

Reset Type		Trigger Source	Control Entity	Operation
(1a)	System Cold Reset (normal mode)	System reset Error system reset	CPG hardware	The CPG hardware asserts the reset signal. After that, the CPG hardware releases the reset through the procedure below. (1) The CPG hardware executes req and ack with P-channels. Though the req/ack handshake occurs in all channels, set PSTATE = ON only for the cluster and core 0 and set PSTATE = OFF for the other items (cores 1 to 3). (2) The CPG hardware releases the reset with the prescribed order and timing. Release the cores from the reset regardless of whether PSTATE = ON or OFF. (3) To start up the items other than core 0, set PSATE = ON by software.
(1b)	System Cold Reset (debug mode)	System reset Error system reset	CPG hardware	Same as (1a) with the exception of the following.
(1c)	Error Cold Reset	CA55 cold reset by WDT	CPG hardware	Same as (1a) with the exception of the following. PSTATE of cores 1 to 3 is in accord with the register setting.
(2a)	Software Cold Reset (at a CM33 cold boot)	Software	Software control by CM33	Assert or de-assert the reset signal by setting the PSTATE register and reset ON/OFF register in the CPG by software. The CPU hardware executes req and ack with P-channels. The req/ack handshake occurs in all channels.
(2b)	Software Cold Reset (after startup of CA55 at a CM33 boot (2a))	Software	Software control by CM33	Assert or de-assert the reset signal by setting the P-channel related register and reset ON/OFF register in the CPG by software.
(3)	Cluster Warm Reset	Software	Software control by CM33	Assert or de-assert the reset signal by setting the P-channel related register and reset ON/OFF register in the CPG by software.
(4)	Core-0 Warm Reset	Software	Software control by CM33	
(5)	Core-1 Warm Reset	Software	Software control by CM33	
(6)	Core-2 Warm Reset	Software	Software control by CM33	
(7)	Core-3 Warm Reset	Software	Software control by CM33	

Note: At the time of a CA55 boot, the CA55 is initialized by (1a) or (1b) before startup.
At the time of a CM33 boot, the CA55 is initialized by (2a) before startup. PD_CA55 to which the CA55 belongs is switched off in the initial state and initialization by (1a) or (1b) is not applied.

[Conditions for reset control by hardware]
- CA55 reset: (1a), (1b), (1c)
- Reset by CA55 P-channel control: (1a), (1b), (1c), (2a)

4.4.6.7.1 CA55 system cold reset (normal mode)

At the time of a system reset from the external pin or an error system reset, the CPG hardware asserts the reset signal. After that, the CPG hardware releases the reset through the procedure below.

1. The CPG hardware executes req and ack with P-channels. Though the req/ack handshake occurs in all channels, set PSTATE = ON only for the cluster and core 0 and set PSTATE = OFF for the other items (cores 1 to 3).
2. The CPG hardware releases the reset with the prescribed order and timing. Release the cores from the reset regardless of whether PSTATE = ON or OFF.
3. To start up the items other than core 0, set PSATE = ON by software, assert PREQ, check PACCEPT, and negate PREQ (handshaking by software).

4.4.6.7.2 CA55 system cold reset (debug mode)

Startup in debug mode is the same as a cold reset of the CA55 system (normal mode).

4.4.6.7.3 CA55 error cold reset

In this mode, the CPG applies a cold reset to a single CA55 by hardware control in response to the WDT CA55 source.

4.4.6.7.4 CA55 software cold reset

A cold reset is applied to a single CA55 by setting the P-channel related register and reset ON/OFF register in the CPG through software control by the CM33. For the resetting procedure, see **2.2 CPU**.

4.4.6.7.5 CA55 software warm reset

A warm reset is applied to a single CA55 by setting the P-channel related register and reset ON/OFF register in the CPG through software control by the CM33. For the resetting procedure, see **2.2 CPU**.

4.4.6.7.6 CA55 reset timing constraints

A CA55 reset is subject to the timing constraint related to the order of release of the reset and the constraint on the minimum reset pulse width.

(1) CA55 reset release timing

The interval between the reset (A) and (B) groups requires at least 64 clock cycles at 24 MHz. At the time of software control, software must satisfy the constraint.

(A) group: CA55_RESET14, CA55_RESET15, CA55_RESET16

(B) group: CA55 reset signals other than the above

(2) CA55 reset pulse width

The reset pulse width for the CA55 requires at least 128 clock cycles at 24 MHz. At the time of software control, software must satisfy the constraint.

4.4.6.8 CM33 Reset

The CM33 has several types of reset, differing in terms of hardware or software control and the procedure, and the CM33 reset signal through which control is applied.

Table 4.4-55 Reset Types of CM33

Reset Type		Trigger Source	Control Entity	Operation
(1a)	System Cold Reset	System reset Error system reset	CPG hardware	The CPG hardware handles the following. (1) Assert CM33_RESET0, CM33_RESET1, and CM33_RESET2. (2) De-assert CM33_RESET2 after 64 cycles (24 MHz) following the assertion of CM33_RESET2. (3) De-assert CM33_RESET0 after 64 cycles (24 MHz) following the de-assertion of CM33_RESET2.
(1b)	Error Cold Reset	CM33 cold reset by WDT	CPG hardware	Same as (1a)
(2)	SW Cold Reset	Software	Software control by CA55	Assert or de-assert the reset signal by setting the reset ON/OFF register in the CPG by software.
(3)	Warm Reset	Software	Software control by CM33 and CPG hardware	The software of CM33 handles the following. (S1) Setting to mask interrupts to the CM33. (S2) Instruction for the CM33 to enter WFI. The CM33 enters WFI, so control is transferred to the CPG hardware after that. (H1) REQ/ACK with the CM33 (H2) Assert the reset of the CM33. (H3) Release masking of interrupts to the CM33. (H4) De-assert the reset of the CM33 after the prescribed period for assertion of the reset is satisfied regardless of the generation of an interrupt.

Note: At the time of a CA55 boot, the CM33 is not started up in the initial state. The CM33 is initialized by (2) before startup.
At the time of a CM33 boot, the CM33 is initialized by (1a) or (1b) before startup.

[Conditions for reset control by hardware]
- (1a), (1b)

4.4.6.8.1 CM33 system cold reset

At the time of a system reset from the external pin or an error system reset, the CPG hardware asserts the reset signal. After that, the CPG hardware releases the reset through the procedure below. Normal mode and debug mode are the same procedure.

1. The CPG hardware releases the reset in the prescribed order.

4.4.6.8.2 CM33 error cold reset

In this mode, the CPG applies a cold reset to a single CM33 by hardware control in response to the WDT CM33 source.

4.4.6.8.3 CM33 software cold reset

A cold reset is applied to a single CM33 by setting the reset ON/OFF register in the CPG through software control by the CA55. For the resetting procedure, see **2.2 CPU**.

4.4.6.8.4 CM33 warm reset

When the CA55 applies a warm reset to the CM33, a handshake is executed between the CPG and the CM33. The reset request signal CM33_RESET1 which is issued following the handshake generates 128 cycles of the pulse at 24 MHz.

The conditions for asserting CM33_RESET1 are CM33_WARMRESETREQ = 1b and CM33_LP_CTL0 = 1b.

The INTMASK_CM33_I bit in the CPG_LP_CTL2 register is for masking interrupts from the ICU to the CM33. The CPG_LP_CTL2 register is set by software before the CM33 enters the warm reset state. It is cleared automatically by hardware.

4.4.6.8.5 CM33 reset timing constraints

A CM33 reset is subject to the timing constraint related to the order of release of the reset and the constraint on the minimum reset pulse width.

(1) CM33 reset release timing

The interval between the reset (A) and (B) groups requires at least 64 clock cycles at 24 MHz. At the time of software control, software must satisfy the constraint.

(A) group: CM33_RESET2

(B) group: CM33_RESET0, CM33_RESET1

(2) CM33 reset pulse width

The reset pulse width for the CM33 requires at least 64 clock cycles at 24 MHz. At the time of software control, software must satisfy the constraint.

4.4.7 Control of Booting

The CPG has the following four types of boot mode. For details, see **1.9 Boot Operation**.

1) CM33 normal boot

The LSI starts up in normal mode. The CPU that starts up at the time of booting is the CM33.

The power domain of PD_AWO is only switched on, and PD_OTHERS and PD_CA55 are switched off.

The SRAM rescue bit is automatically transferred to the PD_AWO domain, but transfer to the PD_OTHERS and PD_CA55 domains requires performing the steps of case 2 in **4.4.7.1 SRAM Rescue Circuit Control** by software after switching the power of the PD_OTHERS and PD_CA55 domains on.

2) CM33 debug boot

The LSI starts up in debug mode. The CPU that starts up at the time of booting is the CM33.

The power domain of PD_AWO is only switched on, and PD_OTHERS and PD_CA55 are switched off.

The SRAM rescue bit is automatically transferred to the PD_AWO domain, but transfer to the PD_OTHERS and PD_CA55 domains requires performing the steps of case 2 in **4.4.7.1 SRAM Rescue Circuit Control** by software after switching the power of the PD_OTHERS and PD_CA55 domains on.

3) CA55 normal boot

The LSI starts up in normal mode. The CPU that starts up at the time of booting is CA55 core 0.

All of the power domains (PD_AWO, PD_OTHERS, and PD_CA55) are switched on.

The SRAM rescue bit is automatically transferred.

4) CA55 debug boot

The LSI starts up in debug mode. The CPU that starts up at the time of booting is CA55 core 0.

All of the power domains (PD_AWO, PD_OTHERS, and PD_CA55) are switched on.

The SRAM rescue bit is automatically transferred.

Remark 1. The CA55 is started up at the time of a CM33 boot by software.

Remark 2. The CM33 is started up at the time of a CA55 boot by software.

4.4.7.1 SRAM Rescue Circuit Control

This LSI has functionality to read the SRAM rescue bit information from the OTP memory and transfer it to the SRAM in the set power domain.

This function is used when switching PD_OTHERS on. For the operating procedure, see **4.5.3.1 Power Domain Control** in **4.5 PMU**.

4.4.7.2 CM33 Boot (Normal Mode)

The CM33 boot (normal mode) is described.

In either case of a CA55 or CM33 boot, the boot sequencer starts up when the external QRESN pin or the error system reset is released. After startup, the CA55/CM33 boot sequence is applied in accord with the value from the external BOOTSELCPU pin. In this LSI, a CM33 boot is initiated when BOOTSELCPU = low.

When MD_BOOT3 = low, a boot is initiated in normal mode.

In a CM33 boot, only the PD_AWO power domain is powered on before the boot sequence starts up.

The units which start up after booting are as follows:

- SRAM0
- SYSTEM BUS (PD_AWO)
- CM33
- DMAC0
- WDT0
- PFC
- SYS
- ICU
- ROM

4.4.7.3 CM33 Boot (Debug Mode)

The CM33 boot (debug mode) is described.

In either case of a CA55 or CM33 boot, the boot sequencer starts up when the external QRESN pin or the error system reset is released. After startup, the CA55/CM33 boot sequence is applied in accord with the value from the external BOOTSELCPU pin. In this LSI, a CM33 boot is initiated when BOOTSELCPU = low.

When MD_BOOT3 = high, a boot is initiated in debug mode.

In a CM33 boot, only the PD_AWO power domain is powered on before the boot sequence starts up.

The units which start up after booting are as follows:

- SRAM0
- SYSTEM BUS (PD_AWO)
- CM33
- DMAC0
- WDT0
- PFC
- SYS
- ICU
- ROM
- CST

4.4.7.4 CA55 Boot (Normal Mode)

The CA55 boot (normal mode) is described.

In either case of a CA55 or CM33 boot, the boot sequencer starts up when the external QRESN pin or the error system reset is released. After startup, the CA55/CM33 boot sequence is applied in accord with the value from the external BOOTSELCPU pin. In this LSI, a CA55 boot is initiated when BOOTSELCPU = high.

Similarly, the normal mode/debug mode boot sequence is applied in accord with the value from MD_BOOT3. When MD_BOOT3 = low, a boot is initiated in normal mode.

In a CA55 boot, all of the power domains (PD_AWO, PD_OTHERS, and PD_CA55 power domains) are powered on before the boot sequence starts up.

The debugging system unit (CST) is not released from the reset and no clock signal is supplied to this unit.

The units which start up after booting are as follows:

- SRAM0, SRAM1
- GIC
- SYC
- SYSTEM BUS
- CA55
- PFC
- SYS
- ICU
- ROM

4.4.7.5 CA55 Boot (Debug Mode)

The CA55 boot (debug mode) is described.

In either case of a CA55 or CM33 boot, the boot sequencer starts up when the external QRESN pin or the error system reset is released. After startup, the CA55/CM33 boot sequence is applied in accord with the value from the external BOOTSELCPU pin. In this LSI, a CA55 boot is initiated when BOOTSELCPU = high.

Similarly, the normal mode/debug mode boot sequence is applied in accord with the value from MD_BOOT3. When MD_BOOT3 = high, a boot is initiated in debug mode.

In a CA55 boot, all of the power domains (PD_AWO, PD_OTHERS, and PD_CA55 power domains) are powered on before the boot sequence starts up.

The debugging system units (CST and CA55 debug) are released from the reset and clock signals are supplied to these units.

The units which start up after booting are as follows:

- SRAM0, SRAM1
- GIC
- SYC
- SYSTEM BUS
- CA55
- PFC
- SYS
- ICU
- ROM
- CST

4.4.8 Control over Lower Power Consumption

Table 4.4-56 lists the low-power consumption modes of this LSI. In CM33 cold boot, there are two low-power modes, with or without power shutdown, whereas in CA55 cold boot, power shutdown is not performed.

Table 4.4-56 Low-Power Consumption Mode and State of Respective CPUs

Cold Boot	Power Mode	Low-Power Consumption Mode	State			Overview of Functions	Recovery Method
			CM33	CA55	Peripheral Unit		
CM33 cold boot	ALL_ON	Sleep_mode (CM33)	WFI (deep standby)/WFE	—	—	CM33 stopped by WFI (deep standby)/WFE	Interrupt*1/ SysTick timer
		Sleep_mode (CA55)	—	WFI/WFE	—	Stop CA55 by WFI/WFE	Interrupt*2
		Module_standby	—	—	Register control by CPG	Stop the function (clock) of the specified peripheral unit by CPG	Register control by CPG
		Low_frequency	Register control by CPG			Reduce the clock supplied to the CPU and peripheral units	Register control by CPG
	AWO	Sleep_mode (CM33)	WFI (deep standby)/WFE	(Power off)	—	Stop CM33 by WFI (deep standby)/WFE	Interrupt*1/ SysTick Timer
		Module_standby	—	(Power off)	Register control by CPG	Stop the function (clock) of the specified peripheral unit by CPG	Register control by CPG
		Low_frequency	Register control by CPG			Reduce the clock supplied to the CPU and peripheral units	Register control by CPG
		Software_standby	WIC standby	(Power off)	Clock control + MSTOP	CM33 performs Q-channel control and further power reduction.	Interrupt*1
CA55 cold boot	ALL_ON (without power shutdown)	Sleep_mode (CM33)	WFI (deep standby)/WFE	—	—	Stop CM33 by WFI (deep standby)/WFE	Interrupt*1/ SysTick timer
		Sleep_mode (CA55)	—	WFI/WFE	—	Stop CA55 by WFI/WFE	Interrupt*2
		Module_standby	—	—	Register control by CPG	Stop the function (clock) of the specified peripheral unit by CPG	Register control by CPG
		Low_frequency	Register control by CPG			Reduce the clock supplied to the CPU and peripheral units	Register control by CPG

Note 1. To return from an interrupt, use the NVIC in the CM33. Only interrupts for recovery must be enabled.

Note 2. The recovery from an interrupt is done via an interrupt input to the ICU. Only interrupts for recovery must be enabled.

4.4.8.1 State Transition for Each Mode

Table 4.4-15 shows the LP mode state transition when the CA55 is booted, and **Figure 4.4-14** shows the LP mode state transition when the CM33 is booted.

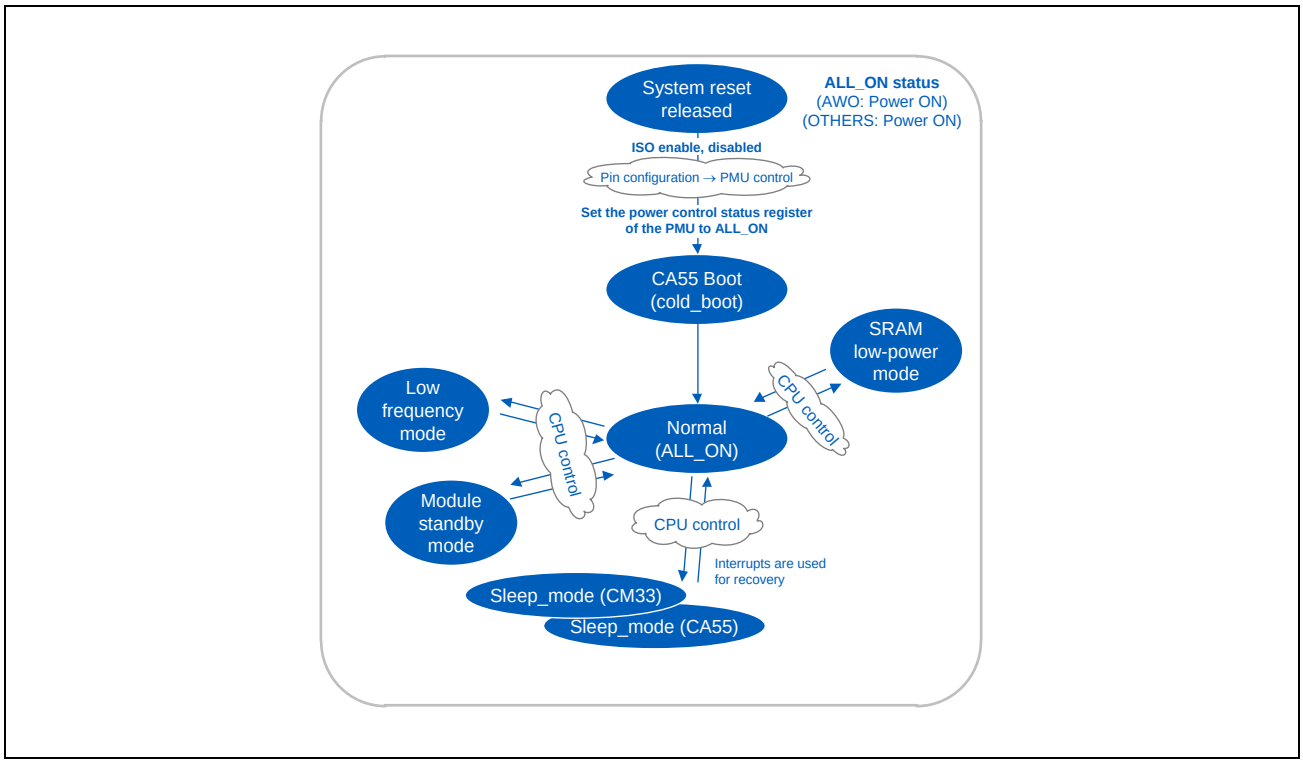


Figure 4.4-13 LP Mode Transition Diagram (CA55 Boot)

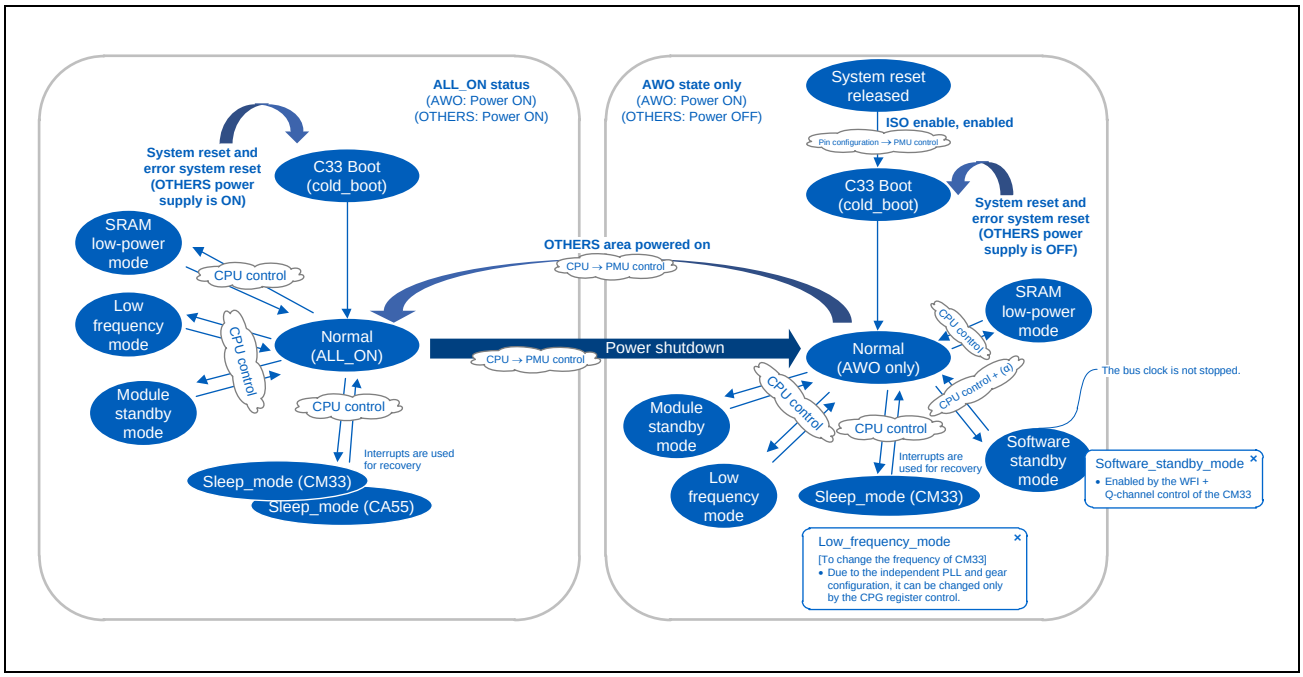


Figure 4.4-14 LP Mode Transition Diagram (CM33 Boot)

4.4.8.2 Power Mode Control

There are two power modes for CM33 cold boot: AWO mode and ALL_ON mode.

The CM33 is in AWO mode immediately after startup, and can be switched between AWO mode and ALL_ON mode after the CM33 has been started. For details on controlling, refer to **4.5 Power Management Unit (PMU)**.

4.4.8.3 Low-Frequency Mode

4.4.8.3.1 Overview

In this mode, the frequency of the clock signals supplied to specified peripheral units is lowered to obtain lower power consumption.

This is achieved by having software control registers of the CPG to switch the frequency divisors for the various domains.

See **4.4.2 List of Clock Signals**, to see which units belong to which domains and have which gears.

4.4.8.3.2 Procedure for transition to the mode

1. **Software control:** Stop units that must not have dynamic changes to the clock frequency*[1].
2. **Software control:** Control the dynamic and static gear control registers (CPG_CDDIVm) and (CPG_CSDIVm) to change the clock signals to lower frequencies.
3. **Software control:** Check the following bit in the dynamic gear status monitor register (CPG_CLKSTATUS0).

Note 1. Refer to the gearing type in list of clock signals (separate Excel file) .

4.4.8.3.3 Procedure for return from the mode

1. **Software control:** Control the dynamic and static gear control registers (CPG_CDDIVm) and (CPG_CSDIVm) to change the clock signals to normal frequencies.
2. **Software control:** Check the following bit in the dynamic gear status monitor register (CPG_CLKSTATUS0).
3. **Software control:** Resume units that must not have dynamic changes to the clock frequency.

4.4.8.4 Module Standby Mode

4.4.8.4.1 Overview

In this mode, the frequency of the clock signals supplied to specified peripheral units is stopped to obtain lower power consumption.

This is achieved by stopping supply of the clock signals by the CPG and, through the given CPG registers, switching the bus MSTOP ports. Error interrupts notify the system of attempted access to units in the MSTOP state.

The CM33 and CA55 are outside the scope of this mode. The sleep mode is used to lower power consumption by the various CPUs and is described in a later section.

4.4.8.4.2 Procedure for transition to the mode

1. **Software control:** In cases where processing must be constant so cannot be stopped until its completion, confirm that processing has definitely been completed.
2. **Software control:** For a unit that requires a procedure for stopping the clock, proceed with the sequence for stopping the clock.
3. **Software control:** Make the MSTOP settings for the target units by controlling the MSTOP control registers (CPG_BUS_m_MSTOP) of the CPG.
4. **Software control:** Stop the clocks for the target units by controlling the clock control registers (CPG_CLKON_m) of the CPG.

4.4.8.4.3 Procedure for return from the mode

1. **Software control:** Supply the clocks for the target units by controlling the clock control registers (CPG_CLKON_m) of the CPG.
2. **Software control:** Clear the MSTOP settings for the target units by controlling the MSTOP control registers (CPG_BUS_m_MSTOP) of the CPG.

4.4.8.4.4 Points to note

Depending on the unit, suddenly stopping the clock may not be possible. For example, the CM33 incorporates an interrupt controller, so abruptly stopping the clock may lead to interrupts becoming non-acceptable.

Particularly, confirmation is essential for units that have a strong possibility of restrictions applying because they can serve as masters, have an AXI bus, and in some other cases.

The items listed in the table below can be given as examples.

Table 4.4-57 Units with Restrictions Applying at the Time of Stopping

Unit Name	Stopping the Clock
CM33	It cannot be stopped.
CA55	It cannot be stopped. If power is to be reduced, cut off the power supply or if faster recovery to operation is a priority, place the core in the WFI state. (The registers for Q-channel control are in the CPG.)
GE3D	Use software control of the Q-channel control register.
CRU	Use software control of the unit's registers.
SYS	It cannot be stopped.
Internal bus	It cannot be stopped.
TZC	It cannot be stopped.

4.4.8.5 SRAM Power-Saving Mode

4.4.8.5.1 Overview

These modes consist of selective precharge mode, power down mode, and retention 2 mode, and are for reducing power consumption by placing the SRAM in the given mode. At these times, stopping and starting of the clock signals for each SRAM are controlled at the same time.

Software control of transitions to and from the SRAM low-power modes is applied through the CPG_LP_SRAM_STBY_CTLm registers of the CPG.

The modes are applicable to the 2 shared SRAM areas and to the unit-internal SRAM areas. All SRAM areas are listed on the following lines.

- (Shared SRAM area) SRAM0
- (Shared SRAM area) SRAM1
- (Unit-internal SRAM area) CANFD-internal SRAM area
- (Unit-internal SRAM area) VCD-internal SRAM area
- (Unit-internal SRAM area) GE3D-internal SRAM area
- (Unit-internal SRAM area) VSPI-internal SRAM area
- (Unit-internal SRAM area) FDP1-internal SRAM area

Control of both the shared and unit-internal SRAM areas is applied per SRAM area.

The standby control signal for the unit-internal SRAM areas is chain-connected between the areas. The standby state can be confirmed by monitoring the standby state signal (RSO) of the last SRAM area in the chain through the CPG register.

Control for the unit-internal SRAM areas can be applied by manipulating the signals on the SDM and RS pins and monitoring the signals on the RSO pins.

Control for the shared SRAM areas can be applied by manipulating the signals on the CENMASK, SDM, and RS pins and monitoring the signals on the RSO pins.

4.4.8.5.2 Procedure for transition to the mode

1. **Software control:** Confirm that operation of the unit that has or units that have access to the target SRAM area has been completed.
2. **Software control:** The CM33 controls the CPG registers (CPG_CLKON_m) and stops the target SRAM clocks.
3. **Software control:**
   (For shared SRAM areas) The CM33 controls the CPG registers (CPG_LP_SRAM_STBY_CTLm) and sets the CENMASK bit to 1b.
4. **Software control:**
   (For shared SRAM areas) The SDM bit is set to 1b in the case of power down and 0b in the case of retention 2.
   (For unit-internal SRAM areas) The SDM bit is set to 1b.
5. **Software control:** The CM33 controls the CPG registers and sets the RS bit of the target SRAM area to 1b.
6. **Software control:** The CM33 controls the CPG registers and confirms that the RSO bit of the target SRAM area is set to 1b.

4.4.8.5.3 Procedure for return from the mode

1. **Software control:** The CM33 controls the CPG registers (CPG_LP_SRAM_STBY_CTLm) and sets the RS bit of the target SRAM area to 0b.
2. **Software control:** The CM33 controls the CPG registers and sets the SDM bit of the target SRAM area to 0b.
3. **Software control:** (For shared SRAM areas) The CM33 controls the CPG registers and sets the CENMASK bit to 0b.
4. **Software control:** The CM33 controls the CPG registers and confirms that the RSO bit of the target SRAM area has been 0b.
5. **Software control:** The CM33 controls the CPG registers (CPG_CLKON_m) and supplies the target SRAM clocks.

4.4.8.6 CM33 Sleep Mode

4.4.8.6.1 Overview

This mode is for stopping the CM33. The transition to this mode proceeds in response to a WFI or WFE instruction having been issued for the CM33. The deep standby mode of the CM33 is used for this transition. Although most clock supply stops in this mode because the clock within the CM33 core is stopped, a clock at low speed is supplied for the 24-bit counter of the SysTick timer and as the NVIC clock for use in release.

4.4.8.6.2 Procedure for transition to the mode

1. **Software control:** The CM33 controls the CPG register (CPG_LP_CTL1) and sets the CM33SLEEP_REQ bit to 1b.
2. **Software control:** The CPG register (CPG_LP_CTL2) of the CM33 is controlled and the interrupt mask flag of the ICU is set so that interrupts cannot enter during the transition.
3. **Software control:** Registers of the NVIC within the CM33 are controlled and the interrupt mask of the source for return from the mode is removed. If the SysTick timer is to cause return of the CM33 from the mode, set the value for counting.
4. **Software control:** Set the SCR.SLEEPDEEP bit inside CM33 to 1.
5. **Software control:** The CM33 controls the CPG registers and decrease the speed of the CM33_CLK0 frequency of the CM33 as required.
6. **Software control:** Execute a WFI or WFE instruction as a CM33 instruction.

4.4.8.6.3 Procedure for return from the mode

1. **Software control:** After interrupt processing is completed, all interrupt factors from sources external to the LSI chip for the various units and within the ICU are cleared.
2. **Software control:** (When the CM33_CLK0 frequency is decreased at the time of a transition) Increase the speed of the CM33_CLK0 frequency of the CM33 by controlling the registers of the CPG.
3. **Software control:** The CM33 controls the CPG register (CPG_LP_CTL1) and clear a CM33SLEEP_REQ or CM33SLEEP_ACK flag.

4.4.8.7 CA55 Sleep Mode

4.4.8.7.1 Overview

This mode is for stopping operations of the CA55 cores. The transition to this mode proceeds in response to a WFI or WFE instruction having been issued for the CA55.

4.4.8.7.2 Procedure for transition to the mode

1. **Software control:** The CA55 controls the CPG register (CPG_LP_CTL1) and set the CA55SLEEP_REQ bit to 1.
2. **Software control:** The CA55 controls the GIC registers and the interrupt mask of the source for return from the mode is removed.
3. **Software control:** Execute the WFI or WFE instruction as a CA55 instruction.

4.4.8.7.3 Procedure for return from the mode

1. **Software control:** After interrupt processing is completed, all interrupt factors from sources external to the LSI chip for the various units and within the ICU are cleared.
2. **Software control:** The CA55 controls the CPG register (CPG_LP_CTL1) and clear a CA55SLEEP_REQ or CA55SLEEP_ACK flag.

4.4.8.8 Software Standby Mode (only for the CM33 Cold Boot)

4.4.8.8.1 Overview

This mode uses the CM33's WIC standby function (for WIC standby function, see the Arm® Cortex®-M33 Processor Technical Reference Manual), and enables lower power consumption than that for the sleep mode.

This mode is valid only when the CM33 is cold booted and the power mode is the AWO mode (AWO power area = ON, OTHERS power area = OFF, and CA55 OD power area = OFF).

In this mode, the unit clock is not stopped by hardware. To stop the unit clock, it should be stopped by software before shifting to this mode.

4.4.8.8.2 Procedure for transition to the mode

1. **Software control:** The CM33 controls the CPG register (CPG_LP_CTL1) and sets STBY to 1.
2. **Software control:** The CPG register (CPG_LP_CTL2) of the CM33 is controlled and the interrupt mask flag of the ICU is set so that interrupts cannot enter during the transition.
3. **Software control:** Registers of the NVIC within the CM33 are controlled and the interrupt mask of the source for return from the mode is removed.
4. **Software control:** Set the SCR.SLEEPDEEP bit inside CM33 to 1.
5. **Software control:** The CM33 controls the CPG register (CPG_LP_CM33_CTL0) and sets CM33_LP_CTL24 to 1.
6. **Software control:** The CM33 reads the CPG register (CPG_LP_CM33_CTL0) and polls it until CM33_LP_CTL25 = 1.

<Steps 7 to 10 are executed only when the FPU is used>

7. **Software control:** The CM33 itself controls internal registers and switches FPU component disable.
8. **Software control:** The CM33 reads the CPG register (CPG_LP_CM33_CTL1) and polls it until CM33_LP_QCH3 = 0.
9. **Software control:** The CM33 controls the CPG register (CPG_LP_CM33_CTL1) and sets CM33_LP_QCH12 to 0.
10. **Software control:** The CM33 reads the CPG register (CPG_LP_CM33_CTL1) and polls it until CM33_LP_QCH20 = 0.
11. **Software control:** Execute a WFI instruction as a CM33 instruction.

4.4.8.8.3 Procedure for return from the mode

<Steps 1 to 4 are executed only when the FPU is used>

1. **Software control:** The CM33 controls the CPG register (CPG_LP_CM33_CTL1) and sets CM33_LP_QCH12 to 1.
2. **Software control:** The CM33 reads the CPG register (CPG_LP_CM33_CTL1) and polls it until CM33_LP_QCH20 = 0.
3. **Software control:** After completion of interrupt processing, clear all interrupt factors from the external chip in each IP or ICU_TOP.

4. **Software control:** The CM33 itself controls internal registers and switches FPU component enable.
5. **Software control:** Th CM33 controls the CPG register (CPG_LP_CM33_CTL0) and sets CM33_LP_CTL24 to 0.
6. **Software control:** The CM33 controls the CPG register (CPG_LP_CTL1) and clears STBY to 0.

4.4.9 Operation

4.4.9.1 Procedure for Switching Unit Clock Signals On and Off

The clock on/off should be handled in the following steps.

Note: The procedure is described using the SRAM0 clock (SRAM_0_ACLK) as an example.

[Clock on setting]

1. Set the SRAM0 clock (SRAM_0_ACLK) to on.

Set the following bits in the CGC control register (CPG_CLKON_1).
Clock on setting: Bit 5 (CLK5_ON) = 1b, bit 21 (CLK5_ONWEN) = 1b

2. Check that the SRAM0 clock (SRAM_0_ACLK) is on.

To check that the clock has been output, read the following bit in the CGC monitor register (CPG_CLKMON_0).
Clock state: Bit 21 (CLK21_MON) 1b: Clock on (0b: Clock off)

[Clock off setting]

1. Set the SRAM0 clock (SRAM_0_ACLK) to off.

Set the following bits in the CGC control register (CPG_CLKON_1).
Clock off setting: Bit 5 (CLK5_ON) = 0b, bit 21 (CLK5_ONWEN) = 1b

2. Check that the SRAM0 clock (SRAM_0_ACLK) is off.

To check that the clock has stopped, read the following bit in the CGC monitor register (CPG_CLKMON_0).
Clock state: Bit 21 (CLK21_MON) 0b: Clock off (1b: Clock on)

The above procedure is applied to the following registers.

- CGC control register (CPG_CLKON_*)
- CGC monitor register (CPG_CLKMON_*)

For clock on/off settings, refer to the section for each unit and follow the procedures described in the section for that unit.

4.4.9.2 Procedure for Switching Unit Resets On and Off

The unit reset on (reset state)/off (reset release) should be performed according to the following procedure.

Note: The procedure is described using the SRAM0 reset (SRAM_0_ARESETN) as an example.

[Reset on setting]

1. Set the SRAM0 reset (SRAM_0_ARESETN) to on.

Set the following bits in the RESET ON/OFF control register (CPG_RST_3).
Reset on setting: Bit 11 (RSTB11) = 0b, bit 27 (RSTB11_WEN) = 1b

2. Check the SRAM0 reset (SRAM_0_ARESETN) to on.

To check that it has been reset, read the following bits in the RESET monitor register (CPG_RSTMON_1).
Reset state: Bit 12 (RST12_MON) 0b: Reset off (reset cancel) 1b: Reset on

[Reset off setting]

1. Set the SRAM0 reset (SRAM_0_ARESETN) to off.

Set the following bits in the RESET ON/OFF control register (CPG_RST_3).
Reset off setting: Bit 11(RSTB11) = 1b, bit 27 (RSTB11_WEN) = 1b

2. Check the SRAM0 reset (SRAM_0_ARESETN) is off.

To check that the reset has been released, read the following bit in the RESET monitor register (CPG_RSTMON_1).
Reset state: Bit 12 (RST12_MON) 0b: Reset off (reset canceled) 1b: Reset on

The above procedure is applied to the following registers.

- RESET on/off control register (CPG_RST_*)
- RESET monitor register (CPG_RSTMON_*)

For reset on/off settings, refer to the section for each unit and follow the procedures described in the section for that unit.

4.4.9.3 Procedure for Starting up Units

The procedures below must be followed to start up inactive units after a system reset.

Note: For unit-specific startup procedures, see the section for each unit.

1. Set the clock connected to the IP to on by the CGC control register; if the PLL of the clock domain to be turned on is stopped, start the PLL first.
2. Poll the CGC monitor register to check that supply of the relevant clock has started.
3. Set the reset connected to the unit to reset release by the RESET on/off control register.
4. Poll the RESET monitor register to check that the corresponding reset has been released.
5. 5. Set the MSTOP register as required.

4.4.9.4 Procedure for Setting PLLs

4.4.9.4.1 Procedure for setting the PLL normal mode (changing the output frequency and SSC mode)

Follow the procedure below to make a setting of PLL from the standby to normal mode (output frequency and SSC mode change).

[n = CLN, DTY, CA55, VDO, ETH, DSI0, DSI1, DDR0]

This operation is not performed for PLLCM33.

1. Checking the operating mode of the PLL (checking that it is in the standby mode)

Check the following bit of the given PLLn monitoring register (CPG_PLLn_MON).
Checking the transition to the standby mode: Bit 0 (PLLn_RESETB) must be 0b (standby mode).

Check the following bit of the given PLLn monitoring register (CPG_PLLn_MON).
Checking that the PLL is not locked: Bit 4 (PLLn_LOCK) must be 0b (PLL unlocked).

*Wait until the above conditions are satisfied.

2. Output clock settings 1

Change the following bits of the given PLLn output clock setting register 1 (CPG_PLLn_CLK1).
For details of the values to set, see **4.4.5.3.5 PLL** and **4.4.4.2 pllname Output Clock Setting Registers 1 (CPG_pllname_CLK1)**.
Setting output frequency (as desired):
Bits 5 to 0 (DIV_P_5-0), bits 15 to 6 (DIV_M_9-0), bits 31 to 16 (DIV_K_15-0)

3. Setting the output clock and SSC modulation value 2

Change the following bits of the given PLLn output clock setting register 2 (CPG_PLLn_CLK2).
For details of the values to set, see **4.4.4.3 pllname Output Clock Setting Registers 2 (CPG_pllname_CLK2)**.
Setting output frequency (as desired): Bits 2 to 0 (DIV_S_2-0)
Setting SSC modulation value (as desired): bit13-8(MRR_5-0), bit23-16(MFR_7-0)

4. Setting the PLL for normal mode

Set the following bits of the given PLLn standby control register (CPG_PLLn_STBY).

Setting the operating mode: Bit 0 (RESETB) = 1b (normal mode)
Setting the SSC mode: Bit 2 (SSC_EN) is set as desired.

5. Switching the PLL to normal mode and checking that the output clock is stable

Check the following bit of the given PLLn monitoring register (CPG_PLLn_MON).
Checking the transition to the normal mode: Bit 0 (PLLn_RESETB) must be 1b (normal mode).

Check the following bit of the given PLLn monitoring register (CPG_PLLn_MON).
Checking that the PLL is locked: Bit 4 (PLL_LOCK) must be 1b (PLL locked).

*Wait until the above conditions are satisfied.

Note: Register settings for parameters that are not to be changed are not required in steps 2 and 3.

4.4.9.4.2 Procedure for setting the PLL for standby mode

Follow the procedure below to make a setting of PLL from the normal to standby mode.

[n = CLN, DTY, CA55, VDO, ETH, DSI, DDR0, GPU, DRP]

This operation is not performed for PLLCM33.

1. Setting the PLL for normal mode

Set the following bit of the given PLLn standby control register (CPG_PLLn_STBY).
Setting the operating mode: Bit 0 (RESETB) = 0b (standby mode) (the other bits are 0b)

2. Checking the transition of the PLL to the standby mode

Check the following bit of the given PLLn monitoring register (CPG_PLLn_MON).
Checking the transition to the standby mode: Bit 0 (PLLn_RESETB) must be 0b (standby mode).

Check the following bit of the given PLLn monitoring register (CPG_PLLn_MON).
Checking that the PLL is not locked: Bit 4 (PLLn_LOCK) must be 0b (PLL unlocked).

*Wait until the above conditions are satisfied.

4.4.9.5 Procedure for Setting Frequency Division Ratios of Dynamic Switching Dividers

To set the frequency divider ratio of a dynamic switching divider, follow the procedure below.

Note: The procedure is described using the example of switching PLLCM33_CST400_GEAR (instance name: CDDIV_2to64_PLLCM33_CST400).

1. Check the status of PLLCM33_CST400_GEAR (make sure that it is not busy)

Check the following bit in the dynamic gear status monitor register (CPG_CLKSTATUS0).
Check the status of PLLCM33_CST400_GEAR: Bit 0 (CLK0_MON) = 0b (switching completion status).

*Wait until the above conditions are satisfied.

2. Set the frequency divider ratio of PLLCM33_CST400_GEAR

Set the following bit in the dynamic gear control register (CPG_CDDIV0).
Frequency division ratio setting: Bits 2 to 0 (DIVCTL0) = any, bit 16 (DIVCTL0WEN) = 1b

3. Check the status of PLLCM33_CST400_GEAR (make sure that it is not busy)

Check the following bit in the dynamic gear status monitor register (CPG_CLKSTATUS0).
Check the status of PLLCM33_CST400_GEAR: Bit 0 (CLK0_MON) = 0b (switching completion status).

*Wait until the above conditions are satisfied.

The above procedure is applied to the following registers.

- CPG_CDDIV0.DIVCTL0: PLLCM33_CST400_GEAR (CDDIV_2to64_PLLCM33_CST400)
- CPG_CDDIV0.DIVCTL1: PLLCM33_GEAR (CDDIV_2to64_PLLCM33)
- CPG_CDDIV0.DIVCTL2: PLLDTY_ACPU_GEAR (CDDIV_2to64_PLLDTY_ACPU)
- CPG_CDDIV0.DIVCTL3: PLLCA55_SCLK_GEAR (CDDIV_1to8_PLLCA55_SCLK)
- CPG_CDDIV1.DIVCTL0: PLLCA55_0_GEAR (CDDIV_1to8_CLK1800_PLLCA55_CA55_0)
- CPG_CDDIV1.DIVCTL1:PLLCA55_1_GEAR (CDDIV_1to8_CLK1800_PLLCA55_CA55_1)
- CPG_CDDIV1.DIVCTL2: PLLCA55_2_GEAR (CDDIV_1to8_CLK1800_PLLCA55_CA55_2)
- CPG_CDDIV1.DIVCTL3: PLLCA55_3_GEAR (CDDIV_1to8_CLK1800_PLLCA55_CA55_3)
- CPG_CDDIV2.DIVCTL0: PLLCA55_PERIPHCLK_GEAR (CDDIV_2to16_CA55PERI)
- CPG_CDDIV3.DIVCTL1: PLLVDO_GPU_GEAR (CDDIV_2to64_PLLVDO_GPU)
- CPG_CDDIV3.DIVCTL2: PLLDTY_RCPU_GEAR (CDDIV_2to64_PLLDTY_RCPU)
- CPG_CDDIV3.DIVCTL3: PLLVDO_CRU0_GEAR (CDDIV_2to4_PLLVDO_CRU0)

The status monitor registers is shown below.

- Dynamic gear status monitor registers (CPG_CLKSTATUS0)

Even if there is no change in the setting value, writing 1 to “xx_WEN” allocated in the upper 16 bits of the register will start clock switching control. Therefore, wait until the status monitor (CPG_CLKSTATUS0) indicates that the switching has been completed.

4.4.9.6 Procedure for Setting Static Switching Dividers and Selectors

The division ratio setting of the static-switching frequency divider and the clock selection of the selector should be performed according to the following procedure.

- When the source clock is running

  Applicable unit clock OFF → Switching setting → Unit clock ON
  If any unit clocks cannot be stopped, switch over during reset of the respective units.

- When the source clock is stopped

  Switching setting only

The above procedure is applied to the following registers.

- CPG_CSDIV0.DIVCTL0: PLLETH_GBE0_GEAR (CSDIV_2to100_PLLETH_GBE0)
- CPG_CSDIV0.DIVCTL1: PLLETH_GBE1_GEAR (CSDIV_2to100_PLLETH_GBE1)
- CPG_CSDIV0.DIVCTL2: PLLETH_LPCLK_GEAR (CSDIV_16to128_PLLETH_LPCLK)
- CPG_CSDIV0.DIVCTL3: PLLCM33_XSPI_GEAR (CSDIV_2to16_PLLCM33_XSPI)
- CPG_CSDIV1.DIVCTL0: PLLCM33_ADC_PCLK_GEAR (CSDIV_8to10_PLLCM33_ADC_PCLK)
- CPG_CSDIV1.DIVCTL1: PLLCM33_ADC_ADCLK_GEAR (CSDIV_2to16_PLLCM33_ADC_ADCLK)
- CPG_CSDIV1.DIVCTL2: PLLDSI0_GEAR (CSDIV_2to16_PLLDSI0)
- CPG_CSDIV1.DIVCTL3: PLLDSI1_GEAR (CSDIV_2to16_PLLDSI1)
- CPG_SSEL0.SELCTL0: SMUX2_CA55_SCLK1
- CPG_SSEL0.SELCTL1: SMUX2_CA55_SCLK0
- CPG_SSEL0.SELCTL2: SMUX2_GBE0_TXCLK
- CPG_SSEL0.SELCTL3: SMUX2_GBE0_RXCLK
- CPG_SSEL1.SELCTL0: SMUX2_GBE1_TXCLK
- CPG_SSEL1.SELCTL1: SMUX2_GBE1_RXCLK
- CPG_SSEL1.SELCTL2: SMUX2_XSPI_CLK0
- CPG_SSEL1.SELCTL3: SMUX2_XSPI_CLK1

4.4.9.7 Boot Sequence (Normal Mode)

For a CA55 boot, after the completion of **4.4.7.2 CM33 Boot (Normal Mode)**, perform the following settings, as required.

- Startup of individual PLLs: See **4.4.9.4.1 Procedure for setting the PLL normal mode (changing the output frequency and SSC mode)**.
- Clock selector settings: See **4.4.9.6 Procedure for Setting Static Switching Dividers and Selectors**.
- Clock divider ratio setting: See **4.4.9.5 Procedure for Setting Frequency Division Ratios of Dynamic Switching Dividers**.
- Setting individual unit clocks to ON: See **4.4.9.1 Procedure for Switching Unit Clock Signals On and Off**.
- Releasing individual unit resets: See **4.4.9.2 Procedure for Switching Unit Resets On and Off**.

For a CM33 boot, after the completion of **4.4.7.2 CM33 Boot (Normal Mode)**, perform the following settings, as required.

- Startup of individual PLLs: See **4.4.9.4.1 Procedure for setting the PLL normal mode (changing the output frequency and SSC mode)**.
- Clock selector settings: See **4.4.9.6 Procedure for Setting Static Switching Dividers and Selectors**.
- Clock divider ratio setting: See **4.4.9.5 Procedure for Setting Frequency Division Ratios of Dynamic Switching Dividers**.
- Setting individual unit clocks to ON: See **4.4.9.1 Procedure for Switching Unit Clock Signals On and Off**.
- Releasing individual unit resets: See **4.4.9.2 Procedure for Switching Unit Resets On and Off**.

Moreover, when the PD_OTHERS power area is to be activated, the following settings should be made.

- Turn on the PD_OTHERS area according to the procedure described in “Initializing the PD_OTHERS Area”. (See **4.5 Power Management Unit (PMU)**)
- Apply the SRAM recovery bit according to the procedure “Case 2: When the PD_OTHERS domain is powered on (CM33 boot)” in **4.4.7.1 SRAM Rescue Circuit Control**.
- Change the source clock of CA55_0_SCLK as required. See **4.4.9.6 Procedure for Setting Static Switching Dividers and Selectors**.
- Change the frequency of CA55_0_SCLK, CA55_0_PERIPHCLK, CA55_0_SCLK, and CA55_0_CORECLK[3:0] as required.
- Start up the CA55.

Release the reset of CA55 and turn on the P-channel by software. (See **2.2 CPU**.)

After that, perform the following settings, as required.

- Startup of individual PLLs: See **4.4.9.4.1 Procedure for setting the PLL normal mode (changing the output frequency and SSC mode)**.
- Clock selector settings: See **4.4.9.6 Procedure for Setting Static Switching Dividers and Selectors**.
- Clock divider ratio setting: See **4.4.9.5 Procedure for Setting Frequency Division Ratios of Dynamic Switching Dividers**.
- Setting individual unit clocks to ON: See **4.4.9.1 Procedure for Switching Unit Clock Signals On and Off**.
- Releasing individual unit resets: See **4.4.9.2 Procedure for Switching Unit Resets On and Off**.

NOTE

If an error system reset occurs after turning on the PD_OTHERS and PD_CA55 power supply areas according to the PD_OTHERS area power control procedure after the CM33 boot, the PD_OTHERS and PD_CA55 power supply areas will remain ON, but the PMU will be in the initial state (PD_OTHERS and PD_CA55 power supply areas are set OFF) and this resulted in an inconsistent state. To resolve this inconsistency, check **4.4.4.15 Error Reset Registers (CPG_ERROR_RSTm) (m = 2 to 8**) after the CPU is booted, and if an error occurs, operate the LSI external power supply control unit (PMIC) to switch AWO = ON and OTHERS = OFF state to match the internal power supply status with the actual power supply status.

4.4.9.8 Boot Sequence (Debug Mode)

The procedure is similar to **4.4.9.7 Boot Sequence (Normal Mode)**.

After the completion of **4.4.7.5 CA55 Boot (Debug Mode)**, perform the following settings, as required.

- Startup of individual PLLs: See **4.4.9.4.1 Procedure for setting the PLL normal mode (changing the output frequency and SSC mode)**.
- Clock selector settings: See **4.4.9.6 Procedure for Setting Static Switching Dividers and Selectors**.
- Clock divider ratio setting: See **4.4.9.5 Procedure for Setting Frequency Division Ratios of Dynamic Switching Dividers**.
- Setting individual unit clocks to ON: See **4.4.9.1 Procedure for Switching Unit Clock Signals On and Off**.
- Releasing individual unit resets: See **4.4.9.2 Procedure for Switching Unit Resets On and Off**.

Moreover, when the PD_OTHERS power area is to be activated, the following settings should be made.

- Turn on the PD_OTHERS area according to the procedure described in “Initializing the PD_OTHERS Area”. (See **4.5 Power Management Unit (PMU)**)
- Apply the SRAM recovery bit according to the procedure “Case 2: When the PD_OTHERS domain is powered on (CM33 boot)” in **4.4.7.1 SRAM Rescue Circuit Control**.
- Change the source clock of CA55_0_SCLK as required. See **4.4.9.6 Procedure for Setting Static Switching Dividers and Selectors**.
- Change the frequency of CA55_0_SCLK, CA55_0_PERIPHCLK, CA55_0_SCLK, and CA55_0_CORECLK[3:0] as required.
- Start up the CA55.

Release the reset of CA55 and turn on the P-channel by software. (See **2.2 CPU**.)

After that, perform the following settings, as required.

- Startup of individual PLLs: See **4.4.9.4.1 Procedure for setting the PLL normal mode (changing the output frequency and SSC mode)**.
- Clock selector settings: See **4.4.9.6 Procedure for Setting Static Switching Dividers and Selectors**.
- Clock divider ratio setting: See **4.4.9.5 Procedure for Setting Frequency Division Ratios of Dynamic Switching Dividers**.
- Setting individual unit clocks to ON: See **4.4.9.1 Procedure for Switching Unit Clock Signals On and Off**.
- Releasing individual unit resets: See **4.4.9.2 Procedure for Switching Unit Resets On and Off**.

NOTE

If an error system reset occurs after turning on the PD_OTHERS and PD_CA55 power supply areas according to the PD_OTHERS area power control procedure after the CM33 boot, the PD_OTHERS and PD_CA55 power supply areas will remain ON, but the PMU will be in the initial state (PD_OTHERS and PD_CA55 power supply areas are set OFF) and this resulted in an inconsistent state. To resolve this inconsistency, check **4.4.4.15 Error Reset Registers (CPG_ERROR_RSTm) (m = 2 to 8)** after the CPU is booted, and if an error occurs, operate the LSI external power supply control unit (PMIC) to switch AWO = ON and OTHERS = OFF state to match the internal power supply status with the actual power supply status.

4.4.9.9 Procedure for Writing to Registers

There are two types of writable registers: those with write enable bits and those without write enable bits.

In the registers with write enable bits, the upper 16 bits are write-enabled and the lower 16 bits are normal data bits, allowing the value of only specific bits to be changed without read-modify-write.

To change the bit value, set the +16 of the bit number to be changed to 1b before writing.

(e.g., to set bit[0] to 1b, write 0001_0001h; to set bit[0] to 0b, write 0001_0000h; for 0000_xxxxh, the value of bit[15:0] does not change.)

Remark Some data registers support multiple bits of data for a single write enable bit. Refer to the functional description of each register.

4.4.9.10 Procedure for CM33 Warm Reset

When applying a CM33 warm reset, follow the procedure below.

1. Set the CPG_LP_CTL1.CM33WRESET_REQ register to 1b (requesting a CM33 warm reset) with the CM33.
2. Set the CPG_LP_CTL2.INTMASK_CM33_I register to 1b (masking interrupts for the Cortex-M33 from the ICU) with the CM33. When this register is set to 1b, the CPG outputs 1 to CM33_INTMASK of the ICU.
3. Set the SYSRESETREQ register in the CM33 to 1b with the CM33. After that, the CPG_LP_CM33_CTL0.CM33_WARMRESETREQ register of the CPG is set to 1b.
4. Execute a WFI with the CM33.

4.4.10 Restrictions

4.4.10.1 CM33 Systick Timer

When the CM33_CLK0 clock frequency is switched, the period of the CM33_CLK0EN signal for Systick timer counting is not guaranteed only for the first pulse after the clock frequency switch. Therefore, the Systick timer count register must be reset.

4.4.10.2 Restrictions on Using the Individual PLLs

Since PLLCM33 is the system clock that is constantly running, the user cannot switch it arbitrarily.

Since PLLCA55 is the operating clock for the CA55, any changes to the settings after the PD_OTHERS area is activated must be performed in CM33.

Because PLLCLN/PLLDTY is the system clock used for the bus, the setting cannot be switched after the PD_OTHERS area is activated.

PLLVDO/PLLETH/PLLDSI0/PLLDSI1/PLLDDR0 can be set, oscillated, and stopped at any timing by the user as long as it does not violate the usage conditions of the corresponding units.

4.4.10.3 Low Power Mode Transition Conditions

The following conditions apply when combining low power (LP) mode transitions.

Table 4.4-58 LP Mode Transition Conditions (CM33 Boot, AWO Mode)

Current LP Mode	Next LP Mode					
	Module Standby	Low-Frequency	SRAM Power-Saving	CM33 Sleep	Software Standby (CM33)	ALL_ON Transition
Module standby	—	○	△*4	△*1	△*1	△*2
Low-frequency	○	—	△*4	△*1	△*1	△*3
SRAM power-saving	○	○	—	△*5	△*5	△*2
CM33 sleep	×	×	×	—	×	×
Software standby (CM33)	×	×	×	×	—	×

Table 4.4-59 LP Mode Transition Conditions (CM33 Boot, ALL_ON Mode)

Current LP Mode	Next LP Mode					
	Module Standby	Low-Frequency	SRAM Power-Saving	CM33 Sleep	CA55 Sleep	AWO Transition
Module standby	—	○	△*1	△*1	△*1	○
Low-frequency	○	—	△*1	△*1	△*1	○
SRAM power-saving	○	○	—	△*5	△*5	○
CM33 sleep	×	×	×	—	×	×
CA55 sleep	△*1	△*1	△*5	○	—	○

Table 4.4-60 LP Mode Transition Conditions (CA55 Boot)

Current LP Mode	Next LP Mode				
	Module Standby	Low-Frequency	SRAM Power-Saving	CM33 Sleep	CA55 Sleep
Module standby	—	○	△*4	△*1	△*1
Low-frequency	○	—	△*4	△*2	△*2
SRAM power-saving	○	○	—	△*4	△*4
CM33 sleep	△*1	△*1	△*5	—	○
CA55 sleep	×	×	×	×	—

Remarks: ○: Transition is possible.
△: Transition is conditionally possible.
×: Transition is not possible.

Note 1. The return source function must be ready for use (clocked and released from the reset).

Note 2. The units that are required for booting the PD_OTHERS area (such as CA55) must be ready for use.

Note 3. The frequency setting of the PD_OTHERS area is initialized.
Accordingly, low-frequency mode for the PD_OTHERS area is released forcibly.

Note 4. The SRAM for the main CPU (when booting the CM33: CM33; when booting the CA55: CA55) must be ready for use.

Note 5. The SRAM used by the return source function must be ready for use.

SECTION 4 SYSTEM

4.5 Power Management Unit (PMU)

This section describes the PMU functions of this LSI.

4.5.1 Overview

The power management unit (PMU) is a unit in the CPG, and includes the power control unit (PCU).

The PCU shuts down the undefined signals from PD_OTHERS to PD_AWO and PD_CA55 to PD_AWO (power supply isolation), which is required to control the power supply On/Off of PD_OTHERS and PD_CA55.

The following is a block diagram of the PMU.

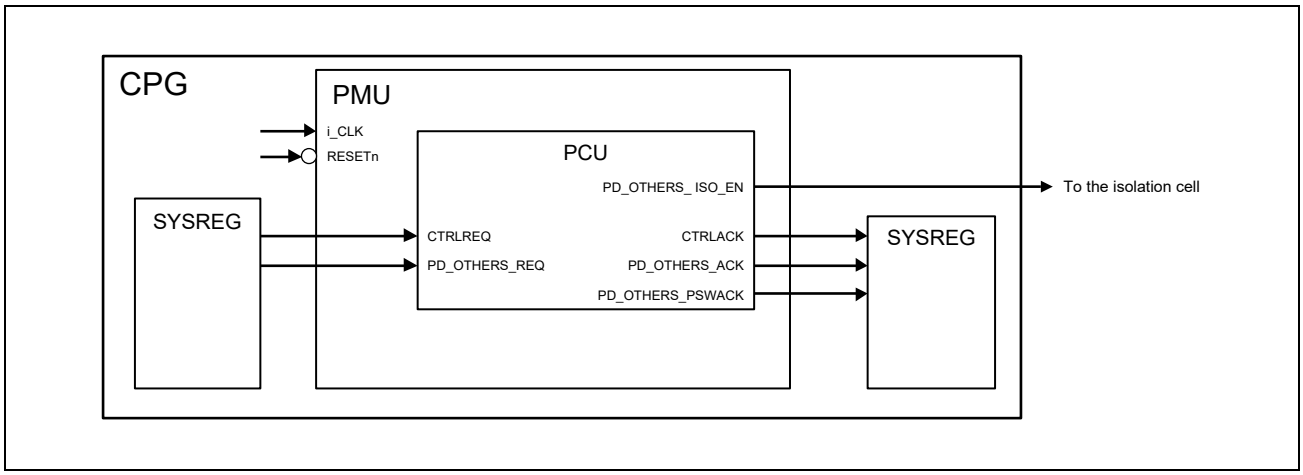


Figure 4.5-1 Block Diagram of PMU

4.5.2 Pin Description

The PMU doesn’t have external pins.

4.5.3 Power Domains

This LSI has multiple power supply domains and performs power ON/OFF control in block units. The power domains are classified into four types: PD_AWO, PD_OTHERS, PD_CA55, and PD_DDR0_IO. See **Table 4.5-1** and **Table 4.5-2** for available combinations.

The PMU controls the isolation cell (hereafter referred to as ISO cell) that is placed between power domains. See **Figure 4.5-2** for the outline of the configuration.

Table 4.5-3 shows the power domain to which the respective unit is assigned.

Table 4.5-1 Operating Combinations of Power Domains for CA55 Booting

	Power Mode		
Power Domain	ALL_OFF	ALL_ON	DDR_RETENTION
PD_AWO	OFF	ON	OFF
PD_OTHERS	OFF	ON	OFF
PD_CA55	OFF	ON	OFF
PD_DDR0_IO	OFF	ON	ON

Table 4.5-2 Operating Combinations of Power Domains for CM33 Booting

	Power Mode			
Power Domain	ALL_OFF	AWO	ALL_ON	AWO_DDR_RETENTION
PD_AWO	OFF	ON	ON	ON
PD_OTHERS	OFF	OFF	ON	OFF
PD_CA55	OFF	OFF	ON	OFF
PD_DDR0_IO	OFF	OFF	ON	ON

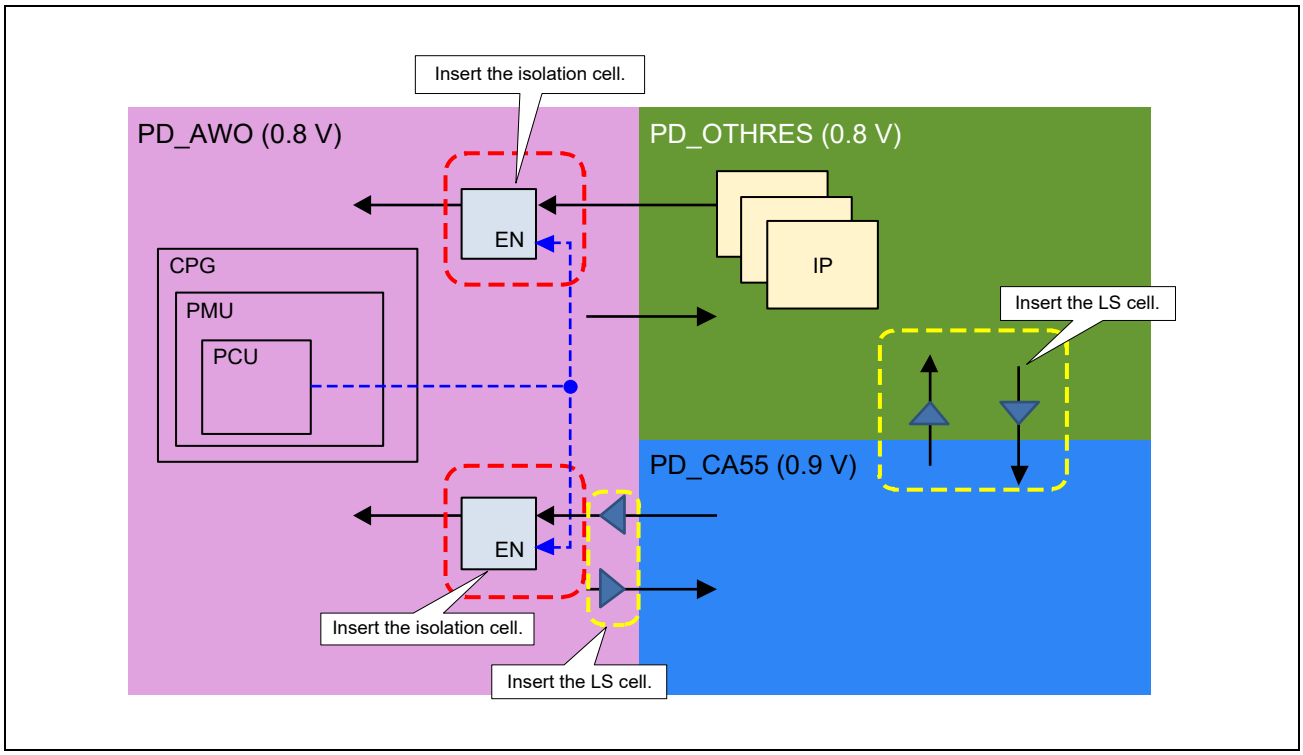


Figure 4.5-2 Control of ISO Cells

Table 4.5-3 Relationship between the Power Domain and the Respective Unit (1/2)

Power Domain	Unit
PD_AWO	ADC0
PD_AWO	ROM
PD_AWO	CPG_AWO
PD_AWO	CST
PD_AWO	ICU
PD_AWO	PFC_AWO
PD_AWO	CM33
PD_AWO	CMTW0
PD_AWO	CMTW1
PD_AWO	CMTW2
PD_AWO	CMTW3
PD_AWO	DMAC0
PD_AWO	GTM0
PD_AWO	GTM1
PD_AWO	RIIC8*1
PD_AWO	SRAM0
PD_OTHERS	SRAM1
PD_AWO	WDT0
PD_AWO	MHU
PD_AWO	OTP
PD_AWO	PDM0
PD_AWO	PDM1
PD_AWO	RTC
PD_AWO	SCIF
PD_AWO	xSPI
PD_AWO	SYS
PD_AWO	Secure IP
PD_AWO	MCPU bus
PD_AWO	TZC400_XSPI
PD_AWO	TZC400_SRAMM
PD_CA55	CA55
PD_CA55	CPG_CA55
PD_OTHERS	DDR0
PD_OTHERS	CPG_DDR0
PD_OTHERS	COM bus
PD_OTHERS	ACPU bus
PD_OTHERS	DMAC1
PD_OTHERS	DMAC2
PD_OTHERS	GTM2
PD_OTHERS	GTM3
PD_OTHERS	RIIC0
PD_OTHERS	RIIC1
PD_OTHERS	RIIC2
PD_OTHERS	RIIC3
PD_OTHERS	RIIC4

Power Domain	Unit
PD_OTHERS	RIIC5
PD_OTHERS	RIIC6
PD_OTHERS	RIIC7
PD_OTHERS	WDT1
PD_OTHERS	ADG
PD_OTHERS	CANFD
PD_OTHERS	CPG_OTHERS
PD_OTHERS	CRC
PD_OTHERS	CRU0
PD_OTHERS	ADMAC
PD_OTHERS	DSI
PD_OTHERS	GBETH0
PD_OTHERS	GBETH1
PD_OTHERS	GIC
PD_OTHERS	GPT0
PD_OTHERS	GPT1
PD_OTHERS	GE3D
PD_OTHERS	I3C0
PD_OTHERS	PFC_OTHERS
PD_OTHERS	LCDC
PD_OTHERS	PCIE
PD_OTHERS	RCPU bus
PD_OTHERS	POEG0A
PD_OTHERS	POEG0B
PD_OTHERS	POEG0C
PD_OTHERS	POEG0D
PD_OTHERS	POEG1A
PD_OTHERS	POEG1B
PD_OTHERS	POEG1C
PD_OTHERS	POEG1D
PD_OTHERS	CMTW4
PD_OTHERS	CMTW5
PD_OTHERS	CMTW6
PD_OTHERS	CMTW7
PD_OTHERS	DMAC3
PD_OTHERS	DMAC4
PD_OTHERS	GTM4
PD_OTHERS	GTM5
PD_OTHERS	GTM6
PD_OTHERS	GTM7
PD_OTHERS	WDT2
PD_OTHERS	WDT3
PD_OTHERS	RSCI0
PD_OTHERS	RSCI1
PD_OTHERS	RSCI2

Table 4.5-3 Relationship between the Power Domain and the Respective Unit (2/2)

Power Domain	Unit
PD_OTHERS	RSCI3
PD_OTHERS	RSCI4
PD_OTHERS	RSCI5
PD_OTHERS	RSCI6
PD_OTHERS	RSCI7
PD_OTHERS	RSCI8
PD_OTHERS	RSCI9
PD_OTHERS	RSPI0
PD_OTHERS	RSPI1
PD_OTHERS	RSPI2
PD_OTHERS	SCU
PD_OTHERS	SD0
PD_OTHERS	SD1
PD_OTHERS	SD2
PD_OTHERS	SPDIF0
PD_OTHERS	SPDIF1
PD_OTHERS	SPDIF2

Power Domain	Unit
PD_OTHERS	SSIU
PD_OTHERS	SYC
PD_OTHERS	TSU1
PD_OTHERS	USB2
PD_OTHERS	USB30
PD_OTHERS	VCD
PD_OTHERS	VIDEO0 bus
PD_OTHERS	VIDEO1 bus
PD_OTHERS	TZC400_AXI_RCPU
PD_OTHERS	TZC400_PCIE
PD_OTHERS	TZC400_DDR00
PD_OTHERS	TZC400_DDR01
PD_OTHERS	VSP
PD_OTHERS	FDP
PD_OTHERS	LVDS
PD_OTHERS	NPU
PD_OTHERS	NPU bus

Note 1. Use this channel, when controlling by RIIC is required while PMIC is in use.

4.5.3.1 Power Domain Control

There are five power modes: ALL_OFF mode, ALL_ON_mode, AWO mode, DDR_RETENTION mode, and AWO_DDR_RETENTION mode.

(1) In ALL_OFF mode, power is not supplied to all power supply areas.
(2) In ALL_ON mode, power is supplied to all power supply areas.
(3) In AWO mode, power is supplied to the PD_AWO area only.
(4) In DDR_RETENTION mode, power is supplied to the PD_DDR0_IO area only.
(5) In AWO_DDR_RETENTION mode, power is supplied to the PD_AWO area and PD_DDR0_IO area.

The power mode is switched to AWO mode immediately after the CM33 booting. Then, it can be switched to ALL_ON mode. Moreover, it can be switched to AWO_DDR_RETENTION mode for lower power consumption.

The power mode is switched to ALL_ON mode after CA55 booting. Then, it can be switched to DDR_RETENTION mode for lower power consumption. Switching to the AWO mode is not supported.

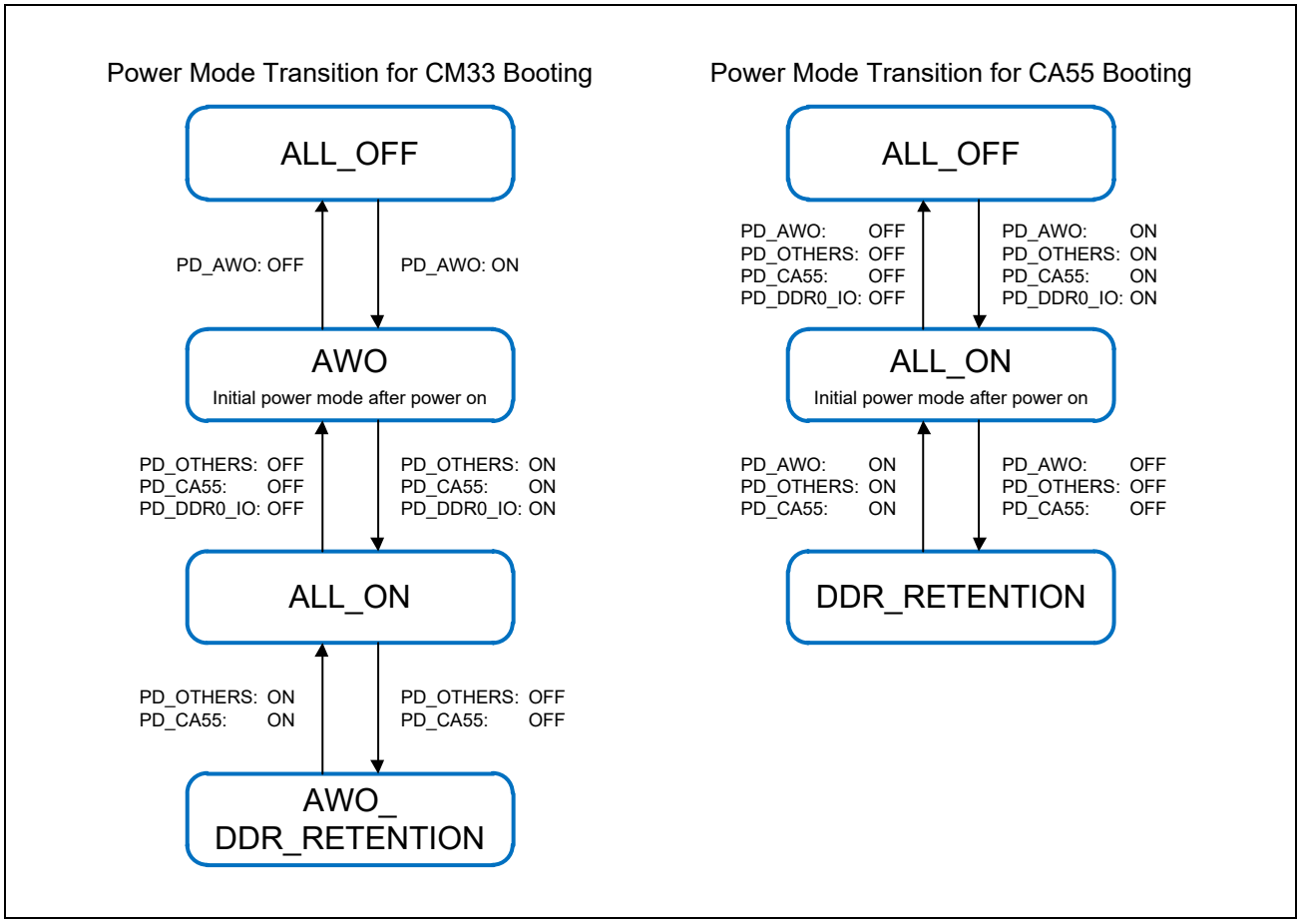


Figure 4.5-3 Power Mode Transition

The following describes the procedure for switching modes and for changing the PD_CA55 voltage.

4.5.3.1.1 Power mode (AWO/AWO_DDR_RETENTION → ALL_ON)

Table 4.5-4 Switching Procedure from AWO/AWO_DDR_RETENTION to ALL_ON

Step	CPU	Processing	Remarks
Pre-process			
1	CM33	Assert the reset of units in the PD_OTHERS area / PD_CA55 area	CPG_RST_n register (n= 0 to 18)
2	CM33	Assert the reset of the CPG circuit in the OTHERS area	CPG_OTHERS_INI.OTHERS_RST = 1b
3	CM33	Power on for the PD_OTHERS area / PD_CA55 area	RIIC and PFC control
4	CM33	Waiting for stabilization of power supply for the PD_OTHERS area / PD_CA55 area	Timer-system unit control
Entry			
1		AWO to ALL_ON: The external PMIC turns on a power other than PD_AWO area according to the power control sequence AWO_DDR_RETENTION to ALL_ON: The external PMIC turns on a power other than PD_AWO area and PD_DDR_IO area according to the power control sequence	See **10.1.3 Power-On/Off Sequence**.
2	CM33	Check that MSTOP between ACPU bus and MCPU bus is set (1b). If not, set it to 1.	CPG_BUS_12_MSTOP.MSTOP9_ON = 1b CPG_BUS_12_MSTOP.MSTOP10_ON = 1b
3	CM33	Deassert the reset of the CPG circuit in the OTHERS area	CPG_OTHERS_INI.OTHERS_RST = 0b
4	CM33	PLL startup: Change the operation mode (reset release)	CPG_*_STBY.RESETB = 1b
5	CM33	PLL startup: Check the Lock (polling)	CPG_PLL*_MON.PLLn__LOCK = 0b to 1b
6	CM33	Supply clock for units in the PD_OTHERS area / PD_CA55 area	See **4.4 Clock Pulse Generator (CPG)**.
7	CM33	Request the PMU to set PD_OTHERS to ON	CPG_LP_PMU_CTL1.PD_OTHERS_REQ = 1b
8	CM33	Request the PMU to control the power supply domain (can be controlled simultaneously with Step 7 above)	CPG_LP_PMU_CTL1.CTRLREQ = 1b
9	CM33	Check (polling) the completion of power domain control and the state (ON state) of PD_OTHERS	CPG_LP_PMU_CTL1.CTRLACK = 0b CPG_LP_PMU_CTL1. PD_OTHERS_ACK = 1b
10	CM33	Clear the request for control of the power domain	CPG_LP_PMU_CTL1.CTRLREQ = 0b
11	CM33	Check (polling) the status of the completion flag for SRAM rescue information transfer	OTP_HANDSHAKE_MON.DFT_DONE=1b
Post-process			
1	CM33	Deassert the reset of units in the PD_OTHERS area	See **4.4 Clock Pulse Generator (CPG)**.
2	CM33	Clear MSTOP between ACPU bus and MCPU bus (0b)	CPG_BUS_12_MSTOP.MSTOP9_ON = 0b CPG_BUS_12_MSTOP.MSTOP10_ON = 0b
3	CM33	Cluster warm reset (Normal reset release and CA55 core 0 startup) procedure	See **2.2 CPU**.
4	CM33	AWO to ALL_ON: DDR initialization sequence AWO_DDR_RETENTION to ALL_ON: Retention mode return sequence	

4.5.3.1.2 Power mode (ALL_ON → AWO/AWO_DDR_RETENTION)

Table 4.5-5 Switching Procedure from ALL_ON to AWO/AWO_DDR_RETENTION

Step	CPU	Processing	Remarks
Pre-process			
1	CM33	Stop DMA for units with DMA	-
2	CM33	ALL_ON to AWO_DDR_RETENTION: Save the DDR training data and shift to retention mode	
3	CM33	Set the multiplexed pins of the PFC to port mode (PMC_mn = 0) *Note*: If the port is connected to a device that is powered off, set it to Hi-Z.	See **4.2 Pin Function Controller (PFC)**.
Entry			
1	CM33	Set MSTOP between ACPU bus and MCPU bus (1b)	CPG_BUS_12_MSTOP.MSTOP9_ON = 1b CPG_BUS_12_MSTOP.MSTOP10_ON = 1b
2	CM33	Request the PMU to set PD_OTHERS to OFF	CPG_LP_PMU_CTL1.PD_OTHERS_REQ = 0b
3	CM33	Request the PMU to control the power supply domain (can be controlled simultaneously with Step 2 above)	CPG_LP_PMU_CTL1.CTRLREQ = 1b
4	CM33	Check (polling) the completion of power domain control and the state (OFF state) of PD_OTHERS	CPG_LP_PMU_CTL1.CTRLACK = 0b CPG_LP_PMU_CTL1.PD_OTHERS_ACK = 0b
5	CM33	Clear the request for control of the power domain	CPG_LP_PMU_CTL1.CTRLREQ = 0b
6	CM33	Set PLLs in the PD_OTHERS area/PD_CA55 area to the reset state Change the operation mode (Active → Reset state)	CPG_*_STBY.RESETB = 0b
Post-process			
1	CM33	ALL_ON to AWO: Shut down power to the PD_OTHERS area / PD_CA55 area / PD_DDR0_IO area ALL_ON to AWO_DDR_RETENTION: Shut down power to the PD_OTHERS area / PD_CA55 area The external PMIC should follow the power control sequence.	RIIC and PFC control See **10.1.3 Power-On/Off Sequence**.

4.5.3.1.3 Power mode (ALL_ON → DDR_RETENTION)

Step	CPU	Processing	Remarks
1	CA55	Stop DMA for units with DMA	
2	CA55	Save the DDR training data to the external device (SD, eMMC, xSPI etc.) and shift to retention mode	
3	CA55	Save the flag which indicates shift to retention mode in the register of the external device	
4	CA55	Notify the power shutdown permission to the external PMIC	RIIC and PFC control
5		The external PMIC should follow the power control sequence to shut down a power other than PD_DDR0_IO area	See **10.1.3 Power-On/Off Sequence**.

4.5.3.1.4 Power mode (DDR_RETENTION → ALL_ON)

Step	CPU	Processing	Remarks
1		The external PMIC turns on a power other than PD_DDR0_IO area according to the power control sequence	See **10.1.3 Power-On/Off Sequence**.
2	CA55	Check the flag in the external device to judge Cold-boot or Warm-boot.	
3	CA55	Restore the DDR training data from the external device to DDR PHY	
4	CA55	Exit from retention mode by controlling DDR controller and PHY	

4.5.3.1.5 PD_CA55 area voltage change

(1) When the power is not shut off (CA55 clock switching procedure is similar)

Table 4.5-6 PD_CA55 Area Voltage Change Procedure

Step	CPU	Processing	Remarks
1	CM33	Execute the shutdown procedure for the CA55	See **2.2 CPU**.
2	CM33	Wait for 128 CLK@24 MHz or longer	Timer-system unit control
3	CM33	Stop all CA55 clocks (CA55_0_PCLK, CA55_0_ATCLK, CA55_0_ACLK, CA55_0_GICCLK, and CA55_0_TSCLK) by controlling the CGC *Note*: CA55_0_CORECLK[3:0] and CA55_0_PERIPHCLK cannot be stopped because they have no CGC.	See **4.4 Clock Pulse Generator (CPG)**.
4	CM33	Assert the reset of PLLCA55	CPG_PLL3_STBY.RESETB = 0b
5	CM33	Changes the voltage in the PD_CA55 area *Note*: Step 5 is skipped if clock switching is only performed without changing the voltage.	RIIC and PFC control
6	CM33	Switch the Frequency of the PLL	See **4.4 Clock Pulse Generator (CPG)**.
7	CM33	Select the PLL which is used as source of CA55_0_SCLK	See **4.4 Clock Pulse Generator (CPG)**.
8	CM33	Supply all CA55 clocks (CA55_0_PCLK, CA55_0_ATCLK, CA55_0_ACLK, CA55_0_GICCLK, and CA55_0_TSCLK) by controlling the CGC	See **4.4 Clock Pulse Generator (CPG)**.
9	CM33	Wait for 128 CLK@24 MHz or longer	Timer-system unit control
10	CM33	Execute the procedure to release a cold reset on the CA55. *Note*: After releasing the reset of group A, release the reset of group B after a lapse of 64 CLK@24 MHz or longer.	See **2.2 CPU**.

(2) When the power is shut off

Set the power mode to AWO, set (change) the voltage in the PD_CA55 area, and then shift the power mode to ALL_ON again.

4.5.4 Usage Note

4.5.4.1 Initialization of PD_OTHERS

Turning the PD_OTHERS area on after the CM33 boot requires initialization of this area. Initialization is done by the CPG register control.

Initialization proceeds by turning the reset on and providing a clock supply period for the units in the PD_OTHERS area. To supply clocks, the PLL oscillation for the PD_OTHERS area is required.

For the list of the units in the PD_OTHERS area that requires initialization, see **Table 4.5-3**.

NOTE

Individual ON/OFF settings for SYSTEM BUS in the PD_OTHERS area are not recommended.

SECTION 4 SYSTEM

4.6 Interrupt Controller

This section describes the interrupt control systems of this LSI chip, specifically the interrupt control unit (ICU), generic interrupt controller (GIC), and message handling unit (MHU).

This manual is a simplified version. For more information, refer to the User's Manual Additional Document.

4.6.1 Interrupt Control Unit (ICU)

The ICU uses internal registers to proceed with the interrupt event processing such as masking, logical inversion, retention of source factors, output of interrupt and event signals, and the generation of error reset triggers.

4.6.1.1 Functional Block Diagram

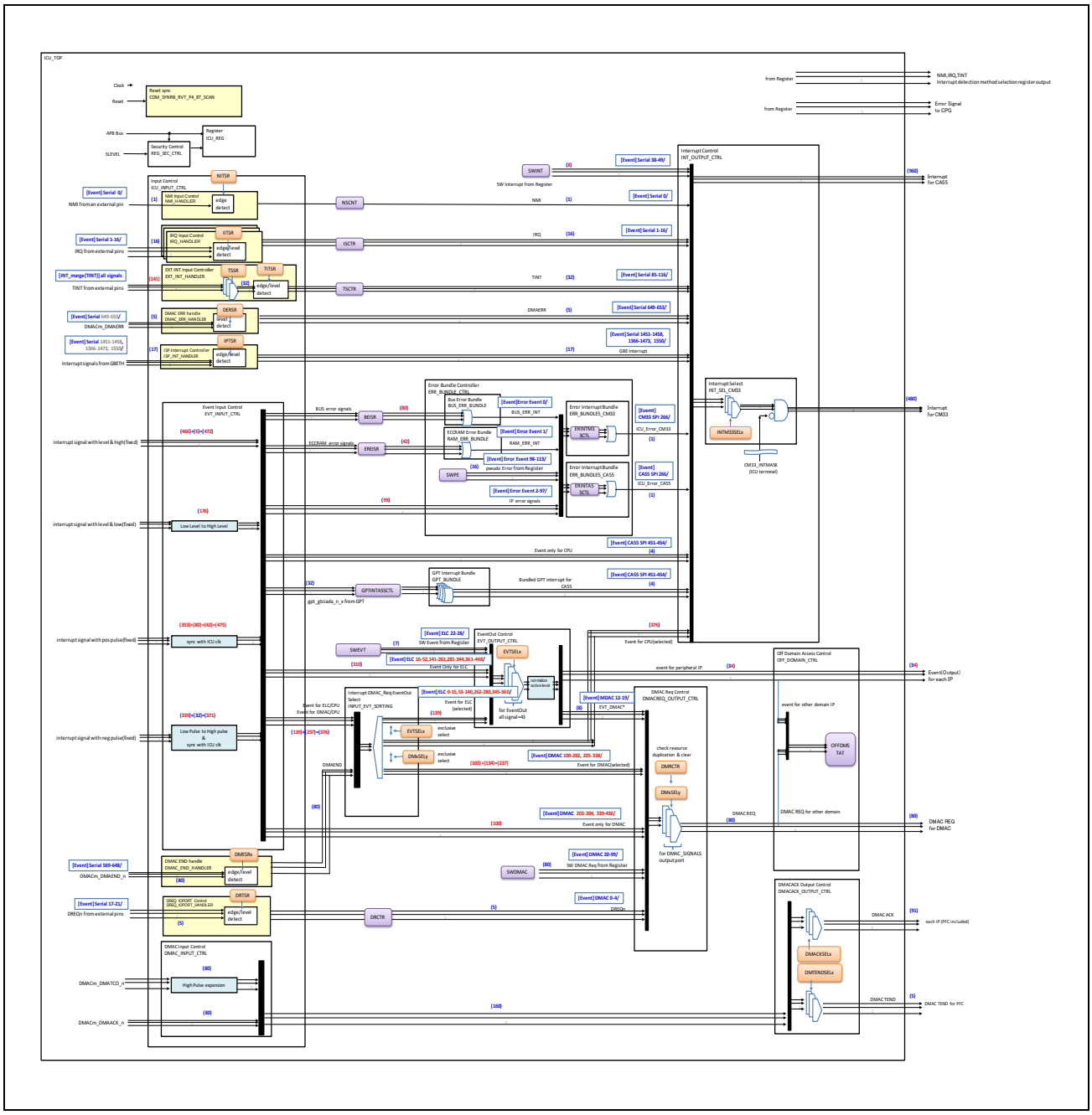


Figure 4.6-1 Block Diagram of the ICU

4.6.1.2 Registers

The base addresses for the ICU registers are given below.

Table 4.6-1 Register Base Addresses

Base Address Name	Base Address
<ICU_base>	0_1040_0000h (5040_0000h*1, 4040_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

4.6.1.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
NMI Status Register	ICU_NSCNT	0000_0000h	0000h	32
NMI Status Clear Register	ICU_NSCLR	0000_0000h	0004h	32
NMI Detection Method Selection Register	ICU_NITSR	0000_0000h	0008h	32
Reserve	-	-	000Ch to 000Fh	-
IRQ Status Register	ICU_ISCTR	0000_0000h	0010h	32
IRQ Status Clear Register	ICU_ISCLR	0000_0000h	0014h	32
IRQ Interrupt Detection Method Selection Register	ICU_IITSR	0000_0000h	0018h	32
Reserve	-	-	001Ch to 004Fh	-
DREQ Status Register	ICU_DRCTR	0000_0000h	0050h	32
DREQ Status Clear Register	ICU_DRCLR	0000_0000h	0054h	32
DREQ Detection Method Selection Register	ICU_DRTSR	0000_0000h	0058h	32
Reserve	-	-	005Ch to 005Fh	-
IP Detection Method Selection Register	ICU_IPTSR	0000_00F0h	0060h	32
Reserve	-	-	0064h to 006Fh	-
Bus Error Interrupt Status Register 0	ICU_BEISR0	0000_0000h	0070h	32
Bus Error Interrupt Status Register 1	ICU_BEISR1	0000_0000h	0074h	32
Bus Error Interrupt Status Register 2	ICU_BEISR2	0000_0000h	0078h	32
Bus Error Interrupt Status Register 3	ICU_BEISR3	0000_0000h	007Ch	32
Bus Error Interrupt Status Clear Register 0	ICU_BECLR0	0000_0000h	0080h	32
Bus Error Interrupt Status Clear Register 1	ICU_BECLR1	0000_0000h	0084h	32
Bus Error Interrupt Status Clear Register 2	ICU_BECLR2	0000_0000h	0088h	32
Bus Error Interrupt Status Clear Register 3	ICU_BECLR3	0000_0000h	008Ch	32
Reserve	-	-	0090h	-
ECCRAM Error Interrupt Status Register 1	ICU_EREISR1	0000_0000h	0094h	32
Reserve	-	-	0098h to 009Fh	-
ECCRAM Error Interrupt Status Register 4	ICU_EREISR4	0000_0000h	00A0h	32
Reserve	-	-	00A4h to 00E3h	-
ECCRAM Error Interrupt Status Clear Register 1	ICU_ERCLR1	0000_0000h	00E4h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	00E8h to 00EFh	-
ECCRAM Error Interrupt Status Clear Register 4	ICU_ERCLR4	0000_0000h	00F0h	32
Reserve	-	-	00F4h to 012Fh	-
SW Interrupt Generation Register	ICU_SWINT	0000_0000h	0130h	32
Reserve	-	-	0134h to 01FFh	-
SELECT Interrupt CM33 Selection Register k (k = 0 to 42)	ICU_INTM33SELk	3FFF_FFFFh	0200h + k x 0004h	32
Reserve	-	-	02ACh to 0303h	-
Error Interrupt CM33 Status Register 0	ICU_ERINTM33CTL0	0000_0000h	0304h	32
Error Interrupt CM33 Status Register 1	ICU_ERINTM33CTL1	0000_0000h	0308h	32
Error Interrupt CM33 Status Register 2	ICU_ERINTM33CTL2	0000_0000h	030Ch	32
Error Interrupt CM33 Status Register 3	ICU_ERINTM33CTL3	0000_0000h	0310h	32
Error Interrupt CM33 Status Clear Register 0	ICU_ERINTM33CLR0	0000_0000h	0314h	32
Reserve	-	-	0318h to 031Bh	-
Error Interrupt CM33 Status Clear Register 1	ICU_ERINTM33CLR1	0000_0000h	031Ch	32
Error Interrupt CM33 Status Clear Register 2	ICU_ERINTM33CLR2	0000_0000h	0320h	32
Error Interrupt CM33 Status Clear Register 3	ICU_ERINTM33CLR3	0000_0000h	0324h	32
Error Interrupt CM33 Mask Control Register 0	ICU_ERINTM33MSK0	0000_0000h	0328h	32
Error Interrupt CM33 Mask Control Register 1	ICU_ERINTM33MSK1	0000_0000h	032Ch	32
Error Interrupt CM33 Mask Control Register 2	ICU_ERINTM33MSK2	0000_0000h	0330h	32
Error Interrupt CM33 Mask Control Register 3	ICU_ERINTM33MSK3	0000_0000h	0334h	32
Error Interrupt CA55 Status Register 0	ICU_ERINTA55CTL0	0000_0000h	0338h	32
Error Interrupt CA55 Status Register 1	ICU_ERINTA55CTL1	0000_0000h	033Ch	32
Error Interrupt CA55 Status Register 2	ICU_ERINTA55CTL2	0000_0000h	0340h	32
Error Interrupt CA55 Status Register 3	ICU_ERINTA55CTL3	0000_0000h	0344h	32
Error Interrupt CA55 Status Clear Register 0	ICU_ERINTA55CLR0	0000_0000h	0348h	32
Error Interrupt CA55 Status Clear Register 1	ICU_ERINTA55CLR1	0000_0000h	034Ch	32
Error Interrupt CA55 Status Clear Register 2	ICU_ERINTA55CLR2	0000_0000h	0350h	32
Error Interrupt CA55 Status Clear Register 3	ICU_ERINTA55CLR3	0000_0000h	0354h	32
Error Interrupt CA55 Mask Control Register 0	ICU_ERINTA55MSK0	0000_0000h	0358h	32
Error Interrupt CA55 Mask Control Register 1	ICU_ERINTA55MSK1	0000_0000h	035Ch	32
Error Interrupt CA55 Mask Control Register 2	ICU_ERINTA55MSK2	0000_0000h	0360h	32
Error Interrupt CA55 Mask Control Register 3	ICU_ERINTA55MSK3	0000_0000h	0364h	32
GPT Interrupt CA55 Status Register	ICU_GPTINTA55CTL	0000_0000h	0368h	32
GPT Interrupt CA55 Status Clear Register	ICU_GPTINTA55CLR	0000_0000h	036Ch	32
SW Enhancing Pseudo Error Generation Register	ICU_SWPE	0000_0000h	0370h	32
Reserve	-	-	0374h to 03FFh	-
SW DMAC Request Generation Register 0	ICU_SWDMAC0	0000_0000h	0400h	32
SW DMAC Request Generation Register 1	ICU_SWDMAC1	0000_0000h	0404h	32
SW DMAC Request Generation Register 2	ICU_SWDMAC2	0000_0000h	0408h	32
SW DMAC Request Generation Register 3	ICU_SWDMAC3	0000_0000h	040Ch	32
SW DMAC Request Generation Register 4	ICU_SWDMAC4	0000_0000h	0410h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	0414h to 041Fh	-
DMAC Factor k Selection Register y (k = 0 to 4, y = 0 to 7)	ICU_DMkSELy	03FF_03FFh	0420h + k x 0020h + y x 0004h	32
DMAC Request Repetition Control Register 0	ICU_DMRCTR0	0000_0000h	04C0h	32
DMAC Request Repetition Control Register 1	ICU_DMRCTR1	0000_0000h	04C4h	32
DMAC Request Repetition Control Register 2	ICU_DMRCTR2	0000_0000h	04C8h	32
Reserve	-	-	04CCh to 04CFh	-
DMAC Request Repetition Clear Register 0	ICU_DMRCLR0	0000_0000h	04D0h	32
DMAC Request Repetition Clear Register 1	ICU_DMRCLR1	0000_0000h	04D4h	32
DMAC Request Repetition Clear Register 2	ICU_DMRCLR2	0000_0000h	04D8h	32
Reserve	-	-	04DCh to 04FFh	-
DMAC ACK Selection Register k (k = 0 to 22)	ICU_DMACKSELk	7F7F_7F7Fh	0500h + k x 0004h	32
DMAC TEND Selection Register 0	ICU_DMTENDSEL0	7F7F_7F7Fh	055Ch	32
DMAC TEND Selection Register 1	ICU_DMTENDSEL1	7F7F_7F7Fh	0560h	32
Reserve	-	-	0564h to 056Fh	-
DMAC END Detection Method Selection Register 0	ICU_DMESR0	0000_0000h	0570h	32
DMAC END Detection Method Selection Register 1	ICU_DMESR1	0000_0000h	0574h	32
DMAC END Detection Method Selection Register 2	ICU_DMESR2	0000_0000h	0578h	32
DMA ERR Interrupt Detection Method Selection Register	ICU_DERSR	0000_0000h	057Ch	32
Reserve	-	-	0580h to 05FFh	-
SW Event Output Generation Register	ICU_SWEVT	0000_0000h	0600h	32
Event Output Factor Selection Register 0	ICU_EVTSEL0	3FFF_FFFFh	0604h	32
Event Output Factor Selection Register 1	ICU_EVTSEL1	3FFF_FFFFh	0608h	32
Event Output Factor Selection Register 2	ICU_EVTSEL2	3FFF_FFFFh	060Ch	32
Event Output Factor Selection Register 3	ICU_EVTSEL3	3FFF_FFFFh	0610h	32
Event Output Factor Selection Register 4	ICU_EVTSEL4	3FFF_FFFFh	0614h	32
Event Output Factor Selection Register 5	ICU_EVTSEL5	3FFF_FFFFh	0618h	32
Event Output Factor Selection Register 6	ICU_EVTSEL6	3FFF_FFFFh	061Ch	32
Event Output Factor Selection Register 7	ICU_EVTSEL7	3FFF_FFFFh	0620h	32
Event Output Factor Selection Register 8	ICU_EVTSEL8	3FFF_FFFFh	0624h	32
Event Output Factor Selection Register 9	ICU_EVTSEL9	3FFF_FFFFh	0628h	32
Event Output Factor Selection Register 10	ICU_EVTSEL10	3FFF_FFFFh	062Ch	32
Event Output Factor Selection Register 11	ICU_EVTSEL11	3FFF_FFFFh	0630h	32
Event Output Factor Selection Register 12	ICU_EVTSEL12	3FFF_FFFFh	0634h	32
Event Output Factor Selection Register 13	ICU_EVTSEL13	3FFF_FFFFh	0638h	32
Event Output Factor Selection Register 14	ICU_EVTSEL14	3FFF_FFFFh	063Ch	32
Reserve	-	-	0640h to 077Fh	-
Off Domain Access Factor Maintenance Register 0	ICU_OFFDMSTAT0	0000_0000h	0780h	32
Off Domain Access Factor Maintenance Register 1	ICU_OFFDMSTAT1	0000_0000h	0784h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Off Domain Access Factor Maintenance Register 2	ICU_OFFDMSTAT2	0000_0000h	0788h	32
Reserve	-	-	078Ch to 0797h	-
Off Domain Access Factor Clear Register 0	ICU_OFFDMCLR0	0000_0000h	0798h	32
Off Domain Access Factor Clear Register 1	ICU_OFFDMCLR1	0000_0000h	079Ch	32
Off Domain Access Factor Clear Register 2	ICU_OFFDMCLR2	0000_0000h	07A0h	32
Reserve	-	-	07A4h to 081Fh	-
TINT Status Register	ICU_TSCTR	0000_0000h	0820h	32
TINT Status Clear Register	ICU_TSCLR	0000_0000h	0824h	32
TINT Detection Method Selection Register 0	ICU_TITSR0	0000_0000h	0828h	32
TINT Detection Method Selection Register 1	ICU_TITSR1	0000_0000h	082Ch	32
TINT Source Selection Register 0	ICU_TSSR0	0000_0000h	0830h	32
TINT Source Selection Register 1	ICU_TSSR1	0000_0000h	0834h	32
TINT Source Selection Register 2	ICU_TSSR2	0000_0000h	0838h	32
TINT Source Selection Register 3	ICU_TSSR3	0000_0000h	083Ch	32
TINT Source Selection Register 4	ICU_TSSR4	0000_0000h	0840h	32
TINT Source Selection Register 5	ICU_TSSR5	0000_0000h	0844h	32
TINT Source Selection Register 6	ICU_TSSR6	0000_0000h	0848h	32
TINT Source Selection Register 7	ICU_TSSR7	0000_0000h	084Ch	32
TINT Source Selection Register 8	ICU_TSSR8	0000_0000h	0850h	32
TINT Source Selection Register 9	ICU_TSSR9	0000_0000h	0854h	32
TINT Source Selection Register 10	ICU_TSSR10	0000_0000h	0858h	32
TINT Source Selection Register 11	ICU_TSSR11	0000_0000h	085Ch	32
TINT Source Selection Register 12	ICU_TSSR12	0000_0000h	0860h	32
TINT Source Selection Register 13	ICU_TSSR13	0000_0000h	0864h	32
TINT Source Selection Register 14	ICU_TSSR14	0000_0000h	0868h	32
TINT Source Selection Register 15	ICU_TSSR15	0000_0000h	086Ch	32

4.6.1.2.2 Register Description

The ICU has two reset signals.

- **PRESETn_I:**
  - Asserted on various resets.
- **ERROR_RESETn:**
  - Asserted at the reset by the QRESN pin.
  - Not asserted on a system software reset or error reset.

Remark Some registers are initialized with ERROR_RESETn. Refer to each register description for details.

The prefix (ICU_) of the register names is omitted in this and subsequent sections.

(1) NMI Status Register (ICU_NSCNT)

[Secure group: Gr1]

This register indicates the state of the NMI.

Access Size : 32 bits
Address : <ICU_base> + 0000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	NSMON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	NSTAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	NSMON	0h	R	Indicates the NMI pin level. 0b: The NMI pin is at the low level. 1b: The NMI pin is at the high level.
15 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	NSTAT	0h	R	Indicates the detection of an NMI interrupt. 0b: No NMI detected 1b: NMI detected

(2) NMI Status Clear Register (ICU_NSCLR)

[Secure group: Gr1]

This register clears the NMI state flag.

Remark When the NMI is to be used, use this register to clear the NMI state flag and enable acceptance of interrupts on the CPU side before doing so.
Clear the status flag after setting the ICU_NITSR register.

Access Size : 32 bits
Address : <ICU_base> + 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	NCLR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	NCLR	0h	W	The read value is undefined Clears the error factor of the event allocated in the corresponding bit by writing 1b.

(3) NMI Detection Method Selection Register (ICU_NITSR)

[Secure group: Gr1]

This register controls the method of detecting the NMI.

Remark When the NMI is to be used, set up this register and enable acceptance of interrupts on the CPU side before doing so.

Access Size : 32 bits
Address : <ICU_base> + 0008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	NTSEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	NTSEL	0h	RW	The interrupt detection method for NMI is selected. 0b: Falling edge detection 1b: Rising edge detection

(4) IRQ Status Register (ICU_ISCTR)

[Secure group: Gr1]

This register indicates the states of the IRQn interrupts.

Access Size : 32 bits

Address : <ICU_base> + 0010h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ISTAT15	ISTAT14	ISTAT13	ISTAT12	ISTAT11	ISTAT10	ISTAT9	ISTAT8	ISTAT7	ISTAT6	ISTAT5	ISTAT4	ISTAT3	ISTAT2	ISTAT1	ISTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	ISTATn	0h	R	Indicates the detection of an IRQn interrupt. 0b: No IRQn detected 1b: IRQn detected

(5) IRQ Status Clear Register (ICU_ISCLR)

[Secure group: Gr1]

This register clears the state flags for the IRQn interrupts.

Remark When the IRQ interrupts are to be used, use this register to clear the IRQ state flags and enable acceptance of interrupts on the CPU side before doing so.
Clear the status flag after setting the ICU_IITSR register.

Access Size : 32 bits
Address : <ICU_base> + 0014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ICLR15	ICLR14	ICLR13	ICLR12	ICLR11	ICLR10	ICLR9	ICLR8	ICLR7	ICLR6	ICLR5	ICLR4	ICLR3	ICLR2	ICLR1	ICLR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	ICLRn	0h	W	The read value is undefined Clears the error factor of the event allocated in the corresponding bit by writing 1b.

(6) IRQ Interrupt Detection Method Selection Register (ICU_IITSR)

[Secure group: Gr1]

This register controls the method of detecting the IRQ interrupts.

Remark 1. When the IRQ interrupts are to be used, set up this register and enable acceptance of interrupts on the CPU side before doing so.

Remark 2. This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read-modify-write operation).

Access Size : 32 bits
Address : <ICU_base> + 0018h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IITSEL15[1:0]		IITSEL14[1:0]		IITSEL13[1:0]		IITSEL12[1:0]		IITSEL11[1:0]		IITSEL10[1:0]		IITSEL9[1:0]		IITSEL8[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IITSEL7[1:0]		IITSEL6[1:0]		IITSEL5[1:0]		IITSEL4[1:0]		IITSEL3[1:0]		IITSEL2[1:0]		IITSEL1[1:0]		IITSEL0[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	IITSEL15[1:0]	0h	RW	The interrupt detection method for IRQ15 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
29, 28	IITSEL14[1:0]	0h	RW	The interrupt detection method for IRQ14 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
27, 26	IITSEL13[1:0]	0h	RW	The interrupt detection method for IRQ13 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
25, 24	IITSEL12[1:0]	0h	RW	The interrupt detection method for IRQ12 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
23, 22	IITSEL11[1:0]	0h	RW	The interrupt detection method for IRQ11 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
21, 20	IITSEL10[1:0]	0h	RW	The interrupt detection method for IRQ10 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection

Bit	Bit Name	Initial Value	R/W	Description
19, 18	IITSEL9[1:0]	0h	RW	The interrupt detection method for IRQ9 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
17, 16	IITSEL8[1:0]	0h	RW	The interrupt detection method for IRQ8 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
15, 14	IITSEL7[1:0]	0h	RW	The interrupt detection method for IRQ7 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
13, 12	IITSEL6[1:0]	0h	RW	The interrupt detection method for IRQ6 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
11, 10	IITSEL5[1:0]	0h	RW	The interrupt detection method for IRQ5 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
9, 8	IITSEL4[1:0]	0h	RW	The interrupt detection method for IRQ4 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
7, 6	IITSEL3[1:0]	0h	RW	The interrupt detection method for IRQ3 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
5, 4	IITSEL2[1:0]	0h	RW	The interrupt detection method for IRQ2 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
3, 2	IITSEL1[1:0]	0h	RW	The interrupt detection method for IRQ1 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
1, 0	IITSEL0[1:0]	0h	RW	The interrupt detection method for IRQ0 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection

(7) TINT Status Register (ICU_TSCTR)

[Secure group: Gr1]

This register indicates the states of the TINTn interrupts.

Access Size : 32 bits

Address : <ICU_base> + 0820h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSTAT 31	TSTAT 30	TSTAT 29	TSTAT 28	TSTAT 27	TSTAT 26	TSTAT 25	TSTAT 24	TSTAT 23	TSTAT 22	TSTAT 21	TSTAT 20	TSTAT 19	TSTAT 18	TSTAT 17	TSTAT 16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSTAT 15	TSTAT 14	TSTAT 13	TSTAT 12	TSTAT 11	TSTAT 10	TSTAT 9	TSTAT 8	TSTAT 7	TSTAT 6	TSTAT 5	TSTAT 4	TSTAT 3	TSTAT 2	TSTAT 1	TSTAT 0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TSTATn	0h	R	Indicates the detection of a TINTn interrupt. 0b: No TINTn detected 1b: TINTn detected

(8) TINT Status Clear Register (ICU_TSCLR)

[Secure group: Gr1]

This register clears the state flags for the TINTn interrupts.

Remark When the TINT interrupts are to be used, use this register to clear the TINT state flags and enable acceptance of interrupts on the CPU side before doing so.
Clear the status flag after setting the ICU_TITSRk and ICU_TSSRk registers.

Access Size : 32 bits
Address : <ICU_base> + 0824h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCLR31	TCLR30	TCLR29	TCLR28	TCLR27	TCLR26	TCLR25	TCLR24	TCLR23	TCLR22	TCLR21	TCLR20	TCLR19	TCLR18	TCLR17	TCLR16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCLR15	TCLR14	TCLR13	TCLR12	TCLR11	TCLR10	TCLR9	TCLR8	TCLR7	TCLR6	TCLR5	TCLR4	TCLR3	TCLR2	TCLR1	TCLR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TCLRn	0h	W	The read value is undefined When the type of TINTn interrupt detection is set for falling edge detection or rising edge detection with the TINT interrupt detection method selection register (TITSR) When TSTATn = 1b 1b: The TINTn interrupt detection is cleared (the status changes from "detected" to "not detected"). 0b: Writing has no effect. When TSTATn = 0b Writing has no effect. When high-level detection or low-level detection is selected in ITSRx, writing to these bits has no effect.

(9) TINT Detection Method Selection Register k (ICU_TITSRk) (k = 0, 1)

[Secure group: Gr1]

These registers control the method of detecting the TINT interrupts.

Remark 1. When the TINT interrupts are to be used, set up these registers and enable acceptance of interrupts on the CPU side before doing so.

Remark 2. This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read-modify-write operation).

Access Size : 32 bits

Address : <ICU_base> + 0828h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TITSELk_15 [1:0]		TITSELk_14 [1:0]		TITSELk_13 [1:0]		TITSELk_12 [1:0]		TITSELk_11 [1:0]		TITSELk_10 [1:0]		TITSELk_9 [1:0]		TITSELk_8[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TITSELk_7[1:0]		TITSELk_6[1:0]		TITSELk_5[1:0]		TITSELk_4[1:0]		TITSELk_3[1:0]		TITSELk_2[1:0]		TITSELk_1[1:0]		TITSELk_0[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	TITSELk_15 [1:0]	0h	RW	The interrupt detection method for TINTk_15 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
29, 28	TITSELk_14 [1:0]	0h	RW	The interrupt detection method for TINTk_14 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
27, 26	TITSELk_13 [1:0]	0h	RW	The interrupt detection method for TINTk_13 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
25, 24	TITSELk_12 [1:0]	0h	RW	The interrupt detection method for TINTk_12 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
23, 22	TITSELk_11 [1:0]	0h	RW	The interrupt detection method for TINTk_11 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
21, 20	TITSELk_10 [1:0]	0h	RW	The interrupt detection method for TINTk_10 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection

Bit	Bit Name	Initial Value	R/W	Description
19, 18	TITSELk_9[1:0]	0h	RW	The interrupt detection method for TINTk_9 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
17, 16	TITSELk_8[1:0]	0h	RW	The interrupt detection method for TINTk_8 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
15, 14	TITSELk_7[1:0]	0h	RW	The interrupt detection method for TINTk_7 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
13, 12	TITSELk_6[1:0]	0h	RW	The interrupt detection method for TINTk_6 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
11, 10	TITSELk_5[1:0]	0h	RW	The interrupt detection method for TINTk_5 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
9, 8	TITSELk_4[1:0]	0h	RW	The interrupt detection method for TINTk_4 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
7, 6	TITSELk_3[1:0]	0h	RW	The interrupt detection method for TINTk_3 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
5, 4	TITSELk_2[1:0]	0h	RW	The interrupt detection method for TINTk_2 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
3, 2	TITSELk_1[1:0]	0h	RW	The interrupt detection method for TINTk_1 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection
1, 0	TITSELk_0[1:0]	0h	RW	The interrupt detection method for TINTk_0 is selected. 00b: Rising edge detection 01b: Falling edge detection 10b: High-level detection 11b: Low-level detection

Table 4.6-2 Relationship between TINT and TITSELx_n

TINT Signal	TITSEL0_n
TINT_0	n = 0
TINT_1	n = 1
TINT_2	n = 2
TINT_3	n = 3
TINT_4	n = 4
TINT_5	n = 5
TINT_6	n = 6
TINT_7	n = 7
TINT_8	n = 8
TINT_9	n = 9
TINT_10	n = 10
TINT_11	n = 11
TINT_12	n = 12
TINT_13	n = 13
TINT_14	n = 14
TINT_15	n = 15

TINT Signal	TITSEL1_n
TINT_16	n = 0
TINT_17	n = 1
TINT_18	n = 2
TINT_19	n = 3
TINT_20	n = 4
TINT_21	n = 5
TINT_22	n = 6
TINT_23	n = 7
TINT_24	n = 8
TINT_25	n = 9
TINT_26	n = 10
TINT_27	n = 11
TINT_28	n = 12
TINT_29	n = 13
TINT_30	n = 14
TINT_31	n = 15

(10) TINT Source Selection Register k (ICU_TSSRk) (k = 0 to 15)

[Secure group: Gr1]

These registers control the input sources of TINTn interrupts.

Remark 1. When the TINT interrupts are to be used, set up these registers and enable acceptance of interrupts on the CPU side before doing so.

Remark 2. This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read-modify-write operation).

Access Size : 32 bits

Address : <ICU_base> + 0830h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIENk_1	-	-	-	-	-	-	-	TSSELk_1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIENk_0	-	-	-	-	-	-	-	TSSELk_0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	TIENk_1	0h	RW	TINT_(k x 2 + 1) enable 1b: Enable 0b: Disable
30 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 16	TSSELk_1[7:0]	0h	RW	The source of TINT_(k x 2 + 1) is selected. 0000_0000b: TINT_GPS0 0000_0001b: TINT_GPS1 ~ 1000_1100b: TINT_GP85 1000_1101b: Can not be set. ~ 1111_1111b: Can not be set. Refer to **Table 4.6.-4**.
15	TIENk_0	0h	RW	TINT_(k x 2) enable 1b: Enable 0b: Disable
14 to 8	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
7 to 0	TSSELk_0[7:0]	0h	RW	The source of TINT_(k x 2) is selected. 0000_0000b: TINT_GPS0 0000_0001b: TINT_GPS1 ~ 1000_1100b: TINT_GP85 1000_1101b: Can not be set. ~ 1111_1111b: Can not be set. Refer to **Table 4.6.-4**.

Table 4.6-3 Relationship between TINT, TIENk_n, and TSSELk_n

TINT Signal	TIENk_n, TSSELk_n	
TINT_0	n = 0	k = 0
TINT_1	n = 1	k = 0
TINT_2	n = 0	k = 1
TINT_3	n = 1	k = 1
TINT_4	n = 0	k = 2
TINT_5	n = 1	k = 2
TINT_6	n = 0	k = 3
TINT_7	n = 1	k = 3
TINT_8	n = 0	k = 4
TINT_9	n = 1	k = 4
TINT_10	n = 0	k = 5
TINT_11	n = 1	k = 5
TINT_12	n = 0	k = 6
TINT_13	n = 1	k = 6
TINT_14	n = 0	k = 7
TINT_15	n = 1	k = 7

TINT Signal	TIENk_n, TSSELk_n	
TINT_16	n = 0	k = 8
TINT_17	n = 1	k = 8
TINT_18	n = 0	k = 9
TINT_19	n = 1	k = 9
TINT_20	n = 0	k = 10
TINT_21	n = 1	k = 10
TINT_22	n = 0	k = 11
TINT_23	n = 1	k = 11
TINT_24	n = 0	k = 12
TINT_25	n = 1	k = 12
TINT_26	n = 0	k = 13
TINT_27	n = 1	k = 13
TINT_28	n = 0	k = 14
TINT_29	n = 1	k = 14
TINT_30	n = 0	k = 15
TINT_31	n = 1	k = 15

Table 4.6-4 Relationship between TSSELk_0/1 and External Pin

TSSELk_0/1 (decimal)	External Pin	TSSELk_0/1 (decimal)	External Pin	TSSELk_0/1 (decimal)	External Pin	TSSELk_0/1 (decimal)	External Pin
0	TINT_GPS0	36	TINT_GPH3	72	TINT_GPE2	108	TINT_GP43
1	TINT_GPS1	37	TINT_GPH4	73	TINT_GPE3	109	TINT_GP44
2	TINT_GPS2	38	TINT_GPH5	74	TINT_GPE4	110	TINT_GP45
3	TINT_GPS3	39	TINT_GPK0	75	TINT_GPE5	111	TINT_GP20
4	TINT_GPJ0	40	TINT_GPK1	76	TINT_GPE6	112	TINT_GP21
5	TINT_GPJ1	41	TINT_GPK2	77	TINT_GPE7	113	TINT_GP50
6	TINT_GPJ2	42	TINT_GPK3	78	TINT_GPF0	114	TINT_GP51
7	TINT_GPJ3	43	TINT_GPA0	79	TINT_GPF1	115	TINT_GP52
8	TINT_GPJ4	44	TINT_GPA1	80	TINT_GPF2	116	TINT_GP53
9	TINT_GPL0	45	TINT_GPA2	81	TINT_GP00	117	TINT_GP54
10	TINT_GPL1	46	TINT_GPA3	82	TINT_GP01	118	TINT_GP55
11	TINT_GPL2	47	TINT_GPA4	83	TINT_GP02	119	TINT_GP56
12	TINT_GPL3	48	TINT_GPA5	84	TINT_GP03	120	TINT_GP60
13	TINT_GPM0	49	TINT_GPA6	85	TINT_GP04	121	TINT_GP61
14	TINT_GPM1	50	TINT_GPA7	86	TINT_GP05	122	TINT_GP62
15	TINT_GPM2	51	TINT_GPB0	87	TINT_GP06	123	TINT_GP63
16	TINT_GPM3	52	TINT_GPB1	88	TINT_GP07	124	TINT_GP64
17	TINT_GPM4	53	TINT_GPB2	89	TINT_GP10	125	TINT_GP65
18	TINT_GPM5	54	TINT_GPB3	90	TINT_GP11	126	TINT_GP66
19	TINT_GPM6	55	TINT_GPB4	91	TINT_GP12	127	TINT_GP70
20	TINT_GPM7	56	TINT_GPB5	92	TINT_GP13	128	TINT_GP71
21	TINT_GPL4	57	TINT_GPB6	93	TINT_GP14	129	TINT_GP72
22	TINT_GPL5	58	TINT_GPB7	94	TINT_GP15	130	TINT_GP73
23	TINT_GPL6	59	TINT_GPC0	95	TINT_GP16	131	TINT_GP74
24	TINT_GPL7	60	TINT_GPC1	96	TINT_GP17	132	TINT_GP75
25	TINT_GPG0	61	TINT_GPC2	97	TINT_GP30	133	TINT_GP76
26	TINT_GPG1	62	TINT_GPD0	98	TINT_GP31	134	TINT_GP77
27	TINT_GPG2	63	TINT_GPD1	99	TINT_GP32	135	TINT_GP80
28	TINT_GPG3	64	TINT_GPD2	100	TINT_GP33	136	TINT_GP81
29	TINT_GPG4	65	TINT_GPD3	101	TINT_GP34	137	TINT_GP82
30	TINT_GPG5	66	TINT_GPD4	102	TINT_GP35	138	TINT_GP83
31	TINT_GPG6	67	TINT_GPD5	103	TINT_GP36	139	TINT_GP84
32	TINT_GPG7	68	TINT_GPD6	104	TINT_GP37	140	TINT_GP85
33	TINT_GPH0	69	TINT_GPD7	105	TINT_GP40		
34	TINT_GPH1	70	TINT_GPE0	106	TINT_GP41		
35	TINT_GPH2	71	TINT_GPE1	107	TINT_GP42		

(11) DREQ Status Register (ICU_DRCTR)

[Secure group: Gr1]

This register indicates the states of the DREQn interrupts.

Access Size : 32 bits

Address : <ICU_base> + 0050h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	DRSTAT4	DRSTAT3	DRSTAT2	DRSTAT1	DRSTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	DRSTAT4	0h	R	Indicates the detection of a DREQ4 interrupt. 0b: No DREQn detected 1b: DREQn detected
3	DRSTAT3	0h	R	Indicates the detection of a DREQ3 interrupt. 0b: No DREQn detected 1b: DREQn detected
2	DRSTAT2	0h	R	Indicates the detection of a DREQ2 interrupt. 0b: No DREQn detected 1b: DREQn detected
1	DRSTAT1	0h	R	Indicates the detection of a DREQ1 interrupt. 0b: No DREQn detected 1b: DREQn detected
0	DRSTAT0	0h	R	Indicates the detection of a DREQ0 interrupt. 0b: No DREQn detected 1b: DREQn detected

(12) DREQ Status Clear Register (ICU_DRCLR)

[Secure group: Gr1]

This register clears the state flags for the DREQn interrupts.

Remark When the DREQ interrupts are to be used, use this register to clear the DREQ state flags before doing so. Clear the status flag after setting the ICU_DRTSR register.

Access Size : 32 bits

Address : <ICU_base> + 0054h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	DRCLR4	DRCLR3	DRCLR2	DRCLR1	DRCLR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4	DRCLR4	0h	W	The read value is undefined Clears the error factor of the event allocated in the corresponding bit by writing 1.
3	DRCLR3	0h	W	The read value is undefined Clears the error factor of the event allocated in the corresponding bit by writing 1.
2	DRCLR2	0h	W	The read value is undefined Clears the error factor of the event allocated in the corresponding bit by writing 1.
1	DRCLR1	0h	W	The read value is undefined Clears the error factor of the event allocated in the corresponding bit by writing 1.
0	DRCLR0	0h	W	The read value is undefined Clears the error factor of the event allocated in the corresponding bit by writing 1.

(13) DREQ Detection Method Selection Register (ICU_DRTSR)

[Secure group: Gr1]

This register controls the method of detecting the DREQ interrupts.

Remark 1. When the DREQ interrupts are to be used, set up this register and make the setting for starting the acceptance of DMAC requests before doing so.

Remark 2. This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read-modify-write operation).

Access Size : 32 bits
Address : <ICU_base> + 0058h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	DRITSEL4[1:0]		DRITSEL3[1:0]		DRITSEL2[1:0]		DRITSEL1[1:0]		DRITSEL0[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9, 8	DRITSEL4[1:0]	0h	RW	The interrupt detection method for DREQ4 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
7, 6	DRITSEL3[1:0]	0h	RW	The interrupt detection method for DREQ3 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
5, 4	DRITSEL2[1:0]	0h	RW	The interrupt detection method for DREQ2 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
3, 2	DRITSEL1[1:0]	0h	RW	The interrupt detection method for DREQ1 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection
1, 0	DRITSEL0[1:0]	0h	RW	The interrupt detection method for DREQ0 is selected. 00b: Low-level detection 01b: Falling edge detection 10b: Rising edge detection 11b: Falling edge/rising edge detection

(14) IP Detection Method Selection Register (ICU_IPTSR)

[Secure group: Gr1]

This register controls the method of detecting interrupt signals from GBETH0 and GBETH1.

Remark 1. This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read-modify-write operation).

For more information, refer to the User's Manual Additional Document.

(15) Bus Error Interrupt Status Register k (ICU_BEISRk) (k =0 to 3)

[Secure group: Gr0]

These registers indicate the state of the bus error factors. Even one factor having been detected keeps the bus error interrupt signal (BUS_ERR_INT) of the ICU at the active level.

These registers are reset by the ERROR_RESETn signal. They are not reset by the PRESETn_I signal.

Access Size : 32 bits

Address : <ICU_base> + 0070h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BESTATk_31	BESTATk_30	BESTATk_29	BESTATk_28	BESTATk_27	BESTATk_26	BESTATk_25	BESTATk_24	BESTATk_23	BESTATk_22	BESTATk_21	BESTATk_20	BESTATk_19	BESTATk_18	BESTATk_17	BESTATk_16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BESTATk_15	BESTATk_14	BESTATk_13	BESTATk_12	BESTATk_11	BESTATk_10	BESTATk_9	BESTATk_8	BESTATk_7	BESTATk_6	BESTATk_5	BESTATk_4	BESTATk_3	BESTATk_2	BESTATk_1	BESTATk_0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BESTATk_n	0h	R	Indicates the detection of a bus error interrupt. 0b: No BUSERR_INTn detected 1b: BUSERR_INTn detected

Table 4.6-5 Relationship between Bus Error Signal and BESTATk_n (1/2)

Bus Error Signal	BESTATk_n	
ERRINT_BRG_CM33_C	n = 0	k = 0
ERRINT_BRG_CM33_S	n = 1	k = 0
BUS_ERR_INTR_W_SXA55M	n = 2	k = 0
BUS_ERR_INTR_R_SXA55M	n = 3	k = 0
BUS_ERR_INTR_W_SXA55P	n = 4	k = 0
BUS_ERR_INTR_R_SXA55P	n = 5	k = 0
BUS_ERR_INTR_W_SXGPU	n = 6	k = 0
BUS_ERR_INTR_R_SXGPU	n = 7	k = 0
BUS_ERR_INTR_W_SXADMC0	n = 8	k = 0
BUS_ERR_INTR_R_SXADMC0	n = 9	k = 0
BUS_ERR_INTR_W_SXADMC1	n = 10	k = 0
BUS_ERR_INTR_R_SXADMC1	n = 11	k = 0
BUS_ERR_INTR_W_SXCSTE	n = 12	k = 0
BUS_ERR_INTR_R_SXCSTE	n = 13	k = 0
BUS_ERR_INTR_W_SXCSTA	n = 14	k = 0
BUS_ERR_INTR_R_SXCSTA	n = 15	k = 0
BUS_ERR_INTR_W_SXM33S	n = 16	k = 0
BUS_ERR_INTR_R_SXM33S	n = 17	k = 0
BUS_ERR_INTR_W_SXM33C	n = 18	k = 0
BUS_ERR_INTR_R_SXM33C	n = 19	k = 0
BUS_ERR_INTR_W_SXMDMC	n = 20	k = 0

Bus Error Signal	BESTATk_n	
BUS_ERR_INTR_R_SXMDMC	n = 21	k = 0
Reserved	n = 22	k = 0
Reserved	n = 23	k = 0
Reserved	n = 24	k = 0
Reserved	n = 25	k = 0
Reserved	n = 26	k = 0
Reserved	n = 27	k = 0
BUS_ERR_INTR_W_SXRDMC0	n = 28	k = 0
BUS_ERR_INTR_R_SXRDMC0	n = 29	k = 0
BUS_ERR_INTR_W_SXRDMC1	n = 30	k = 0
BUS_ERR_INTR_R_SXRDMC1	n = 31	k = 0
BUS_ERR_INTR_W_SXSD0	n = 0	k = 1
BUS_ERR_INTR_R_SXSD0	n = 1	k = 1
BUS_ERR_INTR_W_SXSD1	n = 2	k = 1
BUS_ERR_INTR_R_SXSD1	n = 3	k = 1
BUS_ERR_INTR_W_SXSD2	n = 4	k = 1
BUS_ERR_INTR_R_SXSD2	n = 5	k = 1
BUS_ERR_INTR_W_SXGBE0	n = 6	k = 1
BUS_ERR_INTR_R_SXGBE0	n = 7	k = 1
BUS_ERR_INTR_W_SXGBE1	n = 8	k = 1
BUS_ERR_INTR_R_SXGBE1	n = 9	k = 1

Table 4.6-5 Relationship between Bus Error Signal and BESTATk_n (2/2)

Bus Error Signal	BESTATk_n	
BUS_ERR_INTR_W_SHU2H0	n = 10	k = 1
BUS_ERR_INTR_R_SHU2H0	n = 11	k = 1
BUS_ERR_INTR_W_SHU2F	n = 12	k = 1
BUS_ERR_INTR_R_SHU2F	n = 13	k = 1
BUS_ERR_INTR_W_SHU2H1	n = 14	k = 1
BUS_ERR_INTR_R_SHU2H1	n = 15	k = 1
BUS_ERR_INTR_W_SXU3H0	n = 16	k = 1
BUS_ERR_INTR_R_SXU3H0	n = 17	k = 1
Reserved	n = 18	k = 1
Reserved	n = 19	k = 1
BUS_ERR_INTR_W_SXPCI0	n = 20	k = 1
BUS_ERR_INTR_R_SXPCI0	n = 21	k = 1
Reserved	n = 22	k = 1
Reserved	n = 23	k = 1
BUS_ERR_INTR_W_SXCRV0	n = 24	k = 1
Reserved	n = 25	k = 1
BUS_ERR_INTR_W_SXCRS0	n = 26	k = 1
Reserved	n = 27	k = 1
Reserved	n = 28	k = 1
Reserved	n = 29	k = 1
Reserved	n = 30	k = 1
Reserved	n = 31	k = 1
Reserved	n = 0	k = 2
Reserved	n = 1	k = 2
Reserved	n = 2	k = 2
Reserved	n = 3	k = 2
Reserved	n = 4	k = 2
Reserved	n = 5	k = 2
Reserved	n = 6	k = 2
Reserved	n = 7	k = 2
Reserved	n = 8	k = 2
Reserved	n = 9	k = 2
BUS_ERR_INTR_W_SXLCD0	n = 10	k = 2
BUS_ERR_INTR_R_SXLCD0	n = 11	k = 2
BUS_ERR_INTR_W_SXDSI	n = 12	k = 2
BUS_ERR_INTR_R_SXDSI	n = 13	k = 2
Reserved	n = 14	k = 2
Reserved	n = 15	k = 2
BUS_ERR_INTR_W_SXVCP	n = 16	k = 2
BUS_ERR_INTR_R_SXVCP	n = 17	k = 2
Reserved	n = 18	k = 2
Reserved	n = 19	k = 2
Reserved	n = 20	k = 2

Bus Error Signal	BESTATk_n	
Reserved	n = 21	k = 2
Reserved	n = 22	k = 2
Reserved	n = 23	k = 2
Reserved	n = 24	k = 2
Reserved	n = 25	k = 2
Reserved	n = 26	k = 2
Reserved	n = 27	k = 2
Reserved	n = 28	k = 2
Reserved	n = 29	k = 2
BUS_TZC0INT_DDR_TZCDDR0*1	n = 30	k = 2
BUS_TZC1INT_DDR_TZCDDR0*1	n = 31	k = 2
Reserved	n = 0	k = 3
Reserved	n = 1	k = 3
Reserved	n = 2	k = 3
BUS_TZCINT_PCI*1	n = 3	k = 3
BUS_TZCINT_MSRM*1	n = 4	k = 3
BUS_TZCINT_XSP*1	n = 5	k = 3
BUS_TZCINT_ACRCB*1	n = 6	k = 3
BUS_ERR_INTR_W_SXLCD1	n = 7	k = 3
BUS_ERR_INTR_R_SXLCD1	n = 8	k = 3
BUS_ERR_INTR_W_SXVSP	n = 9	k = 3
BUS_ERR_INTR_R_SXVSP	n = 10	k = 3
BUS_ERR_INTR_W_SXNPU	n = 11	k = 3
BUS_ERR_INTR_R_SXNPU	n = 12	k = 3
Reserved	n = 13	k = 3
BUS_ERR_INTR_R_SXNPUR	n = 14	k = 3
BUS_ERR_INTR_W_SXFDP1	n = 15	k = 3
BUS_ERR_INTR_R_SXFDP1	n = 16	k = 3
Reserved	n = 17 to 31	k = 3

Note 1. To clear the TZC interrupt, clear the BEISRk register in the interrupt controller after clearing the source in the TZC.

(16) Bus Error Interrupt Status Clear Register k (ICU_BECLRk) (k = 0 to 3)

[Secure group: Gr0]

These registers control the bus error factors.

Remark Clear the factor as the source of the bus error before clearing its state flag in the register.

Access Size : 32 bits

Address : <ICU_base> + 0080h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BECLRk_31	BECLRk_30	BECLRk_29	BECLRk_28	BECLRk_27	BECLRk_26	BECLRk_25	BECLRk_24	BECLRk_23	BECLRk_22	BECLRk_21	BECLRk_20	BECLRk_19	BECLRk_18	BECLRk_17	BECLRk_16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BECLRk_15	BECLRk_14	BECLRk_13	BECLRk_12	BECLRk_11	BECLRk_10	BECLRk_9	BECLRk_8	BECLRk_7	BECLRk_6	BECLRk_5	BECLRk_4	BECLRk_3	BECLRk_2	BECLRk_1	BECLRk_0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BECLRk_n	0h	W	The read value is undefined When BESTATk_n = 1 1b: The BUSERR_INTn interrupt detection is cleared (the status changes from "detected" to "not detected"). 0b: Writing has no effect. When BESTATk_n = 0 Writing has no effect.

(17) ECCRAM Error Interrupt Status Register k (ICU_EREISRk) (k = 1, 4)

[Secure group: Gr0]

These registers control SRAM errors. Even one factor having been detected keeps the RAM error interrupt signal (RAM_ERR_INT) of the ICU at the active level.

These registers are reset by the ERROR_RESETn signal. They are not reset by the PRESETn_I signal.

Access Size : 32 bits

Address : <ICU_base> + 0090h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	OFSTATk_7	OFSTATk_6	OFSTATk_5	OFSTATk_4	OFSTATk_3	OFSTATk_2	OFSTATk_1	OFSTATk_0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	2ESTATk_7	2ESTATk_6	2ESTATk_5	2ESTATk_4	2ESTATk_3	2ESTATk_2	2ESTATk_1	2ESTATk_0	1ESTATk_7	1ESTATk_6	1ESTATk_5	1ESTATk_4	1ESTATk_3	1ESTATk_2	1ESTATk_1	1ESTATk_0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	OFSTATk_[7:0]	0h	R	Indicates the detection of an SRAM error. Refer to **Table 4.6-6** for the correspondence with the error signal. 0b: No SRAM error detected 1b: SRAM error detected
15 to 8	2ESTATk_[7:0]	0h	R	Indicates the detection of an SRAM error. Refer to **Table 4.6-6** for the correspondence with the error signal. 0b: No SRAM error detected 1b: SRAM error detected
7 to 0	1ESTATk_[7:0]	0h	R	Indicates the detection of an SRAM error. Refer to **Table 4.6-6** for the correspondence with the error signal. 0b: No SRAM error detected 1b: SRAM error detected

Table 4.6-6 Relationship between SRAM Error Signal and EREISRk

SRAM Error Signal	EREISRk	
SRAM0_EC7TIE1[1:0]	0 and 1 bits	k = 1
Reserved	2 to 7 bits	k = 1
SRAM0_EC7TIE2[1:0]	8 and 9 bits	k = 1
Reserved	10 to 15 bits	k = 1
SRAM0_EC7TIOVF[1:0]	16 and 17 bits	k = 1
Reserved	18 to 23 bits	k = 1

SRAM Error Signal	EREISRk	
SRAM1_EC7TIE1[7:0]	0 to 7 bits	k = 4
SRAM1_EC7TIE2[7:0]	8 to 15 bits	k = 4
SRAM1_EC7TIOVF[7:0]	16 to 23 bits	k = 4

(18) ECCRAM Error Interrupt Status Clear Register k (ICU_ERCLRk) (k = 1, 4)

[Secure group: Gr0]

These registers clear the SRAM error factors.

Remark Clear the factor as the source of the SRAM error before clearing its state flag in the register.

Access Size : 32 bits

Address : <ICU_base> + 00E0h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	OFCLRk_7	OFCLRk_6	OFCLRk_5	OFCLRk_4	OFCLRk_3	OFCLRk_2	OFCLRk_1	OFCLRk_0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	2ECLRk_7	2ECLRk_6	2ECLRk_5	2ECLRk_4	2ECLRk_3	2ECLRk_2	2ECLRk_1	2ECLRk_0	1ECLRk_7	1ECLRk_6	1ECLRk_5	1ECLRk_4	1ECLRk_3	1ECLRk_2	1ECLRk_1	1ECLRk_0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	OFCLRk_[7:0]	0h	W	The read value is undefined When an SRAM error has occurred 1b: The interrupt detection is cleared (the status changes from "detected" to "not detected"). 0b: Writing has no effect. When an SRAM error has not occurred, writing has no effect.
15 to 8	2ECLRk_[7:0]	0h	W	The read value is undefined When an SRAM error has occurred 1b: The interrupt detection is cleared (the status changes from "detected" to "not detected"). 0b: Writing has no effect. When an SRAM error has not occurred, writing has no effect.
7 to 0	1ECLRk_[7:0]	0h	W	The read value is undefined When an SRAM error has occurred 1b: The interrupt detection is cleared (the status changes from "detected" to "not detected"). 0b: Writing has no effect. When an SRAM error has not occurred, writing has no effect.

(19) SW Interrupt Generation Register (ICU_SWINT)

[Secure group: Gr1]

This register generates software interrupts. Writing 1b to the bit corresponding to a given event generates the corresponding software interrupt from the ICU to the CPU.

Access Size : 32 bits
Address : <ICU_base> + 0130h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	IM333	IM332	IM331	IM330
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	IA553	IA552	IA551	IA550
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
19	IM333	0h	W	The read value is undefined Software interrupt INTM33_3 to CM33 is generated by writing 1b.
18	IM332	0h	W	The read value is undefined Software interrupt INTM33_2 to CM33 is generated by writing 1b.
17	IM331	0h	W	The read value is undefined Software interrupt INTM33_1 to CM33 is generated by writing 1b.
16	IM330	0h	W	The read value is undefined Software interrupt INTM33_0 to CM33 is generated by writing 1b.
15 to 4	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
3	IA553	0h	W	The read value is undefined Software interrupt INTA55_3 to CA55 is generated by writing 1b.
2	IA552	0h	W	The read value is undefined Software interrupt INTA55_2 to CA55 is generated by writing 1b.
1	IA551	0h	W	The read value is undefined Software interrupt INTA55_1 to CA55 is generated by writing 1b.
0	IA550	0h	W	The read value is undefined Software interrupt INTA55_0 to CA55 is generated by writing 1b.

(20) SELECT Interrupt CM33 Selection Register k (ICU_INTM33SELk) (k = 0 to 42)

[Secure group: Gr0]

These registers select the sources of interrupts to the CM33 which belong to “SELECT”. The input sources can be selected per CM33 SPI from among those allocated to “SELECT” under the “CM33” column in **Table 4.6-24**. Writing the value equivalent to the number obtained by “SEL + number” under the “CM33 SPI” column in **Table 4.6-24** to the register selects the corresponding event. The inactive value 0 is input to the CM33 SPI in response to any sources that are not selected.

Values in **Table 4.6-24** are decimal numbers.

127 interrupt signals are specifiable. No register bits are allocated to the M33SPIk_SEL1 and M33SPIk_SEL2 signals of the INTM33SEL42 register.

Remark This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read–modify–write operation).

Access Size : 32 bits

Address : <ICU_base> + 0200h + k x 0004h

Initial Value : 3FFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	M33SPIk_SEL2[9:0]										M33SPIk_SEL1[9:6]			
Initial Value	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	M33SPIk_SEL1[5:0]						M33SPIk_SEL0[9:0]									
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29 to 20	M33SPIk_SEL2 [9:0]	3FFh	RW	The source of the event signal input to CM33 SPI No:y+2 is selected. 0h to 246h: A signal which corresponds to the number listed in "SEL + number" is output. 247h to 3FFh: Not available
19 to 10	M33SPIk_SEL1 [9:0]	3FFh	RW	The source of the event signal input to CM33 SPI No:y+1 is selected. 0h to 246h: A signal which corresponds to the number listed in "SEL + number" is output. 247h to 3FFh: Not available
9 to 0	M33SPIk_SEL0 [9:0]	3FFh	RW	The source of the event signal input to CM33 SPI No:y is selected. 0h to 246h: A signal which corresponds to the number listed in "SEL + number" is output. 247h to 3FFh: Not available

y = 3 × k + 353

(21) Error Interrupt CM33 Status Register k (ICU_ERINTM33CTLk) (k = 0 to 3)

[Secure group: Gr0]

These registers hold the source flags for error interrupts to the CM33 when they have occurred. Even one factor having been detected keeps the error interrupt signal (ICU_Error_CM33) to the CM33 at the active level.

The registers do not hold the source flags for interrupt signals that have been masked by the settings of the ERINTM33MSKk registers.

The individual sources prior to their bundling for BUS_ERR_INT and RAM_ERR_INT are indicated in the BEISRk and BEISRk registers, respectively. Therefore, these registers do not hold the source flags for BUS_ERR_INT and RAM_ERR_INT. The values for BUS_ERR_INT and RAM_ERR_INT are readable.

These registers are reset by the ERROR_RESETn signal. They are not reset by the PRESETn_I signal.

Access Size : 32 bits

Address : <ICU_base> + 0304h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ERRM3kSTAT31	ERRM3kSTAT30	ERRM3kSTAT29	ERRM3kSTAT28	ERRM3kSTAT27	ERRM3kSTAT26	ERRM3kSTAT25	ERRM3kSTAT24	ERRM3kSTAT23	ERRM3kSTAT22	ERRM3kSTAT21	ERRM3kSTAT20	ERRM3kSTAT19	ERRM3kSTAT18	ERRM3kSTAT17	ERRM3kSTAT16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRM3kSTAT15	ERRM3kSTAT14	ERRM3kSTAT13	ERRM3kSTAT12	ERRM3kSTAT11	ERRM3kSTAT10	ERRM3kSTAT9	ERRM3kSTAT8	ERRM3kSTAT7	ERRM3kSTAT6	ERRM3kSTAT5	ERRM3kSTAT4	ERRM3kSTAT3	ERRM3kSTAT2	ERRM3kSTAT1	ERRM3kSTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ERRM3kSTATn	0h	R	State of error interrupt k - n 0b: No error interrupt detected 1b: Error interrupt detected Writing has no effect.

Table 4.6-7 Relationship between Error Interrupt Signal, ERRkSTATn, and ERRM3kMKn (1/3)

No.	Error Interrupt Signal	ERRM3kSTATn, ERRM3kMKn	
0	BUS_ERR_INT	n = 0	k = 0
1	RAM_ERR_INT	n = 1	k = 0
2	ACPU_nFAULTIRQ_0	n = 2	k = 0
3	ACPU_nFAULTIRQ_1	n = 3	k = 0
4	ACPU_nFAULTIRQ_2	n = 4	k = 0
5	ACPU_nFAULTIRQ_3	n = 5	k = 0
6	ACPU_nFAULTIRQ_4	n = 6	k = 0
7	ACPU_nERRIRQ_0	n = 7	k = 0
8	ACPU_nERRIRQ_1	n = 8	k = 0
9	ACPU_nERRIRQ_2	n = 9	k = 0
10	ACPU_nERRIRQ_3	n = 10	k = 0
11	ACPU_nERRIRQ_4	n = 11	k = 0
12	Reserved	n = 12	k = 0

No.	Error Interrupt Signal	ERRM3kSTATn, ERRM3kMKn	
13	Reserved	n = 13	k = 0
14	Reserved	n = 14	k = 0
15	Reserved	n = 15	k = 0
16	Reserved	n = 16	k = 0
17	Reserved	n = 17	k = 0
18	Reserved	n = 18	k = 0
19	Reserved	n = 19	k = 0
20	Reserved	n = 20	k = 0
21	Reserved	n = 21	k = 0
22	Reserved	n = 22	k = 0
23	Reserved	n = 23	k = 0
24	Reserved	n = 24	k = 0
25	MCPU_LOCKUP	n = 25	k = 0

Table 4.6-7 Relationship between Error Interrupt Signal, ERRkSTATn, and ERRM3kMKn (2/3)

No.	Error Interrupt Signal	ERRM3kSTATn, ERRM3kMKn	
26	—	n = 26	k = 0
27	—	n = 27	k = 0
28	Reserved	n = 28	k = 0
29	Reserved	n = 29	k = 0
30	WDT_CA55_iwdt_nmiundf_n	n = 30	k = 0
31	WDT_CM33_iwdt_nmiundf_n	n = 31	k = 0
32	WDT_OTHER_0_iwdt_nmiundf_n	n = 0	k = 1
33	WDT_OTHER_1_iwdt_nmiundf_n	n = 1	k = 1
34	ada_adereq_n	n = 2	k = 1
35	Reserved	n = 3	k = 1
36	Reserved	n = 4	k = 1
37	Reserved	n = 5	k = 1
38	Reserved	n = 6	k = 1
39	Reserved	n = 7	k = 1
40	Reserved	n = 8	k = 1
41	Reserved	n = 9	k = 1
42	Reserved	n = 10	k = 1
43	Reserved	n = 11	k = 1
44	Reserved	n = 12	k = 1
45	Reserved	n = 13	k = 1
46	Reserved	n = 14	k = 1
47	Reserved	n = 15	k = 1
48	Reserved	n = 16	k = 1
49	Reserved	n = 17	k = 1
50	GPT_U0_gpt_gtciv_n_0	n = 18	k = 1
51	GPT_U0_gpt_gtciv_n_1	n = 19	k = 1
52	GPT_U0_gpt_gtciv_n_2	n = 20	k = 1
53	GPT_U0_gpt_gtciv_n_3	n = 21	k = 1
54	GPT_U0_gpt_gtciv_n_4	n = 22	k = 1
55	GPT_U0_gpt_gtciv_n_5	n = 23	k = 1
56	GPT_U0_gpt_gtciv_n_6	n = 24	k = 1
57	GPT_U0_gpt_gtciv_n_7	n = 25	k = 1
58	GPT_U0_gpt_gtciu_n_0	n = 26	k = 1
59	GPT_U0_gpt_gtciu_n_1	n = 27	k = 1
60	GPT_U0_gpt_gtciu_n_2	n = 28	k = 1
61	GPT_U0_gpt_gtciu_n_3	n = 29	k = 1
62	GPT_U0_gpt_gtciu_n_4	n = 30	k = 1
63	GPT_U0_gpt_gtciu_n_5	n = 31	k = 1
64	GPT_U0_gpt_gtciu_n_6	n = 0	k = 2
65	GPT_U0_gpt_gtciu_n_7	n = 1	k = 2
66	GPT_U0_gpt_gtdei_n_0	n = 2	k = 2
67	GPT_U0_gpt_gtdei_n_1	n = 3	k = 2
68	GPT_U0_gpt_gtdei_n_2	n = 4	k = 2
69	GPT_U0_gpt_gtdei_n_3	n = 5	k = 2
70	GPT_U0_gpt_gtdei_n_4	n = 6	k = 2
71	GPT_U0_gpt_gtdei_n_5	n = 7	k = 2
72	GPT_U0_gpt_gtdei_n_6	n = 8	k = 2
73	GPT_U0_gpt_gtdei_n_7	n = 9	k = 2
74	GPT_U1_gpt_gtciv_n_0	n = 10	k = 2
75	GPT_U1_gpt_gtciv_n_1	n = 11	k = 2
76	GPT_U1_gpt_gtciv_n_2	n = 12	k = 2
77	GPT_U1_gpt_gtciv_n_3	n = 13	k = 2
78	GPT_U1_gpt_gtciv_n_4	n = 14	k = 2
79	GPT_U1_gpt_gtciv_n_5	n = 15	k = 2
80	GPT_U1_gpt_gtciv_n_6	n = 16	k = 2
81	GPT_U1_gpt_gtciv_n_7	n = 17	k = 2
82	GPT_U1_gpt_gtciu_n_0	n = 18	k = 2
83	GPT_U1_gpt_gtciu_n_1	n = 19	k = 2
84	GPT_U1_gpt_gtciu_n_2	n = 20	k = 2
85	GPT_U1_gpt_gtciu_n_3	n = 21	k = 2
86	GPT_U1_gpt_gtciu_n_4	n = 22	k = 2
87	GPT_U1_gpt_gtciu_n_5	n = 23	k = 2
88	GPT_U1_gpt_gtciu_n_6	n = 24	k = 2
89	GPT_U1_gpt_gtciu_n_7	n = 25	k = 2
90	GPT_U1_gpt_gtdei_n_0	n = 26	k = 2
91	GPT_U1_gpt_gtdei_n_1	n = 27	k = 2
92	GPT_U1_gpt_gtdei_n_2	n = 28	k = 2
93	GPT_U1_gpt_gtdei_n_3	n = 29	k = 2
94	GPT_U1_gpt_gtdei_n_4	n = 30	k = 2
95	GPT_U1_gpt_gtdei_n_5	n = 31	k = 2
96	GPT_U1_gpt_gtdei_n_6	n = 0	k = 3
97	GPT_U1_gpt_gtdei_n_7	n = 1	k = 3

Table 4.6-7 Relationship between Error Interrupt Signal, ERRkSTATn, and ERRM3kMKn (3/3)

No.	Error Interrupt Signal	ERRM3kSTATn, ERRM3kMKn	
98	SW_PSEUDO_ERR_0	n = 2	k = 3
99	SW_PSEUDO_ERR_1	n = 3	k = 3
100	SW_PSEUDO_ERR_2	n = 4	k = 3
101	SW_PSEUDO_ERR_3	n = 5	k = 3
102	SW_PSEUDO_ERR_4	n = 6	k = 3
103	SW_PSEUDO_ERR_5	n = 7	k = 3
104	SW_PSEUDO_ERR_6	n = 8	k = 3
105	SW_PSEUDO_ERR_7	n = 9	k = 3
106	SW_PSEUDO_ERR_8	n = 10	k = 3
107	SW_PSEUDO_ERR_9	n = 11	k = 3
108	SW_PSEUDO_ERR_10	n = 12	k = 3
109	SW_PSEUDO_ERR_11	n = 13	k = 3
110	SW_PSEUDO_ERR_12	n = 14	k = 3
111	SW_PSEUDO_ERR_13	n = 15	k = 3
112	SW_PSEUDO_ERR_14	n = 16	k = 3
113	SW_PSEUDO_ERR_15	n = 17	k = 3
114	Reserved	n = 18	k = 3
115	Reserved	n = 19	k = 3
116	Reserved	n = 20	k = 3
117	Reserved	n = 21	k = 3
118	Reserved	n = 22	k = 3
119	Reserved	n = 23	k = 3
120	Reserved	n = 24	k = 3
121	Reserved	n = 25	k = 3
122	Reserved	n = 26	k = 3
123	Reserved	n = 27	k = 3
124	Reserved	n = 28	k = 3
125	Reserved	n = 29	k = 3
126	Reserved	n = 30	k = 3
127	Reserved	n = 31	k = 3

(22) Error Interrupt CM33 Status Clear Register k (ICU_ERINTM33CLRk) (k = 0 to 3)

[Secure group: Gr0]

These registers clear the state flags for factors of error interrupts to the CM33.

Remark The ERINTM33CTLk registers do not cover BUS_ERR_INT and RAM_ERR_INT so these registers do not clear them.

Access Size : 32 bits

Address : <ICU_base> + 0314h (k = 0)
<ICU_base> + 031Ch (k = 1)
<ICU_base> + 0320h (k = 2)
<ICU_base> + 0324h (k = 3)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ERRM3kCLR31	ERRM3kCLR30	ERRM3kCLR29	ERRM3kCLR28	ERRM3kCLR27	ERRM3kCLR26	ERRM3kCLR25	ERRM3kCLR24	ERRM3kCLR23	ERRM3kCLR22	ERRM3kCLR21	ERRM3kCLR20	ERRM3kCLR19	ERRM3kCLR18	ERRM3kCLR17	ERRM3kCLR16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRM3kCLR15	ERRM3kCLR14	ERRM3kCLR13	ERRM3kCLR12	ERRM3kCLR11	ERRM3kCLR10	ERRM3kCLR9	ERRM3kCLR8	ERRM3kCLR7	ERRM3kCLR6	ERRM3kCLR5	ERRM3kCLR4	ERRM3kCLR3	ERRM3kCLR2	ERRM3kCLR1	ERRM3kCLR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ERRM3kCLRn	0h	W	The read value is undefined Clears the error interrupt detection factor allocated in the corresponding bit by writing 1b. If the value is 0b, it does not change (when read, 0b is always read).

Table 4.6-8 Relationship between Error Interrupt Signal and ERRM3kCLRn (1/3)

No.	Error Interrupt Signal	ERRM3kCLRn
0	ACPU_nFAULTIRQ_0	n = 0 k = 0
1	ACPU_nFAULTIRQ_1	n = 1 k = 0
2	ACPU_nFAULTIRQ_2	n = 2 k = 0
3	ACPU_nFAULTIRQ_3	n = 3 k = 0
4	ACPU_nFAULTIRQ_4	n = 4 k = 0
5	ACPU_nERRIRQ_0	n = 5 k = 0
6	ACPU_nERRIRQ_1	n = 6 k = 0
7	ACPU_nERRIRQ_2	n = 7 k = 0
8	ACPU_nERRIRQ_3	n = 8 k = 0
9	ACPU_nERRIRQ_4	n = 9 k = 0
10	Reserved	n = 10 k = 0
11	Reserved	n = 11 k = 0
12	Reserved	n = 12 k = 0
13	Reserved	n = 13 k = 0
14	Reserved	n = 14 k = 0
15	Reserved	n = 15 k = 0
16	Reserved	n = 16 k = 0

No.	Error Interrupt Signal	ERRM3kCLRn
18	Reserved	n = 18 k = 0
19	Reserved	n = 19 k = 0
20	Reserved	n = 20 k = 0
21	Reserved	n = 21 k = 0
22	Reserved	n = 22 k = 0
23	MCPU_LOCKUP	n = 23 k = 0
24	—	n = 24 k = 0
25	—	n = 25 k = 0
26	Reserved	n = 26 k = 0
27	Reserved	n = 27 k = 0
28	WDT_CA55_iwdt_nmiundf_n	n = 28 k = 0
29	WDT_CM33_iwdt_nmiundf_n	n = 29 k = 0
30	WDT_OTHER_0_iwdt_nmiundf_n	n = 30 k = 0
31	WDT_OTHER_1_iwdt_nmiundf_n	n = 31 k = 0
32	ada_adereq_n	n = 0 k = 1
33	Reserved	n = 1 k = 1
34	Reserved	n = 2 k = 1

Table 4.6-8 Relationship between Error Interrupt Signal and ERRM3kCLRn (2/3)

No.	Error Interrupt Signal	ERRM3kCLRn
17	RCPU_ICEBEMPTY_1	n = 17 k = 0
36	Reserved	n = 4 k = 1
37	Reserved	n = 5 k = 1
38	Reserved	n = 6 k = 1
39	Reserved	n = 7 k = 1
40	Reserved	n = 8 k = 1
41	Reserved	n = 9 k = 1
42	Reserved	n = 10 k = 1
43	Reserved	n = 11 k = 1
44	Reserved	n = 12 k = 1
45	Reserved	n = 13 k = 1
46	Reserved	n = 14 k = 1
47	Reserved	n = 15 k = 1
48	GPT_U0_gpt_gtciv_n_0	n = 16 k = 1
49	GPT_U0_gpt_gtciv_n_1	n = 17 k = 1
50	GPT_U0_gpt_gtciv_n_2	n = 18 k = 1
51	GPT_U0_gpt_gtciv_n_3	n = 19 k = 1
52	GPT_U0_gpt_gtciv_n_4	n = 20 k = 1
53	GPT_U0_gpt_gtciv_n_5	n = 21 k = 1
54	GPT_U0_gpt_gtciv_n_6	n = 22 k = 1
55	GPT_U0_gpt_gtciv_n_7	n = 23 k = 1
56	GPT_U0_gpt_gtciu_n_0	n = 24 k = 1
57	GPT_U0_gpt_gtciu_n_1	n = 25 k = 1
58	GPT_U0_gpt_gtciu_n_2	n = 26 k = 1
59	GPT_U0_gpt_gtciu_n_3	n = 27 k = 1
60	GPT_U0_gpt_gtciu_n_4	n = 28 k = 1
61	GPT_U0_gpt_gtciu_n_5	n = 29 k = 1
62	GPT_U0_gpt_gtciu_n_6	n = 30 k = 1
63	GPT_U0_gpt_gtciu_n_7	n = 31 k = 1
64	GPT_U0_gpt_gtdei_n_0	n = 0 k = 2
65	GPT_U0_gpt_gtdei_n_1	n = 1 k = 2
66	GPT_U0_gpt_gtdei_n_2	n = 2 k = 2
67	GPT_U0_gpt_gtdei_n_3	n = 3 k = 2
68	GPT_U0_gpt_gtdei_n_4	n = 4 k = 2
69	GPT_U0_gpt_gtdei_n_5	n = 5 k = 2
70	GPT_U0_gpt_gtdei_n_6	n = 6 k = 2
71	GPT_U0_gpt_gtdei_n_7	n = 7 k = 2
72	GPT_U1_gpt_gtciv_n_0	n = 8 k = 2
73	GPT_U1_gpt_gtciv_n_1	n = 9 k = 2

No.	Error Interrupt Signal	ERRM3kCLRn
35	Reserved	n = 3 k = 1
74	GPT_U1_gpt_gtciv_n_2	n = 10 k = 2
75	GPT_U1_gpt_gtciv_n_3	n = 11 k = 2
76	GPT_U1_gpt_gtciv_n_4	n = 12 k = 2
77	GPT_U1_gpt_gtciv_n_5	n = 13 k = 2
78	GPT_U1_gpt_gtciv_n_6	n = 14 k = 2
79	GPT_U1_gpt_gtciv_n_7	n = 15 k = 2
80	GPT_U1_gpt_gtciu_n_0	n = 16 k = 2
81	GPT_U1_gpt_gtciu_n_1	n = 17 k = 2
82	GPT_U1_gpt_gtciu_n_2	n = 18 k = 2
83	GPT_U1_gpt_gtciu_n_3	n = 19 k = 2
84	GPT_U1_gpt_gtciu_n_4	n = 20 k = 2
85	GPT_U1_gpt_gtciu_n_5	n = 21 k = 2
86	GPT_U1_gpt_gtciu_n_6	n = 22 k = 2
87	GPT_U1_gpt_gtciu_n_7	n = 23 k = 2
88	GPT_U1_gpt_gtdei_n_0	n = 24 k = 2
89	GPT_U1_gpt_gtdei_n_1	n = 25 k = 2
90	GPT_U1_gpt_gtdei_n_2	n = 26 k = 2
91	GPT_U1_gpt_gtdei_n_3	n = 27 k = 2
92	GPT_U1_gpt_gtdei_n_4	n = 28 k = 2
93	GPT_U1_gpt_gtdei_n_5	n = 29 k = 2
94	GPT_U1_gpt_gtdei_n_6	n = 30 k = 2
95	GPT_U1_gpt_gtdei_n_7	n = 31 k = 2
96	SW_PSEUDO_ERR_0	n = 0 k = 3
97	SW_PSEUDO_ERR_1	n = 1 k = 3
98	SW_PSEUDO_ERR_2	n = 2 k = 3
99	SW_PSEUDO_ERR_3	n = 3 k = 3
100	SW_PSEUDO_ERR_4	n = 4 k = 3
101	SW_PSEUDO_ERR_5	n = 5 k = 3
102	SW_PSEUDO_ERR_6	n = 6 k = 3
103	SW_PSEUDO_ERR_7	n = 7 k = 3
104	SW_PSEUDO_ERR_8	n = 8 k = 3
105	SW_PSEUDO_ERR_9	n = 9 k = 3
106	SW_PSEUDO_ERR_10	n = 10 k = 3
107	SW_PSEUDO_ERR_11	n = 11 k = 3
108	SW_PSEUDO_ERR_12	n = 12 k = 3
109	SW_PSEUDO_ERR_13	n = 13 k = 3
110	SW_PSEUDO_ERR_14	n = 14 k = 3
111	SW_PSEUDO_ERR_15	n = 15 k = 3

Table 4.6-8 Relationship between Error Interrupt Signal and ERRM3kCLRn (3/3)

No.	Error Interrupt Signal	ERRM3kCLRn
112	Reserved	n = 16 k = 3
113	Reserved	n = 17 k = 3
114	Reserved	n = 18 k = 3
115	Reserved	n = 19 k = 3
116	Reserved	n = 20 k = 3
117	Reserved	n = 21 k = 3
118	Reserved	n = 22 k = 3
119	Reserved	n = 23 k = 3

No.	Error Interrupt Signal	ERRM3kCLRn
120	Reserved	n = 24 k = 3
121	Reserved	n = 25 k = 3
122	Reserved	n = 26 k = 3
123	Reserved	n = 27 k = 3
124	Reserved	n = 28 k = 3
125	Reserved	n = 29 k = 3
126	Reserved	n = 30 k = 3
127	Reserved	n = 31 k = 3

(23) Error Interrupt CM33 Mask Control Register k (ICU_ERINTM33MSKk) (k = 0 to 3)

[Secure group: Gr0]

These registers control whether or not to retain factors of error interrupts to the CM33.

Register bits are allocated to the individual error interrupts. Writing 1b to a bit masks the corresponding error interrupt by preventing retention of the factor.

The interrupt signals are allocated to the register bits in the same way as with **Table 4.6-7**.

Remark 1. The ERINTM33CTLk registers do not cover BUS_ERR_INT and RAM_ERR_INT so these registers do not clear them.

Remark 2. This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read-modify-write operation).

Access Size : 32 bits

Address : <ICU_base> + 0328h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ERRM3 kMK31	ERRM3 kMK30	ERRM3 kMK29	ERRM3 kMK28	ERRM3 kMK27	ERRM3 kMK26	ERRM3 kMK25	ERRM3 kMK24	ERRM3 kMK23	ERRM3 kMK22	ERRM3 kMK21	ERRM3 kMK20	ERRM3 kMK19	ERRM3 kMK18	ERRM3 kMK17	ERRM3 kMK16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRM3 kMK15	ERRM3 kMK14	ERRM3 kMK13	ERRM3 kMK12	ERRM3 kMK11	ERRM3 kMK10	ERRM3 kMK9	ERRM3 kMK8	ERRM3 kMK7	ERRM3 kMK6	ERRM3 kMK5	ERRM3 kMK4	ERRM3 kMK3	ERRM3 kMK2	ERRM3 kMK1	ERRM3 kMK0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ERRM3kMKn	0h	RW	State of error interrupt k - n 0b: No error interrupt masked 1b: Error interrupt masked

(24) Error Interrupt CA55 Status Register k (ICU_ERINTA55CTLk) (k = 0 to 3)

[Secure group: Gr0]

These registers hold the source flags for error interrupts to the CA55 when they have occurred.

Even one factor having been detected keeps the error interrupt signal to the CA55 at the active level.

The registers do not hold the source flags for interrupt signals that have been masked by the settings of the ERINTA55MSKk registers.

The individual sources prior to their bundling for BUS_ERR_INT and RAM_ERR_INT are indicated in the BEISRk and BEISRk registers, respectively. Therefore, these registers do not hold the source flags for BUS_ERR_INT and RAM_ERR_INT. The values for BUS_ERR_INT and RAM_ERR_INT are readable.

These registers are reset by the ERROR_RESETn signal. They are not reset by the PRESETn_I signal.

The interrupt signals are allocated to the register bits in the same way as with **Table 4.6-7**.

Access Size : 32 bits

Address : <ICU_base> + 0338h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ERRkSTAT31	ERRkSTAT30	ERRkSTAT29	ERRkSTAT28	ERRkSTAT27	ERRkSTAT26	ERRkSTAT25	ERRkSTAT24	ERRkSTAT23	ERRkSTAT22	ERRkSTAT21	ERRkSTAT20	ERRkSTAT19	ERRkSTAT18	ERRkSTAT17	ERRkSTAT16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRkSTAT15	ERRkSTAT14	ERRkSTAT13	ERRkSTAT12	ERRkSTAT11	ERRkSTAT10	ERRkSTAT9	ERRkSTAT8	ERRkSTAT7	ERRkSTAT6	ERRkSTAT5	ERRkSTAT4	ERRkSTAT3	ERRkSTAT2	ERRkSTAT1	ERRkSTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ERRkSTATn	0h	R	State of error interrupt k - n 0b: No error interrupt detected 1b: Error interrupt detected Writing has no effect.

(25) Error Interrupt CA55 Status Clear Register k (ICU_ERINTA55CLRk) (k = 0 to 3)

[Secure group: Gr0]

These registers clear the state flags for factors of error interrupts to the CA55.

The interrupt signals are allocated to the register bits in the same way as with **Table 4.6-8**.

Remark The ERINTA55CTLx registers do not cover BUS_ERR_INT and RAM_ERR_INT so these registers do not clear them.

Access Size : 32 bits

Address : <ICU_base> + 0348h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ERRA5kCLR31	ERRA5kCLR30	ERRA5kCLR29	ERRA5kCLR28	ERRA5kCLR27	ERRA5kCLR26	ERRA5kCLR25	ERRA5kCLR24	ERRA5kCLR23	ERRA5kCLR22	ERRA5kCLR21	ERRA5kCLR20	ERRA5kCLR19	ERRA5kCLR18	ERRA5kCLR17	ERRA5kCLR16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRA5kCLR15	ERRA5kCLR14	ERRA5kCLR13	ERRA5kCLR12	ERRA5kCLR11	ERRA5kCLR10	ERRA5kCLR9	ERRA5kCLR8	ERRA5kCLR7	ERRA5kCLR6	ERRA5kCLR5	ERRA5kCLR4	ERRA5kCLR3	ERRA5kCLR2	ERRA5kCLR1	ERRA5kCLR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ERRA5kCLRn	0h	W	Clears the error interrupt detection factor allocated in the corresponding bit by writing 1b. If the value is 0b, it does not change (when read, 0b is always read).

(26) Error Interrupt CA55 Mask Control Register k (ICU_ERINTA55MSKk) (k = 0 to 3)

[Secure group: Gr0]

These registers control whether or not to retain factors of error interrupts to the CA55.

Register bits are allocated to the individual error interrupts. Writing 1 to a bit masks the corresponding error interrupt by preventing retention of the factor.

The interrupt signals are allocated to the register bits in the same way as with **Table 4.6-7**.

Remark 1. The ERINTA55CTLk registers do not cover BUS_ERR_INT and RAM_ERR_INT so these registers do not clear them.

Remark 2. This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read-modify-write operation).

Access Size : 32 bits

Address : <ICU_base> + 0358h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ERRA5 kMK31	ERRA5 kMK30	ERRA5 kMK29	ERRA5 kMK28	ERRA5 kMK27	ERRA5 kMK26	ERRA5 kMK25	ERRA5 kMK24	ERRA5 kMK23	ERRA5 kMK22	ERRA5 kMK21	ERRA5 kMK20	ERRA5 kMK19	ERRA5 kMK18	ERRA5 kMK17	ERRA5 kMK16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRA5 kMK15	ERRA5 kMK14	ERRA5 kMK13	ERRA5 kMK12	ERRA5 kMK11	ERRA5 kMK10	ERRA5 kMK9	ERRA5 kMK8	ERRA5 kMK7	ERRA5 kMK6	ERRA5 kMK5	ERRA5 kMK4	ERRA5 kMK3	ERRA5 kMK2	ERRA5 kMK1	ERRA5 kMK0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ERRA5kMKn	0h	RW	State of error interrupt k - n 0b: No error interrupt masked 1b: Error interrupt masked

(27) GPT Interrupt CA55 Status Register (ICU_GPTINTA55CTL)

[Secure group: Gr0]

This register holds the source flags for GPT interrupts (gpt_gtciada_n_0 to gpt_gtciada_n_7 and gpt_gtciadb_n_0 to gpt_gtciadb_n_7) when they have occurred.

Even one factor having been detected keeps the error interrupt signal to the CA55 at the active level.

Bit[7:0] are assigned to the interrupt signals (x = 7 to 0).

Access Size : 32 bits
Address : <ICU_base> + 0368h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	G1IADAk[7:0]								G1IADBk[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	G0IADAk[7:0]								G0IADBk[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	G1IADAk[7:0]	0h	R	State of interrupt gpt_gtciada_n_x of GPT1 0b: No error interrupt detected 1b: Error interrupt detected Writing has no effect.
23 to 16	G1IADBk[7:0]	0h	R	State of interrupt gpt_gtciadb_n_x of GPT1 0b: No error interrupt detected 1b: Error interrupt detected Writing has no effect.
15 to 8	G0IADAk[7:0]	0h	R	State of interrupt gpt_gtciada_n_x of GPT0 0b: No error interrupt detected 1b: Error interrupt detected Writing has no effect.
7 to 0	G0IADBk[7:0]	0h	R	State of interrupt gpt_gtciadb_n_x of GPT0 0b: No error interrupt detected 1b: Error interrupt detected Writing has no effect.

(28) GPT Interrupt CA55 Status Clear Register (ICU_GPTINTA55CLR)

[Secure group: Gr0]

This register clears the state flags for factors of GPT interrupts (gpt_gtciada_n_0 to gpt_gtciada_n_7 and gpt_gtciadb_n_0 to gpt_gtciadb_n_7).

Access Size : 32 bits
Address : <ICU_base> + 036Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	G1IADAkCLR[7:0]								G1IADBkCLR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	G0IADAkCLR[7:0]								G0IADBkCLR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	G1IADAkCLR [7:0]	0h	W	The read value is undefined Clears the error interrupt detection factor allocated in the corresponding bit by writing 1b. If the value is 0b, it does not change (when read, 0b is always read).
23 to 16	G1IADBkCLR [7:0]	0h	W	The read value is undefined Clears the error interrupt detection factor allocated in the corresponding bit by writing 1b. If the value is 0b, it does not change (when read, 0b is always read).
15 to 8	G0IADAkCLR [7:0]	0h	W	The read value is undefined Clears the error interrupt detection factor allocated in the corresponding bit by writing 1b. If the value is 0b, it does not change (when read, 0b is always read).
7 to 0	G0IADBkCLR [7:0]	0h	W	The read value is undefined Clears the error interrupt detection factor allocated in the corresponding bit by writing 1b. If the value is 0b, it does not change (when read, 0b is always read).

(29) SW Enhancing Pseudo Error Generation Register (ICU_SWPE)

[Secure group: Gr1]

This register generates software enhancing pseudo error interrupts. A software enhancing pseudo error interrupt is generated in the ICU by writing 1 to a bit corresponding to an event. A pulse lasting for one cycle of PCLK is generated.

Access Size : 32 bits
Address : <ICU_base> + 0370h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15	PE14	PE13	PE12	PE11	PE10	PE9	PE8	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	PEn	0h	W	The read value is undefined A software enhancing pseudo error interrupt is generated by writing 1.

(30) SW DMAC Request Generation Register k (ICU_SWDMACk) (k = 0 to 4)

[Secure group: Gr1]

These registers generate software DMA requests. A software DMA request for transfer by a DMAC channel is generated in the ICU by writing 1 to the bit corresponding to a channel. A pulse lasting for one cycle of PCLK is generated as the DMAC REQ signal.

Remark To use this function, specify the desired type of event in DMAC Factor k Selection Register y (DMkSELy).

Access Size : 32 bits
Address : <ICU_base> + 0400h + k x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DkC15	DkC14	DkC13	DkC12	DkC11	DkC10	DkC9	DkC8	DkC7	DkC6	DkC5	DkC4	DkC3	DkC2	DkC1	DkC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	DkCn	0h	W	The read value is undefined A software DMAC request for ch. n of the DMAC is generated by writing 1.

(31) DMAC Factor k Selection Register y (ICU_DMkSELy) (k = 0 to 4, y = 0 to 7)

[Secure group: Gr1]

These registers select source events for the DMAREQ signals to the DMACs. The events can be set for each DMAREQ signal to the DMACs (for DMAC0 to DMAC4). Writing the value under "DMAC No." in **Table 4.6-24** to a register selects the corresponding event. Note that all bits of the registers following a reset have the value 1, so no sources are selected at that time and a fixed value is input to the DMAC.

The DkRQ_SEL(y × 2 + 1) and DkRQ_SEL(y × 2) bits can be automatically set for no sources (set to the default value, 3FFh) in response to the DMAC end interrupt. This function is enabled or disabled with the DkSEL_CLRON(y × 2 + 1) and DkSEL_CLRON(y × 2) bits. See **(3) Automatic clearing DMAC factor selection registers**.

Values in **Table 4.6-24** are decimal numbers.

Remark 1. This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read-modify-write operation).

Remark 2. Duplication of settings between registers is prohibited, so do not set the same value. Writing the same value to the DkRQ_SEL(y × 2 + 1) and DkRQ_SEL(y × 2) bits is also prohibited.

Remark 3. When the DkRQ_SEL(y × 2 + 1) and DkRQ_SEL(y × 2) bits are not to be used, write the default value 3FFh to them.

Access Size : 32 bits
Address : <ICU_base> + 0420h + k x 0020h + y x 0004h
Initial Value : 03FF_03FFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DkSEL_CLRON(yx2+1)	-	-	-	-	-	DkRQ_SEL(yx2+1)[9:0]									
Initial Value	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W	RW	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DkSEL_CLRON(yx2)	-	-	-	-	-	DkRQ_SEL(yx2)[9:0]									
Initial Value	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W	RW	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	DkSEL_CLRON(yx2+1)	0h	RW	If the interrupt signal corresponding to DkRQ_SEL(y x 2 + 1) of DMACk is input, the function to enable or disable automatic clearing of the DxRQ_SEL(y x 2 + 1) register (all 1s) is set. 1b: Automatic clearing is enabled. 0b: Automatic clearing is disabled.
30 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 16	DkRQ_SEL(yx2+1)[9:0]	3FFh	RW	The source of the event signal input to DMAREQ((y x 2) + 1) of DMACk is selected. 0h to 1B4h: The signal of the corresponding event number is output. 1B5h to 3FFh: A fixed value is output.
15	DkSEL_CLRON(yx2)	0h	RW	If the interrupt signal corresponding to DkRQ_SEL(y x 2) of DMACx is input, the function to enable or disable automatic clearing of the DxRQ_SEL(y x 2) register (all 1s) is set. 1b: Automatic clearing is enabled. 0b: Automatic clearing is disabled.
14 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
9 to 0	DkRQ_SEL(yx2) [9:0]	3FFh	RW	The source of the event signal input to DMAREQ(y x 2) of DMACk is selected. 0h to 1B4h: The signal of the corresponding event number is output. 1B5h to 3FFh: A fixed value is output

Table 4.6-9 Relationship between DMAC REQch and DMkSELy (1/2)

No.	DMAC REQ Signal	DMAC END Interrupt	DMkSELy		
			k	y	Bit
0	DMAC1_REQ0	DMAC1_DMAEND_0	0	0	Lower
1	DMAC1_REQ1	DMAC1_DMAEND_1	0	0	Upper
2	DMAC1_REQ2	DMAC1_DMAEND_2	0	1	Lower
3	DMAC1_REQ3	DMAC1_DMAEND_3	0	1	Upper
4	DMAC1_REQ4	DMAC1_DMAEND_4	0	2	Lower
5	DMAC1_REQ5	DMAC1_DMAEND_5	0	2	Upper
6	DMAC1_REQ6	DMAC1_DMAEND_6	0	3	Lower
7	DMAC1_REQ7	DMAC1_DMAEND_7	0	3	Upper
8	DMAC1_REQ8	DMAC1_DMAEND_8	0	4	Lower
9	DMAC1_REQ9	DMAC1_DMAEND_9	0	4	Upper
10	DMAC1_REQ10	DMAC1_DMAEND_10	0	5	Lower
11	DMAC1_REQ11	DMAC1_DMAEND_11	0	5	Upper
12	DMAC1_REQ12	DMAC1_DMAEND_12	0	6	Lower
13	DMAC1_REQ13	DMAC1_DMAEND_13	0	6	Upper
14	DMAC1_REQ14	DMAC1_DMAEND_14	0	7	Lower
15	DMAC1_REQ15	DMAC1_DMAEND_15	0	7	Upper
16	DMAC2_REQ0	DMAC2_DMAEND_0	1	0	Lower
17	DMAC2_REQ1	DMAC2_DMAEND_1	1	0	Upper
18	DMAC2_REQ2	DMAC2_DMAEND_2	1	1	Lower
19	DMAC2_REQ3	DMAC2_DMAEND_3	1	1	Upper
20	DMAC2_REQ4	DMAC2_DMAEND_4	1	2	Lower
21	DMAC2_REQ5	DMAC2_DMAEND_5	1	2	Upper
22	DMAC2_REQ6	DMAC2_DMAEND_6	1	3	Lower
23	DMAC2_REQ7	DMAC2_DMAEND_7	1	3	Upper
24	DMAC2_REQ8	DMAC2_DMAEND_8	1	4	Lower
25	DMAC2_REQ9	DMAC2_DMAEND_9	1	4	Upper
26	DMAC2_REQ10	DMAC2_DMAEND_10	1	5	Lower
27	DMAC2_REQ11	DMAC2_DMAEND_11	1	5	Upper
28	DMAC2_REQ12	DMAC2_DMAEND_12	1	6	Lower
29	DMAC2_REQ13	DMAC2_DMAEND_13	1	6	Upper
30	DMAC2_REQ14	DMAC2_DMAEND_14	1	7	Lower
31	DMAC2_REQ15	DMAC2_DMAEND_15	1	7	Upper
32	DMAC3_REQ0	DMAC3_DMAEND_0	2	0	Lower
33	DMAC3_REQ1	DMAC3_DMAEND_1	2	0	Upper
34	DMAC3_REQ2	DMAC3_DMAEND_2	2	1	Lower
35	DMAC3_REQ3	DMAC3_DMAEND_3	2	1	Upper
36	DMAC3_REQ4	DMAC3_DMAEND_4	2	2	Lower
37	DMAC3_REQ5	DMAC3_DMAEND_5	2	2	Upper
38	DMAC3_REQ6	DMAC3_DMAEND_6	2	3	Lower
39	DMAC3_REQ7	DMAC3_DMAEND_7	2	3	Upper

Table 4.6-9 Relationship between DMAC REQch and DMkSELy (2/2)

No.	DMAC REQ Signal	DMAC END Interrupt	DMkSELy		
			k	y	Bit
40	DMAC3_REQ8	DMAC3_DMAEND_8	2	4	Lower
41	DMAC3_REQ9	DMAC3_DMAEND_9	2	4	Upper
42	DMAC3_REQ10	DMAC3_DMAEND_10	2	5	Lower
43	DMAC3_REQ11	DMAC3_DMAEND_11	2	5	Upper
44	DMAC3_REQ12	DMAC3_DMAEND_12	2	6	Lower
45	DMAC3_REQ13	DMAC3_DMAEND_13	2	6	Upper
46	DMAC3_REQ14	DMAC3_DMAEND_14	2	7	Lower
47	DMAC3_REQ15	DMAC3_DMAEND_15	2	7	Upper
48	DMAC4_REQ0	DMAC4_DMAEND_0	3	0	Lower
49	DMAC4_REQ1	DMAC4_DMAEND_1	3	0	Upper
50	DMAC4_REQ2	DMAC4_DMAEND_2	3	1	Lower
51	DMAC4_REQ3	DMAC4_DMAEND_3	3	1	Upper
52	DMAC4_REQ4	DMAC4_DMAEND_4	3	2	Lower
53	DMAC4_REQ5	DMAC4_DMAEND_5	3	2	Upper
54	DMAC4_REQ6	DMAC4_DMAEND_6	3	3	Lower
55	DMAC4_REQ7	DMAC4_DMAEND_7	3	3	Upper
56	DMAC4_REQ8	DMAC4_DMAEND_8	3	4	Lower
57	DMAC4_REQ9	DMAC4_DMAEND_9	3	4	Upper
58	DMAC4_REQ10	DMAC4_DMAEND_10	3	5	Lower
59	DMAC4_REQ11	DMAC4_DMAEND_11	3	5	Upper
60	DMAC4_REQ12	DMAC4_DMAEND_12	3	6	Lower
61	DMAC4_REQ13	DMAC4_DMAEND_13	3	6	Upper
62	DMAC4_REQ14	DMAC4_DMAEND_14	3	7	Lower
63	DMAC4_REQ15	DMAC4_DMAEND_15	3	7	Upper
64	DMAC0_REQ0	DMAC0_DMAEND_0	4	0	Lower
65	DMAC0_REQ1	DMAC0_DMAEND_1	4	0	Upper
66	DMAC0_REQ2	DMAC0_DMAEND_2	4	1	Lower
67	DMAC0_REQ3	DMAC0_DMAEND_3	4	1	Upper
68	DMAC0_REQ4	DMAC0_DMAEND_4	4	2	Lower
69	DMAC0_REQ5	DMAC0_DMAEND_5	4	2	Upper
70	DMAC0_REQ6	DMAC0_DMAEND_6	4	3	Lower
71	DMAC0_REQ7	DMAC0_DMAEND_7	4	3	Upper
72	DMAC0_REQ8	DMAC0_DMAEND_8	4	4	Lower
73	DMAC0_REQ9	DMAC0_DMAEND_9	4	4	Upper
74	DMAC0_REQ10	DMAC0_DMAEND_10	4	5	Lower
75	DMAC0_REQ11	DMAC0_DMAEND_11	4	5	Upper
76	DMAC0_REQ12	DMAC0_DMAEND_12	4	6	Lower
77	DMAC0_REQ13	DMAC0_DMAEND_13	4	6	Upper
78	DMAC0_REQ14	DMAC0_DMAEND_14	4	7	Lower
79	DMAC0_REQ15	DMAC0_DMAEND_15	4	7	Upper

(32) DMAC Request Repetition Control Register k (ICU_DMRCTRk) (k = 0 to 2)

[Secure group: Gr1]

These registers retain the DMAC channels for which the factors that have already been set in the bits of a different register when the DMAC Factor k Selection Register y (DMkSELy) is set.

Access Size : 32 bits

Address : <ICU_base> + 04C0h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMADUPk_31	DMADUPk_30	DMADUPk_29	DMADUPk_28	DMADUPk_27	DMADUPk_26	DMADUPk_25	DMADUPk_24	DMADUPk_23	DMADUPk_22	DMADUPk_21	DMADUPk_20	DMADUPk_19	DMADUPk_18	DMADUPk_17	DMADUPk_16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMADUPk_15	DMADUPk_14	DMADUPk_13	DMADUPk_12	DMADUPk_11	DMADUPk_10	DMADUPk_9	DMADUPk_8	DMADUPk_7	DMADUPk_6	DMADUPk_5	DMADUPk_4	DMADUPk_3	DMADUPk_2	DMADUPk_1	DMADUPk_0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMADUPk_n	0h	R	State of the DMAC ch. for duplication detection, corresponding to the table "Relationship between DMAC CH and DMADUPk_n". 0b: No duplicate setting detected 1b: Duplicate setting detected

Table 4.6-10 Relationship between DMAC CH and DMADUPk_n (1/2)

DMAC CH	DMADUPk_n	
DMAC1_CH0	n = 0	k = 0
DMAC1_CH1	n = 1	k = 0
DMAC1_CH2	n = 2	k = 0
DMAC1_CH3	n = 3	k = 0
DMAC1_CH4	n = 4	k = 0
DMAC1_CH5	n = 5	k = 0
DMAC1_CH6	n = 6	k = 0
DMAC1_CH7	n = 7	k = 0
DMAC1_CH8	n = 8	k = 0
DMAC1_CH9	n = 9	k = 0
DMAC1_CH10	n = 10	k = 0
DMAC1_CH11	n = 11	k = 0
DMAC1_CH12	n = 12	k = 0
DMAC1_CH13	n = 13	k = 0
DMAC1_CH14	n = 14	k = 0
DMAC1_CH15	n = 15	k = 0
DMAC2_CH0	n = 16	k = 0
DMAC2_CH1	n = 17	k = 0
DMAC2_CH2	n = 18	k = 0
DMAC2_CH3	n = 19	k = 0

DMAC CH	DMADUPk_n	
DMAC2_CH6	n = 22	k = 0
DMAC2_CH7	n = 23	k = 0
DMAC2_CH8	n = 24	k = 0
DMAC2_CH9	n = 25	k = 0
DMAC2_CH10	n = 26	k = 0
DMAC2_CH11	n = 27	k = 0
DMAC2_CH12	n = 28	k = 0
DMAC2_CH13	n = 29	k = 0
DMAC2_CH14	n = 30	k = 0
DMAC2_CH15	n = 31	k = 0
DMAC3_CH0	n = 0	k = 1
DMAC3_CH1	n = 1	k = 1
DMAC3_CH2	n = 2	k = 1
DMAC3_CH3	n = 3	k = 1
DMAC3_CH4	n = 4	k = 1
DMAC3_CH5	n = 5	k = 1
DMAC3_CH6	n = 6	k = 1
DMAC3_CH7	n = 7	k = 1
DMAC3_CH8	n = 8	k = 1
DMAC3_CH9	n = 9	k = 1

Table 4.6-10 Relationship between DMAC CH and DMADUPk_n (2/2)

DMAC CH	DMADUPk_n	
DMAC2_CH4	n = 20	k = 0
DMAC2_CH5	n = 21	k = 0
DMAC3_CH12	n = 12	k = 1
DMAC3_CH13	n = 13	k = 1
DMAC3_CH14	n = 14	k = 1
DMAC3_CH15	n = 15	k = 1
DMAC4_CH0	n = 16	k = 1
DMAC4_CH1	n = 17	k = 1
DMAC4_CH2	n = 18	k = 1
DMAC4_CH3	n = 19	k = 1
DMAC4_CH4	n = 20	k = 1
DMAC4_CH5	n = 21	k = 1
DMAC4_CH6	n = 22	k = 1
DMAC4_CH7	n = 23	k = 1
DMAC4_CH8	n = 24	k = 1
DMAC4_CH9	n = 25	k = 1
DMAC4_CH10	n = 26	k = 1
DMAC4_CH11	n = 27	k = 1
DMAC4_CH12	n = 28	k = 1
DMAC4_CH13	n = 29	k = 1

DMAC CH	DMADUPk_n	
DMAC3_CH10	n = 10	k = 1
DMAC3_CH11	n = 11	k = 1
DMAC4_CH14	n = 30	k = 1
DMAC4_CH15	n = 31	k = 1
DMAC0_CH0	n = 0	k = 2
DMAC0_CH1	n = 1	k = 2
DMAC0_CH2	n = 2	k = 2
DMAC0_CH3	n = 3	k = 2
DMAC0_CH4	n = 4	k = 2
DMAC0_CH5	n = 5	k = 2
DMAC0_CH6	n = 6	k = 2
DMAC0_CH7	n = 7	k = 2
DMAC0_CH8	n = 8	k = 2
DMAC0_CH9	n = 9	k = 2
DMAC0_CH10	n = 10	k = 2
DMAC0_CH11	n = 11	k = 2
DMAC0_CH12	n = 12	k = 2
DMAC0_CH13	n = 13	k = 2
DMAC0_CH14	n = 14	k = 2
DMAC0_CH15	n = 15	k = 2

(33) DMAC Request Repetition Clear Register k (ICU_DMRCLRk) (k = 0 to 2)

[Secure group: Gr1]

These registers clear the factors in the DMAC request repetition control registers k (DMRCTRk).

Access Size : 32 bits

Address : <ICU_base> + 04D0h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMADUPCLRk_31	DMADUPCLRk_30	DMADUPCLRk_29	DMADUPCLRk_28	DMADUPCLRk_27	DMADUPCLRk_26	DMADUPCLRk_25	DMADUPCLRk_24	DMADUPCLRk_23	DMADUPCLRk_22	DMADUPCLRk_21	DMADUPCLRk_20	DMADUPCLRk_19	DMADUPCLRk_18	DMADUPCLRk_17	DMADUPCLRk_16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMADUPCLRk_15	DMADUPCLRk_14	DMADUPCLRk_13	DMADUPCLRk_12	DMADUPCLRk_11	DMADUPCLRk_10	DMADUPCLRk_9	DMADUPCLRk_8	DMADUPCLRk_7	DMADUPCLRk_6	DMADUPCLRk_5	DMADUPCLRk_4	DMADUPCLRk_3	DMADUPCLRk_2	DMADUPCLRk_1	DMADUPCLRk_0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMADUPCLRk_n	0h	W	The read value is undefined Clears the factor allocated in the corresponding bit by writing 1b. If the value is 0b, it does not change.

(34) DMAC ACK Selection Register k (ICU_DMACKSELk) (k = 0 to 22)

[Secure group: Gr1]

These registers select source events for the DMA ACK signals from the DMACs. The events can be set for each DMA ACK signal from the DMACs. Each signal can be connected to any DMAC channel by setting DMAC ACK No.0 to 79 in **Table 4.6-11** for the DACK_SEL bit corresponding to "ACK No." in **Table 4.6-28**.

Note that all bits of the registers following a reset have the value 1, so no sources are selected at that time and a fixed level is input as the ACK signal of each unit.

Values in **Table 4.6-28** are decimal numbers.

Remark 1. In this LSI chip, operation of the DMAC ACK signals is assumed to be based on level detection. Set level detection for the ACK settings on the DMAC side. Use the signals in ways that suit the factors selected in the DMACk factor selection registers y (DMkSELy).

Remark 2. This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read-modify-write operation).

Access Size : 32 bits

Address : <ICU_base> + 0500h + k x 0004h

Initial Value : 7F7F_7F7Fh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	DACK_SEL(kx4+3)[6:0]							-	DACK_SEL(kx4+2)[6:0]						
Initial Value	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
R/W	R	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	DACK_SEL(kx4+1)[6:0]							-	DACK_SEL(kx4)[6:0]						
Initial Value	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
R/W	R	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 24	DACK_SEL(kx4+3)[6:0]	7Fh	RW	The source for units of the DMAC ACK signal is selected. 0h to 4Fh: The signal of the corresponding event number is output. 50h to 7Fh: A fixed value is output.
23	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22 to 16	DACK_SEL(kx4+2)[6:0]	7Fh	RW	The source for units of the DMAC ACK signal is selected. 0h to 4Fh: The signal of the corresponding event number is output. 50h to 7Fh: A fixed value is output.
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 8	DACK_SEL(kx4+1)[6:0]	7Fh	RW	The source for units of the DMAC ACK signal is selected. 0h to 4Fh: The signal of the corresponding event number is output. 50h to 7Fh: A fixed value is output.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 0	DACK_SEL(kx4)[6:0]	7Fh	RW	The source for units of the DMAC ACK signal is selected. 0h to 4Fh: The signal of the corresponding event number is output. 50h to 7Fh: A fixed value is output.

Table 4.6-11 DMAC ACK No. and Signal Correlation

ICU_TOP IF (Input) Signal Name	DMACACK No.
DMAC1_DMAACK_0	0
DMAC1_DMAACK_1	1
DMAC1_DMAACK_2	2
DMAC1_DMAACK_3	3
DMAC1_DMAACK_4	4
DMAC1_DMAACK_5	5
DMAC1_DMAACK_6	6
DMAC1_DMAACK_7	7
DMAC1_DMAACK_8	8
DMAC1_DMAACK_9	9
DMAC1_DMAACK_10	10
DMAC1_DMAACK_11	11
DMAC1_DMAACK_12	12
DMAC1_DMAACK_13	13
DMAC1_DMAACK_14	14
DMAC1_DMAACK_15	15
DMAC2_DMAACK_0	16
DMAC2_DMAACK_1	17
DMAC2_DMAACK_2	18
DMAC2_DMAACK_3	19
DMAC2_DMAACK_4	20
DMAC2_DMAACK_5	21
DMAC2_DMAACK_6	22
DMAC2_DMAACK_7	23
DMAC2_DMAACK_8	24
DMAC2_DMAACK_9	25
DMAC2_DMAACK_10	26
DMAC2_DMAACK_11	27
DMAC2_DMAACK_12	28
DMAC2_DMAACK_13	29
DMAC2_DMAACK_14	30
DMAC2_DMAACK_15	31
DMAC3_DMAACK_0	32
DMAC3_DMAACK_1	33
DMAC3_DMAACK_2	34
DMAC3_DMAACK_3	35
DMAC3_DMAACK_4	36
DMAC3_DMAACK_5	37
DMAC3_DMAACK_6	38
DMAC3_DMAACK_7	39

ICU_TOP IF (Input) Signal Name	DMACACK No.
DMAC3_DMAACK_8	40
DMAC3_DMAACK_9	41
DMAC3_DMAACK_10	42
DMAC3_DMAACK_11	43
DMAC3_DMAACK_12	44
DMAC3_DMAACK_13	45
DMAC3_DMAACK_14	46
DMAC3_DMAACK_15	47
DMAC4_DMAACK_0	48
DMAC4_DMAACK_1	49
DMAC4_DMAACK_2	50
DMAC4_DMAACK_3	51
DMAC4_DMAACK_4	52
DMAC4_DMAACK_5	53
DMAC4_DMAACK_6	54
DMAC4_DMAACK_7	55
DMAC4_DMAACK_8	56
DMAC4_DMAACK_9	57
DMAC4_DMAACK_10	58
DMAC4_DMAACK_11	59
DMAC4_DMAACK_12	60
DMAC4_DMAACK_13	61
DMAC4_DMAACK_14	62
DMAC4_DMAACK_15	63
DMAC0_DMAACK_0	64
DMAC0_DMAACK_1	65
DMAC0_DMAACK_2	66
DMAC0_DMAACK_3	67
DMAC0_DMAACK_4	68
DMAC0_DMAACK_5	69
DMAC0_DMAACK_6	70
DMAC0_DMAACK_7	71
DMAC0_DMAACK_8	72
DMAC0_DMAACK_9	73
DMAC0_DMAACK_10	74
DMAC0_DMAACK_11	75
DMAC0_DMAACK_12	76
DMAC0_DMAACK_13	77
DMAC0_DMAACK_14	78
DMAC0_DMAACK_15	79

(35) DMAC TEND Selection Register k (ICU_DMTENDSELk) (k = 0, 1)

[Secure group: Gr1]

These registers select source events for the DMA TEND signals from the DMACs. The events can be set for each DMA TEND signal from the DMACs. Each signal can be connected to any DMAC channel by setting the DMAC TEND No. 0 to 79 in **Table4.6-12** to the DTEND_SEL bit corresponding to the "TEND No." in **Table 4.6-28**.

Note that all bits of the registers following a reset have the value 1, so no sources are selected at that time and a fixed level is input as the TEND signal of each unit.

Values in **Table 4.6-28** are decimal numbers.

Remark 1. The DMATCO signal of the DMAC is extended by 16 cycles of PCLK and output as the DMA TEND signal. Use the signals in ways that suit the factors selected in the DMACk factor selection registers y (DMkSELy).

Remark 2. Five DMA TEND signals are specifiable. No register bits are allocated to the DTEND_SEL7, DTEND_SEL6, and DTEND_SEL5 signals of the DMTENDSEL1 register.

Remark 3. This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read-modify-write operation).

Access Size : 32 bits

Address : <ICU_base> + 055Ch + k x 0004h

Initial Value : 7F7F_7F7Fh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	DTEND_SEL(kx4+3)[6:0]							-	DTEND_SEL(kx4+2)[6:0]						
Initial Value	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
R/W	R	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	DTEND_SEL(kx4+1)[6:0]							-	DTEND_SEL(kx4)[6:0]						
Initial Value	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
R/W	R	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 24	DTEND_SEL(kx4+3)[6:0]	7Fh	RW	The source for units of the DMAC TEND signal is selected. 0h to 4Fh: The signal of the corresponding event number is output. 50h to 7Fh: A fixed value is output.
23	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22 to 16	DTEND_SEL(kx4+2)[6:0]	7Fh	RW	The source for units of the DMAC TEND signal is selected. 0h to 4Fh: The signal of the corresponding event number is output. 50h to 7Fh: A fixed value is output.
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 8	DTEND_SEL(kx4+1)[6:0]	7Fh	RW	The source for units of the DMAC TEND signal is selected. 0h to 4Fh: The signal of the corresponding event number is output. 50h to 7Fh: A fixed value is output.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 0	DTEND_SEL(kx4)[6:0]	7Fh	RW	The source for units of the DMAC TEND signal is selected. 0h to 4Fh: The signal of the corresponding event number is output. 50h to 7Fh: A fixed value is output.

Table4.6-12 DMAC TEND No. and Signal Correlation

ICU_TOP IF (Input) Signal Name	DMAC TEND No.
DMAC1_DMATEND_0	0
DMAC1_DMATEND_1	1
DMAC1_DMATEND_2	2
DMAC1_DMATEND_3	3
DMAC1_DMATEND_4	4
DMAC1_DMATEND_5	5
DMAC1_DMATEND_6	6
DMAC1_DMATEND_7	7
DMAC1_DMATEND_8	8
DMAC1_DMATEND_9	9
DMAC1_DMATEND_10	10
DMAC1_DMATEND_11	11
DMAC1_DMATEND_12	12
DMAC1_DMATEND_13	13
DMAC1_DMATEND_14	14
DMAC1_DMATEND_15	15
DMAC2_DMATEND_0	16
DMAC2_DMATEND_1	17
DMAC2_DMATEND_2	18
DMAC2_DMATEND_3	19
DMAC2_DMATEND_4	20
DMAC2_DMATEND_5	21
DMAC2_DMATEND_6	22
DMAC2_DMATEND_7	23
DMAC2_DMATEND_8	24
DMAC2_DMATEND_9	25
DMAC2_DMATEND_10	26
DMAC2_DMATEND_11	27
DMAC2_DMATEND_12	28
DMAC2_DMATEND_13	29
DMAC2_DMATEND_14	30
DMAC2_DMATEND_15	31
DMAC3_DMATEND_0	32
DMAC3_DMATEND_1	33
DMAC3_DMATEND_2	34
DMAC3_DMATEND_3	35
DMAC3_DMATEND_4	36
DMAC3_DMATEND_5	37
DMAC3_DMATEND_6	38
DMAC3_DMATEND_7	39

ICU_TOP IF (Input) Signal Name	DMAC TEND No.
DMAC3_DMATEND_8	40
DMAC3_DMATEND_9	41
DMAC3_DMATEND_10	42
DMAC3_DMATEND_11	43
DMAC3_DMATEND_12	44
DMAC3_DMATEND_13	45
DMAC3_DMATEND_14	46
DMAC3_DMATEND_15	47
DMAC4_DMATEND_0	48
DMAC4_DMATEND_1	49
DMAC4_DMATEND_2	50
DMAC4_DMATEND_3	51
DMAC4_DMATEND_4	52
DMAC4_DMATEND_5	53
DMAC4_DMATEND_6	54
DMAC4_DMATEND_7	55
DMAC4_DMATEND_8	56
DMAC4_DMATEND_9	57
DMAC4_DMATEND_10	58
DMAC4_DMATEND_11	59
DMAC4_DMATEND_12	60
DMAC4_DMATEND_13	61
DMAC4_DMATEND_14	62
DMAC4_DMATEND_15	63
DMAC0_DMATEND_0	64
DMAC0_DMATEND_1	65
DMAC0_DMATEND_2	66
DMAC0_DMATEND_3	67
DMAC0_DMATEND_4	68
DMAC0_DMATEND_5	69
DMAC0_DMATEND_6	70
DMAC0_DMATEND_7	71
DMAC0_DMATEND_8	72
DMAC0_DMATEND_9	73
DMAC0_DMATEND_10	74
DMAC0_DMATEND_11	75
DMAC0_DMATEND_12	76
DMAC0_DMATEND_13	77
DMAC0_DMATEND_14	78
DMAC0_DMATEND_15	79

(36) DMAC END Detection Method Selection Register k (ICU_DMESRk) (k = 0 to 2)

[Secure group: Gr1]

These registers control the method of detecting DMAC END signals.

Remark This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read–modify–write operation).

Access Size : 32 bits

Address : <ICU_base> + 0570h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMESELk_31	DMESELk_30	DMESELk_29	DMESELk_28	DMESELk_27	DMESELk_26	DMESELk_25	DMESELk_24	DMESELk_23	DMESELk_22	DMESELk_21	DMESELk_20	DMESELk_19	DMESELk_18	DMESELk_17	DMESELk_16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMESELk_15	DMESELk_14	DMESELk_13	DMESELk_12	DMESELk_11	DMESELk_10	DMESELk_9	DMESELk_8	DMESELk_7	DMESELk_6	DMESELk_5	DMESELk_4	DMESELk_3	DMESELk_2	DMESELk_1	DMESELk_0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMESELk_n	0h	RW	The interrupt detection method for DMAC END is selected. 0b: Level detection 1b: Pulse detection

Table 4.6-13 Relationship between DMAC END Signals and DMESRk (1/2)

DMAC END Signal	DMESELk_n		DMAC END Signal	DMESELk_n	
DMAC1_DMAEND_0	n = 0	k = 0	DMAC2_DMAEND_3	n = 19	k = 0
DMAC1_DMAEND_1	n = 1	k = 0	DMAC2_DMAEND_4	n = 20	k = 0
DMAC1_DMAEND_2	n = 2	k = 0	DMAC2_DMAEND_5	n = 21	k = 0
DMAC1_DMAEND_3	n = 3	k = 0	DMAC2_DMAEND_6	n = 22	k = 0
DMAC1_DMAEND_4	n = 4	k = 0	DMAC2_DMAEND_7	n = 23	k = 0
DMAC1_DMAEND_5	n = 5	k = 0	DMAC2_DMAEND_8	n = 24	k = 0
DMAC1_DMAEND_6	n = 6	k = 0	DMAC2_DMAEND_9	n = 25	k = 0
DMAC1_DMAEND_7	n = 7	k = 0	DMAC2_DMAEND_10	n = 26	k = 0
DMAC1_DMAEND_8	n = 8	k = 0	DMAC2_DMAEND_11	n = 27	k = 0
DMAC1_DMAEND_9	n = 9	k = 0	DMAC2_DMAEND_12	n = 28	k = 0
DMAC1_DMAEND_10	n = 10	k = 0	DMAC2_DMAEND_13	n = 29	k = 0
DMAC1_DMAEND_11	n = 11	k = 0	DMAC2_DMAEND_14	n = 30	k = 0
DMAC1_DMAEND_12	n = 12	k = 0	DMAC2_DMAEND_15	n = 31	k = 0
DMAC1_DMAEND_13	n = 13	k = 0	DMAC3_DMAEND_0	n = 0	k = 1
DMAC1_DMAEND_14	n = 14	k = 0	DMAC3_DMAEND_1	n = 1	k = 1
DMAC1_DMAEND_15	n = 15	k = 0	DMAC3_DMAEND_2	n = 2	k = 1
DMAC2_DMAEND_0	n = 16	k = 0	DMAC3_DMAEND_3	n = 3	k = 1
DMAC2_DMAEND_1	n = 17	k = 0	DMAC3_DMAEND_4	n = 4	k = 1
DMAC2_DMAEND_2	n = 18	k = 0	DMAC3_DMAEND_5	n = 5	k = 1

Table 4.6-13 Relationship between DMAC END Signals and DMESRk (2/2)

DMAC END Signal	DMESELk_n		DMAC END Signal	DMESELk_n	
DMAC3_DMAEND_6	n = 6	k = 1	DMAC4_DMAEND_11	n = 27	k = 1
DMAC3_DMAEND_7	n = 7	k = 1	DMAC4_DMAEND_12	n = 28	k = 1
DMAC3_DMAEND_8	n = 8	k = 1	DMAC4_DMAEND_13	n = 29	k = 1
DMAC3_DMAEND_9	n = 9	k = 1	DMAC4_DMAEND_14	n = 30	k = 1
DMAC3_DMAEND_10	n = 10	k = 1	DMAC4_DMAEND_15	n = 31	k = 1
DMAC3_DMAEND_11	n = 11	k = 1	DMAC0_DMAEND_0	n = 0	k = 2
DMAC3_DMAEND_12	n = 12	k = 1	DMAC0_DMAEND_1	n = 1	k = 2
DMAC3_DMAEND_13	n = 13	k = 1	DMAC0_DMAEND_2	n = 2	k = 2
DMAC3_DMAEND_14	n = 14	k = 1	DMAC0_DMAEND_3	n = 3	k = 2
DMAC3_DMAEND_15	n = 15	k = 1	DMAC0_DMAEND_4	n = 4	k = 2
DMAC4_DMAEND_0	n = 16	k = 1	DMAC0_DMAEND_5	n = 5	k = 2
DMAC4_DMAEND_1	n = 17	k = 1	DMAC0_DMAEND_6	n = 6	k = 2
DMAC4_DMAEND_2	n = 18	k = 1	DMAC0_DMAEND_7	n = 7	k = 2
DMAC4_DMAEND_3	n = 19	k = 1	DMAC0_DMAEND_8	n = 8	k = 2
DMAC4_DMAEND_4	n = 20	k = 1	DMAC0_DMAEND_9	n = 9	k = 2
DMAC4_DMAEND_5	n = 21	k = 1	DMAC0_DMAEND_10	n = 10	k = 2
DMAC4_DMAEND_6	n = 22	k = 1	DMAC0_DMAEND_11	n = 11	k = 2
DMAC4_DMAEND_7	n = 23	k = 1	DMAC0_DMAEND_12	n = 12	k = 2
DMAC4_DMAEND_8	n = 24	k = 1	DMAC0_DMAEND_13	n = 13	k = 2
DMAC4_DMAEND_9	n = 25	k = 1	DMAC0_DMAEND_14	n = 14	k = 2
DMAC4_DMAEND_10	n = 26	k = 1	DMAC0_DMAEND_15	n = 15	k = 2

(37) DMA ERR Interrupt Detection Method Selection Register (ICU_DERSR)

[Secure group: Gr1]

This register controls the method of detecting DMAC ERR signals.

The DMAERR signals of the DMACs are used as interrupts. Make the settings in ways that suit the settings (level or pulse) on the DMAC side.

Remark This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read–modify–write operation).

Access Size : 32 bits
Address : <ICU_base> + 057Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	DERSEL_4	DERSEL_3	DERSEL_2	DERSEL_1	DERSEL_0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	DERSEL_4	0h	RW	The interrupt detection method for DMAC ERR is selected. 0b: Level detection 1b: Pulse detection
3	DERSEL_3	0h	RW	The interrupt detection method for DMAC ERR is selected. 0b: Level detection 1b: Pulse detection
2	DERSEL_2	0h	RW	The interrupt detection method for DMAC ERR is selected. 0b: Level detection 1b: Pulse detection
1	DERSEL_1	0h	RW	The interrupt detection method for DMAC ERR is selected. 0b: Level detection 1b: Pulse detection
0	DERSEL_0	0h	RW	The interrupt detection method for DMAC ERR is selected. 0b: Level detection 1b: Pulse detection

Table 4.6-14 Relationship between DMAC ERR Signals and DMESRk

DMAC ERR Signal	DERSEL_n
DMAC1_DMAERR	n = 0
DMAC2_DMAERR	n = 1
DMAC3_DMAERR	n = 2

DMAC ERR Signal	DERSEL_n
DMAC4_DMAERR	n = 3
DMAC0_DMAERR	n = 4

(38) SW Event Output Generation Register (ICU_SWEVT)

[Secure group: Gr1]

This register generates software event output. Software event output is generated in the ICU by writing 1 and then 0 successively to a bit corresponding to the event.

Be sure to write 1b and then 0b successively to a bit.

Remark This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read–modify–write operation).

Access Size : 32 bits
Address : <ICU_base> + 0600h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	SWE6	SWE5	SWE4	SWE3	SWE2	SWE1	SWE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6	SWE6	0h	RW	The software event output is generated by writing 1b. When read, 0b is always read.
5	SWE5	0h	RW	The software event output is generated by writing 1b. When read, 0b is always read.
4	SWE4	0h	RW	The software event output is generated by writing 1b. When read, 0b is always read.
3	SWE3	0h	RW	The software event output is generated by writing 1b. When read, 0b is always read.
2	SWE2	0h	RW	The software event output is generated by writing 1b. When read, 0b is always read.
1	SWE1	0h	RW	The software event output is generated by writing 1b. When read, 0b is always read.
0	SWE0	0h	RW	The software event output is generated by writing 1b. When read, 0b is always read.

(39) Event Output Factor Selection Register k (ICU_EVTSELk) (k = 0 to 14)

[Secure group: Gr1]

These registers select source events for event signal outputs from the ELC. The events can be set for each destination of output from the ELC. Writing the value listed in the ELC column of **Table 4.6-24** to the register corresponding to the value of "Event Output No." in **Table 4.6-25** will select the desired event for the specified output destination.

Note that all bits of the registers following a reset have the value 1b, so no sources are selected at that time and a fixed value, which depends on the destination of output from the ELC, is output.

Values in **Table 4.6-25** are decimal numbers.

Remark 1. Using a DMAC REQ signal generated by an ICU_SW_DMAC register k (SWDMACk) as an event output is not allowed.

Remark 2. This register can be set in a one-word unit, so making the settings for multiple interrupts at the same time is possible. Take care with the settings so that only the target bits are changed (for example, use a read-modify-write operation).

Access Size : 32 bits

Address : <ICU_base> + 0604h + k x 0004h

Initial Value : 3FFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	EC_SEL(kx3+2)[9:0]										EC_SEL(kx3+1)[9:6]			
Initial Value	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EC_SEL(kx3+1)[5:0]						EC_SEL(kx3)[9:0]									
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29 to 20	EC_SEL(kx3+2)[9:0]	3FFh	RW	The source of the event signal output to EventOutNo(k x 3 + 2) is selected. 0h to 1C7h: The signal of the corresponding event number is output. 1C8h to 3FFh: A fixed value is output
19 to 10	EC_SEL(kx3+1)[9:0]	3FFh	RW	The source of the event signal output to EventOutNo(k x 3 + 1) is selected. 0h to 1C7h: The signal of the corresponding event number is output. 1C8h to 3FFh: A fixed value is output.
9 to 0	EC_SEL(kx3)[9:0]	3FFh	RW	The source of the event signal output to EventOutNo(k x 3) is selected. 0h to 1C7h: The signal of the corresponding event number is output. 1C8h to 3FFh: A fixed value is output

(40) Off Domain Access Factor Maintenance Register k (ICU_OFFDMSTATk) (k = 0 to 2)

[Secure group: Gr1]

These registers retain the factors on generation of DMAC requests or event outputs to an off-domain area. They retain the factors when an event output to another area is generated while the area is disabled. Clear the retained factors with the off domain access factor clear registers k (OFFDMCLRk) as required.

Access Size : 32 bits

Address : <ICU_base> + 0780h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ODkSTAT31	ODkSTAT30	ODkSTAT29	ODkSTAT28	ODkSTAT27	ODkSTAT26	ODkSTAT25	ODkSTAT24	ODkSTAT23	ODkSTAT22	ODkSTAT21	ODkSTAT20	ODkSTAT19	ODkSTAT18	ODkSTAT17	ODkSTAT16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ODkSTAT15	ODkSTAT14	ODkSTAT13	ODkSTAT12	ODkSTAT11	ODkSTAT10	ODkSTAT9	ODkSTAT8	ODkSTAT7	ODkSTAT6	ODkSTAT5	ODkSTAT4	ODkSTAT3	ODkSTAT2	ODkSTAT1	ODkSTAT0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ODkSTATn	0h	R	Indicates the detection of access to the off domain area. 0b: No access to the off domain detected 1b: Access to the off domain detected Writing has no effect.

Table 4.6-15 Relationship between Off-Domain Access Signal and ODkSTATn (1/2)

Off Domain Access Signal	ODkSTATn	
DMAC1_CH0	n = 0	k = 0
DMAC1_CH1	n = 1	k = 0
DMAC1_CH2	n = 2	k = 0
DMAC1_CH3	n = 3	k = 0
DMAC1_CH4	n = 4	k = 0
DMAC1_CH5	n = 5	k = 0
DMAC1_CH6	n = 6	k = 0
DMAC1_CH7	n = 7	k = 0
DMAC1_CH8	n = 8	k = 0
DMAC1_CH9	n = 9	k = 0
DMAC1_CH10	n = 10	k = 0
DMAC1_CH11	n = 11	k = 0
DMAC1_CH12	n = 12	k = 0
DMAC1_CH13	n = 13	k = 0
DMAC1_CH14	n = 14	k = 0
DMAC1_CH15	n = 15	k = 0
DMAC2_CH0	n = 16	k = 0
DMAC2_CH1	n = 17	k = 0
DMAC2_CH2	n = 18	k = 0
DMAC2_CH3	n = 19	k = 0

Off Domain Access Signal	ODkSTATn	
DMAC2_CH5	n = 21	k = 0
DMAC2_CH6	n = 22	k = 0
DMAC2_CH7	n = 23	k = 0
DMAC2_CH8	n = 24	k = 0
DMAC2_CH9	n = 25	k = 0
DMAC2_CH10	n = 26	k = 0
DMAC2_CH11	n = 27	k = 0
DMAC2_CH12	n = 28	k = 0
DMAC2_CH13	n = 29	k = 0
DMAC2_CH14	n = 30	k = 0
DMAC2_CH15	n = 31	k = 0
DMAC3_CH0	n = 0	k = 1
DMAC3_CH1	n = 1	k = 1
DMAC3_CH2	n = 2	k = 1
DMAC3_CH3	n = 3	k = 1
DMAC3_CH4	n = 4	k = 1
DMAC3_CH5	n = 5	k = 1
DMAC3_CH6	n = 6	k = 1
DMAC3_CH7	n = 7	k = 1
DMAC3_CH8	n = 8	k = 1

Table 4.6-15 Relationship between Off-Domain Access Signal and ODkSTATn (2/2)

Off Domain Access Signal	ODkSTATn	
DMAC2_CH4	n = 20	k = 0
DMAC3_CH10	n = 10	k = 1
DMAC3_CH11	n = 11	k = 1
DMAC3_CH12	n = 12	k = 1
DMAC3_CH13	n = 13	k = 1
DMAC3_CH14	n = 14	k = 1
DMAC3_CH15	n = 15	k = 1
DMAC4_CH0	n = 16	k = 1
DMAC4_CH1	n = 17	k = 1
DMAC4_CH2	n = 18	k = 1
DMAC4_CH3	n = 19	k = 1
DMAC4_CH4	n = 20	k = 1
DMAC4_CH5	n = 21	k = 1
DMAC4_CH6	n = 22	k = 1
DMAC4_CH7	n = 23	k = 1
DMAC4_CH8	n = 24	k = 1
DMAC4_CH9	n = 25	k = 1
DMAC4_CH10	n = 26	k = 1
DMAC4_CH11	n = 27	k = 1
DMAC4_CH12	n = 28	k = 1
DMAC4_CH13	n = 29	k = 1
DMAC4_CH14	n = 30	k = 1
DMAC4_CH15	n = 31	k = 1
GPT_U0_elc_gptact_0	n = 0	k = 2
GPT_U0_elc_gptact_1	n = 1	k = 2
GPT_U0_elc_gptact_2	n = 2	k = 2
GPT_U0_elc_gptact_3	n = 3	k = 2
GPT_U0_elc_gptact_4	n = 4	k = 2

Off Domain Access Signal	ODkSTATn	
DMAC3_CH9	n = 9	k = 1
GPT_U0_elc_gptact_5	n = 5	k = 2
GPT_U0_elc_gptact_6	n = 6	k = 2
GPT_U0_elc_gptact_7	n = 7	k = 2
GPT_U1_elc_gptact_0	n = 8	k = 2
GPT_U1_elc_gptact_1	n = 9	k = 2
GPT_U1_elc_gptact_2	n = 10	k = 2
GPT_U1_elc_gptact_3	n = 11	k = 2
GPT_U1_elc_gptact_4	n = 12	k = 2
GPT_U1_elc_gptact_5	n = 13	k = 2
GPT_U1_elc_gptact_6	n = 14	k = 2
GPT_U1_elc_gptact_7	n = 15	k = 2
TSU1_TSU_trg	n = 16	k = 2
—	n = 17	k = 2
—	n = 18	k = 2
—	n = 19	k = 2
—	n = 20	k = 2
—	n = 21	k = 2
—	n = 22	k = 2
—	n = 23	k = 2
—	n = 24	k = 2
Reserved	n = 25	k = 2
Reserved	n = 26	k = 2
Reserved	n = 27	k = 2
Reserved	n = 28	k = 2
Reserved	n = 29	k = 2
Reserved	n = 30	k = 2
Reserved	n = 31	k = 2

(41) Off Domain Access Factor Clear Register k (ICU_OFFDMCLRk) (k = 0 to 2)

[Secure group: Gr1]

These registers clear the factors of off-domain access. The registers are used to clear the factors retained in **(40) Off Domain Access Factor Maintenance Register k (ICU_OFFDMSTATk)**.

The relationship between the off-domain access signal and the bit to be cleared is the same as that shown in **Table 4.6-15** in **(40) Off Domain Access Factor Maintenance Register k (ICU_OFFDMSTATk)**.

Access Size : 32 bits

Address : <ICU_base> + 0798h + k x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ODCLR31	ODCLR30	ODCLR29	ODCLR28	ODCLR27	ODCLR26	ODCLR25	ODCLR24	ODCLR23	ODCLR22	ODCLR21	ODCLR20	ODCLR19	ODCLR18	ODCLR17	ODCLR16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ODCLR15	ODCLR14	ODCLR13	ODCLR12	ODCLR11	ODCLR10	ODCLR9	ODCLR8	ODCLR7	ODCLR6	ODCLR5	ODCLR4	ODCLR3	ODCLR2	ODCLR1	ODCLR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ODCLRn	0h	W	Clears the access generation detection factor for the off domain area allocated in the corresponding bit by writing 1b. If the value is 0b, it does not change (when read, 0b is always read).

4.6.1.3 Detailed Specifications

4.6.1.3.1 Control over input events

(1) Input event normalization/synchronization and event output

The table below shows the methods of detecting interrupts.

Table 4.6-16 Interrupt Detection Methods

Type	Detection Method	Selecting Method
NMI	Positive, negative	Register setting
IRQ	Positive, negative, positive/negative, low level	Register setting
TINT	Positive, negative, high level, low level	Register setting

Outputting interrupts

NMI, IRQ, and TINT signals are output at the high level regardless of their input type. The factors are retained in the ICU and the signals continue to be output while the factors are being held. Clearing all factors stops their output.

Similarly, the bus error, RAM error, and error interrupt signals, for which the ICU has bundling functions and factor-retention functions, continue to be output while their factors are being held.

Table 4.6-17 Output by Input Type: Part 1

Input Type	Output
NMI	High level while the factors are being held
IRQ	High level while the factors are being held
TINT	High level while the factors are being held
Error interrupt	High level while the factors are being held

For interrupts other than those in the table above, the input type determines the output type.

Outputting CPU interrupt signals

Table 4.6-18 Output by Input Type: Part 2

Input Type	Output
Low pulse	High pulse
High pulse	High pulse
Low level	High level
High level	High level

Outputting DMAC requests

In the case of level input, high-level signals are output.

In the case of pulse input, high-level pulses are output.

Outputting events

In the case of level input, high-level signals are output.

In the case of pulse input, high-level pulses are output.

Register access security control

Each register is classified into two groups (Gr0, Gr1).

Security control is possible for each group.

(2) Exclusive output control of interrupts and event output requests

Exclusive output control applies to events with ✓ under the “ELC” column and an entry under the interrupt column (“CA55” or “CM33”) in **Table 4.6-24**. For input events specified in an event output factor selection register k (EVTSELk), an event is output but an interrupt is not. On the other hand, for input events not specified in an event output factor selection register k (EVTSELk), an interrupt is output.

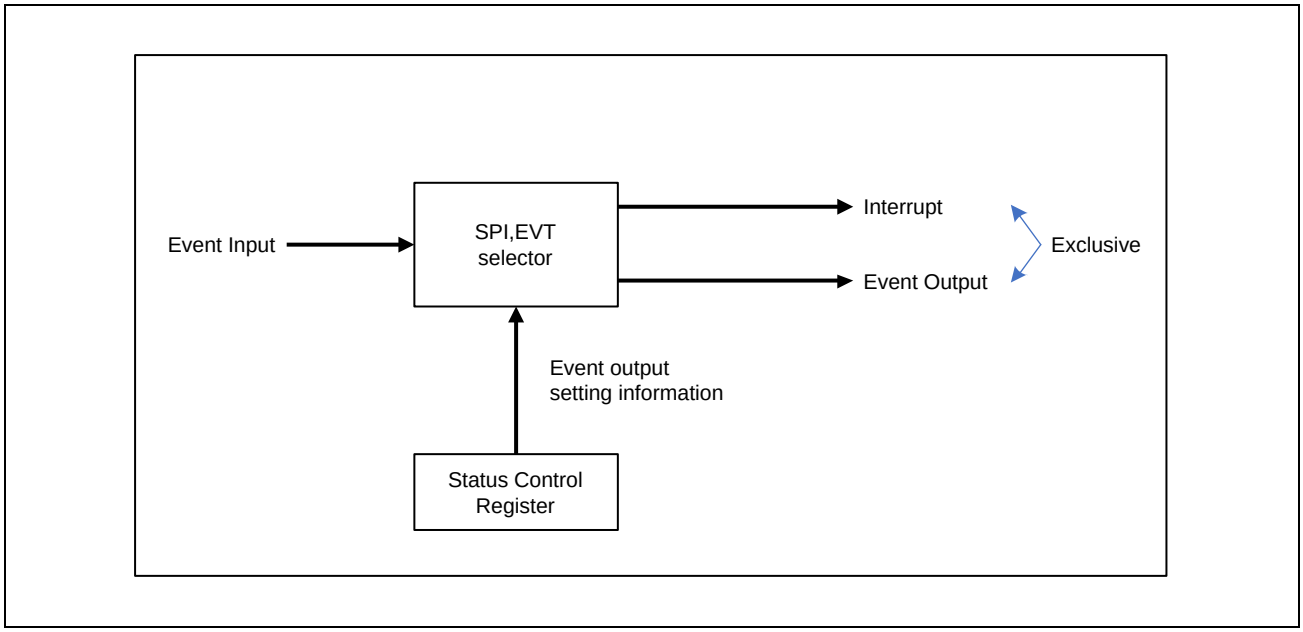


Figure 4.6-2 Exclusive Output Control of Interrupt and Event Output

(3) Exclusive output control of interrupts and DMAC requests

Exclusive output control applies to events with ✓ under the “DMAC” column and an entry under the interrupt column (“CA55” or “CM33”) in **Table 4.6-24**. For input events specified in an DMAC Factor k Selection Register y (DMkSELy), a DMA request is output but an interrupt is not. On the other hand, for input events not specified in a DMAC Factor k Selection Register y (DMkSELy), an interrupt is output.

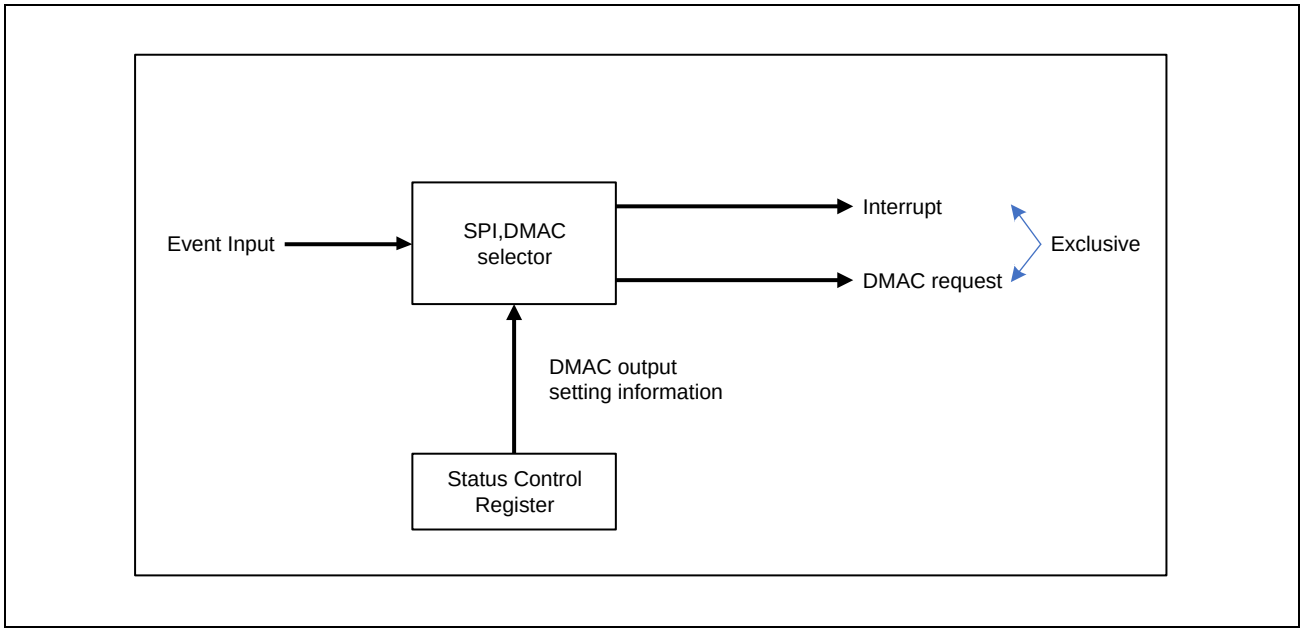


Figure 4.6-3 Exclusive Output Control of Interrupt and DMAC Request

Detecting an interrupt from the DMAC clears the setting of a DMAC Factor k Selection Register y (DMkSELy) for the corresponding DMAC channel. Change the setting to the value with all 1s (1_1111_1111b).

This clearing function can be enabled or disabled with the DMAC Factor k Selection Register y (DMkSELy).

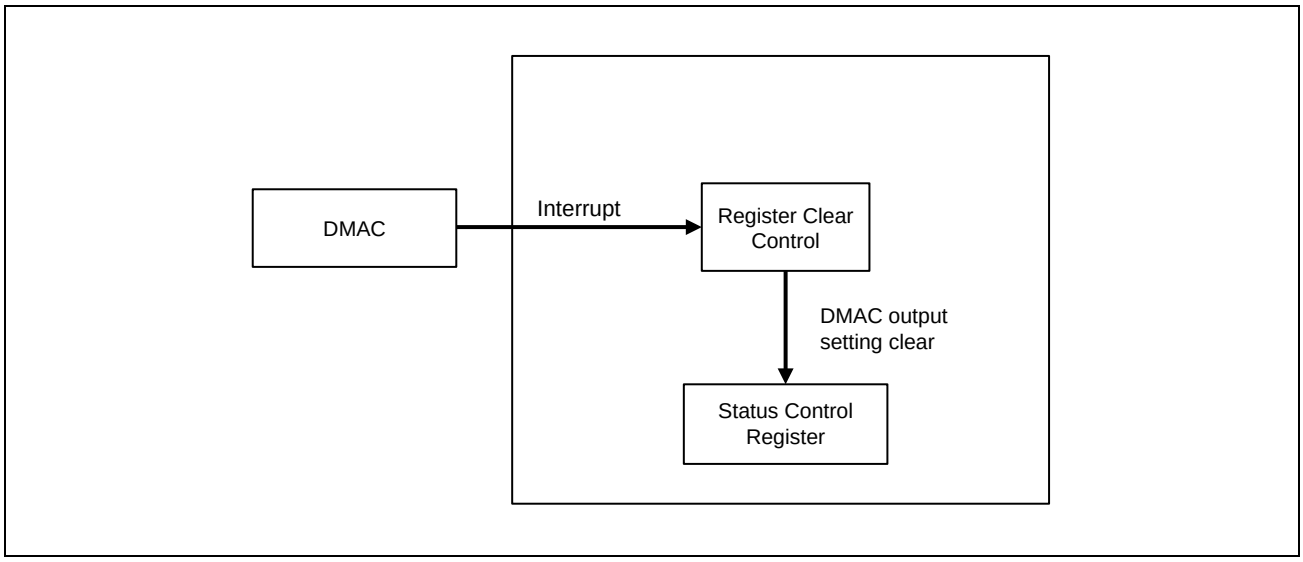


Figure 4.6-4 DMAC Request Setting Clearing

(4) Control over NMI

The configuration for control over NMI is shown in **Figure 4.6-5**. The NMI signal that is input from an external source to the LSI chip is input to the ICU through the GPIO noise filter. The ICU handles NMI synchronization, control over the interrupt state, and interrupt masking control.

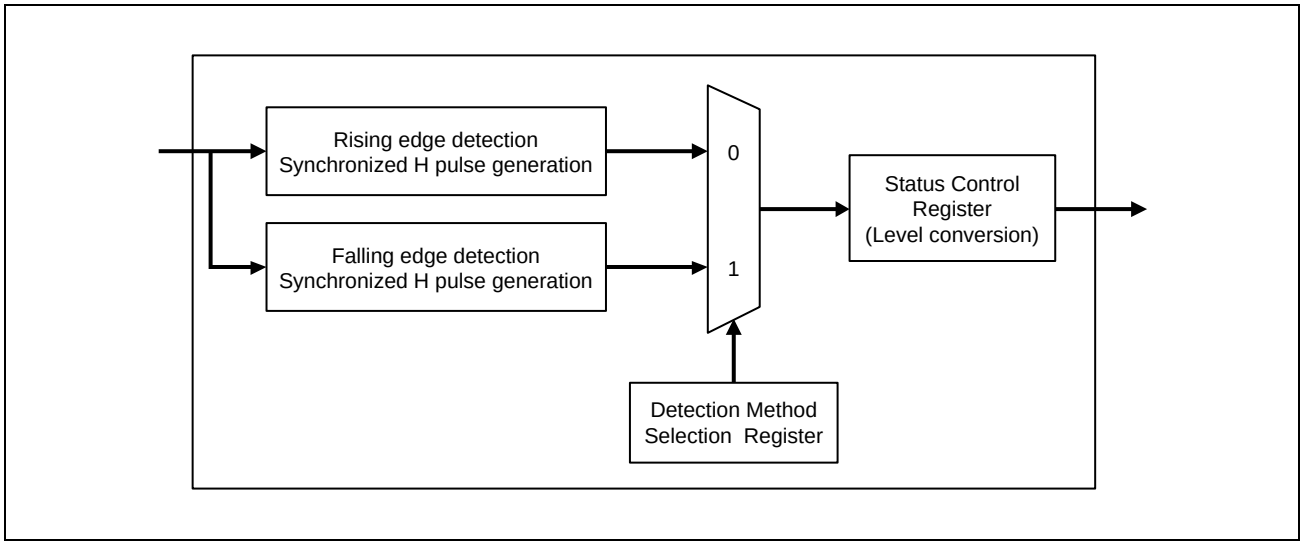


Figure 4.6-5 NMI Control Configuration

Selecting the method of interrupt detection

Two methods of detection are available for the NMI: falling edge detection and rising edge detection. The NMI detection method selection register (NITSR) is used to select either of them.

The setting of the NITSR can also be used for system control by referring to it from outside the ICU.

Retaining the interrupt detected state

When the NMI signal is asserted, the NMI status register (NSCNT) retains the detection of the interrupt.

Clearing the interrupt detected state

The interrupt detected state is cleared when 1 is written to the corresponding bit in the NMI status clear register (NSCLR).

(5) Control over IRQ

The configuration for control over IRQ is shown in the figure below. The IRQ0 to IRQ15 signals that are input from external sources to the LSI chip are input to the IM33 through the GPIO noise filters. The ICU handles IRQ synchronization, control over the interrupt state, and interrupt masking control.

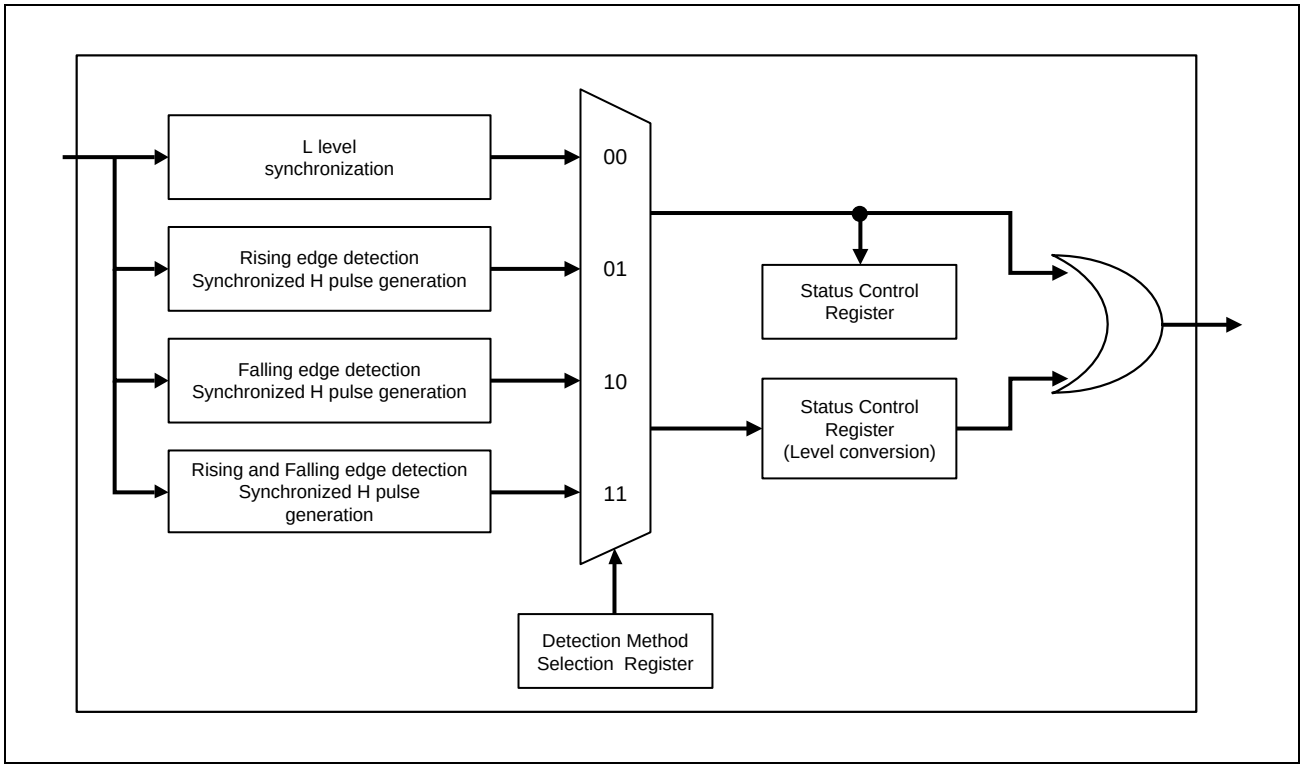


Figure 4.6-6 IRQ Control Configuration

Selecting the method of interrupt detection

Four methods of detection are available for the IRQ interrupts: low-level detection, falling edge detection, rising edge detection, and rising and falling edge detection. The IRQ interrupt detection method selection registers (IITSR) are used to select one of them.

The setting of IITSR can also be used for system control by referring to it from outside the ICU.

Retaining the interrupt detected state

When an IRQ signal is asserted, the IRQ status register (ISCTR) retains the detection of the interrupt.

Clearing the interrupt detected state

If level-sensing of an interrupt is selected in the IITSR register, the interrupt detected state is cleared when the interrupt signal is de-asserted by the source of the interrupt request.

If edge-sensing of an interrupt is selected in the IITSR register, the interrupt detected state is cleared when 1 is written to the corresponding ISCLR bit.

(6) Control over TINT

The configuration for control over TINT is shown in the figure below. The TNT0 to TNT31 signals that are input from external sources to the LSI chip are input as 141 IO_TOP_TINT signals to the ICU through the GPIO noise filters. The ICU handles TINT allocation, TINT synchronization, control over the interrupt state, and interrupt masking control.

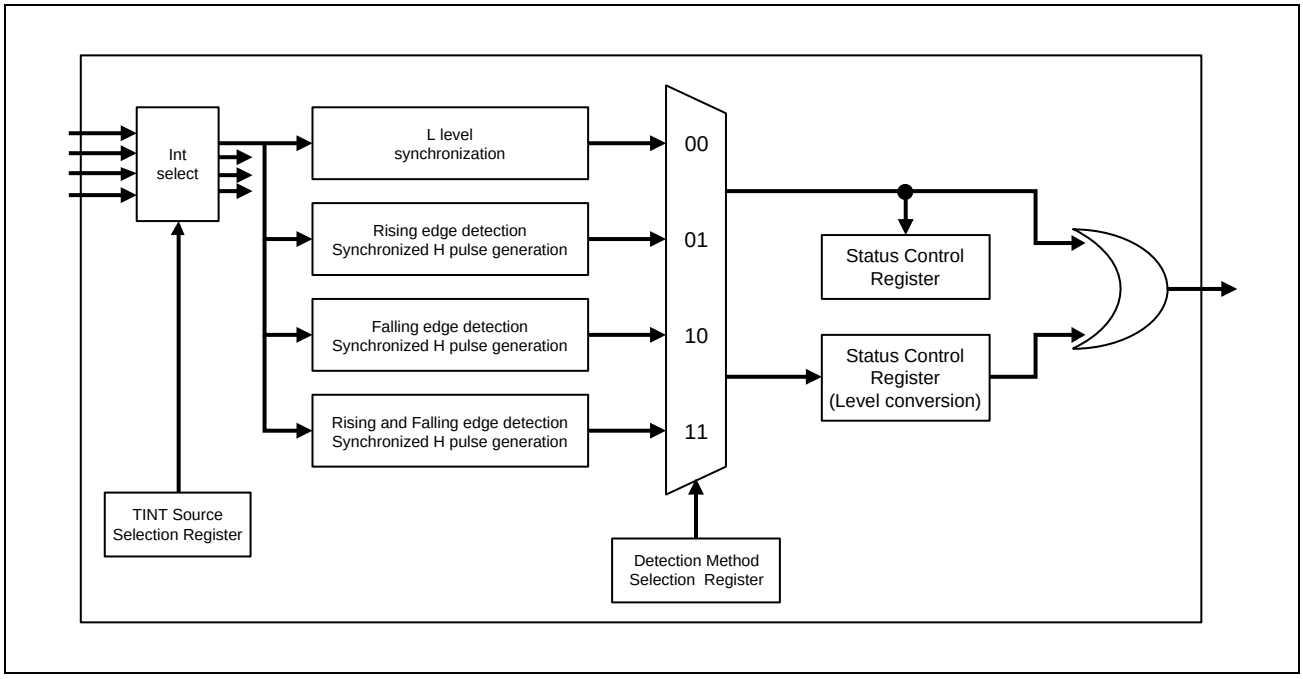


Figure 4.6-7 TINT Control Configuration

Allocating interrupts

32 desired signals from among the 141 IO_TOP_TINT signals are allocated to TINT0 to TINT31 according to the settings of the TINT source selection registers (TSSR0 to TSSR15).

Selecting the method of interrupt detection

Four methods of detection are available for the TINT interrupts: rising edge detection, falling edge detection, high-level detection, and low-level detection. The TINT detection method selection registers (TITSR0 and TITSR1) are used to select one of them.

The setting of TITSR can also be used for system control by referring to it from outside the ICU.

Retaining the interrupt detected state

When a TINT signal is asserted, the TINT status register (TSCTR) retains the detection of the interrupt.

Clearing the interrupt detected state

If level-sensing of an interrupt is selected in the TITSR0 or TITSR1 register, the interrupt detected state is cleared when the interrupt signal is de-asserted by the source of the interrupt request.

If level-sensing of an interrupt is selected in the TITSR0 or TITSR1 register, the interrupt detected state is cleared when 0 is written to the corresponding TSCR bit.

(7) Control over internal interrupts

The configuration for control over internal interrupts is shown in the figure below. The ICU handles synchronization of interrupt signals from the units and interrupt masking control.

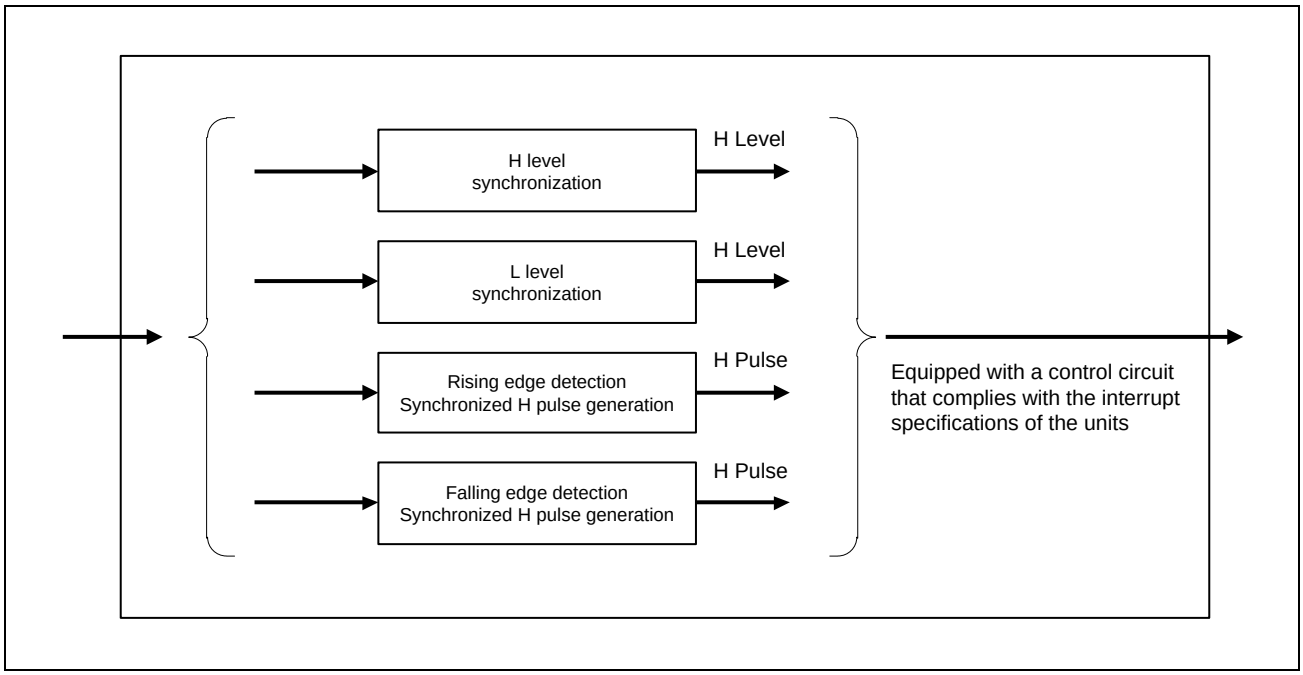


Figure 4.6-8 Internal Interrupt Control Configuration

Selecting the method of interrupt detection

Four methods of detection are available for the internal interrupts: low-level detection, high-level detection, falling edge detection, and rising edge detection. Synchronization proceeds in ways that suit the specifications of the units as interrupt request sources. The available methods are listed under “Level/Edge” and “Active” columns in **Table 4.6-24**.

4.6.1.3.2 CPU interrupt output

(1) Overview of CPU interrupt output

The ICU outputs event signals as outputs for the CA55 and CM33. The applicable events are those with “SYSTEM”, “COMMON”, “SINGLE” or “SELECT” under the “CA55” and “CM33” columns in **Table 4.6-24**. The CA55 and CM33 each output these signals.

To use interrupts classified as “SELECT”, use **(20) SELECT Interrupt CM33 Selection Register k (ICU_INTM33SELk)** to specify the interrupts.

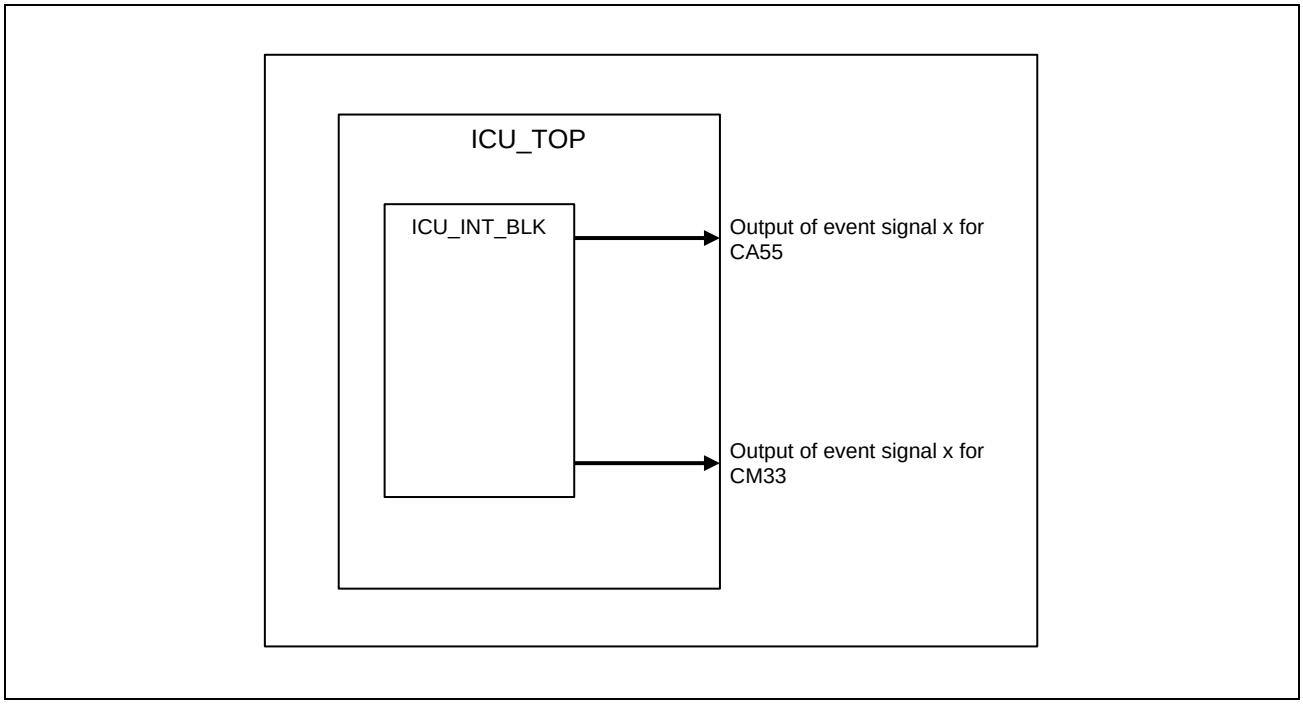


Figure 4.6-9 Interrupt Output

(2) Control over the bundling of bus error interrupts

The configuration for control over the bundling of bus error interrupts is shown in **Figure 4.6-10**. The ICU handles synchronization of bus error interrupts generated by the system bus, control over the interrupt state, and interrupt bundling control. Furthermore, it handles interrupt masking control.

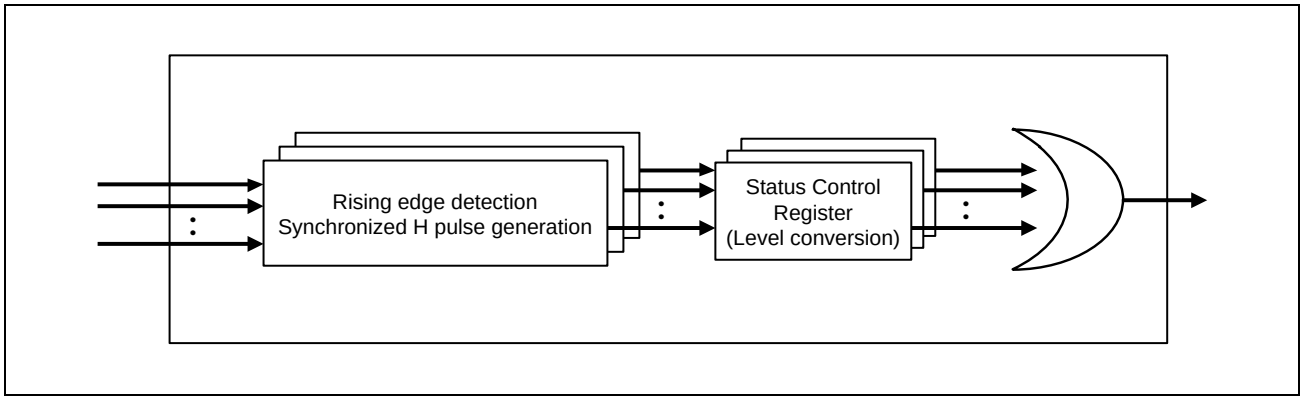


Figure 4.6-10 Bus Error Bundle Control

Retaining the interrupt detected state

When a system bus error signal is asserted, a bus error interrupt status register k (BEISRk) retains the detection of the interrupt.

Clearing the interrupt detected state

The interrupt detected state is cleared when 1b is written to the corresponding bus error interrupt status clear register k (BECLRk) bit.

Bundling of interrupts

The 42 bus error signals output by the system bus are bundled into a single bus error interrupt signal.

For the list of events, see **Table 4.6-26**.

(3) Control over the bundling of SRAM error interrupts

The configuration for control over the bundling of SRAM error interrupts is shown in **Figure 4.6-11**. The ICU handles synchronization of SRAM error interrupts, control over the interrupt state, interrupt bundling control, and interrupt masking control.

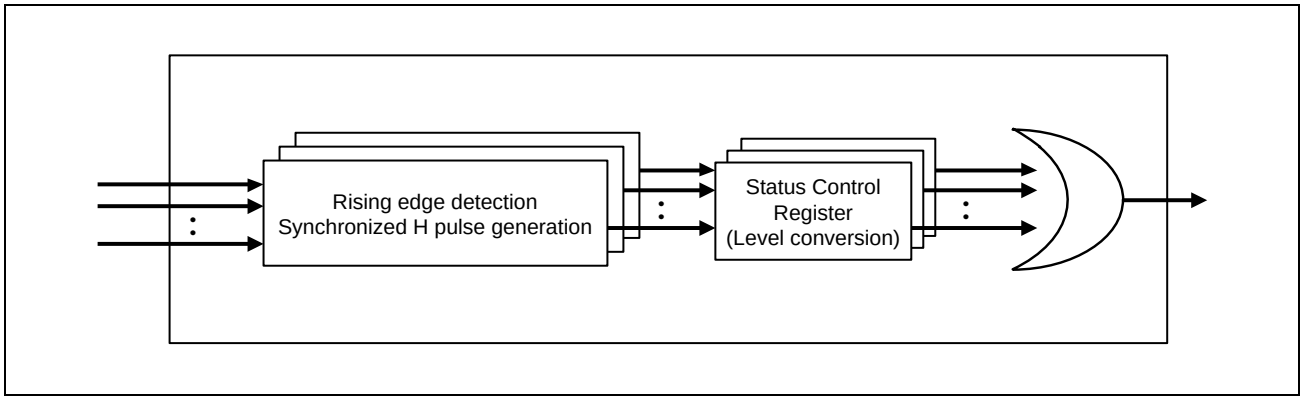


Figure 4.6-11 SRAM Error Bundle Control

Retaining the interrupt detected state

When an SRAM error signal is asserted, an SRAM error interrupt status register k (EREISRk) retains the detection of the interrupt.

Clearing the interrupt detected state

The interrupt detected state is cleared when 1b is written to the corresponding SRAM error interrupt status clear register k (ERCLRk) bit.

Bundling of interrupts

Error interrupts output by the SRAM are bundled into a single interrupt signal.

For the list of events, see **Table 4.6-27**.

(4) Control over the bundling of error interrupts

Error interrupt signals are output individually for the CA55 and CM33. When an error interrupt is generated, each factor is retained in **(21) Error Interrupt CM33 Status Register k (ICU_ERINTM33CTLk)** or **(24) Error Interrupt CA55 Status Register k (ICU_ERINTA55CTLk)**.

For the bundled bus error signal (BUS_ERR_INT) and SRAM error interrupt signal (RAM_ERR_INT), the factors before bundling of the respective signals are retained, so the bundled error signals themselves are not covered.

The factors are retained until they are cleared. The interrupt signals for the CM33 and CA55 (ICU_Error_CM33 and ICU_Error_CM55 in **Table 4.6-24**) are output according to the results of the logical OR of the source factors. (The interrupt signals continue to be output until all factors are cleared.) In addition, the masking function that determines whether or not to retain the factor can be used for each factor.

The target events for error interrupts are those with ✓ under the “Error Event” column in **Table 4.6-24**.

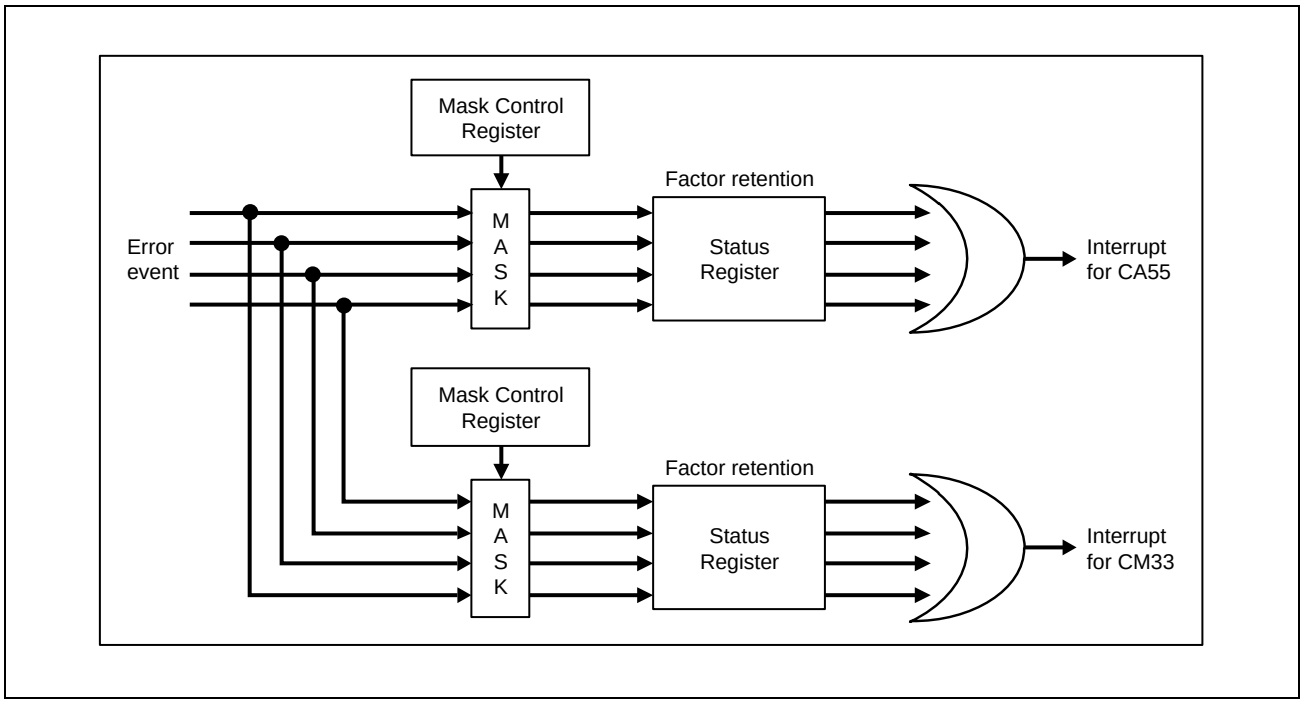


Figure 4.6-12 Error Interrupt Bundle Control

Retaining the interrupt detected state

When an error interrupt signal is asserted, an error interrupt CM33 status register k (ERINTM33CTLk) or error interrupt CA55 status register k (ERINTA55CTLk) retains the detection of the interrupt.

Clearing the interrupt detected state

The interrupt detected state is cleared when 1 is written to the corresponding interrupt status clear register bit.

Bundling of interrupts

Error interrupts are bundled per interrupt source for CPU (CM33 or CA55) into a single interrupt signal.

Points to note in processing per CPU

Registers for masking, retaining, and clearing source factors are provided individually for the CM33 and CA55. Even if the factor on one CPU side is cleared on generation of an error interrupt, the error interrupt on the other CPU may continue to be output.

Apply countermeasures such as masking error interrupts for CPUs that do not require them or clearing the factors on both CPU sides as required.

Operations involved in LSI error reset generation

The registers listed below can only be reset by ERROR_RESETn that is generated following a system reset of the LSI.

Accordingly, the settings of the registers listed below are retained when an LSI error reset is generated.

- **(15) Bus Error Interrupt Status Register k (ICU_BEISRk)**
- **(17) ECCRAM Error Interrupt Status Register k (ICU_EREISRk)**
- **(21) Error Interrupt CM33 Status Register k (ICU_ERINTM33CTLk)**
- **(24) Error Interrupt CA55 Status Register k (ICU_ERINTA55CTLk)**

If it returns to normal operation after error reset, it is necessary to confirm error status and clear error factors after restarting.

(5) Control over the bundling of GPT interrupts for the CA55

The configuration for control over the bundling of GPT interrupts (gpt_gtciada_n_0 to gpt_gtciada_n_7 and gpt_gtciadb_n_0 to gpt_gtciadb_n_7) is shown in **Figure 4.6-13**. The ICU handles synchronization of GPT interrupts, control over the interrupt state, and interrupt bundling control.

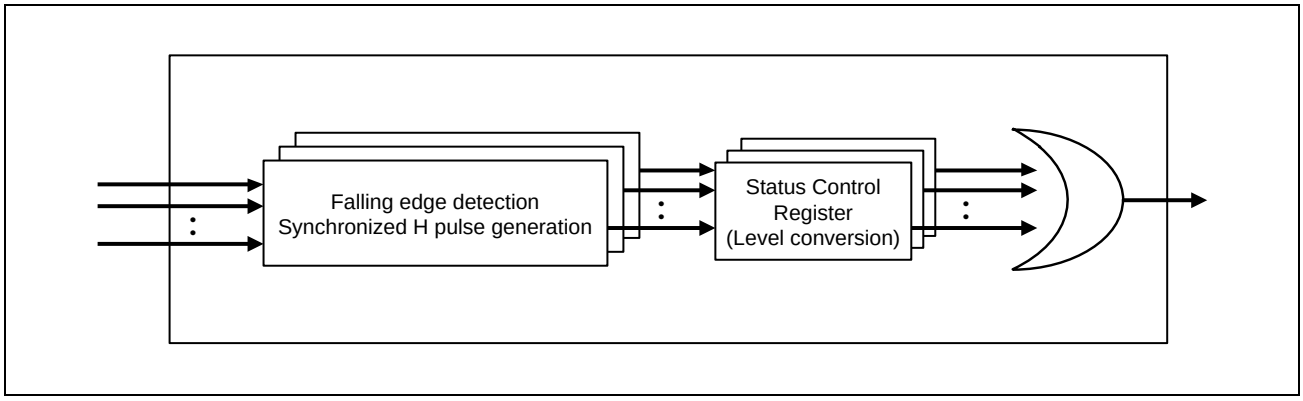


Figure 4.6-13 GPT Interrupt Bundle Control

Retaining the interrupt detected state

When any of the GPT interrupt signals (gpt_gtciada_n_0 to gpt_gtciada_n_7 and gpt_gtciadb_n_0 to gpt_gtciadb_n_7) is asserted, the GPT interrupt CA55 status register (GPTINTA55CTL) retains the detection of the interrupt.

Clearing the interrupt detected state

The interrupt detected state is cleared when 1 is written to the corresponding GPT interrupt CA55 status clear register (GPTINTA55CLR) bit.

Bundling of interrupts

The GPT interrupts (gpt_gtciada_n_0 to gpt_gtciada_n_7 and gpt_gtciadb_n_0 to gpt_gtciadb_n_7) are bundled per interrupt factor into a single interrupt signal. The interrupt factors are gpt_gtciada_n_0 to gpt_gtciada_n_7 for channel 0, gpt_gtciadb_n_0 to gpt_gtciadb_n_7 for channel 0, gpt_gtciada_n_0 to gpt_gtciada_n_7 for channel 1, and gpt_gtciadb_n_0 to gpt_gtciadb_n_7 for channel 1. The respective eight interrupt factors are bundled into a single interrupt signal (gpt_gtciada_n or gpt_gtciadb_n).

(6) Generating enhancing pseudo error interrupts

The generation of pseudo error factors is possible. A software-enhancing pseudo error is generated by writing 1 to the corresponding bit in the SW enhancing pseudo error generation register (SWPE) and behaves like an actual error interrupt. The target interrupts are SW_PSEUDO_ERR_0 to SW_PSEUDO_ERR_15 in **Table 4.6-24**.

(7) Assignment of CPU interrupts

The numbers of specifiable interrupt signals for the CA55 and CM33 are 960 and 480, respectively.

The interrupts signals are assigned as shown in the table below. Details on the assignment are shown under the “CA55” and “CM33” columns in **Table 4.6-24**.

Table 4.6-19 Interrupt Assignment

SPI No.	CA55	CM33
0 to 251	COMMON	
252 to 352	SYSTEM (CA55)	SYSTEM (CM33)
353 to 479	SINGLE (CA55)	SELECT (CM33)
480 to 959		Reserved

Table 4.6-20 Outline of Each Item

Class	Description
SYSTEM (CA55)	System-related interrupt signals for the CA55 (allocated as independent interrupts with the SPI numbers)
SYSTEM (CM33)	System-related interrupt signals for the CM33 (allocated as independent interrupts with the SPI numbers)
COMMON	Interrupt signal common to the CA55 and CM33 (allocated as an interrupt with a common SPI number)
SINGLE (CA55)	Interrupt signals for the CA55 (allocated as independent interrupts with the SPI numbers)
SELECT (CM33)	Interrupt signals for the CM33 (selectable from among 538 interrupt signals)

Note: The SPI numbers per CPU for SYSTEM (CA55), SYSTEM (CM33), COMMON, and SINGLE (CA55) are shown under the "CA55 SPI No." and "CM33 SPI No." columns in **Table 4.6-24**.

Note: For SELECT (CM33), signals are selectable so the fixed SPI numbers are not allocated. These signals are described as "SELECT" under the "CA55" and "CM33" columns in **Table 4.6-24**.

Selecting SELECT (CM33)

If the value set for the selsig[8:0] select signal from among event input signals for SELECT is x, the insel[x] signal is selected and output. Note that if the value is all 1s (3FFh), the fixed value 0 is output.

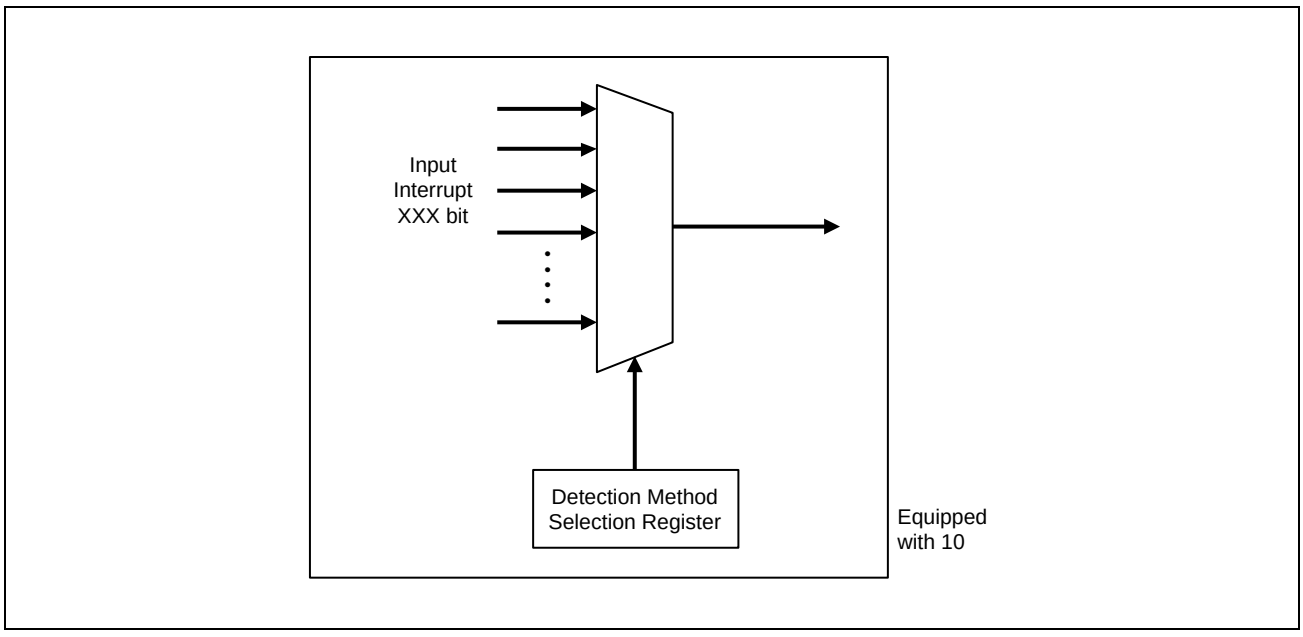


Figure 4.6-14 SELECT (CM33) Control

(8) Pulse or level for CPU interrupts

The active level and pulse or level operation for interrupt signals output to the CPUs are listed below.

The output signals are active high regardless of the input active level. For pulse or level, whichever of the two selected on the input side is output.

Table 4.6-21 Interrupt Sense

Input		Output	
Active	Pulse/Level	Active	Pulse/Level
High	Pulse	High	Pulse
Low	Pulse	High	Pulse
High	Level	High	Level
Low	Level	High	Level

4.6.1.3.3 Control over the event link controller (ELC)

The ICU has a function of the event link controller (ELC). The ELC selects the destinations for output of the event signals input from the units in the ICU and then outputs the signals to the selected destinations.

The figure below is a schematic view of the data flow by the ELC.

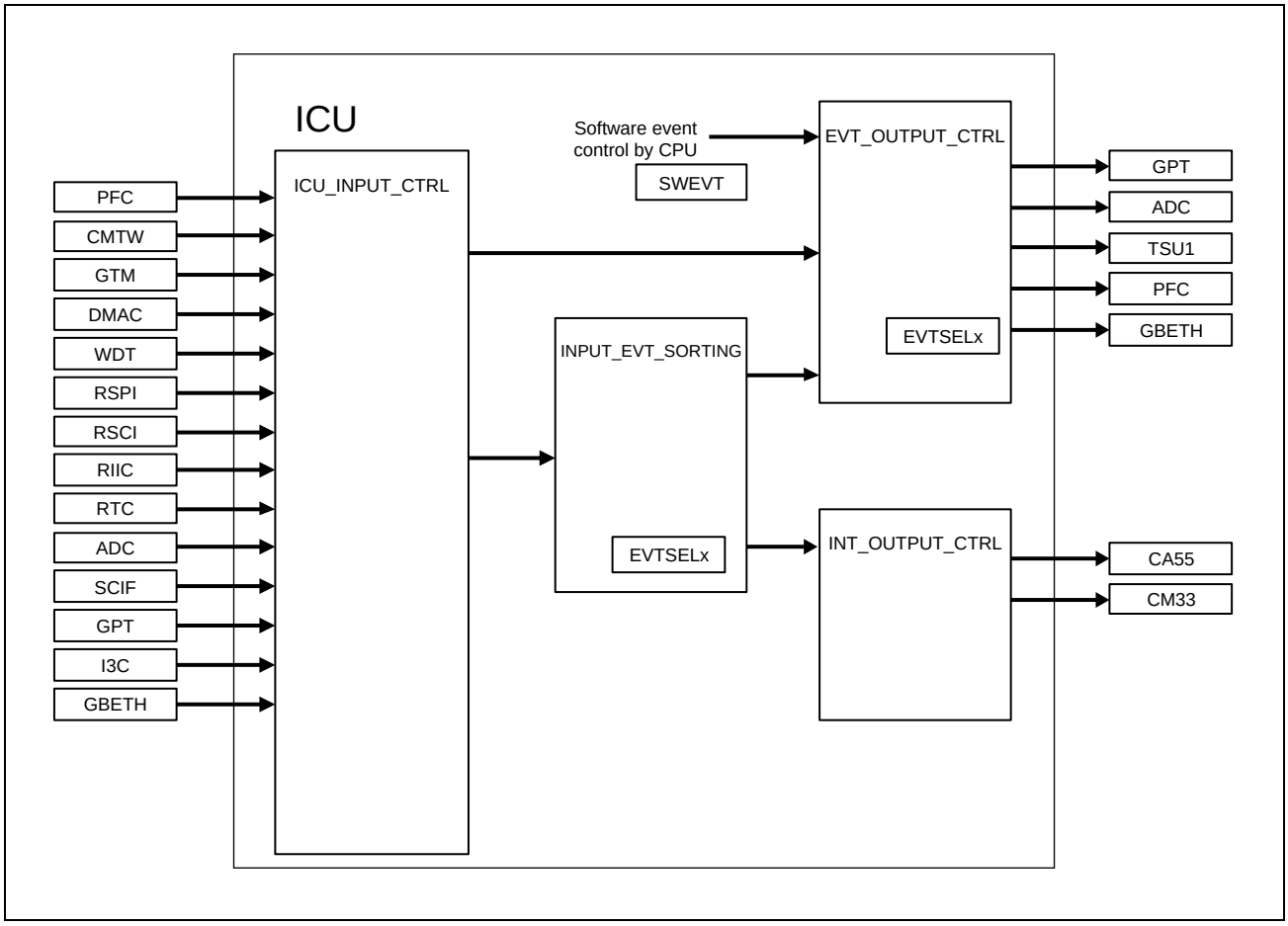


Figure 4.6-15 ELC Flow Diagram

The following three types of input are available.

- ELC-dedicated input (inputs from ICU_INPUT_CTRL)
- Input for use as both interrupts and ELC events (inputs from INPUT_EVT_SORTING)
- Input of an event generated under control of the ICU registers

(1) Event input

The signals with ✓ under the “ELC” column in **Table 4.6-24** are used as the input signals for the ELC.

(2) Generating a software event

Software events can be generated. A software event is generated by writing 1 to the corresponding bit in the ICU_SW_EVT register (SWEVT) and behaves in the same way as an event signal input by another unit. The target events are SW_EVENT_0 to SW_EVENT_6 in **Table 4.6-24**.

(3) Event output

The ELC outputs the event signals to the units. The destination units and port names for output are listed in **Table 4.6-25**. The input source can be selected per output event signal. Specifying the value of the corresponding event number in the event output factor selection register (EVTSELk) allows the selection of a desired event.

CAUTION

Using a DMAC REQ signal generated by an ICU_SW_DMAC register k (SWDMACk) as an event output is not allowed.

4.6.1.3.4 Control over the DMAC

The figure below is a schematic view of the data flow of the DMAC REQ signal.

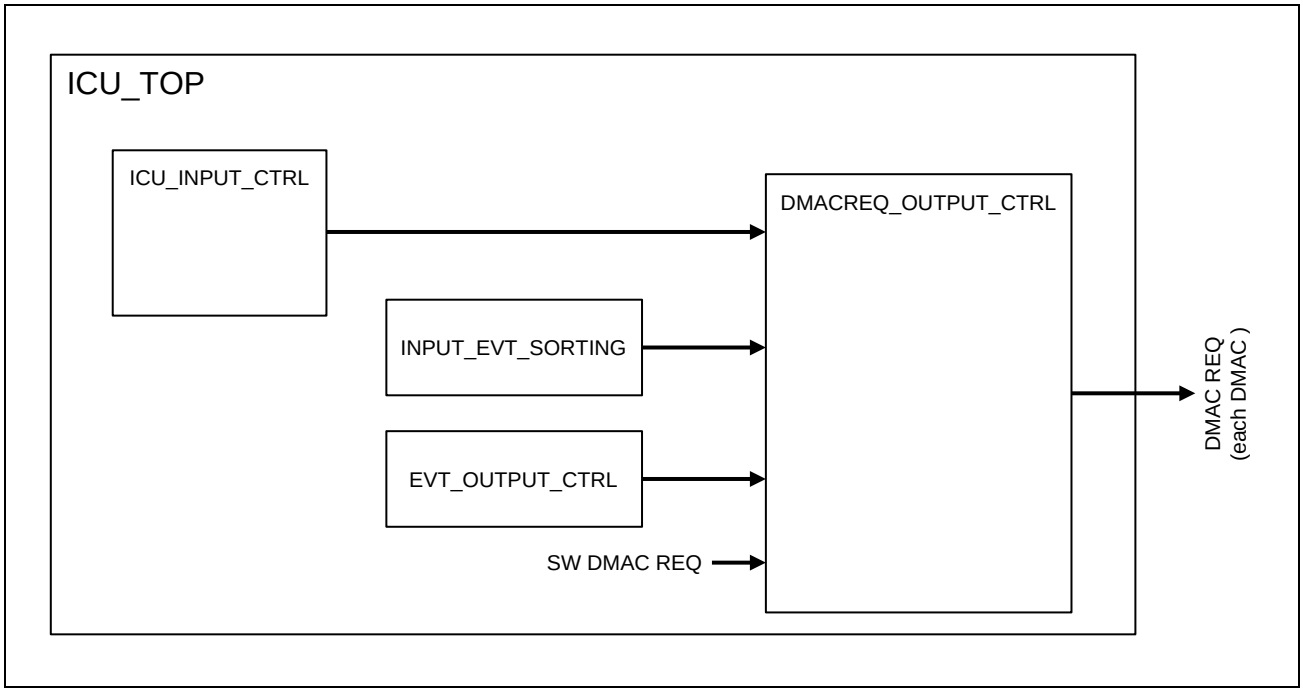


Figure 4.6-16 DMAC REQ Flow Diagram

The following four types of input are available.

- DMAC REQ-dedicated input (inputs from ICU_INPUT_CTRL)
- Input for use as both interrupts and DMAC REQ signals (inputs from INPUT_EVT_SORTING)
- Input when event output is used as a DMAC REQ signal (inputs from EVT_OUTPUT_CTRL)
- Input of an event generated under control of the ICU registers

(1) DMAC request output control module

This module selects the sources for signals (active high) processed in the input section (ICU_INPUT_CONT) for each of the event signals for the modules and outputs the signals to the DMAC as DMAREQ signals for the DMAC.

(2) Processing when DMAC request selections overlap

When the DMAC factor selection register (DMkSELy) is set, the ICU checks for duplication of the register bit setting with those of another DMAC factor selection register (DMkSELy) or the other bit field in the same register.

When duplication of the register bit setting with those of another register or the other bit field in the same register is detected, the ICU proceeds with the following processing.

- “Clears” the setting of the DMAC factor selection register bits (DkRQ_SEL(y × 2+1) or DkRQ_SEL(y × 2)) that caused the duplication of the setting of the bits in another register or the other field in the same register; that is, the bits are set to 3FFh (all effective bits are 1).
- Sets 1 in the bit of a DMAC request repetition control register k (DMRCTRk) corresponding to the DMAC channel for which the DMAC request setting caused the duplication of the setting of the bits in another register or the other field in the same register.
- The value set in the DMAC request repetition control register k (DMRCTRk) can be cleared by writing 1b to the corresponding bit of a DMAC request repetition clear register k (DMRCLRk).

CAUTION

In writing to the DMAC Factor k Selection Register y (DMkSELy), setting the same value in the higher-order (DkRQ_SEL(y × 2 + 1)) and lower-order (DkRQ_SEL(y × 2)) bits is prohibited.

If the same value has been set, the state information (DMADUP*_n) of the DMRCTRk registers, the factor settings (D*RQ_SEL*) of the DMkSELy registers, and the behavior of the overlap interrupt signal are not guaranteed.

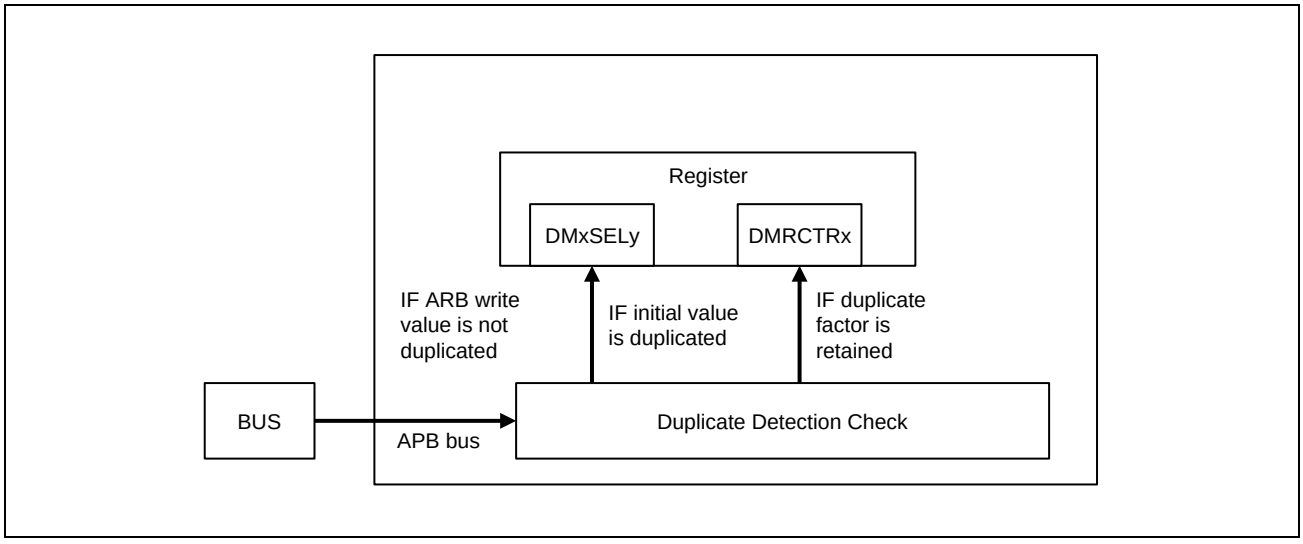


Figure 4.6-17 DMAC Selection Duplication Control

(3) Automatic clearing DMAC factor selection registers

k = 0 to 4, y = 0 to 7, z = 0 to 15

The ICU has a function for automatically clearing the settings (all effective bits are 1) of the DkRQ_SEL(y × 2) and DkRQ_SEL(y × 2 + 1) bits in the DMAC Factor k Selection Register y (DMkSELy) in response to receiving a DMAC end interrupt.

This function is triggered by the DMAC end interrupts (DMACk_INTz: k = 0 to 4, z = 0 to 15). Receiving an interrupt clears the settings (all effective bits are 1) of the DkRQ_SEL(y × 2) and DkRQ_SEL(y × 2 + 1) bits.

The DkSEL_CLRON(y × 2) and DkSEL_CLRON(y × 2 + 1) bits can be used to enable or disable this function.

- When the values of the DkSEL_CLRON(y × 2) and DkSEL_CLRON(y × 2 + 1) bits are 1b: Automatic clearing proceeds.
- When the values of the DkSEL_CLRON(y × 2) and DkSEL_CLRON(y × 2 + 1) bits are 0b: Automatic clearing does not proceed.

(4) Pulse or level for DMAC request output

The active level and pulse or level operation for REQ signals output to the DMACs are listed below.

The output signals are active high regardless of the input active level. For pulse or level, whichever of the two selected on the input side is output.

Table 4.6-22 DMAC REQ Sense

Input		Output	
Active	Pulse/Level	Active	Pulse/Level
High	Pulse	High	Pulse
Low	Pulse	High	Pulse
High	Level	High	Level
Low	Level	High	Level

(5) Control over response to the DMAC ACK and DMAC TEND signals

The DMAC ACK signals (80 in all) are input from the DMAC channels. The ICU selects the DMAC ACK signals output in response to the DMAC ACK signals of the units according to the settings of the DMAC ACK selection registers x (DMACKSELk).

For a DMAC ACK signal as an input, set it for active-high level detection. When it is an output, set it for active-high or active-low level detection. Details on the active level (high or low) are described under the “Active” column in **Table 4.6-28**.

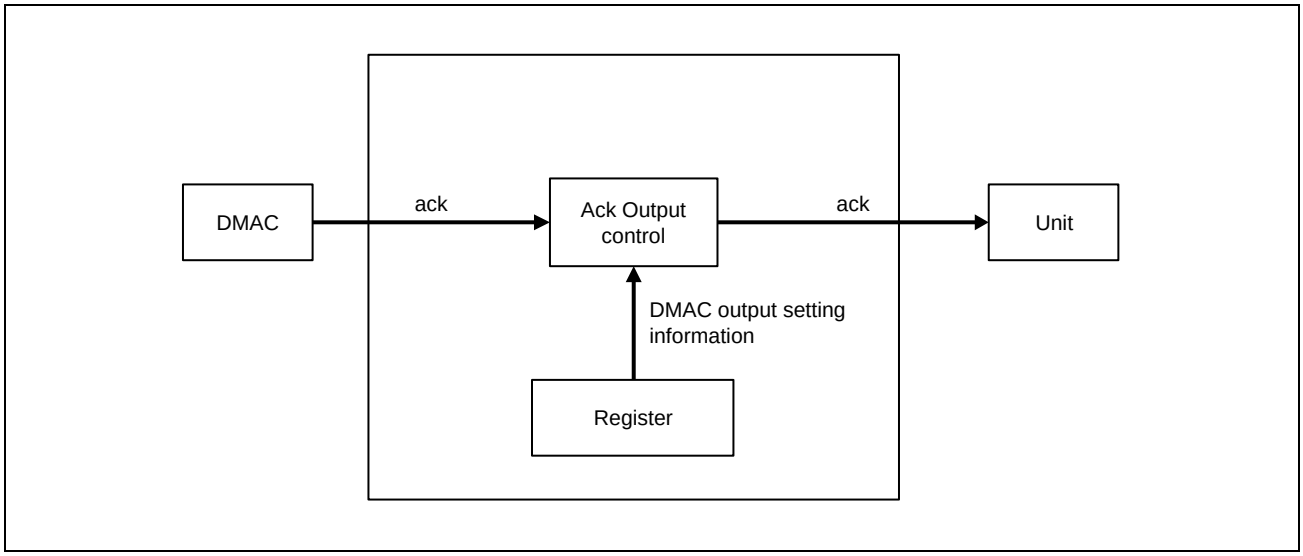


Figure 4.6-18 DMAC ACK Control

The DMAC TCO signals (80 in all) are input from the DMAC channels. Select the DMAC TCO signals to be output for the DTEND0 to DTEND4 signals of the PFC (IOTOP) according to the settings of the DMAC TEND selection registers k (DMTENDSELk).

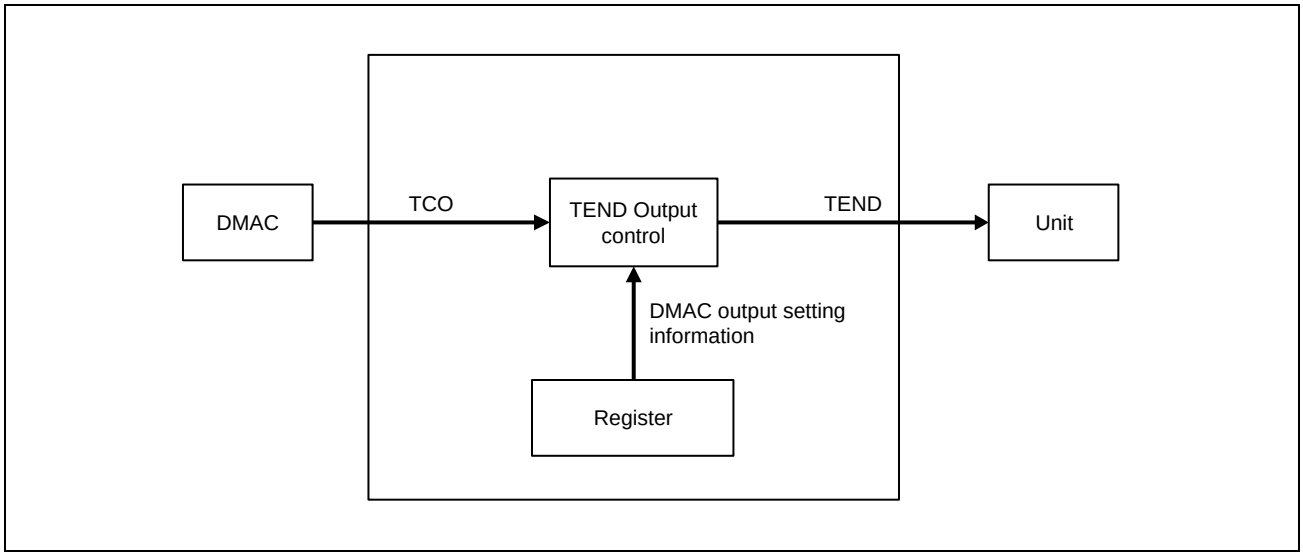


Figure 4.6-19 DMAC TEND Control

(6) DMAC request output with the use of event output

The ICU has a function for using event outputs as DMAC request outputs.

Eight event outputs, EVT_DMAC_0 to EVT_DMAC_7 in **Table 4.6-25** can be used as DMAC request outputs.

4.6.1.3.5 Other control

(1) CM33 interrupt output masking control

The figure below shows the configuration for outputting interrupt signals to the CPUs. All interrupt signals for the CM33 can be masked with the CM33_INTMASK signal (CM33_INTMASK signal can be controlled by the INTMASK_CM33_I bit of the CPG_LP_CTL2 register in the CPG). The interrupt signals for the CA55 cannot be masked.

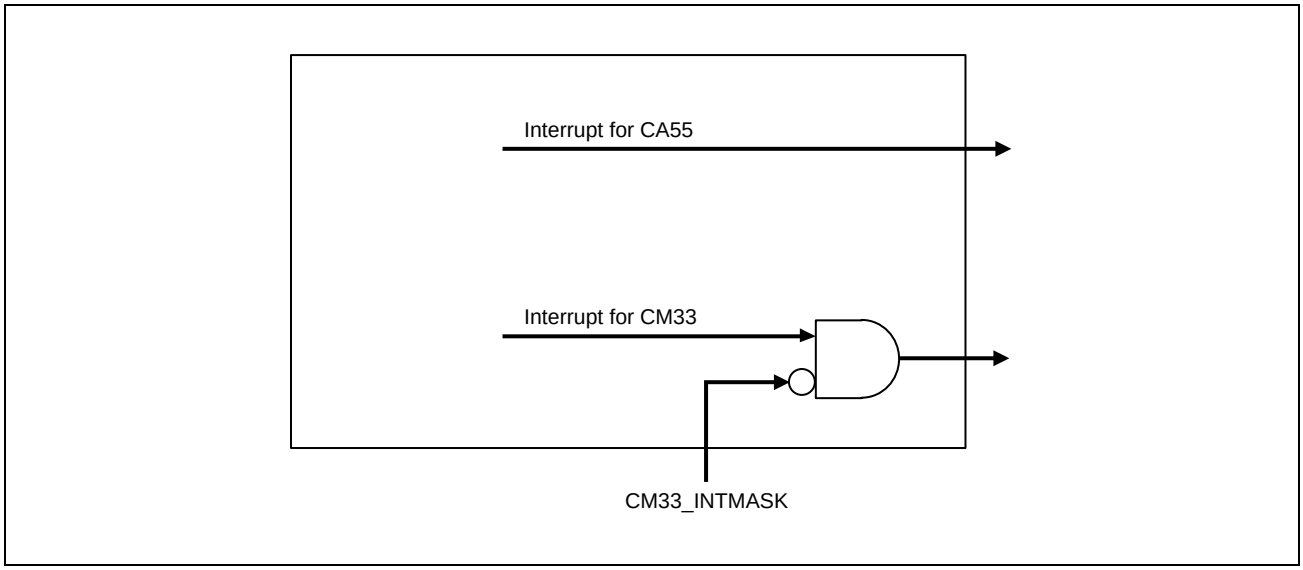


Figure 4.6-20 CM33 Interrupt Masking Control

(2) Retaining factors in access to the off domain

The ICU has a function for retaining the factor on generation of a DMAC request for a DMAC in the other area or an event output to a unit when the other area of this LSI chip is disabled. When the on/off information signal in the other area that is input to the ICU is active, this function monitors generation of DMAC requests for the other area or event output. When such access is attempted, the ICU retains the factor in the corresponding register.

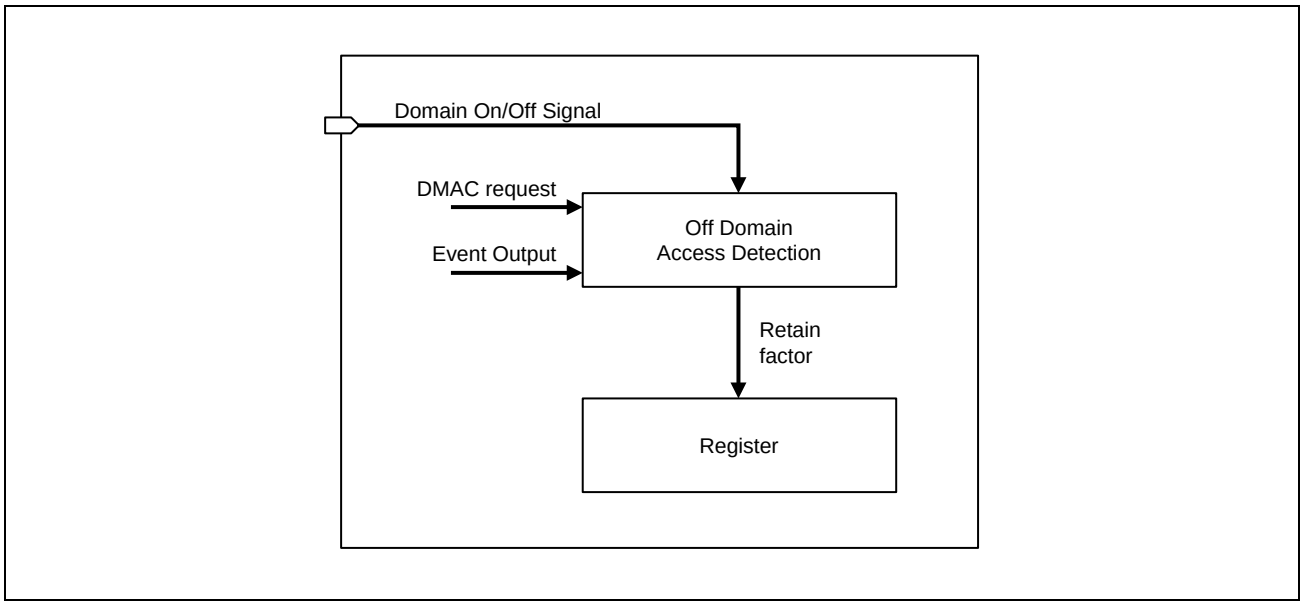


Figure 4.6-21 Off Domain Access Control

(3) Error signal output for the CPG

The ICU has a function for outputting error signals to the CPG. The ICU outputs the values of the error interrupt CM33 status registers k (ERINTM33CTLk) to pass the values retaining the error factors to the CPG. These retained signals can be cleared by writing to the error interrupt CM33 status clear registers k (ERINTM33CLRk).

(4) Register access security control

Each register of the ICU has a security control function. This function is controlled with the use of two pairs of SLEVEL[1:0] input signals from the SYS registers (MPICU0_SL and MPICU1_SL), which respectively control the registers of group 0 and group 1. For the assignment of the registers to groups 0 and 1, see **4.6.1.2.2 Register Description**. Details on control with the use of the SLEVL[1:0] signals are given in the table below.

Table 4.6-23 Secure Control Table

	Non-secure		Secure	
SLEVEL[1:0]	User	Privileged	User	Privileged
00b	RD + WR	RD + WR	RD + WR	RD + WR
01b	None	RD + WR	RD + WR	RD + WR
10b	None	None	RD + WR	RD + WR
11b	None	None	None	RD + WR

(5) Masking control

This unit has the following functions for masking output signals. For details on the respective functions, see the sections listed below.

- Masking all interrupts for the CM33
  - **(1) CM33 interrupt output masking control**
- Individually masking the error interrupts for the CM33
  - **(4) Control over the bundling of error interrupts**
- Individually masking the error interrupts for the CA55
  - **(4) Control over the bundling of error interrupts**

4.6.1.4 Restrictions

- Restriction on operation with the emergency clock

  This unit handles output based on the clock frequency ratio during normal operation with the width which can be received by the output destination unit. The unit, however, does not support the frequency ratio during operation with the emergency clock. Therefore, operation with the emergency clock of the unit is not guaranteed.

- Restriction on the interval for the input of events and interrupts

  Signals to the ICU from the other units must be input with an interval of at least 34 cycles at 100 MHz between them. If an interval is shorter than this, input signals may be lost.

- Using a DMAC REQ signal generated by an ICU_SW_DMAC register k (SWDMACk) as an event output is not allowed.

- Writing to registers is in 32 bits. When writing to them, other bits of the same register should not be affected (for example, use a read‑modify‑write operation).

- The inputs from the PFC (PORT_IRQ0 to PORT_IRQ15) following a reset are at the low level, so they are asserted by SPI nos. 1 to 16 for the CPU. De-assert them before using them.

- Error interrupt signals are asserted after release from the error reset state. De-assert them before using them.

- The DMAC REQ signals output from the ICU are active high.

- Use of the DMAC DTEND signals outside the LSI chip

- These signals are output with a pulse width of 16T at 100 MHz as stated in this manual.

4.6.1.5 Event List

The interrupt events related to the ICU are listed below and on the following pages.

4.6.1.5.1 List of input events

Table 4.6-24 List of Input Events (1/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
0	0				PFC	—	NMI	Non-maskable interrupt	Edge	COM MON	COM MON	×	×	×
1	1		0		PFC	—	PORT_IRQ0	External pin interrupt 0	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
2	2		1		PFC	—	PORT_IRQ1	External pin interrupt 1	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
3	3		2		PFC	—	PORT_IRQ2	External pin interrupt 2	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
4	4		3		PFC	—	PORT_IRQ3	External pin interrupt 3	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
5	5		4		PFC	—	PORT_IRQ4	External pin interrupt 4	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
6	6		5		PFC	—	PORT_IRQ5	External pin interrupt 5	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
7	7		6		PFC	—	PORT_IRQ6	External pin interrupt 6	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
8	8		7		PFC	—	PORT_IRQ7	External pin interrupt 7	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
9	9		8		PFC	—	PORT_IRQ8	External pin interrupt 8	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
10	10		9		PFC	—	PORT_IRQ9	External pin interrupt 9	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
11	11		10		PFC	—	PORT_IRQ10	External pin interrupt 10	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
12	12		11		PFC	—	PORT_IRQ11	External pin interrupt 11	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
13	13		12		PFC	—	PORT_IRQ12	External pin interrupt 12	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
14	14		13		PFC	—	PORT_IRQ13	External pin interrupt 13	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
15	15		14		PFC	—	PORT_IRQ14	External pin interrupt 14	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
16	16		15		PFC	—	PORT_IRQ15	External pin interrupt 15	Low-level detection/rising edge/falling edge/both edges	COM MON	COM MON	×	✓	×
17	17		53		GTM	CH0	GTM0_GTMTINT	Interrupt output signal.	Edge	COM MON	COM MON	×	✓	×
18	18		54		GTM	CH1	GTM1_GTMTINT	Interrupt output signal.	Edge	COM MON	COM MON	×	✓	×
19	19		55		GTM	CH2	GTM2_GTMTINT	Interrupt output signal.	Edge	COM MON	COM MON	×	✓	×
20	20		56		GTM	CH3	GTM3_GTMTINT	Interrupt output signal.	Edge	COM MON	COM MON	×	✓	×

Table 4.6-24 List of Input Events (2/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
21	21		57		GTM	CH4	GTM4_GTMTINT	Interrupt output signal.	Edge	COMMON	COMMON	×	✓	×
22	22		58		GTM	CH5	GTM5_GTMTINT	Interrupt output signal.	Edge	COMMON	COMMON	×	✓	×
23	23		59		GTM	CH6	GTM6_GTMTINT	Interrupt output signal.	Edge	COMMON	COMMON	×	✓	×
24	24		60		GTM	CH7	GTM7_GTMTINT	Interrupt output signal.	Edge	COMMON	COMMON	×	✓	×
25	25		61		DMAC	CH1	DMAC1_DMAEND_0	DMAC1 transfer completion 0	Level/edge	COMMON	COMMON	×	✓	×
26	26		62		DMAC	CH1	DMAC1_DMAEND_1	DMAC1 transfer completion 1	Level/edge	COMMON	COMMON	×	✓	×
27	27		63		DMAC	CH1	DMAC1_DMAEND_2	DMAC1 transfer completion 2	Level/edge	COMMON	COMMON	×	✓	×
28	28		64		DMAC	CH1	DMAC1_DMAEND_3	DMAC1 transfer completion 3	Level/edge	COMMON	COMMON	×	✓	×
29	29		65		DMAC	CH1	DMAC1_DMAEND_4	DMAC1 transfer completion 4	Level/edge	COMMON	COMMON	×	✓	×
30	30		66		DMAC	CH1	DMAC1_DMAEND_5	DMAC1 transfer completion 5	Level/edge	COMMON	COMMON	×	✓	×
31	31		67		DMAC	CH1	DMAC1_DMAEND_6	DMAC1 transfer completion 6	Level/edge	COMMON	COMMON	×	✓	×
32	32		68		DMAC	CH1	DMAC1_DMAEND_7	DMAC1 transfer completion 7	Level/edge	COMMON	COMMON	×	✓	×
33	33		69		DMAC	CH1	DMAC1_DMAEND_8	DMAC1 transfer completion 8	Level/edge	COMMON	COMMON	×	✓	×
34	34		70		DMAC	CH1	DMAC1_DMAEND_9	DMAC1 transfer completion 9	Level/edge	COMMON	COMMON	×	✓	×
35	35		71		DMAC	CH1	DMAC1_DMAEND_10	DMAC1 transfer completion 10	Level/edge	COMMON	COMMON	×	✓	×
36	36		72		DMAC	CH1	DMAC1_DMAEND_11	DMAC1 transfer completion 11	Level/edge	COMMON	COMMON	×	✓	×
37	37		73		DMAC	CH1	DMAC1_DMAEND_12	DMAC1 transfer completion 12	Level/edge	COMMON	COMMON	×	✓	×
38	38		74		DMAC	CH1	DMAC1_DMAEND_13	DMAC1 transfer completion 13	Level/edge	COMMON	COMMON	×	✓	×
39	39		75		DMAC	CH1	DMAC1_DMAEND_14	DMAC1 transfer completion 14	Level/edge	COMMON	COMMON	×	✓	×
40	40		76		DMAC	CH1	DMAC1_DMAEND_15	DMAC1 transfer completion 15	Level/edge	COMMON	COMMON	×	✓	×
41	41		77		DMAC	CH2	DMAC2_DMAEND_0	DMAC2 transfer completion 0	Level/edge	COMMON	COMMON	×	✓	×
42	42		78		DMAC	CH2	DMAC2_DMAEND_1	DMAC2 transfer completion 1	Level/edge	COMMON	COMMON	×	✓	×
43	43		79		DMAC	CH2	DMAC2_DMAEND_2	DMAC2 transfer completion 2	Level/edge	COMMON	COMMON	×	✓	×
44	44		80		DMAC	CH2	DMAC2_DMAEND_3	DMAC2 transfer completion 3	Level/edge	COMMON	COMMON	×	✓	×
45	45		81		DMAC	CH2	DMAC2_DMAEND_4	DMAC2 transfer completion 4	Level/edge	COMMON	COMMON	×	✓	×
46	46		82		DMAC	CH2	DMAC2_DMAEND_5	DMAC2 transfer completion 5	Level/edge	COMMON	COMMON	×	✓	×
47	47		83		DMAC	CH2	DMAC2_DMAEND_6	DMAC2 transfer completion 6	Level/edge	COMMON	COMMON	×	✓	×
48	48		84		DMAC	CH2	DMAC2_DMAEND_7	DMAC2 transfer completion 7	Level/edge	COMMON	COMMON	×	✓	×
49	49		85		DMAC	CH2	DMAC2_DMAEND_8	DMAC2 transfer completion 8	Level/edge	COMMON	COMMON	×	✓	×
50	50		86		DMAC	CH2	DMAC2_DMAEND_9	DMAC2 transfer completion 9	Level/edge	COMMON	COMMON	×	✓	×
51	51		87		DMAC	CH2	DMAC2_DMAEND_10	DMAC2 transfer completion 10	Level/edge	COMMON	COMMON	×	✓	×

Table 4.6-24 List of Input Events (3/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
52	52		88		DMAC	CH2	DMAC2_DMAEND_11	DMAC2 transfer completion 11	Level/edge	COMMON	COMMON	×	✓	×
53	53		89		DMAC	CH2	DMAC2_DMAEND_12	DMAC2 transfer completion 12	Level/edge	COMMON	COMMON	×	✓	×
54	54		90		DMAC	CH2	DMAC2_DMAEND_13	DMAC2 transfer completion 13	Level/edge	COMMON	COMMON	×	✓	×
55	55		91		DMAC	CH2	DMAC2_DMAEND_14	DMAC2 transfer completion 14	Level/edge	COMMON	COMMON	×	✓	×
56	56		92		DMAC	CH2	DMAC2_DMAEND_15	DMAC2 transfer completion 15	Level/edge	COMMON	COMMON	×	✓	×
57	57		93		DMAC	CH3	DMAC3_DMAEND_0	DMAC3 transfer completion 0	Level/edge	COMMON	COMMON	×	✓	×
58	58		94		DMAC	CH3	DMAC3_DMAEND_1	DMAC3 transfer completion 1	Level/edge	COMMON	COMMON	×	✓	×
59	59		95		DMAC	CH3	DMAC3_DMAEND_2	DMAC3 transfer completion 2	Level/edge	COMMON	COMMON	×	✓	×
60	60		96		DMAC	CH3	DMAC3_DMAEND_3	DMAC3 transfer completion 3	Level/edge	COMMON	COMMON	×	✓	×
61	61		97		DMAC	CH3	DMAC3_DMAEND_4	DMAC3 transfer completion 4	Level/edge	COMMON	COMMON	×	✓	×
62	62		98		DMAC	CH3	DMAC3_DMAEND_5	DMAC3 transfer completion 5	Level/edge	COMMON	COMMON	×	✓	×
63	63		99		DMAC	CH3	DMAC3_DMAEND_6	DMAC3 transfer completion 6	Level/edge	COMMON	COMMON	×	✓	×
64	64		100		DMAC	CH3	DMAC3_DMAEND_7	DMAC3 transfer completion 7	Level/edge	COMMON	COMMON	×	✓	×
65	65		101		DMAC	CH3	DMAC3_DMAEND_8	DMAC3 transfer completion 8	Level/edge	COMMON	COMMON	×	✓	×
66	66		102		DMAC	CH3	DMAC3_DMAEND_9	DMAC3 transfer completion 9	Level/edge	COMMON	COMMON	×	✓	×
67	67		103		DMAC	CH3	DMAC3_DMAEND_10	DMAC3 transfer completion 10	Level/edge	COMMON	COMMON	×	✓	×
68	68		104		DMAC	CH3	DMAC3_DMAEND_11	DMAC3 transfer completion 11	Level/edge	COMMON	COMMON	×	✓	×
69	69		105		DMAC	CH3	DMAC3_DMAEND_12	DMAC3 transfer completion 12	Level/edge	COMMON	COMMON	×	✓	×
70	70		106		DMAC	CH3	DMAC3_DMAEND_13	DMAC3 transfer completion 13	Level/edge	COMMON	COMMON	×	✓	×
71	71		107		DMAC	CH3	DMAC3_DMAEND_14	DMAC3 transfer completion 14	Level/edge	COMMON	COMMON	×	✓	×
72	72		108		DMAC	CH3	DMAC3_DMAEND_15	DMAC3 transfer completion 15	Level/edge	COMMON	COMMON	×	✓	×
73	73		109		DMAC	CH4	DMAC4_DMAEND_0	DMAC4 transfer completion 0	Level/edge	COMMON	COMMON	×	✓	×
74	74		110		DMAC	CH4	DMAC4_DMAEND_1	DMAC4 transfer completion 1	Level/edge	COMMON	COMMON	×	✓	×
75	75		111		DMAC	CH4	DMAC4_DMAEND_2	DMAC4 transfer completion 2	Level/edge	COMMON	COMMON	×	✓	×
76	76		112		DMAC	CH4	DMAC4_DMAEND_3	DMAC4 transfer completion 3	Level/edge	COMMON	COMMON	×	✓	×
77	77		113		DMAC	CH4	DMAC4_DMAEND_4	DMAC4 transfer completion 4	Level/edge	COMMON	COMMON	×	✓	×
78	78		114		DMAC	CH4	DMAC4_DMAEND_5	DMAC4 transfer completion 5	Level/edge	COMMON	COMMON	×	✓	×
79	79		115		DMAC	CH4	DMAC4_DMAEND_6	DMAC4 transfer completion 6	Level/edge	COMMON	COMMON	×	✓	×
80	80		116		DMAC	CH4	DMAC4_DMAEND_7	DMAC4 transfer completion 7	Level/edge	COMMON	COMMON	×	✓	×
81	81		117		DMAC	CH4	DMAC4_DMAEND_8	DMAC4 transfer completion 8	Level/edge	COMMON	COMMON	×	✓	×
82	82		118		DMAC	CH4	DMAC4_DMAEND_9	DMAC4 transfer completion 9	Level/edge	COMMON	COMMON	×	✓	×

Table 4.6-24 List of Input Events (4/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
83	83		119		DMAC	CH4	DMAC4_DMAEND_10	DMAC4 transfer completion 10	Level/edge	COMMON	COMMON	×	✓	×
84	84		120		DMAC	CH4	DMAC4_DMAEND_11	DMAC4 transfer completion 11	Level/edge	COMMON	COMMON	×	✓	×
85	85		121		DMAC	CH4	DMAC4_DMAEND_12	DMAC4 transfer completion 12	Level/edge	COMMON	COMMON	×	✓	×
86	86		122		DMAC	CH4	DMAC4_DMAEND_13	DMAC4 transfer completion 13	Level/edge	COMMON	COMMON	×	✓	×
87	87		123		DMAC	CH4	DMAC4_DMAEND_14	DMAC4 transfer completion 14	Level/edge	COMMON	COMMON	×	✓	×
88	88		124		DMAC	CH4	DMAC4_DMAEND_15	DMAC4 transfer completion 15	Level/edge	COMMON	COMMON	×	✓	×
89	89		125		DMAC	CH0	DMAC0_DMAEND_0	DMAC0 transfer completion 0	Level/edge	COMMON	COMMON	×	✓	×
90	90		126		DMAC	CH0	DMAC0_DMAEND_1	DMAC0 transfer completion 1	Level/edge	COMMON	COMMON	×	✓	×
91	91		127		DMAC	CH0	DMAC0_DMAEND_2	DMAC0 transfer completion 2	Level/edge	COMMON	COMMON	×	✓	×
92	92		128		DMAC	CH0	DMAC0_DMAEND_3	DMAC0 transfer completion 3	Level/edge	COMMON	COMMON	×	✓	×
93	93		129		DMAC	CH0	DMAC0_DMAEND_4	DMAC0 transfer completion 4	Level/edge	COMMON	COMMON	×	✓	×
94	94		130		DMAC	CH0	DMAC0_DMAEND_5	DMAC0 transfer completion 5	Level/edge	COMMON	COMMON	×	✓	×
95	95		131		DMAC	CH0	DMAC0_DMAEND_6	DMAC0 transfer completion 6	Level/edge	COMMON	COMMON	×	✓	×
96	96		132		DMAC	CH0	DMAC0_DMAEND_7	DMAC0 transfer completion 7	Level/edge	COMMON	COMMON	×	✓	×
97	97		133		DMAC	CH0	DMAC0_DMAEND_8	DMAC0 transfer completion 8	Level/edge	COMMON	COMMON	×	✓	×
98	98		134		DMAC	CH0	DMAC0_DMAEND_9	DMAC0 transfer completion 9	Level/edge	COMMON	COMMON	×	✓	×
99	99		135		DMAC	CH0	DMAC0_DMAEND_10	DMAC0 transfer completion 10	Level/edge	COMMON	COMMON	×	✓	×
100	100		136		DMAC	CH0	DMAC0_DMAEND_11	DMAC0 transfer completion 11	Level/edge	COMMON	COMMON	×	✓	×
101	101		137		DMAC	CH0	DMAC0_DMAEND_12	DMAC0 transfer completion 12	Level/edge	COMMON	COMMON	×	✓	×
102	102		138		DMAC	CH0	DMAC0_DMAEND_13	DMAC0 transfer completion 13	Level/edge	COMMON	COMMON	×	✓	×
103	103		139		DMAC	CH0	DMAC0_DMAEND_14	DMAC0 transfer completion 14	Level/edge	COMMON	COMMON	×	✓	×
104	104		140		DMAC	CH0	DMAC0_DMAEND_15	DMAC0 transfer completion 15	Level/edge	COMMON	COMMON	×	✓	×
105	105				RSPI	CH0	RSPI_CH0_sp_idint_n	Idle interrupt (RSPI idle)	Level	COMMON	COMMON	×	×	×
106	106				RSPI	CH0	RSPI_CH0_sp_errint_n	Error interrupt (mode fault, overrun, parity error)	Level	COMMON	COMMON	×	×	×
107	107				RSPI	CH0	RSPI_CH0_sp_ceintpls_n	Communications end interrupt	Edge	COMMON	COMMON	×	×	×
108	108				RSPI	CH1	RSPI_CH1_sp_idint_n	Idle interrupt (RSPI idle)	Level	COMMON	COMMON	×	×	×
109	109				RSPI	CH1	RSPI_CH1_sp_errint_n	Error interrupt (mode fault, overrun, parity error)	Level	COMMON	COMMON	×	×	×
110	110				RSPI	CH1	RSPI_CH1_sp_ceintpls_n	Communications end interrupt	Edge	COMMON	COMMON	×	×	×
111	111				RSPI	CH2	RSPI_CH2_sp_idint_n	Idle interrupt (RSPI idle)	Level	COMMON	COMMON	×	×	×
112	112				RSPI	CH2	RSPI_CH2_sp_errint_n	Error interrupt (mode fault, overrun, parity error)	Level	COMMON	COMMON	×	×	×
113	113				RSPI	CH2	RSPI_CH2_sp_ceintpls_n	Communications end interrupt	Edge	COMMON	COMMON	×	×	×

Table 4.6-24 List of Input Events (5/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
114	114				RSCI	CH0	RSCI_CH0_INT_sc_eri_n	Error interrupt, bus contention detection interrupt	Level	COMMON	COMMON	×	×	×
115	115	146			RSCI	CH0	RSCI_CH0_INT_sc_rxi_n	Simplified I2C: reception end interrupt, others: reception data full interrupt	Edge	COMMON	COMMON	✓	×	×
116	116	147			RSCI	CH0	RSCI_CH0_INT_sc_txi_n	Simplified I2C, smart card interface: transmission end interrupt, others: transmission data empty interrupt, break field transmission end interrupt	Edge	COMMON	COMMON	✓	×	×
117	117				RSCI	CH0	RSCI_CH0_INT_sc_tei_n	Simplified I2C: completion of generating start, re-start, and stop conditions (STI interrupt), others: transmission end interrupt	Level	COMMON	COMMON	×	×	×
118	118	148			RSCI	CH0	RSCI_CH0_INT_sc_aed_n	Active edge detection interrupt	Edge	COMMON	COMMON	✓	×	×
119	119				RSCI	CH0	RSCI_CH0_INT_sc_bfd_n	Break field detection interrupt	Level	COMMON	COMMON	×	×	×
120	120				RSCI	CH1	RSCI_CH1_INT_sc_eri_n	Error interrupt, bus contention detection interrupt	Level	COMMON	COMMON	×	×	×
121	121	149			RSCI	CH1	RSCI_CH1_INT_sc_rxi_n	Simplified I2C: reception end interrupt, others: reception data full interrupt	Edge	COMMON	COMMON	✓	×	×
122	122	150			RSCI	CH1	RSCI_CH1_INT_sc_txi_n	Simplified I2C, smart card interface: transmission end interrupt, others: transmission data empty interrupt, break field transmission end interrupt	Edge	COMMON	COMMON	✓	×	×
123	123				RSCI	CH1	RSCI_CH1_INT_sc_tei_n	Simplified I2C: completion of generating start, re-start, and stop conditions (STI interrupt), others: transmission end interrupt	Level	COMMON	COMMON	×	×	×
124	124	151			RSCI	CH1	RSCI_CH1_INT_sc_aed_n	Active edge detection interrupt	Edge	COMMON	COMMON	✓	×	×
125	125				RSCI	CH1	RSCI_CH1_INT_sc_bfd_n	Break field detection interrupt	Level	COMMON	COMMON	×	×	×
126	126				RSCI	CH2	RSCI_CH2_INT_sc_eri_n	Error interrupt, bus contention detection interrupt	Level	COMMON	COMMON	×	×	×
127	127	152			RSCI	CH2	RSCI_CH2_INT_sc_rxi_n	Simplified I2C: reception end interrupt, others: reception data full interrupt	Edge	COMMON	COMMON	✓	×	×
128	128	153			RSCI	CH2	RSCI_CH2_INT_sc_txi_n	Simplified I2C, smart card interface: transmission end interrupt, others: transmission data empty interrupt, break field transmission end interrupt	Edge	COMMON	COMMON	✓	×	×
129	129				RSCI	CH2	RSCI_CH2_INT_sc_tei_n	Simplified I2C: completion of generating start, re-start, and stop conditions (STI interrupt), others: transmission end interrupt	Level	COMMON	COMMON	×	×	×
130	130	154			RSCI	CH2	RSCI_CH2_INT_sc_aed_n	Active edge detection interrupt	Edge	COMMON	COMMON	✓	×	×
131	131				RSCI	CH2	RSCI_CH2_INT_sc_bfd_n	Break field detection interrupt	Level	COMMON	COMMON	×	×	×
132	132				RSCI	CH3	RSCI_CH3_INT_sc_eri_n	Error interrupt, bus contention detection interrupt	Level	COMMON	COMMON	×	×	×
133	133	155			RSCI	CH3	RSCI_CH3_INT_sc_rxi_n	Simplified I2C: reception end interrupt, others: reception data full interrupt	Edge	COMMON	COMMON	✓	×	×

Table 4.6-24 List of Input Events (6/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
134	134	156			RSCI	CH3	RSCI_CH3_INT_sc_txi_n	Simplified I2C, smart card interface: transmission end interrupt, others: transmission data empty interrupt, break field transmission end interrupt	Edge	COMMON	COMMON	✓	×	×
135	135				RSCI	CH3	RSCI_CH3_INT_sc_tei_n	Simplified I2C: completion of generating start, re-start, and stop conditions (STI interrupt), others: transmission end interrupt	Level	COMMON	COMMON	×	×	×
136	136	157			RSCI	CH3	RSCI_CH3_INT_sc_aed_n	Active edge detection interrupt	Edge	COMMON	COMMON	✓	×	×
137	137				RSCI	CH3	RSCI_CH3_INT_sc_bfd_n	Break field detection interrupt	Level	COMMON	COMMON	×	×	×
138	138				RSCI	CH4	RSCI_CH4_INT_sc_eri_n	Error interrupt, bus contention detection interrupt	Level	COMMON	COMMON	×	×	×
139	139	158			RSCI	CH4	RSCI_CH4_INT_sc_rxi_n	Simplified I2C: reception end interrupt, others: reception data full interrupt	Edge	COMMON	COMMON	✓	×	×
140	140	159			RSCI	CH4	RSCI_CH4_INT_sc_txi_n	Simplified I2C, smart card interface: transmission end interrupt, others: transmission data empty interrupt, break field transmission end interrupt	Edge	COMMON	COMMON	✓	×	×
141	141				RSCI	CH4	RSCI_CH4_INT_sc_tei_n	Simplified I2C: completion of generating start, re-start, and stop conditions (STI interrupt), others: transmission end interrupt	Level	COMMON	COMMON	×	×	×
142	142	160			RSCI	CH4	RSCI_CH4_INT_sc_aed_n	Active edge detection interrupt	Edge	COMMON	COMMON	✓	×	×
143	143				RSCI	CH4	RSCI_CH4_INT_sc_bfd_n	Break field detection interrupt	Level	COMMON	COMMON	×	×	×
144	144				RSCI	CH5	RSCI_CH5_INT_sc_eri_n	Error interrupt, bus contention detection interrupt	Level	COMMON	COMMON	×	×	×
145	145	161			RSCI	CH5	RSCI_CH5_INT_sc_rxi_n	Simplified I2C: reception end interrupt, others: reception data full interrupt	Edge	COMMON	COMMON	✓	×	×
146	146	162			RSCI	CH5	RSCI_CH5_INT_sc_txi_n	Simplified I2C, smart card interface: transmission end interrupt, others: transmission data empty interrupt, break field transmission end interrupt	Edge	COMMON	COMMON	✓	×	×
147	147				RSCI	CH5	RSCI_CH5_INT_sc_tei_n	Simplified I2C: completion of generating start, re-start, and stop conditions (STI interrupt), others: transmission end interrupt	Level	COMMON	COMMON	×	×	×
148	148	163			RSCI	CH5	RSCI_CH5_INT_sc_aed_n	Active edge detection interrupt	Edge	COMMON	COMMON	✓	×	×
149	149				RSCI	CH5	RSCI_CH5_INT_sc_bfd_n	Break field detection interrupt	Level	COMMON	COMMON	×	×	×
150	150				RSCI	CH6	RSCI_CH6_INT_sc_eri_n	Error interrupt, bus contention detection interrupt	Level	COMMON	COMMON	×	×	×
151	151	164			RSCI	CH6	RSCI_CH6_INT_sc_rxi_n	Simplified I2C: reception end interrupt, others: reception data full interrupt	Edge	COMMON	COMMON	✓	×	×
152	152	165			RSCI	CH6	RSCI_CH6_INT_sc_txi_n	Simplified I2C, smart card interface: transmission end interrupt, others: transmission data empty interrupt, break field transmission end interrupt	Edge	COMMON	COMMON	✓	×	×

Table 4.6-24 List of Input Events (7/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
153	153				RSCI	CH6	RSCI_CH6_INT_sc_tei_n	Simplified I2C: completion of generating start, re-start, and stop conditions (STI interrupt), others: transmission end interrupt	Level	COMMON	COMMON	×	×	×
154	154	166			RSCI	CH6	RSCI_CH6_INT_sc_aed_n	Active edge detection interrupt	Edge	COMMON	COMMON	✓	×	×
155	155				RSCI	CH6	RSCI_CH6_INT_sc_bfd_n	Break field detection interrupt	Level	COMMON	COMMON	×	×	×
156	156				RSCI	CH7	RSCI_CH7_INT_sc_eri_n	Error interrupt, bus contention detection interrupt	Level	COMMON	COMMON	×	×	×
157	157	167			RSCI	CH7	RSCI_CH7_INT_sc_rxi_n	Simplified I2C: reception end interrupt, others: reception data full interrupt	Edge	COMMON	COMMON	✓	×	×
158	158	168			RSCI	CH7	RSCI_CH7_INT_sc_txi_n	Simplified I2C, smart card interface: transmission end interrupt, others: transmission data empty interrupt, break field transmission end interrupt	Edge	COMMON	COMMON	✓	×	×
159	159				RSCI	CH7	RSCI_CH7_INT_sc_tei_n	Simplified I2C: completion of generating start, re-start, and stop conditions (STI interrupt), others: transmission end interrupt	Level	COMMON	COMMON	×	×	×
160	160	169			RSCI	CH7	RSCI_CH7_INT_sc_aed_n	Active edge detection interrupt	Edge	COMMON	COMMON	✓	×	×
161	161				RSCI	CH7	RSCI_CH7_INT_sc_bfd_n	Break field detection interrupt	Level	COMMON	COMMON	×	×	×
162	162				RSCI	CH8	RSCI_CH8_INT_sc_eri_n	Error interrupt, bus contention detection interrupt	Level	COMMON	COMMON	×	×	×
163	163	170			RSCI	CH8	RSCI_CH8_INT_sc_rxi_n	Simplified I2C: reception end interrupt, others: reception data full interrupt	Edge	COMMON	COMMON	✓	×	×
164	164	171			RSCI	CH8	RSCI_CH8_INT_sc_txi_n	Simplified I2C, smart card interface: transmission end interrupt, others: transmission data empty interrupt, break field transmission end interrupt	Edge	COMMON	COMMON	✓	×	×
165	165				RSCI	CH8	RSCI_CH8_INT_sc_tei_n	Simplified I2C: completion of generating start, re-start, and stop conditions (STI interrupt), others: transmission end interrupt	Level	COMMON	COMMON	×	×	×
166	166	172			RSCI	CH8	RSCI_CH8_INT_sc_aed_n	Active edge detection interrupt	Edge	COMMON	COMMON	✓	×	×
167	167				RSCI	CH8	RSCI_CH8_INT_sc_bfd_n	Break field detection interrupt	Level	COMMON	COMMON	×	×	×
168	168				RSCI	CH9	RSCI_CH9_INT_sc_eri_n	Error interrupt, bus contention detection interrupt	Level	COMMON	COMMON	×	×	×
169	169	173			RSCI	CH9	RSCI_CH9_INT_sc_rxi_n	Simplified I2C: reception end interrupt, others: reception data full interrupt	Edge	COMMON	COMMON	✓	×	×
170	170	174			RSCI	CH9	RSCI_CH9_INT_sc_txi_n	Simplified I2C, smart card interface: transmission end interrupt, others: transmission data empty interrupt, break field transmission end interrupt	Edge	COMMON	COMMON	✓	×	×
171	171				RSCI	CH9	RSCI_CH9_INT_sc_tei_n	Simplified I2C: completion of generating start, re-start, and stop conditions (STI interrupt), others: transmission end interrupt	Level	COMMON	COMMON	×	×	×

Table 4.6-24 List of Input Events (8/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
172	172	175			RSCI	CH9	RSCI_CH9_INT_sc_aed_n	Active edge detection interrupt	Edge	COMMON	COMMON	✓	×	×
173	173				RSCI	CH9	RSCI_CH9_INT_sc_bfd_n	Break field detection interrupt	Level	COMMON	COMMON	×	×	×
174	174				RIIC	CH0	RIIC_CH0_tei_n	Transmission end interrupt request	Level	COMMON	COMMON	×	×	×
175	175				RIIC	CH0	RIIC_CH0_naki_n	No acknowledge bit interrupt request	Level	COMMON	COMMON	×	×	×
176	176				RIIC	CH0	RIIC_CH0_spi_n	STOP found interrupt request	Level	COMMON	COMMON	×	×	×
177	177				RIIC	CH0	RIIC_CH0_sti_n	START found interrupt request	Level	COMMON	COMMON	×	×	×
178	178				RIIC	CH0	RIIC_CH0_ali_n	Arbitration lost interrupt request	Level	COMMON	COMMON	×	×	×
179	179				RIIC	CH0	RIIC_CH0_tmoi_n	Timer timeout interrupt request	Level	COMMON	COMMON	×	×	×
180	180				RIIC	CH1	RIIC_CH1_tei_n	Transmission end interrupt request	Level	COMMON	COMMON	×	×	×
181	181				RIIC	CH1	RIIC_CH1_naki_n	No acknowledge bit interrupt request	Level	COMMON	COMMON	×	×	×
182	182				RIIC	CH1	RIIC_CH1_spi_n	STOP found interrupt request	Level	COMMON	COMMON	×	×	×
183	183				RIIC	CH1	RIIC_CH1_sti_n	START found interrupt request	Level	COMMON	COMMON	×	×	×
184	184				RIIC	CH1	RIIC_CH1_ali_n	Arbitration lost interrupt request	Level	COMMON	COMMON	×	×	×
185	185				RIIC	CH1	RIIC_CH1_tmoi_n	Timer timeout interrupt request	Level	COMMON	COMMON	×	×	×
186	186				RIIC	CH2	RIIC_CH2_tei_n	Transmission end interrupt request	Level	COMMON	COMMON	×	×	×
187	187				RIIC	CH2	RIIC_CH2_naki_n	No acknowledge bit interrupt request	Level	COMMON	COMMON	×	×	×
188	188				RIIC	CH2	RIIC_CH2_spi_n	STOP found interrupt request	Level	COMMON	COMMON	×	×	×
189	189				RIIC	CH2	RIIC_CH2_sti_n	START found interrupt request	Level	COMMON	COMMON	×	×	×
190	190				RIIC	CH2	RIIC_CH2_ali_n	Arbitration lost interrupt request	Level	COMMON	COMMON	×	×	×
191	191				RIIC	CH2	RIIC_CH2_tmoi_n	Timer timeout interrupt request	Level	COMMON	COMMON	×	×	×
192	192				RIIC	CH3	RIIC_CH3_tei_n	Transmission end interrupt request	Level	COMMON	COMMON	×	×	×
193	193				RIIC	CH3	RIIC_CH3_naki_n	No acknowledge bit interrupt request	Level	COMMON	COMMON	×	×	×
194	194				RIIC	CH3	RIIC_CH3_spi_n	STOP found interrupt request	Level	COMMON	COMMON	×	×	×
195	195				RIIC	CH3	RIIC_CH3_sti_n	START found interrupt request	Level	COMMON	COMMON	×	×	×
196	196				RIIC	CH3	RIIC_CH3_ali_n	Arbitration lost interrupt request	Level	COMMON	COMMON	×	×	×
197	197				RIIC	CH3	RIIC_CH3_tmoi_n	Timer timeout interrupt request	Level	COMMON	COMMON	×	×	×
198	198				RIIC	CH4	RIIC_CH4_tei_n	Transmission end interrupt request	Level	COMMON	COMMON	×	×	×
199	199				RIIC	CH4	RIIC_CH4_naki_n	No acknowledge bit interrupt request	Level	COMMON	COMMON	×	×	×
200	200				RIIC	CH4	RIIC_CH4_spi_n	STOP found interrupt request	Level	COMMON	COMMON	×	×	×
201	201				RIIC	CH4	RIIC_CH4_sti_n	START found interrupt request	Level	COMMON	COMMON	×	×	×
202	202				RIIC	CH4	RIIC_CH4_ali_n	Arbitration lost interrupt request	Level	COMMON	COMMON	×	×	×

Table 4.6-24 List of Input Events (9/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
203	203				RIIC	CH4	RIIC_CH4_tmoi_n	Timer timeout interrupt request	Level	COM MON	COM MON	×	×	×
204	204				RIIC	CH5	RIIC_CH5_tei_n	Transmission end interrupt request	Level	COM MON	COM MON	×	×	×
205	205				RIIC	CH5	RIIC_CH5_naki_n	No acknowledge bit interrupt request	Level	COM MON	COM MON	×	×	×
206	206				RIIC	CH5	RIIC_CH5_spi_n	STOP found interrupt request	Level	COM MON	COM MON	×	×	×
207	207				RIIC	CH5	RIIC_CH5_sti_n	START found interrupt request	Level	COM MON	COM MON	×	×	×
208	208				RIIC	CH5	RIIC_CH5_ali_n	Arbitration lost interrupt request	Level	COM MON	COM MON	×	×	×
209	209				RIIC	CH5	RIIC_CH5_tmoi_n	Timer timeout interrupt request	Level	COM MON	COM MON	×	×	×
210	210				RIIC	CH6	RIIC_CH6_tei_n	Transmission end interrupt request	Level	COM MON	COM MON	×	×	×
211	211				RIIC	CH6	RIIC_CH6_naki_n	No acknowledge bit interrupt request	Level	COM MON	COM MON	×	×	×
212	212				RIIC	CH6	RIIC_CH6_spi_n	STOP found interrupt request	Level	COM MON	COM MON	×	×	×
213	213				RIIC	CH6	RIIC_CH6_sti_n	START found interrupt request	Level	COM MON	COM MON	×	×	×
214	214				RIIC	CH6	RIIC_CH6_ali_n	Arbitration lost interrupt request	Level	COM MON	COM MON	×	×	×
215	215				RIIC	CH6	RIIC_CH6_tmoi_n	Timer timeout interrupt request	Level	COM MON	COM MON	×	×	×
216	216				RIIC	CH7	RIIC_CH7_tei_n	Transmission end interrupt request	Level	COM MON	COM MON	×	×	×
217	217				RIIC	CH7	RIIC_CH7_naki_n	No acknowledge bit interrupt request	Level	COM MON	COM MON	×	×	×
218	218				RIIC	CH7	RIIC_CH7_spi_n	STOP found interrupt request	Level	COM MON	COM MON	×	×	×
219	219				RIIC	CH7	RIIC_CH7_sti_n	START found interrupt request	Level	COM MON	COM MON	×	×	×
220	220				RIIC	CH7	RIIC_CH7_ali_n	Arbitration lost interrupt request	Level	COM MON	COM MON	×	×	×
221	221				RIIC	CH7	RIIC_CH7_tmoi_n	Timer timeout interrupt request	Level	COM MON	COM MON	×	×	×
222	222				RIIC	CH8	RIIC_CH8_tei_n	Transmission end interrupt request	Level	COM MON	COM MON	×	×	×
223	223				RIIC	CH8	RIIC_CH8_naki_n	No acknowledge bit interrupt request	Level	COM MON	COM MON	×	×	×
224	224				RIIC	CH8	RIIC_CH8_spi_n	STOP found interrupt request	Level	COM MON	COM MON	×	×	×
225	225				RIIC	CH8	RIIC_CH8_sti_n	START found interrupt request	Level	COM MON	COM MON	×	×	×
226	226				RIIC	CH8	RIIC_CH8_ali_n	Arbitration lost interrupt request	Level	COM MON	COM MON	×	×	×
227	227				RIIC	CH8	RIIC_CH8_tmoi_n	Timer timeout interrupt request	Level	COM MON	COM MON	×	×	×
228	228				xSPI	—	int_spi_pulse	Interrupt pulse signal by factors excluding errors	Edge	COM MON	COM MON	×	×	×
229	229				xSPI	—	int_spi_err_pulse	Interrupt pulse signal by error factors	Edge	COM MON	COM MON	×	×	×
230	230				—	—	Reserved	Reserved	—	—	—	—	—	—
231	231				—	—	Reserved	Reserved	—	—	—	—	—	—
232	232	194			PDM	UNIT0	INT_PDM_DAT0	Data reception interrupt CH0	Level	COM MON	COM MON	✓	×	×
233	233	195			PDM	UNIT0	INT_PDM_DAT1	Data reception interrupt CH1 (*2)	Level	COM MON	COM MON	✓	×	×
234	234	196			PDM	UNIT0	INT_PDM_DAT2	Data reception interrupt CH2 (*3)	Level	COM MON	COM MON	✓	×	×

Table 4.6-24 List of Input Events (10/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
235	235				PDM	UNIT0	INT_PDM_SDET	Sound detection interrupt (Shared between channels)	Level	COMMON	COMMON	×	×	×
236	236				PDM	UNIT0	INT_PDM_ERR0	Error detection interrupt CH0	Level	COMMON	COMMON	×	×	×
237	237				PDM	UNIT0	INT_PDM_ERR1	Error detection interrupt CH1 (*2)	Level	COMMON	COMMON	×	×	×
238	238				PDM	UNIT0	INT_PDM_ERR2	Error detection interrupt CH2 (*3)	Level	COMMON	COMMON	×	×	×
239	239	197			PDM	UNIT1	INT_PDM_DAT0	Data reception interrupt CH0	Level	COMMON	COMMON	✓	×	×
240	240	198			PDM	UNIT1	INT_PDM_DAT1	Data reception interrupt CH1 (*2)	Level	COMMON	COMMON	✓	×	×
241	241	199			PDM	UNIT1	INT_PDM_DAT2	Data reception interrupt CH2 (*3)	Level	COMMON	COMMON	✓	×	×
242	242				PDM	UNIT1	INT_PDM_SDET	Sound detection interrupt (Shared between channels)	Level	COMMON	COMMON	×	×	×
243	243				PDM	UNIT1	INT_PDM_ERR0	Error detection interrupt CH0	Level	COMMON	COMMON	×	×	×
244	244				PDM	UNIT1	INT_PDM_ERR1	Error detection interrupt CH1 (*2)	Level	COMMON	COMMON	×	×	×
245	245				PDM	UNIT1	INT_PDM_ERR2	Error detection interrupt CH2 (*3)	Level	COMMON	COMMON	×	×	×
246	246				ADC	—	ada_compai_n	Window compare match(windowA)	Level	COMMON	COMMON	×	×	×
247	247				ADC	—	ada_compbi_n	Window compare match(windowB)	Level	COMMON	COMMON	×	×	×
248	248	205			—	—	Reserved	Reserved	—	—	—	—	—	—
249	249				—	—	Reserved	Reserved	—	—	—	—	—	—
250	250	206			TSU	—	S12TSUADI1	A/D conversion-end	Edge	COMMON	COMMON	✓	×	×
251	251				TSU	—	S12TSUADCMPI1	Window compare match	Level	COMMON	COMMON	×	×	×
252	252	5			—	—	Reserved	Reserved	—	—	—	—	—	—
253	253	6			—	—	Reserved	Reserved	—	—	—	—	—	—
254	254	7			—	—	Reserved	Reserved	—	—	—	—	—	—
255	255	8			—	—	Reserved	Reserved	—	—	—	—	—	—
256	256	9			—	—	Reserved	Reserved	—	—	—	—	—	—
257	257				—	—	Reserved	Reserved	—	—	—	—	—	—
258	258				—	—	Reserved	Reserved	—	—	—	—	—	—
259	259				—	—	Reserved	Reserved	—	—	—	—	—	—
260	260	10			—	—	Reserved	Reserved	—	—	—	—	—	—
261	261	11			—	—	Reserved	Reserved	—	—	—	—	—	—
262					ICU	SWINT	INT_CA55_0	Occurs by writing to the ICU_TOP register (for CPU Interrupt)	Edge	SYSTEM	—	×	×	×
263					ICU	SWINT	INT_CA55_1	Occurs by writing to the ICU_TOP register (for CPU Interrupt)	Edge	SYSTEM	—	×	×	×
264					ICU	SWINT	INT_CA55_2	Occurs by writing to the ICU_TOP register (for CPU Interrupt)	Edge	SYSTEM	—	×	×	×
265					ICU	SWINT	INT_CA55_3	Occurs by writing to the ICU_TOP register (for CPU Interrupt)	Edge	SYSTEM	—	×	×	×
266					ICU	ICUERR	ICU_Error_CA55	Signal with error interrupts combined into one in ICU_TOP	Level	SYSTEM	—	×	×	×
267					CA55	—	nVCPUMNTIRQ_0	Virtual CPU interface maintenance interrupt PPI output.	Level	SYSTEM	—	×	×	×

Table 4.6-24 List of Input Events (11/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
268					CA55	—	nVCPUMNTIRQ_1	Virtual CPU interface maintenance interrupt PPI output.	Level	SYSTEM	—	×	×	×
269					CA55	—	nVCPUMNTIRQ_2	Virtual CPU interface maintenance interrupt PPI output.	Level	SYSTEM	—	×	×	×
270					CA55	—	nVCPUMNTIRQ_3	Virtual CPU interface maintenance interrupt PPI output.	Level	SYSTEM	—	×	×	×
271					CA55	—	nCNTHPIRQ_0	Hypervisor physical timer event.	Level	SYSTEM	—	×	×	×
272					CA55	—	nCNTHPIRQ_1	Hypervisor physical timer event.	Level	SYSTEM	—	×	×	×
273					CA55	—	nCNTHPIRQ_2	Hypervisor physical timer event.	Level	SYSTEM	—	×	×	×
274					CA55	—	nCNTHPIRQ_3	Hypervisor physical timer event.	Level	SYSTEM	—	×	×	×
275					CA55	—	nCNTPNSIRQ_0	Non-secure physical timer event.	Level	SYSTEM	—	×	×	×
276					CA55	—	nCNTPNSIRQ_1	Non-secure physical timer event.	Level	SYSTEM	—	×	×	×
277					CA55	—	nCNTPNSIRQ_2	Non-secure physical timer event.	Level	SYSTEM	—	×	×	×
278					CA55	—	nCNTPNSIRQ_3	Non-secure physical timer event.	Level	SYSTEM	—	×	×	×
279					CA55	—	nCNTPSIRQ_0	Secure physical timer event.	Level	SYSTEM	—	×	×	×
280					CA55	—	nCNTPSIRQ_1	Secure physical timer event.	Level	SYSTEM	—	×	×	×
281					CA55	—	nCNTPSIRQ_2	Secure physical timer event.	Level	SYSTEM	—	×	×	×
282					CA55	—	nCNTPSIRQ_3	Secure physical timer event.	Level	SYSTEM	—	×	×	×
283					CA55	—	nCNTVIRQ_0	Virtual timer event.	Level	SYSTEM	—	×	×	×
284					CA55	—	nCNTVIRQ_1	Virtual timer event.	Level	SYSTEM	—	×	×	×
285					CA55	—	nCNTVIRQ_2	Virtual timer event.	Level	SYSTEM	—	×	×	×
286					CA55	—	nCNTVIRQ_3	Virtual timer event.	Level	SYSTEM	—	×	×	×
287					CA55	—	nCNTHVIRQ_0	Hypervisor virtual timer event.	Level	SYSTEM	—	×	×	×
288					CA55	—	nCNTHVIRQ_1	Hypervisor virtual timer event.	Level	SYSTEM	—	×	×	×
289					CA55	—	nCNTHVIRQ_2	Hypervisor virtual timer event.	Level	SYSTEM	—	×	×	×
290					CA55	—	nCNTHVIRQ_3	Hypervisor virtual timer event.	Level	SYSTEM	—	×	×	×
291					CA55	—	nCOMMIRQ_0	Communication channel receive or transmit interrupt request.	Level	SYSTEM	—	×	×	×
292					CA55	—	nCOMMIRQ_1	Communication channel receive or transmit interrupt request.	Level	SYSTEM	—	×	×	×
293					CA55	—	nCOMMIRQ_2	Communication channel receive or transmit interrupt request.	Level	SYSTEM	—	×	×	×
294					CA55	—	nCOMMIRQ_3	Communication channel receive or transmit interrupt request.	Level	SYSTEM	—	×	×	×
295					CA55	—	CTIIRQ_0	CTI interrupt.	Level	SYSTEM	—	×	×	×

Table 4.6-24 List of Input Events (12/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
296					CA55	—	CTIIRQ_1	CTI interrupt.	Level	SYSTEM	—	×	×	×
297					CA55	—	CTIIRQ_2	CTI interrupt.	Level	SYSTEM	—	×	×	×
298					CA55	—	CTIIRQ_3	CTI interrupt.	Level	SYSTEM	—	×	×	×
299					CA55	—	nCLUSTERPMUIRQ	Cluster PMU interrupt request.	Level	SYSTEM	—	×	×	×
300					CA55	—	nPMUIRQ_0	PMU interrupt request.	Level	SYSTEM	—	×	×	×
301					CA55	—	nPMUIRQ_1	PMU interrupt request.	Level	SYSTEM	—	×	×	×
302					CA55	—	nPMUIRQ_2	PMU interrupt request.	Level	SYSTEM	—	×	×	×
303					CA55	—	nPMUIRQ_3	PMU interrupt request.	Level	SYSTEM	—	×	×	×
304	277				DDR	CH0	—	—	—	SYSTEM	SYSTEM	×	×	×
305	278				DDR	CH0	—	—	—	SYSTEM	SYSTEM	×	×	×
306	279				—	—	Reserved	Reserved	—	—	—	—	—	—
307	280				—	—	Reserved	Reserved	—	—	—	—	—	—
308					MHU	—	msg_ch0_s	Secure message interrupt ch0	Level	SYSTEM	—	×	×	×
309					MHU	—	msg_ch1_s	Secure message interrupt ch1	Level	SYSTEM	—	×	×	×
310					MHU	—	msg_ch2_s	Secure Message interrupt ch2	Level	SYSTEM	—	×	×	×
311					MHU	—	msg_ch6_s	Secure message interrupt ch6	Level	SYSTEM	—	×	×	×
312					MHU	—	msg_ch7_s	Secure message interrupt ch7	Level	SYSTEM	—	×	×	×
313					MHU	—	msg_ch8_s	Secure message interrupt ch8	Level	SYSTEM	—	×	×	×
314					MHU	—	msg_ch12_s	Secure message interrupt ch12	Level	SYSTEM	—	×	×	×
315					MHU	—	msg_ch13_s	Secure message interrupt ch13	Level	SYSTEM	—	×	×	×
316					MHU	—	msg_ch14_s	Secure message interrupt ch14	Level	SYSTEM	—	×	×	×
317					MHU	—	msg_ch18_s	Secure message interrupt ch18	Level	SYSTEM	—	×	×	×
318					MHU	—	msg_ch19_s	Secure message interrupt ch19	Level	SYSTEM	—	×	×	×
319					MHU	—	msg_ch20_s	Secure message interrupt ch20	Level	SYSTEM	—	×	×	×
320					MHU	—	msg_ch24_s	Secure message interrupt ch24 #Reserved	Level	SYSTEM	—	×	×	×
321					MHU	—	msg_ch25_s	Secure message interrupt ch25 #Reserved	Level	SYSTEM	—	×	×	×
322					MHU	—	msg_ch26_s	Secure message interrupt ch26 #Reserved	Level	SYSTEM	—	×	×	×
323					MHU	—	msg_ch27_s	Secure message interrupt ch27 #Reserved	Level	SYSTEM	—	×	×	×
324					MHU	—	msg_ch30_s	Secure message interrupt ch30 #Reserved	Level	SYSTEM	—	×	×	×
325					MHU	—	msg_ch31_s	Secure message interrupt ch31 #Reserved	Level	SYSTEM	—	×	×	×
326					MHU	—	msg_ch32_s	Secure message interrupt ch32 #Reserved	Level	SYSTEM	—	×	×	×
327					MHU	—	msg_ch33_s	Secure message interrupt ch33	Level	SYSTEM	—	×	×	×

Table 4.6-24 List of Input Events (13/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
328					MHU	—	msg_ch36_s	Secure message interrupt ch36	Level	SYSTEM	—	×	×	×
329					MHU	—	msg_ch37_s	Secure message interrupt ch37	Level	SYSTEM	—	×	×	×
330					MHU	—	msg_ch38_s	Secure message interrupt ch38	Level	SYSTEM	—	×	×	×
331					MHU	—	msg_ch39_s	Secure message interrupt ch39	Level	SYSTEM	—	×	×	×
332					MHU	—	rsp_ch0_s	Secure response interrupt ch0	Level	SYSTEM	—	×	×	×
333					MHU	—	rsp_ch1_s	Secure response interrupt ch1	Level	SYSTEM	—	×	×	×
334					MHU	—	rsp_ch2_s	Secure response interrupt ch2	Level	SYSTEM	—	×	×	×
335					MHU	—	rsp_ch6_s	Secure response interrupt ch6 #Reserved	Level	SYSTEM	—	×	×	×
336					MHU	—	rsp_ch7_s	Secure response interrupt ch7 #Reserved	Level	SYSTEM	—	×	×	×
337					MHU	—	rsp_ch8_s	Secure response interrupt ch8	Level	SYSTEM	—	×	×	×
338					MHU	—	rsp_ch12_s	Secure response interrupt ch12	Level	SYSTEM	—	×	×	×
339					MHU	—	rsp_ch13_s	Secure response interrupt ch13	Level	SYSTEM	—	×	×	×
340					MHU	—	rsp_ch14_s	Secure response interrupt ch14	Level	SYSTEM	—	×	×	×
341					MHU	—	rsp_ch18_s	Secure response interrupt ch18 #Reserved	Level	SYSTEM	—	×	×	×
342					MHU	—	rsp_ch19_s	Secure response interrupt ch19 #Reserved	Level	SYSTEM	—	×	×	×
343					MHU	—	rsp_ch20_s	Secure response interrupt ch20	Level	SYSTEM	—	×	×	×
344					MHU	—	rsp_ch24_s	Secure response interrupt ch24	Level	SYSTEM	—	×	×	×
345					MHU	—	rsp_ch25_s	Secure response interrupt ch25	Level	SYSTEM	—	×	×	×
346					MHU	—	rsp_ch26_s	Secure response interrupt ch26	Level	SYSTEM	—	×	×	×
347					MHU	—	rsp_ch27_s	Secure response interrupt ch27 #Reserved	Level	SYSTEM	—	×	×	×
348					MHU	—	rsp_ch30_s	Secure response interrupt ch30 #Reserved	Level	SYSTEM	—	×	×	×
349					MHU	—	rsp_ch31_s	Secure response interrupt ch31	Level	SYSTEM	—	×	×	×
350					MHU	—	rsp_ch32_s	Secure response interrupt ch32	Level	SYSTEM	—	×	×	×
351					MHU	—	rsp_ch33_s	Secure response interrupt ch33	Level	SYSTEM	—	×	×	×
352					MHU	—	rsp_ch36_s	Secure response interrupt ch36	Level	SYSTEM	—	×	×	×
353					MHU	—	rsp_ch37_s	Secure response interrupt ch37 #Reserved	Level	SYSTEM	—	×	×	×
354					MHU	—	rsp_ch38_s	Secure response interrupt ch38 #Reserved	Level	SYSTEM	—	×	×	×
355					MHU	—	rsp_ch39_s	Secure response interrupt ch39	Level	SYSTEM	—	×	×	×
356					MHU	—	msg_ch0_ns	Non-secure message interrupt ch0	Level	SYSTEM	—	×	×	×
357					MHU	—	msg_ch1_ns	Non-secure message interrupt ch1	Level	SYSTEM	—	×	×	×
358					MHU	—	msg_ch2_ns	Non-secure message interrupt ch2	Level	SYSTEM	—	×	×	×

Table 4.6-24 List of Input Events (14/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
359					MHU	—	msg_ch6_ns	Non-secure message interrupt ch6	Level	SYSTEM	—	×	×	×
360					MHU	—	msg_ch7_ns	Non-secure message interrupt ch7	Level	SYSTEM	—	×	×	×
361					MHU	—	msg_ch8_ns	Non-secure message interrupt ch8	Level	SYSTEM	—	×	×	×
362					MHU	—	msg_ch12_ns	Non-secure message interrupt ch12	Level	SYSTEM	—	×	×	×
363					MHU	—	msg_ch13_ns	Non-secure message interrupt ch13	Level	SYSTEM	—	×	×	×
364					MHU	—	msg_ch14_ns	Non-secure message interrupt ch14	Level	SYSTEM	—	×	×	×
365					MHU	—	msg_ch18_ns	Non-secure message interrupt ch18	Level	SYSTEM	—	×	×	×
366					MHU	—	msg_ch19_ns	Non-secure message interrupt ch19	Level	SYSTEM	—	×	×	×
367					MHU	—	msg_ch20_ns	Non-secure message interrupt ch20	Level	SYSTEM	—	×	×	×
368					MHU	—	msg_ch24_ns	Non-secure message interrupt ch24 #Reserved	Level	SYSTEM	—	×	×	×
369					MHU	—	msg_ch25_ns	Non-secure message interrupt ch25 #Reserved	Level	SYSTEM	—	×	×	×
370					MHU	—	msg_ch26_ns	Non-secure message interrupt ch26 #Reserved	Level	SYSTEM	—	×	×	×
371					MHU	—	msg_ch27_ns	Non-secure message interrupt ch27 #Reserved	Level	SYSTEM	—	×	×	×
372					MHU	—	msg_ch30_ns	Non-secure message interrupt ch30 #Reserved	Level	SYSTEM	—	×	×	×
373					MHU	—	msg_ch31_ns	Non-secure message interrupt ch31 #Reserved	Level	SYSTEM	—	×	×	×
374					MHU	—	msg_ch32_ns	Non-secure message interrupt ch32 #Reserved	Level	SYSTEM	—	×	×	×
375					MHU	—	msg_ch33_ns	Non-secure message interrupt ch33 #Reserved	Level	SYSTEM	—	×	×	×
376					MHU	—	msg_ch36_ns	Non-secure message interrupt ch36	Level	SYSTEM	—	×	×	×
377					MHU	—	msg_ch37_ns	Non-secure message interrupt ch37	Level	SYSTEM	—	×	×	×
378					MHU	—	msg_ch38_ns	Non-secure message interrupt ch38	Level	SYSTEM	—	×	×	×
379					MHU	—	msg_ch39_ns	Non-secure message interrupt ch39	Level	SYSTEM	—	×	×	×
380					MHU	—	rsp_ch0_ns	Non-secure response interrupt ch0	Level	SYSTEM	—	×	×	×
381					MHU	—	rsp_ch1_ns	Non-secure response interrupt ch1	Level	SYSTEM	—	×	×	×
382					MHU	—	rsp_ch2_ns	Non-secure response interrupt ch2	Level	SYSTEM	—	×	×	×
383					MHU	—	rsp_ch6_ns	Non-secure response interrupt ch6 #Reserved	Level	SYSTEM	—	×	×	×
384					MHU	—	rsp_ch7_ns	Non-secure response interrupt ch7 #Reserved	Level	SYSTEM	—	×	×	×
385					MHU	—	rsp_ch8_ns	Non-secure response interrupt ch8	Level	SYSTEM	—	×	×	×
386					MHU	—	rsp_ch12_ns	Non-secure response interrupt ch12	Level	SYSTEM	—	×	×	×
387					MHU	—	rsp_ch13_ns	Non-secure response interrupt ch13	Level	SYSTEM	—	×	×	×
388					MHU	—	rsp_ch14_ns	Non-secure response interrupt ch14	Level	SYSTEM	—	×	×	×
389					MHU	—	rsp_ch18_ns	Non-secure response interrupt ch18 #Reserved	Level	SYSTEM	—	×	×	×

Table 4.6-24 List of Input Events (15/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
390					MHU	—	rsp_ch19_ns	Non-secure response interrupt ch19 #Reserved	Level	SYSTEM	—	×	×	×
391					MHU	—	rsp_ch20_ns	Non-secure response interrupt ch20	Level	SYSTEM	—	×	×	×
392					MHU	—	rsp_ch24_ns	Non-secure response interrupt ch24	Level	SYSTEM	—	×	×	×
393					MHU	—	rsp_ch25_ns	Non-secure response interrupt ch25	Level	SYSTEM	—	×	×	×
394					MHU	—	rsp_ch26_ns	Non-secure response interrupt ch26	Level	SYSTEM	—	×	×	×
395					MHU	—	rsp_ch27_ns	Non-secure response interrupt ch27 #Reserved	Level	SYSTEM	—	×	×	×
396					MHU	—	rsp_ch30_ns	Non-secure response interrupt ch30 #Reserved	Level	SYSTEM	—	×	×	×
397					MHU	—	rsp_ch31_ns	Non-secure response interrupt ch31	Level	SYSTEM	—	×	×	×
398					MHU	—	rsp_ch32_ns	Non-secure response interrupt ch32	Level	SYSTEM	—	×	×	×
399					MHU	—	rsp_ch33_ns	Non-secure response interrupt ch33	Level	SYSTEM	—	×	×	×
400					MHU	—	rsp_ch36_ns	Non-secure response interrupt ch36	Level	SYSTEM	—	×	×	×
401					MHU	—	rsp_ch37_ns	Non-secure response interrupt ch37 #Reserved	Level	SYSTEM	—	×	×	×
402					MHU	—	rsp_ch38_ns	Non-secure response interrupt ch38 #Reserved	Level	SYSTEM	—	×	×	×
403					MHU	—	rsp_ch39_ns	Non-secure response interrupt ch39	Level	SYSTEM	—	×	×	×
404					MHU	—	swint_ch12_ns	Software interrupt 12ch #Reserved	Level	SYSTEM	—	×	×	×
405					MHU	—	swint_ch13_ns	Software interrupt 13ch #Reserved	Level	SYSTEM	—	×	×	×
406					MHU	—	swint_ch14_ns	Software interrupt 14ch2 #Reserved	Level	SYSTEM	—	×	×	×
407					MHU	—	swint_ch15_ns	Software interrupt 15ch #Reserved	Level	SYSTEM	—	×	×	×
408					MHU	—	swint_ch17_ns	Software interrupt 17ch #Reserved	Level	SYSTEM	—	×	×	×
409					MHU	—	swint_ch18_ns	Software interrupt 18ch #Reserved	Level	SYSTEM	—	×	×	×
410					MHU	—	swint_ch19_ns	Software interrupt 19ch #Reserved	Level	SYSTEM	—	×	×	×
411					MHU	—	swint_ch20_ns	Software interrupt 20ch #Reserved	Level	SYSTEM	—	×	×	×
412					MHU	—	swint_ch22_ns	Software interrupt 22ch-CM33_to_CA55Core0	Level	SYSTEM	—	×	×	×
413					MHU	—	swint_ch23_ns	Software interrupt 23ch-CM33_to_CA55Core1	Level	SYSTEM	—	×	×	×
414					MHU	—	swint_ch24_ns	Software interrupt 24ch-CM33_to_CA55Core2	Level	SYSTEM	—	×	×	×
415					MHU	—	swint_ch25_ns	Software interrupt 25ch-CM33_to_CA55Core3	Level	SYSTEM	—	×	×	×
416	311				GIC	—	fault_int	Fault handling interrupt.	Level	SYSTEM	SYSTEM	×	×	×
417	312				GIC	—	err_int	Error handling interrupt.	Level	SYSTEM	SYSTEM	×	×	×
418	313				GIC	—	pmu_int	PMU overflow length.	Level	SYSTEM	SYSTEM	×	×	×
419	SEL0				ICU	TINT	TINT32_00	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×

Table 4.6-24 List of Input Events (16/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
420	SEL1				ICU	TINT	TINT32_01	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
421	SEL2				ICU	TINT	TINT32_02	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
422	SEL3				ICU	TINT	TINT32_03	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
423	SEL4				ICU	TINT	TINT32_04	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
424	SEL5				ICU	TINT	TINT32_05	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
425	SEL6				ICU	TINT	TINT32_06	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
426	SEL7				ICU	TINT	TINT32_07	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
427	SEL8				ICU	TINT	TINT32_08	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
428	SEL9				ICU	TINT	TINT32_09	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
429	SEL10				ICU	TINT	TINT32_10	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
430	SEL11				ICU	TINT	TINT32_11	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
431	SEL12				ICU	TINT	TINT32_12	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
432	SEL13				ICU	TINT	TINT32_13	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
433	SEL14				ICU	TINT	TINT32_14	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
434	SEL15				ICU	TINT	TINT32_15	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
435	SEL16				ICU	TINT	TINT32_16	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
436	SEL17				ICU	TINT	TINT32_17	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×

Table 4.6-24 List of Input Events (17/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
437	SEL18				ICU	TINT	TINT32_18	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
438	SEL19				ICU	TINT	TINT32_19	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
439	SEL20				ICU	TINT	TINT32_20	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
440	SEL21				ICU	TINT	TINT32_21	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
441	SEL22				ICU	TINT	TINT32_22	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
442	SEL23				ICU	TINT	TINT32_23	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
443	SEL24				ICU	TINT	TINT32_24	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
444	SEL25				ICU	TINT	TINT32_25	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
445	SEL26				ICU	TINT	TINT32_26	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
446	SEL27				ICU	TINT	TINT32_27	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
447	SEL28				ICU	TINT	TINT32_28	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
448	SEL29				ICU	TINT	TINT32_29	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
449	SEL30				ICU	TINT	TINT32_30	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
450	SEL31				ICU	TINT	TINT32_31	Signal after being merged within INT_marge(TINT)	Rising edge/falling edge/high-level detection/low-level detection	SINGLE	SELECT	×	×	×
451					ICU	GPT	GPT_U0_gpt_gtciada_n	Signal after being merged within INT_marge(GPT)	Level	SINGLE	—	×	×	×
452					ICU	GPT	GPT_U0_gpt_gtciadb_n	Signal after being merged within INT_marge(GPT)	Level	SINGLE	—	×	×	×
453					ICU	GPT	GPT_U1_gpt_gtciada_n	Signal after being merged within INT_marge(GPT)	Level	SINGLE	—	×	×	×
454					ICU	GPT	GPT_U1_gpt_gtciadb_n	Signal after being merged within INT_marge(GPT)	Level	SINGLE	—	×	×	×
455	SEL32	100			CMTW	CH0	CMTW_CH0_cmt2_cmp_pls_n	Interrupt due to compare match	Edge	SINGLE	SELECT	✓	×	×
456	SEL33	101			CMTW	CH1	CMTW_CH1_cmt2_cmp_pls_n	Interrupt due to compare match	Edge	SINGLE	SELECT	✓	×	×

Table 4.6-24 List of Input Events (18/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
457	SEL34	102			CMTW	CH2	CMTW_CH2_cmt2_cmp_pls_n	Interrupt due to compare match	Edge	SINGLE	SELECT	✓	×	×
458	SEL35	103			CMTW	CH3	CMTW_CH3_cmt2_cmp_pls_n	Interrupt due to compare match	Edge	SINGLE	SELECT	✓	×	×
459	SEL36	104			CMTW	CH4	CMTW_CH4_cmt2_cmp_pls_n	Interrupt due to compare match	Edge	SINGLE	SELECT	✓	×	×
460	SEL37	105			CMTW	CH5	CMTW_CH5_cmt2_cmp_pls_n	Interrupt due to compare match	Edge	SINGLE	SELECT	✓	×	×
461	SEL38	106			CMTW	CH6	CMTW_CH6_cmt2_cmp_pls_n	Interrupt due to compare match	Edge	SINGLE	SELECT	✓	×	×
462	SEL39	107			CMTW	CH7	CMTW_CH7_cmt2_cmp_pls_n	Interrupt due to compare match	Edge	SINGLE	SELECT	✓	×	×
463	SEL40	108			CMTW	CH0	CMTW_CH0_cmt2_ic0_pls_n	Interrupt due to input capture 0	Edge	SINGLE	SELECT	✓	×	×
464	SEL41	109			CMTW	CH1	CMTW_CH1_cmt2_ic0_pls_n	Interrupt due to input capture 0	Edge	SINGLE	SELECT	✓	×	×
465	SEL42	110			CMTW	CH2	CMTW_CH2_cmt2_ic0_pls_n	Interrupt due to input capture 0	Edge	SINGLE	SELECT	✓	×	×
466	SEL43	111			CMTW	CH3	CMTW_CH3_cmt2_ic0_pls_n	Interrupt due to input capture 0	Edge	SINGLE	SELECT	✓	×	×
467	SEL44	112			—	—	Reserved	Reserved	—	—	—	—	—	—
468	SEL45	113			—	—	Reserved	Reserved	—	—	—	—	—	—
469	SEL46	114			—	—	Reserved	Reserved	—	—	—	—	—	—
470	SEL47	115			—	—	Reserved	Reserved	—	—	—	—	—	—
471	SEL48	116			CMTW	CH0	CMTW_CH0_cmt2_ic1_pls_n	Interrupt due to input capture 1	Edge	SINGLE	SELECT	✓	×	×
472	SEL49	117			CMTW	CH1	CMTW_CH1_cmt2_ic1_pls_n	Interrupt due to input capture 1	Edge	SINGLE	SELECT	✓	×	×
473	SEL50	118			CMTW	CH2	CMTW_CH2_cmt2_ic1_pls_n	Interrupt due to input capture 1	Edge	SINGLE	SELECT	✓	×	×
474	SEL51	119			CMTW	CH3	CMTW_CH3_cmt2_ic1_pls_n	Interrupt due to input capture 1	Edge	SINGLE	SELECT	✓	×	×
475	SEL52	120			—	—	Reserved	Reserved	—	—	—	—	—	—
476	SEL53	121			—	—	Reserved	Reserved	—	—	—	—	—	—
477	SEL54	122			—	—	Reserved	Reserved	—	—	—	—	—	—
478	SEL55	123			—	—	Reserved	Reserved	—	—	—	—	—	—
479	SEL56	124			CMTW	CH0	CMTW_CH0_cmt2_oc0_pls_n	Interrupt due to output compare 0	Edge	SINGLE	SELECT	✓	×	×
480	SEL57	125			CMTW	CH1	CMTW_CH1_cmt2_oc0_pls_n	Interrupt due to output compare 0	Edge	SINGLE	SELECT	✓	×	×
481	SEL58	126			CMTW	CH2	CMTW_CH2_cmt2_oc0_pls_n	Interrupt due to output compare 0	Edge	SINGLE	SELECT	✓	×	×
482	SEL59	127			CMTW	CH3	CMTW_CH3_cmt2_oc0_pls_n	Interrupt due to output compare 0	Edge	SINGLE	SELECT	✓	×	×
483	SEL60	128			—	—	Reserved	Reserved	—	—	—	—	—	—
484	SEL61	129			—	—	Reserved	Reserved	—	—	—	—	—	—
485	SEL62	130			—	—	Reserved	Reserved	—	—	—	—	—	—
486	SEL63	131			—	—	Reserved	Reserved	—	—	—	—	—	—
487	SEL64	132			CMTW	CH0	CMTW_CH0_cmt2_oc1_pls_n	Interrupt due to output compare 1	Edge	SINGLE	SELECT	✓	×	×
488	SEL65	133			CMTW	CH1	CMTW_CH1_cmt2_oc1_pls_n	Interrupt due to output compare 1	Edge	SINGLE	SELECT	✓	×	×
489	SEL66	134			CMTW	CH2	CMTW_CH2_cmt2_oc1_pls_n	Interrupt due to output compare 1	Edge	SINGLE	SELECT	✓	×	×
490	SEL67	135			CMTW	CH3	CMTW_CH3_cmt2_oc1_pls_n	Interrupt due to output compare 1	Edge	SINGLE	SELECT	✓	×	×
491	SEL68	136			—	—	Reserved	Reserved	—	—	—	—	—	—
492	SEL69	137			—	—	Reserved	Reserved	—	—	—	—	—	—
493	SEL70	138			—	—	Reserved	Reserved	—	—	—	—	—	—

Table 4.6-24 List of Input Events (19/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
494	SEL71	139			—	—	Reserved	Reserved	—	—	—	—	—	—
495	SEL72				DMAC	CH1	DMAC1_DMAERR	DMAC1 Error Interrupt	Level/edge	SINGLE	SELECT	×	×	×
496	SEL73				DMAC	CH2	DMAC2_DMAERR	DMAC2 Error Interrupt	Level/edge	SINGLE	SELECT	×	×	×
497	SEL74				DMAC	CH3	DMAC3_DMAERR	DMAC3 Error Interrupt	Level/edge	SINGLE	SELECT	×	×	×
498	SEL75				DMAC	CH4	DMAC4_DMAERR	DMAC4 Error Interrupt	Level/edge	SINGLE	SELECT	×	×	×
499	SEL76				DMAC	CH0	DMAC0_DMAERR	DMAC0 Error Interrupt	Level/edge	SINGLE	SELECT	×	×	×
500	SEL77	140			RSPI	CH0	RSPI_CH0_sp_rxintpls_n	Receive buffer full interrupt	Edge	SINGLE	SELECT	✓	×	×
501	SEL78	141			RSPI	CH0	RSPI_CH0_sp_txintpls_n	Transmit buffer empty interrupt	Edge	SINGLE	SELECT	✓	×	×
502	SEL79	142			RSPI	CH1	RSPI_CH1_sp_rxintpls_n	Receive buffer full interrupt	Edge	SINGLE	SELECT	✓	×	×
503	SEL80	143			RSPI	CH1	RSPI_CH1_sp_txintpls_n	Transmit buffer empty interrupt	Edge	SINGLE	SELECT	✓	×	×
504	SEL81	144			RSPI	CH2	RSPI_CH2_sp_rxintpls_n	Receive buffer full interrupt	Edge	SINGLE	SELECT	✓	×	×
505	SEL82	145			RSPI	CH2	RSPI_CH2_sp_txintpls_n	Transmit buffer empty interrupt	Edge	SINGLE	SELECT	✓	×	×
506	SEL83	176			RIIC	CH0	RIIC_CH0_ti_n	Trans interrupt request	Edge	SINGLE	SELECT	✓	×	×
507	SEL84	177			RIIC	CH0	RIIC_CH0_ri_n	Receive interrupt request	Edge	SINGLE	SELECT	✓	×	×
508	SEL85	178			RIIC	CH1	RIIC_CH1_ti_n	Trans interrupt request	Edge	SINGLE	SELECT	✓	×	×
509	SEL86	179			RIIC	CH1	RIIC_CH1_ri_n	Receive interrupt request	Edge	SINGLE	SELECT	✓	×	×
510	SEL87	180			RIIC	CH2	RIIC_CH2_ti_n	Trans interrupt request	Edge	SINGLE	SELECT	✓	×	×
511	SEL88	181			RIIC	CH2	RIIC_CH2_ri_n	Receive interrupt request	Edge	SINGLE	SELECT	✓	×	×
512	SEL89	182			RIIC	CH3	RIIC_CH3_ti_n	Trans interrupt request	Edge	SINGLE	SELECT	✓	×	×
513	SEL90	183			RIIC	CH3	RIIC_CH3_ri_n	Receive interrupt request	Edge	SINGLE	SELECT	✓	×	×
514	SEL91	184			RIIC	CH4	RIIC_CH4_ti_n	Trans interrupt request	Edge	SINGLE	SELECT	✓	×	×
515	SEL92	185			RIIC	CH4	RIIC_CH4_ri_n	Receive interrupt request	Edge	SINGLE	SELECT	✓	×	×
516	SEL93	186			RIIC	CH5	RIIC_CH5_ti_n	Trans interrupt request	Edge	SINGLE	SELECT	✓	×	×
517	SEL94	187			RIIC	CH5	RIIC_CH5_ri_n	Receive interrupt request	Edge	SINGLE	SELECT	✓	×	×
518	SEL95	188			RIIC	CH6	RIIC_CH6_ti_n	Trans interrupt request	Edge	SINGLE	SELECT	✓	×	×
519	SEL96	189			RIIC	CH6	RIIC_CH6_ri_n	Receive interrupt request	Edge	SINGLE	SELECT	✓	×	×
520	SEL97	190			RIIC	CH7	RIIC_CH7_ti_n	Trans interrupt request	Edge	SINGLE	SELECT	✓	×	×
521	SEL98	191			RIIC	CH7	RIIC_CH7_ri_n	Receive interrupt request	Edge	SINGLE	SELECT	✓	×	×
522	SEL99	192			RIIC	CH8	RIIC_CH8_ti_n	Trans interrupt request	Edge	SINGLE	SELECT	✓	×	×
523	SEL100	193			RIIC	CH8	RIIC_CH8_ri_n	Receive interrupt request	Edge	SINGLE	SELECT	✓	×	×
524	SEL101				RTC	—	intreq_rtc_almpls_n	Alarm interrupt request (pulse)	Edge	SINGLE	SELECT	×	×	×

Table 4.6-24 List of Input Events (20/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
525	SEL102				RTC	—	intreq_rtc_prdpls_n	Periodical interrupt request (pulse)	Edge	SINGLE	SELECT	×	×	×
526	SEL103				RTC	—	intreq_rtc_cuppls_n	Carry interrupt request (pulse)	Edge	SINGLE	SELECT	×	×	×
527	SEL104				—	—	Reserved	Reserved	—	—	—	—	—	—
528	SEL105				—	—	Reserved	Reserved	—	—	—	—	—	—
529	SEL106				SCIF	—	ub1_rerr_n	INTC: framing error/parity error interrupt signal (level-sensed)	Level	SINGLE	SELECT	×	×	×
530	SEL107				SCIF	—	ub1_brk_n	INTC: break/overrun interrupt signal (level-sensed)	Level	SINGLE	SELECT	×	×	×
531	SEL108		262		SCIF	—	ub1_tei_n	INTC: transmission end interrupt signal (level-sensed)	Level	SINGLE	SELECT	×	✓	×
532	SEL109	207			SCIF	—	ub1_rxi_n	INTC: reception data full interrupt signal (level-sensed)	Level	SINGLE	SELECT	✓	×	×
533	SEL110	208			SCIF	—	ub1_txi_n	INTC: transmission data empty interrupt signal (level-sensed)	Level	SINGLE	SELECT	✓	×	×
534	SEL111				SCIF	—	ub1_dri_n	INTC: reception data ready interrupt signal (level-sensed)	Level	SINGLE	SELECT	×	×	×
535	SEL112				SCIF	—	ub1_tei_dri_n	INTC: transmission end/reception data ready interrupt signal (level-sensed)	Level	SINGLE	SELECT	×	×	×
536	SEL113		263		SCIF	—	ub1_rxi_edge_n	INTC: reception data full interrupt signal (edge-sensed)	Edge	SINGLE	SELECT	×	✓	×
537	SEL114		264		SCIF	—	ub1_txi_edge_n	INTC: transmission data empty interrupt signal (edge-sensed)	Edge	SINGLE	SELECT	×	✓	×
538	SEL115	209			GPT	Unit0	GPT_U0_gpt_gtcia_n_0	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
539	SEL116	210			GPT	Unit0	GPT_U0_gpt_gtcia_n_1	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
540	SEL117	211			GPT	Unit0	GPT_U0_gpt_gtcia_n_2	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
541	SEL118	212			GPT	Unit0	GPT_U0_gpt_gtcia_n_3	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
542	SEL119	213			GPT	Unit0	GPT_U0_gpt_gtcia_n_4	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
543	SEL120	214			GPT	Unit0	GPT_U0_gpt_gtcia_n_5	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
544	SEL121	215			GPT	Unit0	GPT_U0_gpt_gtcia_n_6	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
545	SEL122	216			GPT	Unit0	GPT_U0_gpt_gtcia_n_7	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
546	SEL123	217			GPT	Unit0	GPT_U0_gpt_gtcib_n_0	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
547	SEL124	218			GPT	Unit0	GPT_U0_gpt_gtcib_n_1	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×

Table 4.6-24 List of Input Events (21/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
548	SEL125	219			GPT	Unit0	GPT_U0_gpt_gtcib_n_2	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
549	SEL126	220			GPT	Unit0	GPT_U0_gpt_gtcib_n_3	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
550	SEL127	221			GPT	Unit0	GPT_U0_gpt_gtcib_n_4	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
551	SEL128	222			GPT	Unit0	GPT_U0_gpt_gtcib_n_5	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
552	SEL129	223			GPT	Unit0	GPT_U0_gpt_gtcib_n_6	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
553	SEL130	224			GPT	Unit0	GPT_U0_gpt_gtcib_n_7	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
554	SEL131	225			GPT	Unit0	GPT_U0_gpt_gtcic_n_0	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
555	SEL132	226			GPT	Unit0	GPT_U0_gpt_gtcic_n_1	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
556	SEL133	227			GPT	Unit0	GPT_U0_gpt_gtcic_n_2	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
557	SEL134	228			GPT	Unit0	GPT_U0_gpt_gtcic_n_3	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
558	SEL135	229			GPT	Unit0	GPT_U0_gpt_gtcic_n_4	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
559	SEL136	230			GPT	Unit0	GPT_U0_gpt_gtcic_n_5	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
560	SEL137	231			GPT	Unit0	GPT_U0_gpt_gtcic_n_6	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
561	SEL138	232			GPT	Unit0	GPT_U0_gpt_gtcic_n_7	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
562	SEL139	233			GPT	Unit0	GPT_U0_gpt_gtcid_n_0	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
563	SEL140	234			GPT	Unit0	GPT_U0_gpt_gtcid_n_1	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
564	SEL141	235			GPT	Unit0	GPT_U0_gpt_gtcid_n_2	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
565	SEL142	236			GPT	Unit0	GPT_U0_gpt_gtcid_n_3	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
566	SEL143	237			GPT	Unit0	GPT_U0_gpt_gtcid_n_4	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
567	SEL144	238			GPT	Unit0	GPT_U0_gpt_gtcid_n_5	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
568	SEL145	239			GPT	Unit0	GPT_U0_gpt_gtcid_n_6	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
569	SEL146	240			GPT	Unit0	GPT_U0_gpt_gtcid_n_7	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
570	SEL147	241			GPT	Unit0	GPT_U0_gpt_gtcie_n_0	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
571	SEL148	242			GPT	Unit0	GPT_U0_gpt_gtcie_n_1	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
572	SEL149	243			GPT	Unit0	GPT_U0_gpt_gtcie_n_2	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
573	SEL150	244			GPT	Unit0	GPT_U0_gpt_gtcie_n_3	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
574	SEL151	245			GPT	Unit0	GPT_U0_gpt_gtcie_n_4	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
575	SEL152	246			GPT	Unit0	GPT_U0_gpt_gtcie_n_5	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
576	SEL153	247			GPT	Unit0	GPT_U0_gpt_gtcie_n_6	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×

Table 4.6-24 List of Input Events (22/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
577	SEL154	248			GPT	Unit0	GPT_U0_gpt_gtcie_n_7	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
578	SEL155	249			GPT	Unit0	GPT_U0_gpt_gtcif_n_0	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
579	SEL156	250			GPT	Unit0	GPT_U0_gpt_gtcif_n_1	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
580	SEL157	251			GPT	Unit0	GPT_U0_gpt_gtcif_n_2	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
581	SEL158	252			GPT	Unit0	GPT_U0_gpt_gtcif_n_3	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
582	SEL159	253			GPT	Unit0	GPT_U0_gpt_gtcif_n_4	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
583	SEL160	254			GPT	Unit0	GPT_U0_gpt_gtcif_n_5	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
584	SEL161	255			GPT	Unit0	GPT_U0_gpt_gtcif_n_6	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
585	SEL162	256			GPT	Unit0	GPT_U0_gpt_gtcif_n_7	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
586	SEL195		265		GPT	Unit0	GPT_U0_gpt_gtcih_n_0	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
587	SEL196		266		GPT	Unit0	GPT_U0_gpt_gtcih_n_1	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
588	SEL197		267		GPT	Unit0	GPT_U0_gpt_gtcih_n_2	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
589	SEL198		268		GPT	Unit0	GPT_U0_gpt_gtcih_n_3	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
590	SEL199		269		GPT	Unit0	GPT_U0_gpt_gtcih_n_4	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
591	SEL200		270		GPT	Unit0	GPT_U0_gpt_gtcih_n_5	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
592	SEL201		271		GPT	Unit0	GPT_U0_gpt_gtcih_n_6	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
593	SEL202		272		GPT	Unit0	GPT_U0_gpt_gtcih_n_7	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
594	SEL203		273		GPT	Unit0	GPT_U0_gpt_gtcil_n_0	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
595	SEL204		274		GPT	Unit0	GPT_U0_gpt_gtcil_n_1	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
596	SEL205		275		GPT	Unit0	GPT_U0_gpt_gtcil_n_2	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
597	SEL206		276		GPT	Unit0	GPT_U0_gpt_gtcil_n_3	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
598	SEL207		277		GPT	Unit0	GPT_U0_gpt_gtcil_n_4	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
599	SEL208		278		GPT	Unit0	GPT_U0_gpt_gtcil_n_5	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
600	SEL209		279		GPT	Unit0	GPT_U0_gpt_gtcil_n_6	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
601	SEL210		280		GPT	Unit0	GPT_U0_gpt_gtcil_n_7	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
602	SEL219	273			GPT	Unit1	GPT_U1_gpt_gtcia_n_0	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
603	SEL220	274			GPT	Unit1	GPT_U1_gpt_gtcia_n_1	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
604	SEL221	275			GPT	Unit1	GPT_U1_gpt_gtcia_n_2	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
605	SEL222	276			GPT	Unit1	GPT_U1_gpt_gtcia_n_3	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×

Table 4.6-24 List of Input Events (23/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
606	SEL223	277			GPT	Unit1	GPT_U1_gpt_gtcia_n_4	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
607	SEL224	278			GPT	Unit1	GPT_U1_gpt_gtcia_n_5	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
608	SEL225	279			GPT	Unit1	GPT_U1_gpt_gtcia_n_6	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
609	SEL226	280			GPT	Unit1	GPT_U1_gpt_gtcia_n_7	Input capture/compare match of the GTCCRA register	Edge	SINGLE	SELECT	✓	×	×
610	SEL227	281			GPT	Unit1	GPT_U1_gpt_gtcib_n_0	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
611	SEL228	282			GPT	Unit1	GPT_U1_gpt_gtcib_n_1	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
612	SEL229	283			GPT	Unit1	GPT_U1_gpt_gtcib_n_2	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
613	SEL230	284			GPT	Unit1	GPT_U1_gpt_gtcib_n_3	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
614	SEL231	285			GPT	Unit1	GPT_U1_gpt_gtcib_n_4	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
615	SEL232	286			GPT	Unit1	GPT_U1_gpt_gtcib_n_5	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
616	SEL233	287			GPT	Unit1	GPT_U1_gpt_gtcib_n_6	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
617	SEL234	288			GPT	Unit1	GPT_U1_gpt_gtcib_n_7	Input capture/compare match of the GTCCRB register	Edge	SINGLE	SELECT	✓	×	×
618	SEL235	289			GPT	Unit1	GPT_U1_gpt_gtcic_n_0	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
619	SEL236	290			GPT	Unit1	GPT_U1_gpt_gtcic_n_1	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
620	SEL237	291			GPT	Unit1	GPT_U1_gpt_gtcic_n_2	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
621	SEL238	292			GPT	Unit1	GPT_U1_gpt_gtcic_n_3	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
622	SEL239	293			GPT	Unit1	GPT_U1_gpt_gtcic_n_4	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
623	SEL240	294			GPT	Unit1	GPT_U1_gpt_gtcic_n_5	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
624	SEL241	295			GPT	Unit1	GPT_U1_gpt_gtcic_n_6	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
625	SEL242	296			GPT	Unit1	GPT_U1_gpt_gtcic_n_7	Compare match with the GTCCRC register	Edge	SINGLE	SELECT	✓	×	×
626	SEL243	297			GPT	Unit1	GPT_U1_gpt_gtcid_n_0	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
627	SEL244	298			GPT	Unit1	GPT_U1_gpt_gtcid_n_1	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
628	SEL245	299			GPT	Unit1	GPT_U1_gpt_gtcid_n_2	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
629	SEL246	300			GPT	Unit1	GPT_U1_gpt_gtcid_n_3	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
630	SEL247	301			GPT	Unit1	GPT_U1_gpt_gtcid_n_4	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
631	SEL248	302			GPT	Unit1	GPT_U1_gpt_gtcid_n_5	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×

Table 4.6-24 List of Input Events (24/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
632	SEL249	303			GPT	Unit1	GPT_U1_gpt_gtcid_n_6	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
633	SEL250	304			GPT	Unit1	GPT_U1_gpt_gtcid_n_7	Compare match with the GTCCRD register	Edge	SINGLE	SELECT	✓	×	×
634	SEL251	305			GPT	Unit1	GPT_U1_gpt_gtcie_n_0	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
635	SEL252	306			GPT	Unit1	GPT_U1_gpt_gtcie_n_1	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
636	SEL253	307			GPT	Unit1	GPT_U1_gpt_gtcie_n_2	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
637	SEL254	308			GPT	Unit1	GPT_U1_gpt_gtcie_n_3	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
638	SEL255	309			GPT	Unit1	GPT_U1_gpt_gtcie_n_4	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
639	SEL256	310			GPT	Unit1	GPT_U1_gpt_gtcie_n_5	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
640	SEL257	311			GPT	Unit1	GPT_U1_gpt_gtcie_n_6	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
641	SEL258	312			GPT	Unit1	GPT_U1_gpt_gtcie_n_7	Compare match with the GTCCRE register	Edge	SINGLE	SELECT	✓	×	×
642	SEL259	313			GPT	Unit1	GPT_U1_gpt_gtcif_n_0	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
643	SEL260	314			GPT	Unit1	GPT_U1_gpt_gtcif_n_1	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
644	SEL261	315			GPT	Unit1	GPT_U1_gpt_gtcif_n_2	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
645	SEL262	316			GPT	Unit1	GPT_U1_gpt_gtcif_n_3	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
646	SEL263	317			GPT	Unit1	GPT_U1_gpt_gtcif_n_4	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
647	SEL264	318			GPT	Unit1	GPT_U1_gpt_gtcif_n_5	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
648	SEL265	319			GPT	Unit1	GPT_U1_gpt_gtcif_n_6	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
649	SEL266	320			GPT	Unit1	GPT_U1_gpt_gtcif_n_7	Compare match with the GTCCRF register	Edge	SINGLE	SELECT	✓	×	×
650	SEL299		345		GPT	Unit1	GPT_U1_gpt_gtcih_n_0	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
651	SEL300		346		GPT	Unit1	GPT_U1_gpt_gtcih_n_1	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
652	SEL301		347		GPT	Unit1	GPT_U1_gpt_gtcih_n_2	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
653	SEL302		348		GPT	Unit1	GPT_U1_gpt_gtcih_n_3	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
654	SEL303		349		GPT	Unit1	GPT_U1_gpt_gtcih_n_4	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
655	SEL304		350		GPT	Unit1	GPT_U1_gpt_gtcih_n_5	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
656	SEL305		351		GPT	Unit1	GPT_U1_gpt_gtcih_n_6	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
657	SEL306		352		GPT	Unit1	GPT_U1_gpt_gtcih_n_7	A and B both high interrupt	Edge	SINGLE	SELECT	×	✓	×
658	SEL307		353		GPT	Unit1	GPT_U1_gpt_gtcil_n_0	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
659	SEL308		354		GPT	Unit1	GPT_U1_gpt_gtcil_n_1	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
660	SEL309		355		GPT	Unit1	GPT_U1_gpt_gtcil_n_2	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
661	SEL310		356		GPT	Unit1	GPT_U1_gpt_gtcil_n_3	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
662	SEL311		357		GPT	Unit1	GPT_U1_gpt_gtcil_n_4	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×

Table 4.6-24 List of Input Events (25/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
663	SEL312		358		GPT	Unit1	GPT_U1_gpt_gtcil_n_5	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
664	SEL313		359		GPT	Unit1	GPT_U1_gpt_gtcil_n_6	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
665	SEL314		360		GPT	Unit1	GPT_U1_gpt_gtcil_n_7	A and B both low interrupt	Edge	SINGLE	SELECT	×	✓	×
666	SEL323				POEG	CH0	POEG_CH0_mhc_elcoutdis	POEG group A interrupt	Level	SINGLE	SELECT	×	×	×
667	SEL324				POEG	CH1	POEG_CH1_mhc_elcoutdis	POEG group B interrupt	Level	SINGLE	SELECT	×	×	×
668	SEL325				POEG	CH2	POEG_CH2_mhc_elcoutdis	POEG group C interrupt	Level	SINGLE	SELECT	×	×	×
669	SEL326				POEG	CH3	POEG_CH3_mhc_elcoutdis	POEG group D interrupt	Level	SINGLE	SELECT	×	×	×
670	SEL327				POEG	CH4	POEG_CH4_mhc_elcoutdis	POEG group A interrupt	Level	SINGLE	SELECT	×	×	×
671	SEL328				POEG	Ch5	POEG_CH5_mhc_elcoutdis	POEG group B interrupt	Level	SINGLE	SELECT	×	×	×
672	SEL329				POEG	Ch6	POEG_CH6_mhc_elcoutdis	POEG group C interrupt	Level	SINGLE	SELECT	×	×	×
673	SEL330				POEG	CH7	POEG_CH7_mhc_elcoutdis	POEG group D interrupt	Level	SINGLE	SELECT	×	×	×
674	SEL331				I3C	—	INT_ri3c_ierr_n	Non-recoverable internal error interrupt	Level	SINGLE	SELECT	×	×	×
675	SEL332				I3C	—	INT_ri3c_terr_n	Normal transfer error interrupt	Level	SINGLE	SELECT	×	×	×
676	SEL333				I3C	—	INT_ri3c_abort_n	Normal transfer abort interrupt	Level	SINGLE	SELECT	×	×	×
677	SEL334				I3C	—	INT_ri3c_resp_n	Normal response status buffer full interrupt	Edge	SINGLE	SELECT	×	×	×
678	SEL335				I3C	—	INT_ri3c_cmd_n	Normal command buffer empty interrupt	Edge	SINGLE	SELECT	×	×	×
679	SEL336				I3C	—	INT_ri3c_ibi_n	Normal IBI status buffer full interrupt	Edge	SINGLE	SELECT	×	×	×
680	SEL337	337			I3C	—	INT_ri3c_rx_n	Normal Rx data buffer full interrupt	Edge	SINGLE	SELECT	✓	×	×
681	SEL338	338			I3C	—	INT_ri3c_tx_n	Normal Tx data buffer empty interrupt	Edge	SINGLE	SELECT	✓	×	×
682	SEL339				I3C	—	INT_ri3c_rcv_n	Normal receive status buffer full interrupt	Edge	SINGLE	SELECT	×	×	×
683	SEL340				—	—	Reserved	Reserved	—	—	—	—	—	—
684	SEL341				—	—	Reserved	Reserved	—	—	—	—	—	—
685	SEL342				—	—	Reserved	Reserved	—	—	—	—	—	—
686	SEL343				—	—	Reserved	Reserved	—	—	—	—	—	—
687	SEL344				—	—	Reserved	Reserved	—	—	—	—	—	—
688	SEL345				—	—	Reserved	Reserved	—	—	—	—	—	—
689	SEL346				I3C	—	INT_ri3c_st_n	START condition detection interrupt	Level	SINGLE	SELECT	×	×	×
690	SEL347				I3C	—	INT_ri3c_sp_n	STOP condition detection interrupt	Level	SINGLE	SELECT	×	×	×
691	SEL348				—	—	Reserved	Reserved	—	—	—	—	—	—
692	SEL349				I3C	—	INT_ri3c_tend_n	Transmit end interrupt	Level	SINGLE	SELECT	×	×	×
693	SEL350				I3C	—	INT_ri3c_nack_n	NACK detection interrupt	Level	SINGLE	SELECT	×	×	×

Table 4.6-24 List of Input Events (26/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
694	SEL351				I3C	—	INT_ri3c_al_n	Arbitration lost interrupt	Level	SINGLE	SELECT	×	×	×
695	SEL352				I3C	—	INT_ri3c_tmo_n	Timeout detection interrupt	Level	SINGLE	SELECT	×	×	×
696	SEL353				I3C	—	INT_ri3c_wu_n	Wake-up condition detection interrupt	Level	SINGLE	SELECT	×	×	×
697	SEL354				CANFD	—	can_cherr_int_0	Channel CAN error interrupt lines	Level	SINGLE	SELECT	×	×	×
698	SEL355				CANFD	—	can_cherr_int_1	Channel CAN error interrupt lines	Level	SINGLE	SELECT	×	×	×
699	SEL356				CANFD	—	can_cherr_int_2	Channel CAN error interrupt lines	Level	SINGLE	SELECT	×	×	×
700	SEL357				CANFD	—	can_cherr_int_3	Channel CAN error interrupt lines	Level	SINGLE	SELECT	×	×	×
701	SEL358				CANFD	—	can_cherr_int_4	Channel CAN error interrupt lines	Level	SINGLE	SELECT	×	×	×
702	SEL359				CANFD	—	can_cherr_int_5	Channel CAN error interrupt lines	Level	SINGLE	SELECT	×	×	×
703	SEL360				CANFD	—	can_comfrx_int_0	COM RX FIFO or TXQ interrupt lines	Level	SINGLE	SELECT	×	×	×
704	SEL361				CANFD	—	can_comfrx_int_1	COM RX FIFO or TXQ interrupt lines	Level	SINGLE	SELECT	×	×	×
705	SEL362				CANFD	—	can_comfrx_int_2	COM RX FIFO or TXQ interrupt lines	Level	SINGLE	SELECT	×	×	×
706	SEL363				CANFD	—	can_comfrx_int_3	COM RX FIFO or TXQ interrupt lines	Level	SINGLE	SELECT	×	×	×
707	SEL364				CANFD	—	can_comfrx_int_4	COM RX FIFO or TXQ interrupt lines	Level	SINGLE	SELECT	×	×	×
708	SEL365				CANFD	—	can_comfrx_int_5	COM RX FIFO or TXQ interrupt lines	Level	SINGLE	SELECT	×	×	×
709	SEL366				CANFD	—	can_glerr_int	Global Error Interrupt line	Level	SINGLE	SELECT	×	×	×
710	SEL367				CANFD	—	can_rxf_int	RX FIFO interrupt line	Level	SINGLE	SELECT	×	×	×
711	SEL368				CANFD	—	can_tx_int_0	Channel TX interrupt lines	Level	SINGLE	SELECT	×	×	×
712	SEL369				CANFD	—	can_tx_int_1	Channel TX interrupt lines	Level	SINGLE	SELECT	×	×	×
713	SEL370				CANFD	—	can_tx_int_2	Channel TX interrupt lines	Level	SINGLE	SELECT	×	×	×
714	SEL371				CANFD	—	can_tx_int_3	Channel TX interrupt lines	Level	SINGLE	SELECT	×	×	×
715	SEL372				CANFD	—	can_tx_int_4	Channel TX interrupt lines	Level	SINGLE	SELECT	×	×	×
716	SEL373				CANFD	—	can_tx_int_5	Channel TX interrupt lines	Level	SINGLE	SELECT	×	×	×
717	SEL374				—	—	Reserved	Reserved	—	—	—	—	—	—
718	SEL375				—	—	Reserved	Reserved	—	—	—	—	—	—
719	SEL376				—	—	Reserved	Reserved	—	—	—	—	—	—
720	SEL377				—	—	Reserved	Reserved	—	—	—	—	—	—
721	SEL378				—	—	Reserved	Reserved	—	—	—	—	—	—
722	SEL379				—	—	Reserved	Reserved	—	—	—	—	—	—
723	SEL380				—	—	Reserved	Reserved	—	—	—	—	—	—
724	SEL381				—	—	Reserved	Reserved	—	—	—	—	—	—

Table 4.6-24 List of Input Events (27/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
725	SEL382				—	—	Reserved	Reserved	—	—	—	—	—	—
726	SEL383				—	—	Reserved	Reserved	—	—	—	—	—	—
727	SEL384				—	—	Reserved	Reserved	—	—	—	—	—	—
728	SEL385				—	—	Reserved	Reserved	—	—	—	—	—	—
729	SEL386				—	—	Reserved	Reserved	—	—	—	—	—	—
730	SEL387				—	—	Reserved	Reserved	—	—	—	—	—	—
731	SEL388				—	—	Reserved	Reserved	—	—	—	—	—	—
732	SEL389				—	—	Reserved	Reserved	—	—	—	—	—	—
733	SEL390				—	—	Reserved	Reserved	—	—	—	—	—	—
734	SEL391				—	—	Reserved	Reserved	—	—	—	—	—	—
735	SEL392				SD	ch0	OXMNIRQ	Interrupt	Level	SINGLE	SELECT	×	×	×
736	SEL393				SD	ch0	OXASIOIRQ	SDIO Interrupt(async)	Level	SINGLE	SELECT	×	×	×
737	SEL394				SD	ch1	OXMNIRQ	Interrupt	Level	SINGLE	SELECT	×	×	×
738	SEL395				SD	ch1	OXASIOIRQ	SDIO Interrupt(async)	Level	SINGLE	SELECT	×	×	×
739	SEL396				SD	ch2	OXMNIRQ	Interrupt	Level	SINGLE	SELECT	×	×	×
740	SEL397				SD	ch2	OXASIOIRQ	SDIO Interrupt(async)	Level	SINGLE	SELECT	×	×	×
741	SEL398				USB2	Host0	—	—	—	SINGLE	SELECT	×	×	×
742	SEL399				USB2	Host0	—	—	—	SINGLE	SELECT	×	×	×
743	SEL400				USB2	Host0	—	—	—	SINGLE	SELECT	×	×	×
744	SEL401				USB2	Host0	—	—	—	SINGLE	SELECT	×	×	×
745	SEL402				USB2	Host0	—	—	—	SINGLE	SELECT	×	×	×
746	SEL403				USB2	Host1	—	—	—	SINGLE	SELECT	×	×	×
747	SEL404				USB2	Host1	—	—	—	SINGLE	SELECT	×	×	×
748	SEL405				USB2	Host1	—	—	—	SINGLE	SELECT	×	×	×
749	SEL406				USB2	Host1	—	—	—	SINGLE	SELECT	×	×	×
750	SEL407				USB2	Host1	—	—	—	SINGLE	SELECT	×	×	×
751	SEL408				USB2	Peri	—	—	—	SINGLE	SELECT	×	×	×
752	SEL409				USB2	Peri	—	—	—	SINGLE	SELECT	×	×	×
753	SEL410				USB2	Peri	—	—	—	SINGLE	SELECT	×	×	×
754	SEL411				USB2	Peri	—	—	—	SINGLE	SELECT	×	×	×
755	SEL412				USB3	Host0	—	—	—	SINGLE	SELECT	×	×	×

Table 4.6-24 List of Input Events (28/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
756	SEL413				USB3	Host0	—	—	—	SINGLE	SELECT	×	×	×
757	SEL414				USB3	Host0	—	—	—	SINGLE	SELECT	×	×	×
758	SEL415				USB3	Host0	—	—	—	SINGLE	SELECT	×	×	×
759	SEL416				USB3	Host0	—	—	—	SINGLE	SELECT	×	×	×
760	SEL417				—	—	Reserved	Reserved	—	—	—	—	—	—
761	SEL418				—	—	Reserved	Reserved	—	—	—	—	—	—
762	SEL419				—	—	Reserved	Reserved	—	—	—	—	—	—
763	SEL420				—	—	Reserved	Reserved	—	—	—	—	—	—
764	SEL421				—	—	Reserved	Reserved	—	—	—	—	—	—
765	SEL422				GBETH	PORT 0	—	—	—	SINGLE	SELECT	×	×	×
766	SEL423				GBETH	PORT 0	—	—	—	SINGLE	SELECT	×	×	×
767	SEL424				GBETH	PORT 0	—	—	—	SINGLE	SELECT	×	×	×
768	SEL425				GBETH	PORT 0	—	—	—	SINGLE	SELECT	×	×	×
769	SEL426				GBETH	PORT 0	—	—	—	SINGLE	SELECT	×	×	×
770	SEL427				GBETH	PORT 0	—	—	—	SINGLE	SELECT	×	×	×
771	SEL428				GBETH	PORT 0	—	—	—	SINGLE	SELECT	×	×	×
772	SEL429				GBETH	PORT 0	—	—	—	SINGLE	SELECT	×	×	×
773	SEL430				GBETH	PORT 0	—	—	—	SINGLE	SELECT	×	×	×
774	SEL431				GBETH	PORT 0	—	—	—	SINGLE	SELECT	×	×	×
775	SEL432				GBETH	PORT 0	—	—	—	SINGLE	SELECT	×	×	×
776	SEL433				—	—	Reserved	Reserved	—	—	—	—	—	—
777	SEL434				—	—	Reserved	Reserved	—	—	—	—	—	—
778	SEL435				—	—	Reserved	Reserved	—	—	—	—	—	—
779	SEL436				—	—	Reserved	Reserved	—	—	—	—	—	—
780	SEL437				GBETH	PORT 1	—	—	—	SINGLE	SELECT	×	×	×
781	SEL438				GBETH	PORT 1	—	—	—	SINGLE	SELECT	×	×	×
782	SEL439				GBETH	PORT 1	—	—	—	SINGLE	SELECT	×	×	×
783	SEL440				GBETH	PORT 1	—	—	—	SINGLE	SELECT	×	×	×
784	SEL441				GBETH	PORT 1	—	—	—	SINGLE	SELECT	×	×	×
785	SEL442				GBETH	PORT 1	—	—	—	SINGLE	SELECT	×	×	×
786	SEL443				GBETH	PORT 1	—	—	—	SINGLE	SELECT	×	×	×

Table 4.6-24 List of Input Events (29/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
787	SEL444				GBETH	PORT1	—	—	—	SINGLE	SELECT	×	×	×
788	SEL445				GBETH	PORT1	—	—	—	SINGLE	SELECT	×	×	×
789	SEL446				GBETH	PORT1	—	—	—	SINGLE	SELECT	×	×	×
790	SEL447				GBETH	PORT1	—	—	—	SINGLE	SELECT	×	×	×
791	SEL448				PCIE	CH0	INT_ALL	Logical OR of all interrupt signals	Level	SINGLE	SELECT	×	×	×
792	SEL449				PCIE	CH0	INTA_RC	INTA reception	Level	SINGLE	SELECT	×	×	×
793	SEL450				PCIE	CH0	INTB_RC	INTB reception	Level	SINGLE	SELECT	×	×	×
794	SEL451				PCIE	CH0	INTC_RC	INTC reception	Level	SINGLE	SELECT	×	×	×
795	SEL452				PCIE	CH0	INTD_RC	INTD reception	Level	SINGLE	SELECT	×	×	×
796	SEL453				PCIE	CH0	INTMSI_RC	MSI reception	Level	SINGLE	SELECT	×	×	×
797	SEL454				PCIE	CH0	INT_LINK_BANDWIDTH	Link width change	Level	SINGLE	SELECT	×	×	×
798	SEL455				PCIE	CH0	INT_LINK_EQUALIZATION_REQUEST	Link equalization request	Level	SINGLE	SELECT	×	×	×
799	SEL456				PCIE	CH0	INT_PM_PME	PM_PME reception	Level	SINGLE	SELECT	×	×	×
800	SEL457				PCIE	CH0	INT_SERR	SERR detection	Level	SINGLE	SELECT	×	×	×
801	SEL458				PCIE	CH0	INT_SERR_COR	Correctable error detection	Level	SINGLE	SELECT	×	×	×
802	SEL459				PCIE	CH0	INT_SERR_NONFATAL	Non-fatal error detection	Level	SINGLE	SELECT	×	×	×
803	SEL460				PCIE	CH0	INT_SERR_FATAL	Fatal error detection	Level	SINGLE	SELECT	×	×	×
804	SEL461				PCIE	CH0	DMA_INT	DMA Event	Level	SINGLE	SELECT	×	×	×
805	SEL462				PCIE	CH0	PCIE_EVT_INT	PCI Event	Level	SINGLE	SELECT	×	×	×
806	SEL463				PCIE	CH0	AXI_ERR_INT	ACI Event	Level	SINGLE	SELECT	×	×	×
807	SEL464				PCIE	CH0	MSG_INT	Message reception	Level	SINGLE	SELECT	×	×	×
808	SEL465				PCIE	CH0	TURN_OFF_EVENT	Turn off message reception	Level	SINGLE	SELECT	×	×	×
809	SEL466				PCIE	CH0	PMU_POWEROFF	L2 own power off notification	Level	SINGLE	SELECT	×	×	×
810	SEL467				PCIE	CH0	D3_EVENT_F0	Non-D0 transition request notification	Level	SINGLE	SELECT	×	×	×
811	SEL468				PCIE	CH0	D3_EVENT_F1	Non-D0 transition request notification	Level	SINGLE	SELECT	×	×	×
812	SEL469				PCIE	CH0	CFG_PMCSR_PME_STATUS_WRITECLEAR_F0	PME status clearing notification	Level	SINGLE	SELECT	×	×	×
813	SEL470				PCIE	CH0	CFG_PMCSR_PME_STATUS_WRITECLEAR_F1	PME status clearing notification	Level	SINGLE	SELECT	×	×	×
814	SEL471				—	—	Reserved	Reserved	—	—	—	—	—	—
815	SEL472				—	—	Reserved	Reserved	—	—	—	—	—	—
816	SEL473				—	—	Reserved	Reserved	—	—	—	—	—	—

Table 4.6-24 List of Input Events (30/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
817	SEL474				—	—	Reserved	Reserved	—	—	—	—	—	—
818	SEL475				—	—	Reserved	Reserved	—	—	—	—	—	—
819	SEL476				—	—	Reserved	Reserved	—	—	—	—	—	—
820	SEL477				—	—	Reserved	Reserved	—	—	—	—	—	—
821	SEL478				—	—	Reserved	Reserved	—	—	—	—	—	—
822	SEL479				—	—	Reserved	Reserved	—	—	—	—	—	—
823	SEL480				—	—	Reserved	Reserved	—	—	—	—	—	—
824	SEL481				—	—	Reserved	Reserved	—	—	—	—	—	—
825	SEL482				—	—	Reserved	Reserved	—	—	—	—	—	—
826	SEL483				—	—	Reserved	Reserved	—	—	—	—	—	—
827	SEL484				—	—	Reserved	Reserved	—	—	—	—	—	—
828	SEL485				—	—	Reserved	Reserved	—	—	—	—	—	—
829	SEL486				—	—	Reserved	Reserved	—	—	—	—	—	—
830	SEL487				—	—	Reserved	Reserved	—	—	—	—	—	—
831	SEL488				—	—	Reserved	Reserved	—	—	—	—	—	—
832	SEL489				—	—	Reserved	Reserved	—	—	—	—	—	—
833	SEL490				—	—	Reserved	Reserved	—	—	—	—	—	—
834	SEL491				—	—	Reserved	Reserved	—	—	—	—	—	—
835	SEL492				—	—	Reserved	Reserved	—	—	—	—	—	—
836	SEL493				—	—	Reserved	Reserved	—	—	—	—	—	—
837	SEL494				CRU	CH0	csi2_link_int	CRU Receive interrupt.	Level	SINGLE	SELECT	×	×	×
838	SEL495				CRU	CH0	image_conv_int	CRU Interrupt for image_conv	Level	SINGLE	SELECT	×	×	×
839	SEL496				CRU	CH0	axi_mst_err_int	CRU AXI master error interrupt	Level	SINGLE	SELECT	×	×	×
840	SEL497				CRU	CH0	cru_vd_addr_wend	CRU Video Data AXI Master Address 0 Write End interrupt	Edge	SINGLE	SELECT	×	×	×
841	SEL498				CRU	CH0	cru_sd_addr_wend	CRU Statistics Data AXI Master Address 0 Write End interrupt	Edge	SINGLE	SELECT	×	×	×
842	SEL499				CRU	CH0	cru_vsd_addr_wend	CRU Video & Statistics Data AXI Master Address 0 Write End interrupt	Level	SINGLE	SELECT	×	×	×
843	SEL500				—	—	Reserved	Reserved	—	—	—	—	—	—
844	SEL501				—	—	Reserved	Reserved	—	—	—	—	—	—
845	SEL502				—	—	Reserved	Reserved	—	—	—	—	—	—
846	SEL503				—	—	Reserved	Reserved	—	—	—	—	—	—

Table 4.6-24 List of Input Events (31/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
847	SEL504				—	—	Reserved	Reserved	—	—	—	—	—	—
848	SEL505				—	—	Reserved	Reserved	—	—	—	—	—	—
849	SEL506				—	—	Reserved	Reserved	—	—	—	—	—	—
850	SEL507				—	—	Reserved	Reserved	—	—	—	—	—	—
851	SEL508				—	—	Reserved	Reserved	—	—	—	—	—	—
852	SEL509				—	—	Reserved	Reserved	—	—	—	—	—	—
853	SEL510				—	—	Reserved	Reserved	—	—	—	—	—	—
854	SEL511				—	—	Reserved	Reserved	—	—	—	—	—	—
855	SEL512				—	—	Reserved	Reserved	—	—	—	—	—	—
856	SEL513				—	—	Reserved	Reserved	—	—	—	—	—	—
857	SEL514				—	—	Reserved	Reserved	—	—	—	—	—	—
858	SEL515				—	—	Reserved	Reserved	—	—	—	—	—	—
859	SEL516				—	—	Reserved	Reserved	—	—	—	—	—	—
860	SEL517				—	—	Reserved	Reserved	—	—	—	—	—	—
861	SEL518				—	—	Reserved	Reserved	—	—	—	—	—	—
862	SEL519				—	—	Reserved	Reserved	—	—	—	—	—	—
863	SEL520				—	—	Reserved	Reserved	—	—	—	—	—	—
864	SEL521				—	—	Reserved	Reserved	—	—	—	—	—	—
865	SEL522				—	—	Reserved	Reserved	—	—	—	—	—	—
866	SEL523				—	—	Reserved	Reserved	—	—	—	—	—	—
867	SEL524		449		—	—	Reserved	Reserved	—	—	—	—	—	—
868	SEL525		450		—	—	Reserved	Reserved	—	—	—	—	—	—
869	SEL526		451		—	—	Reserved	Reserved	—	—	—	—	—	—
870	SEL527		452		—	—	Reserved	Reserved	—	—	—	—	—	—
871	SEL528				—	—	Reserved	Reserved	—	—	—	—	—	—
872	SEL529				—	—	Reserved	Reserved	—	—	—	—	—	—
873	SEL530				—	—	Reserved	Reserved	—	—	—	—	—	—
874	SEL531				DSI	—	dsi_int_sq0	DSI Sequence Operation channel 0 interrupt	Level	SINGLE	SELECT	×	×	×
875	SEL532				DSI	—	dsi_int_sq1	DSI Sequence Operation channel 1 interrupt	Level	SINGLE	SELECT	×	×	×
876	SEL533				DSI	—	dsi_int_vin1	DSI Video-Input Operation channel 1 interrupt	Level	SINGLE	SELECT	×	×	×
877	SEL534				DSI	—	dsi_int_rcv	DSI Packet Receive interrupt	Level	SINGLE	SELECT	×	×	×

Table 4.6-24 List of Input Events (32/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
878	SEL535				DSI	—	dsi_int_ferr	DSI Fatal Error interrupt	Level	SINGLE	SELECT	×	×	×
879	SEL536				DSI	—	dsi_int_ppi	DSI D-PHY PPI related interrupt	Level	SINGLE	SELECT	×	×	×
880	SEL537				DSI	—	dsi_int_debug	Debug interrupt	Level	SINGLE	SELECT	×	×	×
881	SEL538				LCDC	CH0	vs1o_intreq_n	VSPD interrupt	Level	SINGLE	SELECT	×	×	×
882	SEL539				LCDC	CH0	duo_intr_n	DU interrupt	Level	SINGLE	SELECT	×	×	×
883	SEL540				GE3D	—	IRQGPU	GE3D interrupt request	Level	SINGLE	SELECT	×	×	×
884	SEL541				GE3D	—	IRQJOB	Job interrupt request	Level	SINGLE	SELECT	×	×	×
885	SEL542				GE3D	—	IRQMMU	MMU interrupt request	Level	SINGLE	SELECT	×	×	×
886	SEL543				GE3D	—	IRQEVENT	GE3D event request	Level	SINGLE	SELECT	×	×	×
887	SEL544				VCD	—	vpo_vintreq_n	VLC interrupt	Level	SINGLE	SELECT	×	×	×
888	SEL545				VCD	—	vpo_cintreq_n	CE interrupt	Level	SINGLE	SELECT	×	×	×
889	SEL546				SSIU	—	intreq_ssi0_main	Interrupt output (level, active high)	Level	SINGLE	SELECT	×	×	×
890	SEL547				SSIU	—	intreq_ssi1_main	Interrupt output (level, active high)	Level	SINGLE	SELECT	×	×	×
891	SEL548				SSIU	—	intreq_ssi2_main	Interrupt output (level, active high)	Level	SINGLE	SELECT	×	×	×
892	SEL549				SSIU	—	intreq_ssi3_main	Interrupt output (level, active high)	Level	SINGLE	SELECT	×	×	×
893	SEL550				SSIU	—	intreq_ssi4_main	Interrupt output (level, active high)	Level	SINGLE	SELECT	×	×	×
894	SEL551				SSIU	—	intreq_ssi5_main	Interrupt output (level, active high)	Level	SINGLE	SELECT	×	×	×
895	SEL552				SSIU	—	intreq_ssi6_main	Interrupt output (level, active high)	Level	SINGLE	SELECT	×	×	×
896	SEL553				SSIU	—	intreq_ssi7_main	Interrupt output (level, active high)	Level	SINGLE	SELECT	×	×	×
897	SEL554				SSIU	—	intreq_ssi8_main	Interrupt output (level, active high)	Level	SINGLE	SELECT	×	×	×
898	SEL555				SSIU	—	intreq_ssi9_main	Interrupt output (level, active high)	Level	SINGLE	SELECT	×	×	×
899	SEL556				SPDIF	CH0	intreq_spdif_n	Interrupt signal	Level	SINGLE	SELECT	×	×	×
900	SEL557				SPDIF	CH1	intreq_spdif_n	Interrupt signal	Level	SINGLE	SELECT	×	×	×
901	SEL558				SPDIF	CH2	intreq_spdif_n	Interrupt signal	Level	SINGLE	SELECT	×	×	×
902	SEL559				SCU	—	intreq_scu0_arm	Interrupt signal	Level	SINGLE	SELECT	×	×	×
903	SEL560				SCU	—	intreq_scu1_arm	Interrupt signal	Level	SINGLE	SELECT	×	×	×
904	SEL561				SCU	—	intreq_scu2_arm	Interrupt signal	Level	SINGLE	SELECT	×	×	×
905	SEL562				SCU	—	intreq_scu3_arm	Interrupt signal	Level	SINGLE	SELECT	×	×	×
906	SEL563				SCU	—	intreq_scu4_arm	Interrupt signal	Level	SINGLE	SELECT	×	×	×
907	SEL564				SCU	—	intreq_scu5_arm	Interrupt signal	Level	SINGLE	SELECT	×	×	×
908	SEL565				SCU	—	intreq_scu6_arm	Interrupt signal	Level	SINGLE	SELECT	×	×	×

Table 4.6-24 List of Input Events (33/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
909	SEL566				SCU	—	intreq_scu7_arm	Interrupt signal	Level	SINGLE	SELECT	×	×	×
910	SEL567				SCU	—	intreq_scu8_arm	Interrupt signal	Level	SINGLE	SELECT	×	×	×
911	SEL568				SCU	—	intreq_scu9_arm	Interrupt signal	Level	SINGLE	SELECT	×	×	×
912	SEL569				—	—	Reserved	Reserved	—	—	—	—	—	—
913	SEL570				—	—	Reserved	Reserved	—	—	—	—	—	—
914	SEL571				—	—	Reserved	Reserved	—	—	—	—	—	—
915	SEL572				—	—	Reserved	Reserved	—	—	—	—	—	—
916	SEL573				—	—	Reserved	Reserved	—	—	—	—	—	—
917	SEL574				—	—	Reserved	Reserved	—	—	—	—	—	—
918	SEL575	200			ADC	—	ada_adireq_n	Scan end (GrA)	Edge	SINGLE	SELECT	✓	×	×
919	SEL576	201			ADC	—	ada_gbadireq_n	Scan end (GrB)	Edge	SINGLE	SELECT	✓	×	×
920	SEL577	202			ADC	—	ada_gcadireq_n	Scan end (GrC)	Edge	SINGLE	SELECT	✓	×	×
921	SEL578				LCDC	CH1	vs1o_intreq_n	VSPD interrupt	Level	SINGLE	SELECT	×	×	×
922	SEL579				LCDC	CH1	duo_intr_n	DU interrupt	Level	SINGLE	SELECT	×	×	×
923	SEL580				VSPI	—	intreq_vspi0_n	VSPI interrupt	Level	SINGLE	SELECT	×	×	×
924	SEL581				EThosU55	—	IRQ	Interrupt	Level	SINGLE	SELECT	×	×	×
925	SEL582				FDP1	—	intreq_fdp1_n	FDP1 interrupt	Level	SINGLE	SELECT	×	×	×
				32	WDT	OTHER_0	WDT_OTHER_0_iwdt_nmiundf_n	Down-counter underflow/refresh error	Edge	—	—	×	×	✓
				33	WDT	OTHER_1	WDT_OTHER_1_iwdt_nmiundf_n	Down-counter underflow/refresh error	Edge	—	—	×	×	✓
	262				ICU	SWINT	INT_CM33_0	Occurs by writing to the ICU_TOP register (for CPU Interrupt)	Edge	—	SYSTEM	×	×	×
	263				ICU	SWINT	INT_CM33_1	Occurs by writing to the ICU_TOP register (for CPU Interrupt)	Edge	—	SYSTEM	×	×	×
	264				ICU	SWINT	INT_CM33_2	Occurs by writing to the ICU_TOP register (for CPU Interrupt)	Edge	—	SYSTEM	×	×	×
	265				ICU	SWINT	INT_CM33_3	Occurs by writing to the ICU_TOP register (for CPU Interrupt)	Edge	—	SYSTEM	×	×	×
	266				ICU	ICUERR	ICU_Error_CM33	Signal with error interrupts combined into one in ICU_TOP	Level	—	SYSTEM	×	×	×
	267				CM33	—	CTIIRQ_0	CTI IRQ Request	Level	—	SYSTEM	×	×	×
	268				CM33	—	CTIIRQ_1	CTI IRQ Request	Level	—	SYSTEM	×	×	×
	269				CM33	—	FPIXC	Inexact	Level	—	SYSTEM	×	×	×
	270				CM33	—	FPIDC	Input denormal	Level	—	SYSTEM	×	×	×
	271				CM33	—	FPOFC	Overflow	Level	—	SYSTEM	×	×	×

Table 4.6-24 List of Input Events (34/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
	272				CM33	—	FPUFC	Underflow	Level	—	SYSTEM	×	×	×
	273				CM33	—	FPDZC	Divide-by-zero	Level	—	SYSTEM	×	×	×
	274				CM33	—	FPIOC	Invalid operation	Level	—	SYSTEM	×	×	×
	275				CM33	—	ERRINT_BRG_CM33_C		Edge	—	SYSTEM	×	×	×
	276				CM33	—	ERRINT_BRG_CM33_S		Edge	—	SYSTEM	×	×	×
	281				MHU	—	msg_ch5_s	Secure message interrupt ch5	Level	—	SYSTEM	×	×	×
	282				MHU	—	msg_ch11_s	Secure message interrupt ch11	Level	—	SYSTEM	×	×	×
	283				MHU	—	msg_ch17_s	Secure message interrupt ch17	Level	—	SYSTEM	×	×	×
	284				MHU	—	msg_ch23_s	Secure message interrupt ch23	Level	—	SYSTEM	×	×	×
	285				MHU	—	msg_ch29_s	Secure message interrupt ch29 #Reserved	Level	—	SYSTEM	×	×	×
	286				MHU	—	msg_ch35_s	Secure message interrupt ch35 #Reserved	Level	—	SYSTEM	×	×	×
	287				MHU	—	rsp_ch5_s	Secure response interrupt ch5	Level	—	SYSTEM	×	×	×
	288				MHU	—	rsp_ch11_s	Secure response interrupt ch11	Level	—	SYSTEM	×	×	×
	289				MHU	—	rsp_ch17_s	Secure response interrupt ch17	Level	—	SYSTEM	×	×	×
	290				MHU	—	rsp_ch23_s	Secure response interrupt ch23	Level	—	SYSTEM	×	×	×
	291				MHU	—	rsp_ch29_s	Secure response interrupt ch29 #Reserved	Level	—	SYSTEM	×	×	×
	292				MHU	—	rsp_ch35_s	Secure response interrupt ch35 #Reserved	Level	—	SYSTEM	×	×	×
	293				MHU	—	msg_ch5_ns	Non-secure message interrupt ch5	Level	—	SYSTEM	×	×	×
	294				MHU	—	msg_ch11_ns	Non-secure message interrupt ch11	Level	—	SYSTEM	×	×	×
	295				MHU	—	msg_ch17_ns	Non-secure message interrupt ch17	Level	—	SYSTEM	×	×	×
	296				MHU	—	msg_ch23_ns	Non-secure message interrupt ch23	Level	—	SYSTEM	×	×	×
	297				MHU	—	msg_ch29_ns	Non-secure message interrupt ch29 #Reserved	Level	—	SYSTEM	×	×	×
	298				MHU	—	msg_ch35_ns	Non-secure message interrupt ch35 #Reserved	Level	—	SYSTEM	×	×	×
	299				MHU	—	rsp_ch5_ns	Non-secure response interrupt ch5	Level	—	SYSTEM	×	×	×
	300				MHU	—	rsp_ch11_ns	Non-secure response interrupt ch11	Level	—	SYSTEM	×	×	×
	301				MHU	—	rsp_ch17_ns	Non-secure response interrupt ch17	Level	—	SYSTEM	×	×	×
	302				MHU	—	rsp_ch23_ns	Non-secure response interrupt ch23	Level	—	SYSTEM	×	×	×
	303				MHU	—	rsp_ch29_ns	Non-secure response interrupt ch29 #Reserved	Level	—	SYSTEM	×	×	×
	304				MHU	—	rsp_ch35_ns	Non-secure response interrupt ch35 #Reserved	Level	—	SYSTEM	×	×	×
	305				MHU	—	swint_ch2_ns	Software interrupt 2ch-CA55Core0_to_CM33	Level	—	SYSTEM	×	×	×
	306				MHU	—	swint_ch5_ns	Software interrupt 5ch-CA55Core1_to_CM33	Level	—	SYSTEM	×	×	×

Table 4.6-24 List of Input Events (35/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
	307				MHU	—	swint_ch8_ns	Software interrupt 8ch-CA55Core2_to_CM33	Level	—	SYSTEM	×	×	×
	308				MHU	—	swint_ch11_ns	Software interrupt 11ch-CA55Core3_to_CM33	Level	—	SYSTEM	×	×	×
	309				MHU	—	swint_ch16_ns	Software interrupt 16ch-#Reserved	Level	—	SYSTEM	×	×	×
	310				MHU	—	swint_ch21_ns	Software interrupt 21ch-#Reserved	Level	—	SYSTEM	×	×	×
		0			PFC	—	DREQ0	External DMA request	Low-level detection/rising edge/falling edge/both edges	—	—	✓	×	×
		1			PFC	—	DREQ1	External DMA request	Low-level detection/rising edge/falling edge/both edges	—	—	✓	×	×
		2			PFC	—	DREQ2	External DMA request	Low-level detection/rising edge/falling edge/both edges	—	—	✓	×	×
		3			PFC	—	DREQ3	External DMA request	Low-level detection/rising edge/falling edge/both edges	—	—	✓	×	×
		4			PFC	—	DREQ4	External DMA request	Low-level detection/rising edge/falling edge/both edges	—	—	✓	×	×
		12			ICU	EVT_DMAC	EVT_DMAC_0	This signal that uses the Event Input as a DMAC REQ	Edge	—	—	✓	×	×
		13			ICU	EVT_DMAC	EVT_DMAC_1	This signal that uses the Event Input as a DMAC REQ	Edge	—	—	✓	×	×
		14			ICU	EVT_DMAC	EVT_DMAC_2	This signal that uses the Event Input as a DMAC REQ	Edge	—	—	✓	×	×
		15			ICU	EVT_DMAC	EVT_DMAC_3	This signal that uses the Event Input as a DMAC REQ	Edge	—	—	✓	×	×
		16			ICU	EVT_DMAC	EVT_DMAC_4	This signal that uses the Event Input as a DMAC REQ	Edge	—	—	✓	×	×
		17			ICU	EVT_DMAC	EVT_DMAC_5	This signal that uses the Event Input as a DMAC REQ	Edge	—	—	✓	×	×
		18			ICU	EVT_DMAC	EVT_DMAC_6	This signal that uses the Event Input as a DMAC REQ	Edge	—	—	✓	×	×
		19			ICU	EVT_DMAC	EVT_DMAC_7	This signal that uses the Event Input as a DMAC REQ	Edge	—	—	✓	×	×
		20			ICU	SWDMAC	DMAC_SW_Trigger_1_0	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		21			ICU	SWDMAC	DMAC_SW_Trigger_1_1	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		22			ICU	SWDMAC	DMAC_SW_Trigger_1_2	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		23			ICU	SWDMAC	DMAC_SW_Trigger_1_3	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		24			ICU	SWDMAC	DMAC_SW_Trigger_1_4	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		25			ICU	SWDMAC	DMAC_SW_Trigger_1_5	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×

Table 4.6-24 List of Input Events (36/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
		26			ICU	SWD MAC	DMAC_SW_Trigger_1_6	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		27			ICU	SWD MAC	DMAC_SW_Trigger_1_7	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		28			ICU	SWD MAC	DMAC_SW_Trigger_1_8	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		29			ICU	SWD MAC	DMAC_SW_Trigger_1_9	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		30			ICU	SWD MAC	DMAC_SW_Trigger_1_10	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		31			ICU	SWD MAC	DMAC_SW_Trigger_1_11	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		32			ICU	SWD MAC	DMAC_SW_Trigger_1_12	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		33			ICU	SWD MAC	DMAC_SW_Trigger_1_13	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		34			ICU	SWD MAC	DMAC_SW_Trigger_1_14	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		35			ICU	SWD MAC	DMAC_SW_Trigger_1_15	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		36			ICU	SWD MAC	DMAC_SW_Trigger_2_0	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		37			ICU	SWD MAC	DMAC_SW_Trigger_2_1	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		38			ICU	SWD MAC	DMAC_SW_Trigger_2_2	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		39			ICU	SWD MAC	DMAC_SW_Trigger_2_3	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		40			ICU	SWD MAC	DMAC_SW_Trigger_2_4	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		41			ICU	SWD MAC	DMAC_SW_Trigger_2_5	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		42			ICU	SWD MAC	DMAC_SW_Trigger_2_6	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		43			ICU	SWD MAC	DMAC_SW_Trigger_2_7	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		44			ICU	SWD MAC	DMAC_SW_Trigger_2_8	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		45			ICU	SWD MAC	DMAC_SW_Trigger_2_9	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		46			ICU	SWD MAC	DMAC_SW_Trigger_2_10	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		47			ICU	SWD MAC	DMAC_SW_Trigger_2_11	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×

Table 4.6-24 List of Input Events (37/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
		48			ICU	SWD MAC	DMAC_SW_Trigger_2_12	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		49			ICU	SWD MAC	DMAC_SW_Trigger_2_13	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		50			ICU	SWD MAC	DMAC_SW_Trigger_2_14	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		51			ICU	SWD MAC	DMAC_SW_Trigger_2_15	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		52			ICU	SWD MAC	DMAC_SW_Trigger_3_0	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		53			ICU	SWD MAC	DMAC_SW_Trigger_3_1	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		54			ICU	SWD MAC	DMAC_SW_Trigger_3_2	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		55			ICU	SWD MAC	DMAC_SW_Trigger_3_3	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		56			ICU	SWD MAC	DMAC_SW_Trigger_3_4	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		57			ICU	SWD MAC	DMAC_SW_Trigger_3_5	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		58			ICU	SWD MAC	DMAC_SW_Trigger_3_6	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		59			ICU	SWD MAC	DMAC_SW_Trigger_3_7	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		60			ICU	SWD MAC	DMAC_SW_Trigger_3_8	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		61			ICU	SWD MAC	DMAC_SW_Trigger_3_9	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		62			ICU	SWD MAC	DMAC_SW_Trigger_3_10	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		63			ICU	SWD MAC	DMAC_SW_Trigger_3_11	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		64			ICU	SWD MAC	DMAC_SW_Trigger_3_12	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		65			ICU	SWD MAC	DMAC_SW_Trigger_3_13	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		66			ICU	SWD MAC	DMAC_SW_Trigger_3_14	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		67			ICU	SWD MAC	DMAC_SW_Trigger_3_15	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		68			ICU	SWD MAC	DMAC_SW_Trigger_4_0	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		69			ICU	SWD MAC	DMAC_SW_Trigger_4_1	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×

Table 4.6-24 List of Input Events (38/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
		70			ICU	SWD MAC	DMAC_SW_Trigger_4_2	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		71			ICU	SWD MAC	DMAC_SW_Trigger_4_3	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		72			ICU	SWD MAC	DMAC_SW_Trigger_4_4	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		73			ICU	SWD MAC	DMAC_SW_Trigger_4_5	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		74			ICU	SWD MAC	DMAC_SW_Trigger_4_6	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		75			ICU	SWD MAC	DMAC_SW_Trigger_4_7	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		76			ICU	SWD MAC	DMAC_SW_Trigger_4_8	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		77			ICU	SWD MAC	DMAC_SW_Trigger_4_9	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		78			ICU	SWD MAC	DMAC_SW_Trigger_4_10	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		79			ICU	SWD MAC	DMAC_SW_Trigger_4_11	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		80			ICU	SWD MAC	DMAC_SW_Trigger_4_12	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		81			ICU	SWD MAC	DMAC_SW_Trigger_4_13	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		82			ICU	SWD MAC	DMAC_SW_Trigger_4_14	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		83			ICU	SWD MAC	DMAC_SW_Trigger_4_15	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		84			ICU	SWD MAC	DMAC_SW_Trigger_0_0	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		85			ICU	SWD MAC	DMAC_SW_Trigger_0_1	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		86			ICU	SWD MAC	DMAC_SW_Trigger_0_2	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		87			ICU	SWD MAC	DMAC_SW_Trigger_0_3	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		88			ICU	SWD MAC	DMAC_SW_Trigger_0_4	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		89			ICU	SWD MAC	DMAC_SW_Trigger_0_5	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		90			ICU	SWD MAC	DMAC_SW_Trigger_0_6	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		91			ICU	SWD MAC	DMAC_SW_Trigger_0_7	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×

Table 4.6-24 List of Input Events (39/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
		92			ICU	SWD MAC	DMAC_SW_Trigger_0_8	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		93			ICU	SWD MAC	DMAC_SW_Trigger_0_9	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		94			ICU	SWD MAC	DMAC_SW_Trigger_0_10	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		95			ICU	SWD MAC	DMAC_SW_Trigger_0_11	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		96			ICU	SWD MAC	DMAC_SW_Trigger_0_12	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		97			ICU	SWD MAC	DMAC_SW_Trigger_0_13	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		98			ICU	SWD MAC	DMAC_SW_Trigger_0_14	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		99			ICU	SWD MAC	DMAC_SW_Trigger_0_15	Occurs by writing to the ICU_TOP register (for DMAC REQ)	Edge	—	—	✓	×	×
		203	257		ADC	—	ada_elccondmtch	Compare match (event of a match with the condition for comparison) (for ELC)	Edge	—	—	✓	✓	×
		204	258		ADC	—	ada_elccondunmtch	Compare mis-match (event of a non-match with the condition for comparison) (for ELC)	Edge	—	—	✓	✓	×
		339			CANFD	—	can_rf_dmareq_0	This is the DMA transfer request pulse signal for the DMAC of the RX-FIFOs	Edge	—	—	✓	×	×
		340			CANFD	—	can_rf_dmareq_1	This is the DMA transfer request pulse signal for the DMAC of the RX-FIFOs	Edge	—	—	✓	×	×
		341			CANFD	—	can_rf_dmareq_2	This is the DMA transfer request pulse signal for the DMAC of the RX-FIFOs	Edge	—	—	✓	×	×
		342			CANFD	—	can_rf_dmareq_3	This is the DMA transfer request pulse signal for the DMAC of the RX-FIFOs	Edge	—	—	✓	×	×
		343			CANFD	—	can_rf_dmareq_4	This is the DMA transfer request pulse signal for the DMAC of the RX-FIFOs	Edge	—	—	✓	×	×
		344			CANFD	—	can_rf_dmareq_5	This is the DMA transfer request pulse signal for the DMAC of the RX-FIFOs	Edge	—	—	✓	×	×
		345			CANFD	—	can_rf_dmareq_6	This is the DMA transfer request pulse signal for the DMAC of the RX-FIFOs	Edge	—	—	✓	×	×
		346			CANFD	—	can_rf_dmareq_7	This is the DMA transfer request pulse signal for the DMAC of the RX-FIFOs	Edge	—	—	✓	×	×
		347			CANFD	—	can_cf_dmareq_0	This is the DMA transfer request pulse signal for the DMAC of the first COM-FIFO of each channel	Edge	—	—	✓	×	×
		348			CANFD	—	can_cf_dmareq_1	This is the DMA transfer request pulse signal for the DMAC of the first COM-FIFO of each channel	Edge	—	—	✓	×	×
		349			CANFD	—	can_cf_dmareq_2	This is the DMA transfer request pulse signal for the DMAC of the first COM-FIFO of each channel	Edge	—	—	✓	×	×

Table 4.6-24 List of Input Events (40/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
		350			CANFD	—	can_cf_dmareq_3	This is the DMA transfer request pulse signal for the DMAC of the first COM-FIFO of each channel	Edge	—	—	✓	×	×
		351			CANFD	—	can_cf_dmareq_4	This is the DMA transfer request pulse signal for the DMAC of the first COM-FIFO of each channel	Edge	—	—	✓	×	×
		352			CANFD	—	can_cf_dmareq_5	This is the DMA transfer request pulse signal for the DMAC of the first COM-FIFO of each channel	Edge	—	—	✓	×	×
		353			SSIU	—	ssip00_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		354			SSIU	—	ssip00_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		355			SSIU	—	ssip01_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		356			SSIU	—	ssip01_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		357			SSIU	—	ssip02_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		358			SSIU	—	ssip02_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		359			SSIU	—	ssip03_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		360			SSIU	—	ssip03_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		361			SSIU	—	ssip10_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		362			SSIU	—	ssip10_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		363			SSIU	—	ssip11_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		364			SSIU	—	ssip11_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		365			SSIU	—	ssip12_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		366			SSIU	—	ssip12_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		367			SSIU	—	ssip13_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		368			SSIU	—	ssip13_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		369			SSIU	—	ssip20_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		370			SSIU	—	ssip20_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		371			SSIU	—	ssip21_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		372			SSIU	—	ssip21_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		373			SSIU	—	ssip22_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		374			SSIU	—	ssip22_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		375			SSIU	—	ssip23_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		376			SSIU	—	ssip23_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		377			SSIU	—	ssip30_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		378			SSIU	—	ssip30_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×

Table 4.6-24 List of Input Events (41/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
		379			SSIU	—	ssip31_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		380			SSIU	—	ssip31_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		381			SSIU	—	ssip32_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		382			SSIU	—	ssip32_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		383			SSIU	—	ssip33_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		384			SSIU	—	ssip33_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		385			SSIU	—	ssip40_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		386			SSIU	—	ssip40_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		387			SSIU	—	ssip41_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		388			SSIU	—	ssip41_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		389			SSIU	—	ssip42_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		390			SSIU	—	ssip42_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		391			SSIU	—	ssip43_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		392			SSIU	—	ssip43_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		393			SSIU	—	ssip5_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		394			SSIU	—	ssip5_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		395			SSIU	—	ssip6_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		396			SSIU	—	ssip6_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		397			SSIU	—	ssip7_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		398			SSIU	—	ssip7_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		399			SSIU	—	ssip8_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		400			SSIU	—	ssip8_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		401			SSIU	—	ssip90_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		402			SSIU	—	ssip90_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		403			SSIU	—	ssip91_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		404			SSIU	—	ssip91_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		405			SSIU	—	ssip92_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		406			SSIU	—	ssip92_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		407			SSIU	—	ssip93_dreq_rx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		408			SSIU	—	ssip93_dreq_tx	General-purpose DMAC REQ signal	Level	—	—	✓	×	×
		409			SPDIF	CH0	rbdmareqn_tx	Request for DMA transfer	Level	—	—	✓	×	×
		410			SPDIF	CH0	rbdmareqn_rx	Request for DMA transfer	Level	—	—	✓	×	×

Table 4.6-24 List of Input Events (42/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
		411			SPDIF	CH1	rbdmareqn_tx	Request for DMA transfer	Level	—	—	✓	×	×
		412			SPDIF	CH1	rbdmareqn_rx	Request for DMA transfer	Level	—	—	✓	×	×
		413			SPDIF	CH2	rbdmareqn_tx	Request for DMA transfer	Level	—	—	✓	×	×
		414			SPDIF	CH2	rbdmareqn_rx	Request for DMA transfer	Level	—	—	✓	×	×
		415			SCU	—	pdreq_sr0_in	Request for DMA transfer	Level	—	—	✓	×	×
		416			SCU	—	pdreq_sr1_in	Request for DMA transfer	Level	—	—	✓	×	×
		417			SCU	—	pdreq_sr2_in	Request for DMA transfer	Level	—	—	✓	×	×
		418			SCU	—	pdreq_sr3_in	Request for DMA transfer	Level	—	—	✓	×	×
		419			SCU	—	pdreq_sr4_in	Request for DMA transfer	Level	—	—	✓	×	×
		420			SCU	—	pdreq_sr5_in	Request for DMA transfer	Level	—	—	✓	×	×
		421			SCU	—	pdreq_sr6_in	Request for DMA transfer	Level	—	—	✓	×	×
		422			SCU	—	pdreq_sr7_in	Request for DMA transfer	Level	—	—	✓	×	×
		423			SCU	—	pdreq_sr8_in	Request for DMA transfer	Level	—	—	✓	×	×
		424			SCU	—	pdreq_sr9_in	Request for DMA transfer	Level	—	—	✓	×	×
		425			SCU	—	pdreq_sr0_out	Request for DMA transfer	Level	—	—	✓	×	×
		426			SCU	—	pdreq_sr1_out	Request for DMA transfer	Level	—	—	✓	×	×
		427			SCU	—	pdreq_sr2_out	Request for DMA transfer	Level	—	—	✓	×	×
		428			SCU	—	pdreq_sr3_out	Request for DMA transfer	Level	—	—	✓	×	×
		429			SCU	—	pdreq_sr4_out	Request for DMA transfer	Level	—	—	✓	×	×
		430			SCU	—	pdreq_sr5_out	Request for DMA transfer	Level	—	—	✓	×	×
		431			SCU	—	pdreq_sr6_out	Request for DMA transfer	Level	—	—	✓	×	×
		432			SCU	—	pdreq_sr7_out	Request for DMA transfer	Level	—	—	✓	×	×
		433			SCU	—	pdreq_sr8_out	Request for DMA transfer	Level	—	—	✓	×	×
		434			SCU	—	pdreq_sr9_out	Request for DMA transfer	Level	—	—	✓	×	×
		435			SCU	—	pdreq_cmd0_out	Request for DMA transfer	Level	—	—	✓	×	×
		436			SCU	—	pdreq_cmd1_out	Request for DMA transfer	Level	—	—	✓	×	×
			16		PFC	—	IOPORT_GROUP0	Input edge detection of input port group 1	Edge	—	—	×	✓	×
			17		PFC	—	IOPORT_GROUP1	Input edge detection of input port group 2	Edge	—	—	×	✓	×
			18		PFC	—	IOPORT_SINGLE0	Input edge detection of single input port 0	Edge	—	—	×	✓	×
			19		PFC	—	IOPORT_SINGLE1	Input edge detection of single input port 1	Edge	—	—	×	✓	×
			20		PFC	—	IOPORT_SINGLE2	Input edge detection of single input port 2	Edge	—	—	×	✓	×
			21		PFC	—	IOPORT_SINGLE3	Input edge detection of single input port 3	Edge	—	—	×	✓	×
			22		ICU	SWEVT	SW_EVENT_0	Occurs by writing to the ICU_TOP register (for Event)	Edge	—	—	×	✓	×
			23		ICU	SWEVT	SW_EVENT_1	Occurs by writing to the ICU_TOP register (for Event)	Edge	—	—	×	✓	×
			24		ICU	SWEVT	SW_EVENT_2	Occurs by writing to the ICU_TOP register (for Event)	Edge	—	—	×	✓	×
			25		ICU	SWEVT	SW_EVENT_3	Occurs by writing to the ICU_TOP register (for Event)	Edge	—	—	×	✓	×
			26		ICU	SWEVT	SW_EVENT_4	Occurs by writing to the ICU_TOP register (for Event)	Edge	—	—	×	✓	×
			27		ICU	SWEVT	SW_EVENT_5	Occurs by writing to the ICU_TOP register (for Event)	Edge	—	—	×	✓	×
			28		ICU	SWEVT	SW_EVENT_6	Occurs by writing to the ICU_TOP register (for Event)	Edge	—	—	×	✓	×

Table 4.6-24 List of Input Events (43/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
			29		CMTW	CH0	CMTW_CH0_cmt2_elccmp	Compare match event	Edge	—	—	×	✓	×
			30		CMTW	CH1	CMTW_CH1_cmt2_elccmp	Compare match event	Edge	—	—	×	✓	×
			31		CMTW	CH2	CMTW_CH2_cmt2_elccmp	Compare match event	Edge	—	—	×	✓	×
			32		CMTW	CH3	CMTW_CH3_cmt2_elccmp	Compare match event	Edge	—	—	×	✓	×
			33		CMTW	CH4	CMTW_CH4_cmt2_elccmp	Compare match event	Edge	—	—	×	✓	×
			34		CMTW	CH5	CMTW_CH5_cmt2_elccmp	Compare match event	Edge	—	—	×	✓	×
			35		CMTW	CH6	CMTW_CH6_cmt2_elccmp	Compare match event	Edge	—	—	×	✓	×
			36		CMTW	CH7	CMTW_CH7_cmt2_elccmp	Compare match event	Edge	—	—	×	✓	×
			37		CMTW	CH0	CMTW_CH0_cmt2_elcocp0	Output compare 0 event	Edge	—	—	×	✓	×
			38		CMTW	CH1	CMTW_CH1_cmt2_elcocp0	Output compare 0 event	Edge	—	—	×	✓	×
			39		CMTW	CH2	CMTW_CH2_cmt2_elcocp0	Output compare 0 event	Edge	—	—	×	✓	×
			40		CMTW	CH3	CMTW_CH3_cmt2_elcocp0	Output compare 0 event	Edge	—	—	×	✓	×
			41		CMTW	CH4	CMTW_CH4_cmt2_elcocp0	Output compare 0 event	Edge	—	—	×	✓	×
			42		CMTW	CH5	CMTW_CH5_cmt2_elcocp0	Output compare 0 event	Edge	—	—	×	✓	×
			43		CMTW	CH6	CMTW_CH6_cmt2_elcocp0	Output compare 0 event	Edge	—	—	×	✓	×
			44		CMTW	CH7	CMTW_CH7_cmt2_elcocp0	Output compare 0 event	Edge	—	—	×	✓	×
			45		CMTW	CH0	CMTW_CH0_cmt2_elcocp1	Output compare 1 event	Edge	—	—	×	✓	×
			46		CMTW	CH1	CMTW_CH1_cmt2_elcocp1	Output compare 1 event	Edge	—	—	×	✓	×
			47		CMTW	CH2	CMTW_CH2_cmt2_elcocp1	Output compare 1 event	Edge	—	—	×	✓	×
			48		CMTW	CH3	CMTW_CH3_cmt2_elcocp1	Output compare 1 event	Edge	—	—	×	✓	×
			49		CMTW	CH4	CMTW_CH4_cmt2_elcocp1	Output compare 1 event	Edge	—	—	×	✓	×
			50		CMTW	CH5	CMTW_CH5_cmt2_elcocp1	Output compare 1 event	Edge	—	—	×	✓	×
			51		CMTW	CH6	CMTW_CH6_cmt2_elcocp1	Output compare 1 event	Edge	—	—	×	✓	×
			52		CMTW	CH7	CMTW_CH7_cmt2_elcocp1	Output compare 1 event	Edge	—	—	×	✓	×
			141		WDT	CA55	iwdt_elcwun	underflow or refresh error event	Edge	—	—	×	✓	×
			142		WDT	CM33	iwdt_elcwun	underflow or refresh error event	Edge	—	—	×	✓	×
			143		WDT	OTHER_0	iwdt_elcwun	underflow or refresh error event	Edge	—	—	×	✓	×
			144		WDT	OTHER_1	iwdt_elcwun	underflow or refresh error event	Edge	—	—	×	✓	×
			145		RSPI	CH0	sp_elcrdrf	Receive buffer full event	Edge	—	—	×	✓	×
			146		RSPI	CH0	sp_elctdre	Transmit buffer empty event	Edge	—	—	×	✓	×
			147		RSPI	CH0	sp_elcerr	Error event (mode fault/underrun/overrun/parity error)	Edge	—	—	×	✓	×
			148		RSPI	CH0	sp_elcid	Idle event	Edge	—	—	×	✓	×

Table 4.6-24 List of Input Events (44/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
			149		RSPI	CH0	sp_elccend	Communications end event	Edge	—	—	×	✓	×
			150		RSPI	CH1	sp_elcrdrf	Receive buffer full event	Edge	—	—	×	✓	×
			151		RSPI	CH1	sp_elctdre	Transmit buffer empty event	Edge	—	—	×	✓	×
			152		RSPI	CH1	sp_elcerr	Error event (mode fault/underrun/overrun/parity error)	Edge	—	—	×	✓	×
			153		RSPI	CH1	sp_elcid	Idle event	Edge	—	—	×	✓	×
			154		RSPI	CH1	sp_elccend	Communications end event	Edge	—	—	×	✓	×
			155		RSPI	CH2	sp_elcrdrf	Receive buffer full event	Edge	—	—	×	✓	×
			156		RSPI	CH2	sp_elctdre	Transmit buffer empty event	Edge	—	—	×	✓	×
			157		RSPI	CH2	sp_elcerr	Error event (mode fault/underrun/overrun/parity error)	Edge	—	—	×	✓	×
			158		RSPI	CH2	sp_elcid	Idle event	Edge	—	—	×	✓	×
			159		RSPI	CH2	sp_elccend	Communications end event	Edge	—	—	×	✓	×
			160		RSCI	CH0	sc_elcer	Receive error event	Edge	—	—	×	✓	×
			161		RSCI	CH0	sc_elcrdrf	Receive data full event	Edge	—	—	×	✓	×
			162		RSCI	CH0	sc_elcdcmf	Receive data compare match event	Edge	—	—	×	✓	×
			163		RSCI	CH0	sc_elctdre	Transmit data empty event	Edge	—	—	×	✓	×
			164		RSCI	CH0	sc_elctend	Transmit end event	Edge	—	—	×	✓	×
			165		RSCI	CH0	sc_elcdcuf	Receive data compare non-match event	Edge	—	—	×	✓	×
			166		RSCI	CH1	sc_elcer	Receive error event	Edge	—	—	×	✓	×
			167		RSCI	CH1	sc_elcrdrf	Receive data full event	Edge	—	—	×	✓	×
			168		RSCI	CH1	sc_elcdcmf	Receive data compare match event	Edge	—	—	×	✓	×
			169		RSCI	CH1	sc_elctdre	Transmit data empty event	Edge	—	—	×	✓	×
			170		RSCI	CH1	sc_elctend	Transmit end event	Edge	—	—	×	✓	×
			171		RSCI	CH1	sc_elcdcuf	Receive data compare non-match event	Edge	—	—	×	✓	×
			172		RSCI	CH2	sc_elcer	Receive error event	Edge	—	—	×	✓	×
			173		RSCI	CH2	sc_elcrdrf	Receive data full event	Edge	—	—	×	✓	×
			174		RSCI	CH2	sc_elcdcmf	Receive data compare match event	Edge	—	—	×	✓	×
			175		RSCI	CH2	sc_elctdre	Transmit data empty event	Edge	—	—	×	✓	×
			176		RSCI	CH2	sc_elctend	Transmit end event	Edge	—	—	×	✓	×
			177		RSCI	CH2	sc_elcdcuf	Receive data compare non-match event	Edge	—	—	×	✓	×
			178		RSCI	CH3	sc_elcer	Receive error event	Edge	—	—	×	✓	×
			179		RSCI	CH3	sc_elcrdrf	Receive data full event	Edge	—	—	×	✓	×
			180		RSCI	CH3	sc_elcdcmf	Receive data compare match event	Edge	—	—	×	✓	×
			181		RSCI	CH3	sc_elctdre	Transmit data empty event	Edge	—	—	×	✓	×
			182		RSCI	CH3	sc_elctend	Transmit end event	Edge	—	—	×	✓	×
			183		RSCI	CH3	sc_elcdcuf	Receive data compare non-match event	Edge	—	—	×	✓	×
			184		RSCI	CH4	sc_elcer	Receive error event	Edge	—	—	×	✓	×
			185		RSCI	CH4	sc_elcrdrf	Receive data full event	Edge	—	—	×	✓	×
			186		RSCI	CH4	sc_elcdcmf	Receive data compare match event	Edge	—	—	×	✓	×
			187		RSCI	CH4	sc_elctdre	Transmit data empty event	Edge	—	—	×	✓	×
			188		RSCI	CH4	sc_elctend	Transmit end event	Edge	—	—	×	✓	×
			189		RSCI	CH4	sc_elcdcuf	Receive data compare non-match event	Edge	—	—	×	✓	×
			190		RSCI	CH5	sc_elcer	Receive error event	Edge	—	—	×	✓	×

Table 4.6-24 List of Input Events (45/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
			191		RSCI	CH5	sc_elcrdrf	Receive data full event	Edge	—	—	×	✓	×
			192		RSCI	CH5	sc_elcdcmf	Receive data compare match event	Edge	—	—	×	✓	×
			193		RSCI	CH5	sc_elctdre	Transmit data empty event	Edge	—	—	×	✓	×
			194		RSCI	CH5	sc_elctend	Transmit end event	Edge	—	—	×	✓	×
			195		RSCI	CH5	sc_elcdcuf	Receive data compare non-match event	Edge	—	—	×	✓	×
			196		RSCI	CH6	sc_elcer	Receive error event	Edge	—	—	×	✓	×
			197		RSCI	CH6	sc_elcrdrf	Receive data full event	Edge	—	—	×	✓	×
			198		RSCI	CH6	sc_elcdcmf	Receive data compare match event	Edge	—	—	×	✓	×
			199		RSCI	CH6	sc_elctdre	Transmit data empty event	Edge	—	—	×	✓	×
			200		RSCI	CH6	sc_elctend	Transmit end event	Edge	—	—	×	✓	×
			201		RSCI	CH6	sc_elcdcuf	Receive data compare non-match event	Edge	—	—	×	✓	×
			202		RSCI	CH7	sc_elcer	Receive error event	Edge	—	—	×	✓	×
			203		RSCI	CH7	sc_elcrdrf	Receive data full event	Edge	—	—	×	✓	×
			204		RSCI	CH7	sc_elcdcmf	Receive data compare match event	Edge	—	—	×	✓	×
			205		RSCI	CH7	sc_elctdre	Transmit data empty event	Edge	—	—	×	✓	×
			206		RSCI	CH7	sc_elctend	Transmit end event	Edge	—	—	×	✓	×
			207		RSCI	CH7	sc_elcdcuf	Receive data compare non-match event	Edge	—	—	×	✓	×
			208		RSCI	CH8	sc_elcer	Receive error event	Edge	—	—	×	✓	×
			209		RSCI	CH8	sc_elcrdrf	Receive data full event	Edge	—	—	×	✓	×
			210		RSCI	CH8	sc_elcdcmf	Receive data compare match event	Edge	—	—	×	✓	×
			211		RSCI	CH8	sc_elctdre	Transmit data empty event	Edge	—	—	×	✓	×
			212		RSCI	CH8	sc_elctend	Transmit end event	Edge	—	—	×	✓	×
			213		RSCI	CH8	sc_elcdcuf	Receive data compare non-match event	Edge	—	—	×	✓	×
			214		RSCI	CH9	sc_elcer	Receive error event	Edge	—	—	×	✓	×
			215		RSCI	CH9	sc_elcrdrf	Receive data full event	Edge	—	—	×	✓	×
			216		RSCI	CH9	sc_elcdcmf	Receive data compare match event	Edge	—	—	×	✓	×
			217		RSCI	CH9	sc_elctdre	Transmit data empty event	Edge	—	—	×	✓	×
			218		RSCI	CH9	sc_elctend	Transmit end event	Edge	—	—	×	✓	×
			219		RSCI	CH9	sc_elcdcuf	Receive data compare non-match event	Edge	—	—	×	✓	×
			220		RIIC	CH0	RIIC_CH0_iic_elcerrp	ERRP ELC signal	Edge	—	—	×	✓	×
			221		RIIC	CH0	RIIC_CH0_iic_elctdrep	TDREP ELC signal	Edge	—	—	×	✓	×
			222		RIIC	CH0	RIIC_CH0_iic_elcrdrfp	RDRF ELC signal	Edge	—	—	×	✓	×
			223		RIIC	CH0	RIIC_CH0_iic_elctendp	TEND ELC signal	Edge	—	—	×	✓	×
			224		RIIC	CH1	RIIC_CH1_iic_elcerrp	ERRP ELC signal	Edge	—	—	×	✓	×
			225		RIIC	CH1	RIIC_CH1_iic_elctdrep	TDREP ELC signal	Edge	—	—	×	✓	×
			226		RIIC	CH1	RIIC_CH1_iic_elcrdrfp	RDRF ELC signal	Edge	—	—	×	✓	×
			227		RIIC	CH1	RIIC_CH1_iic_elctendp	TEND ELC signal	Edge	—	—	×	✓	×
			228		RIIC	CH2	RIIC_CH2_iic_elcerrp	ERRP ELC signal	Edge	—	—	×	✓	×
			229		RIIC	CH2	RIIC_CH2_iic_elctdrep	TDREP ELC signal	Edge	—	—	×	✓	×
			230		RIIC	CH2	RIIC_CH2_iic_elcrdrfp	RDRF ELC signal	Edge	—	—	×	✓	×
			231		RIIC	CH2	RIIC_CH2_iic_elctendp	TEND ELC signal	Edge	—	—	×	✓	×
			232		RIIC	CH3	RIIC_CH3_iic_elcerrp	ERRP ELC signal	Edge	—	—	×	✓	×
			233		RIIC	CH3	RIIC_CH3_iic_elctdrep	TDREP ELC signal	Edge	—	—	×	✓	×
			234		RIIC	CH3	RIIC_CH3_iic_elcrdrfp	RDRF ELC signal	Edge	—	—	×	✓	×
			235		RIIC	CH3	RIIC_CH3_iic_elctendp	TEND ELC signal	Edge	—	—	×	✓	×

Table 4.6-24 List of Input Events (46/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
			236		RIIC	CH4	RIIC_CH4_iic_elcerrp	ERRP ELC signal	Edge	—	—	×	✓	×
			237		RIIC	CH4	RIIC_CH4_iic_elctdrep	TDREP ELC signal	Edge	—	—	×	✓	×
			238		RIIC	CH4	RIIC_CH4_iic_elcrdrfp	RDRF ELC signal	Edge	—	—	×	✓	×
			239		RIIC	CH4	RIIC_CH4_iic_elctendp	TEND ELC signal	Edge	—	—	×	✓	×
			240		RIIC	CH5	RIIC_CH5_iic_elcerrp	ERRP ELC signal	Edge	—	—	×	✓	×
			241		RIIC	CH5	RIIC_CH5_iic_elctdrep	TDREP ELC signal	Edge	—	—	×	✓	×
			242		RIIC	CH5	RIIC_CH5_iic_elcrdrfp	RDRF ELC signal	Edge	—	—	×	✓	×
			243		RIIC	CH5	RIIC_CH5_iic_elctendp	TEND ELC signal	Edge	—	—	×	✓	×
			244		RIIC	CH6	RIIC_CH6_iic_elcerrp	ERRP ELC signal	Edge	—	—	×	✓	×
			245		RIIC	CH6	RIIC_CH6_iic_elctdrep	TDREP ELC signal	Edge	—	—	×	✓	×
			246		RIIC	CH6	RIIC_CH6_iic_elcrdrfp	RDRF ELC signal	Edge	—	—	×	✓	×
			247		RIIC	CH6	RIIC_CH6_iic_elctendp	TEND ELC signal	Edge	—	—	×	✓	×
			248		RIIC	CH7	RIIC_CH7_iic_elcerrp	ERRP ELC signal	Edge	—	—	×	✓	×
			249		RIIC	CH7	RIIC_CH7_iic_elctdrep	TDREP ELC signal	Edge	—	—	×	✓	×
			250		RIIC	CH7	RIIC_CH7_iic_elcrdrfp	RDRF ELC signal	Edge	—	—	×	✓	×
			251		RIIC	CH7	RIIC_CH7_iic_elctendp	TEND ELC signal	Edge	—	—	×	✓	×
			252		RIIC	CH8	RIIC_CH8_iic_elcerrp	ERRP ELC signal	Edge	—	—	×	✓	×
			253		RIIC	CH8	RIIC_CH8_iic_elctdrep	TDREP ELC signal	Edge	—	—	×	✓	×
			254		RIIC	CH8	RIIC_CH8_iic_elcrdrfp	RDRF ELC signal	Edge	—	—	×	✓	×
			255		RIIC	CH8	RIIC_CH8_iic_elctendp	TEND ELC signal	Edge	—	—	×	✓	×
			256		ADC	—	ada_adelcreq	Scan end event (for ELC)	Edge	—	—	×	✓	×
			259		RTC	—	rtc_elcalm	Alarm out signal for event link controller	Edge	—	—	×	✓	×
			260		RTC	—	rtc_elccup	Carry signal for event link controller	Edge	—	—	×	✓	×
			261		RTC	—	rtc_elcprd	Periodically generated signal for event link controller	Edge	—	—	×	✓	×
			281		GPT	Unit0	GPT_U0_gpt_elccmpa_0	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			282		GPT	Unit0	GPT_U0_gpt_elccmpa_1	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			283		GPT	Unit0	GPT_U0_gpt_elccmpa_2	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			284		GPT	Unit0	GPT_U0_gpt_elccmpa_3	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			285		GPT	Unit0	GPT_U0_gpt_elccmpa_4	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			286		GPT	Unit0	GPT_U0_gpt_elccmpa_5	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			287		GPT	Unit0	GPT_U0_gpt_elccmpa_6	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			288		GPT	Unit0	GPT_U0_gpt_elccmpa_7	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			289		GPT	Unit0	GPT_U0_gpt_elccmpb_0	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			290		GPT	Unit0	GPT_U0_gpt_elccmpb_1	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			291		GPT	Unit0	GPT_U0_gpt_elccmpb_2	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×

Table 4.6-24 List of Input Events (47/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
			292		GPT	Unit0	GPT_U0_gpt_elccmpb_3	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			293		GPT	Unit0	GPT_U0_gpt_elccmpb_4	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			294		GPT	Unit0	GPT_U0_gpt_elccmpb_5	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			295		GPT	Unit0	GPT_U0_gpt_elccmpb_6	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			296		GPT	Unit0	GPT_U0_gpt_elccmpb_7	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			297		GPT	Unit0	GPT_U0_gpt_elccmpc_0	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			298		GPT	Unit0	GPT_U0_gpt_elccmpc_1	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			299		GPT	Unit0	GPT_U0_gpt_elccmpc_2	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			300		GPT	Unit0	GPT_U0_gpt_elccmpc_3	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			301		GPT	Unit0	GPT_U0_gpt_elccmpc_4	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			302		GPT	Unit0	GPT_U0_gpt_elccmpc_5	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			303		GPT	Unit0	GPT_U0_gpt_elccmpc_6	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			304		GPT	Unit0	GPT_U0_gpt_elccmpc_7	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			305		GPT	Unit0	GPT_U0_gpt_elccmpd_0	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			306		GPT	Unit0	GPT_U0_gpt_elccmpd_1	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			307		GPT	Unit0	GPT_U0_gpt_elccmpd_2	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			308		GPT	Unit0	GPT_U0_gpt_elccmpd_3	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			309		GPT	Unit0	GPT_U0_gpt_elccmpd_4	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			310		GPT	Unit0	GPT_U0_gpt_elccmpd_5	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			311		GPT	Unit0	GPT_U0_gpt_elccmpd_6	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			312		GPT	Unit0	GPT_U0_gpt_elccmpd_7	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			313		GPT	Unit0	GPT_U0_gpt_elccmpe_0	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			314		GPT	Unit0	GPT_U0_gpt_elccmpe_1	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			315		GPT	Unit0	GPT_U0_gpt_elccmpe_2	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			316		GPT	Unit0	GPT_U0_gpt_elccmpe_3	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			317		GPT	Unit0	GPT_U0_gpt_elccmpe_4	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			318		GPT	Unit0	GPT_U0_gpt_elccmpe_5	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			319		GPT	Unit0	GPT_U0_gpt_elccmpe_6	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			320		GPT	Unit0	GPT_U0_gpt_elccmpe_7	Compare match with the GTCCRE register	Edge	—	—	×	✓	×

Table 4.6-24 List of Input Events (48/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
			321		GPT	Unit0	GPT_U0_gpt_elccmpf_0	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			322		GPT	Unit0	GPT_U0_gpt_elccmpf_1	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			323		GPT	Unit0	GPT_U0_gpt_elccmpf_2	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			324		GPT	Unit0	GPT_U0_gpt_elccmpf_3	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			325		GPT	Unit0	GPT_U0_gpt_elccmpf_4	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			326		GPT	Unit0	GPT_U0_gpt_elccmpf_5	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			327		GPT	Unit0	GPT_U0_gpt_elccmpf_6	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			328		GPT	Unit0	GPT_U0_gpt_elccmpf_7	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			329		GPT	Unit0	GPT_U0_gpt_elcovf_0	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			330		GPT	Unit0	GPT_U0_gpt_elcovf_1	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			331		GPT	Unit0	GPT_U0_gpt_elcovf_2	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			332		GPT	Unit0	GPT_U0_gpt_elcovf_3	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			333		GPT	Unit0	GPT_U0_gpt_elcovf_4	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			334		GPT	Unit0	GPT_U0_gpt_elcovf_5	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			335		GPT	Unit0	GPT_U0_gpt_elcovf_6	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			336		GPT	Unit0	GPT_U0_gpt_elcovf_7	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			337		GPT	Unit0	GPT_U0_gpt_elcudf_0	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			338		GPT	Unit0	GPT_U0_gpt_elcudf_1	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			339		GPT	Unit0	GPT_U0_gpt_elcudf_2	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			340		GPT	Unit0	GPT_U0_gpt_elcudf_3	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			341		GPT	Unit0	GPT_U0_gpt_elcudf_4	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			342		GPT	Unit0	GPT_U0_gpt_elcudf_5	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			343		GPT	Unit0	GPT_U0_gpt_elcudf_6	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			344		GPT	Unit0	GPT_U0_gpt_elcudf_7	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			361		GPT	Unit1	GPT_U1_gpt_elccmpa_0	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			362		GPT	Unit1	GPT_U1_gpt_elccmpa_1	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			363		GPT	Unit1	GPT_U1_gpt_elccmpa_2	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			364		GPT	Unit1	GPT_U1_gpt_elccmpa_3	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			365		GPT	Unit1	GPT_U1_gpt_elccmpa_4	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×

Table 4.6-24 List of Input Events (49/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
			366		GPT	Unit1	GPT_U1_gpt_elccmpa_5	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			367		GPT	Unit1	GPT_U1_gpt_elccmpa_6	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			368		GPT	Unit1	GPT_U1_gpt_elccmpa_7	Input capture/compare match of the GTCCRA register	Edge	—	—	×	✓	×
			369		GPT	Unit1	GPT_U1_gpt_elccmpb_0	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			370		GPT	Unit1	GPT_U1_gpt_elccmpb_1	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			371		GPT	Unit1	GPT_U1_gpt_elccmpb_2	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			372		GPT	Unit1	GPT_U1_gpt_elccmpb_3	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			373		GPT	Unit1	GPT_U1_gpt_elccmpb_4	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			374		GPT	Unit1	GPT_U1_gpt_elccmpb_5	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			375		GPT	Unit1	GPT_U1_gpt_elccmpb_6	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			376		GPT	Unit1	GPT_U1_gpt_elccmpb_7	Input capture/compare match of the GTCCRB register	Edge	—	—	×	✓	×
			377		GPT	Unit1	GPT_U1_gpt_elccmpc_0	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			378		GPT	Unit1	GPT_U1_gpt_elccmpc_1	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			379		GPT	Unit1	GPT_U1_gpt_elccmpc_2	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			380		GPT	Unit1	GPT_U1_gpt_elccmpc_3	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			381		GPT	Unit1	GPT_U1_gpt_elccmpc_4	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			382		GPT	Unit1	GPT_U1_gpt_elccmpc_5	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			383		GPT	Unit1	GPT_U1_gpt_elccmpc_6	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			384		GPT	Unit1	GPT_U1_gpt_elccmpc_7	Compare match with the GTCCRC register	Edge	—	—	×	✓	×
			385		GPT	Unit1	GPT_U1_gpt_elccmpd_0	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			386		GPT	Unit1	GPT_U1_gpt_elccmpd_1	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			387		GPT	Unit1	GPT_U1_gpt_elccmpd_2	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			388		GPT	Unit1	GPT_U1_gpt_elccmpd_3	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			389		GPT	Unit1	GPT_U1_gpt_elccmpd_4	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			390		GPT	Unit1	GPT_U1_gpt_elccmpd_5	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			391		GPT	Unit1	GPT_U1_gpt_elccmpd_6	Compare match with the GTCCRD register	Edge	—	—	×	✓	×
			392		GPT	Unit1	GPT_U1_gpt_elccmpd_7	Compare match with the GTCCRD register	Edge	—	—	×	✓	×

Table 4.6-24 List of Input Events (50/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
			393		GPT	Unit1	GPT_U1_gpt_elccmpe_0	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			394		GPT	Unit1	GPT_U1_gpt_elccmpe_1	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			395		GPT	Unit1	GPT_U1_gpt_elccmpe_2	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			396		GPT	Unit1	GPT_U1_gpt_elccmpe_3	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			397		GPT	Unit1	GPT_U1_gpt_elccmpe_4	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			398		GPT	Unit1	GPT_U1_gpt_elccmpe_5	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			399		GPT	Unit1	GPT_U1_gpt_elccmpe_6	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			400		GPT	Unit1	GPT_U1_gpt_elccmpe_7	Compare match with the GTCCRE register	Edge	—	—	×	✓	×
			401		GPT	Unit1	GPT_U1_gpt_elccmpf_0	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			402		GPT	Unit1	GPT_U1_gpt_elccmpf_1	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			403		GPT	Unit1	GPT_U1_gpt_elccmpf_2	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			404		GPT	Unit1	GPT_U1_gpt_elccmpf_3	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			405		GPT	Unit1	GPT_U1_gpt_elccmpf_4	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			406		GPT	Unit1	GPT_U1_gpt_elccmpf_5	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			407		GPT	Unit1	GPT_U1_gpt_elccmpf_6	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			408		GPT	Unit1	GPT_U1_gpt_elccmpf_7	Compare match with the GTCCRF register	Edge	—	—	×	✓	×
			409		GPT	Unit1	GPT_U1_gpt_elcovf_0	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			410		GPT	Unit1	GPT_U1_gpt_elcovf_1	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			411		GPT	Unit1	GPT_U1_gpt_elcovf_2	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			412		GPT	Unit1	GPT_U1_gpt_elcovf_3	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			413		GPT	Unit1	GPT_U1_gpt_elcovf_4	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			414		GPT	Unit1	GPT_U1_gpt_elcovf_5	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			415		GPT	Unit1	GPT_U1_gpt_elcovf_6	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			416		GPT	Unit1	GPT_U1_gpt_elcovf_7	Overflow of the GTCNT counter	Edge	—	—	×	✓	×
			417		GPT	Unit1	GPT_U1_gpt_elcudf_0	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			418		GPT	Unit1	GPT_U1_gpt_elcudf_1	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			419		GPT	Unit1	GPT_U1_gpt_elcudf_2	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			420		GPT	Unit1	GPT_U1_gpt_elcudf_3	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			421		GPT	Unit1	GPT_U1_gpt_elcudf_4	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			422		GPT	Unit1	GPT_U1_gpt_elcudf_5	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			423		GPT	Unit1	GPT_U1_gpt_elcudf_6	Underflow of the GTCNT counter	Edge	—	—	×	✓	×

Table 4.6-24 List of Input Events (51/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
			424		GPT	Unit1	GPT_U1_gpt_elcudf_7	Underflow of the GTCNT counter	Edge	—	—	×	✓	×
			425		I3C	—	ri3c_elccommu	Communication event	Edge	—	—	×	✓	×
			426		I3C	—	ri3c_elcresp	Response buffer full event	Edge	—	—	×	✓	×
			427		I3C	—	ri3c_elccmd	Command buffer empty event	Edge	—	—	×	✓	×
			428		I3C	—	ri3c_elcibi	IBI status buffer full event	Edge	—	—	×	✓	×
			429		I3C	—	ri3c_elcrx	Rx data buffer full event	Edge	—	—	×	✓	×
			430		I3C	—	ri3c_elctx	Tx data buffer empty event	Edge	—	—	×	✓	×
			431		I3C	—	ri3c_elcrcv	Receive status buffer full event	Edge	—	—	×	✓	×
			432		—	—	Reserved	Reserved	—	—	—	—	—	—
			433		—	—	Reserved	Reserved	—	—	—	—	—	—
			434		—	—	Reserved	Reserved	—	—	—	—	—	—
			435		—	—	Reserved	Reserved	—	—	—	—	—	—
			436		I3C	—	ri3c_elctend	Transmit end event	Edge	—	—	×	✓	×
			437		I3C	—	ri3c_elcstev	Synchronous Timing Event	Edge	—	—	×	✓	×
			438		I3C	—	ri3c_mrefovf	MREF Counter Overflow	Edge	—	—	×	✓	×
			439		I3C	—	ri3c_mrefcpt	MREF Capture Event	Edge	—	—	×	✓	×
			440		I3C	—	ri3c_elcamev	Additional Master-initiated bus Event	Edge	—	—	×	✓	×
			441		GBETH	PORT 1	—	—	—	—	—	×	✓	×
			442		GBETH	PORT 1	—	—	—	—	—	×	✓	×
			443		GBETH	PORT 1	—	—	—	—	—	×	✓	×
			444		GBETH	PORT 1	—	—	—	—	—	×	✓	×
			445		GBETH	PORT 0	—	—	—	—	—	×	✓	×
			446		GBETH	PORT 0	—	—	—	—	—	×	✓	×
			447		GBETH	PORT 0	—	—	—	—	—	×	✓	×
			448		GBETH	PORT 0	—	—	—	—	—	×	✓	×
			453		—	—	Reserved	Reserved	—	—	—	—	—	—
			454		—	—	Reserved	Reserved	—	—	—	—	—	—
			455		—	—	Reserved	Reserved	—	—	—	—	—	—
				0	ICU	—	BUS_ERR_INT	Signal after being merged within INT_marge(BUS)	Level	—	—	×	×	✓
				1	ICU	—	RAM_ERR_INT	Signal after being merged within INT_marge(RAM)	Level	—	—	×	×	✓
				2	CA55	—	ACPU_nFAULTIRQ_0	Fault indicator for a detected 1-bit or 2-bit ECC or Parity error in the RAMs.	Level	—	—	×	×	✓
				3	CA55	—	ACPU_nFAULTIRQ_1	Fault indicator for a detected 1-bit or 2-bit ECC or Parity error in the RAMs.	Level	—	—	×	×	✓
				4	CA55	—	ACPU_nFAULTIRQ_2	Fault indicator for a detected 1-bit or 2-bit ECC or Parity error in the RAMs.	Level	—	—	×	×	✓
				5	CA55	—	ACPU_nFAULTIRQ_3	Fault indicator for a detected 1-bit or 2-bit ECC or Parity error in the RAMs.	Level	—	—	×	×	✓
				6	CA55	—	ACPU_nFAULTIRQ_4	Fault indicator for a detected 1-bit or 2-bit ECC or Parity error in the RAMs.	Level	—	—	×	×	✓

Table 4.6-24 List of Input Events (52/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
				7	CA55	—	ACPU_nERRIRQ_0	Error indicator for an ECC error that causes potential data corruption or loss of coherency.	Level	—	—	×	×	✓
				8	CA55	—	ACPU_nERRIRQ_1	Error indicator for an ECC error that causes potential data corruption or loss of coherency.	Level	—	—	×	×	✓
				9	CA55	—	ACPU_nERRIRQ_2	Error indicator for an ECC error that causes potential data corruption or loss of coherency.	Level	—	—	×	×	✓
				10	CA55	—	ACPU_nERRIRQ_3	Error indicator for an ECC error that causes potential data corruption or loss of coherency.	Level	—	—	×	×	✓
				11	CA55	—	ACPU_nERRIRQ_4	Error indicator for an ECC error that causes potential data corruption or loss of coherency.	Level	—	—	×	×	✓
				25	CM33	—	MCPU_LOCKUP	Processor is in lockup state	Level	—	—	×	×	✓
				26	DDR	CH0	—		—	—	—	×	×	✓
				27	DDR	CH0	—		—	—	—	×	×	✓
				28	—	—	Reserved	Reserved	—	—	—	—	—	—
				29	—	—	Reserved	Reserved	—	—	—	—	—	—
				30	WDT	CA55	WDT_CA55_iwdt_nmiundf_n	Down-counter underflow/refresh error	Edge	—	—	×	×	✓
				31	WDT	CM33	WDT_CM33_iwdt_nmiundf_n	Down-counter underflow/refresh error	Edge	—	—	×	×	✓
				34	ADC	—	ada_adereq_n	Check for overwriting data registers	Edge	—	—	×	×	✓
				35	—	—	Reserved	Reserved	—	—	—	—	—	—
				36	—	—	Reserved	Reserved	—	—	—	—	—	—
				37	—	—	Reserved	Reserved	—	—	—	—	—	—
				38	—	—	Reserved	Reserved	—	—	—	—	—	—
				39	—	—	Reserved	Reserved	—	—	—	—	—	—
				40	—	—	Reserved	Reserved	—	—	—	—	—	—
				41	—	—	Reserved	Reserved	—	—	—	—	—	—
				42	—	—	Reserved	Reserved	—	—	—	—	—	—
				43	—	—	Reserved	Reserved	—	—	—	—	—	—
				44	—	—	Reserved	Reserved	—	—	—	—	—	—
				45	—	—	Reserved	Reserved	—	—	—	—	—	—
				46	—	—	Reserved	Reserved	—	—	—	—	—	—
				47	—	—	Reserved	Reserved	—	—	—	—	—	—
				48	—	—	Reserved	Reserved	—	—	—	—	—	—
				49	—	—	Reserved	Reserved	—	—	—	—	—	—
				98	ICU	SWPE	SW_PSEUDO_ERR_0	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				99	ICU	SWPE	SW_PSEUDO_ERR_1	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				100	ICU	SWPE	SW_PSEUDO_ERR_2	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				101	ICU	SWPE	SW_PSEUDO_ERR_3	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				102	ICU	SWPE	SW_PSEUDO_ERR_4	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓

Table 4.6-24 List of Input Events (53/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
				103	ICU	SWPE	SW_PSEUDO_ERR_5	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				104	ICU	SWPE	SW_PSEUDO_ERR_6	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				105	ICU	SWPE	SW_PSEUDO_ERR_7	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				106	ICU	SWPE	SW_PSEUDO_ERR_8	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				107	ICU	SWPE	SW_PSEUDO_ERR_9	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				108	ICU	SWPE	SW_PSEUDO_ERR_10	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				109	ICU	SWPE	SW_PSEUDO_ERR_11	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				110	ICU	SWPE	SW_PSEUDO_ERR_12	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				111	ICU	SWPE	SW_PSEUDO_ERR_13	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				112	ICU	SWPE	SW_PSEUDO_ERR_14	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
				113	ICU	SWPE	SW_PSEUDO_ERR_15	Occurs by writing to the ICU_TOP register (for Error Interrupt)	Edge	—	—	×	×	✓
	SEL163				GPT	U0	GPT_U0_gpt_gtciada_n_0	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL164				GPT	U0	GPT_U0_gpt_gtciada_n_1	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL165				GPT	U0	GPT_U0_gpt_gtciada_n_2	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL166				GPT	U0	GPT_U0_gpt_gtciada_n_3	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL167				GPT	U0	GPT_U0_gpt_gtciada_n_4	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL168				GPT	U0	GPT_U0_gpt_gtciada_n_5	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL169				GPT	U0	GPT_U0_gpt_gtciada_n_6	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL170				GPT	U0	GPT_U0_gpt_gtciada_n_7	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL171				GPT	U0	GPT_U0_gpt_gtciadb_n_0	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL172				GPT	U0	GPT_U0_gpt_gtciadb_n_1	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL173				GPT	U0	GPT_U0_gpt_gtciadb_n_2	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL174				GPT	U0	GPT_U0_gpt_gtciadb_n_3	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL175				GPT	U0	GPT_U0_gpt_gtciadb_n_4	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL176				GPT	U0	GPT_U0_gpt_gtciadb_n_5	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL177				GPT	U0	GPT_U0_gpt_gtciadb_n_6	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL178				GPT	U0	GPT_U0_gpt_gtciadb_n_7	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×

Table 4.6-24 List of Input Events (54/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
	SEL179	257		50	GPT	U0	GPT_U0_gpt_gtciv_n_0	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL180	258		51	GPT	U0	GPT_U0_gpt_gtciv_n_1	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL181	259		52	GPT	U0	GPT_U0_gpt_gtciv_n_2	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL182	260		53	GPT	U0	GPT_U0_gpt_gtciv_n_3	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL183	261		54	GPT	U0	GPT_U0_gpt_gtciv_n_4	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL184	262		55	GPT	U0	GPT_U0_gpt_gtciv_n_5	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL185	263		56	GPT	U0	GPT_U0_gpt_gtciv_n_6	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL186	264		57	GPT	U0	GPT_U0_gpt_gtciv_n_7	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL187	265		58	GPT	U0	GPT_U0_gpt_gtciu_n_0	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL188	266		59	GPT	U0	GPT_U0_gpt_gtciu_n_1	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL189	267		60	GPT	U0	GPT_U0_gpt_gtciu_n_2	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL190	268		61	GPT	U0	GPT_U0_gpt_gtciu_n_3	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL191	269		62	GPT	U0	GPT_U0_gpt_gtciu_n_4	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL192	270		63	GPT	U0	GPT_U0_gpt_gtciu_n_5	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL193	271		64	GPT	U0	GPT_U0_gpt_gtciu_n_6	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL194	272		65	GPT	U0	GPT_U0_gpt_gtciu_n_7	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL211			66	GPT	U0	GPT_U0_gpt_gtdei_n_0	Dead time error	Edge	—	SELECT	×	×	✓
	SEL212			67	GPT	U0	GPT_U0_gpt_gtdei_n_1	Dead time error	Edge	—	SELECT	×	×	✓
	SEL213			68	GPT	U0	GPT_U0_gpt_gtdei_n_2	Dead time error	Edge	—	SELECT	×	×	✓
	SEL214			69	GPT	U0	GPT_U0_gpt_gtdei_n_3	Dead time error	Edge	—	SELECT	×	×	✓
	SEL215			70	GPT	U0	GPT_U0_gpt_gtdei_n_4	Dead time error	Edge	—	SELECT	×	×	✓
	SEL216			71	GPT	U0	GPT_U0_gpt_gtdei_n_5	Dead time error	Edge	—	SELECT	×	×	✓
	SEL217			72	GPT	U0	GPT_U0_gpt_gtdei_n_6	Dead time error	Edge	—	SELECT	×	×	✓
	SEL218			73	GPT	U0	GPT_U0_gpt_gtdei_n_7	Dead time error	Edge	—	SELECT	×	×	✓
	SEL267				GPT	U1	GPT_U1_gpt_gtciada_n_0	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL268				GPT	U1	GPT_U1_gpt_gtciada_n_1	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL269				GPT	U1	GPT_U1_gpt_gtciada_n_2	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL270				GPT	U1	GPT_U1_gpt_gtciada_n_3	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL271				GPT	U1	GPT_U1_gpt_gtciada_n_4	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL272				GPT	U1	GPT_U1_gpt_gtciada_n_5	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL273				GPT	U1	GPT_U1_gpt_gtciada_n_6	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×

Table 4.6-24 List of Input Events (55/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
	SEL274				GPT	U1	GPT_U1_gpt_gtciada_n_7	Compare match with the GTADTRA register	Edge	—	SELECT	×	×	×
	SEL275				GPT	U1	GPT_U1_gpt_gtciadb_n_0	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL276				GPT	U1	GPT_U1_gpt_gtciadb_n_1	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL277				GPT	U1	GPT_U1_gpt_gtciadb_n_2	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL278				GPT	U1	GPT_U1_gpt_gtciadb_n_3	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL279				GPT	U1	GPT_U1_gpt_gtciadb_n_4	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL280				GPT	U1	GPT_U1_gpt_gtciadb_n_5	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL281				GPT	U1	GPT_U1_gpt_gtciadb_n_6	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL282				GPT	U1	GPT_U1_gpt_gtciadb_n_7	Compare match with the GTADTRB register	Edge	—	SELECT	×	×	×
	SEL283	321		74	GPT	U1	GPT_U1_gpt_gtciv_n_0	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL284	322		75	GPT	U1	GPT_U1_gpt_gtciv_n_1	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL285	323		76	GPT	U1	GPT_U1_gpt_gtciv_n_2	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL286	324		77	GPT	U1	GPT_U1_gpt_gtciv_n_3	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL287	325		78	GPT	U1	GPT_U1_gpt_gtciv_n_4	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL288	326		79	GPT	U1	GPT_U1_gpt_gtciv_n_5	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL289	327		80	GPT	U1	GPT_U1_gpt_gtciv_n_6	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL290	328		81	GPT	U1	GPT_U1_gpt_gtciv_n_7	Overflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL291	329		82	GPT	U1	GPT_U1_gpt_gtciu_n_0	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL292	330		83	GPT	U1	GPT_U1_gpt_gtciu_n_1	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL293	331		84	GPT	U1	GPT_U1_gpt_gtciu_n_2	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL294	332		85	GPT	U1	GPT_U1_gpt_gtciu_n_3	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL295	333		86	GPT	U1	GPT_U1_gpt_gtciu_n_4	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL296	334		87	GPT	U1	GPT_U1_gpt_gtciu_n_5	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL297	335		88	GPT	U1	GPT_U1_gpt_gtciu_n_6	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL298	336		89	GPT	U1	GPT_U1_gpt_gtciu_n_7	Underflow of the GTCNT counter	Edge	—	SELECT	✓	×	✓
	SEL315			90	GPT	U1	GPT_U1_gpt_gtdei_n_0	Dead time error	Edge	—	SELECT	×	×	✓
	SEL316			91	GPT	U1	GPT_U1_gpt_gtdei_n_1	Dead time error	Edge	—	SELECT	×	×	✓
	SEL317			92	GPT	U1	GPT_U1_gpt_gtdei_n_2	Dead time error	Edge	—	SELECT	×	×	✓
	SEL318			93	GPT	U1	GPT_U1_gpt_gtdei_n_3	Dead time error	Edge	—	SELECT	×	×	✓
	SEL319			94	GPT	U1	GPT_U1_gpt_gtdei_n_4	Dead time error	Edge	—	SELECT	×	×	✓
	SEL320			95	GPT	U1	GPT_U1_gpt_gtdei_n_5	Dead time error	Edge	—	SELECT	×	×	✓

Table 4.6-24 List of Input Events (56/56)

CA55 SPI No.	CM33 SPI No.	DMAC No.	ELC No.	Error Event	Unit	CH	Name	Interrupt Source	Level/Edge	CA55	CM33	DMAC	ELC	Error Event
	SEL321			96	GPT	U1	GPT_U1_gpt_gtdei_n_6	Dead time error	Edge	—	SELECT	×	×	✓
	SEL322			97	GPT	U1	GPT_U1_gpt_gtdei_n_7	Dead time error	Edge	—	SELECT	×	×	✓

4.6.1.5.2 List of output events

Table 4.6-25 List of Output Events (1/2)

Event Output No.	Unit	CH	Name	Input/ Output	Sync	Active	Initial Value	Description
0	GPT	Unit 0	GPT_U0_elc_gptact_0	Output	PCLK	H	0	GPT event sources A to H
1	GPT	Unit 0	GPT_U0_elc_gptact_1	Output	PCLK	H	0	GPT event sources A to H
2	GPT	Unit 0	GPT_U0_elc_gptact_2	Output	PCLK	H	0	GPT event sources A to H
3	GPT	Unit 0	GPT_U0_elc_gptact_3	Output	PCLK	H	0	GPT event sources A to H
4	GPT	Unit 0	GPT_U0_elc_gptact_4	Output	PCLK	H	0	GPT event sources A to H
5	GPT	Unit 0	GPT_U0_elc_gptact_5	Output	PCLK	H	0	GPT event sources A to H
6	GPT	Unit 0	GPT_U0_elc_gptact_6	Output	PCLK	H	0	GPT event sources A to H
7	GPT	Unit 0	GPT_U0_elc_gptact_7	Output	PCLK	H	0	GPT event sources A to H
8	GPT	Unit 1	GPT_U1_elc_gptact_0	Output	PCLK	H	0	GPT event sources A to H
9	GPT	Unit 1	GPT_U1_elc_gptact_1	Output	PCLK	H	0	GPT event sources A to H
10	GPT	Unit 1	GPT_U1_elc_gptact_2	Output	PCLK	H	0	GPT event sources A to H
11	GPT	Unit 1	GPT_U1_elc_gptact_3	Output	PCLK	H	0	GPT event sources A to H
12	GPT	Unit 1	GPT_U1_elc_gptact_4	Output	PCLK	H	0	GPT event sources A to H
13	GPT	Unit 1	GPT_U1_elc_gptact_5	Output	PCLK	H	0	GPT event sources A to H
14	GPT	Unit 1	GPT_U1_elc_gptact_6	Output	PCLK	H	0	GPT event sources A to H
15	GPT	Unit 1	GPT_U1_elc_gptact_7	Output	PCLK	H	0	GPT event sources A to H
16	ADC	—	trg49_ada_n	Output	PCLK	L	1	ELC trigger
17	—	—	Reserved	—	—	—	—	—
18	TSU1	—	TSU_trg	Output	PCLK	L	1	ELC trigger (timer)
19	PFC	—	elc_input_port_group_1	Output	PCLK	H	0	GPIO single/group setting
20	PFC	—	elc_input_port_group_0	Output	PCLK	H	0	GPIO single/group setting
21	PFC	—	elc_single_port_0	Output	PCLK	H	0	GPIO single/group setting
22	PFC	—	elc_single_port_1	Output	PCLK	H	0	GPIO single/group setting
23	PFC	—	elc_single_port_2	Output	PCLK	H	0	GPIO single/group setting
24	PFC	—	elc_single_port_3	Output	PCLK	H	0	GPIO single/group setting
25	PFC	—	elc_output_port_group_0	Output	PCLK	H	0	GPIO single/group setting
26	PFC	—	elc_output_port_group_1	Output	PCLK	H	0	GPIO single/group setting
27	GBETH	Port 0	—	—	—	—	—	—
28	GBETH	Port 0	—	—	—	—	—	—
29	GBETH	Port 0	—	—	—	—	—	—
30	GBETH	Port 0	—	—	—	—	—	—
31	GBETH	Port 1	—	—	—	—	—	—
32	GBETH	Port 1	—	—	—	—	—	—
33	GBETH	Port 1	—	—	—	—	—	—
34	GBETH	Port 1	—	—	—	—	—	—
35	ICU	—	EVT_DMAC_0	ICU_TOP internal	PCLK	H	0	Event (input) to DMAC Req
36	ICU	—	EVT_DMAC_1	ICU_TOP internal	PCLK	H	0	Event (input) to DMAC Req
37	ICU	—	EVT_DMAC_2	ICU_TOP internal	PCLK	H	0	Event (input) to DMAC Req
38	ICU	—	EVT_DMAC_3	ICU_TOP internal	PCLK	H	0	Event (input) to DMAC Req

Table 4.6-25 List of Output Events (2/2)

Event Output No.	Unit	CH	Name	I/O	Sync	Active	Initial Value	Description
39	ICU	—	EVT_DMAC_4	ICU_TOP internal	PCLK	H	0	Event (input) to DMAC Req
40	ICU	—	EVT_DMAC_5	ICU_TOP internal	PCLK	H	0	Event (input) to DMAC Req
41	ICU	—	EVT_DMAC_6	ICU_TOP internal	PCLK	H	0	Event (input) to DMAC Req
42	ICU	—	EVT_DMAC_7	ICU_TOP internal	PCLK	H	0	Event (input) to DMAC Req

4.6.1.5.3 List of bus error events

The bus error events are listed below and on the following pages. The ICU bundles these interrupts into the BUS_ERR_INT signal.

Table 4.6-26 List of Bus Error Events (1/2)

Unit	CH	Signal	Level/ Edge	Active Level	Description
CM33	—	ERRINT_BRG_CM33_C	Edge	High	Interrupt when a C-AHB bridge error occurs
CM33	—	ERRINT_BRG_CM33_S	Edge	High	Interrupt when a C-AHB bridge error occurs
BUS	—	BUS_ERR_INTR_W_SXA55M	Edge	High	Bus write error interrupt for CA55
BUS	—	BUS_ERR_INTR_R_SXA55M	Edge	High	Bus read error interrupt for CA55
BUS	—	BUS_ERR_INTR_W_SXA55P	Edge	High	Bus write error interrupt for CA55 peripherals
BUS	—	BUS_ERR_INTR_R_SXA55P	Edge	High	Bus read error interrupt for CA55 peripherals
BUS	—	BUS_ERR_INTR_W_SXGPU	Edge	High	Bus write error interrupt for GE3D
BUS	—	BUS_ERR_INTR_R_SXGPU	Edge	High	Bus read error interrupt for GE3D
BUS	—	BUS_ERR_INTR_W_SXADMC0	Edge	High	Bus write error interrupt for DMAC1
BUS	—	BUS_ERR_INTR_R_SXADMC0	Edge	High	Bus read error interrupt for DMAC1
BUS	—	BUS_ERR_INTR_W_SXADMC1	Edge	High	Bus write error interrupt for DMAC2
BUS	—	BUS_ERR_INTR_R_SXADMC1	Edge	High	Bus read error interrupt for DMAC2
BUS	—	BUS_ERR_INTR_W_SXCSTE	Edge	High	Bus write error interrupt for CST_ETR
BUS	—	BUS_ERR_INTR_R_SXCSTE	Edge	High	Bus read error interrupt for CST_ETR
BUS	—	BUS_ERR_INTR_W_SXCSTA	Edge	High	Bus write error interrupt for CST_AP
BUS	—	BUS_ERR_INTR_R_SXCSTA	Edge	High	Bus read error interrupt for CST_AP
BUS	—	BUS_ERR_INTR_W_SXM33S	Edge	High	Bus write error interrupt for CM33_S
BUS	—	BUS_ERR_INTR_R_SXM33S	Edge	High	Bus read error interrupt for CM33_S
BUS	—	BUS_ERR_INTR_W_SXM33C	Edge	High	Bus write error interrupt for CM33_C
BUS	—	BUS_ERR_INTR_R_SXM33C	Edge	High	Bus read error interrupt for CM33_C
BUS	—	BUS_ERR_INTR_W_SXMDMC	Edge	High	Bus write error interrupt for DMAC0
BUS	—	BUS_ERR_INTR_R_SXMDMC	Edge	High	Bus read error interrupt for DMAC0
BUS	—	BUS_ERR_INTR_W_SXRDMC0	Edge	High	Bus write error interrupt for DMAC3
BUS	—	BUS_ERR_INTR_R_SXRDMC0	Edge	High	Bus read error interrupt for DMAC3
BUS	—	BUS_ERR_INTR_W_SXRDMC1	Edge	High	Bus write error interrupt for DMAC4
BUS	—	BUS_ERR_INTR_R_SXRDMC1	Edge	High	Bus read error interrupt for DMAC4
BUS	—	BUS_ERR_INTR_W_SXSD0	Edge	High	Bus write error interrupt for SD channel 0
BUS	—	BUS_ERR_INTR_R_SXSD0	Edge	High	Bus read error interrupt for SD channel 0
BUS	—	BUS_ERR_INTR_W_SXSD1	Edge	High	Bus write error interrupt for SD channel 1
BUS	—	BUS_ERR_INTR_R_SXSD1	Edge	High	Bus read error interrupt for SD channel 1
BUS	—	BUS_ERR_INTR_W_SXSD2	Edge	High	Bus write error interrupt for SD channel 2
BUS	—	BUS_ERR_INTR_R_SXSD2	Edge	High	Bus read error interrupt for SD channel 2
BUS	—	BUS_ERR_INTR_W_SXGBE0	Edge	High	Bus write error interrupt for GBETH channel 0
BUS	—	BUS_ERR_INTR_R_SXGBE0	Edge	High	Bus read error interrupt for GBETH channel 0
BUS	—	BUS_ERR_INTR_W_SXGBE1	Edge	High	Bus write error interrupt for GBETH channel 1
BUS	—	BUS_ERR_INTR_R_SXGBE1	Edge	High	Bus read error interrupt for GBETH channel 1
BUS	—	BUS_ERR_INTR_W_SHU2H0	Edge	High	Bus write error interrupt for USB2 channel 0 host
BUS	—	BUS_ERR_INTR_R_SHU2H0	Edge	High	Bus read error interrupt for USB2 channel 0 host
BUS	—	BUS_ERR_INTR_W_SHU2F	Edge	High	Bus write error interrupt for USB2 channel 0 function
BUS	—	BUS_ERR_INTR_R_SHU2F	Edge	High	Bus read error interrupt for USB2 channel 0 function

Table 4.6-26 List of Bus Error Events (2/2)

Unit	CH	Signal	Level/ Edge	Active Level	Description
BUS	—	BUS_ERR_INTR_W_SHU2H1	Edge	High	Bus write error interrupt for USB2 channel 1 host
BUS	—	BUS_ERR_INTR_R_SHU2H1	Edge	High	Bus read error interrupt for USB2 channel 1 host
BUS	—	BUS_ERR_INTR_W_SXU3H0	Edge	High	Bus write error interrupt for USB3 channel 0 host
BUS	—	BUS_ERR_INTR_R_SXU3H0	Edge	High	Bus read error interrupt for USB3 channel 0 host
BUS	—	BUS_ERR_INTR_W_SXPCIE0	Edge	High	Bus write error interrupt for PCIE0
BUS	—	BUS_ERR_INTR_R_SXPCIE0	Edge	High	Bus read error interrupt for PCIE0
BUS	—	BUS_ERR_INTR_W_SXCRV0	Edge	High	Bus write error interrupt for CRU channel 0 video
BUS	—	BUS_ERR_INTR_W_SXCRS0	Edge	High	Bus write error interrupt for CRU channel 0 statistics
BUS	—	BUS_ERR_INTR_W_SXLCD0	Edge	High	Bus write error interrupt for LCDC0
BUS	—	BUS_ERR_INTR_R_SXLCD0	Edge	High	Bus read error interrupt for LCDC0
BUS	—	BUS_ERR_INTR_W_SXLCD1	Edge	High	Bus write error interrupt for LCDC1
BUS	—	BUS_ERR_INTR_R_SXLCD1	Edge	High	Bus read error interrupt for LCDC1
BUS	—	BUS_ERR_INTR_W_SXDSI	Edge	High	Bus write error interrupt for MIPI DSI link
BUS	—	BUS_ERR_INTR_R_SXDSI	Edge	High	Bus read error interrupt for MIPI DSI link
BUS	—	BUS_ERR_INTR_W_SXVCP	Edge	High	Bus write error interrupt for VCD
BUS	—	BUS_ERR_INTR_R_SXVCP	Edge	High	Bus read error interrupt for VCD
BUS	—	BUS_TZC0INT_DDR_TZCDDR0	Level	High	Conveys access security violations or region overlap errors.
BUS	—	BUS_TZC1INT_DDR_TZCDDR0	Level	High	Conveys access security violations or region overlap errors.
BUS	—	BUS_TZCINT_PCI	Level	High	Conveys access security violations or region overlap errors.
BUS	—	BUS_TZCINT_MSRM	Level	High	Conveys access security violations or region overlap errors.
BUS	—	BUS_TZCINT_XSP	Level	High	Conveys access security violations or region overlap errors.
BUS	—	BUS_TZCINT_ACRCB	Level	High	Conveys access security violations or region overlap errors.
BUS	—	BUS_ERR_INTR_W_SXVSP	Edge	High	Bus Write Error Interrupt for VSPI
BUS	—	BUS_ERR_INTR_R_SXVSP	Edge	High	Bus Read Error Interrupt for VSPI
BUS	—	BUS_ERR_INTR_W_SXNPU	Edge	High	Bus Write Error Interrupt for NPU Master Port0
BUS	—	BUS_ERR_INTR_R_SXNPU	Edge	High	Bus Read Error Interrupt for NPU Master Port0
BUS	—	BUS_ERR_INTR_R_SXNPUR	Edge	High	Bus Write Error Interrupt for NPU Master Port1(Read Only)
BUS	—	BUS_ERR_INTR_W_SXFDP1	Edge	High	Bus Write Error Interrupt for FDP1
BUS	—	BUS_ERR_INTR_R_SXFDP1	Edge	High	Bus Read Error Interrupt for FDP1

4.6.1.5.4 List of SRAM error events

The SRAM error events are listed below. The ICU bundles these interrupts into the RAM_ERR_INT signal.

Table 4.6-27 List of SRAM Error Events

Unit	Signal	Level/ Edge	Active Level	Description
SRAM0	SRAM0_EC7TIE1[1:0]	Edge	High	ECC 1-bit error interrupt
SRAM0	SRAM0_EC7TIE2[1:0]	Edge	High	ECC 2-bit error interrupt
SRAM0	SRAM0_EC7TIOVF[1:0]	Edge	High	Interrupt on overflow of capturing ECC error addresses
SRAM1	SRAM1_EC7TIE1[1:0]	Edge	High	ECC 1-bit error interrupt
SRAM1	SRAM1_EC7TIE2[1:0]	Edge	High	ECC 2-bit error interrupt
SRAM1	SRAM1_EC7TIOVF[1:0]	Edge	High	Interrupt on overflow of capturing ECC error addresses

4.6.1.5.5 List of the DMA ACK signals

The DMA ACK signals output from the ICU are listed below and on the following pages.

Table 4.6-28 List of the DMA ACK Output Signals (1/3)

Ack No.	TEND No.	Unit	CH	Signal	Description	Level/Edge	Active Level	Corresponding Req Signal
0		SPDIF	CH0	dmaack_tx_n	DMA ACK	Level	Low	rbdmareqn_tx
1		SPDIF	CH0	dmaack_rx_n	DMA ACK	Level	Low	rbdmareqn_rx
2		SPDIF	CH1	dmaack_tx_n	DMA ACK	Level	Low	rbdmareqn_tx
3		SPDIF	CH1	dmaack_rx_n	DMA ACK	Level	Low	rbdmareqn_rx
4		SPDIF	CH2	dmaack_tx_n	DMA ACK	Level	Low	rbdmareqn_tx
5		SPDIF	CH2	dmaack_rx_n	DMA ACK	Level	Low	rbdmareqn_rx
6		SCU	—	pdrack_sr0_in	DMA ACK	Level	High	pdreq_sr0_in
7		SCU	—	pdrack_sr1_in	DMA ACK	Level	High	pdreq_sr1_in
8		SCU	—	pdrack_sr2_in	DMA ACK	Level	High	pdreq_sr2_in
9		SCU	—	pdrack_sr3_in	DMA ACK	Level	High	pdreq_sr3_in
10		SCU	—	pdrack_sr4_in	DMA ACK	Level	High	pdreq_sr4_in
11		SCU	—	pdrack_sr5_in	DMA ACK	Level	High	pdreq_sr5_in
12		SCU	—	pdrack_sr6_in	DMA ACK	Level	High	pdreq_sr6_in
13		SCU	—	pdrack_sr7_in	DMA ACK	Level	High	pdreq_sr7_in
14		SCU	—	pdrack_sr8_in	DMA ACK	Level	High	pdreq_sr8_in
15		SCU	—	pdrack_sr9_in	DMA ACK	Level	High	pdreq_sr9_in
16		SCU	—	pdrack_sr0_out	DMA ACK	Level	High	pdreq_sr0_out
17		SCU	—	pdrack_sr1_out	DMA ACK	Level	High	pdreq_sr1_out
18		SCU	—	pdrack_sr2_out	DMA ACK	Level	High	pdreq_sr2_out
19		SCU	—	pdrack_sr3_out	DMA ACK	Level	High	pdreq_sr3_out
20		SCU	—	pdrack_sr4_out	DMA ACK	Level	High	pdreq_sr4_out
21		SCU	—	pdrack_sr5_out	DMA ACK	Level	High	pdreq_sr5_out
22		SCU	—	pdrack_sr6_out	DMA ACK	Level	High	pdreq_sr6_out
23		SCU	—	pdrack_sr7_out	DMA ACK	Level	High	pdreq_sr7_out
24		SCU	—	pdrack_sr8_out	DMA ACK	Level	High	pdreq_sr8_out
25		SCU	—	pdrack_sr9_out	DMA ACK	Level	High	pdreq_sr9_out
26		SCU	—	pdrack_cmd0_out	DMA ACK	Level	High	pdreq_cmd0_out
27		SCU	—	pdrack_cmd1_out	DMA ACK	Level	High	pdreq_cmd1_out
28		SSIU	—	ssip00_drack_rx	DMA ACK	Level	High	ssip00_dreq_rx
29		SSIU	—	ssip00_drack_tx	DMA ACK	Level	High	ssip00_dreq_tx
30		SSIU	—	ssip01_drack_rx	DMA ACK	Level	High	ssip01_dreq_rx
31		SSIU	—	ssip01_drack_tx	DMA ACK	Level	High	ssip01_dreq_tx
32		SSIU	—	ssip02_drack_rx	DMA ACK	Level	High	ssip02_dreq_rx
33		SSIU	—	ssip02_drack_tx	DMA ACK	Level	High	ssip02_dreq_tx
34		SSIU	—	ssip03_drack_rx	DMA ACK	Level	High	ssip03_dreq_rx
35		SSIU	—	ssip03_drack_tx	DMA ACK	Level	High	ssip03_dreq_tx
36		SSIU	—	ssip10_drack_rx	DMA ACK	Level	High	ssip10_dreq_rx
37		SSIU	—	ssip10_drack_tx	DMA ACK	Level	High	ssip10_dreq_tx
38		SSIU	—	ssip11_drack_rx	DMA ACK	Level	High	ssip11_dreq_rx
39		SSIU	—	ssip11_drack_tx	DMA ACK	Level	High	ssip11_dreq_tx
40		SSIU	—	ssip12_drack_rx	DMA ACK	Level	High	ssip12_dreq_rx

Table 4.6-28 List of the DMA ACK Output Signals (2/3)

Ack No.	TEND No.	Unit	CH	Signal	Description	Level/Edge	Active Level	Corresponding Req Signal
41		SSIU	—	ssip12_drack_tx	DMA ACK	Level	High	ssip12_dreq_tx
42		SSIU	—	ssip13_drack_rx	DMA ACK	Level	High	ssip13_dreq_rx
43		SSIU	—	ssip13_drack_tx	DMA ACK	Level	High	ssip13_dreq_tx
44		SSIU	—	ssip20_drack_rx	DMA ACK	Level	High	ssip20_dreq_rx
45		SSIU	—	ssip20_drack_tx	DMA ACK	Level	High	ssip20_dreq_tx
46		SSIU	—	ssip21_drack_rx	DMA ACK	Level	High	ssip21_dreq_rx
47		SSIU	—	ssip21_drack_tx	DMA ACK	Level	High	ssip21_dreq_tx
48		SSIU	—	ssip22_drack_rx	DMA ACK	Level	High	ssip22_dreq_rx
49		SSIU	—	ssip22_drack_tx	DMA ACK	Level	High	ssip22_dreq_tx
50		SSIU	—	ssip23_drack_rx	DMA ACK	Level	High	ssip23_dreq_rx
51		SSIU	—	ssip23_drack_tx	DMA ACK	Level	High	ssip23_dreq_tx
52		SSIU	—	ssip30_drack_rx	DMA ACK	Level	High	ssip30_dreq_rx
53		SSIU	—	ssip30_drack_tx	DMA ACK	Level	High	ssip30_dreq_tx
54		SSIU	—	ssip31_drack_rx	DMA ACK	Level	High	ssip31_dreq_rx
55		SSIU	—	ssip31_drack_tx	DMA ACK	Level	High	ssip31_dreq_tx
56		SSIU	—	ssip32_drack_rx	DMA ACK	Level	High	ssip32_dreq_rx
57		SSIU	—	ssip32_drack_tx	DMA ACK	Level	High	ssip32_dreq_tx
58		SSIU	—	ssip33_drack_rx	DMA ACK	Level	High	ssip33_dreq_rx
59		SSIU	—	ssip33_drack_tx	DMA ACK	Level	High	ssip33_dreq_tx
60		SSIU	—	ssip40_drack_rx	DMA ACK	Level	High	ssip40_dreq_rx
61		SSIU	—	ssip40_drack_tx	DMA ACK	Level	High	ssip40_dreq_tx
62		SSIU	—	ssip41_drack_rx	DMA ACK	Level	High	ssip41_dreq_rx
63		SSIU	—	ssip41_drack_tx	DMA ACK	Level	High	ssip41_dreq_tx
64		SSIU	—	ssip42_drack_rx	DMA ACK	Level	High	ssip42_dreq_rx
65		SSIU	—	ssip42_drack_tx	DMA ACK	Level	High	ssip42_dreq_tx
66		SSIU	—	ssip43_drack_rx	DMA ACK	Level	High	ssip43_dreq_rx
67		SSIU	—	ssip43_drack_tx	DMA ACK	Level	High	ssip43_dreq_tx
68		SSIU	—	ssip5_drack_rx	DMA ACK	Level	High	ssip5_dreq_rx
69		SSIU	—	ssip5_drack_tx	DMA ACK	Level	High	ssip5_dreq_tx
70		SSIU	—	ssip6_drack_rx	DMA ACK	Level	High	ssip6_dreq_rx
71		SSIU	—	ssip6_drack_tx	DMA ACK	Level	High	ssip6_dreq_tx
72		SSIU	—	ssip7_drack_rx	DMA ACK	Level	High	ssip7_dreq_rx
73		SSIU	—	ssip7_drack_tx	DMA ACK	Level	High	ssip7_dreq_tx
74		SSIU	—	ssip8_drack_rx	DMA ACK	Level	High	ssip8_dreq_rx
75		SSIU	—	ssip8_drack_tx	DMA ACK	Level	High	ssip8_dreq_tx
76		SSIU	—	ssip90_drack_rx	DMA ACK	Level	High	ssip90_dreq_rx
77		SSIU	—	ssip90_drack_tx	DMA ACK	Level	High	ssip90_dreq_tx
78		SSIU	—	ssip91_drack_rx	DMA ACK	Level	High	ssip91_dreq_rx
79		SSIU	—	ssip91_drack_tx	DMA ACK	Level	High	ssip91_dreq_tx
80		SSIU	—	ssip92_drack_rx	DMA ACK	Level	High	ssip92_dreq_rx
81		SSIU	—	ssip92_drack_tx	DMA ACK	Level	High	ssip92_dreq_tx
82		SSIU	—	ssip93_drack_rx	DMA ACK	Level	High	ssip93_dreq_rx
83		SSIU	—	ssip93_drack_tx	DMA ACK	Level	High	ssip93_dreq_tx
84		PFC	—	DACK0	DMA ACK	Level	High	DREQ0
85		PFC	—	DACK1	DMA ACK	Level	High	DREQ1

Table 4.6-28 List of the DMA ACK Output Signals (3/3)

Ack No.	TEND No.	Unit	CH	Signal	Description	Level/Edge	Active Level	Corresponding Req Signal
86		PFC	—	DACK2	DMA ACK	Level	High	DREQ2
87		PFC	—	DACK3	DMA ACK	Level	High	DREQ3
88		PFC	—	DACK4	DMA ACK	Level	High	DREQ4
89		—	—	Reserved	—	—	—	—
90		—	—	Reserved	—	—	—	—
91		—	—	Reserved	—	—	—	—
	0	PFC	—	DTEND0	DMA transfer end	Edge	High	DREQ0
	1	PFC	—	DTEND1	DMA transfer end	Edge	High	DREQ1
	2	PFC	—	DTEND2	DMA transfer end	Edge	High	DREQ2
	3	PFC	—	DTEND3	DMA transfer end	Edge	High	DREQ3
	4	PFC	—	DTEND4	DMA transfer end	Edge	High	DREQ4

4.6.2 Generic Interrupt Controller (GIC)

GIC is equipped with Arm® CoreLink™ GIC-600 Generic Interrupt Controller for Cortex-A55.

4.6.2.1 Features

■ **Equipped with Arm® CoreLink™ GIC-600 Generic Interrupt Controller*1 for Arm Cortex-A55.**

- 32 levels of priority.
- SGI (Software Generated Interrupt): 16 interrupts
- PPI (Private Peripheral Interrupt): 16 interrupts
- SPI (Shared Peripheral Interrupt): 960 interrupts
- LPI (Locality-specific Peripheral Interrupt): Not supported

Note 1. Refer to Arm® CoreLink™ GIC-600 Generic Interrupt Controller Revision: r1p6 Technical Reference Manual.

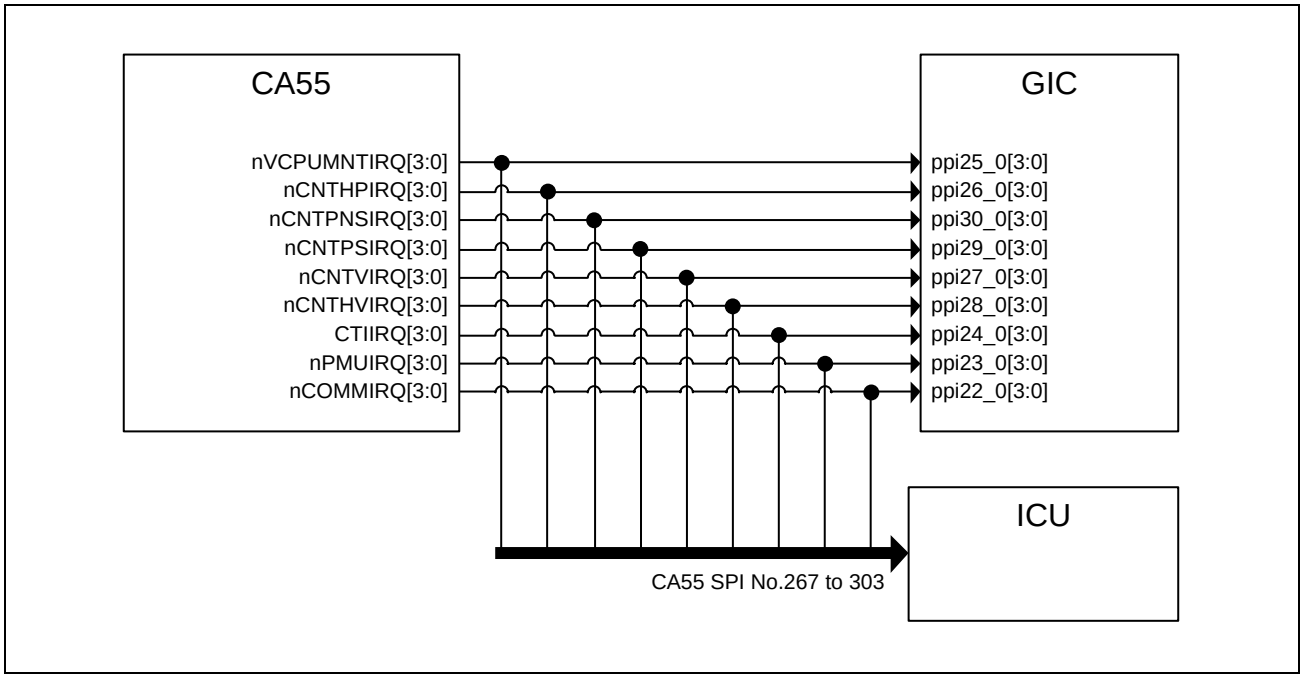


Figure 4.6-22 Diagram of SPI Signals

4.6.2.2 GIC-600 and NVIC Register Configuration

■ **GIC-600 Register**

Refer to Arm® CoreLink™ GIC-600 Generic Interrupt Controller Revision: r1p6 Technical Reference Manual.

ITS and LPI are not supported.

Base Address: 0_1490_0000h (Cortex-A55 Address Space)

Base Address: 0_4490_0000h (Cortex-M33 Address Space Secure)
Base Address: 0_5490_0000h (Cortex-M33 Address Space Non-secure)

4.6.2.3 GIC-600 and NVIC Register Descriptions

■ **GIC-600 Register**

Refer to Arm® CoreLink™ GIC-600 Generic Interrupt Controller Revision: r1p6 Technical Reference Manual for detail.

4.6.3 Message Handling Unit (MHU)

MHU is a function for message communication between each core of Cortex-A55(CA55) and Cortex-M33(CM33). Message communication is done by shared RAM (on-chip RAM and external memory) for passing messages and responses between CPUs and the function (MHU) for notifying when messages and responses are stored in the memory.

4.6.3.1 Features

Table 4.6-29 shows the MHU feature summary.

Table 4.6-29 Feature summary

Feature	Description
Interrupt by message and response	Generate interrupt to notify each core of CA55 and CM33 when the message/response is stored in shared RAM.
Software interrupt control	Control registers for interrupt setting, clearing, and status confirmation

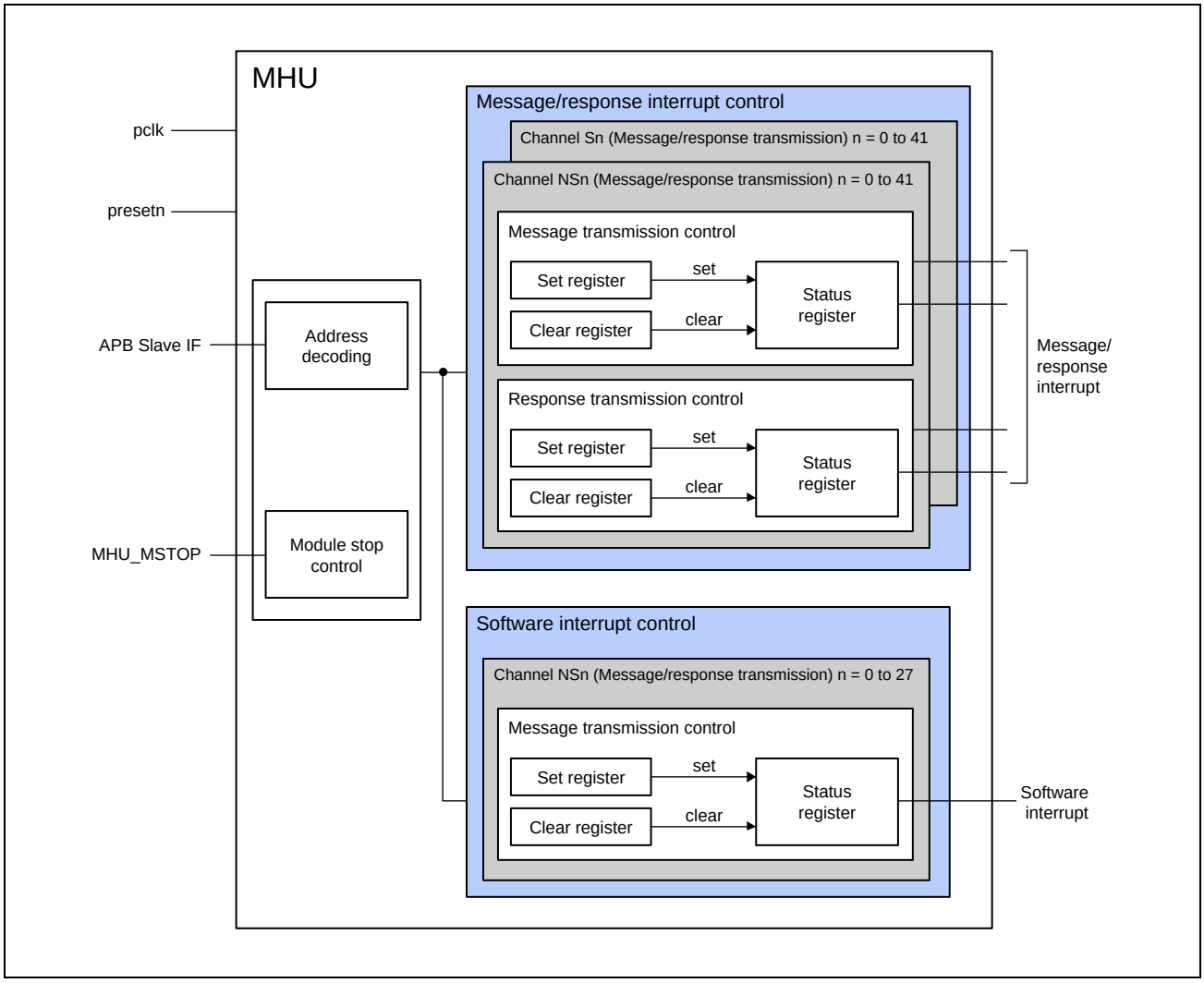


Figure 4.6-23 MHU Block Diagram

Message/response interrupt control registers

Registers that control message/response interrupts.

The message/response interrupt control register controls interrupts with the set register, clear register, and status register. An interrupt is asserted by writing “1” to the set register. The interrupt is negated by writing “1” to the clear register. The status can be checked by reading the status register. With these registers as one set, one channel is composed of a pair of the message transmission processing register and the response transmission processing register, and a total of 42 channels are mounted.

Software interrupt control register

Registers that control software interrupts.

The software interrupt control register controls interrupts using the set register, clear register, and status register. An interrupt is asserted by writing “1” to the set register. The interrupt is negated by writing “1” to the clear register. You can also check the interrupt status by reading the status register. With these registers as one set, one channel is composed of a pair of the message transmission processing register and the response transmission processing register, and a total of 28 channels are mounted.

Module stop control

This circuit refers to the module stop control (MHU_MSTOP) in CPG_MHU_MSTOP register on Clock Pulse Generator (CPG) and returns an error response if an access occurs while the module is stopped.

4.6.3.2 Interrupt

Table 4.6-30 shows the interrupt outputs.

Table 4.6-30 Interrupt type (1/5)

Output	Description	Pulse/Level	Active Level
msg_ch0_ns	Non secure message transmission interrupt (CA55 Core0 -> CA55 Core1)	Level	High
rsp_ch0_ns	Non secure response transmission interrupt (CA55 Core1 -> CA55 Core0)	Level	High
msg_ch1_ns	Non secure message transmission interrupt (CA55 Core0 -> CA55 Core2)	Level	High
rsp_ch1_ns	Non secure response transmission interrupt (CA55 Core2 -> CA55 Core0)	Level	High
msg_ch2_ns	Non secure message transmission interrupt (CA55 Core0 -> CA55 Core3)	Level	High
rsp_ch2_ns	Non secure response transmission interrupt (CA55 Core3 -> CA55 Core0)	Level	High
msg_ch3_ns	Not used	—	—
rsp_ch3_ns	Not used	—	—
msg_ch4_ns	Not used	—	—
rsp_ch4_ns	Not used	—	—
msg_ch5_ns	Non secure message transmission interrupt (CA55 Core0 -> CM33)	Level	High
rsp_ch5_ns	Non secure response transmission interrupt (CA55 Core0 -> CM33)	Level	High
msg_ch6_ns	Non secure message transmission interrupt (CA55 Core1 -> CA55 Core0)	Level	High
rsp_ch6_ns	Not used	—	—
msg_ch7_ns	Non secure message transmission interrupt (CA55 Core1 -> CA55 Core2)	Level	High
rsp_ch7_ns	Not used	—	—
msg_ch8_ns	Non secure message transmission interrupt (CA55 Core1 -> CA55 Core3)	Level	High
rsp_ch8_ns	Non secure response transmission interrupt (CM33 -> CA55 Core0)	Level	High
msg_ch9_ns	Not used	—	—
rsp_ch9_ns	Not used	—	—
msg_ch10_ns	Not used	—	—
rsp_ch10_ns	Not used	—	—
msg_ch11_ns	Non secure message transmission interrupt (CA55 Core1 -> CM33)	Level	High
rsp_ch11_ns	Non secure response transmission interrupt (CA55 Core1 -> CM33)	Level	High
msg_ch12_ns	Non secure message transmission interrupt (CA55 Core2 -> CA55 Core0)	Level	High
rsp_ch12_ns	Non secure response transmission interrupt (CA55 Core0 -> CA55 Core1)	Level	High
msg_ch13_ns	Non secure message transmission interrupt (CA55 Core2 -> CA55 Core1)	Level	High
rsp_ch13_ns	Non secure response transmission interrupt (CA55 Core2 -> CA55 Core1)	Level	High
msg_ch14_ns	Non secure message transmission interrupt (CA55 Core2 -> CA55 Core3)	Level	High
rsp_ch14_ns	Non secure response transmission interrupt (CA55 Core3 -> CA55 Core1)	Level	High
msg_ch15_ns	Not used	—	—
rsp_ch15_ns	Not used	—	—
msg_ch16_ns	Not used	—	—
rsp_ch16_ns	Not used	—	—
msg_ch17_ns	Non secure message transmission interrupt (CA55 Core2 -> CM33)	Level	High
rsp_ch17_ns	Non secure response transmission interrupt (CA55 Core2 -> CM33)	Level	High
msg_ch18_ns	Non secure message transmission interrupt (CA55 Core3 -> CA55 Core0)	Level	High
rsp_ch18_ns	Not used	—	—
msg_ch19_ns	Non secure message transmission interrupt (CA55 Core3 -> CA55 Core1)	Level	High
rsp_ch19_ns	Not used	—	—
msg_ch20_ns	Non secure message transmission interrupt (CA55 Core3 -> CA55 Core2)	Level	High

Table 4.6-30 Interrupt type (2/5)

Output	Description	Pulse/Level	Active Level
rsp_ch20_ns	Non secure response transmission interrupt (CM33 -> CA55 Core1)	Level	High
msg_ch21_ns	Not used	—	—
rsp_ch21_ns	Not used	—	—
msg_ch22_ns	Not used	—	—
rsp_ch22_ns	Not used	—	—
msg_ch23_ns	Non secure message transmission interrupt (CA55 Core3 -> CM33)	Level	High
rsp_ch23_ns	Non secure response transmission interrupt (CA55 Core3 -> CM33)	Level	High
msg_ch24_ns	Not used	—	—
rsp_ch24_ns	Non secure response transmission interrupt (CA55 Core0 -> CA55 Core2)	Level	High
msg_ch25_ns	Not used	—	—
rsp_ch25_ns	Non secure response transmission interrupt (CA55 Core1 -> CA55 Core2)	Level	High
msg_ch26_ns	Not used	—	—
rsp_ch26_ns	Non secure response transmission interrupt (CA55 Core3 -> CA55 Core2)	Level	High
msg_ch27_ns	Not used	—	—
rsp_ch27_ns	Not used	—	—
msg_ch28_ns	Not used	—	—
rsp_ch28_ns	Not used	—	—
msg_ch29_ns	Not used	—	—
rsp_ch29_ns	Not used	—	—
msg_ch30_ns	Not used	—	—
rsp_ch30_ns	Not used	—	—
msg_ch31_ns	Not used	—	—
rsp_ch31_ns	Non secure response transmission interrupt (CM33 -> CA55 Core2)	Level	High
msg_ch32_ns	Not used	—	—
rsp_ch32_ns	Non secure response transmission interrupt (CA55 Core0 -> CA55 Core3)	Level	High
msg_ch33_ns	Not used	—	—
rsp_ch33_ns	Non secure response transmission interrupt (CA55 Core1 -> CA55 Core3)	Level	High
msg_ch34_ns	Not used	—	—
rsp_ch34_ns	Not used	—	—
msg_ch35_ns	Not used	—	—
rsp_ch35_ns	Not used	—	—
msg_ch36_ns	Non secure message transmission interrupt (CM33 -> CA55 Core0)	Level	High
rsp_ch36_ns	Non secure response transmission interrupt (CA55 Core2 -> CA55 Core3)	Level	High
msg_ch37_ns	Non secure message transmission interrupt (CM33 -> CA55 Core1)	Level	High
rsp_ch37_ns	Not used	—	—
msg_ch38_ns	Non secure message transmission interrupt (CM33 -> CA55 Core2)	Level	High
rsp_ch38_ns	Not used	—	—
msg_ch39_ns	Non secure message transmission interrupt (CM33 -> CA55 Core3)	Level	High
rsp_ch39_ns	Non secure response transmission interrupt (CM33 -> CA55 Core3)	Level	High
msg_ch40_ns	Not used	—	—
rsp_ch40_ns	Not used	—	—
msg_ch41_ns	Not used	—	—
rsp_ch41_ns	Not used	—	—
msg_ch0_s	Secure message transmission interrupt (CA55 Core0 -> CA55 Core1)	Level	High
rsp_ch0_s	Secure response transmission interrupt (CA55 Core1 -> CA55 Core0)	Level	High

Table 4.6-30 Interrupt type (3/5)

Output	Description	Pulse/Level	Active Level
msg_ch1_s	Secure message transmission interrupt (CA55 Core0 -> CA55 Core2)	Level	High
rsp_ch1_s	Secure response transmission interrupt (CA55 Core2 -> CA55 Core0)	Level	High
msg_ch2_s	Secure message transmission interrupt (CA55 Core0 -> CA55 Core3)	Level	High
rsp_ch2_s	Secure response transmission interrupt (CA55 Core3 -> CA55 Core0)	Level	High
msg_ch3_s	Not used	—	—
rsp_ch3_s	Not used	—	—
msg_ch4_s	Not used	—	—
rsp_ch4_s	Not used	—	—
msg_ch5_s	Secure message transmission interrupt (CA55 Core0 -> CM33)	Level	High
rsp_ch5_s	Secure response transmission interrupt (CA55 Core0 -> CM33)	Level	High
msg_ch6_s	Secure message transmission interrupt (CA55 Core1 -> CA55 Core0)	Level	High
rsp_ch6_s	Not used	—	—
msg_ch7_s	Secure message transmission interrupt (CA55 Core1 -> CA55 Core2)	Level	High
rsp_ch7_s	Not used	—	—
msg_ch8_s	Secure message transmission interrupt (CA55 Core1 -> CA55 Core3)	Level	High
rsp_ch8_s	Secure response transmission interrupt (CM33 -> CA55 Core0)	Level	High
msg_ch9_s	Not used	—	—
rsp_ch9_s	Not used	—	—
msg_ch10_s	Not used	—	—
rsp_ch10_s	Not used	—	—
msg_ch11_s	Secure message transmission interrupt (CA55 Core1 -> CM33)	Level	High
rsp_ch11_s	Secure response transmission interrupt (CA55 Core1 -> CM33)	Level	High
msg_ch12_s	Secure message transmission interrupt (CA55 Core2 -> CA55 Core0)	Level	High
rsp_ch12_s	Secure response transmission interrupt (CA55 Core0 -> CA55 Core1)	Level	High
msg_ch13_s	Secure message transmission interrupt (CA55 Core2 -> CA55 Core1)	Level	High
rsp_ch13_s	Secure response transmission interrupt (CA55 Core2 -> CA55 Core1)	Level	High
msg_ch14_s	Secure message transmission interrupt (CA55 Core2 -> CA55 Core3)	Level	High
rsp_ch14_s	Secure response transmission interrupt (CA55 Core3 -> CA55 Core1)	Level	High
msg_ch15_s	Not used	—	—
rsp_ch15_s	Not used	—	—
msg_ch16_s	Not used	—	—
rsp_ch16_s	Not used	—	—
msg_ch17_s	Secure message transmission interrupt (CA55 Core2 -> CM33)	Level	High
rsp_ch17_s	Secure response transmission interrupt (CA55 Core2 -> CM33)	Level	High
msg_ch18_s	Secure message transmission interrupt (CA55 Core3 -> CA55 Core0)	Level	High
rsp_ch18_s	Not used	—	—
msg_ch19_s	Secure message transmission interrupt (CA55 Core3 -> CA55 Core1)	Level	High
rsp_ch19_s	Not used	—	—
msg_ch20_s	Secure message transmission interrupt (CA55 Core3 -> CA55 Core2)	Level	High
rsp_ch20_s	Secure response transmission interrupt (CM33 -> CA55 Core1)	Level	High
msg_ch21_s	Not used	—	—
rsp_ch21_s	Not used	—	—
msg_ch22_s	Not used	—	—
rsp_ch22_s	Not used	—	—
msg_ch23_s	Secure message transmission interrupt (CA55 Core3 -> CM33)	Level	High

Table 4.6-30 Interrupt type (4/5)

Output	Description	Pulse/Level	Active Level
rsp_ch23_s	Secure response transmission interrupt (CA55 Core3 -> CM33)	Level	High
msg_ch24_s	Not used	—	—
rsp_ch24_s	Secure response transmission interrupt (CA55 Core0 -> CA55 Core2)	Level	High
msg_ch25_s	Not used	—	—
rsp_ch25_s	Secure response transmission interrupt (CA55 Core1 -> CA55 Core2)	Level	High
msg_ch26_s	Not used	—	—
rsp_ch26_s	Secure response transmission interrupt (CA55 Core3 -> CA55 Core2)	Level	High
msg_ch27_s	Not used	—	—
rsp_ch27_s	Not used	—	—
msg_ch28_s	Not used	—	—
rsp_ch28_s	Not used	—	—
msg_ch29_s	Not used	—	—
rsp_ch29_s	Not used	—	—
msg_ch30_s	Not used	—	—
rsp_ch30_s	Not used	—	—
msg_ch31_s	Not used	—	—
rsp_ch31_s	Secure response transmission interrupt (CM33 -> CA55 Core2)	Level	High
msg_ch32_s	Not used	—	—
rsp_ch32_s	Secure response transmission interrupt (CA55 Core0 -> CA55 Core3)	Level	High
msg_ch33_s	Not used	—	—
rsp_ch33_s	Secure response transmission interrupt (CA55 Core1 -> CA55 Core3)	Level	High
msg_ch34_s	Not used	—	—
rsp_ch34_s	Not used	—	—
msg_ch35_s	Not used	—	—
rsp_ch35_s	Not used	—	—
msg_ch36_s	Secure message transmission interrupt (CM33 -> CA55 Core0)	Level	High
rsp_ch36_s	Secure response transmission interrupt (CA55 Core2 -> CA55 Core3)	Level	High
msg_ch37_s	Secure message transmission interrupt (CM33 -> CA55 Core1)	Level	High
rsp_ch37_s	Not used	—	—
msg_ch38_s	Secure message transmission interrupt (CM33 -> CA55 Core2)	Level	High
rsp_ch38_s	Not used	—	—
msg_ch39_s	Secure message transmission interrupt (CM33 -> CA55 Core3)	Level	High
rsp_ch39_s	Secure response transmission interrupt (CM33 -> CA55 Core3)	Level	High
msg_ch40_s	Not used	—	—
rsp_ch40_s	Not used	—	—
msg_ch41_s	Not used	—	—
rsp_ch41_s	Not used	—	—
swint_ch0_ns	Not used	—	—
swint_ch1_ns	Not used	—	—
swint_ch2_ns	Non secure software interrupt 2 (CA55 Core0 -> CM33)	Level	High
swint_ch3_ns	Not used	—	—
swint_ch4_ns	Not used	—	—
swint_ch5_ns	Non secure software interrupt 5 (CA55 Core1 -> CM33)	Level	High
swint_ch6_ns	Not used	—	—
swint_ch7_ns	Not used	—	—

Table 4.6-30 Interrupt type (5/5)

Output	Description	Pulse/Level	Active Level
swint_ch8_ns	Non secure software interrupt 8 (CA55 Core2 -> CM33)	Level	High
swint_ch9_ns	Not used	—	—
swint_ch10_ns	Not used	—	—
swint_ch11_ns	Non secure software interrupt 11 (CA55 Core3 -> CM33)	Level	High
swint_ch12_ns	Not used	—	—
swint_ch13_ns	Not used	—	—
swint_ch14_ns	Not used	—	—
swint_ch15_ns	Not used	—	—
swint_ch16_ns	Not used	—	—
swint_ch17_ns	Not used	—	—
swint_ch18_ns	Not used	—	—
swint_ch19_ns	Not used	—	—
swint_ch20_ns	Not used	—	—
swint_ch21_ns	Not used	—	—
swint_ch22_ns	Non secure software interrupt 22 (CM33 -> CA55 Core0)	Level	High
swint_ch23_ns	Non secure software interrupt 23 (CM33 -> CA55 Core1)	Level	High
swint_ch24_ns	Non secure software interrupt 24 (CM33 -> CA55 Core2)	Level	High
swint_ch25_ns	Non secure software interrupt 25 (CM33 -> CA55 Core3)	Level	High
swint_ch26_ns	Not used	—	—
swint_ch27_ns	Not used	—	—

4.6.3.3 Registers

Table 4.6-31 Register Base Address

Base Register Name	The Base Address
<MHU_base>	0_1048_0000h (5048_0000h*1, 4048_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

Table 4.6-32 shows offset address mapping for MHU.

Table 4.6-32 MHU Address map

Offset Address	Description
0000h to 07FFh	Non-secure message/response interrupt control register
0800h to 0FFFh	Non-secure software interrupt control register
1000h to 17FFh	Secure message/response interrupt control register
1800h to FFFFh	Reserved

Note: Access to the reserved area is prohibited.

4.6.3.3.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Non secure message transmission interrupt Status register	MHU_MSG_INT_STSn_NS <n=0-41>	0000_0000h	0000h + n x 0020h	32
Non secure message transmission interrupt Set register	MHU_MSG_INT_SETn_NS <n=0-41>	0000_0000h	0004h + n x 0020h	32
Non secure message transmission interrupt Clear register	MHU_MSG_INT_CLRn_NS <n=0-41>	0000_0000h	0008h + n x 0020h	32
Non secure response transmission interrupt Status register	MHU_RSP_INT_STSn_NS <n=0-41>	0000_0000h	0010h + n x 0020h	32
Non secure response transmission interrupt Set register	MHU_RSP_INT_SETn_NS <n=0-41>	0000_0000h	0014h + n x 0020h	32
Non secure response transmission interrupt Clear register	MHU_RSP_INT_CLRn_NS <n=0-41>	0000_0000h	0018h + n x 0020h	32
Reserve	-	-	0540h to 07FFh	-
Non secure software interrupt Status register	MHU_SW_INT_STSn_NS <n=0-27>	0000_0000h	0800h + n x 0010h	32
Non secure software interrupt Set register	MHU_SW_INT_SETn_NS <n=0-27>	0000_0000h	0804h + n x 0010h	32
Non secure software interrupt Clear register	MHU_SW_INT_CLRn_NS <n=0-27>	0000_0000h	0808h + n x 0010h	32
Reserve	-	-	09C0h to 0FFFh	-
Secure message transmission interrupt Status register	MHU_MSG_INT_STSn_S <n=0-41>	0000_0000h	1000h + n x 0020h	32
Secure message transmission interrupt Set register	MHU_MSG_INT_SETn_S <n=0-41>	0000_0000h	1004h + n x 0020h	32
Secure message transmission interrupt Clear register	MHU_MSG_INT_CLRn_S <n=0-41>	0000_0000h	1008h + n x 0020h	32
Secure response transmission interrupt Status register	MHU_RSP_INT_STSn_S <n=0-41>	0000_0000h	1010h + n x 0020h	32
Secure response transmission interrupt Set register	MHU_RSP_INT_SETn_S <n=0-41>	0000_0000h	1014h + n x 0020h	32
Secure response transmission interrupt Clear register	MHU_RSP_INT_CLRn_S <n=0-41>	0000_0000h	1018h + n x 0020h	32

4.6.3.3.2 Register Description

The prefix (MHU_) of the register names is omitted in this and subsequent sections.

(1) Non-Secure Message Transmission Interrupt Status Register (MHU_MSG_INT_STSn_NS) (n = 0 to 41)

- n=0: CA55 Core0 -> CA55 Core1
- n=1: CA55 Core0 -> CA55 Core2
- n=2: CA55 Core0 -> CA55 Core3
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core0 -> CM33
- n=6: CA55 Core1 -> CA55 Core0
- n=7: CA55 Core1 -> CA55 Core2
- n=8: CA55 Core1 -> CA55 Core3
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core1 -> CM33
- n=12: CA55 Core2 -> CA55 Core0
- n=13: CA55 Core2 -> CA55 Core1
- n=14: CA55 Core2 -> CA55 Core3
- n=15: Not used
- n=16: Not used
- n=17: CA55 Core2 -> CM33
- n=18: CA55 Core3 -> CA55 Core0
- n=19: CA55 Core3 -> CA55 Core1
- n=20: CA55 Core3 -> CA55 Core2
- n=21: Not used
- n=22: Not used
- n=23: CA55 Core3 -> CM33
- n=24: Not used
- n=25: Not used
- n=26: Not used
- n=27: Not used
- n=28: Not used
- n=29: Not used
- n=30: Not used
- n=31: Not used
- n=32: Not used
- n=33: Not used
- n=34: Not used
- n=35: Not used
- n=36: CM33 -> CA55 Core0
- n=37: CM33 -> CA55 Core1
- n=38: CM33 -> CA55 Core2
- n=39: CM33 -> CA55 Core3
- n=40: Not used
- n=41: Not used

Status register for non-secure message transmission interrupt

Access Size : 32 bits
Address : <MHU_base> + 0000h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	STAT	0h	R	Indicates the status of the message transmission interrupt. This bit is read-only. Writing to it has no effect. 0b: No message transmission interrupt (no interrupt is notified to the interrupt controller of the message destination). 1b: Assert the message transmission interrupt (the interrupt is notified to the interrupt controller of the message destination).

(2) Non-Secure Message Transmission Interrupt Set Register (MHU_MSG_INT_SETn_NS) (n = 0 to 41)

- n=0: CA55 Core0 -> CA55 Core1
- n=1: CA55 Core0 -> CA55 Core2
- n=2: CA55 Core0 -> CA55 Core3
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core0 -> CM33
- n=6: CA55 Core1 -> CA55 Core0
- n=7: CA55 Core1 -> CA55 Core2
- n=8: CA55 Core1 -> CA55 Core3
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core1 -> CM33
- n=12: CA55 Core2 -> CA55 Core0
- n=13: CA55 Core2 -> CA55 Core1
- n=14: CA55 Core2 -> CA55 Core3
- n=15: Not used
- n=16: Not used
- n=17: CA55 Core2 -> CM33
- n=18: CA55 Core3 -> CA55 Core0
- n=19: CA55 Core3 -> CA55 Core1
- n=20: CA55 Core3 -> CA55 Core2
- n=21: Not used
- n=22: Not used
- n=23: CA55 Core3 -> CM33
- n=24: Not used
- n=25: Not used
- n=26: Not used
- n=27: Not used
- n=28: Not used
- n=29: Not used
- n=30: Not used
- n=31: Not used
- n=32: Not used
- n=33: Not used
- n=34: Not used
- n=35: Not used
- n=36: CM33 -> CA55 Core0
- n=37: CM33 -> CA55 Core1
- n=38: CM33 -> CA55 Core2
- n=39: CM33 -> CA55 Core3
- n=40: Not used
- n=41: Not used

Set register for non-secure message transmission interrupt

Access Size : 32 bits
Address : <MHU_base> + 0004h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	SET	0h	R0W1	Sets the message transmission interrupt. Only 1b can be written to this bit. It is always read as 0b. 0b: Invalid 1b: Issue the interrupt to the interrupt controller of the message destination.

(3) Non-Secure Message Transmission Interrupt Clear Register (MHU_MSG_INT_CLRn_NS) (n = 0 to 41)

- n=0: CA55 Core0 -> CA55 Core1
- n=1: CA55 Core0 -> CA55 Core2
- n=2: CA55 Core0 -> CA55 Core3
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core0 -> CM33
- n=6: CA55 Core1 -> CA55 Core0
- n=7: CA55 Core1 -> CA55 Core2
- n=8: CA55 Core1 -> CA55 Core3
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core1 -> CM33
- n=12: CA55 Core2 -> CA55 Core0
- n=13: CA55 Core2 -> CA55 Core1
- n=14: CA55 Core2 -> CA55 Core3
- n=15: Not used
- n=16: Not used
- n=17: CA55 Core2 -> CM33
- n=18: CA55 Core3 -> CA55 Core0
- n=19: CA55 Core3 -> CA55 Core1
- n=20: CA55 Core3 -> CA55 Core2
- n=21: Not used
- n=22: Not used
- n=23: CA55 Core3 -> CM33
- n=24: Not used
- n=25: Not used
- n=26: Not used
- n=27: Not used
- n=28: Not used
- n=29: Not used
- n=30: Not used
- n=31: Not used
- n=32: Not used
- n=33: Not used
- n=34: Not used
- n=35: Not used
- n=36: CM33 -> CA55 Core0
- n=37: CM33 -> CA55 Core1
- n=38: CM33 -> CA55 Core2
- n=39: CM33 -> CA55 Core3
- n=40: Not used
- n=41: Not used

Clear register for non-secure message transmission interrupt

Access Size : 32 bits
Address : <MHU_base> + 0008h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	CLEAR	0h	R0W1	Clears the message transmission interrupt. Only 1b can be written to this bit. It is always read as 0b. 0b: Invalid 1b: Clear the interrupt to the interrupt controller of the message destination.

(4) Non-Secure Response Transmission Interrupt Status Register (MHU_RSP_INT_STSn_NS) (n = 0 to 41)

- n=0: CA55 Core1 -> CA55 Core0
- n=1: CA55 Core2 -> CA55 Core0
- n=2: CA55 Core3 -> CA55 Core0
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core0 -> CM33
- n=6: Not used
- n=7: Not used
- n=8: CM33 -> CA55 Core0
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core1 -> CM33
- n=12: CA55 Core0 -> CA55 Core1
- n=13: CA55 Core2 -> CA55 Core1
- n=14: CA55 Core3 -> CA55 Core1
- n=15: Not used
- n=16: Not used
- n=17: CA55 Core2 -> CM33
- n=18: Not used
- n=19: Not used
- n=20: CM33 -> CA55 Core1
- n=21: Not used
- n=22: Not used
- n=23: CA55 Core3 -> CM33
- n=24: CA55 Core0 -> CA55 Core2
- n=25: CA55 Core1 -> CA55 Core2
- n=26: CA55 Core2 -> CA55 Core2
- n=27: Not used
- n=28: Not used
- n=29: Not used
- n=30: Not used
- n=31: CM33 -> CA55 Core2
- n=32: CA55 Core0 -> CA55 Core3
- n=33: CA55 Core1 -> CA55 Core3
- n=34: Not used
- n=35: Not used
- n=36: CA55 Core2 -> CA55 Core3
- n=37: Not used
- n=38: Not used
- n=39: CM33 -> CA55 Core3
- n=40: Not used
- n=41: Not used

Status register for non-secure response transmission interrupt

Access Size : 32 bits
Address : <MHU_base> + 0010h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	STAT	0h	R	Indicates the status of the message transmission interrupt. This bit is read-only. Writing to it has no effect. 0b: No message transmission interrupt (no interrupt is notified to the interrupt controller of the message destination). 1b: Assert the message transmission interrupt (the interrupt is notified to the interrupt controller of the message destination).

(5) Non-Secure Response Transmission Interrupt Set Register (MHU_RSP_INT_SETn_NS) (n = 0 to 41)

- n=0: CA55 Core1 -> CA55 Core0
- n=1: CA55 Core2 -> CA55 Core0
- n=2: CA55 Core3 -> CA55 Core0
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core0 -> CM33
- n=6: Not used
- n=7: Not used
- n=8: CM33 -> CA55 Core0
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core1 -> CM33
- n=12: CA55 Core0 -> CA55 Core1
- n=13: CA55 Core2 -> CA55 Core1
- n=14: CA55 Core3 -> CA55 Core1
- n=15: Not used
- n=16: Not used
- n=17: CA55 Core2 -> CM33
- n=18: Not used
- n=19: Not used
- n=20: CM33 -> CA55 Core1
- n=21: Not used
- n=22: Not used
- n=23: CA55 Core3 -> CM33
- n=24: CA55 Core0 -> CA55 Core2
- n=25: CA55 Core1 -> CA55 Core2
- n=26: CA55 Core2 -> CA55 Core2
- n=27: Not used
- n=28: Not used
- n=29: Not used
- n=30: Not used
- n=31: CM33 -> CA55 Core2
- n=32: CA55 Core0 -> CA55 Core3
- n=33: CA55 Core1 -> CA55 Core3
- n=34: Not used
- n=35: Not used
- n=36: CA55 Core2 -> CA55 Core3
- n=37: Not used
- n=38: Not used
- n=39: CM33 -> CA55 Core3
- n=40: Not used
- n=41: Not used

Set register for non-secure response transmission interrupt

Access Size : 32 bits
Address : <MHU_base> + 0014h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	SET	0h	R0W1	Sets the message transmission interrupt. Only 1b can be written to this bit. It is always read as 0b. 0b: Invalid 1b: Issue the interrupt to the interrupt controller of the message destination.

(6) Non-Secure Response Transmission Interrupt Clear Register (MHU_RSP_INT_CLRn_NS) (n = 0 to 41)

- n=0: CA55 Core1 -> CA55 Core0
- n=1: CA55 Core2 -> CA55 Core0
- n=2: CA55 Core3 -> CA55 Core0
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core0 -> CM33
- n=6: Not used
- n=7: Not used
- n=8: CM33 -> CA55 Core0
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core1 -> CM33
- n=12: CA55 Core0 -> CA55 Core1
- n=13: CA55 Core2 -> CA55 Core1
- n=14: CA55 Core3 -> CA55 Core1
- n=15: Not used
- n=16: Not used
- n=17: CA55 Core2 -> CM33
- n=18: Not used
- n=19: Not used
- n=20: CM33 -> CA55 Core1
- n=21: Not used
- n=22: Not used
- n=23: CA55 Core3 -> CM33
- n=24: CA55 Core0 -> CA55 Core2
- n=25: CA55 Core1 -> CA55 Core2
- n=26: CA55 Core2 -> CA55 Core2
- n=27: Not used
- n=28: Not used
- n=29: Not used
- n=30: Not used
- n=31: CM33 -> CA55 Core2
- n=32: CA55 Core0 -> CA55 Core3
- n=33: CA55 Core1 -> CA55 Core3
- n=34: Not used
- n=35: Not used
- n=36: CA55 Core2 -> CA55 Core3
- n=37: Not used
- n=38: Not used
- n=39: CM33 -> CA55 Core3
- n=40: Not used
- n=41: Not used

Clear register for non-secure response transmission interrupt

Access Size : 32 bits
Address : <MHU_base> + 0018h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	CLEAR	0h	R0W1	Clears the message transmission interrupt. Only 1b can be written to this bit. It is always read as 0b. 0b: Invalid 1b: Clear the interrupt to the interrupt controller of the message destination.

(7) Non-Secure Software Interrupt Status Register (Ch0 to Ch27) (MHU_SW_INT_STSn_NS) (n = 0 to 27)

- n=0: Not used
- n=1: Not used
- n=2: CA55 Core0 -> CM33
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core1 -> CM33
- n=6: Not used
- n=7: Not used
- n=8: CA55 Core2 -> CM33
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core3 => CM33
- n=12: Not used
- n=13: Not used
- n=14: Not used
- n=15: Not used
- n=16: Not used
- n=17: Not used
- n=18: Not used
- n=19: Not used
- n=20: Not used
- n=21: Not used
- n=22: CM33 -> CA55 Core0
- n=23: CM33 -> CA55 Core1
- n=24: CM33 -> CA55 Core2
- n=25: CM33 -> CA55 Core3
- n=26: Not used
- n=27: Not used

Status register for non-secure software interrupt status register

Access Size : 32 bits
Address : <MHU_base> + 0800h + n x 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	STAT	0h	R	Indicates the status of the non-secure software interrupt. Writing to this bit has no effect. 0b: No interrupt (no interrupt is notified to the interrupt controller). 1b: Software interrupt is asserted (the interrupt is notified to the interrupt controller).

(8) Non-Secure Software Interrupt Set Register (Ch0 to Ch27) (MHU_SW_INT_SETn_NS) (n = 0 to 27)

- n=0: Not used
- n=1: Not used
- n=2: CA55 Core0 -> CM33
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core1 -> CM33
- n=6: Not used
- n=7: Not used
- n=8: CA55 Core2 -> CM33
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core3 => CM33
- n=12: Not used
- n=13: Not used
- n=14: Not used
- n=15: Not used
- n=16: Not used
- n=17: Not used
- n=18: Not used
- n=19: Not used
- n=20: Not used
- n=21: Not used
- n=22: CM33 -> CA55 Core0
- n=23: CM33 -> CA55 Core1
- n=24: CM33 -> CA55 Core2
- n=25: CM33 -> CA55 Core3
- n=26: Not used
- n=27: Not used

Set register for non-secure software interrupt

Access Size : 32 bits
Address : <MHU_base> + 0804h + n x 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	SET	0h	R0W1	Sets the non-secure software interrupt. Only 1b can be written to this bit. It is always read as 0b. 0b: Invalid 1b: Issue the interrupt to the interrupt controller.

(9) Non-Secure Software Interrupt Clear Register (ch0 to ch27) (MHU_SW_INT_CLRn_NS) (n = 0 to 27)

- n=0: Not used
- n=1: Not used
- n=2: CA55 Core0 -> CM33
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core1 -> CM33
- n=6: Not used
- n=7: Not used
- n=8: CA55 Core2 -> CM33
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core3 => CM33
- n=12: Not used
- n=13: Not used
- n=14: Not used
- n=15: Not used
- n=16: Not used
- n=17: Not used
- n=18: Not used
- n=19: Not used
- n=20: Not used
- n=21: Not used
- n=22: CM33 -> CA55 Core0
- n=23: CM33 -> CA55 Core1
- n=24: CM33 -> CA55 Core2
- n=25: CM33 -> CA55 Core3
- n=26: Not used
- n=27: Not used

Clear register for non-secure software interrupt

Access Size : 32 bits
Address : <MHU_base> + 0808h + n x 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	CLEAR	0h	R0W1	Clears the non-secure software interrupt. Only 1b can be written to this bit. It is always read as 0b. 0b: Invalid 1b: Clear the interrupt to the interrupt controller.

(10) Secure Message Transmission Interrupt Status Register (MHU_ MSG_INT_STSn_S) (n = 0 to 41)

- n=0: CA55 Core0 -> CA55 Core1
- n=1: CA55 Core0 -> CA55 Core2
- n=2: CA55 Core0 -> CA55 Core3
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core0 -> CM33
- n=6: CA55 Core1 -> CA55 Core0
- n=7: CA55 Core1 -> CA55 Core2
- n=8: CA55 Core1 -> CA55 Core3
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core1 -> CM33
- n=12: CA55 Core2 -> CA55 Core0
- n=13: CA55 Core2 -> CA55 Core1
- n=14: CA55 Core2 -> CA55 Core3
- n=15: Not used
- n=16: Not used
- n=17: CA55 Core2 -> CM33
- n=18: CA55 Core3 -> CA55 Core0
- n=19: CA55 Core3 -> CA55 Core1
- n=20: CA55 Core3 -> CA55 Core2
- n=21: Not used
- n=22: Not used
- n=23: CA55 Core3 -> CM33
- n=24: Not used
- n=25: Not used
- n=26: Not used
- n=27: Not used
- n=28: Not used
- n=29: Not used
- n=30: Not used
- n=31: Not used
- n=32: Not used
- n=33: Not used
- n=34: Not used
- n=35: Not used
- n=36: CM33 -> CA55 Core0
- n=37: CM33 -> CA55 Core1
- n=38: CM33 -> CA55 Core2
- n=39: CM33 -> CA55 Core3
- n=40: Not used
- n=41: Not used

Status register for secure message transmission interrupt

Access Size : 32 bits

Address : <MHU_base> + 1000h + n x 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	STAT	0h	R	Indicates the status of the message transmission interrupt. This bit is read-only. Writing to it has no effect. 0b: No message transmission interrupt (no interrupt is notified to the interrupt controller of the message destination). 1b: Assert the message transmission interrupt (the interrupt is notified to the interrupt controller of the message destination).

(11) Secure Message Transmission Interrupt Set Register (MHU_MSG_INT_SETn_S) (n = 0 to 41)

- n=0: CA55 Core0 -> CA55 Core1
- n=1: CA55 Core0 -> CA55 Core2
- n=2: CA55 Core0 -> CA55 Core3
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core0 -> CM33
- n=6: CA55 Core1 -> CA55 Core0
- n=7: CA55 Core1 -> CA55 Core2
- n=8: CA55 Core1 -> CA55 Core3
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core1 -> CM33
- n=12: CA55 Core2 -> CA55 Core0
- n=13: CA55 Core2 -> CA55 Core1
- n=14: CA55 Core2 -> CA55 Core3
- n=15: Not used
- n=16: Not used
- n=17: CA55 Core2 -> CM33
- n=18: CA55 Core3 -> CA55 Core0
- n=19: CA55 Core3 -> CA55 Core1
- n=20: CA55 Core3 -> CA55 Core2
- n=21: Not used
- n=22: Not used
- n=23: CA55 Core3 -> CM33
- n=24: Not used
- n=25: Not used
- n=26: Not used
- n=27: Not used
- n=28: Not used
- n=29: Not used
- n=30: Not used
- n=31: Not used
- n=32: Not used
- n=33: Not used
- n=34: Not used
- n=35: Not used
- n=36: CM33 -> CA55 Core0
- n=37: CM33 -> CA55 Core1
- n=38: CM33 -> CA55 Core2
- n=39: CM33 -> CA55 Core3
- n=40: Not used
- n=41: Not used

Set register for secure message transmission interrupt

Access Size : 32 bits
Address : <MHU_base> + 1004h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	SET	0h	R0W1	Sets the message transmission interrupt. Only 1b can be written to this bit. It is always read as 0b. 0b: Invalid 1b: Issue the interrupt to the interrupt controller of the message destination.

(12) Secure Message Transmission Interrupt Clear Register (MHU_MSG_INT_CLRn_S) (n = 0 to 41)

- n=0: CA55 Core0 -> CA55 Core1
- n=1: CA55 Core0 -> CA55 Core2
- n=2: CA55 Core0 -> CA55 Core3
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core0 -> CM33
- n=6: CA55 Core1 -> CA55 Core0
- n=7: CA55 Core1 -> CA55 Core2
- n=8: CA55 Core1 -> CA55 Core3
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core1 -> CM33
- n=12: CA55 Core2 -> CA55 Core0
- n=13: CA55 Core2 -> CA55 Core1
- n=14: CA55 Core2 -> CA55 Core3
- n=15: Not used
- n=16: Not used
- n=17: CA55 Core2 -> CM33
- n=18: CA55 Core3 -> CA55 Core0
- n=19: CA55 Core3 -> CA55 Core1
- n=20: CA55 Core3 -> CA55 Core2
- n=21: Not used
- n=22: Not used
- n=23: CA55 Core3 -> CM33
- n=24: Not used
- n=25: Not used
- n=26: Not used
- n=27: Not used
- n=28: Not used
- n=29: Not used
- n=30: Not used
- n=31: Not used
- n=32: Not used
- n=33: Not used
- n=34: Not used
- n=35: Not used
- n=36: CM33 -> CA55 Core0
- n=37: CM33 -> CA55 Core1
- n=38: CM33 -> CA55 Core2
- n=39: CM33 -> CA55 Core3
- n=40: Not used
- n=41: Not used

Clear register for secure message transmission interrupt

Access Size : 32 bits
Address : <MHU_base> + 1008h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	CLEAR	0h	R0W1	Clears the message transmission interrupt. Only 1b can be written to this bit. It is always read as 0b. 0b: Invalid 1b: Clear the interrupt to the interrupt controller of the message destination.

(13) Secure Response Transmission Interrupt Status Register (MHU_RSP_INT_STSn_S) (n = 0 to 41)

- n=0: CA55 Core1 -> CA55 Core0
- n=1: CA55 Core2 -> CA55 Core0
- n=2: CA55 Core3 -> CA55 Core0
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core0 -> CM33
- n=6: Not used
- n=7: Not used
- n=8: CM33 -> CA55 Core0
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core1 -> CM33
- n=12: CA55 Core0 -> CA55 Core1
- n=13: CA55 Core2 -> CA55 Core1
- n=14: CA55 Core3 -> CA55 Core1
- n=15: Not used
- n=16: Not used
- n=17: CA55 Core2 -> CM33
- n=18: Not used
- n=19: Not used
- n=20: CM33 -> CA55 Core1
- n=21: Not used
- n=22: Not used
- n=23: CA55 Core3 -> CM33
- n=24: CA55 Core0 -> CA55 Core2
- n=25: CA55 Core1 -> CA55 Core2
- n=26: CA55 Core2 -> CA55 Core2
- n=27: Not used
- n=28: Not used
- n=29: Not used
- n=30: Not used
- n=31: CM33 -> CA55 Core2
- n=32: CA55 Core0 -> CA55 Core3
- n=33: CA55 Core1 -> CA55 Core3
- n=34: Not used
- n=35: Not used
- n=36: CA55 Core2 -> CA55 Core3
- n=37: Not used
- n=38: Not used
- n=39: CM33 -> CA55 Core3
- n=40: Not used
- n=41: Not used

Status register for secure response transmission interrupt

Access Size : 32 bits
Address : <MHU_base> + 1010h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STAT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	STAT	0h	R	Indicates the status of the message transmission interrupt. This bit is read-only. Writing to it has no effect. 0b: No message transmission interrupt (no interrupt is notified to the interrupt controller of the message destination). 1b: Assert the message transmission interrupt (the interrupt is notified to the interrupt controller of the message destination).

(14) Secure Response Transmission Interrupt Set Register (MHU_RSP_INT_SETn_S) (n = 0 to 41)

- n=0: CA55 Core1 -> CA55 Core0
- n=1: CA55 Core2 -> CA55 Core0
- n=2: CA55 Core3 -> CA55 Core0
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core0 -> CM33
- n=6: Not used
- n=7: Not used
- n=8: CM33 -> CA55 Core0
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core1 -> CM33
- n=12: CA55 Core0 -> CA55 Core1
- n=13: CA55 Core2 -> CA55 Core1
- n=14: CA55 Core3 -> CA55 Core1
- n=15: Not used
- n=16: Not used
- n=17: CA55 Core2 -> CM33
- n=18: Not used
- n=19: Not used
- n=20: CM33 -> CA55 Core1
- n=21: Not used
- n=22: Not used
- n=23: CA55 Core3 -> CM33
- n=24: CA55 Core0 -> CA55 Core2
- n=25: CA55 Core1 -> CA55 Core2
- n=26: CA55 Core2 -> CA55 Core2
- n=27: Not used
- n=28: Not used
- n=29: Not used
- n=30: Not used
- n=31: CM33 -> CA55 Core2
- n=32: CA55 Core0 -> CA55 Core3
- n=33: CA55 Core1 -> CA55 Core3
- n=34: Not used
- n=35: Not used
- n=36: CA55 Core2 -> CA55 Core3
- n=37: Not used
- n=38: Not used
- n=39: CM33 -> CA55 Core3
- n=40: Not used
- n=41: Not used

Set register for secure response transmission interrupt

Access Size : 32 bits
Address : <MHU_base> + 1014h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SET
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	SET	0h	R0W1	Sets the message transmission interrupt. Only 1b can be written to this bit. It is always read as 0b. 0b: Invalid 1b: Issue the interrupt to the interrupt controller of the message destination.

(15) Secure Response Transmission Interrupt Clear Register (MHU_RSP_INT_CLRn_S) (n = 0 to 41)

- n=0: CA55 Core1 -> CA55 Core0
- n=1: CA55 Core2 -> CA55 Core0
- n=2: CA55 Core3 -> CA55 Core0
- n=3: Not used
- n=4: Not used
- n=5: CA55 Core0 -> CM33
- n=6: Not used
- n=7: Not used
- n=8: CM33 -> CA55 Core0
- n=9: Not used
- n=10: Not used
- n=11: CA55 Core1 -> CM33
- n=12: CA55 Core0 -> CA55 Core1
- n=13: CA55 Core2 -> CA55 Core1
- n=14: CA55 Core3 -> CA55 Core1
- n=15: Not used
- n=16: Not used
- n=17: CA55 Core2 -> CM33
- n=18: Not used
- n=19: Not used
- n=20: CM33 -> CA55 Core1
- n=21: Not used
- n=22: Not used
- n=23: CA55 Core3 -> CM33
- n=24: CA55 Core0 -> CA55 Core2
- n=25: CA55 Core1 -> CA55 Core2
- n=26: CA55 Core2 -> CA55 Core2
- n=27: Not used
- n=28: Not used
- n=29: Not used
- n=30: Not used
- n=31: CM33 -> CA55 Core2
- n=32: CA55 Core0 -> CA55 Core3
- n=33: CA55 Core1 -> CA55 Core3
- n=34: Not used
- n=35: Not used
- n=36: CA55 Core2 -> CA55 Core3
- n=37: Not used
- n=38: Not used
- n=39: CM33 -> CA55 Core3
- n=40: Not used
- n=41: Not used

Clear register for secure response transmission interrupt

Access Size : 32 bits
Address : <MHU_base> + 1018h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CLEAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	CLEAR	0h	R0W1	Clears the message transmission interrupt. Only 1b can be written to this bit. It is always read as 0b. 0b: Invalid 1b: Clear the interrupt to the interrupt controller of the message destination.

4.6.3.4 Function Description

4.6.3.4.1 Inter-CPU communication

The hardware configuration is shown in **Figure 4.6-24**. Inter-CPU communication is established by sending a message from the message source to the message destination and sending a response from the message destination to the message source. The MHU uses an interrupt to notify that the message or response to the CPU has been written to the shared RAM. Interrupts are controlled by the Set register, Clear register, and Status register. Interrupt can be asserted by writing “1” to the Set register. Interrupt is negated by writing “1” to the Clear register, and check the interrupt status by reading the Status register. With these registers as one set, one channel is composed of a pair of the message transmission processing register and the response transmission processing register. A total of 84 channels (non-secure: 42 channels; secure: 42 channels) are installed so that bidirectional communication can be performed between CA55 Core0/Core1/Core2/Core3 and CM33. **Table 4.6-33** shows the message source/destination correspondence.

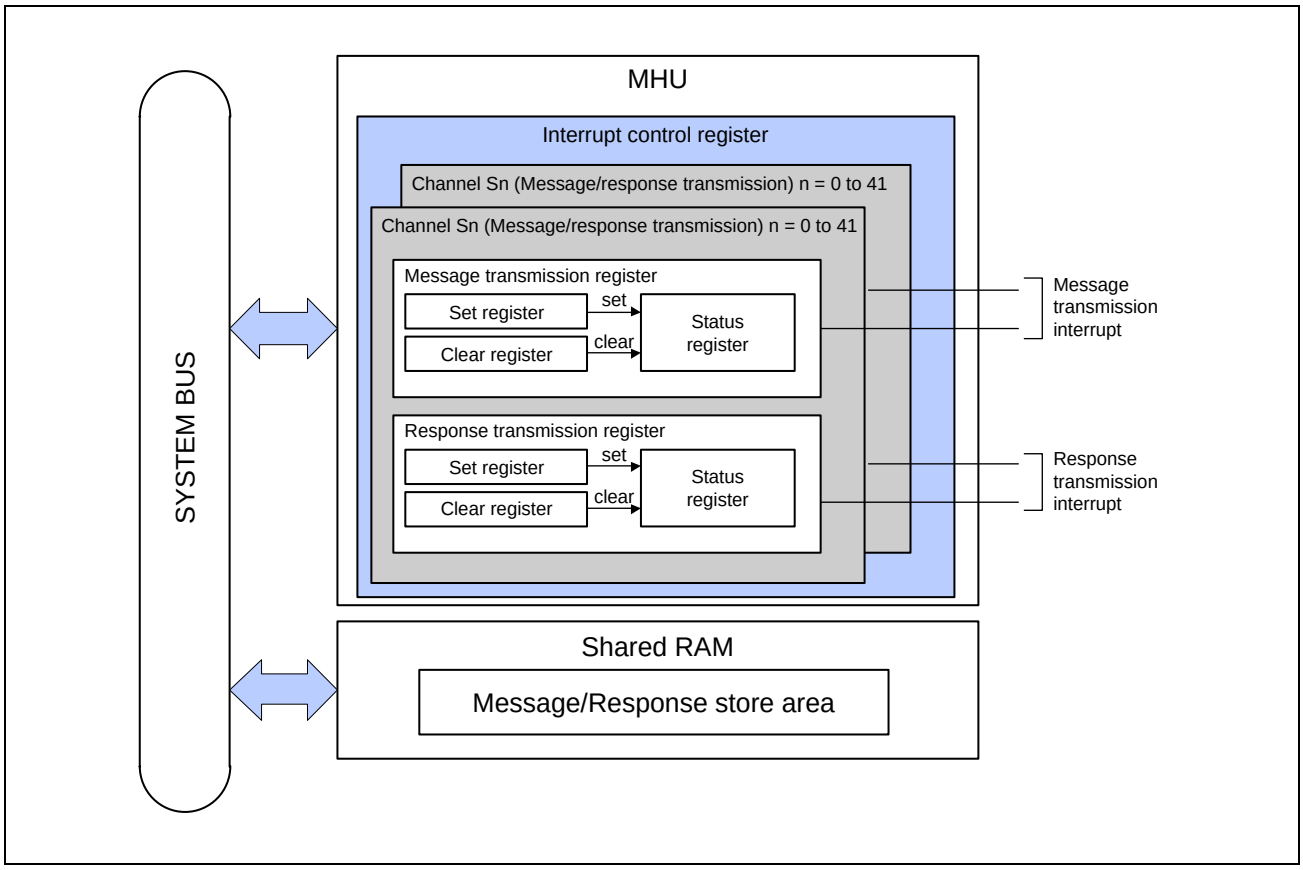


Figure 4.6-24 Hardware Configuration

Table 4.6-33 Message destination/source reference

Source	Destination	Message transmission register	Response transmission register
CA55 Core0	CA55 Core1	MSG_INT_(SEL/CLR/STS)0_NS	RSP_INT_(SET/CLR/STS)0_NS
CA55 Core0	CA55 Core2	MSG_INT_(SEL/CLR/STS)1_NS	RSP_INT_(SET/CLR/STS)1_NS
CA55 Core0	CA55 Core3	MSG_INT_(SEL/CLR/STS)2_NS	RSP_INT_(SET/CLR/STS)2_NS
CA55 Core0	CM33	MSG_INT_(SEL/CLR/STS)5_NS	RSP_INT_(SET/CLR/STS)8_NS
CA55 Core1	CA55 Core0	MSG_INT_(SEL/CLR/STS)6_NS	RSP_INT_(SET/CLR/STS)12_NS
CA55 Core1	CA55 Core2	MSG_INT_(SEL/CLR/STS)7_NS	RSP_INT_(SET/CLR/STS)13_NS
CA55 Core1	CA55 Core3	MSG_INT_(SEL/CLR/STS)8_NS	RSP_INT_(SET/CLR/STS)14_NS
CA55 Core1	CM33	MSG_INT_(SEL/CLR/STS)11_NS	RSP_INT_(SET/CLR/STS)20_NS
CA55 Core2	CA55 Core0	MSG_INT_(SEL/CLR/STS)12_NS	RSP_INT_(SET/CLR/STS)24_NS
CA55 Core2	CA55 Core1	MSG_INT_(SEL/CLR/STS)13_NS	RSP_INT_(SET/CLR/STS)25_NS
CA55 Core2	CA55 Core3	MSG_INT_(SEL/CLR/STS)14_NS	RSP_INT_(SET/CLR/STS)26_NS
CA55 Core2	CM33	MSG_INT_(SEL/CLR/STS)17_NS	RSP_INT_(SET/CLR/STS)31_NS
CA55 Core3	CA55 Core0	MSG_INT_(SEL/CLR/STS)18_NS	RSP_INT_(SET/CLR/STS)32_NS
CA55 Core3	CA55 Core1	MSG_INT_(SEL/CLR/STS)19_NS	RSP_INT_(SET/CLR/STS)33_NS
CA55 Core3	CA55 Core2	MSG_INT_(SEL/CLR/STS)20_NS	RSP_INT_(SET/CLR/STS)36_NS
CA55 Core3	CM33	MSG_INT_(SEL/CLR/STS)23_NS	RSP_INT_(SET/CLR/STS)39_NS
CM33	CA55 Core0	MSG_INT_(SEL/CLR/STS)36_NS	RSP_INT_(SET/CLR/STS)5_NS
CM33	CA55 Core1	MSG_INT_(SEL/CLR/STS)37_NS	RSP_INT_(SET/CLR/STS)11_NS
CM33	CA55 Core2	MSG_INT_(SEL/CLR/STS)38_NS	RSP_INT_(SET/CLR/STS)17_NS
CM33	CA55 Core3	MSG_INT_(SEL/CLR/STS)39_NS	RSP_INT_(SET/CLR/STS)23_NS
CA55 Core0	CA55 Core1	MSG_INT_(SEL/CLR/STS)0_S	RSP_INT_(SET/CLR/STS)0_S
CA55 Core0	CA55 Core2	MSG_INT_(SEL/CLR/STS)1_S	RSP_INT_(SET/CLR/STS)1_S
CA55 Core0	CA55 Core3	MSG_INT_(SEL/CLR/STS)2_S	RSP_INT_(SET/CLR/STS)2_S
CA55 Core0	CM33	MSG_INT_(SEL/CLR/STS)5_S	RSP_INT_(SET/CLR/STS)8_S
CA55 Core1	CA55 Core0	MSG_INT_(SEL/CLR/STS)6_S	RSP_INT_(SET/CLR/STS)12_S
CA55 Core1	CA55 Core2	MSG_INT_(SEL/CLR/STS)7_S	RSP_INT_(SET/CLR/STS)13_S
CA55 Core1	CA55 Core3	MSG_INT_(SEL/CLR/STS)8_S	RSP_INT_(SET/CLR/STS)14_S
CA55 Core1	CM33	MSG_INT_(SEL/CLR/STS)11_S	RSP_INT_(SET/CLR/STS)20_S
CA55 Core2	CA55 Core0	MSG_INT_(SEL/CLR/STS)12_S	RSP_INT_(SET/CLR/STS)24_S
CA55 Core2	CA55 Core1	MSG_INT_(SEL/CLR/STS)13_S	RSP_INT_(SET/CLR/STS)25_S
CA55 Core2	CA55 Core3	MSG_INT_(SEL/CLR/STS)14_S	RSP_INT_(SET/CLR/STS)26_S
CA55 Core2	CM33	MSG_INT_(SEL/CLR/STS)17_S	RSP_INT_(SET/CLR/STS)31_S
CA55 Core3	CA55 Core0	MSG_INT_(SEL/CLR/STS)18_S	RSP_INT_(SET/CLR/STS)32_S
CA55 Core3	CA55 Core1	MSG_INT_(SEL/CLR/STS)19_S	RSP_INT_(SET/CLR/STS)33_S
CA55 Core3	CA55 Core2	MSG_INT_(SEL/CLR/STS)20_S	RSP_INT_(SET/CLR/STS)36_S
CA55 Core3	CM33	MSG_INT_(SEL/CLR/STS)23_S	RSP_INT_(SET/CLR/STS)39_S
CM33	CA55 Core0	MSG_INT_(SEL/CLR/STS)36_S	RSP_INT_(SET/CLR/STS)5_S
CM33	CA55 Core1	MSG_INT_(SEL/CLR/STS)37_S	RSP_INT_(SET/CLR/STS)11_S
CM33	CA55 Core2	MSG_INT_(SEL/CLR/STS)38_S	RSP_INT_(SET/CLR/STS)17_S
CM33	CA55 Core3	MSG_INT_(SEL/CLR/STS)39_S	RSP_INT_(SET/CLR/STS)23_S

4.6.3.4.2 Software interrupt

Software interrupts are controlled by the set register, clear register, and status registers. An interrupt can be asserted by writing “1” to the set register. The interrupt is negated by writing “1” to the clear register, and the interrupt status can be checked by reading the status register. With these registers as one set, one channel is composed of a pair of message transmission processing / response transmission processing, and a total of 28 channels are mounted. **Table 4.6-34** lists the interrupt source/destination correspondence.

Table 4.6-34 Interrupt Destination/Source Reference

Source	Destination	Message Transmission Register
CA55 Core0	CM33	SW_INT_(SET/CLR/STS)2_NS
CA55 Core1	CM33	SW_INT_(SET/CLR/STS)5_NS
CA55 Core2	CM33	SW_INT_(SET/CLR/STS)8_NS
CA55 Core3	CM33	SW_INT_(SET/CLR/STS)11_NS
CM33	CA55 Core0	SW_INT_(SET/CLR/STS)22_NS
CM33	CA55 Core1	SW_INT_(SET/CLR/STS)23_NS
CM33	CA55 Core2	SW_INT_(SET/CLR/STS)24_NS
CM33	CA55 Core3	SW_INT_(SET/CLR/STS)25_NS

4.6.3.5 Operation Sequence

4.6.3.5.1 Message transmission sequence

The message transmission sequence from CM33 to CA55 Core0 is described below and shown in **Figure 4.6-25**.

(1) CM33: Check “0” for bit 0 in register: MSG_INT_STS36.
(Confirm cleared status and move to (2))

(2) CM33: Write message to shared RAM

(3) CM33: Write “1” to bit 0 in register: MSG_INT_SET36

(4) MHU channel 36: Assert interrupt: msg_ch36 for CA55

(5) CA55 Core0: Check interrupt source (Detect message transmission)

(6) CA55 Core0: Check “1” for bit 0 in register: MSG_INT_STS36

(7) CA55 Core0: Read message from shared RAM

(8) CA55 Core0: Write “1” to bit 0 in register: MSG_INT_CLR36

(9) MHU channel 36: Negate interrupt: msg_ch36 to CA55

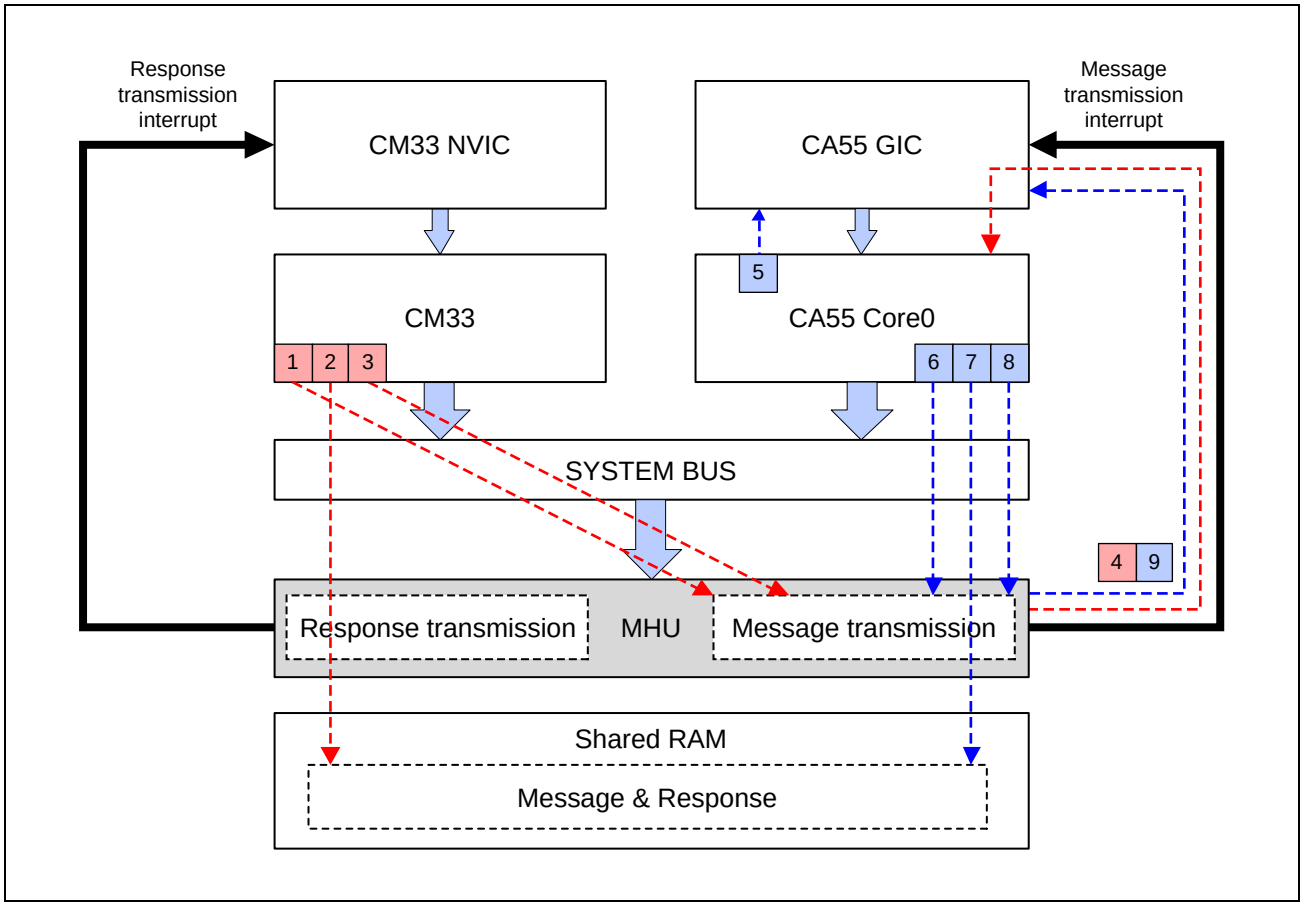


Figure 4.6-25 Message Transmission Sequence (CM33 -> CA55 Core0)

4.6.3.5.2 Response transmission sequence

The response transmission sequence from CA55 Core0 to CM33 is described below and shown in **Figure 4.6-26**.

(1) CA55 Core0: Check “0” for bit 0 in register: RSP_INT_STS5.
(Confirm cleared status and move to (2))

(2) CA55 Core0: Write response to shared RAM

(3) CA55 Core0: Write “1” to bit 0 in register: RSP_INT_SET5

(4) MHU channel 5: Assert interrupt: rsp_ch5 for CM33

(5) CM33: Check interrupt source (Detect response transmission)

(6) CM33: Check “1” for bit 0 in register: RSP_INT_STS5

(7) CM33: Read response from shared RAM

(8) CM33: Write “1” to bit 0 in register: RSP_INT_CLR5

(9) MHU channel 5: Negate interrupt:rsp_ch5 to CM33

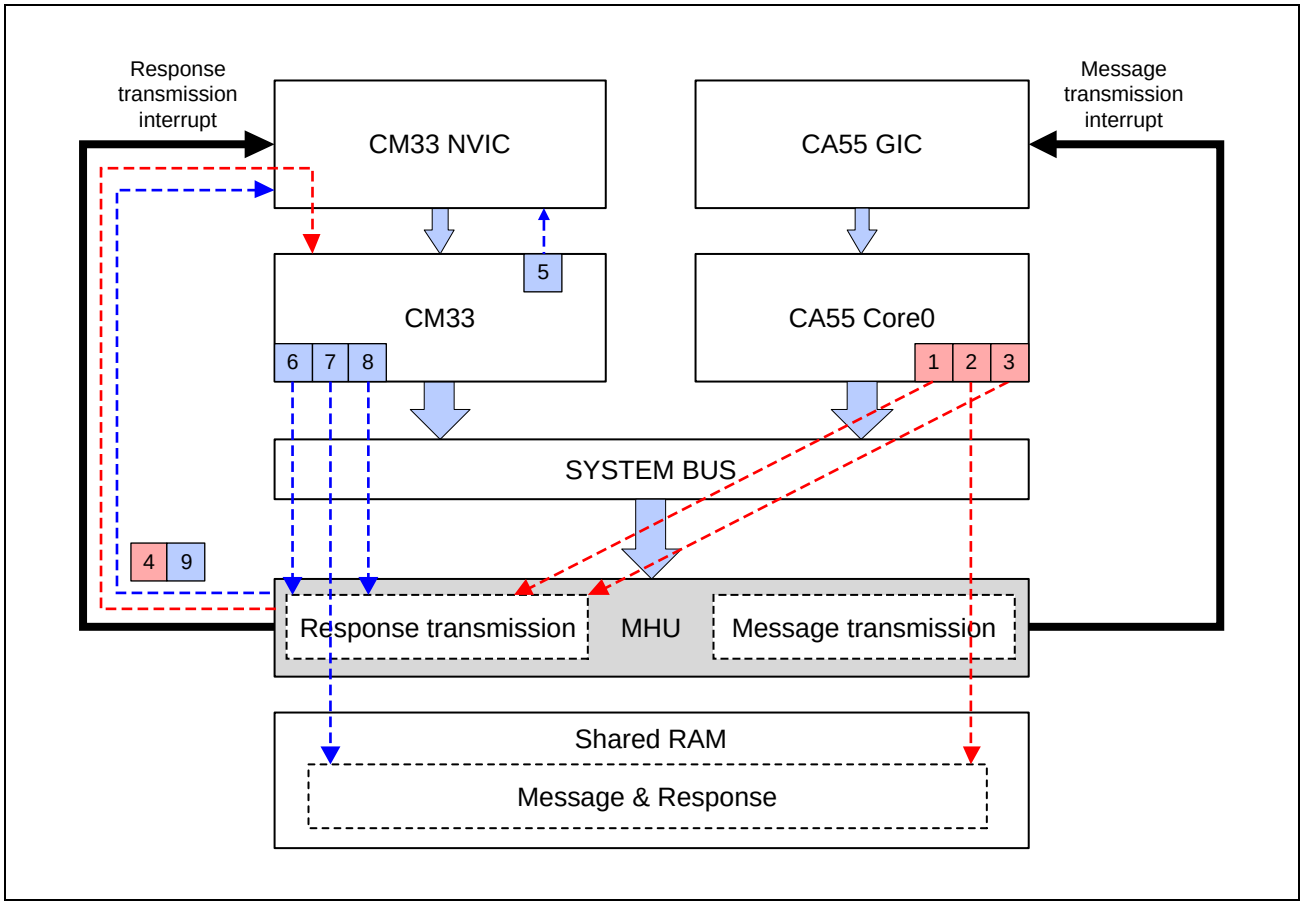


Figure 4.6-26 Response Transmission Sequence (CA55 Core0 -> CM33)

4.6.3.5.3 Software interrupt sequence

Show Software interrupt sequence from CA55 Core0 to CM33 in below.

(1)	CA55 Core0:	Check “0” for bit0 in register: SWI_INT_STS2_NS. (Confirm cleared status and move to (2))
(2)	CA55 Core0:	Write “1” to bit 0 in register: SWI_INT_SET2_NS
(3)	Software channel 2:	Assert interrupt: swint_ch2_ns for CM33
(4)	CM33:	Check interrupt reason (Detect software interrupt)
(5)	CM33:	Check “1” for bit 0 in register: SWI_INT_STS2_NS
(6)	CA55 Core0:	Write “1” to bit0 in register: SWI_INT_CLR2_NS
(7)	Software channel 2:	Negate interrupt: swint_ch2_ns to CM33

SECTION 4 SYSTEM

4.7 DMA Controller (DMAC)

This section describes the functions of the DMA controller (DMAC).

4.7.1 Functional Overview

The DMAC controller (DMAC) is made up of DMACA and DMACB (each having eight channels).This LSI has 5 DMAC modules. Each module has 16 channels, so there are 80 channels in this LSI.

It supports hardware activation, software activation, and link transfer with the use of descriptors.

4.7.1.1 Features

- The AOF register setting in SYS allows access to the address space over 4 Gbytes.

- DMA transaction setting modes

  Register mode and link mode are supported.

  - Register mode
    In this mode, DMA transfer proceeds with the use of settings from the CPU. Up to two register sets (Next0 and Next1 register sets) can be set and alternating consecutive transfer according to the respective settings can proceed.
  - Link mode
    In this mode, the settings (descriptors) are allocated to the external memory and captured by the DMAC, after which DMA transfer proceeds in accord with the values. By preparing multiple settings which specify the addresses of the descriptors to control next transfers, sequential execution of transfers is possible.
    Additionally, suspension and resumption of the next DMA transfer can be specified by the header of the descriptor.

- Triggering modes

  The following two types are supported to start DMA transfer.

  - Software activation
    DMA transfer is started by writing to an internal register.
  - Hardware activation
    DMA transfer is started according to the state of the DMAREQ input. The detection mode is selectable from those listed below. Masking is also possible.

    — Rising-edge detection
    — Falling-edge detection
    — Change-point detection
    — High-level detection
    — Low-level detection

- Interrupt

  When a transfer is completed, the transfer completion interrupt DMAEND[7:0] pins are asserted (can also be masked).
  In the event of a bus error, the error interrupt DMAERR pin is asserted.

- Method of transfer

  Transfer sizes for the source and the destination are individually selectable as values from among 1 to 128 bytes.
  For transfer addresses, increment mode and fixed mode are supported.

- Buffer flushing

  Data already captured in the buffer can be forcibly flushed.

- Suspension

  DMA transfer in progress can be suspended.

- Interval function

  The interval between DMA transfers is specifiable to adjust the bus occupancy ratio.

4.7.1.2 DMA Transfer Specifications

- Maximum number of bytes for DMA transfer

  Byte counting is specifiable per channel. The maximum number is $(2^{32} - 1)$ bytes = (4G − 1) bytes.

4.7.1.3 Restrictions

If transfer proceeds with the use of this unit in such a way that the data source and destination areas are the same or parts of these areas overlap, data consistency cannot be guaranteed. Accordingly, make settings such that the data source and destination address areas do not overlap.

4.7.2 Connection Configuration

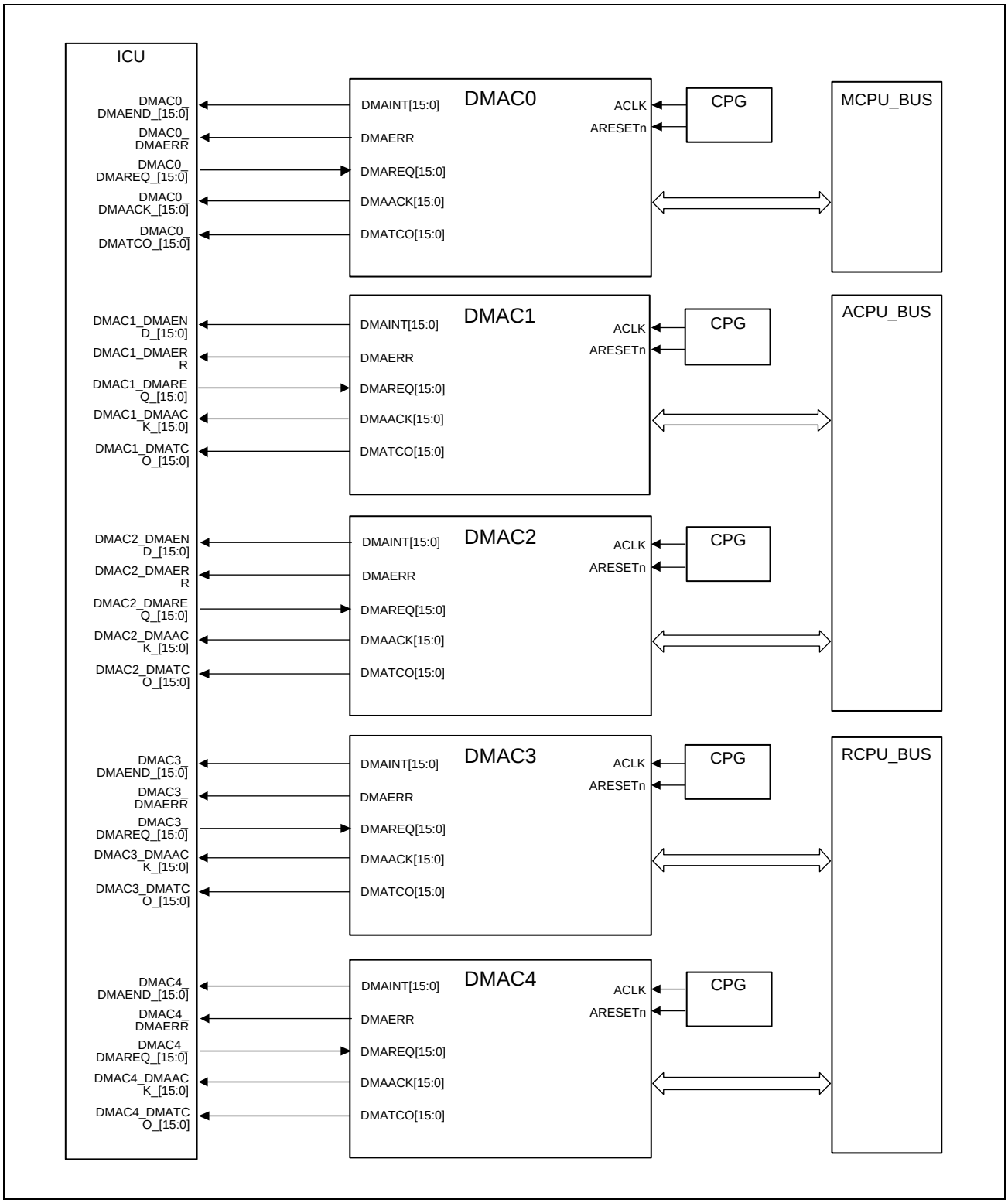


Figure 4.7-1 Connection Configuration

4.7.3 Pin Functions

4.7.3.1 List of External Pins

Table 4.7-1 List of External Pins

Pin Name	Input/Output	Functions
DREQ[4:0]	Input	DMA transfer request input from an external device to DMAC
DACK[4:0]	Output	Output of DMAC transfer request acceptance from DMAC to an external device
TEND[4:0]	Output	Output of transfer completion from DMAC to an external device

Note: The DMAC external pins are multiplexed pins. For the settings of each pin, see **1.2 Pin** and **4.2 Pin Function Controller (PFC)**.
When these pins are used, set the resource to PFC (external pin) by the ICU registers (ICU_DMkSELy, ICU_DMTENDSELk, and ICU_DMACKSELk).

4.7.3.2 List of Internal Pins

Table 4.7-2 lists the internal pins of the DMAC.

Table 4.7-2 List of Internal Pins

Pin Name	Input/Output	Functions
DMA interfaces		
DMAREQ[15:0]	Input	DMA transfer request input *Note*: If DMAREQ is generated with the clock which is not synchronized with ACLK, use level-detection mode.
DMAACK[15:0]	Output	DMA acknowledge output *Note*: If the DMAACK signal is to be fetched with the clock which is not synchronized with ACLK, use level-output mode.
DMATCO[15:0]	Output	DMA terminal count output (pulse)
Interrupt interfaces		
DMAINT[15:0]	Output	DMA transaction completed interrupt
DMAERR	Output	Error response (ERROR) interrupt

4.7.3.2.1 DMA Interfaces

Each DMAC can be set to select one request from among eight transfer requests by software.

The DMAREQ, DMAACK, and DMATCO signals are referred to as the DMA signals and are on pins for use in hardware activation.

Eight DMAREQ, DMAACK, and DMATCO signals are assigned to each DMAC unit. The bit positions to be used are switchable by using the SEL field of the DMACm_CHCFG_n register.

DMAREQ (Input)

This is a request input signal for DMA transfer.

Edge detection (rising-edge detection, falling-edge detection, and change-point detection) and level detection (high-level detection and low-level detection) are supported (masking requests is also possible).

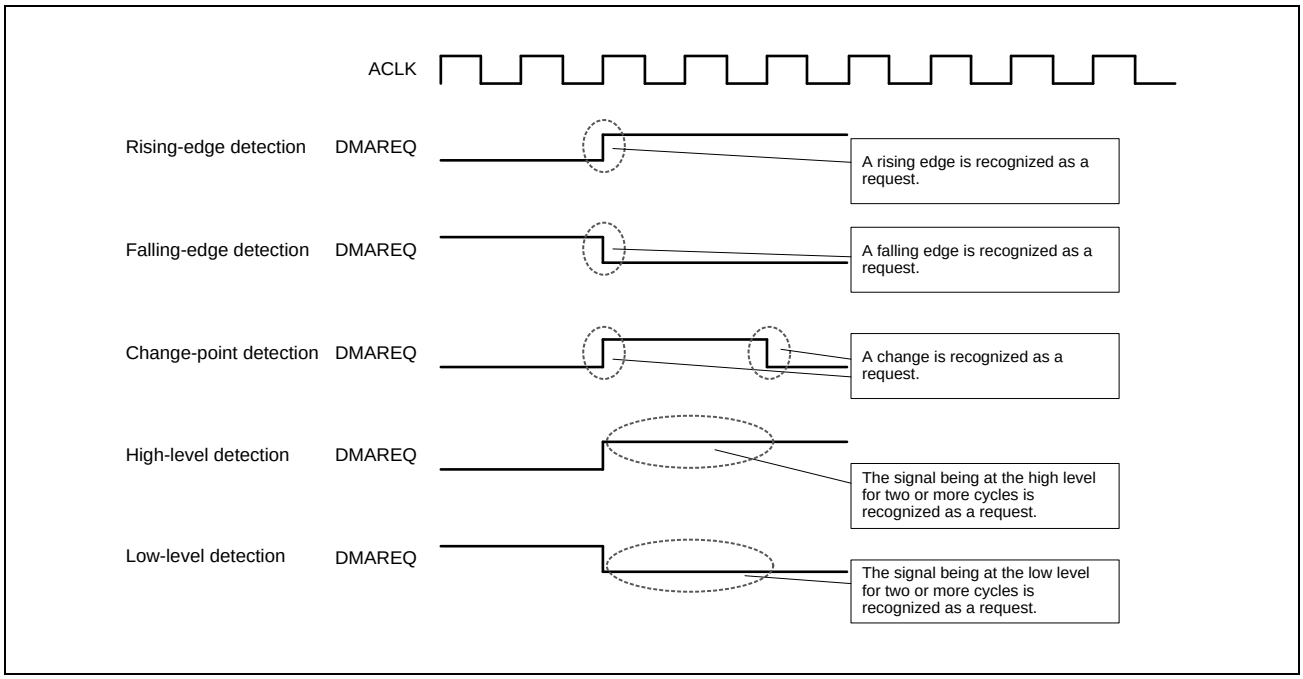


Figure 4.7-2 DMAREQ Detection Mode

DMAACK (Output)

This is an acknowledge output signal for the DMAREQ signal.

This signal is activated in response to the start of transfer corresponding to the DMAREQ signal.

It supports pulse mode, level mode, and bus cycle mode (masking the output is also possible).

If the DMAACK signal is in level mode, it remains at the high level until the DMAREQ signal is deactivated.

A next DMA request is not accepted while the DMAACK signal is at the high level. Use this signal for handshaking with the DMAREQ signal.

The figure below shows the output modes of the respective DMAACK signals and the example timing charts.

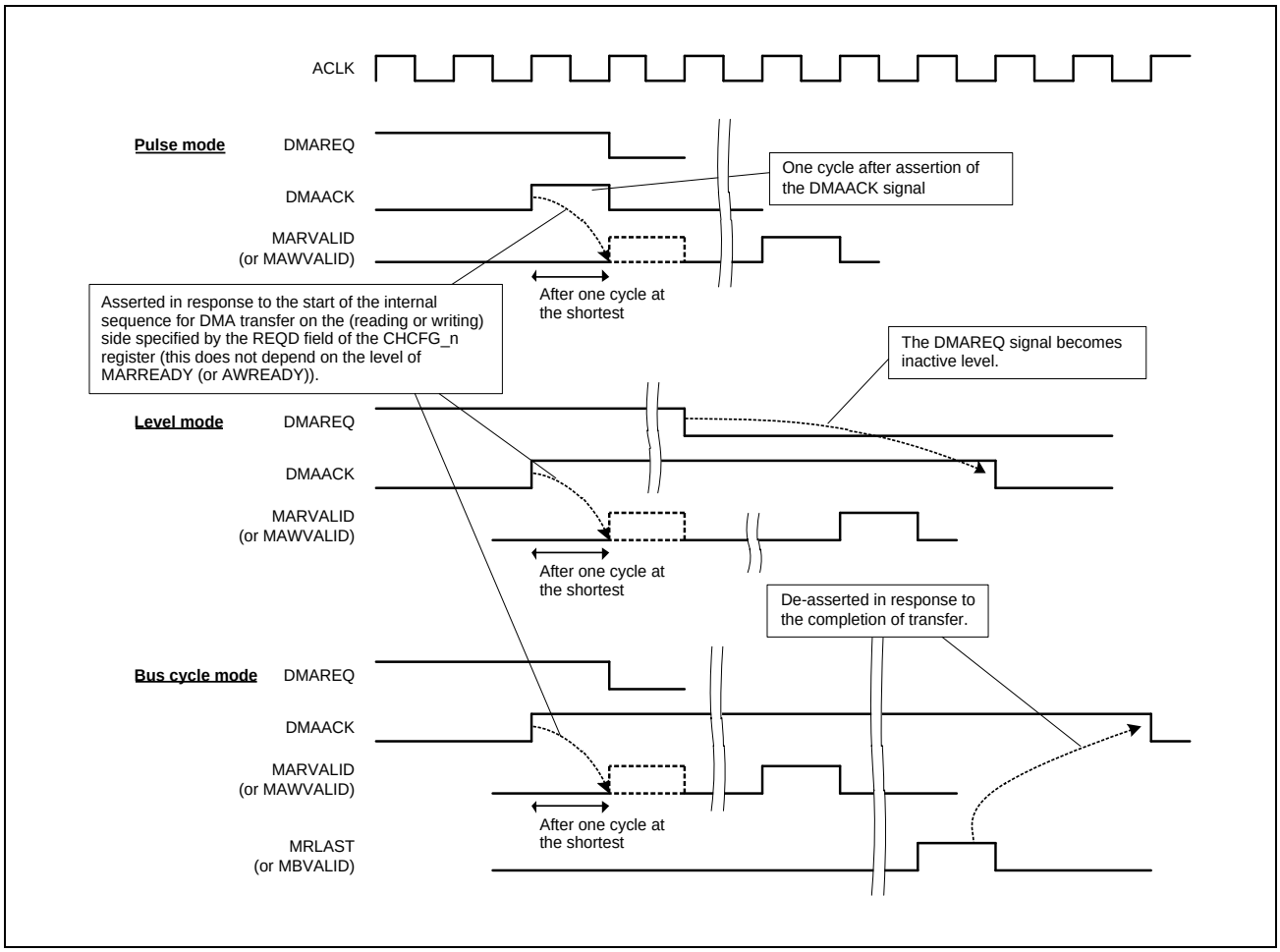


Figure 4.7-3 DMAACK Signal Output

DMATCO (Output)

This is an output signal that indicates the end of a DMA transaction.

4.7.3.2.2 Interrupt Interfaces

DMAINT (Output)

This is a DMA transaction completed interrupt signal. This signal is asserted on the completion of the DMA transaction. It is also asserted when an invalid descriptor is read in link mode.

The DMAINT pins correspond to the DMAEND pins of the internal DMAC. **Table 4.7-3** shows the assignment of the channels of the DMAC units to the DMAINT pins.

Table 4.7-3 DMAINT Pin Assignment

DMAINT pin	[15]	[14]	[13]	[12]	[11]	[10]	[9]	[8]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]
DMAC unit	DMACB								DMACA							
Channel	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0

DMAERR (Output)

This is an error response (ERROR) interrupt signal. This signal is asserted when an error response is returned to the transfer request issued by the master interface. This pin is one bit and is common to all channels.

The format of this interrupt output is switchable between pulse output and level output through the setting of the register. After a reset, pulse output is selected.

4.7.4 Address Space

Table 4.7-4 lists the allocation of the registers for DMACm (m = 0 to 4).

Table 4.7-4 Address Space

Base Address Name	Group	Channel n	Base Address
DMAC0_base	Registers for DMAC0 (m = 0)	n = 0 to 7	0_1140_0000h (5140_0000h*1, 4140_0000h*2)
		n = 8 to 15	0_1140_0400h (5140_0400h*1, 4140_0400h*2)
DMAC1_base	Registers for DMAC1 (m = 1)	n = 0 to 7	0_1483_0000h (5483_0000h*1, 4483_0000h*2)
		n = 8 to 15	0_1483_0400h (5483_0400h*1, 4483_0400h*2)
DMAC2_base	Registers for DMAC2 (m = 2)	n = 0 to 7	0_1484_0000h (5484_0000h*1, 4484_0000h*2)
		n = 8 to 15	0_1484_0400h (5484_0400h*1, 4484_0400h*2)
DMAC3_base	Registers for DMAC3 (m = 3)	n = 0 to 7	0_1200_0000h (5200_0000h*1, 4200_0000h*2)
		n = 8 to 15	0_1200_0400h (5200_0400h*1, 4200_0400h*2)
DMAC4_base	Registers for DMAC4 (m = 4)	n = 0 to 7	0_1201_0000h (5201_0000h*1, 4201_0000h*2)
		n = 8 to 15	0_1201_0400h (5201_0400h*1, 4201_0400h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

4.7.5 Register Descriptions

The following lists the registers of one DMAC unit.

4.7.5.1 List of Registers

The following table lists the registers of the DMAC unit.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Next0 Source Address Register n	DMACm_N0SA_n	0000_0000h	0000h + k x 0040h	32
Next0 Destination Address Register n	DMACm_N0DA_n	0000_0000h	0004h + k x 0040h	32
Next0 Transaction Byte Register n	DMACm_N0TB_n	0000_0000h	0008h + k x 0040h	32
Next1 Source Address Register n	DMACm_N1SA_n	0000_0000h	000Ch + k x 0040h	32
Next1 Destination Address Register n	DMACm_N1DA_n	0000_0000h	0010h + k x 0040h	32
Next1 Transaction Byte Register n	DMACm_N1TB_n	0000_0000h	0014h + k x 0040h	32
Current Source Address Register n	DMACm_CRSA_n	0000_0000h	0018h + k x 0040h	32
Current Destination Address Register n	DMACm_CRDA_n	0000_0000h	001Ch + k x 0040h	32
Current Transaction Byte Register n	DMACm_CRTB_n	0000_0000h	0020h + k x 0040h	32
Channel Status Register n	DMACm_CHSTAT_n	0000_0000h	0024h + k x 0040h	32
Channel Control Register n	DMACm_CHCTRL_n	0000_0000h	0028h + k x 0040h	32
Channel Configuration Register n	DMACm_CHCFG_n	0000_0000h	002Ch + k x 0040h	32
Channel Interval Register n	DMACm_CHITVL_n	0000_0000h	0030h + k x 0040h	32
Channel Extension Register n	DMACm_CHEXT_n	0000_0000h	0034h + k x 0040h	32
Next Link Address Register n	DMACm_NXLA_n	0000_0000h	0038h + k x 0040h	32
Current Link Address Register n	DMACm_CRLA_n	0000_0000h	003Ch + k x 0040h	32
Reserve	-	-	0200h to 02FFh	-
DMA Control Register	DMACm_DCTRL	0000_0000h	0300h	32
Reserve	-	-	0304h to 030Fh	-
DMA Status EN Register	DMACm_DST_EN	0000_0000h	0310h	32
DMA Status ER Register	DMACm_DST_ER	0000_0000h	0314h	32
DMA Status END Register	DMACm_DST_END	0000_0000h	0318h	32
DMA Status TC Register	DMACm_DST_TC	0000_0000h	031Ch	32
DMA Status SUS Register	DMACm_DST_SUS	0000_0000h	0320h	32

Note: k = 0 to 7

CAUTIONS

1. If the reserved area is accessed, an OK response is returned. The initial value and the read/write attribute of this area may be changed by the functional extension. Do not write the code which expects that the value read from this area is 0b by software. When writing to it, write 0b.
2. If the undefined area is accessed, an error response is returned. The control registers will not be changed.
3. Do not modify the registers by software while DMA transfer is in progress (EN = 1b), with the exception of the following registers.
   - Register set on the side that has not transferred data in register mode
   - DMACm_CHCTRL_n

4.7.5.2 Register Descriptions

The functional description of each register is given below.

The prefix (DMACm_) of the register names is omitted in the register descriptions and the field descriptions in this section.

The figure below shows the register configuration.

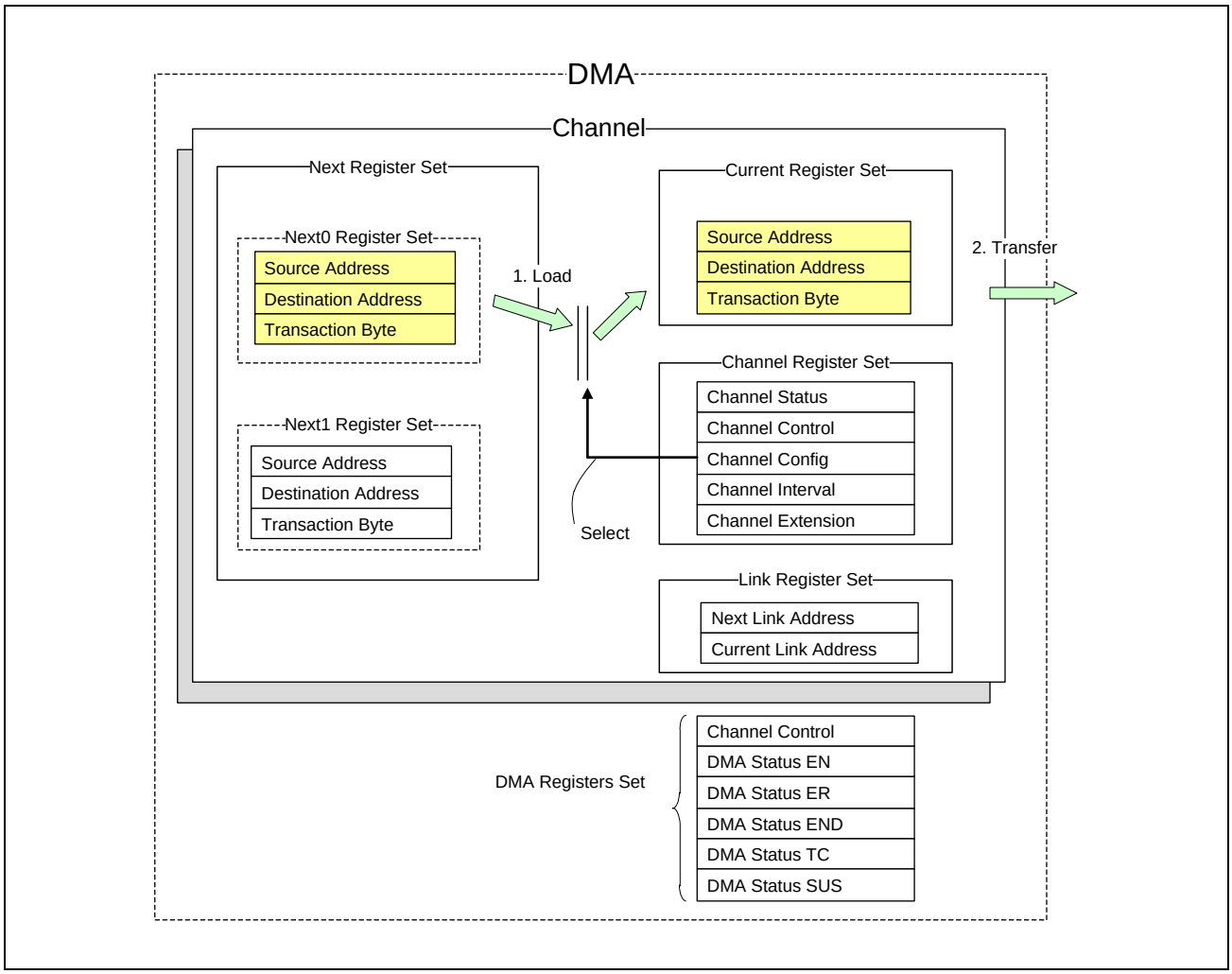


Figure 4.7-4 Register Configuration

■ Next Register Set

This register set sets the transfer source address, transfer destination address, and number of bytes to be transferred in the next DMA transaction to be executed.

It is composed of the Next0 and Next1 register sets.
In register mode, it is set by software. In link mode, the descriptor read data are automatically set in the Next0 register set.

The values of these register sets are loaded to the current register set and used for DMA transfer.

■ Current Register Set

This register set indicates the transfer source address, transfer destination address, and number of bytes being transferred in the DMA transaction that is currently in progress.

It is loaded from the Next0/1 register set (in register mode) or the descriptor read data (in link mode). It cannot be directly written by the user.

It is automatically updated every time the DMA transaction is executed.

■ Channel Register Set

This register set is used to make settings for DMA transfer.

It indicates the channel state, controls the channels, sets up DMA transactions, sets the interval between DMA transactions, and so on.

■ Link Register Set

This register set is composed of a register for setting the next descriptor address to be loaded (Next Link Address Register) and a register that indicates the descriptor address currently being executed (Current Link Address Register).

The Current Link Address Register is automatically updated by reading the descriptor and cannot be directly written by the user.

■ DMA Register Set

This register set is composed of a register that controls the entire DMAC and a register that indicates the state of the given channel. It is used to control the priority of channels, check the state of EN, ER, END, TCO, and SUS of the given channel, and so on.

4.7.5.2.1 Next0 Source Address Register n (DMACm_N0SA_n) (m = 0 to 4, n = 0 to 15)

This register sets the DMA transfer source address (32 bits) of the DMAC channel n.

Access Size : 32 bits
Address : <DMACm_base> + 0000h + k x 0040h (k = 0 to 7)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SA[31:0]	0h	RW	Source Address Sets the start address of the DMA transfer source.

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.2 Next0 Destination Address Register n (DMACm_N0DA_n) (m = 0 to 4, n = 0 to 15)

This register sets the DMA transfer destination address (32 bits) of the DMAC channel n.

Access Size : 32 bits
Address : <DMACm_base> + 0004h + k x 0040h (k = 0 to 7)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DA[31:0]	0h	RW	Destination Address Sets the start address of the DMA transfer destination.

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.3 Next0 Transaction Byte Register n (DMACm_N0TB_n) (m = 0 to 4, n = 0 to 15)

This register sets the total number of bytes transferred to the DMA channel n.

Access Size : 32 bits
Address : <DMACm_base> + 0008h + k x 0040h (k = 0 to 7)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TB[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TB[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TB[31:0]	0h	RW	Transaction Byte Sets the total number of transfer bytes. Note: Sets a non-zero value when starting a DMA transaction.

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.4 Next1 Source Address Register n (DMACm_N1SA_n) (m = 0 to 4, n = 0 to 15)

This register sets the DMA transfer source address (32 bits) of the DMAC channel n.

Access Size : 32 bits
Address : <DMACm_base> + 000Ch + k x 0040h (k = 0 to 7)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SA[31:0]	0h	RW	Source Address Sets the start address of the DMA transfer source.

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.5 Next1 Destination Address Register n (DMACm_N1DA_n) (m = 0 to 4, n = 0 to 15)

This register sets the DMA transfer destination address (32 bits) of the DMA channel n.

Access Size : 32 bits
Address : <DMACm_base> + 0010h + k x 0040h (k = 0 to 7)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DA[31:0]	0h	RW	Destination Address Sets the start address of the DMA transfer destination.

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.6 Next1 Transaction Byte Register n (DMACm_N1TB_n) (m = 0 to 4, n = 0 to 15)

This register sets the total number of bytes transferred to the DMA channel n.

Access Size : 32 bits
Address : <DMACm_base> + 0014h + k x 0040h (k = 0 to 7)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TB[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TB[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TB[31:0]	0h	RW	Transaction Byte Sets the total number of transfer bytes. Note: Set a non-zero value when starting a DMA transaction.

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.7 Current Source Address Register n (DMACm_CRSA_n) (m = 0 to 4, n = 0 to 15)

This register displays the DMA transfer source address of DMAC channel n.

Note: Indicates the Next 0/1 register value in register mode and the descriptor value in link mode. Cannot be written by software.

Access Size : 32 bits
Address : <DMACm_base> + 0018h + k x 0040h (k = 0 to 7)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRSA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRSA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRSA[31:0]	0h	R	Current Source Address Register Indicates the read address of the next DMA transaction. During a DMA transaction, increment is performed automatically. (Fixed when CHCFG_n.SAD = 1b.) The initial value is loaded from the following register. Register mode: Load source address from Next 0/1. Link mode: Load source address from the descriptor. Increment is performed at the start of read transfer. Read this register after the DMAC stops (CHSTAT_n.EN = 0b). (Treat the value during DMA operation as a reference value.)

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.8 Current Destination Address Register n (DMACm_CRDA_n) (m = 0 to 4, n = 0 to 15)

This register displays the DMA transfer destination address of DMAC channel n.

Note: Indicates the Next 0/1 register value in register mode and the descriptor value in link mode. Cannot be written by software.

Access Size : 32 bits

Address : <DMACm_base> + 001Ch + k x 0040h (k = 0 to 7)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRDA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRDA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRDA[31:0]	0h	R	Current Destination Address Register Indicates the write address of the next DMA transaction. During a DMA transaction, increment performed automatically.(Fixed when CHCFG_n.DAD = 1b.) The initial value is loaded from the following register. Register mode: Load forwarding destination address from Next 0/1. Link mode: Load forwarding destination address from the descriptor. Increment is performed at the start of write transfer. Read this register after the DMAC stops (CHSTAT_n.EN = 0b). (Teat the value during DMA operation as a reference value.)

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.9 Current Transaction Byte Register n (DMACm_CRTB_n) (m = 0 to 4, n = 0 to 15)

This register displays the total number of bytes transferred for DMAC channel n.

Note: Indicates the Next 0/1 register value in register mode and the descriptor value in link mode. Cannot be written by software.

Access Size : 32 bits
Address : <DMACm_base> + 0020h + k x 0040h (k = 0 to 7)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRTB[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRTB[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRTB[31:0]	0h	R	Current Transaction Byte Register Indicates the number of bytes remaining in the current DMA transaction. During a DMA transaction, it is automatically decremented. The initial value is loaded from the following register. Register mode: Load transfer byte count from Next 0/1 Link mode: Loads transfer byte from the descriptor. Decrements when write transfer is completed. Read this register after the DMAC stops (CHSTAT_n.EN = 0b). (Treat the value during DMA operation as a reference value.)

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.10 Channel Status Register n (DMACm_CHSTAT_n) (m = 0 to 4, n = 0 to 15)

This register indicates the status of DMAC channel n.

Access Size : 32 bits

Address : <DMACm_base> + 0024h + k x 0040h (k = 0 to 7)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INTMSK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	MODE	DER	DW	DL	SR	TC	END	ER	SUS	TACT	RQST	EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved These bits are read as 0b.
16	INTMSK	0h	R	Displays the temporary mask status of DMAENDn interrupt pin output. 1b: Temporary mask state 0b: Temporary mask release state Setting condition: When SETINTMSK is set to 1b. Clearing condition: When CLRINTMSK or SWRST is set to 1b.
15 to 12	-	All 0	R	Reserved These bits are read as 0b.
11	MODE	0h	R	DMA Mode Indicates the DMA mode. Displays the setting value of the DMS bit in the CHCFG_n register. 0b: Register mode 1b: Link mode
10	DER	0h	R	Descriptor Error Indicates that the read register was invalid (LV = 0b). (It does not depend on the DIM level of the CHCFG_n register.) 0b: Descriptor Error has not occurred 1b: Descriptor Error has occurred Setting condition: Link mode descriptor load LV is 0b. Clearing condition: When SWRST is set to 1b.
9	DW	0h	R	Descriptor Write-back Indicates the descriptor write-back status. If a bus error is received during descriptor write-back, 1b is retained. 0b: Header in link mode is not written back. 1b: (ER = 0b) Header in link mode is being written back. (ER = 1b) A bus error occurred while writing back the header in link mode. Setting condition: At the start of write-back of the header in link mode. Clearing condition: Header write-back in link mode ends with an OK response, or SWRST (CHCTRLn) is set to 1b.

Bit	Bit Name	Initial Value	R/W	Description
8	DL	0h	R	Descriptor Load Indicates that the descriptor is loaded. If a bus error is received during descriptor loading, 1b is held. 0b: No descriptor is loaded. 1b: (ER = 0b) Descriptor in link mode is being loaded. (ER = 1b) Bus error occurred during descriptor loading in link mode. Setting condition: At the start of descriptor loading in link mode. Clearing condition: Descriptor loading in link mode ends with an OK response, or SWRST (CHCTRLn) is set to 1b.
7	SR	0h	R	Selected Register Set Indicates the selected register set in register mode. 0b: Next0 Register Set 1b: Next1 Register Set Setting condition: When RSEL is set to 1b. Clearing condition: When RSEL is set to 0b.
6	TC	0h	R	Terminal Count This status bit indicates that the DMAC transaction is completed. It is set only when TCM = 0b in the CHCFG_n register. 0b: DMA transfer is incomplete 1b: DMA transfer is complete Setting conditions: – When transfer for the total number of transfer bytes set in the CRTB register is completed in register mode. – When transfer of the total number of transfer bytes set in the CRTB register is completed with WBD = 1b in the header of the descriptor in link mode. – Descriptor write-back ends with WBD = 0b in the header of the descriptor in link mode. Clearing conditions: – CLRTC (CHCTRL_n) is set to 1b. – SWRST (CHCTRL_n) is set to 1b.
5	END	0h	R	DMAEND Interrupt This bit indicates that a DMA transaction has been completed and a DMAEND interrupt has occurred. 0b: DMA transfer is incomplete 1b: DMA transfer is complete Setting conditions: – When the TC bit is set and DEM = 0b in the CHCFG_n register. – When LV = 0b in the header and DIM = 0b when a descriptor is read in link mode. Clearing conditions: – CLREND (CHCTRL_n) is set to 1b. – SWRST (CHCTRL_n) is set to 1b.
4	ER	0h	R	Error Bit Indicates that a DMAERR interrupt has been generated in response to an ERROR response during DMA transfer. 0b: ERROR response not received 1b: ERROR response received Setting condition: When an error response is received in the bus cycle. Clearing condition: SWRST (CHCTRL_n) is set to 1b.
3	SUS	0h	R	Suspend This bit indicates that the channel is paused. 0b: Channel_n is not paused 1b: Channel_n is paused Setting condition: SETSUS is set to 1b during DMA transfer of Channel_n, and the internal status becomes SUSPEND. Clearing conditions: CLRSUS is set to 1b. CLREN is set to 1b.

Bit	Bit Name	Initial Value	R/W	Description
2	TACT	0h	R	Transaction Active This bit indicates that the DMAC is operating. This bit is used to confirm that the channel is completely stopped. 0b: Channel_n DMAC is stopped 1b: Channel_n DMAC is operating Setting condition: At the start of DMA transaction by Channel_n. Clearing condition: When DMA transaction is completed.
1	RQST	0h	R	Request This bit indicates that a transfer request is accepted. 0b: DMA transfer request is not received. 1b: DMA transfer request is received. Setting conditions: – The STG-bit is set to 1b. – When a transfer request is accepted from the DMAREQ pin set by the CHCFG_n register. Clearing conditions: – SWRST (CHCTRL_n) is set to 1b. – CLRRQ (CHCTRL_n) is set to 1b. – When executing transfer on the side specified by REQD in single transfer (TM = 0b) mode. – When all DMA transactions are completed in register mode (Transaction completed with REN = 0b). – When DMA transfer of the last descriptor (LE = 1b) is completed in link mode. – When transfer is stopped in descriptor mode (LV = 0b) in link mode. – When the DMA transaction is terminated with DEM = 0b in link mode. – When the master interface receives a bus error.
0	EN	0h	R	Enable Displays whether DMAC channel n is enabled or stopped. 0b: Operation stopped state 1b: Operation enabled state Setting condition: SETEN (CHCTRL_n) is set to 1b. Clearing conditions: – SWRST (CHCTRL_n) is set to 1b. – CLREN(CHCTRL_n) is set to 1b. – When an error response is received during transfer. – When all DMA transactions are completed in register mode (transaction complete when REN = 0b). – When DMA transfer (writeback if WBD = 0b) of the last descriptor (LE = 1b) is completed in link mode. – When transfer is stopped (LV = 0b) when a descriptor is read in link mode

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

CAUTION

Treat transfers with the ER bit set as if the series of transfers is invalid. To interrupt a DMA transaction, mask and clear the transfer request or clear the enable. If a transfer request by DMA transfer request pin (DMAREQm) input and a transfer request by software (Set the STG bit) are used together for the same channel, the activated activation factor cannot be specified. Use only one of the transfer requests in the system. When performing a transfer request by software, perform the next STG bit operation after completion (Check with Current Register etc.) of the previously requested DMA transfer operation.

4.7.5.2.11 Channel Control Register n (DMACm_CHCTRL_n) (m = 0 to 4, n = 0 to 15)

This register controls the DMA transfer operation of DMA channel n.

Access Size : 32 bits

Address : <DMACm_base> + 0028h + k x 0040h (k = 0 to 7)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CLRINTMSK	SETINTMSK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	CLRSUS	SETSUS	-	CLRTC	CLRENDD	CLRRQ	SWRST	STG	CLREN	SETEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
17	CLRINTMSK	0h	RW	Setting this bit cancels the state of masking the DMAENDn pin output. Also, the INTMSK bit of the CHSTAT_n register is 0b. If the mask is released while LVINT = 1b in the DCTRL register and END = 1b in the CHSTAT_n register, the DMAENDn pin output becomes active. (It is not active when LVINT = 0b.) When reading, 0b can be read. 1b: Cancels the mask set with SETINTMSK. 0b: Does not affect the operation.
16	SETINTMSK	0h	RW	Setting this bit temporarily masks the DMAENDn pin output. Also, the INTMSK bit of the CHSTATm register is set to 1b. When reading, 0b can be read. 1b: Masks DMAENDn. 0b: Does not affect the operation.
15 to 10	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
9	CLRSUS	0h	RW	Clear suspend Releases the pause state. When this bit is set to 1b when SUS of the CHSTAT_n register is 1b, the suspended state can be released. When this bit is read, 0b can be read. 1b: Cancel suspension of DMA transfer in progress. 0b: Does not affect operation.
8	SETSUS	0h	RW	Set Suspend Suspends the ongoing DMA transfer. When this bit is set to 1b when EN of the CHSTAT_n register is 1b, DMA transfer in progress can be paused. When this bit is read, 0b can be read. 1b: Suspend DMA transfer in progress. 0b: Does not affect operation.
7	-	0h	R	Reserved These bits are read as 0b. The write value should always be 0b.
6	CLRTC	0h	RW	Clear TC Bit By setting this bit, the TC bit of the CHSTAT_n register can be cleared. When this bit is read, 0b can be read. 1b: Clear TC bit. 0b: Does not affect operation.
5	CLREND	0h	RW	Clear End Bit By setting this bit, the END bit of the CHSTAT_n register can be cleared. It also clears the DMAEND interrupt pin to low level. When this bit is read, 0b can be read. 1b: Clear END bit. 0b: Does not affect operation.
4	CLRRQ	0h	RW	Clear Request Bit By setting this bit, the RQST bit of the CHSTAT_n register can be cleared. When this bit is read, 0b can be read. 1b: Clear RQST bit. 0b: Does not affect operation.

Bit	Bit Name	Initial Value	R/W	Description
3	SWRST	0h	RW	Software Reset The status register can be cleared by setting this bit. Set this bit when the EN bit is 0b and the TACT bit is 0b. When this bit is read, 0b can be read. 1b: Channel status register reset. 0b: Does not affect operation.
2	STG	0h	RW	Software Trigger Setting this bit sets an internal transfer request (software activation). If it is set at the same time as the SWRST bit, clearing by the SWRST bit has priority. When this bit is read, 0b can be read. 1b: Set transfer request by software (set RQST bit). 0b: Does not affect operation.
1	CLREN	0h	RW	Clear Enable The EN bit can be cleared by setting this bit. When this bit is read, 0b can be read. 1b: DMA transfer stopped (EN bit cleared). 0b: Does not affect operation.
0	SETEN	0h	RW	Set Enable Sets DMA transfer permission for DMAC channel n. If it is set at the same time as the SWRST bit, clearing by the SWRST bit has priority and transfer does not start. When this bit is read, 0b can be read. 1b: DMA transfer enabled (EN bit set). 0b: Does not affect operation.

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.12 Channel Configuration Register n (DMACm_CHCFG_n) (m = 0 to 4, n = 0 to 15)

This register controls the DMA transfer operation of DMA channel n.

Access Size : 32 bits

Address : <DMACm_base> + 002Ch + k x 0040h (k = 0 to 7)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMS	REN	RSW	RSEL	SBE	-	TCM	DEM	-	TM	DAD	SAD	DDS[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	R	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDS[3:0]				-	AM[2:0]			-	LVL	HIEN	LOEN	REQD	SEL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	DMS	0h	RW	DMA Mode Select Sets the DMA mode. 0b: Register mode (initial value) 1b: Link mode
30	REN	0h	RW	Register Set Enable After the completion of the DMA transaction, DMA transfer is performed using the Next register set selected by RSEL. This bit is valid only in register mode. 0b: Does not perform DMA transfer accordingly. 1b: Performs DMA transfer accordingly. Setting condition: 1b is written to this bit. Clearing condition: 0b is written to this bit, or DMA transaction is completed with REN = 1b.
29	RSW	0h	RW	Register Select Switch After the DMA transaction ends, RSEL is automatically inverted. This bit is valid only in register mode. 0b: Do not reverse RSEL after completion of DMA transaction. 1b: Reverse RSEL after completion of DMA transaction.
28	RSEL	0h	RW	Register Set Select Select the next register set to be executed next. This bit is valid only in register mode. When RSW = 1b, it is automatically reversed when the DMA transaction is completed. 0b: Execute Next0 Register Set. 1b: Execute Next1 Register Set. Transition condition: When RSW = 1b and DMA transaction completes.
27	SBE	0h	RW	Sweep Buffer Enable If the enable is cleared to 0b during a DMA transaction, select whether to sweep (by writing) the data that has already been read and captured in the buffer and stop it, or stop without sweeping. The sweep mode can be used only when REQD = 0b. 0b: Cancels transfer without sweeping buffer. 1b: Transfer is canceled by sweeping out the buffer.
26	-	0h	R	Reserved These bits are read as 0b. The write value should always be 0b.
25	TCM	0h	RW	DMATCO Mask Masks DMATCOn interrupt pin output. If this bit is 1b at the DMATCO interrupt output timing, DMATCOn is not asserted. At this time, TCM is automatically cleared to 0b. This bit should be set when DMA transaction is controlled by software. 0b: Do not mask. 1b: Mask. Clearing condition: When TCM = 1b and DMA transaction completes

Bit	Bit Name	Initial Value	R/W	Description
24	DEM	0h	RW	DMAEND Mask Masks DMAEND [s] (s: pin selected by SEL) interrupt pin output during register mode transfer. If this bit is 1b at the DMAEND interrupt output timing, DMAEND [s] is not asserted. At this time, DEM is automatically cleared to 0b. 0b: Do not mask. 1b: Mask. Clearing condition: When DEM = 1b and DMA transaction completes
23	-	0h	R	Reserved These bits are read as 0b. The write value should always be 0b.
22	TM	0h	RW	Transfer Mode Sets the DMA transfer mode. 0b: Single transfer mode 1b: Block transfer mode
21	DAD	0h	RW	Sets the count direction of the transfer destination address of DMAC channel n. 0b: Increment 1b: Fixed If the destination address is beat unaligned, specify DAD = 0b (increment). If the destination address is beat aligned, both DAD = 0b and 1b can be set.
20	SAD	0h	RW	Sets the count direction of the transfer source address of DMAC channel n. 0b: Increment 1b: Fixed If the source address is beat unaligned, specify SAD = 0b (increment). If the source address is beat aligned, both SAD = 0b and 1b can be set.
19 to 16	DDS[3:0]	0h	RW	Destination Data Size Set the destination transfer size. Set Transfer Size Remarks 0000b 8 bits Initial value 0001b 16 bits 0010b 32 bits 0011b 64 bits 0100b 128 bits 0101b 256 bits 0110b 512 bits This can be set only when the number of buffer stages is 8 or 16. Setting is prohibited when the number of buffer stages is 4. 0111b 1024 bits This can be set only when the number of buffer stages is 16. Setting is prohibited when the number of buffer stages is 4 or 8. Others — Setting prohibited
15 to 12	SDS[3:0]	0h	RW	Source Data Size Set the source transfer size. Set Transfer Size Remarks 0000b 8 bits Initial value 0001b 16 bits 0010b 32 bits 0011b 64 bits 0100b 128 bits 0101b 256 bits 0110b 512 bits This can be set only when the number of buffer stages is 8 or 16. Setting is prohibited when the number of buffer stages is 4. 0111b 1024 bits This can be set only when the number of buffer stages is 16. Setting is prohibited when the number of buffer stages is 4 or 8. Others — Setting prohibited
11	-	0h	R	Reserved These bits are read as 0b. The write value should always be 0b.
10 to 8	AM[2:0]	0h	RW	ACK Mode Sets the DMAACKn output mode. 000b: Pulse mode (active for 1 clock) 001b: Level mode (active until the selected DMAREQ input becomes inactive) 01xb: Bus cycle mode (DMA transfer is active during bus cycle) 1xxb: Do not output DMAACKn
7	-	0h	R	Reserved These bits are read as 0b. The write value should always be 0b.
6	LVL	0h	RW	Level Select whether to detect the DMA request at the signal level or at the edge. 0b: Detect at edge (Initial value) 1b: Detect by level

Bit	Bit Name	Initial Value	R/W	Description
5	HIEN	0h	RW	High Enable Select to detect the DMA request at the high level or rising edge of the signal. If LVL = 0b: HIEN = 1b: Recognizes that there was a request when the signal rises HIEN = 0b: Does not recognize the request even when the signal rises If LVL = 1b: HIEN = 1b: Recognizes that there was a request when the signal is high HIEN = 0b: Does not recognize the request even if the signal is high
4	LOEN	0h	RW	Low Enable Select to detect the DMA request at the low level or falling edge of the signal. If LVL = 0b: LOEN = 1b: Recognizes that there was a request when the signal falls LOEN = 0b: Does not recognize the request even if the signal falls If LVL = 1b: LOEN = 1b: Recognizes that there was a request when the signal is low LOEN = 0b: Does not recognize the request even if the signal is low
3	REQD	0h	RW	Request Direction Selects whether the DMAREQ selected by the SEL bit is on the source side or the destination side. This bit also selects the timing at which DMAACK becomes active. 0b: Source side. DMAACK is active when reading 1b: Destination side, DMAACK is active when writing
2 to 0	SEL[2:0]	0h	RW	Terminal Select Select one from eight DMAREQ / DMAACK / DMATCO signals. SEL[2:0] Select Signal Remarks 000b DMAREQ[0], DMAACK[0], DMATCO[0] 001b DMAREQ[1], DMAACK[1], DMATCO[1] 010b DMAREQ[2], DMAACK[2], DMATCO[2] 011b DMAREQ[3], DMAACK[3], DMATCO[3] 100b DMAREQ[4], DMAACK[4], DMATCO[4] 101b DMAREQ[5], DMAACK[5], DMATCO[5] 110b DMAREQ[6], DMAACK[6], DMATCO[6] 111b DMAREQ[7], DMAACK[7], DMATCO[7]

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

NOTE

If units with different clocks are DMA transfer targets and require DMAACK, the DMAACK signal may not be received successfully due to the synchronization clock. In such a case, set AM[2:0] to 001b or 010b so that DMAACK becomes active for a long time.

4.7.5.2.13 Channel Interval Register n (DMACm_CHITVL_n) (m = 0 to 4, n = 0 to 15)

This register sets the transfer interval for DMAC channel n.

Access Size : 32 bits

Address : <DMACm_base> + 0030h + k x 0040h (k = 0 to 7)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ITVL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
15 to 0	ITVL[15:0]	0h	RW	Sets the channel transfer interval.

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.14 Channel Extension Register n (DMACm_CHEXT_n) (m = 0 to 4, n = 0 to 15)

This register functions as the DMAC channel n expansion register.

Access Size : 32 bits

Address : <DMACm_base> + 0034h + k x 0040h (k = 0 to 7)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCA[3:0]				-	DPR[2:0]			SCA[3:0]				-	SPR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
15 to 12	DCA[3:0]	0h	RW	Destination CACHE Sets the output value to CACHE[3:0] of DMA write transfer. The initial value is 0000b.
11	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
10 to 8	DPR[2:0]	0h	RW	Destination PROT Sets the output value to PROT[2:0] of DMA write transfer. The initial value is 000b.
7 to 4	SCA[3:0]	0h	RW	Source CACHE Sets the output value to CACHE[3:0] of DMA read transfer. The initial value is 0000b.
3	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
2 to 0	SPR[2:0]	0h	RW	Source PROT Sets the output value to PROT[2:0] of DMA read transfer. The initial value is 000b.

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.15 Next Link Address Register n (DMACm_NXLA_n) (m = 0 to 4, n = 0 to 15)

This register sets the descriptor address to load next in link mode.

Access Size : 32 bits

Address : <DMACm_base> + 0038h + k x 0040h (k = 0 to 7)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NXLA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NXLA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NXLA[31:0]	0h	RW	Set the link destination address. The lower 2 bits are masked with 0b. Only word-aligned addresses can be set.

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.16 Current Link Address Register n (DMACm_CRLA_n) (m = 0 to 4, n = 0 to 15)

This is a 32-bit register that displays the address of the currently executing descriptor.

Access Size : 32 bits
Address : <DMACm_base> + 003Ch + k x 0040h (k = 0 to 7)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRLA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRLA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRLA[31:0]	0h	R	The address of the currently executing descriptor is displayed.

Note: k = 0: n = 0, n = 8
k = 1: n = 1, n = 9
k = 2: n = 2, n = 10
k = 3: n = 3, n = 11
k = 4: n = 4, n = 12
k = 5: n = 5, n = 13
k = 6: n = 6, n = 14
k = 7: n = 7, n = 15

4.7.5.2.17 DMA Control Register (DMACm_DCTRL) (m = 0 to 4)

This register sets the transfer type for descriptor access and arbitration between channels.

Access Size : 32 bits
Address : <DMACm_base> + 0300h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LWCA[3:0]				-	LWPR[2:0]			LDCA[3:0]				-	LDPR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LVINT	PR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	LWCA[3:0]	0h	RW	Link Writeback CACHE Sets the value to be output to CACHE[3:0] during descriptor write back in link mode.
27	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
26 to 24	LWPR[2:0]	0h	RW	Link Writeback PROT Sets the value to be output to MHPROT[2:0] during descriptor write-back in link mode.
23 to 20	LDCA[3:0]	0h	RW	Link Descriptor CACHE Sets the value to be output to CACHE[3:0] when loading a descriptor in link mode.
19	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
18 to 16	LDPR[2:0]	0h	RW	Link Descriptor PROT Sets the value to be output to MHPROT[2:0] when loading a descriptor in link mode.
15 to 2	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
1	LVINT	0h	RW	Sets whether to output DMAEND[7:0], DMAERR as a pulse or level. 0b: Pulse output (initial value) 1b: Level output
0	PR	0h	RW	Sets the transfer priority control mode between channels 0b: Fixed priority mode (initial value) 1b: Round robin mode

4.7.5.2.18 DMA Status EN Register (DMACm_DST_EN) (m = 0 to 4)

This register displays the EN bit status of all channels.

Writing to this register does not change the value of each bit.

Access Size : 32 bits

Address : <DMACm_base> + 0310h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved These bits are read as 0b.
7	EN7	0h	R	Displays the state of the DMAC channel 7 EN bit.
6	EN6	0h	R	Displays the state of the DMAC channel 6 EN bit.
5	EN5	0h	R	Displays the state of the DMAC channel 5 EN bit.
4	EN4	0h	R	Displays the state of the DMAC channel 4 EN bit.
3	EN3	0h	R	Displays the state of the DMAC channel 3 EN bit.
2	EN2	0h	R	Displays the state of the DMAC channel 2 EN bit.
1	EN1	0h	R	Displays the state of the DMAC channel 1 EN bit.
0	EN0	0h	R	Displays the state of the DMAC channel 0 EN bit.

NOTE

Channels that do not exist in the channel configuration are set to 0b.

4.7.5.2.19 DMA Status ER Register (DMACm_DST_ER) (m = 0 to 4)

This register displays the ER bit status of all channels.

Writing to this register does not change the value of each bit.

Access Size : 32 bits

Address : <DMACm_base> + 0314h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved These bits are read as 0b.
7	ER7	0h	R	Displays the state of the DMAC channel 7 ER bit.
6	ER6	0h	R	Displays the state of the DMAC channel 6 ER bit.
5	ER5	0h	R	Displays the state of the DMAC channel 5 ER bit.
4	ER4	0h	R	Displays the state of the DMAC channel 4 ER bit.
3	ER3	0h	R	Displays the state of the DMAC channel 3 ER bit.
2	ER2	0h	R	Displays the state of the DMAC channel 2 ER bit.
1	ER1	0h	R	Displays the state of the DMAC channel 1 ER bit.
0	ER0	0h	R	Displays the state of the DMAC channel 0 ER bit.

NOTE

Channels that do not exist in the channel configuration are set to 0b.

4.7.5.2.20 DMA Status END Register (DMACm_DST_END) (m = 0 to 4)

This register displays the END bit status of all channels.

Writing to this register does not change the value of each bit.

Access Size : 32 bits

Address : <DMACm_base> + 0318h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	END7	END6	END5	END4	END3	END2	END1	END0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved These bits are read as 0b.
7	END7	0h	R	Displays the state of the DMAC channel 7 END bit.
6	END6	0h	R	Displays the state of the DMAC channel 6 END bit.
5	END5	0h	R	Displays the state of the DMAC channel 5 END bit.
4	END4	0h	R	Displays the state of the DMAC channel 4 END bit.
3	END3	0h	R	Displays the state of the DMAC channel 3 END bit.
2	END2	0h	R	Displays the state of the DMAC channel 2 END bit.
1	END1	0h	R	Displays the state of the DMAC channel 1 END bit.
0	END0	0h	R	Displays the state of the DMAC channel 0 END bit.

NOTE

Channels that do not exist in the channel configuration are set to 0b.

4.7.5.2.21 DMA Status TC Register (DMACm_DST_TC) (m = 0 to 4)

This register displays the TC bit status of all channels.

Writing to this register does not change the value of each bit.

Access Size : 32 bits

Address : <DMACm_base> + 031Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved These bits are read as 0b.
7	TC7	0h	R	Displays the state of the DMAC channel 7 TC bit.
6	TC6	0h	R	Displays the state of the DMAC channel 6 TC bit.
5	TC5	0h	R	Displays the state of the DMAC channel 5 TC bit.
4	TC4	0h	R	Displays the state of the DMAC channel 4 TC bit.
3	TC3	0h	R	Displays the state of the DMAC channel 3 TC bit.
2	TC2	0h	R	Displays the state of the DMAC channel 2 TC bit.
1	TC1	0h	R	Displays the state of the DMAC channel 1 TC bit.
0	TC0	0h	R	Displays the state of the DMAC channel 0 TC bit.

NOTE

Channels that do not exist in the channel configuration are set to 0b.

4.7.5.2.22 DMA Status SUS Register (DMACm_DST_SUS) (m = 0 to 4)

This register displays the SUS bit status of all channels.

Writing to this register does not change the value of each bit.

Access Size : 32 bits

Address : <DMACm_base> + 0320h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SUS7	SUS6	SUS5	SUS4	SUS3	SUS2	SUS1	SUS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved These bits are read as 0b.
7	SUS7	0h	R	Displays the state of the DMAC channel 7 SUS bit.
6	SUS6	0h	R	Displays the state of the DMAC channel 6 SUS bit.
5	SUS5	0h	R	Displays the state of the DMAC channel 5 SUS bit.
4	SUS4	0h	R	Displays the state of the DMAC channel 4 SUS bit.
3	SUS3	0h	R	Displays the state of the DMAC channel 3 SUS bit.
2	SUS2	0h	R	Displays the state of the DMAC channel 2 SUS bit.
1	SUS1	0h	R	Displays the state of the DMAC channel 1 SUS bit.
0	SUS0	0h	R	Displays the state of the DMAC channel 0 SUS bit.

NOTE

Channels that do not exist in the channel configuration are set to 0b.

4.7.6 Functional Description

The prefix (DMACm_) of the register names is omitted in this and subsequent sections.

4.7.6.1 DMA Mode

This section describes the DMAC mode specifications of the DMAC unit.

4.7.6.1.1 Mode setting

The DMA mode can be switched between register mode and link mode according to the value of the DMS field in the CHCFG_n register.

Table 4.7-5 DMA Mode Setting

DMS (CHCFG_n)	Mode	Description
0b	Register mode	The values set in the next register set are used to proceed with DMA transfer.
1b	Link mode	The descriptor is set in the current register to proceed with DMA transfer. Loading of the descriptor and DMA transfer are repeated unless these are stopped with the setting by the descriptor or by the control register.

4.7.6.1.2 Register mode

In register mode, the values set in the internal registers are used for DMA transfer.

Two types (Next0 Register Set and Next1 Register Set) of transfer source address, transfer destination address, and number of transfer bytes can be set. The Next register set to be used can be selected and transferred, or two Next register sets can be transferred consecutively.

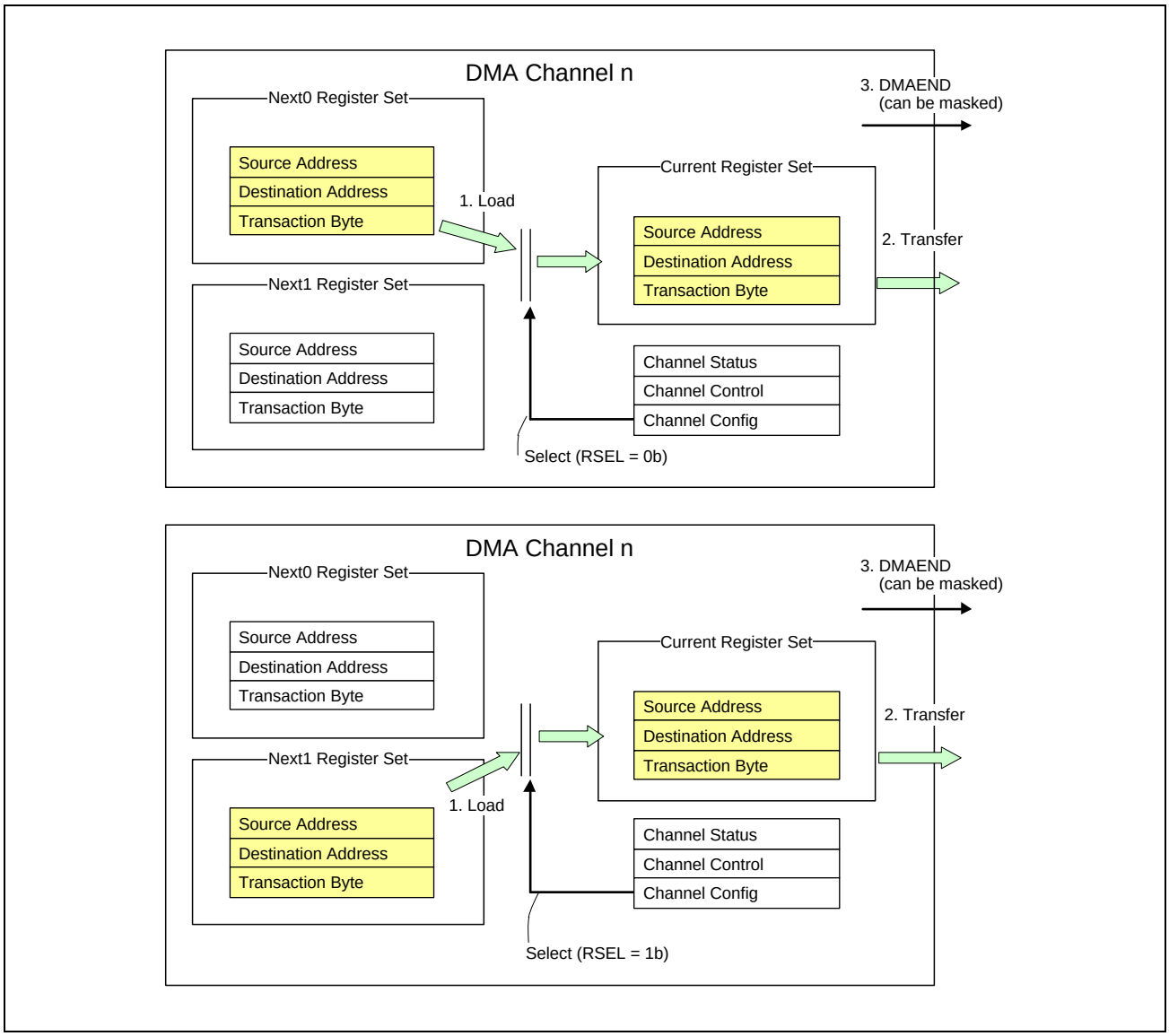


Figure 4.7-5 Overview of Register Normal Mode

The above figure illustrates operations when Next0 is executed (upper part of the figure) and when Next1 is executed (lower part of the figure).

(1) Flow of operations

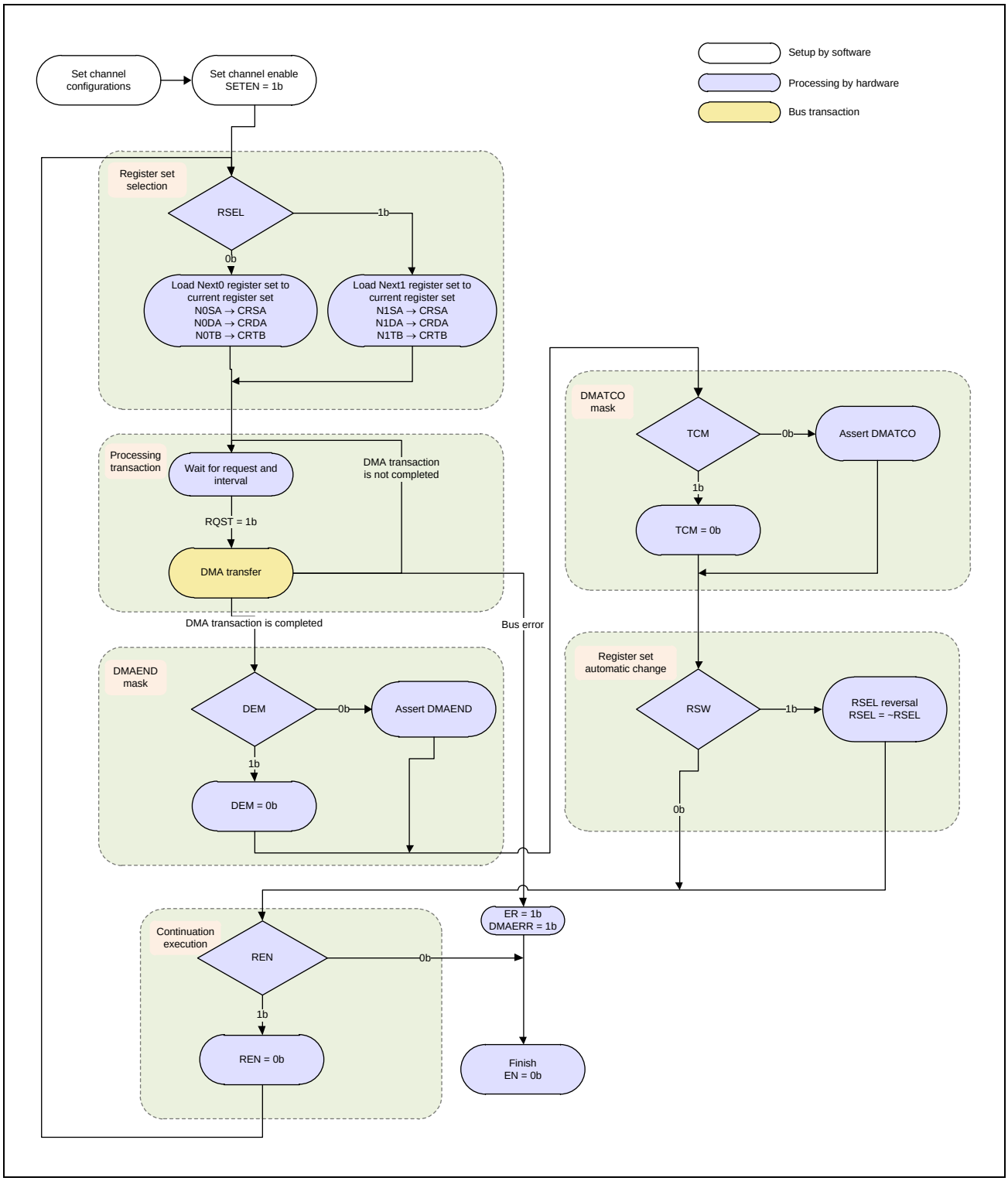


Figure 4.7-6 Register Mode Flow

<Description of the Register Mode Flow>

1. Set channel configuration

   In addition, set the DMA register set (REQ, DMAACK, the amount of data for transfer, etc.) in the channel register set (See **4.7.6.2 DMA Transfer**).

2. Register set selection

   When the EN bit is set to 1b, the setting of the next register set selected by RSEL is loaded to the current register set.

3. Processing DMA transaction

   DMA transfer proceeds in accord with the set value. For details of transfer, see **4.7.6.2 DMA Transfer**.

4. Masking DMAEND

   DMAEND is masked in accord with the value set in the DEM bit of CHCFG_n. DMAEND is not output if DEM = 1b. After that, DEM is automatically cleared to 0b.

5. Masking DMATCO

   DMATCO is masked in accord with the value set in the TCM bit of CHCFG_n. DMATCO is not output if TCM = 1b. After that, TCM is automatically cleared to 0b.

6. Register set automatic change

   Whether or not the other next register set is to be used is determined by the value set in the RSW bit of CHCFG_n.

7. Continuous execution

   Whether or not DMA transfer is to be consecutively executed is determined by the value set in the REN bit of CHCFG_n. Transfer is executed consecutively if REN = 1b. After that, REN is automatically cleared to 0b.

(2) Register settings

(a) Register mode setting

Select the register set to be executed.

Table 4.7-6 Register Mode Setting

DMS (CHCFG_n)	RSEL (CHCFG_n)	Description
0b	0b	Execute the Next0 register set.
	1b	Execute the Next1 register set.

(b) DMAEND mask setting

DMAEND can be masked for each register set.

Table 4.7-7 DMAEND Mask Setting

DEM (CHCFG_n)	Operation	Remarks
0b	DMAEND is issued on completion of the DMA transaction.	
1b	DMAEND is not issued even if the DMA transaction has completed. DEM is cleared to 0b by hardware following the completion of the DMA transaction.	

(c) DMATCO mask setting

DMATCO can be masked for each register set.

Table 4.7-8 DMATCO Mask Setting

TCM (CHCFG_n)	Operation	Remarks
0b	DMATCO is issued on completion of the DMA transaction.	
1b	DMATCO is not issued even if the DMA transaction has completed. TCM is cleared to 0b by hardware following the completion of the DMA transaction.	

(d) Automatic register set execution setting

After DMA transfer, a DMA transaction of the selected register set is automatically executed.

Table 4.7-9 Automatic Register Set Execution Setting

REN (CHCFG_n)	Operation	Remarks
0b	When the DMA transaction of the register set that is set in the RSEL bit is completed, the EN bit is cleared to end DMA operation.	Set this to execute the DMA transaction once.
1b	After the completion of the DMA transaction, DMA transfer of the contents of the selected register set follows. When continuous transfer is established, REN is cleared to 0b.	Set this to execute the contents of the register set consecutively.

(e) Automatic register set execution setting

When REN = 1b, the next register set to be automatically executed can be switched following the completion of the DMA transaction.

Table 4.7-10 Automatic Register Set Execution Setting

RSW (CHCFG_n)	Operation	Remarks
0b	When REN = 1b and the DMA transaction is completed, the register set is not switched.	Set this when using one register set only.
1b	When REN = 1b and the DMA transaction is completed, RSEL is automatically inverted and the other register set is selected.	Set this when switching the register set.

(3) Example settings

(a) When using the next0 register set only

Table 4.7-11 Example Register Mode Setting 1

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	TCM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0b (Register mode)	0b (Next0)	0b (No masking)	0b (No masking)	0b (No switching)	0b (No continuous execution)

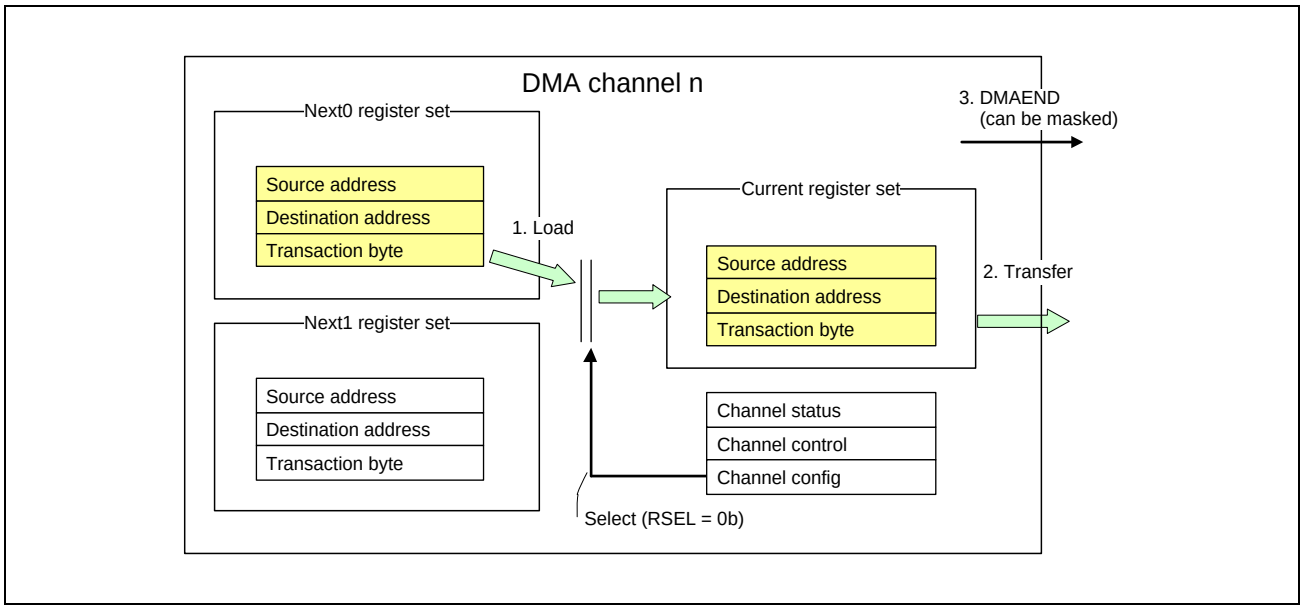


Figure 4.7-7 Example Register Mode Setting 1

1. Set EN = 1b (SETEN = 1b) to load the Next0 register set to the current register set.
2. Execute a DMA transaction according to the values of the current register set and channel register set.
3. DMAEND is issued following the completion of the DMA transaction.
4. DMATCO is issued following the completion of the DMA transaction.
5. Since REN is 0b, clear EN to 0b to end the operation.

(b) When using two register sets consecutively

Table 4.7-12 Automatic Register Set Execution Setting

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	TCM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0b (Register mode)	0b (Next0)	1b (Masking)	0b (No masking)	1b (Switching)	1b (Continuous execution)

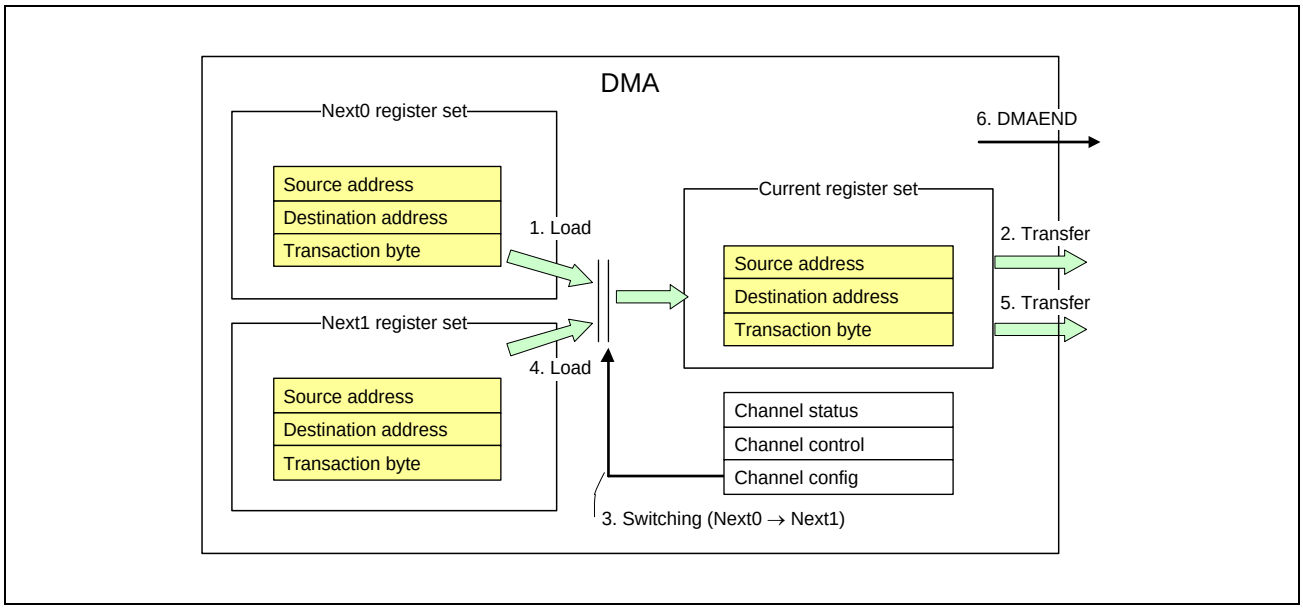


Figure 4.7-8 Example Register Mode Setting 2

1. Set EN = 1b (SETEN = 1b) to load the Next0 register set to the current register set.
2. Execute a DMA transaction according to the values of the current register set and channel register set.
3. Since DEM is 1b, DMAEND is not output following the completion of the DMA transaction. DEM is automatically cleared to 0b.
4. Since REN is 1b, the operation is executed continuously. REN is automatically cleared to 0b.
5. Since RSW is 1b, the next register set to be executed is switched (RSEL = 0b → 1b).
6. Load the Next1 register set to the current register set.
7. Execute a DMA transaction according to the values of the current register set and channel register set.
8. Since DEM is 0b, DMAEND is issued following the completion of the DMA transaction.
9. Since TCM is 0b, DMATCO is issued following the completion of the DMA transaction.
10. Since REN is 0b, clear EN to 0b to end the operation.

4.7.6.1.3 Link mode

In link mode, a DMA transaction is executed by loading descriptors in the external memory area as the settings. The DMAC has a next link address and current link address per channel and these are used to set the address of a next descriptor to be executed and to indicate the address of a descriptor for the DMA transaction that is currently in progress, respectively.

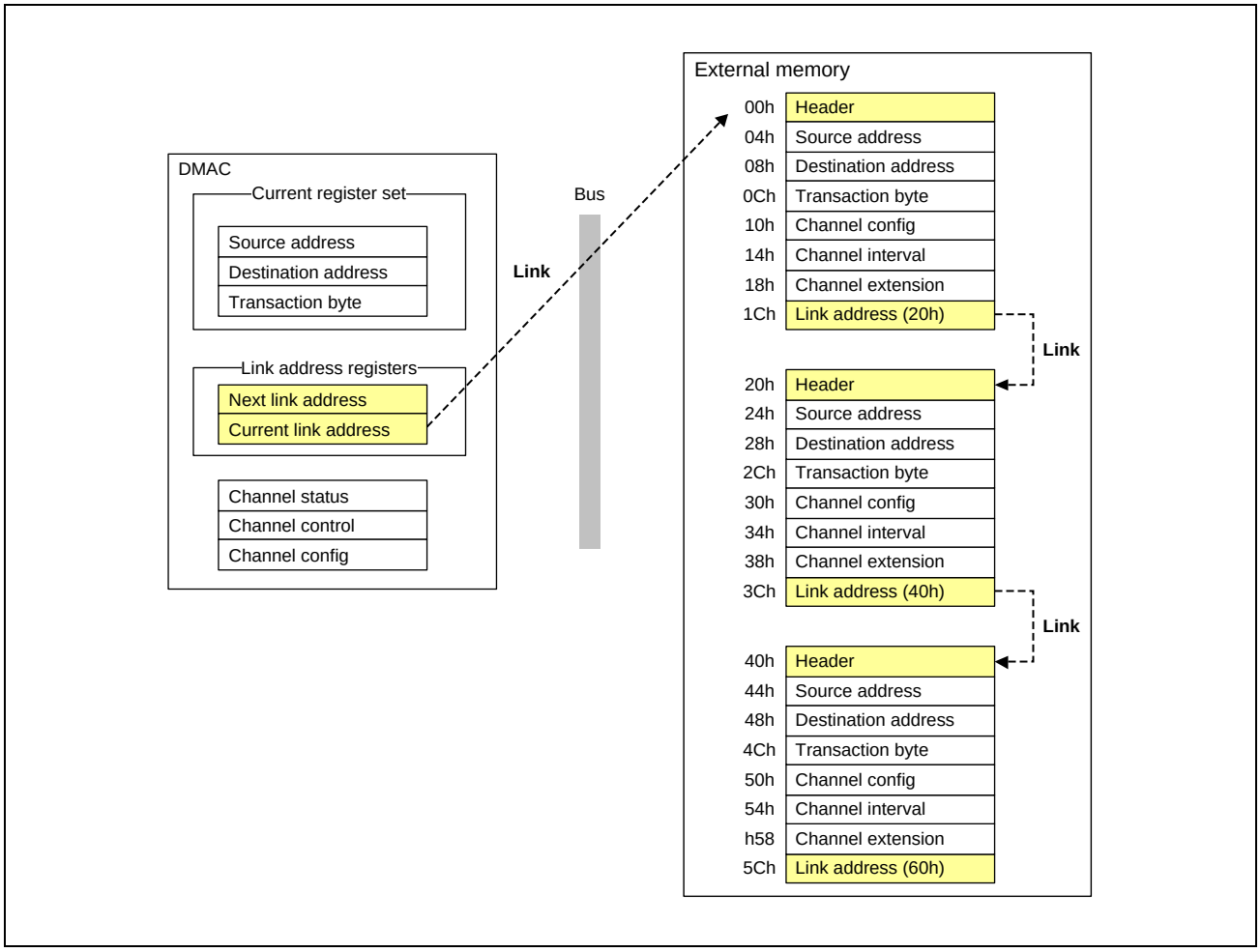


Figure 4.7-9 Overview of Link Mode

(1) Flow of operations

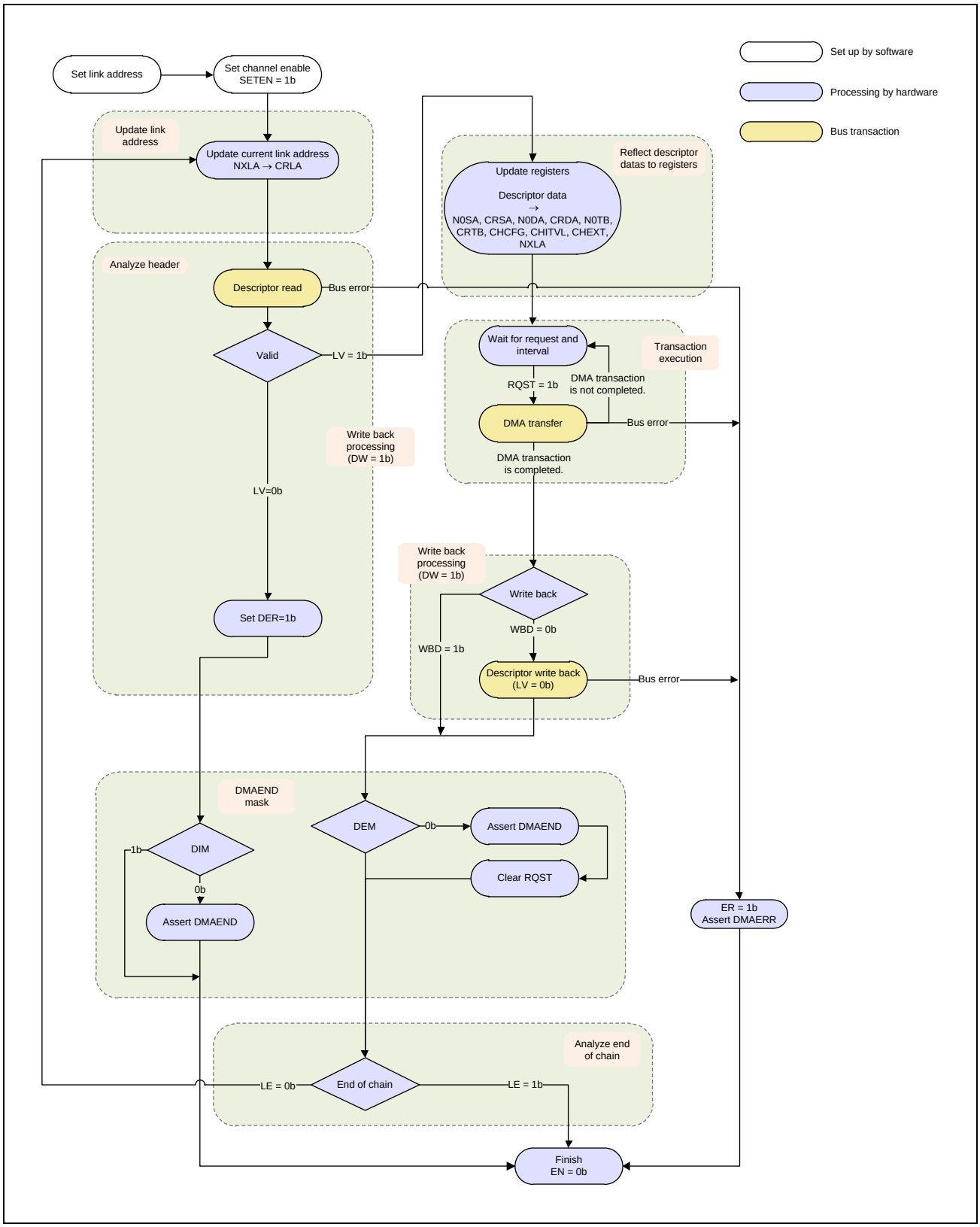


Figure 4.7-10 Flow of Link Mode

<Description of the Link Mode Flow>

1. Set channel configuration

   Set the address where the link destination is to start in NXLA_n.

2. Link address update

   Setting EN = 1b (setting SETEN to 1b) loads the link address set in NXLA_n to CRLA_n.

3. Reading the descriptor and judging the header

   The DMAC starts to load the descriptor and checks the header contents. If LV = 0b, the descriptor is discarded and DER is set to 1b indicating the end state (EN = 0b). If DIM within the header is 0b at this time, DMAEND is issued.

4. Descriptor setting

   Set the loaded descriptor in the current register set and channel register set. Also, set the next link destination in NXLA_n.

5. Processing DMA transaction

   Execute a DMA transaction according to the set values.

6. Header write-back

   If WBD in the header = 0b, the DMAC writes back 0b to the LV bit in the header.

7. Masking DMAEND

   If the DEM bit in CHCFG_n is 0b, DMAEND is issued.

8. Masking DMATCO

   If the TCM bit in CHCFG_n is 0b, DMATCO is issued.

9. Judging the end of the link

   If LE in the header = 1b, after the transfer according to the descriptor settings, TCO is issued (enabling masking by CHCFG_n) and EN is cleared to 0b to stop the operation. If LE = 0b, the current register set is updated and the DMAC starts to load the next descriptor.

(2) Register settings

(a) Link mode setting

To use link mode, set the DMS bit in the CHCFG_n register to 1b.

Table 4.7-13 Link Mode Setting

DMS (CHCFG_n)	Description
1b	Operation is in link mode. The value of this bit cannot be changed by the descriptor.

(b) LINk address setting

There are the next link address register and the current link address register as registers which indicate the link destination.

When starting link mode, set the link destination in the next link address register.

The next link address indicates the next link destination after loading of the descriptor. The current link address indicates the address of the link currently being executed.

Table 4.7-14 Link Address Register Set

Register	Description
Next Link Address Register (NXLA_n)	Sets and indicates the next link destination. Before starting link mode, set the address of the link destination in this register.
Current Link Address Register (CRLA_n)	Indicates the link destination currently being executed. This register is only readable.

CAUTION

In link mode, the setting can be changed by reading the descriptor; however the timing of the change to the setting and a hardware request cannot be synchronized. Therefore, when using hardware requests, set AM, LVL, HIEN, LOEN, and SEL of the CHCFG_n register before enabling hardware requests and also make sure that these setting bits have the same values in the descriptor.

(3) Descriptor settings

From the link destination address, prepare the parts of each descriptor in the following order.

The DMAC reads the descriptors in bursts.

(a) Descriptor arrangement

Table 4.7-15 Descriptor Setting Arrangement

Address	Data	Remark
LinkAddress + 00h	Header	
LinkAddress + 04h	Source address	
LinkAddress + 08h	Destination address	
LinkAddress + 0Ch	Transaction byte	
LinkAddress + 10h	Config	Register mode cannot be set.
LinkAddress + 14h	Interval	
LinkAddress + 18h	Extension	
LinkAddress + 1Ch	Next link address	

Note: For the link destination, set the addresses aligned in 32-bit boundaries.

(b) Header

The header indicates the descriptor state as shown below.

This area is automatically read by the DMAC before starting DMA transfer in link mode. After the completion of the DMA transaction, the DMAC writes back the header and sets the LV bit to 0b to indicate the state of transfer.

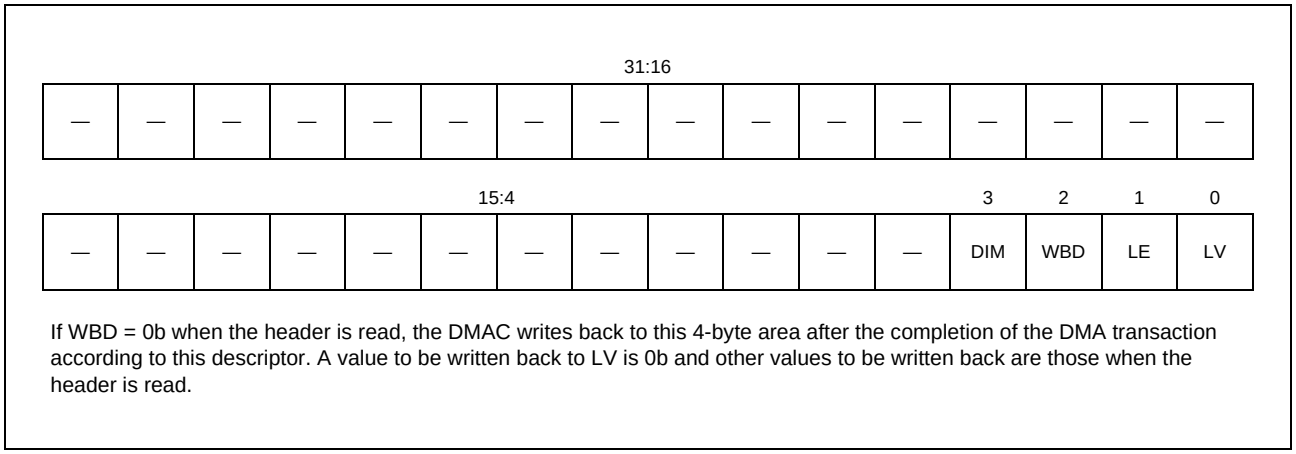


Figure 4.7-11 Header Area

Table 4.7-16 Header Area

Bit Position	Bit Name	Function
31:4	—	—
3	DIM	Descriptor Interrupt Mask If LV = 0b when the header is loaded, this bit sets whether or not to mask DMAEND. 0b: DMAEND is issued. 1b: DMAEND is not issued.
2	WBD	Write Back Disable This bit masks execution of write-back to the LV bit. When the setting of this bit is 1b, the DMAC does not proceed with write-back operation. 0b: The LV bit is written back to 0b. 1b: The LV bit is not written back.
1	LE	Link End This bit indicates that the link ends in the DMA transaction of this descriptor. Set this bit to 1b to indicate the end of the link. 0b: The link continues. 1b: The link ends.
0	LV	Link Valid This bit indicates that this descriptor is enabled. When WBD = 0b, the DMAC writes 0b to this bit following the execution of the DMA transaction written in the descriptor. When setting the header, set this bit to 1b. 0b: The descriptor is disabled. 1b: The descriptor is enabled

(c) Setting the descriptors other than the header

The values in the descriptors other than the header have the same specifications as those of the internal registers (however, the DMS bit of the CHCFG_n register cannot be modified by the descriptor). For the specifications of the internal registers, see **4.7.5 Register Descriptions**.

For example settings of the descriptors, see **4.7.7.1.4 Example setting 4 (link mode)**.

(d) PROT and CACHE settings at the time of access to the descriptors

The PROT and CACHE settings at the time of access to the descriptors can be set in LWCA, LWPR, LDCA, and LDPR described in **4.7.5.2.17 DMA Control Register (DMACm_DCTRL) (m = 0 to 4)**. Make the settings according to the destination for access in which the descriptors are prepared.

(e) Descriptor areas and DMA transfer areas

The figure below shows an overview of the descriptor areas and DMA transfer areas, which are accessed by the DMAC.

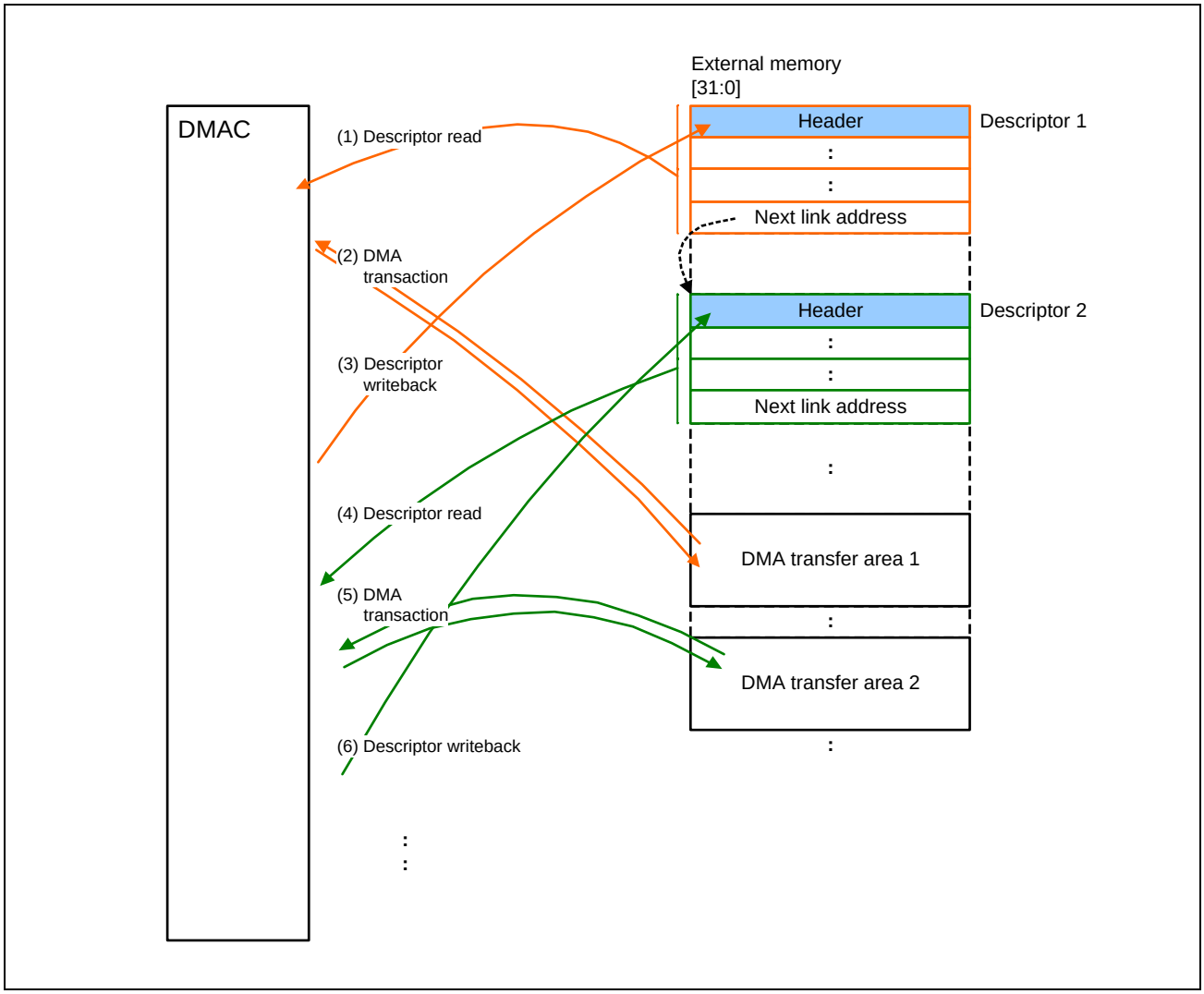


Figure 4.7-12 Overview of the Descriptor Areas and DMA Transfer Areas

(1) Descriptor read

Load the value set in the internal next link address register to the current link address register, and then read the descriptor from the external memory space (descriptor 1) indicated by the current link address register.

(2) DMA transfer

If LV in the header of the descriptor = 1b, execute DMA transfer according to the descriptor information.

(3) Descriptor write-back

After the DMA transfer for the specified number of bytes, if WBD in the header = 0b, write-back is executed on the header of descriptor 1 in the burst size of 32 bits with 0b for LV and the value read in step 1 for the other fields as data.

(4) Descriptor read

If LE in the header of the descriptor which has been previously read (in step 1) = 0b, read the next descriptor from the address (descriptor 2) indicated by the next link address in the descriptor.

(5) DMA transfer

If LV in the header of the descriptor = 1b, execute DMA transfer according to the descriptor information.

(6) Descriptor write-back

After the DMA transfer for the specified number of bytes, if WBD in the header = 0b, write-back is executed on the header of descriptor 2 in the burst size of 32 bits with 0 for LV and the value read in step 4 for the other fields as data.

After that, repeat steps (4) to (6).

If LE in the header = 1b and WBD = 0b, DMA transfer proceeds in accord with the descriptor settings, and 0 is written back to the LV bit of the header, after which the operation ends.

If LE in the header = 1b and WBD = 1b, DMA transfer proceeds in accord with the descriptor settings, after which the operation ends (write-back does not proceed).

If LV in the header = 0b, the operation stops (DMA transfer does not proceed).

(4) Example descriptor configuration

In link mode, descriptors can be configured as shown in the figure below.

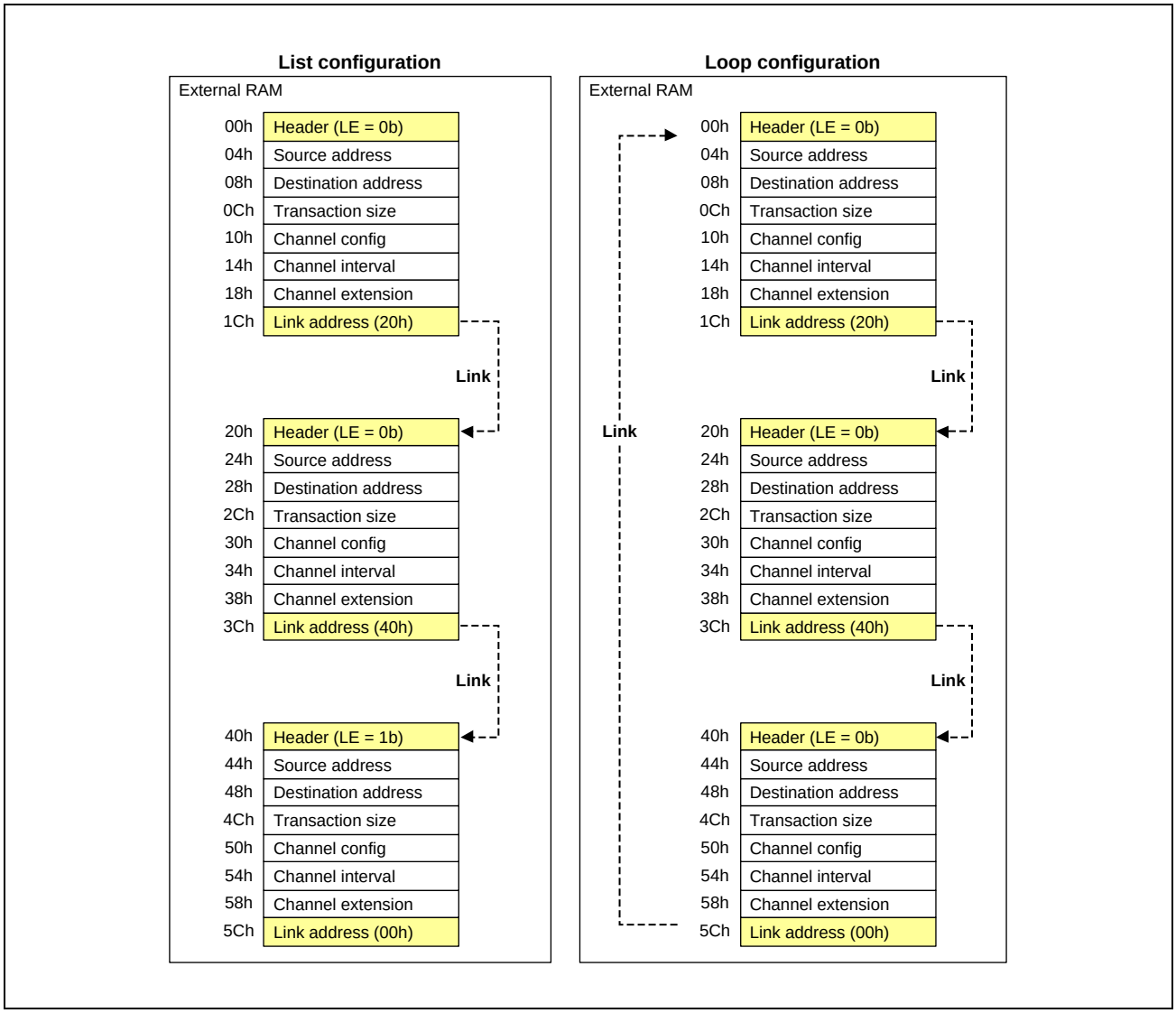


Figure 4.7-13 Example Descriptor Configuration

- List configuration

  Setting the LE bit in the header of the last descriptor to 1b ends the link.

- Loop configuration

  Setting the address of the first descriptor as the link address for the last descriptor configures the descriptors in a loop. To end the loop, either overwrite the LE bit of the header with 1b before the DMAC reads the descriptor or stop the DMAC according to the transfer suspension procedure.

4.7.6.2 DMA Transfer

This section describes basic operations of DMA transfer by the DMAC unit.

4.7.6.2.1 Transfer mode

As the transfer mode, single-transfer mode and block-transfer mode are supported.

Select the mode by using the TM bit of CHCFG_n per channel.

Use single-transfer mode for normal DMA transfer with the use of the DMAREQm input and block-transfer mode for memory-to-memory transfer, etc., with the use of software activation.

Table 4.7-17 Basic Transfer Setting

Transfer Mode	TM (CHCFG_n)	Function	Usage
Single transfer	0b	A single round of DMA transfer proceeds in response to a single DMAREQ.	Use this mode for DMA transfer with the use of the DMAREQ input.
Block transfer	1b	Transfer proceeds in response to a single software activation until the completion of the DMA transaction.	Use this mode for memory-to-memory transfer with the use of software activation.

(1) Single-transfer mode

When a DMA transfer request is received, DMA transfer is executed once in the direction (source or destination) indicated by REQD and DMAACK is asserted. A single round of transfer proceeds every time a transfer request is received and this operation is repeated for the number of bytes loaded to CRTB_n (arbitration between channels proceeds per DMA transfer).

The timing of DMAACK and the timing of counting CRTB vary with the setting of REQD and the transfer size (DDS, SDS). For details, see **4.7.6.2.9 Differences in operation with the transfer size**.

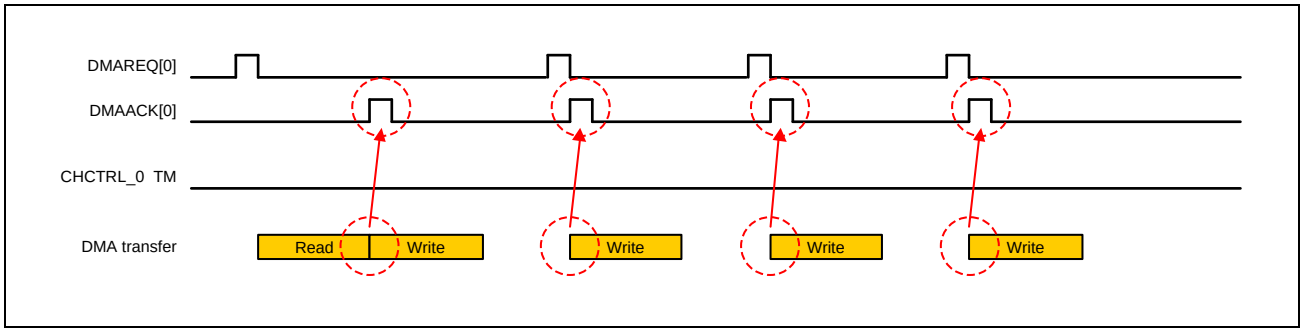


Figure 4.7-14 Single-Transfer Mode (REQD = 1b, SDS > DDS)

(2) Block-transfer mode

Once a DMA transfer request is received, transfer continues until a transfer for the number of bytes loaded to the DMA transfer byte register (CRTB_n register) is completed (the completion of the DMA transaction) (arbitration between channels proceeds per DMA transfer).

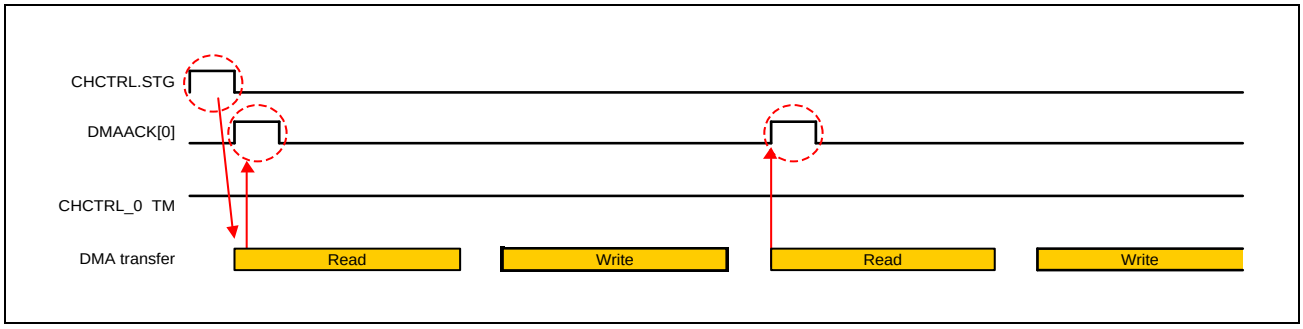


Figure 4.7-15 Block-Transfer Mode (REQD = 0b, SDS < DDS)

4.7.6.2.2 Control over the priority of DMAC channels

Fixed priority mode and round-robin mode are supported as the order of priority among channels 0 to 7 of DMACA and among channels 0 to 7 of DMACB. The mode is selected by using the PR bit of the DMA control register (DCTRL). When the setting of the PR bit is 0b, operation is in fixed priority mode; when it is 1b, operation is in round-robin mode.

Table 4.7-18 Priority Control Setting

Mode	PR (DCTRL)	Function	Usage
Fixed priority	0b	Controls requests in the fixed priority order (high: CH0 > CH1.> CH2 > CH3 > CH4 > CH5 > CH6 > CH7: low).	Use this mode when channels have the priority order.
Round-robin	1b	Controls requests in round-robin fashion.	Use this mode to execute each request evenly.

Round-robin mode is only supported as the order of priority between DMACA and DMACB. In round-robin mode, the DMAC of DMAC No. (which is currently executing transfer) + 1 has the highest priority. After a reset, DMACA is given the highest priority.

Table 4.7-19 Priority of Transfer Requests for DMAC during Transfer

Current DMAC \ Next DMAC	DMACA	DMACB
DMACA	2	1
DMACB	1	2

Note: Priority: 1 (high) ⇔ 2 (low)

The priority of reading and that of writing are controlled independently.

The DMAC issues transfer requests to the individual channels concurrently without waiting for the completion of transfer and handles them in the order the responses are returned. Therefore, the orders in which transactions start and end do not necessarily match.

(1) Fixed priority mode

In fixed priority mode, the order of priority among channels 0 to 7 of DMACA and among channels 0 to 7 of DMACB is fixed. The order of priority between DMACA and DMACB is round-robin. The order of priority after a reset is as follows.

High D0_CH0 > D1_CH0 > D0_CH1 > D1_CH1 > D0_CH2 > D1_CH2 > D0_CH3 > D1_CH3 > D0_CH4 > D1_CH4 > D0_CH5 > D1_CH5 > D0_CH6 > D1_CH6 > D0_CH7 > D1_CH7 Low

Note: D0 and D1 refer to DMACA and DMACB, respectively.

If a transfer request for channel 0 of DMACA is output in this state, transfer by channel 0 of DMACA proceeds. The order of priority after the end of transfer is as follows.

High D1_CH1 > D0_CH0 > D1_CH1 > D0_CH1 > D1_CH2 > D0_CH2 > D1_CH3 > D0_CH3 > D1_CH4 > D0_CH4 > D1_CH5 > D0_CH5 > D1_CH6 >D0_CH6 > D1_CH7 > D0_CH7 Low

Note: D0 and D1 refer to DMACA and DMACB, respectively.

If DMA transfer requests are generated on multiple channels simultaneously, the request for a smaller number channel is given priority. The figure below shows an example when another DMA transfer request with higher priority is generated while DMA transfer proceeds in fixed priority mode.

(2) Round-robin mode

In round-robin mode, the order of priority is changed so that the channel which has proceeded with the immediately preceding transfer is given the lowest priority every time a transfer among channels 0 to 7 of DMACA and among channels 0 to 7 of DMACB is accepted.

The order of priority between DMACA and DMACB is also determined in round-robin.

As with fixed priority mode, the order or priority after a reset is as follows.

High D0_CH0 > D1_CH0 > D0_CH1 > D1_CH1 > D0_CH2 > D1_CH2 > D0_CH3 > D1_CH3 > D0_CH4 > D1_CH4 > D0_CH5 > D1_CH5 > D0_CH6 > D1_CH6 > D0_CH7 > D1_CH7 Low

Note: D0 and D1 refer to DMACA and DMACB, respectively.

If a transfer request for channel 2 of DMACA is output in this state, transfer by channel 2 of DMACA proceeds. The order of priority after the end of transfer is as follows.

High D1_CH0 > D0_CH3 > D1_CH1 > D0_CH4 > D1_CH2 > D0_CH5 > D1_CH3 > D0_CH6 > D1_CH4 > D0_CH7 > D1_CH5 > D0_CH0 > D1_CH6 > D0_CH1 > D1_CH7 > D0_CH2 Low

Note: D0 and D1 refer to DMACA and DMACB, respectively.

4.7.6.2.3 DMA transfer request

Edge detection or level detection is selectable by using the LVL bit of the CHCFG_n register.

The HIEN and LOEN bits of the CHCFG_n register are used to select rising or falling edge for edge detection and the high or low level for level detection.

Table 4.7-20 DMAREQ Pin Selection Setting

SEL[2:0] (CHCFG_n)	Request Pin	Acknowledge Pin	Terminal Count Pin	Usage
000b	DMAREQ[0]	DMAACK[0]	DMATCO[0]	Set the DMAC pin to be used for channel n.
001b	DMAREQ[1]	DMAACK[1]	DMATCO[1]	
010b	DMAREQ[2]	DMAACK[2]	DMATCO[2]	
011b	DMAREQ[3]	DMAACK[3]	DMATCO[3]	
100b	DMAREQ[4]	DMAACK[4]	DMATCO[4]	
101b	DMAREQ[5]	DMAACK[5]	DMATCO[5]	
110b	DMAREQ[6]	DMAACK[6]	DMATCO[6]	
111b	DMAREQ[7]	DMAACK[7]	DMATCO[7]	

Table 4.7-21 DMAREQ Detection Setting

Mode	LVL (CHCFG_n)	HIEN (CHCFG_n)	LOEN (CHCFG_n)	Function	Usage
Edge detection	0b	0b	0b	Edge detection cannot be performed in this setting. Select this setting when not using a hardware request.	Set the mode of DMAREQ and the method for detecting rising and falling edges.
			1b	Detection of falling edges of DMAREQm.	
		1b	0b	Detection of rising edges of DMAREQm.	
			1b	Detection of rising and falling edges of DMAREQm.	
Level detection	1b	0b	0b	Level detection cannot be performed in this setting.	
			1b	DMAREQm is detected in the low-level mode.	
		1b	0b	DMAREQm is detected in high-level mode.	
			1b	Transfer starts in response to the SETEN bit of the CHCTRL_n register being set to 1b regardless of the input level of DMAREQm. In this setting, set DMAACK to the mode other than level mode. If level mode is set, the operation will be deadlocked.	

DMA activation requests have three types: software request, external request, and on-chip peripheral module request. Select the transfer request source for an external request, on-chip peripheral module request, and software request with the ICU_DMkSELy.DkRQ_SELn register (k = 0 to 4, y = 0 to 7, n = 0 to 15).

For software requests, see the description of the ICU_SWDMACk register (k = 0 to 4).

For each DMA transfer source, set the CHCFG_n register according to **Table 4.7-22**. Regarding external requests, make the setting to suit the system.

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (1/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_ SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
ICU EVT_DMAC	EVT_DMAC_0	Arbitrary	Arbitrary	00Ch	0/1	100	0	1	0	0	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
ICU EVT_DMAC	EVT_DMAC_1	Arbitrary	Arbitrary	00Dh	0/1	100	0	1	0	0	
ICU EVT_DMAC	EVT_DMAC_2	Arbitrary	Arbitrary	00Eh	0/1	100	0	1	0	0	
ICU EVT_DMAC	EVT_DMAC_3	Arbitrary	Arbitrary	00Fh	0/1	100	0	1	0	0	
ICU EVT_DMAC	EVT_DMAC_4	Arbitrary	Arbitrary	010h	0/1	100	0	1	0	0	
ICU EVT_DMAC	EVT_DMAC_5	Arbitrary	Arbitrary	011h	0/1	100	0	1	0	0	
ICU EVT_DMAC	EVT_DMAC_6	Arbitrary	Arbitrary	012h	0/1	100	0	1	0	0	
ICU EVT_DMAC	EVT_DMAC_7	Arbitrary	Arbitrary	013h	0/1	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_0	Arbitrary	Arbitrary	014h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_1	Arbitrary	Arbitrary	015h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_2	Arbitrary	Arbitrary	016h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_3	Arbitrary	Arbitrary	017h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_4	Arbitrary	Arbitrary	018h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_5	Arbitrary	Arbitrary	019h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_6	Arbitrary	Arbitrary	01Ah	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_7	Arbitrary	Arbitrary	01Bh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_8	Arbitrary	Arbitrary	01Ch	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_9	Arbitrary	Arbitrary	01Dh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_10	Arbitrary	Arbitrary	01Eh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_11	Arbitrary	Arbitrary	01Fh	0	100	0	1	0	0	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (2/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
ICU SWDMAC	DMAC_SW_Trigger_1_12	Arbitrary	Arbitrary	020h	0	100	0	1	0	0	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
ICU SWDMAC	DMAC_SW_Trigger_1_13	Arbitrary	Arbitrary	021h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_14	Arbitrary	Arbitrary	022h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_1_15	Arbitrary	Arbitrary	023h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_0	Arbitrary	Arbitrary	024h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_1	Arbitrary	Arbitrary	025h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_2	Arbitrary	Arbitrary	026h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_3	Arbitrary	Arbitrary	027h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_4	Arbitrary	Arbitrary	028h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_5	Arbitrary	Arbitrary	029h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_6	Arbitrary	Arbitrary	02Ah	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_7	Arbitrary	Arbitrary	02Bh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_8	Arbitrary	Arbitrary	02Ch	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_9	Arbitrary	Arbitrary	02Dh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_10	Arbitrary	Arbitrary	02Eh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_11	Arbitrary	Arbitrary	02Fh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_12	Arbitrary	Arbitrary	030h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_13	Arbitrary	Arbitrary	031h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_14	Arbitrary	Arbitrary	032h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_2_15	Arbitrary	Arbitrary	033h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_0	Arbitrary	Arbitrary	034h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_1	Arbitrary	Arbitrary	035h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_2	Arbitrary	Arbitrary	036h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_3	Arbitrary	Arbitrary	037h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_4	Arbitrary	Arbitrary	038h	0	100	0	1	0	0	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (3/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
ICU SWDMAC	DMAC_SW_Trigger_3_5	Arbitrary	Arbitrary	039h	0	100	0	1	0	0	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
ICU SWDMAC	DMAC_SW_Trigger_3_6	Arbitrary	Arbitrary	03Ah	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_7	Arbitrary	Arbitrary	03Bh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_8	Arbitrary	Arbitrary	03Ch	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_9	Arbitrary	Arbitrary	03Dh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_10	Arbitrary	Arbitrary	03Eh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_11	Arbitrary	Arbitrary	03Fh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_12	Arbitrary	Arbitrary	040h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_13	Arbitrary	Arbitrary	041h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_14	Arbitrary	Arbitrary	042h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_3_15	Arbitrary	Arbitrary	043h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_0	Arbitrary	Arbitrary	044h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_1	Arbitrary	Arbitrary	045h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_2	Arbitrary	Arbitrary	046h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_3	Arbitrary	Arbitrary	047h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_4	Arbitrary	Arbitrary	048h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_5	Arbitrary	Arbitrary	049h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_6	Arbitrary	Arbitrary	04Ah	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_7	Arbitrary	Arbitrary	04Bh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_8	Arbitrary	Arbitrary	04Ch	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_9	Arbitrary	Arbitrary	04Dh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_10	Arbitrary	Arbitrary	04Eh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_11	Arbitrary	Arbitrary	04Fh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_12	Arbitrary	Arbitrary	050h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_4_13	Arbitrary	Arbitrary	051h	0	100	0	1	0	0	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (4/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
ICU SWDMAC	DMAC_SW_Trigger_4_14	Arbitrary	Arbitrary	052h	0	100	0	1	0	0	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
ICU SWDMAC	DMAC_SW_Trigger_4_15	Arbitrary	Arbitrary	053h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_0	Arbitrary	Arbitrary	054h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_1	Arbitrary	Arbitrary	055h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_2	Arbitrary	Arbitrary	056h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_3	Arbitrary	Arbitrary	057h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_4	Arbitrary	Arbitrary	058h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_5	Arbitrary	Arbitrary	059h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_6	Arbitrary	Arbitrary	05Ah	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_7	Arbitrary	Arbitrary	05Bh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_8	Arbitrary	Arbitrary	05Ch	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_9	Arbitrary	Arbitrary	05Dh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_10	Arbitrary	Arbitrary	05Eh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_11	Arbitrary	Arbitrary	05Fh	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_12	Arbitrary	Arbitrary	060h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_13	Arbitrary	Arbitrary	061h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_14	Arbitrary	Arbitrary	062h	0	100	0	1	0	0	
ICU SWDMAC	DMAC_SW_Trigger_0_15	Arbitrary	Arbitrary	063h	0	100	0	1	0	0	
CMTW CH0	CMTW_CH0_cmt2_cmp_pls_n	Arbitrary	Arbitrary	064h	1/0	000	0	1	0	1/0	
CMTW CH1	CMTW_CH1_cmt2_cmp_pls_n	Arbitrary	Arbitrary	065h	1/0	000	0	1	0	1/0	
CMTW CH2	CMTW_CH2_cmt2_cmp_pls_n	Arbitrary	Arbitrary	066h	1/0	000	0	1	0	1/0	
CMTW CH3	CMTW_CH3_cmt2_cmp_pls_n	Arbitrary	Arbitrary	067h	1/0	000	0	1	0	1/0	
CMTW CH4	CMTW_CH4_cmt2_cmp_pls_n	Arbitrary	Arbitrary	068h	1/0	000	0	1	0	1/0	
CMTW CH5	CMTW_CH5_cmt2_cmp_pls_n	Arbitrary	Arbitrary	069h	1/0	000	0	1	0	1/0	
CMTW CH6	CMTW_CH6_cmt2_cmp_pls_n	Arbitrary	Arbitrary	06Ah	1/0	000	0	1	0	1/0	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (5/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
CMTW CH7	CMTW_CH7_cmt2_cmp_pls_n	Arbitrary	Arbitrary	06Bh	1/0	000	0	1	0	1/0	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
CMTW CH0	CMTW_CH0_cmt2_ic0_pls_n	Arbitrary	Arbitrary	06Ch	1/0	000	0	1	0	1/0	
CMTW CH1	CMTW_CH1_cmt2_ic0_pls_n	Arbitrary	Arbitrary	06Dh	1/0	000	0	1	0	1/0	
CMTW CH2	CMTW_CH2_cmt2_ic0_pls_n	Arbitrary	Arbitrary	06Eh	1/0	000	0	1	0	1/0	
CMTW CH3	CMTW_CH3_cmt2_ic0_pls_n	Arbitrary	Arbitrary	06Fh	1/0	000	0	1	0	1/0	
CMTW CH0	CMTW_CH0_cmt2_ic1_pls_n	Arbitrary	Arbitrary	074h	1/0	000	0	1	0	1/0	
CMTW CH1	CMTW_CH1_cmt2_ic1_pls_n	Arbitrary	Arbitrary	075h	1/0	000	0	1	0	1/0	
CMTW CH2	CMTW_CH2_cmt2_ic1_pls_n	Arbitrary	Arbitrary	076h	1/0	000	0	1	0	1/0	
CMTW CH3	CMTW_CH3_cmt2_ic1_pls_n	Arbitrary	Arbitrary	077h	1/0	000	0	1	0	1/0	
CMTW CH0	CMTW_CH0_cmt2_oc0_pls_n	Arbitrary	Arbitrary	07Ch	1/0	000	0	1	0	1/0	
CMTW CH1	CMTW_CH1_cmt2_oc0_pls_n	Arbitrary	Arbitrary	07Dh	1/0	000	0	1	0	1/0	
CMTW CH2	CMTW_CH2_cmt2_oc0_pls_n	Arbitrary	Arbitrary	07Eh	1/0	000	0	1	0	1/0	
CMTW CH3	CMTW_CH3_cmt2_oc0_pls_n	Arbitrary	Arbitrary	07Fh	1/0	000	0	1	0	1/0	
CMTW CH0	CMTW_CH0_cmt2_oc1_pls_n	Arbitrary	Arbitrary	084h	1/0	000	0	1	0	1/0	
CMTW CH1	CMTW_CH1_cmt2_oc1_pls_n	Arbitrary	Arbitrary	085h	1/0	000	0	1	0	1/0	
CMTW CH2	CMTW_CH2_cmt2_oc1_pls_n	Arbitrary	Arbitrary	086h	1/0	000	0	1	0	1/0	
CMTW CH3	CMTW_CH3_cmt2_oc1_pls_n	Arbitrary	Arbitrary	087h	1/0	000	0	1	0	1/0	
RSPI CH0	RSPI_CH0_sp_rxintpls_n	SPDR	Arbitrary	08Ch	0	100	0	1	0	0	
RSPI CH0	RSPI_CH0_sp_txintpls_n	Arbitrary	SPDR	08Dh	0	100	0	1	0	1	
RSPI CH1	RSPI_CH1_sp_rxintpls_n	SPDR	Arbitrary	08Eh	0	100	0	1	0	0	
RSPI CH1	RSPI_CH1_sp_txintpls_n	Arbitrary	SPDR	08Fh	0	100	0	1	0	1	
RSPI CH2	RSPI_CH2_sp_rxintpls_n	SPDR	Arbitrary	090h	0	100	0	1	0	0	
RSPI CH2	RSPI_CH2_sp_txintpls_n	Arbitrary	SPDR	091h	0	100	0	1	0	1	
RSCI CH0	RSCI_CH0_INT_sc_rxi_n	RDR	Arbitrary	092h	0	100	0	1	0	0	
RSCI CH0	RSCI_CH0_INT_sc_txi_n	Arbitrary	TDR	093h	0	100	0	1	0	1	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (6/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
RSCI CH0	RSCI_CH0_INT_sc_aed_n	Arbitrary	Arbitrary	094h	0	100	0	1	0	0	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
RSCI CH1	RSCI_CH1_INT_sc_rxi_n	RDR	Arbitrary	095h	0	100	0	1	0	0	
RSCI CH1	RSCI_CH1_INT_sc_txi_n	Arbitrary	TDR	096h	0	100	0	1	0	1	
RSCI CH1	RSCI_CH1_INT_sc_aed_n	Arbitrary	Arbitrary	097h	0	100	0	1	0	0	
RSCI CH2	RSCI_CH2_INT_sc_rxi_n	RDR	Arbitrary	098h	0	100	0	1	0	0	
RSCI CH2	RSCI_CH2_INT_sc_txi_n	Arbitrary	TDR	099h	0	100	0	1	0	1	
RSCI CH2	RSCI_CH2_INT_sc_aed_n	Arbitrary	Arbitrary	09Ah	0	100	0	1	0	0	
RSCI CH3	RSCI_CH3_INT_sc_rxi_n	RDR	Arbitrary	09Bh	0	100	0	1	0	0	
RSCI CH3	RSCI_CH3_INT_sc_txi_n	Arbitrary	TDR	09Ch	0	100	0	1	0	1	
RSCI CH3	RSCI_CH3_INT_sc_aed_n	Arbitrary	Arbitrary	09Dh	0	100	0	1	0	0	
RSCI CH4	RSCI_CH4_INT_sc_rxi_n	RDR	Arbitrary	09Eh	0	100	0	1	0	0	
RSCI CH4	RSCI_CH4_INT_sc_txi_n	Arbitrary	TDR	09Fh	0	100	0	1	0	1	
RSCI CH4	RSCI_CH4_INT_sc_aed_n	Arbitrary	Arbitrary	0A0h	0	100	0	1	0	0	
RSCI CH5	RSCI_CH5_INT_sc_rxi_n	RDR	Arbitrary	0A1h	0	100	0	1	0	0	
RSCI CH5	RSCI_CH5_INT_sc_txi_n	Arbitrary	TDR	0A2h	0	100	0	1	0	1	
RSCI CH5	RSCI_CH5_INT_sc_aed_n	Arbitrary	Arbitrary	0A3h	0	100	0	1	0	0	
RSCI CH6	RSCI_CH6_INT_sc_rxi_n	RDR	Arbitrary	0A4h	0	100	0	1	0	0	
RSCI CH6	RSCI_CH6_INT_sc_txi_n	Arbitrary	TDR	0A5h	0	100	0	1	0	1	
RSCI CH6	RSCI_CH6_INT_sc_aed_n	Arbitrary	Arbitrary	0A6h	0	100	0	1	0	0	
RSCI CH7	RSCI_CH7_INT_sc_rxi_n	RDR	Arbitrary	0A7h	0	100	0	1	0	0	
RSCI CH7	RSCI_CH7_INT_sc_txi_n	Arbitrary	TDR	0A8h	0	100	0	1	0	1	
RSCI CH7	RSCI_CH7_INT_sc_aed_n	Arbitrary	Arbitrary	0A9h	0	100	0	1	0	0	
RSCI CH8	RSCI_CH8_INT_sc_rxi_n	RDR	Arbitrary	0AAh	0	100	0	1	0	0	
RSCI CH8	RSCI_CH8_INT_sc_txi_n	Arbitrary	TDR	0ABh	0	100	0	1	0	1	
RSCI CH8	RSCI_CH8_INT_sc_aed_n	Arbitrary	Arbitrary	0ACh	0	100	0	1	0	0	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (7/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
RSCI CH9	RSCI_CH9_INT_sc_rxi_n	RDR	Arbitrary	0ADh	0	100	0	1	0	0	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
RSCI CH9	RSCI_CH9_INT_sc_txi_n	Arbitrary	TDR	0AEh	0	100	0	1	0	1	
RSCI CH9	RSCI_CH9_INT_sc_aed_n	Arbitrary	Arbitrary	0AFh	0	100	0	1	0	0	
RIIC CH0	RIIC_CH0_ti_n	Arbitrary	ICDRT	0B0h	0	100	0	1	0	1	
RIIC CH0	RIIC_CH0_ri_n	ICDRR	Arbitrary	0B1h	0	100	0	1	0	0	
RIIC CH1	RIIC_CH1_ti_n	Arbitrary	ICDRT	0B2h	0	100	0	1	0	1	
RIIC CH1	RIIC_CH1_ri_n	ICDRR	Arbitrary	0B3h	0	100	0	1	0	0	
RIIC CH2	RIIC_CH2_ti_n	Arbitrary	ICDRT	0B4h	0	100	0	1	0	1	
RIIC CH2	RIIC_CH2_ri_n	ICDRR	Arbitrary	0B5h	0	100	0	1	0	0	
RIIC CH3	RIIC_CH3_ti_n	Arbitrary	ICDRT	0B6h	0	100	0	1	0	1	
RIIC CH3	RIIC_CH3_ri_n	ICDRR	Arbitrary	0B7h	0	100	0	1	0	0	
RIIC CH4	RIIC_CH4_ti_n	Arbitrary	ICDRT	0B8h	0	100	0	1	0	1	
RIIC CH4	RIIC_CH4_ri_n	ICDRR	Arbitrary	0B9h	0	100	0	1	0	0	
RIIC CH5	RIIC_CH5_ti_n	Arbitrary	ICDRT	0BAh	0	100	0	1	0	1	
RIIC CH5	RIIC_CH5_ri_n	ICDRR	Arbitrary	0BBh	0	100	0	1	0	0	
RIIC CH6	RIIC_CH6_ti_n	Arbitrary	ICDRT	0BCh	0	100	0	1	0	1	
RIIC CH6	RIIC_CH6_ri_n	ICDRR	Arbitrary	0BDh	0	100	0	1	0	0	
RIIC CH7	RIIC_CH7_ti_n	Arbitrary	ICDRT	0BEh	0	100	0	1	0	1	
RIIC CH7	RIIC_CH7_ri_n	ICDRR	Arbitrary	0BFh	0	100	0	1	0	0	
RIIC CH8	RIIC_CH8_ti_n	Arbitrary	ICDRT	0C0h	0	100	0	1	0	1	
RIIC CH8	RIIC_CH8_ri_n	ICDRR	Arbitrary	0C1h	0	100	0	1	0	0	
PDM0 CH0	INT_PDM_DAT0	PDDRRCH0	Arbitrary	0C2h	0	010	1	1	0	0	
PDM0 CH1	INT_PDM_DAT1	PDDRRCH1	Arbitrary	0C3h	0	010	1	1	0	0	
PDM0 CH2	INT_PDM_DAT2	PDDRRCH2	Arbitrary	0C4h	0	010	1	1	0	0	
PDM1 CH0	INT_PDM_DAT0	PDDRRCH0	Arbitrary	0C5h	0	010	1	1	0	0	
PDM1 CH1	INT_PDM_DAT1	PDDRRCH1	Arbitrary	0C6h	0	010	1	1	0	0	
PDM1 CH2	INT_PDM_DAT2	PDDRRCH2	Arbitrary	0C7h	0	010	1	1	0	0	
ADC0	ada_adireq_n	ADDRn	Arbitrary	0C8h	0/1	010	0	1	0	0/1	
ADC0	ada_gbadireq_n	ADDRn	Arbitrary	0C9h	0/1	010	0	1	0	0/1	
ADC0	ada_gcadireq_n	ADDRn	Arbitrary	0CAh	0/1	010	0	1	0	0/1	
ADC0	ada_elccondmtch	ADDRn	Arbitrary	0CBh	0/1	010	0	1	0	0/1	
ADC0	ada_elccondunmtch	ADDRn	Arbitrary	0CCh	0/1	010	0	1	0	0/1	
TSU	S12TSUADI0	SCRR	Arbitrary	0CDh	0	100	0	1	0	0	
TSU	S12TSUADI1	SCRR	Arbitrary	0CEh	0	100	0	1	0	0	
SCIF	ub1_rxi_n	FRDR	Arbitrary	0CFh	0	100	1	1	0	0	
SCIF	ub1_txi_n	Arbitrary	FTDR	0D0h	0	100	1	1	0	1	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (8/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
GPT0 CH0	GPT_U0_gpt_gtcia_n_0	Arbitrary	Arbitrary	0D1h	0/1	000	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
GPT0 CH1	GPT_U0_gpt_gtcia_n_1	Arbitrary	Arbitrary	0D2h	0/1	000	0	1	0	0/1	
GPT0 CH2	GPT_U0_gpt_gtcia_n_2	Arbitrary	Arbitrary	0D3h	0/1	000	0	1	0	0/1	
GPT0 CH3	GPT_U0_gpt_gtcia_n_3	Arbitrary	Arbitrary	0D4h	0/1	000	0	1	0	0/1	
GPT0 CH4	GPT_U0_gpt_gtcia_n_4	Arbitrary	Arbitrary	0D5h	0/1	000	0	1	0	0/1	
GPT0 CH5	GPT_U0_gpt_gtcia_n_5	Arbitrary	Arbitrary	0D6h	0/1	000	0	1	0	0/1	
GPT0 CH6	GPT_U0_gpt_gtcia_n_6	Arbitrary	Arbitrary	0D7h	0/1	000	0	1	0	0/1	
GPT0 CH7	GPT_U0_gpt_gtcia_n_7	Arbitrary	Arbitrary	0D8h	0/1	000	0	1	0	0/1	
GPT0 CH0	GPT_U0_gpt_gtcib_n_0	Arbitrary	Arbitrary	0D9h	0/1	000	0	1	0	0/1	
GPT0 CH1	GPT_U0_gpt_gtcib_n_1	Arbitrary	Arbitrary	0DAh	0/1	000	0	1	0	0/1	
GPT0 CH2	GPT_U0_gpt_gtcib_n_2	Arbitrary	Arbitrary	0DBh	0/1	000	0	1	0	0/1	
GPT0 CH3	GPT_U0_gpt_gtcib_n_3	Arbitrary	Arbitrary	0DCh	0/1	000	0	1	0	0/1	
GPT0 CH4	GPT_U0_gpt_gtcib_n_4	Arbitrary	Arbitrary	0DDh	0/1	000	0	1	0	0/1	
GPT0 CH5	GPT_U0_gpt_gtcib_n_5	Arbitrary	Arbitrary	0DEh	0/1	000	0	1	0	0/1	
GPT0 CH6	GPT_U0_gpt_gtcib_n_6	Arbitrary	Arbitrary	0DFh	0/1	000	0	1	0	0/1	
GPT0 CH7	GPT_U0_gpt_gtcib_n_7	Arbitrary	Arbitrary	0E0h	0/1	000	0	1	0	0/1	
GPT0 CH0	GPT_U0_gpt_gtcic_n_0	Arbitrary	Arbitrary	0E1h	0/1	000	0	1	0	0/1	
GPT0 CH1	GPT_U0_gpt_gtcic_n_1	Arbitrary	Arbitrary	0E2h	0/1	000	0	1	0	0/1	
GPT0 CH2	GPT_U0_gpt_gtcic_n_2	Arbitrary	Arbitrary	0E3h	0/1	000	0	1	0	0/1	
GPT0 CH3	GPT_U0_gpt_gtcic_n_3	Arbitrary	Arbitrary	0E4h	0/1	000	0	1	0	0/1	
GPT0 CH4	GPT_U0_gpt_gtcic_n_4	Arbitrary	Arbitrary	0E5h	0/1	000	0	1	0	0/1	
GPT0 CH5	GPT_U0_gpt_gtcic_n_5	Arbitrary	Arbitrary	0E6h	0/1	000	0	1	0	0/1	
GPT0 CH6	GPT_U0_gpt_gtcic_n_6	Arbitrary	Arbitrary	0E7h	0/1	000	0	1	0	0/1	
GPT0 CH7	GPT_U0_gpt_gtcic_n_7	Arbitrary	Arbitrary	0E8h	0/1	000	0	1	0	0/1	
GPT0 CH0	GPT_U0_gpt_gtcid_n_0	Arbitrary	Arbitrary	0E9h	0/1	000	0	1	0	0/1	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (9/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
GPT0 CH1	GPT_U0_gpt_gtcid_n_1	Arbitrary	Arbitrary	0EAh	0/1	000	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
GPT0 CH2	GPT_U0_gpt_gtcid_n_2	Arbitrary	Arbitrary	0EBh	0/1	000	0	1	0	0/1	
GPT0 CH3	GPT_U0_gpt_gtcid_n_3	Arbitrary	Arbitrary	0ECh	0/1	000	0	1	0	0/1	
GPT0 CH4	GPT_U0_gpt_gtcid_n_4	Arbitrary	Arbitrary	0EDh	0/1	000	0	1	0	0/1	
GPT0 CH5	GPT_U0_gpt_gtcid_n_5	Arbitrary	Arbitrary	0EEh	0/1	000	0	1	0	0/1	
GPT0 CH6	GPT_U0_gpt_gtcid_n_6	Arbitrary	Arbitrary	0EFh	0/1	000	0	1	0	0/1	
GPT0 CH7	GPT_U0_gpt_gtcid_n_7	Arbitrary	Arbitrary	0F0h	0/1	000	0	1	0	0/1	
GPT0 CH0	GPT_U0_gpt_gtcie_n_0	Arbitrary	Arbitrary	0F1h	0/1	000	0	1	0	0/1	
GPT0 CH1	GPT_U0_gpt_gtcie_n_1	Arbitrary	Arbitrary	0F2h	0/1	000	0	1	0	0/1	
GPT0 CH2	GPT_U0_gpt_gtcie_n_2	Arbitrary	Arbitrary	0F3h	0/1	000	0	1	0	0/1	
GPT0 CH3	GPT_U0_gpt_gtcie_n_3	Arbitrary	Arbitrary	0F4h	0/1	000	0	1	0	0/1	
GPT0 CH4	GPT_U0_gpt_gtcie_n_4	Arbitrary	Arbitrary	0F5h	0/1	000	0	1	0	0/1	
GPT0 CH5	GPT_U0_gpt_gtcie_n_5	Arbitrary	Arbitrary	0F6h	0/1	000	0	1	0	0/1	
GPT0 CH6	GPT_U0_gpt_gtcie_n_6	Arbitrary	Arbitrary	0F7h	0/1	000	0	1	0	0/1	
GPT0 CH7	GPT_U0_gpt_gtcie_n_7	Arbitrary	Arbitrary	0F8h	0/1	000	0	1	0	0/1	
GPT0 CH0	GPT_U0_gpt_gtcif_n_0	Arbitrary	Arbitrary	0F9h	0/1	000	0	1	0	0/1	
GPT0 CH1	GPT_U0_gpt_gtcif_n_1	Arbitrary	Arbitrary	0FAh	0/1	000	0	1	0	0/1	
GPT0 CH2	GPT_U0_gpt_gtcif_n_2	Arbitrary	Arbitrary	0FBh	0/1	000	0	1	0	0/1	
GPT0 CH3	GPT_U0_gpt_gtcif_n_3	Arbitrary	Arbitrary	0FCh	0/1	000	0	1	0	0/1	
GPT0 CH4	GPT_U0_gpt_gtcif_n_4	Arbitrary	Arbitrary	0FDh	0/1	000	0	1	0	0/1	
GPT0 CH5	GPT_U0_gpt_gtcif_n_5	Arbitrary	Arbitrary	0FEh	0/1	000	0	1	0	0/1	
GPT0 CH6	GPT_U0_gpt_gtcif_n_6	Arbitrary	Arbitrary	0FFh	0/1	000	0	1	0	0/1	
GPT0 CH7	GPT_U0_gpt_gtcif_n_7	Arbitrary	Arbitrary	100h	0/1	000	0	1	0	0/1	
GPT0 CH0	GPT_U0_gpt_gtciv_n_0	Arbitrary	Arbitrary	101h	0/1	000	0	1	0	0/1	
GPT0 CH1	GPT_U0_gpt_gtciv_n_1	Arbitrary	Arbitrary	102h	0/1	000	0	1	0	0/1	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (10/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
GPT0 CH2	GPT_U0_gpt_gtciv_n_2	Arbitrary	Arbitrary	103h	0/1	000	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
GPT0 CH3	GPT_U0_gpt_gtciv_n_3	Arbitrary	Arbitrary	104h	0/1	000	0	1	0	0/1	
GPT0 CH4	GPT_U0_gpt_gtciv_n_4	Arbitrary	Arbitrary	105h	0/1	000	0	1	0	0/1	
GPT0 CH5	GPT_U0_gpt_gtciv_n_5	Arbitrary	Arbitrary	106h	0/1	000	0	1	0	0/1	
GPT0 CH6	GPT_U0_gpt_gtciv_n_6	Arbitrary	Arbitrary	107h	0/1	000	0	1	0	0/1	
GPT0 CH7	GPT_U0_gpt_gtciv_n_7	Arbitrary	Arbitrary	108h	0/1	000	0	1	0	0/1	
GPT0 CH0	GPT_U0_gpt_gtciu_n_0	Arbitrary	Arbitrary	109h	0/1	000	0	1	0	0/1	
GPT0 CH1	GPT_U0_gpt_gtciu_n_1	Arbitrary	Arbitrary	10Ah	0/1	000	0	1	0	0/1	
GPT0 CH2	GPT_U0_gpt_gtciu_n_2	Arbitrary	Arbitrary	10Bh	0/1	000	0	1	0	0/1	
GPT0 CH3	GPT_U0_gpt_gtciu_n_3	Arbitrary	Arbitrary	10Ch	0/1	000	0	1	0	0/1	
GPT0 CH4	GPT_U0_gpt_gtciu_n_4	Arbitrary	Arbitrary	10Dh	0/1	000	0	1	0	0/1	
GPT0 CH5	GPT_U0_gpt_gtciu_n_5	Arbitrary	Arbitrary	10Eh	0/1	000	0	1	0	0/1	
GPT0 CH6	GPT_U0_gpt_gtciu_n_6	Arbitrary	Arbitrary	10Fh	0/1	000	0	1	0	0/1	
GPT0 CH7	GPT_U0_gpt_gtciu_n_7	Arbitrary	Arbitrary	110h	0/1	000	0	1	0	0/1	
GPT1 CH0	GPT_U1_gpt_gtcia_n_0	Arbitrary	Arbitrary	111h	0/1	000	0	1	0	0/1	
GPT1 CH1	GPT_U1_gpt_gtcia_n_1	Arbitrary	Arbitrary	112h	0/1	000	0	1	0	0/1	
GPT1 CH2	GPT_U1_gpt_gtcia_n_2	Arbitrary	Arbitrary	113h	0/1	000	0	1	0	0/1	
GPT1 CH3	GPT_U1_gpt_gtcia_n_3	Arbitrary	Arbitrary	114h	0/1	000	0	1	0	0/1	
GPT1 CH4	GPT_U1_gpt_gtcia_n_4	Arbitrary	Arbitrary	115h	0/1	000	0	1	0	0/1	
GPT1 CH5	GPT_U1_gpt_gtcia_n_5	Arbitrary	Arbitrary	116h	0/1	000	0	1	0	0/1	
GPT1 CH6	GPT_U1_gpt_gtcia_n_6	Arbitrary	Arbitrary	117h	0/1	000	0	1	0	0/1	
GPT1 CH7	GPT_U1_gpt_gtcia_n_7	Arbitrary	Arbitrary	118h	0/1	000	0	1	0	0/1	
GPT1 CH0	GPT_U1_gpt_gtcib_n_0	Arbitrary	Arbitrary	119h	0/1	000	0	1	0	0/1	
GPT1 CH1	GPT_U1_gpt_gtcib_n_1	Arbitrary	Arbitrary	11Ah	0/1	000	0	1	0	0/1	
GPT1 CH2	GPT_U1_gpt_gtcib_n_2	Arbitrary	Arbitrary	11Bh	0/1	000	0	1	0	0/1	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (11/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
GPT1 CH3	GPT_U1_gpt_gtcib_n_3	Arbitrary	Arbitrary	11Ch	0/1	000	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
GPT1 CH4	GPT_U1_gpt_gtcib_n_4	Arbitrary	Arbitrary	11Dh	0/1	000	0	1	0	0/1	
GPT1 CH5	GPT_U1_gpt_gtcib_n_5	Arbitrary	Arbitrary	11Eh	0/1	000	0	1	0	0/1	
GPT1 CH6	GPT_U1_gpt_gtcib_n_6	Arbitrary	Arbitrary	11Fh	0/1	000	0	1	0	0/1	
GPT1 CH7	GPT_U1_gpt_gtcib_n_7	Arbitrary	Arbitrary	120h	0/1	000	0	1	0	0/1	
GPT1 CH0	GPT_U1_gpt_gtcic_n_0	Arbitrary	Arbitrary	121h	0/1	000	0	1	0	0/1	
GPT1 CH1	GPT_U1_gpt_gtcic_n_1	Arbitrary	Arbitrary	122h	0/1	000	0	1	0	0/1	
GPT1 CH2	GPT_U1_gpt_gtcic_n_2	Arbitrary	Arbitrary	123h	0/1	000	0	1	0	0/1	
GPT1 CH3	GPT_U1_gpt_gtcic_n_3	Arbitrary	Arbitrary	124h	0/1	000	0	1	0	0/1	
GPT1 CH4	GPT_U1_gpt_gtcic_n_4	Arbitrary	Arbitrary	125h	0/1	000	0	1	0	0/1	
GPT1 CH5	GPT_U1_gpt_gtcic_n_5	Arbitrary	Arbitrary	126h	0/1	000	0	1	0	0/1	
GPT1 CH6	GPT_U1_gpt_gtcic_n_6	Arbitrary	Arbitrary	127h	0/1	000	0	1	0	0/1	
GPT1 CH7	GPT_U1_gpt_gtcic_n_7	Arbitrary	Arbitrary	128h	0/1	000	0	1	0	0/1	
GPT1 CH0	GPT_U1_gpt_gtcid_n_0	Arbitrary	Arbitrary	129h	0/1	000	0	1	0	0/1	
GPT1 CH1	GPT_U1_gpt_gtcid_n_1	Arbitrary	Arbitrary	12Ah	0/1	000	0	1	0	0/1	
GPT1 CH2	GPT_U1_gpt_gtcid_n_2	Arbitrary	Arbitrary	12Bh	0/1	000	0	1	0	0/1	
GPT1 CH3	GPT_U1_gpt_gtcid_n_3	Arbitrary	Arbitrary	12Ch	0/1	000	0	1	0	0/1	
GPT1 CH4	GPT_U1_gpt_gtcid_n_4	Arbitrary	Arbitrary	12Dh	0/1	000	0	1	0	0/1	
GPT1 CH5	GPT_U1_gpt_gtcid_n_5	Arbitrary	Arbitrary	12Eh	0/1	000	0	1	0	0/1	
GPT1 CH6	GPT_U1_gpt_gtcid_n_6	Arbitrary	Arbitrary	12Fh	0/1	000	0	1	0	0/1	
GPT1 CH7	GPT_U1_gpt_gtcid_n_7	Arbitrary	Arbitrary	130h	0/1	000	0	1	0	0/1	
GPT1 CH0	GPT_U1_gpt_gtcie_n_0	Arbitrary	Arbitrary	131h	0/1	000	0	1	0	0/1	
GPT1 CH1	GPT_U1_gpt_gtcie_n_1	Arbitrary	Arbitrary	132h	0/1	000	0	1	0	0/1	
GPT1 CH2	GPT_U1_gpt_gtcie_n_2	Arbitrary	Arbitrary	133h	0/1	000	0	1	0	0/1	
GPT1 CH3	GPT_U1_gpt_gtcie_n_3	Arbitrary	Arbitrary	134h	0/1	000	0	1	0	0/1	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (12/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
GPT1 CH4	GPT_U1_gpt_gtcie_n_4	Arbitrary	Arbitrary	135h	0/1	000	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
GPT1 CH5	GPT_U1_gpt_gtcie_n_5	Arbitrary	Arbitrary	136h	0/1	000	0	1	0	0/1	
GPT1 CH6	GPT_U1_gpt_gtcie_n_6	Arbitrary	Arbitrary	137h	0/1	000	0	1	0	0/1	
GPT1 CH7	GPT_U1_gpt_gtcie_n_7	Arbitrary	Arbitrary	138h	0/1	000	0	1	0	0/1	
GPT1 CH0	GPT_U1_gpt_gtcif_n_0	Arbitrary	Arbitrary	139h	0/1	000	0	1	0	0/1	
GPT1 CH1	GPT_U1_gpt_gtcif_n_1	Arbitrary	Arbitrary	13Ah	0/1	000	0	1	0	0/1	
GPT1 CH2	GPT_U1_gpt_gtcif_n_2	Arbitrary	Arbitrary	13Bh	0/1	000	0	1	0	0/1	
GPT1 CH3	GPT_U1_gpt_gtcif_n_3	Arbitrary	Arbitrary	13Ch	0/1	000	0	1	0	0/1	
GPT1 CH4	GPT_U1_gpt_gtcif_n_4	Arbitrary	Arbitrary	13Dh	0/1	000	0	1	0	0/1	
GPT1 CH5	GPT_U1_gpt_gtcif_n_5	Arbitrary	Arbitrary	13Eh	0/1	000	0	1	0	0/1	
GPT1 CH6	GPT_U1_gpt_gtcif_n_6	Arbitrary	Arbitrary	13Fh	0/1	000	0	1	0	0/1	
GPT1 CH7	GPT_U1_gpt_gtcif_n_7	Arbitrary	Arbitrary	140h	0/1	000	0	1	0	0/1	
GPT1 CH0	GPT_U1_gpt_gtciv_n_0	Arbitrary	Arbitrary	141h	0/1	000	0	1	0	0/1	
GPT1 CH1	GPT_U1_gpt_gtciv_n_1	Arbitrary	Arbitrary	142h	0/1	000	0	1	0	0/1	
GPT1 CH2	GPT_U1_gpt_gtciv_n_2	Arbitrary	Arbitrary	143h	0/1	000	0	1	0	0/1	
GPT1 CH3	GPT_U1_gpt_gtciv_n_3	Arbitrary	Arbitrary	144h	0/1	000	0	1	0	0/1	
GPT1 CH4	GPT_U1_gpt_gtciv_n_4	Arbitrary	Arbitrary	145h	0/1	000	0	1	0	0/1	
GPT1 CH5	GPT_U1_gpt_gtciv_n_5	Arbitrary	Arbitrary	146h	0/1	000	0	1	0	0/1	
GPT1 CH6	GPT_U1_gpt_gtciv_n_6	Arbitrary	Arbitrary	147h	0/1	000	0	1	0	0/1	
GPT1 CH7	GPT_U1_gpt_gtciv_n_7	Arbitrary	Arbitrary	148h	0/1	000	0	1	0	0/1	
GPT1 CH0	GPT_U1_gpt_gtciu_n_0	Arbitrary	Arbitrary	149h	0/1	000	0	1	0	0/1	
GPT1 CH1	GPT_U1_gpt_gtciu_n_1	Arbitrary	Arbitrary	14Ah	0/1	000	0	1	0	0/1	
GPT1 CH2	GPT_U1_gpt_gtciu_n_2	Arbitrary	Arbitrary	14Bh	0/1	000	0	1	0	0/1	
GPT1 CH3	GPT_U1_gpt_gtciu_n_3	Arbitrary	Arbitrary	14Ch	0/1	000	0	1	0	0/1	
GPT1 CH4	GPT_U1_gpt_gtciu_n_4	Arbitrary	Arbitrary	14Dh	0/1	000	0	1	0	0/1	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (13/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
GPT1 CH5	GPT_U1_gpt_gtciu_n_5	Arbitrary	Arbitrary	14Eh	0/1	000	0	1	0	0/1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
GPT1 CH6	GPT_U1_gpt_gtciu_n_6	Arbitrary	Arbitrary	14Fh	0/1	000	0	1	0	0/1	
GPT1 CH7	GPT_U1_gpt_gtciu_n_7	Arbitrary	Arbitrary	150h	0/1	000	0	1	0	0/1	
I3C	NT_ri3c_rx_n	NTDTBP0	Arbitrary	151h	0	100	0	1	0	0	
I3C	INT_ri3c_tx_n	Arbitrary	NTDTBP0	152h	0	100	0	1	0	1	
CANFD	can_rf_dmareq_0	CFDRFDF00	Arbitrary	153h	0/1	100	0	1	0	0	
CANFD	can_rf_dmareq_1	CFDRFDF01	Arbitrary	154h	0/1	100	0	1	0	0	
CANFD	can_rf_dmareq_2	CFDRFDF02	Arbitrary	155h	0/1	100	0	1	0	0	
CANFD	can_rf_dmareq_3	CFDRFDF03	Arbitrary	156h	0/1	100	0	1	0	0	
CANFD	can_rf_dmareq_4	CFDRFDF04	Arbitrary	157h	0/1	100	0	1	0	0	
CANFD	can_rf_dmareq_5	CFDRFDF05	Arbitrary	158h	0/1	100	0	1	0	0	
CANFD	can_rf_dmareq_6	CFDRFDF06	Arbitrary	159h	0/1	100	0	1	0	0	
CANFD	can_rf_dmareq_7	CFDRFDF07	Arbitrary	15Ah	0/1	100	0	1	0	0	
CANFD	can_cf_dmareq_0	CFDCFDF00	Arbitrary	15Bh	0/1	100	0	1	0	0	
CANFD	can_cf_dmareq_1	CFDCFDF03	Arbitrary	15Ch	0/1	100	0	1	0	0	
CANFD	can_cf_dmareq_2	CFDCFDF06	Arbitrary	15Dh	0/1	100	0	1	0	0	
CANFD	can_cf_dmareq_3	CFDCFDF09	Arbitrary	15Eh	0/1	100	0	1	0	0	
CANFD	can_cf_dmareq_4	CFDCFDF12	Arbitrary	15Fh	0/1	100	0	1	0	0	
CANFD	can_cf_dmareq_5	CFDCFDF15	Arbitrary	160h	0/1	100	0	1	0	0	
SSIU	ssip00_dreq_rx	BUSIF	Arbitrary	161h	0	001	1	1	0	0	
SSIU	ssip00_dreq_tx	Arbitrary	BUSIF	162h	0	001	1	1	0	1	
SSIU	ssip01_dreq_rx	BUSIF	Arbitrary	163h	0	001	1	1	0	0	
SSIU	ssip01_dreq_tx	Arbitrary	BUSIF	164h	0	001	1	1	0	1	
SSIU	ssip02_dreq_rx	BUSIF	Arbitrary	165h	0	001	1	1	0	0	
SSIU	ssip02_dreq_tx	Arbitrary	BUSIF	166h	0	001	1	1	0	1	
SSIU	ssip03_dreq_rx	BUSIF	Arbitrary	167h	0	001	1	1	0	0	
SSIU	ssip03_dreq_tx	Arbitrary	BUSIF	168h	0	001	1	1	0	1	
SSIU	ssip10_dreq_rx	BUSIF	Arbitrary	169h	0	001	1	1	0	0	
SSIU	ssip10_dreq_tx	Arbitrary	BUSIF	16Ah	0	001	1	1	0	1	
SSIU	ssip11_dreq_rx	BUSIF	Arbitrary	16Bh	0	001	1	1	0	0	
SSIU	ssip11_dreq_tx	Arbitrary	BUSIF	16Ch	0	001	1	1	0	1	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (14/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
SSIU	ssip12_dreq_rx	BUSIF	Arbitrary	16Dh	0	001	1	1	0	0	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
SSIU	ssip12_dreq_tx	Arbitrary	BUSIF	16Eh	0	001	1	1	0	1	
SSIU	ssip13_dreq_rx	BUSIF	Arbitrary	16Fh	0	001	1	1	0	0	
SSIU	ssip13_dreq_tx	Arbitrary	BUSIF	170h	0	001	1	1	0	1	
SSIU	ssip20_dreq_rx	BUSIF	Arbitrary	171h	0	001	1	1	0	0	
SSIU	ssip20_dreq_tx	Arbitrary	BUSIF	172h	0	001	1	1	0	1	
SSIU	ssip21_dreq_rx	BUSIF	Arbitrary	173h	0	001	1	1	0	0	
SSIU	ssip21_dreq_tx	Arbitrary	BUSIF	174h	0	001	1	1	0	1	
SSIU	ssip22_dreq_rx	BUSIF	Arbitrary	175h	0	001	1	1	0	0	
SSIU	ssip22_dreq_tx	Arbitrary	BUSIF	176h	0	001	1	1	0	1	
SSIU	ssip23_dreq_rx	BUSIF	Arbitrary	177h	0	001	1	1	0	0	
SSIU	ssip23_dreq_tx	Arbitrary	BUSIF	178h	0	001	1	1	0	1	
SSIU	ssip30_dreq_rx	BUSIF	Arbitrary	179h	0	001	1	1	0	0	
SSIU	ssip30_dreq_tx	Arbitrary	BUSIF	17Ah	0	001	1	1	0	1	
SSIU	ssip31_dreq_rx	BUSIF	Arbitrary	17Bh	0	001	1	1	0	0	
SSIU	ssip31_dreq_tx	Arbitrary	BUSIF	17Ch	0	001	1	1	0	1	
SSIU	ssip32_dreq_rx	BUSIF	Arbitrary	17Dh	0	001	1	1	0	0	
SSIU	ssip32_dreq_tx	Arbitrary	BUSIF	17Eh	0	001	1	1	0	1	
SSIU	ssip33_dreq_rx	BUSIF	Arbitrary	17Fh	0	001	1	1	0	0	
SSIU	ssip33_dreq_tx	Arbitrary	BUSIF	180h	0	001	1	1	0	1	
SSIU	ssip40_dreq_rx	BUSIF	Arbitrary	181h	0	001	1	1	0	0	
SSIU	ssip40_dreq_tx	Arbitrary	BUSIF	182h	0	001	1	1	0	1	
SSIU	ssip41_dreq_rx	BUSIF	Arbitrary	183h	0	001	1	1	0	0	
SSIU	ssip41_dreq_tx	Arbitrary	BUSIF	184h	0	001	1	1	0	1	
SSIU	ssip42_dreq_rx	BUSIF	Arbitrary	185h	0	001	1	1	0	0	
SSIU	ssip42_dreq_tx	Arbitrary	BUSIF	186h	0	001	1	1	0	1	
SSIU	ssip43_dreq_rx	BUSIF	Arbitrary	187h	0	001	1	1	0	0	
SSIU	ssip43_dreq_tx	Arbitrary	BUSIF	188h	0	001	1	1	0	1	
SSIU	ssip5_dreq_rx	BUSIF	Arbitrary	189h	0	001	1	1	0	0	
SSIU	ssip5_dreq_tx	Arbitrary	BUSIF	18Ah	0	001	1	1	0	1	
SSIU	ssip6_dreq_rx	BUSIF	Arbitrary	18Bh	0	001	1	1	0	0	
SSIU	ssip6_dreq_tx	Arbitrary	BUSIF	18Ch	0	001	1	1	0	1	
SSIU	ssip7_dreq_rx	BUSIF	Arbitrary	18Dh	0	001	1	1	0	0	
SSIU	ssip7_dreq_tx	Arbitrary	BUSIF	18Eh	0	001	1	1	0	1	
SSIU	ssip8_dreq_rx	BUSIF	Arbitrary	18Fh	0	001	1	1	0	0	
SSIU	ssip8_dreq_tx	Arbitrary	BUSIF	190h	0	001	1	1	0	1	
SSIU	ssip90_dreq_rx	BUSIF	Arbitrary	191h	0	001	1	1	0	0	
SSIU	ssip90_dreq_tx	Arbitrary	BUSIF	192h	0	001	1	1	0	1	
SSIU	ssip91_dreq_rx	BUSIF	Arbitrary	193h	0	001	1	1	0	0	
SSIU	ssip91_dreq_tx	Arbitrary	BUSIF	194h	0	001	1	1	0	1	
SSIU	ssip92_dreq_rx	BUSIF	Arbitrary	195h	0	001	1	1	0	0	
SSIU	ssip92_dreq_tx	Arbitrary	BUSIF	196h	0	001	1	1	0	1	
SSIU	ssip93_dreq_rx	BUSIF	Arbitrary	197h	0	001	1	1	0	0	

Table 4.7-22 DMA Transfer Request Detection Operation Setting Table (15/15)

Request Source	DMA Transfer Request Signal	Transfer source	Transfer destination	DkRQ_SELn	CHCFG_n						
					TM	AM [2:0]	LVL	HIEN	LOEN	REQD	SEL[2:0]
SSIU	ssip93_dreq_tx	Arbitrary	BUSIF	198h	0	001	1	1	0	1	CH0:000 CH1:001 CH2:010 CH3:011 CH4:100 CH5:101 CH6:110 CH7:111 CH8:000 CH9:001 CH10:010 CH11:011 CH12:100 CH13:101 CH14:110 CH15:111
SPDIF CH0	rbdmareqn_tx	Arbitrary	TDAD	199h	0	001	1	1	0	1	
SPDIF CH0	rbdmareqn_rx	RDAD	Arbitrary	19Ah	0	001	1	1	0	0	
SPDIF CH1	rbdmareqn_tx	Arbitrary	TDAD	19Bh	0	001	1	1	0	1	
SPDIF CH1	rbdmareqn_rx	RDAD	Arbitrary	19Ch	0	001	1	1	0	0	
SPDIF CH2	rbdmareqn_tx	Arbitrary	TDAD	19Dh	0	001	1	1	0	1	
SPDIF CH2	rbdmareqn_rx	RDAD	Arbitrary	19Eh	0	001	1	1	0	0	
SCU	pdreq_sr0_in	Arbitrary	in_BUSIF	19Fh	0	001	1	1	0	1	
SCU	pdreq_sr1_in	Arbitrary	in_BUSIF	1A0h	0	001	1	1	0	1	
SCU	pdreq_sr2_in	Arbitrary	in_BUSIF	1A1h	0	001	1	1	0	1	
SCU	pdreq_sr3_in	Arbitrary	in_BUSIF	1A2h	0	001	1	1	0	1	
SCU	pdreq_sr4_in	Arbitrary	in_BUSIF	1A3h	0	001	1	1	0	1	
SCU	pdreq_sr5_in	Arbitrary	in_BUSIF	1A4h	0	001	1	1	0	1	
SCU	pdreq_sr6_in	Arbitrary	in_BUSIF	1A5h	0	001	1	1	0	1	
SCU	pdreq_sr7_in	Arbitrary	in_BUSIF	1A6h	0	001	1	1	0	1	
SCU	pdreq_sr8_in	Arbitrary	in_BUSIF	1A7h	0	001	1	1	0	1	
SCU	pdreq_sr9_in	Arbitrary	in_BUSIF	1A8h	0	001	1	1	0	1	
SCU	pdreq_sr0_out	out_BUSIF	Arbitrary	1A9h	0	001	1	1	0	0	
SCU	pdreq_sr1_out	out_BUSIF	Arbitrary	1AAh	0	001	1	1	0	0	
SCU	pdreq_sr2_out	out_BUSIF	Arbitrary	1ABh	0	001	1	1	0	0	
SCU	pdreq_sr3_out	out_BUSIF	Arbitrary	1ACh	0	001	1	1	0	0	
SCU	pdreq_sr4_out	out_BUSIF	Arbitrary	1ADh	0	001	1	1	0	0	
SCU	pdreq_sr5_out	out_BUSIF	Arbitrary	1AEh	0	001	1	1	0	0	
SCU	pdreq_sr6_out	out_BUSIF	Arbitrary	1AFh	0	001	1	1	0	0	
SCU	pdreq_sr7_out	out_BUSIF	Arbitrary	1B0h	0	001	1	1	0	0	
SCU	pdreq_sr8_out	out_BUSIF	Arbitrary	1B1h	0	001	1	1	0	0	
SCU	pdreq_sr9_out	out_BUSIF	Arbitrary	1B2h	0	001	1	1	0	0	
SCU	pdreq_cmd0_out	out_BUSIF	Arbitrary	1B3h	0	001	1	1	0	0	
SCU	pdreq_cmd1_out	out_BUSIF	Arbitrary	1B4h	0	001	1	1	0	0	

Remark: CHCFG_n register setting values

TM bit
- 0: Single transfer
- 1: Block transfer

AM[2:0] bits
- 000: ACK pulse output
- 001: ACK level output
- 010: ACK bus cycle output
- 100: No ACK

LVL bit
- 0: Detects the edge of a DMA request
- 1: Detects the level of a DMA request

REQD bit
- 0: DACK output during read operation
- 1: DACK output during write operation

(1) Edge detection

Edge detection is selected by setting the LVL bit in the CHCFG_n register.

Setting the HIEN bit of the CHCFG_n register to 1b selects rising-edge detection and setting the LOEN bit to 1b selects falling-edge detection.

Make sure that the unit to be connected issues a next DMA transfer request after waiting for detection of DMAACK.

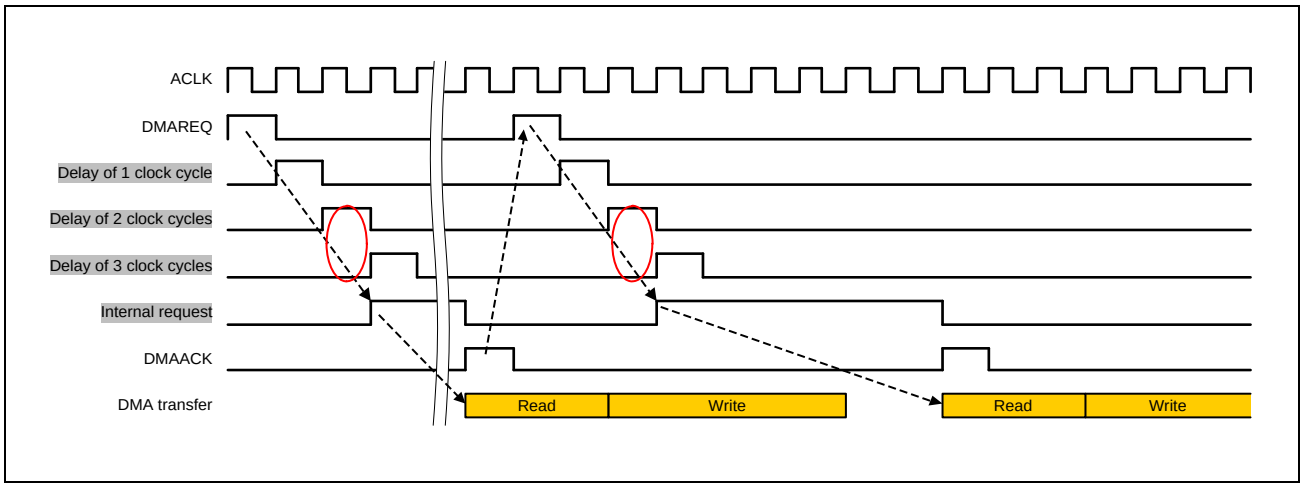


Figure 4.7-16 Edge Detection Timing (HIEN = 1b, REQD = 0b)

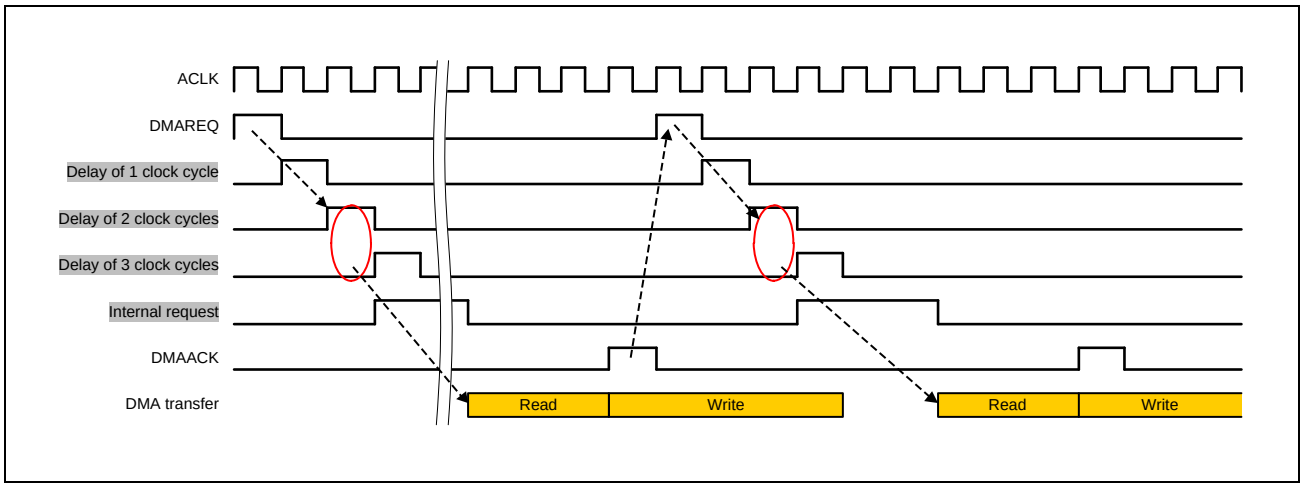


Figure 4.7-17 Edge Detection Timing (HIEN = 1b, REQD = 1b)

(2) Level detection

Setting the LVL bit of the CHCFG_n register to 1b selects level detection.

If DMAREQ is at the active level (by the setting of HIEN or LOEN) for at least two consecutive clock cycles, it is recognized as a correct DMAREQ.

If DMAACK is set to level mode, DMAACK is at the high level until DMAREQ is de-asserted, so when a next DMA transfer request is to be output, assert DMAREQ after waiting for DMAACK to be de-asserted.

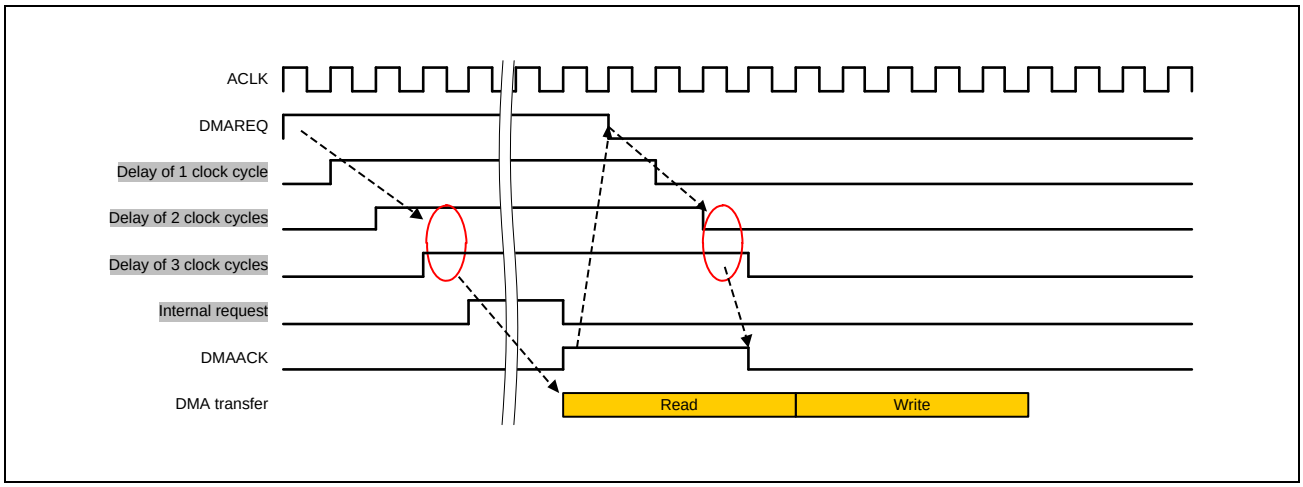


Figure 4.7-18 Level Detection Timing (HIEN = 1b, REQD = 0b, AM[2:0] = 001b)

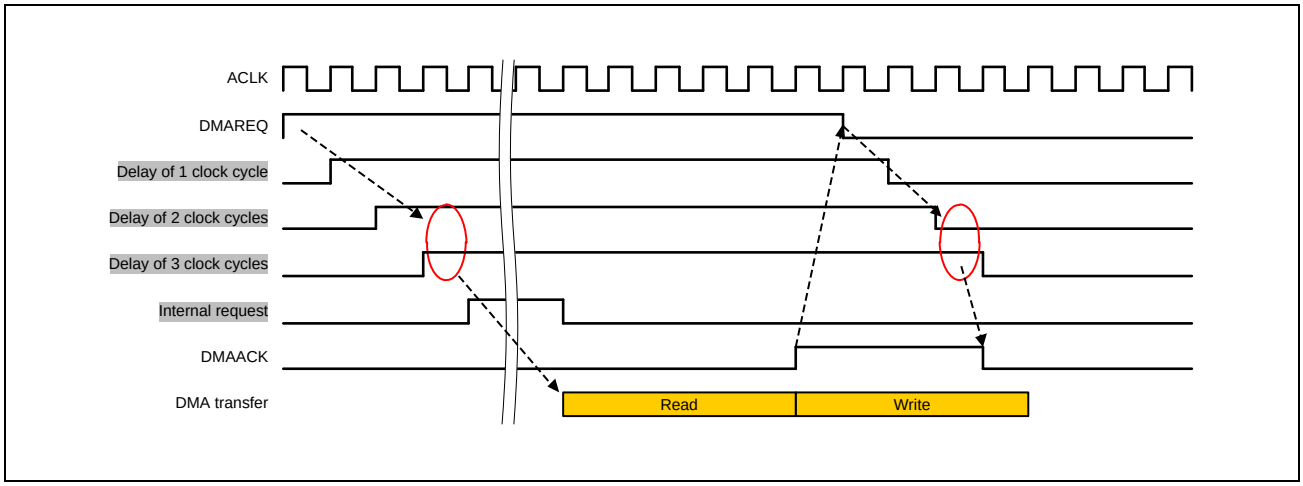


Figure 4.7-19 Level Detection Timing (HIEN = 1b, REQD = 1b, AM[2:0] = 001b)

4.7.6.2.4 DMA acknowledge output

DMAACK[7:0] is an acknowledge signal for the request source that outputs a DMA transfer request. This unit supports pulse output, level output, and bus cycle output to output DMAACK. For handshaking between REQ and ACK, use level output in this LSI.

The SEL bit of the CHCFG_n register is used to select one from among the eight DMAACK[7:0] outputs per channel (See **Table 4.7-20**).

(1) DMA acknowledge signal output timing setting

When the DMA transfer request is accepted, the acknowledge signal becomes active level (high level output). The REQD and AM[2:0] bits of the CHCFG_n register can be used to make the settings as follows.

Table 4.7-23 DMAACK Output Timing Setting

Mode	AM[2] (CHCFG_n)	AM[1:0] (CHCFG_n)	REQD (CHCFG_n)	Usage
Pulse	0b	00b	0b (Active at the time of reading)	DMAACK is output with a pulse. Use this mode when the unit to which DMAACK is connected can receive DMAACK with a pulse.
			1b (Active at the time of writing)	
Level	0b	01b	0b (Active at the time of reading)	DMAACK is output with a level. DMAACK continues to be asserted until DMAREQ is de-asserted. Use this mode when this output is asynchronous with the destination to which DMAACK is connected.
			1b (Active at the time of writing)	
Bus cycle	0b	10b 11b	0b (Active at the time of reading)	DMAACK is output during the bus cycle period. Use this mode when DMAACK is to be asserted until the completion of a bus cycle.
			1b (Active at the time of writing)	
Masking	1b	—	—	DMAACK is fixed at the low level. Use this mode when DMAACK is not to be issued to the destination for connection.

(2) Pulse Output

Setting the AM bits of the CHCFG_m register to 000 selects pulse output.

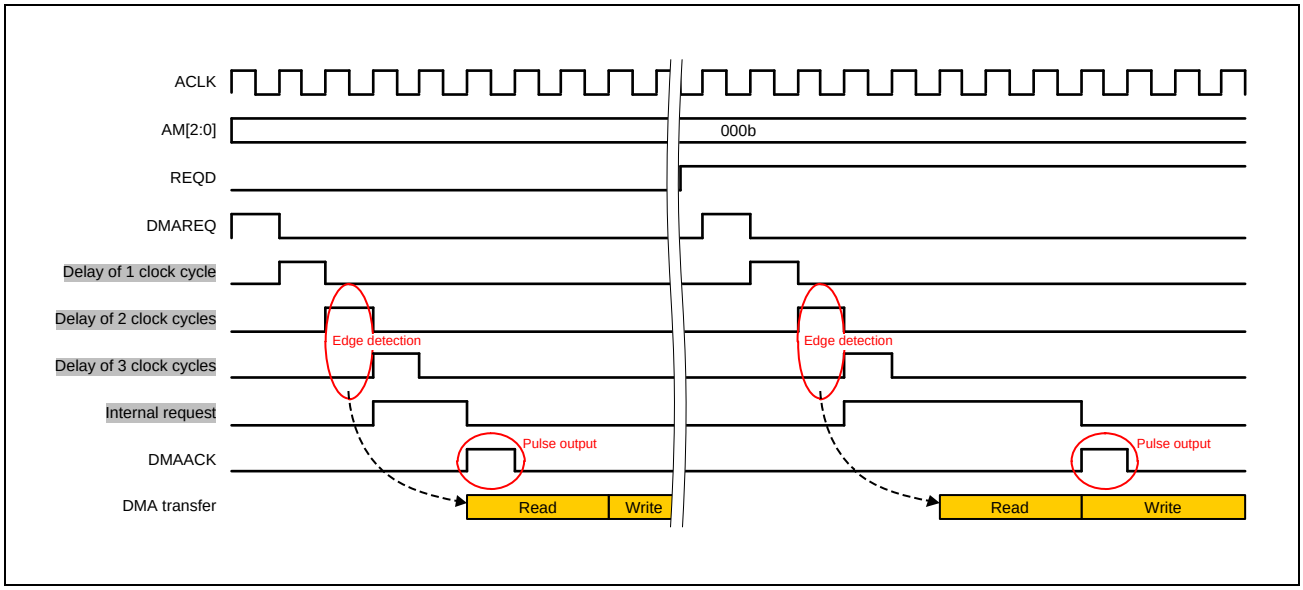


Figure 4.7-20 DMAACK Output Timing (AM[2:0] = 000b, LVL = 0b)

When REQD = 0b, DMAACK is asserted when a bus cycle for reading starts.

When REQD = 1b, DMAACK is asserted when a bus cycle for writing starts.

(3) Level output

Setting the AM bits of the CHCFG_n register to 001b selects level output. DMAACK continues to be asserted until DMAREQ is de-asserted.

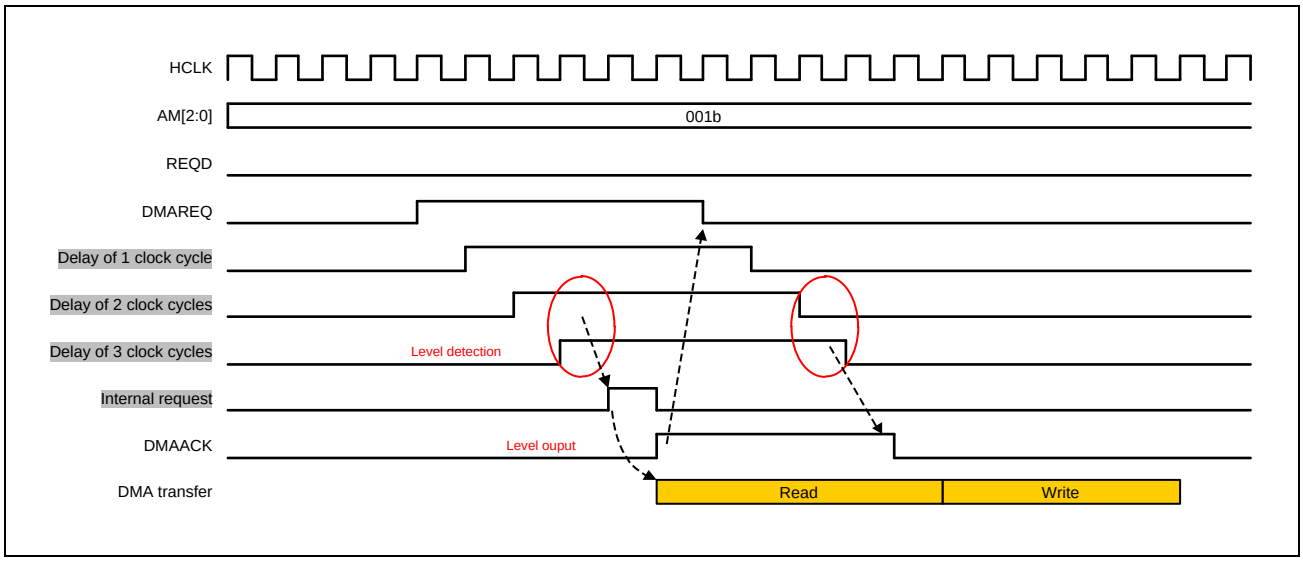


Figure 4.7-21 DMAACK Output Timing (AM[2:0] = 001b, LVL = 1b, REQD = 0b)

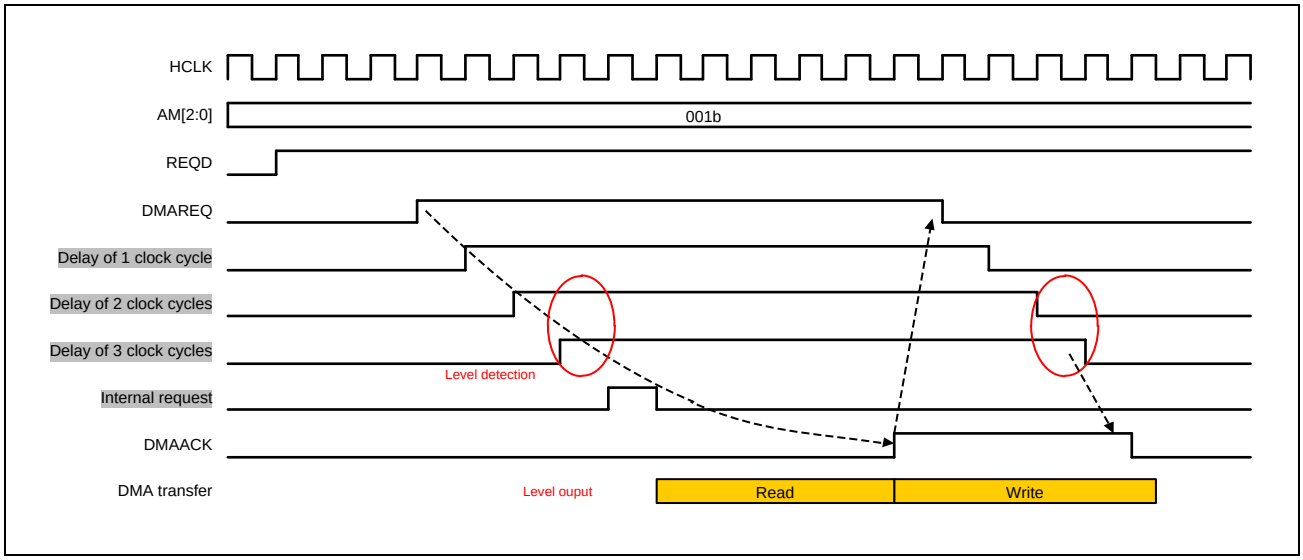


Figure 4.7-22 DMAACK Output Timing (AM[2:0] = 001b, LVL = 1b, REQD = 1b)

(4) Bus Cycle Output

Setting the AM bits of the CHCTRL_m register to 010 selects bus cycle output. DMAACK is at the active level during the bus cycle period.

DMAREQ is masked within this unit while DMAACK is at the active level. Accordingly, even if DMAREQ is set for level detection, the output side does not need to de-assert it once.

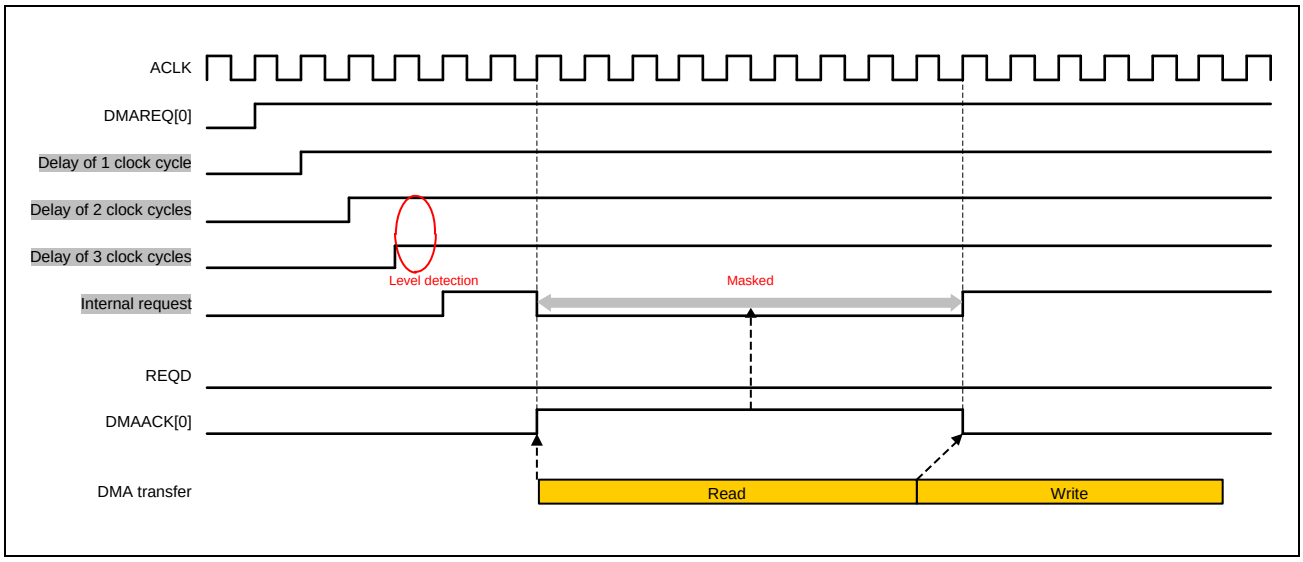


Figure 4.7-23 Bus Cycle Output Timing (AM[2:1] = 01b, LVL = 1b, REQD = 0b)

- When DMAACK is at the active level at the time of reading (REQD = 0b), it is active during the period from the start of the read cycle to one cycle after the end of the read cycle.
  - Start of the read cycle: MARVALID = 1b
  - End of the read cycle: MRVALID & MRLAST & MRREADY = 1b

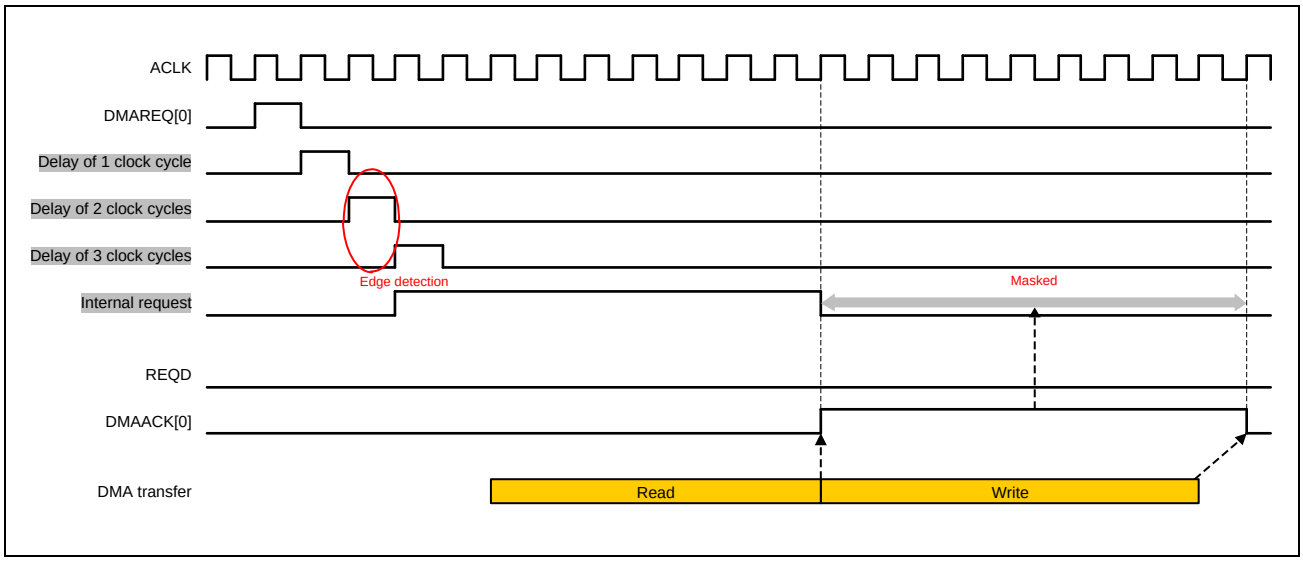


Figure 4.7-24 Bus Cycle Output Timing (AM[2:1] = 01b, LVL = 0b, REQD = 1b)

- When DMAACK is at the active level at the time of writing (REQD = 1b), it is active during the period from the start of the write cycle to one cycle after the end of the write cycle.
  - Start of the write cycle: MAWVALID = 1b
  - End of the write cycle: MBVALID & MBREADY = 1b

4.7.6.2.5 DMA transfer completed interrupt

DMAEND[7:0] are interrupt request signals that indicate the completion of a DMA transaction.

Each bit of DMAEND[7:0] corresponds to a respective channel.

If a transfer for the total number of transfer bytes loaded to CRTB (Current Transaction Byte) is completed with an OKAY response, END of the CHSTAT_n register is set to 1b. If DEM of the CHCFG_n register = 0b at this time, the high level is output from the DMAEND[n] pin (n = 7 to 0) (when write-back proceeds in link mode, the high level is output after the completion of write-back).

In link mode, if LV = 0b in the header of the descriptor that has been read, DER of the CHSTAT_n register is set to 1b. If DIM of the header = 0b at this time, the high level is output from the DMAEND[n] pin.

Use this signal for detection of a transfer completed interrupt by the interrupt controller.

Table 4.7-24 DMAEND Assertion Conditions

Source	Condition	DMAEND[n] Mask Signal
Completion of the DMAC transaction	A transfer for the total number of transfer bytes loaded to CRTB (current transaction byte) being completed with an OKAY response (after the completion of write-back if write-back proceeds in link mode).	DEM bit of the CHCFG_n register
Descriptor invalid	In link mode, LV = 0b in the header of the descriptor that has been read while DIM of the header = 0b.	DIM bit of the header

4.7.6.2.6 DMA terminal count signal output function

DMATCO[7:0] are transaction completion signals to the request source that has issued the DMA transfer request.

One of the eight outputs of DMATCO[7:0] can be selected per channel, which is set by the SEL bit in the CHCFG_n register. (See **Table 4.7-20**).

When a transaction is completed with an OKAY response for the total number of transferred bytes that have been loaded into CRTB_n (Current Transaction Byte), set TC in the CHSTAT_n register to 1b and activate the DMA terminal count signal (DMATCO) for one cycle period (n = 7 to 0).

The DMATCO output signal can be masked by the TCM bit in the CHCFG_n register. When using software startup (setting the STG bit in the CHCTRL_n register), it should be masked.

Table 4.7-25 DMATCO Settings

TCM (CHCFG_n)	Function	Usage
0	Output DMATCO	Used when transfer, count end, or end of link mode is detected by the hardware.
1	Mask DMATCO	Used when DMA transfer is controlled by software; TCM is cleared to 0 after a DMA transaction.

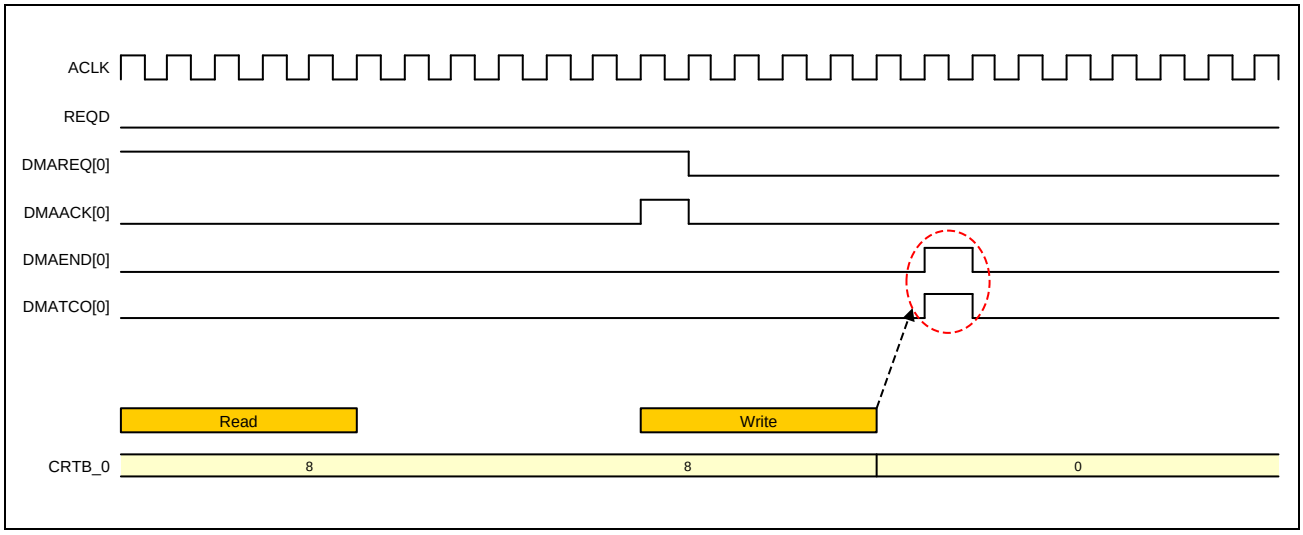


Figure 4.7-25 DMAEND and DMATCO Timing 2 (REQD = 1)

4.7.6.2.7 DMA error interrupt (DMAERR)

If an error response is received in DMA transfer and descriptor access, this unit judges this to be an error and stops transfer. If an error response is received, the EN bit of the CHSTAT_n register for channel n during transfer is cleared to 0b and the ER bit is set to 1b (n = 0 to 7). Also, the high level is output from the DMAERR pin (on the other hand, DMAERR is not output if an error occurs in access to a slave of this unit).

The DMAERR signal cannot be masked.

The series of transfers which causes the error does not guarantee that data. Be sure to start transfer over again from the beginning by following the procedure below.

1. Set the SWRST bit of the CHCTRL_n register to 1b.
2. Re-set each register.

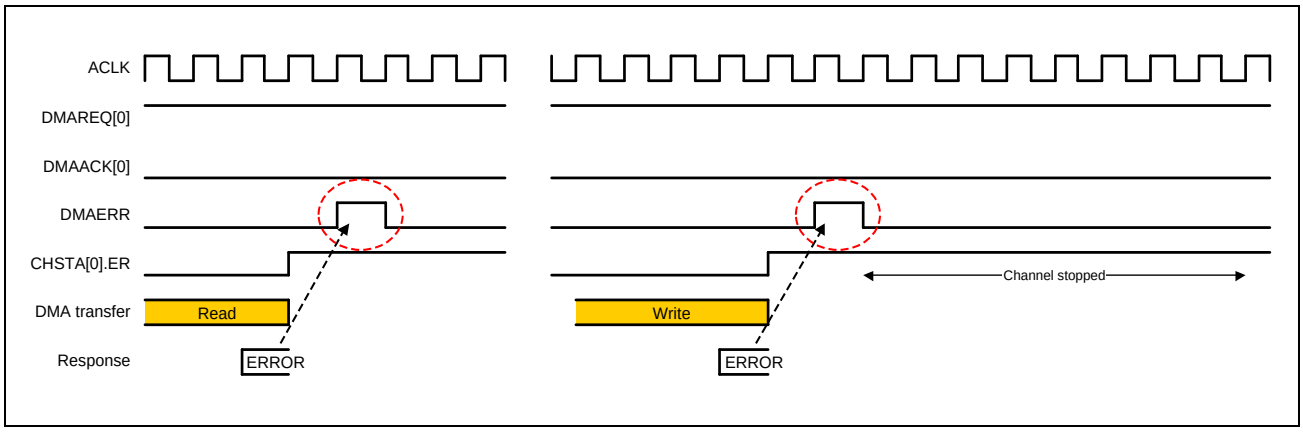


Figure 4.7-26 Timing of Stopping Transfer by an Error Response (ERROR)

4.7.6.2.8 Interval counting

The interval between the execution of DMA transfers can be adjusted by the setting of the ITVL bit of the channel interval register (CHITVL_n). This functionality is intended to make sure that the DMA controller does not occupy the bus.

When one round of reading or writing is completed, counting down starts from the value set in CHITVL_n. A next internal request is not executed until the value counted reaches to 0.

The figure below shows an example of operations.

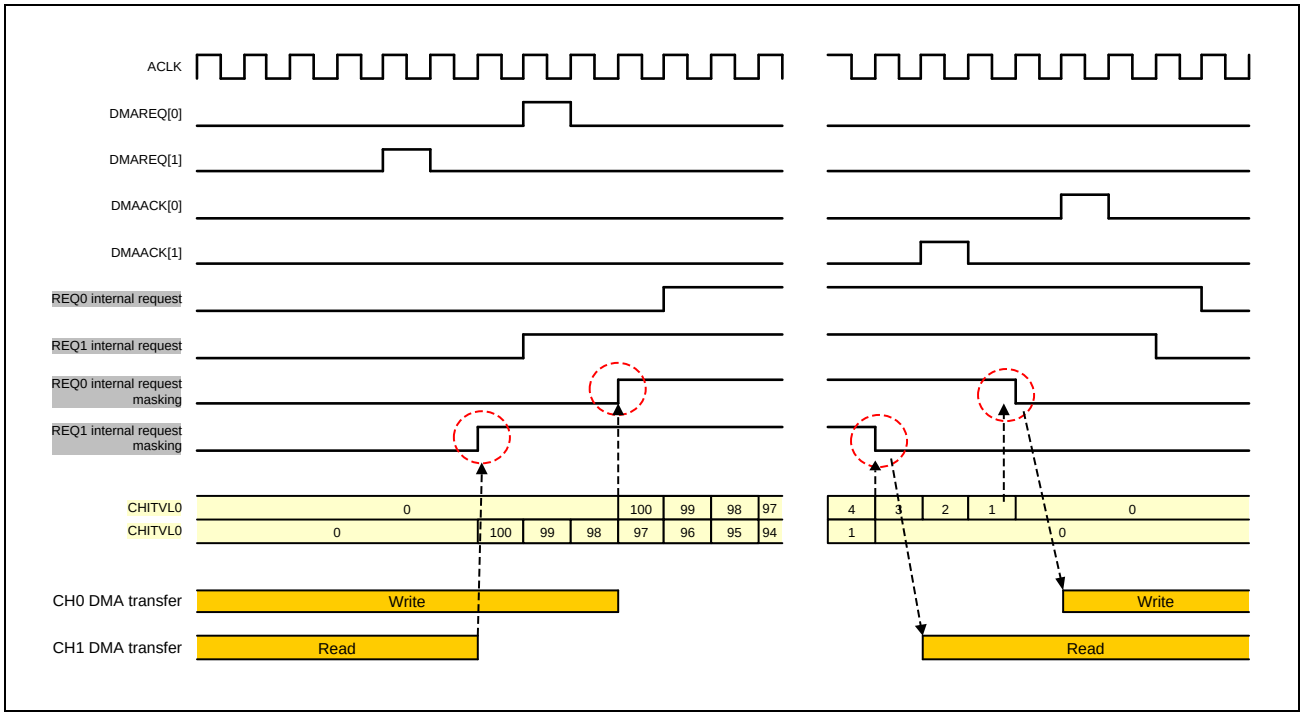


Figure 4.7-27 Interval Counting

4.7.6.2.9 Differences in operation with the transfer size

(1) When the transfer size of the source is small

When reading of data corresponding to the destination transfer size is completed, writing to the destination proceeds.

The figure below is a timing chart when the transfer size of the source is 8 bits and that of the destination is 32 bits (in the case of rising-edge detection).

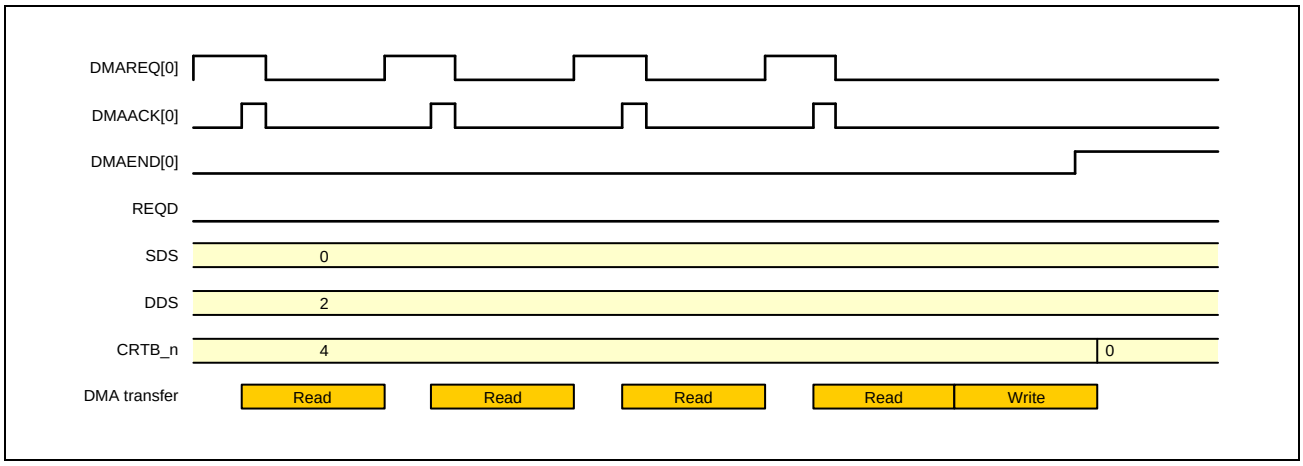


Figure 4.7-28 When the Transfer Size of the Source is Small (LVL of CHCFG_n = 0b, HIEN = 1b, REQD = 0b, SDS < DDS)

(2) When the transfer size of the destination is small

Since the transfer size of the source is larger than that of the destination, after reading from the source once, writing to the destination proceeds several times. The figure below is a timing chart when the transfer size of the source is 64 bits and that of the destination is 16 bits (in the case of rising-edge detection and (REQD of the CHCFG_n register is set to 1).

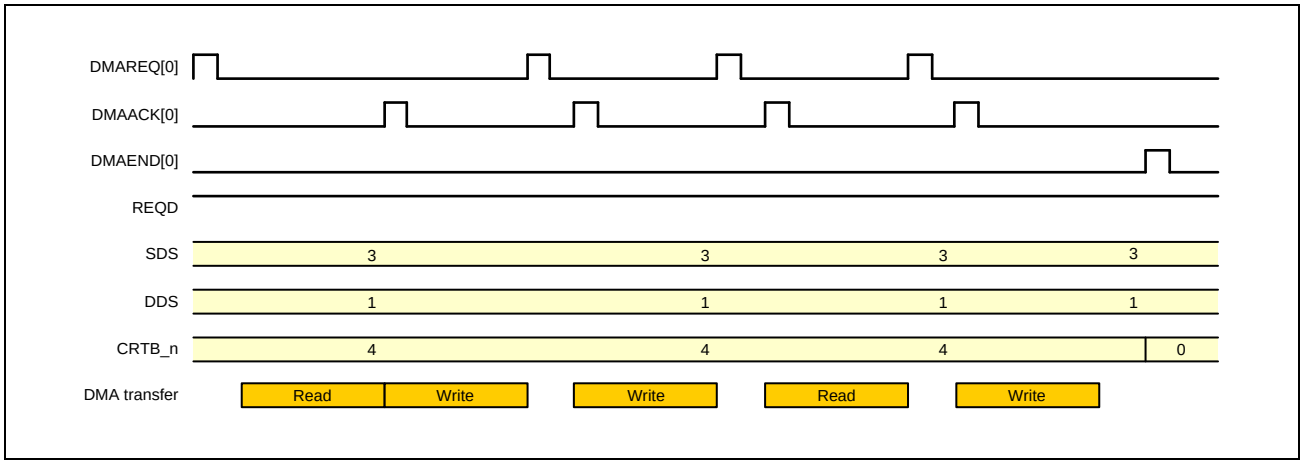


Figure 4.7-29 When the Transfer Size of the Destination is Small (LVL of CHCFG_n = 0b, HIEN = 1b, REQD = 1b, SDS > DDS)

(3) When the transfer sizes of the source and destination are the same

Reading from the source and writing to the source proceed every time a DMA transfer request is detected.

The figure below is a timing chart when the transfer sizes of the source and destination are 8 bits (in the case of rising-edge detection and when REQD of the CHCFG_n register is set to 1b).

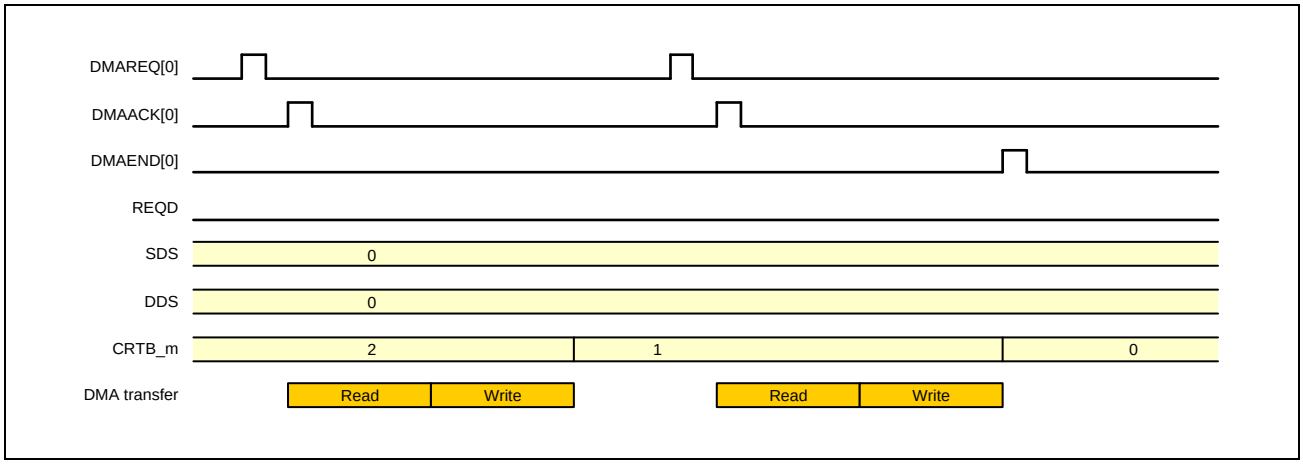


Figure 4.7-30 When the Transfer Sizes of the Source and Destination are the Same (LVL of CHCFG_n = 0b, HIEN = 1b, REQD = 0b, SDS = DDS)

4.7.6.2.10 Transfer state

The channel status register indicates the DMA transfer state of the channel.

(1) Transfer state

The transfer state is indicated by TACT as shown below.

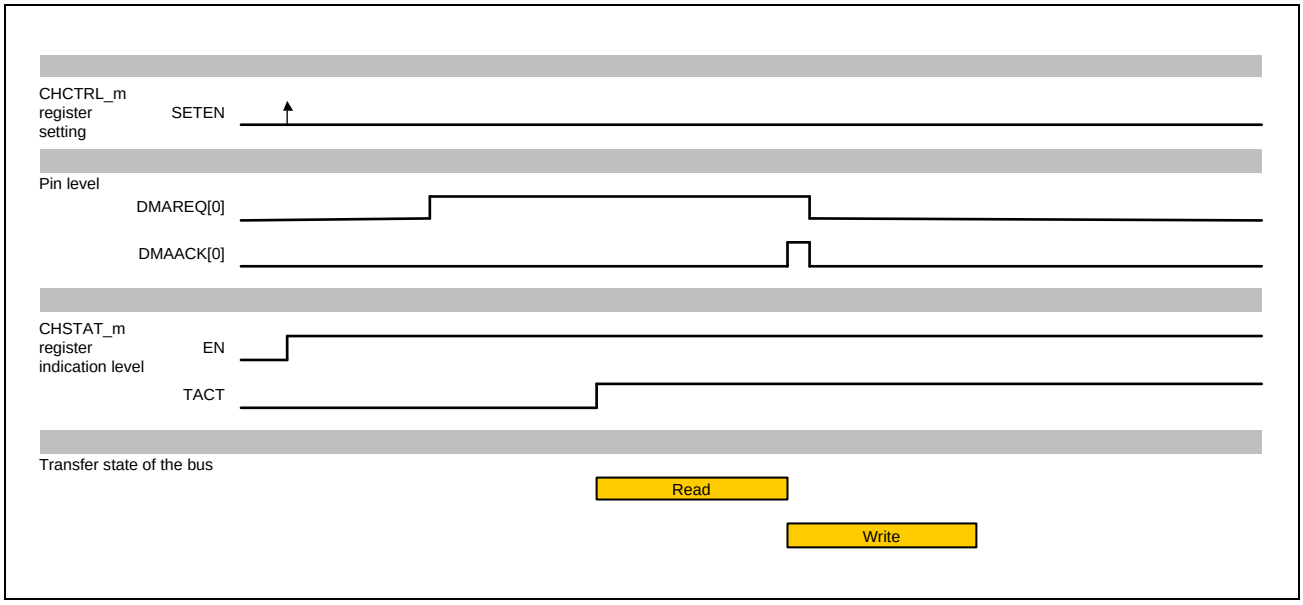


Figure 4.7-31 DMAC State Example 1 (Hardware Request)

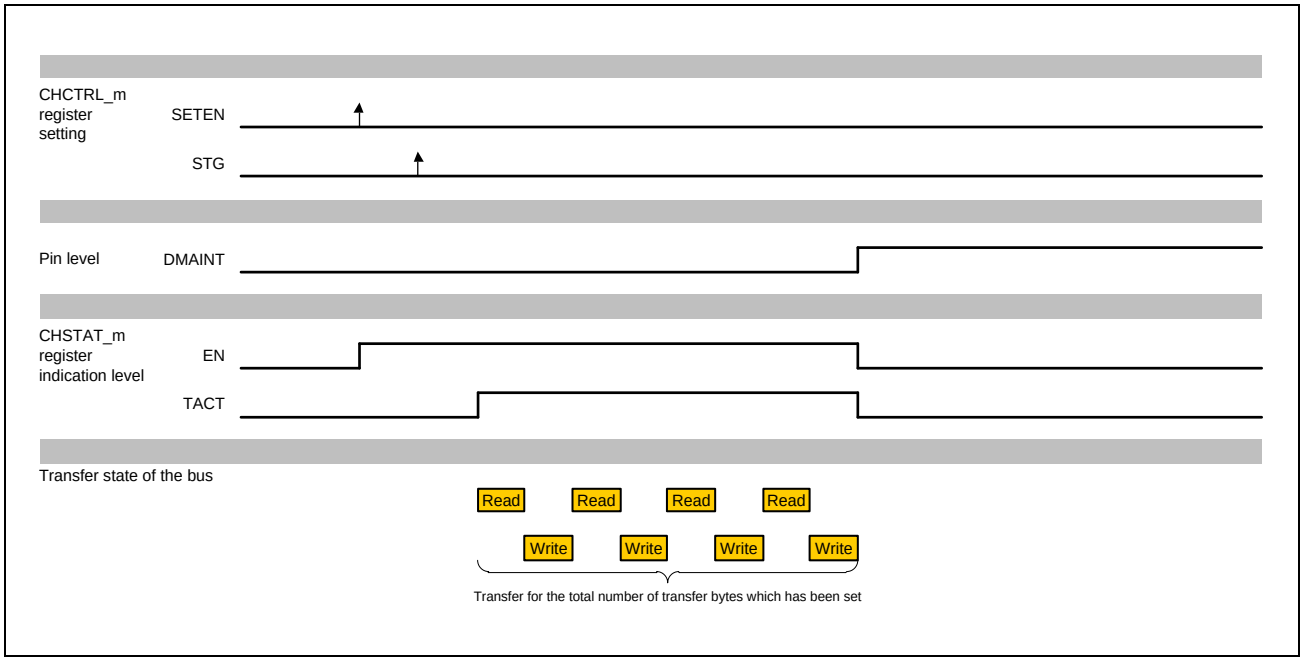


Figure 4.7-32 DMAC State Example 2 (Software Request)

(2) Suspension

The SETSUS bit of CHCTRL_n can be used to suspend DMA transfer. At that time, if the bus cycle is already is in progress, the DMAC is suspended after waiting for the completion of the cycle. It can be resumed from the suspended state by writing 1b to the CLRSUS bit.

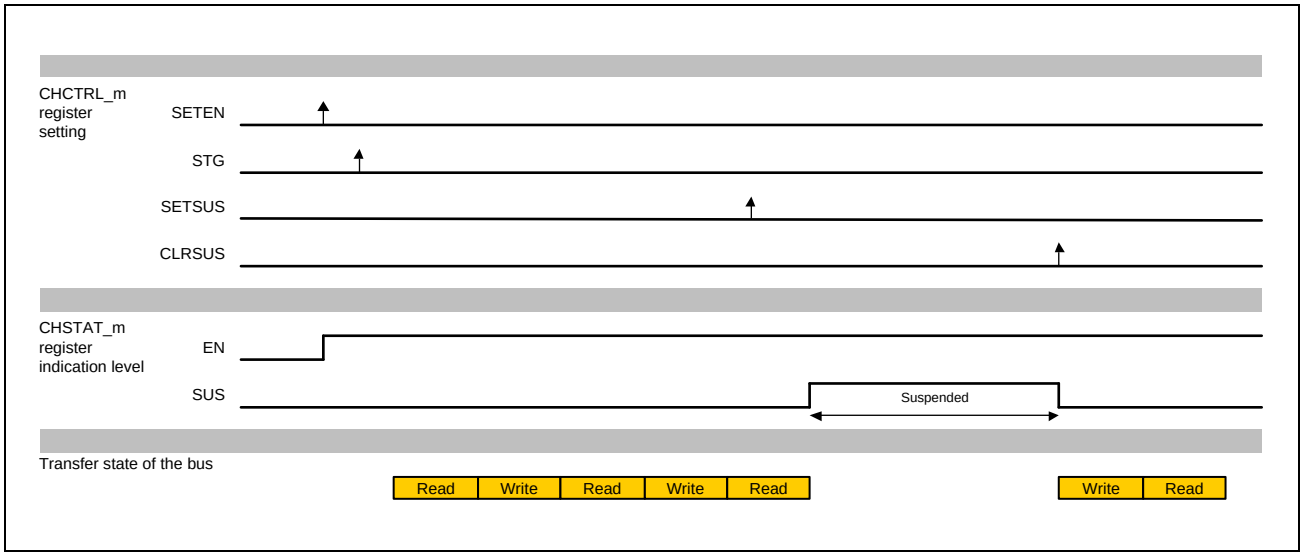


Figure 4.7-33 DMAC Suspended State (Block Transfer by Software Request)

In the above case, the DMAC is suspended at the time read transfer is completed.

If the DMAC has already executed transfer, it is suspended at the time that transfer is completed. To confirm that the transfer is suspended, set SETSUS and then check that the SUS bit for the corresponding channel is 1b by reading the CHSTAT or DST_SUS register.

(3) Aborting transfer

Writing 1 to CLREN during the DMA transaction aborts the DMA transaction by the corresponding channel. As processing after the transaction has been aborted, select either of the following modes by using the SBE bit in the CHCFG_n register: the mode where the data remaining in the buffer are flushed at the time the transaction is aborted (SBE = 1b) and the mode where the data are not flushed (SBE = 0b). By default, SBE = 0b.

When the transfer in progress is aborted by setting CLREN to 1b while this flushing mode is enabled, if data remain in the buffer of the DMAC, that data are flushed before the transaction is completed.

(a) Aborting transfer (no buffer flushing: SBE = 0b)

When CLREN is set during DMA transfer, the DMA transfer is aborted and then stopped. The timing of stopping the transfer depends on the value set in REQD. After the transfer has been stopped, be sure to clear the internal state of the DMAC before making the settings for next transfer.

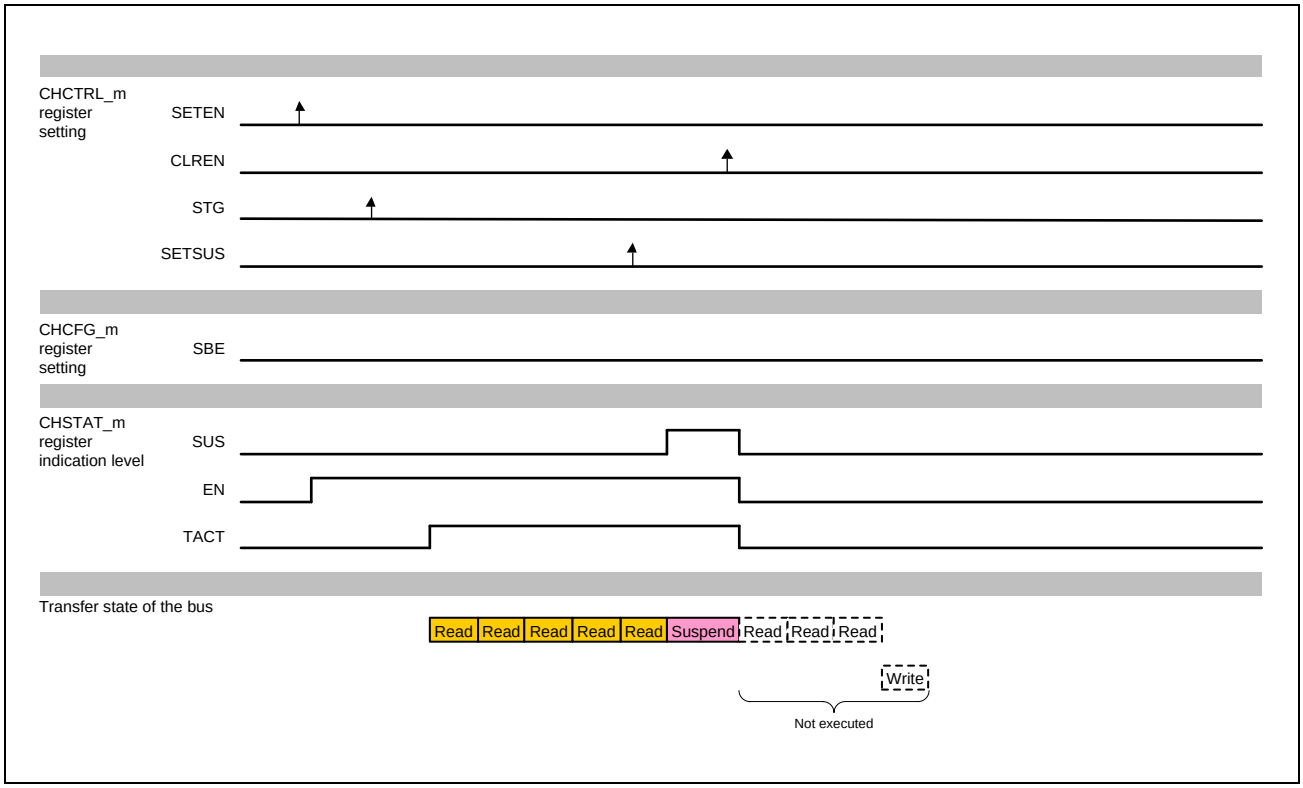


Figure 4.7-34 Aborting DMA Transfer

- Complete stopping of the channel can be confirmed at the time the TACT bit is cleared.
- If the DMA transfer in progress is aborted, the DMAEND pin is not asserted.
- If REQD = 0b, the transfer is stopped at the time next reading is completed. Note that if the buffer contains data that can be written, the transfer is stopped after the data are written.
- If REQD = 1b, the transfer is stopped at the time next writing is completed.

(b) Aborting transfer (with buffer flushing: SBE = 1b)

Setting CLREN during DMA transfer aborts the DMA transfer. If REQD = 0b, data that have already been read are flushed (written) and then the DMA transfer is stopped. After the transfer has been stopped, set SWRST and clear the internal state of the DMAC before making the settings for next transfer.

If flushing mode is used when a hardware request is in use while REQD = 1b, data that have been read in advance may be written, so only use this mode while REQD = 0b.

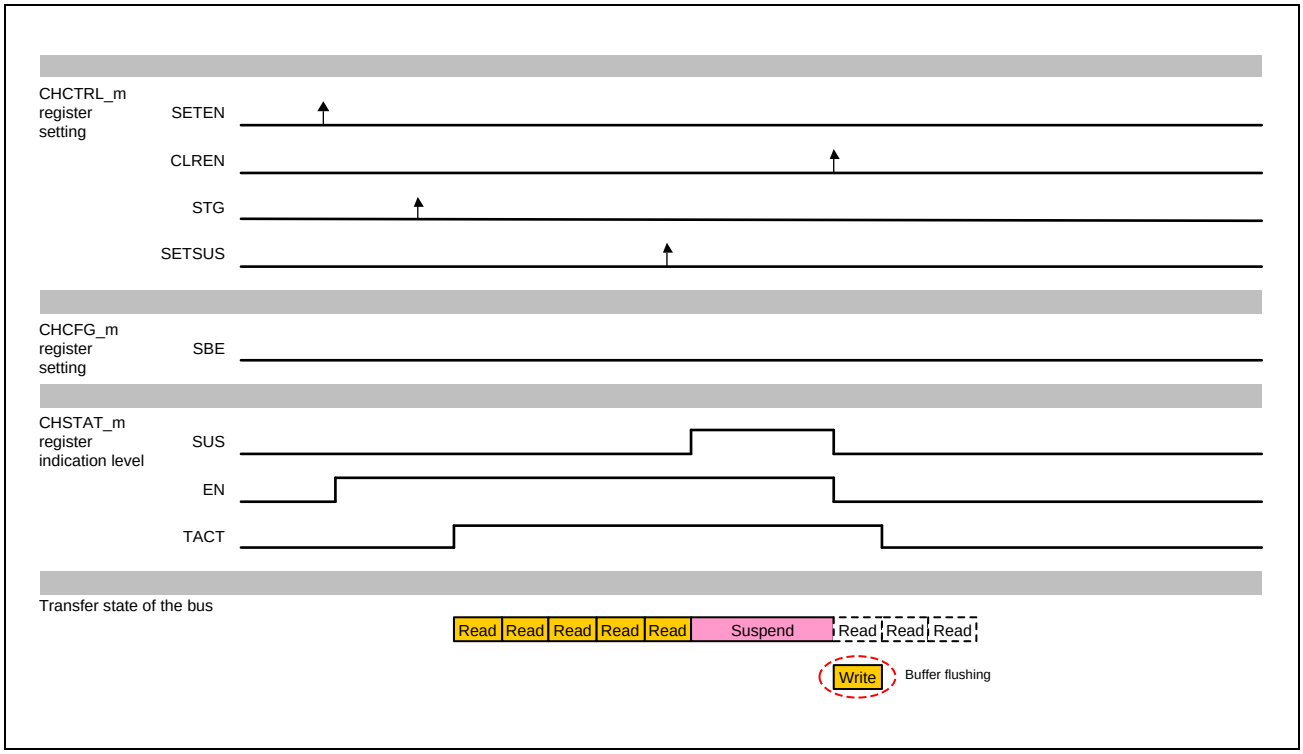


Figure 4.7-35 Aborting DMA Transfer (Buffer Flushing Mode)

- Complete stopping of the channel can be confirmed at the time the TACT bit is cleared.
- If the transfer is aborted during the fifth read transfer in flushing mode (SBE = 1b), the DMAC is stopped after writing the data that have been read.

(c) Checking stopping of the channel

Even if the EN bit is cleared to 0b, the DMAC cannot be stopped immediately if transfer has already been executed on the bus. Accordingly, to confirm that the DMAC has been stopped completely, check that both the EN and TACT bits are 0b.

(d) Procedure for aborting transfer

The following describes the procedure for stopping transfer.

1. Set SETSUS of CHCTRL_n.

2. Poll the SUS bit of CHSTAT_n until it becomes 1b (if EN is already 0b at this time, proceed to step 6 because the DMAC has been stopped).

3. Set CLREN of CHCTRL_n.

4. If SBE = 0b, the DMAC is stopped according to the value of REQD, and if SBE = 1b, the data are flushed. If the setting of SBE = 1b, set REQD = 0b beforehand.

5. Read CHSTAT_n to check that the TACT bit is 0b. TACT = 0b means that the DMAC has been stopped completely. If TACT = 1b, poll this bit until it becomes 0b.

6. To proceed with a next DMA transfer after aborting the transfer, be sure to set the SWRST (software reset) bit of CHCTRLn before the next transfer starts.

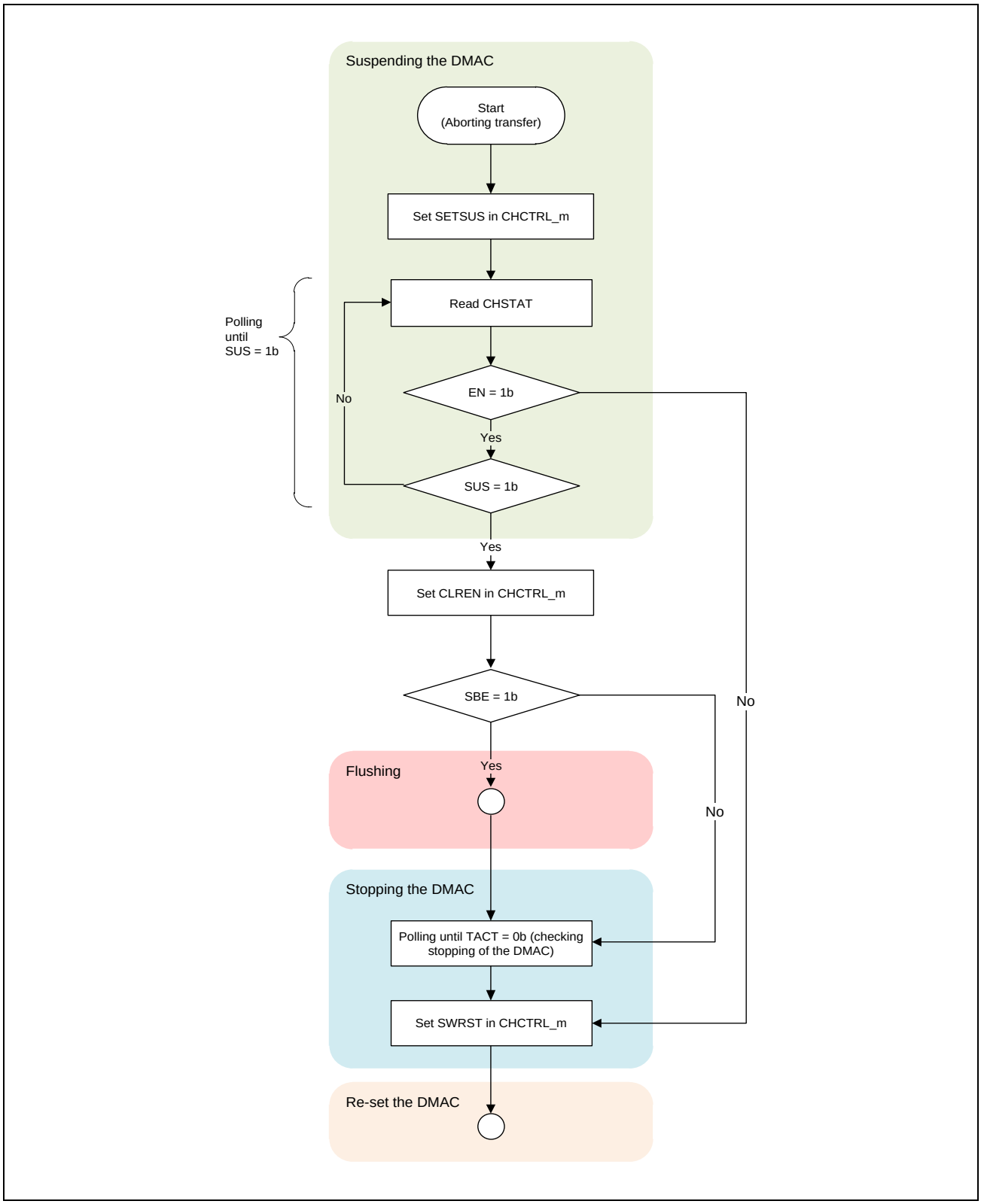


Figure 4.7-36 Flow of Aborting Transfer

4.7.6.3 AXI Address Conversion

Two bank switching address bits input by SYS are appended to the higher-order bits of the AXI master addresses to extend them to 34-bit master addresses, which makes a 16-GB address space accessible.

Since three bank switching addresses for the source, destination, and descriptor are input per DMAC number and channel number, the source, destination, and descriptor areas can be set in separate 4-GB address spaces.

Bank switching addresses are set by using the registers of the SYS unit.

The figure below shows an example of access to the 16-GB address space.

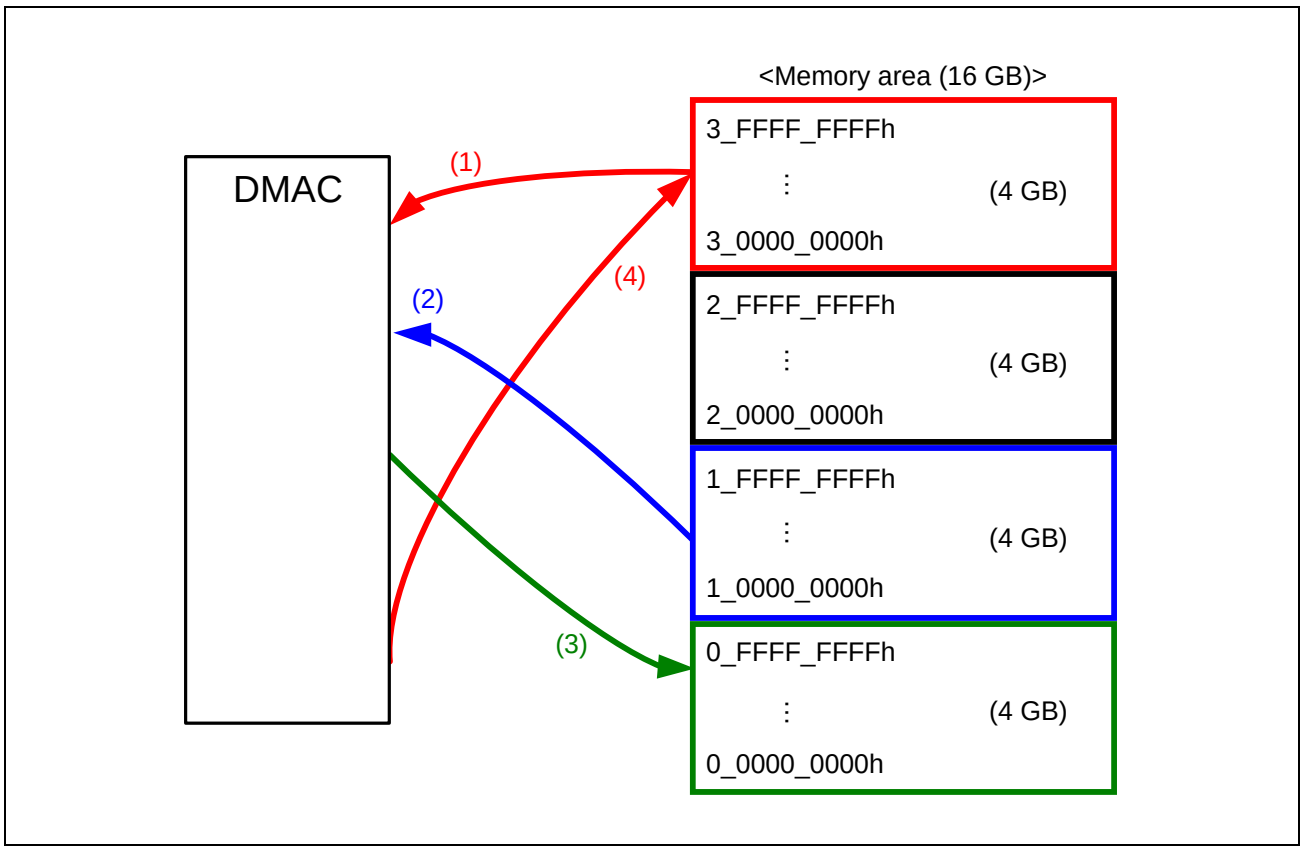


Figure 4.7-37 Example of Access to the 16-GB Address Space

(1) Descriptor read

- Access to a desired 4-GB address space is possible by setting the bank switching address (for the descriptor) for the corresponding DMAC number and channel.
- In the example of the above figure, the setting is 3.
- Setting is only required when link mode is in use.

(2) DMA transfer (source data read)

- Access to a desired 4-GB address space is possible by setting the bank switching address (for the source) for the corresponding DMAC number and channel.
- In the example of the above figure, the setting is 1.

(3) DMA transfer (destination data write)

- Access to a desired 4-GB address space is possible by setting the bank switching address (for the destination) for the corresponding DMAC number and channel.
- In the example of the above figure, the setting is 0.

(4) Descriptor write-back

- The same area as that in step 1.

4.7.6.3.1 Prohibited items

This function has the following prohibited items.

- Changing a bank switching address value for a DMAC number and channel that is currently operating is prohibited.
- Usage that spans the 4-GB boundaries is prohibited.

4.7.7 Operating Procedure

For the individual setting procedures, see the following. The following gives supplementary information.

4.7.7.1 Example DMAC Settings

This section describes example settings when DMA transfer is to proceed with the use of the DMAC unit.

The conditions for transfer with the individual example settings are listed below.

Table 4.7-26 List of the Conditions for Transfer with Example DMA Transfer Settings

	DMA Mode	Transfer Mode	Transfer Request
Example setting 1	Register	Single	Hardware
Example setting 2	Register	Block	Software
Example setting 3	Register (continuous execution)	Block	Software
Example setting 4	Link	Block	Software

For details of the settings, see the individual example settings.

4.7.7.1.1 Example setting 1 (hardware request in register mode)

The table below lists the settings when DMA transfer is to proceed with example setting 1.

Table 4.7-27 Example Setting 1 for DMA Transfer

Item		Description	
Channel to be used		3	
DMA mode		Register	
Transfer mode		Single transfer	
Register set to be used		Next0	
Transfer source/destination		Transfer source	Transfer destination
	Start address	1111_0000h	2222_0000h
	Address direction	Incremental	Incremental
	Transfer size	32 bits	32 bits
Number of bytes for DMA transfer		64 bytes	
DMAREQ/ACK/TCO		Select DMAREQ[3], DMAACK[3], and DMATCO[3].	
DMA transfer request		Rising-edge detection by hardware (the DMAREQ[3] pin)	
DMAACK signal		Pulse output at the time of reading	
DMAEND masking		No	
AXI setting (PROT, CACHE)		Initial value	

Example setting 1

N0SA = 1111_0000h (Transfer source address)

N0DA = 2222_0000h (Transfer destination address)

N0TB = 0000_0040h (Number of transfer bytes)

CHCFG = 0002_2023h (Configuration)

CHITVL = 0000_0000h (Interval)

CHEXT = 0000_0000h (AXI setting)

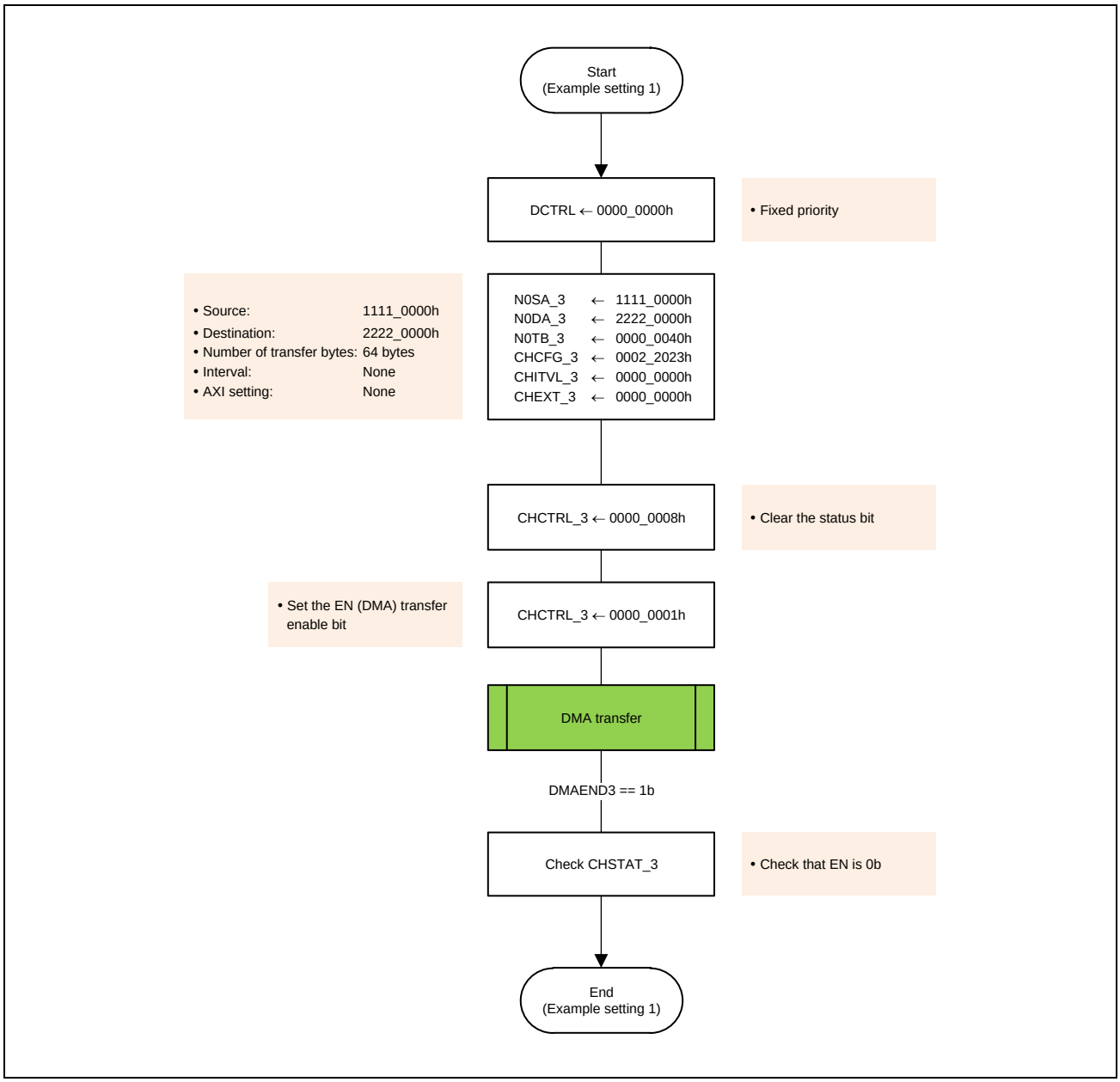


Figure 4.7-38 Example Setting 1

4.7.7.1.2 Example setting 2 (software request in register mode)

The table below lists the settings when DMA transfer is to proceed with example setting 2.

Table 4.7-28 Example Setting 2 for DMA Transfer

Item		Description	
Channel to be used		2	
Priority control		Round-robin	
DMA mode		Register	
Transfer mode		Block transfer	
Register set to be used		Next1	
Transfer source/destination		Transfer source	Transfer destination
	Start address	0FFF_E000h	3333_0000h
	Address direction	Incremental	Incremental
	Transfer size	8 bits	256 bits
Number of bytes for DMA transfer		128 bytes	
DMAREQ/ACK/TCO		Select DMAREQ[7], DMAACK[7], and DMATCO[7].	
DMA transfer request		Software request	
DMAACK signal		Masking	
DMAEND masking		No	
AXI setting (PROT, CACHE)		Initial value	

Example setting 2

DCTRL = 0000_0001h (DMAC setting)

N1SA = 0FFF_E000h (Transfer source address)

N1DA = 3333_0000h (Transfer destination address)

N1TB = 0000_0080h (Number of transfer bytes)

CHCFG = 1045_0407h (Configuration)

CHITVL = 0000_0000h (Interval)

CHEXT = 0000_0000h (AXI setting)

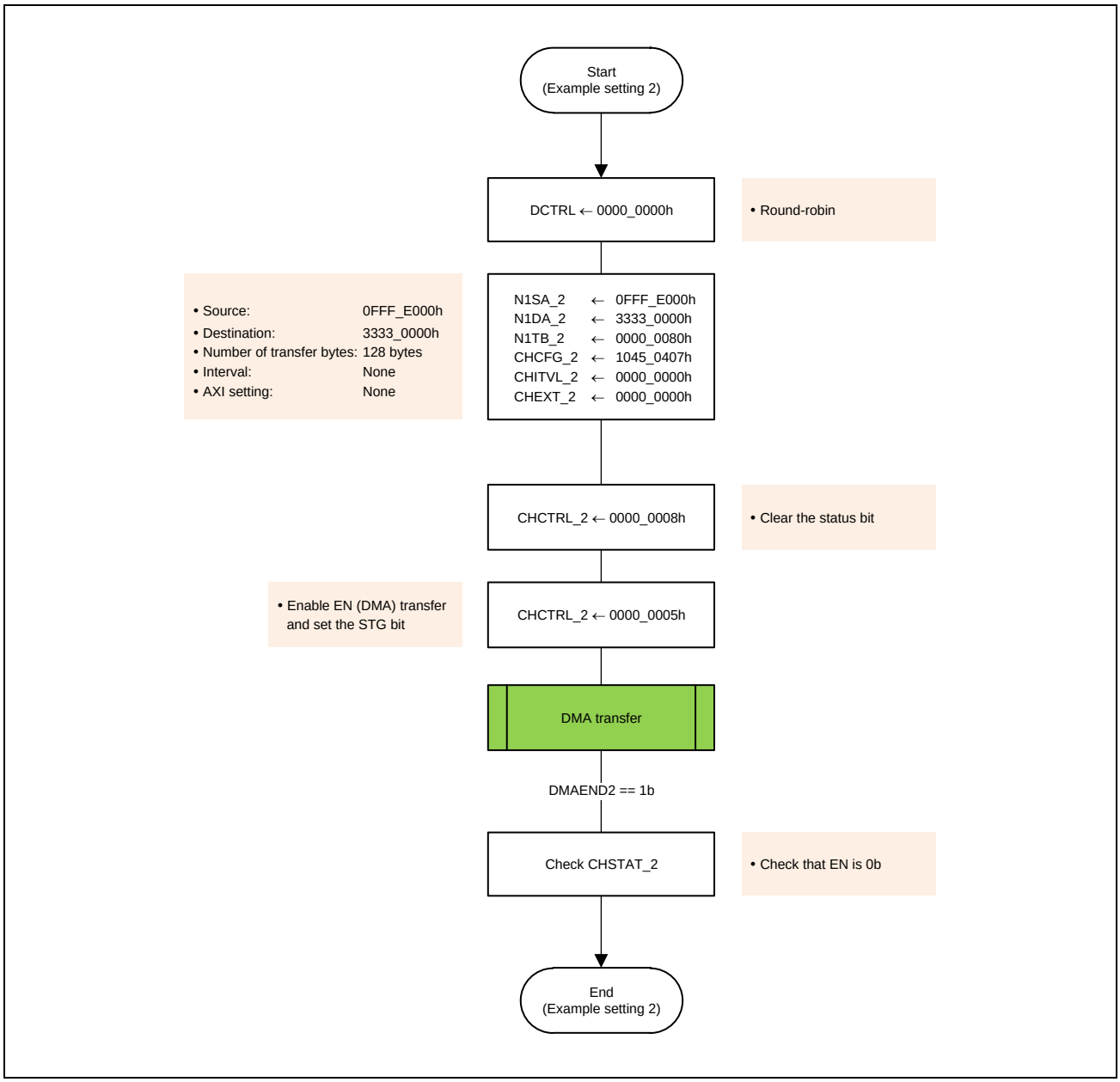


Figure 4.7-39 Example Setting 2

4.7.7.1.3 Example setting 3 (continuous execution in register mode)

The table below lists the settings when DMA transfer is to proceed with example setting 3.

Table 4.7-29 Example Setting 3 for DMA Transfer

Item		Description	
Channel to be used		1	
Priority control		Round-robin	
DMA mode		Register	
Transfer mode		Block transfer	
Register set to be used		Continuous from Next0 to Next1	
Next0		Transfer source	Transfer destination
	Start address	1111_0000h	3333_0000h
	Address direction	Fixed	Fixed
	Transfer size	32 bits	512 bits
	Number of bytes for DMA transfer	512 bytes	
Next1		Transfer source	Transfer destination
	Start address	2222_0000h	4444_0000h
	Address direction	Fixed	Fixed
	Transfer size	32 bits	512 bits
	Number of bytes for DMA transfer	2048 bytes	
DMAREQ/ACK/TCO		Select DMAREQ[7], DMAACK[7], and DMATCO[7].	
DMA transfer request		Software request	
DMAACK signal		Not output	
DMAEND masking		Masking DMAEND on completion of Next0	
AXI setting (PROT, CACHE)		Initial value	

Example setting 3

DCTRL = 0000_0001h (DMAC setting)

N0SA = 1111_0000h (Transfer source address)

N0DA = 3333_0000h (Transfer destination address)

N0TB = 0000_0200h (Number of transfer bytes)

N1SA = 2222_0000h (Transfer source address)

N1DA = 4444_0000h (Transfer destination address)

N1TB = 0000_0800h (Number of transfer bytes)

CHCFG = 6176_2007h (Configuration)

CHITVL = 0000_0000h (Interval)

CHEXT = 0000_0000h (AXI setting)

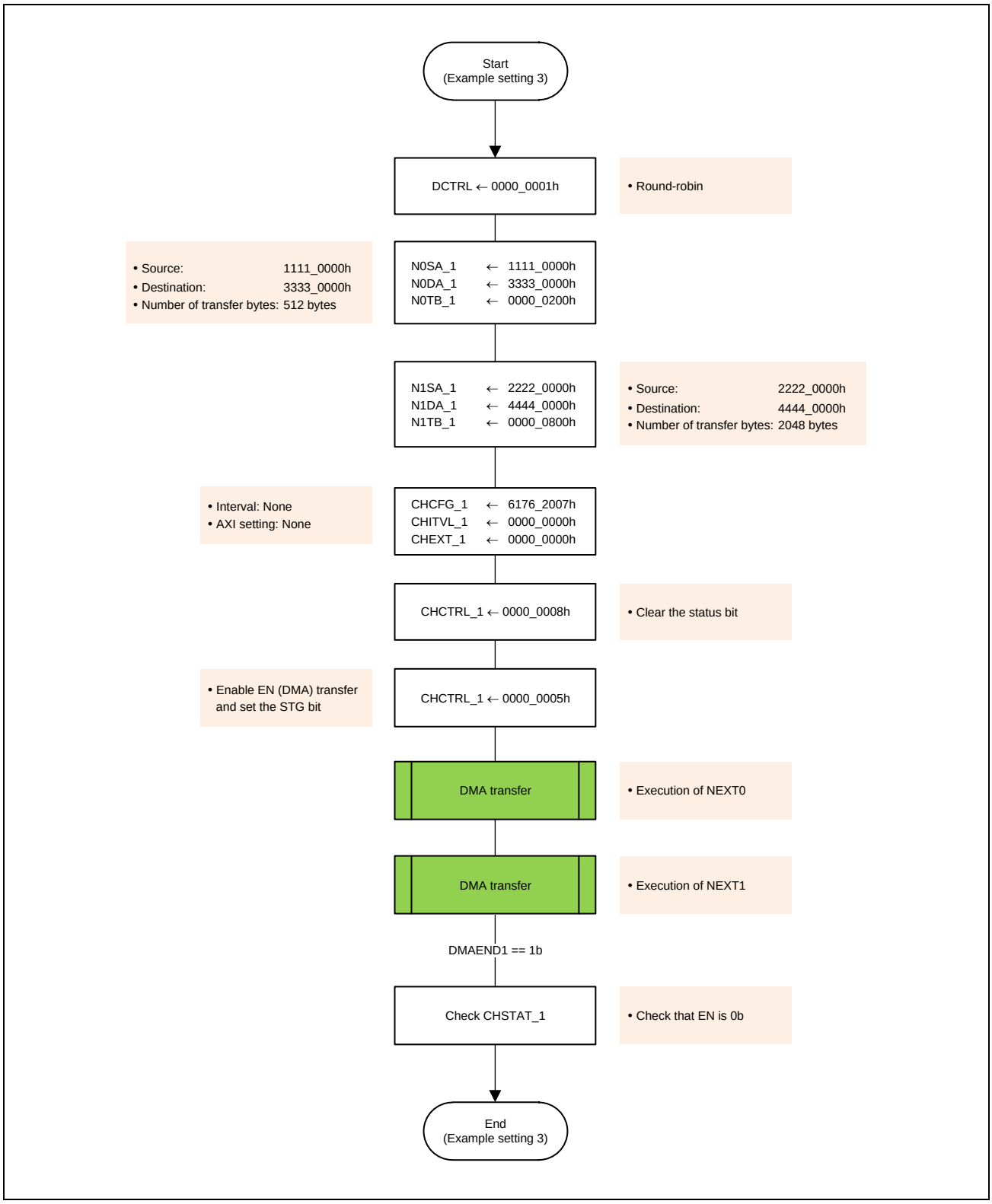


Figure 4.7-40 Example Setting 3

4.7.7.1.4 Example setting 4 (link mode)

The table below lists the settings when DMA transfer is to proceed with example setting 4.

Table 4.7-30 Example Setting 4 for DMA Transfer

Item	Description
Channel to be used	0
Priority control	Round-robin
DMA mode	Link
Transfer mode	Block transfer
Register set to be used	—
Descriptor start address	0000_1000h

Table 4.7-31 Example Setting 4 for DMA Transfer (Descriptor 1)

Item		Description	
Descriptor start address		0000_1000h	
Next descriptor start address		0000_2000h	
Transfer mode		Block transfer	
Next0		Transfer source	Transfer destination
	Start address	1111_0000h	3333_0000h
	Address direction	Incremental	Incremental
	Transfer size	32 bits	32 bits
	Number of bytes for DMA transfer	2048 bytes	
DMAREQ/ACK/TCO		Select DMAREQ[0], DMAACK[0], and DMATCO[0].	
DMA transfer request		Software activation (STG)	
DMAACK signal		Not output	
DMAEND masking		Yes	
AXI setting (PROT, CACHE)		Initial value	
Header			
	DMAEND when LV = 1b	Issued (DIM = 0b)	
	LV write-back	Yes (WBD = 0b)	
	Next link destination	Yes (LE = 0b)	
	Descriptor enable	Enable (LV = 1b)	

Table 4.7-32 Example Setting 4 for DMA Transfer (Descriptor 2)

Item		Description	
Descriptor start address		0000_2000h	
Next descriptor start address		0000_5000h	
Transfer mode		Block transfer	
Next0		Transfer source	Transfer destination
	Start address	4444_0000h	5555_0000h
	Address direction	Incremental	Incremental
	Transfer size	64 bits	256 bits
	Number of bytes for DMA transfer	1024 bytes	
DMAREQ/ACK/TCO		Select DMAREQ[0], DMAACK[0], and DMATCO[0].	
DMA transfer request		Software activation (STG)	
DMAACK signal		Not output	
DMAEND masking		Yes	
AXI setting (PROT, CACHE)		Initial value	
Header			
	DMAEND when LV = 1b	Issued (DIM = 0b)	
	LV write-back	Yes (WBD = 0b)	
	Next link destination	Yes (LE = 0b)	
	Descriptor enable	Enable (LV = 1b)	

Table 4.7-33 Example Setting 4 for DMA Transfer (Descriptor 3)

Item		Description	
Descriptor start address		0000_5000h	
Next descriptor start address		—	
Transfer mode		Block transfer	
Next0		Transfer source	Transfer destination
	Start address	7777_0000h	AAAA_0000h
	Address direction	Incremental	Incremental
	Transfer size	512 bits	512 bits
	Number of bytes for DMA transfer	4096 bytes	
DMAREQ/ACK/TCO		Select DMAREQ[0], DMAACK[0], and DMATCO[0].	
DMA transfer request		Software activation (STG)	
DMAACK signal		Not output	
DMAEND masking		No	
AXI setting (PROT, CACHE)		Initial value	
Header			
	DMAEND when LV = 1b	Issued (DIM = 0b)	
	LV write-back	Yes (WBD = 0b)	
	Next link destination	No (LE = 1b)	
	Descriptor enable	Enable (LV = 1b)	

Example setting 4

DCTRL= 0000_0001h (DMAC setting)

NXLA = 0000_1000h (Descriptor start address)

CHCFG = 8000_0000h (Configuration)

Table 4.7-34 Descriptor Settings

	Descriptor 1	Descriptor 2	Descriptor 3
Header	0000_0001h	0000_0001h	0000_0003h
SA (source address)	1111_0000h	4444_0000h	7777_0000h
DA (destination address)	3333_0000h	5555_0000h	AAAA_0000h
TB (transaction byte)	0000_0800h	0000_0400h	0000_1000h
CFG (configuration)	8342_2008h	8345_3008h	8246_6008h
ITVL (interval)	0000_0000h	0000_0000h	0000_0000h
EXT (extension)	0000_0000h	0000_0000h	0000_0000h
NXLA (next link address)	0000_2000h	0000_5000h	0000_0000h

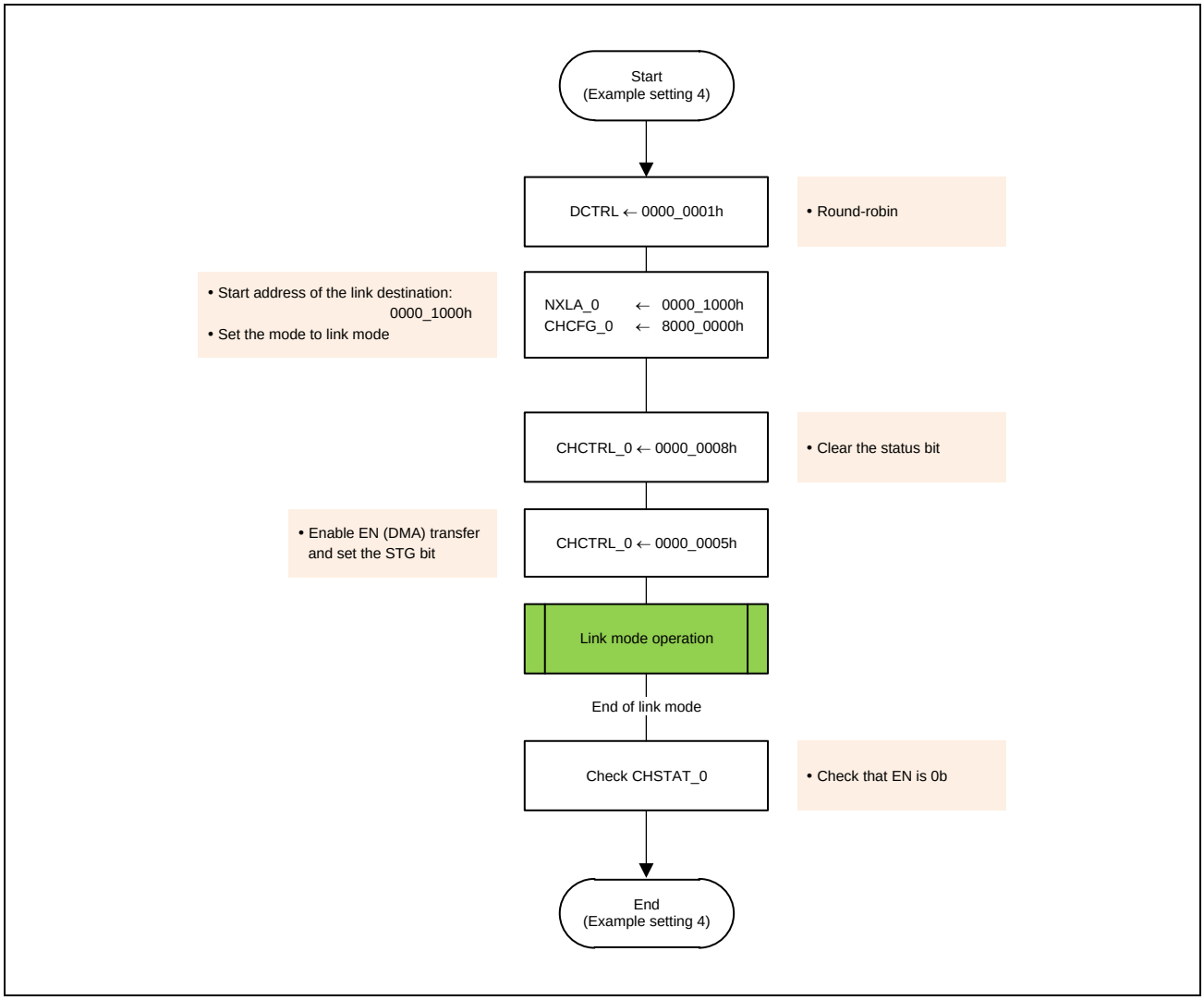


Figure 4.7-41 Example Setting 4

4.7.7.1.5 Setting for next register continuous execution

The figure below is a flowchart when DMA transfer continues with the use of two Next register sets in register mode. While the DMA transaction of one Next register is in progress, the other Next register is set to continue with DMA transfer.

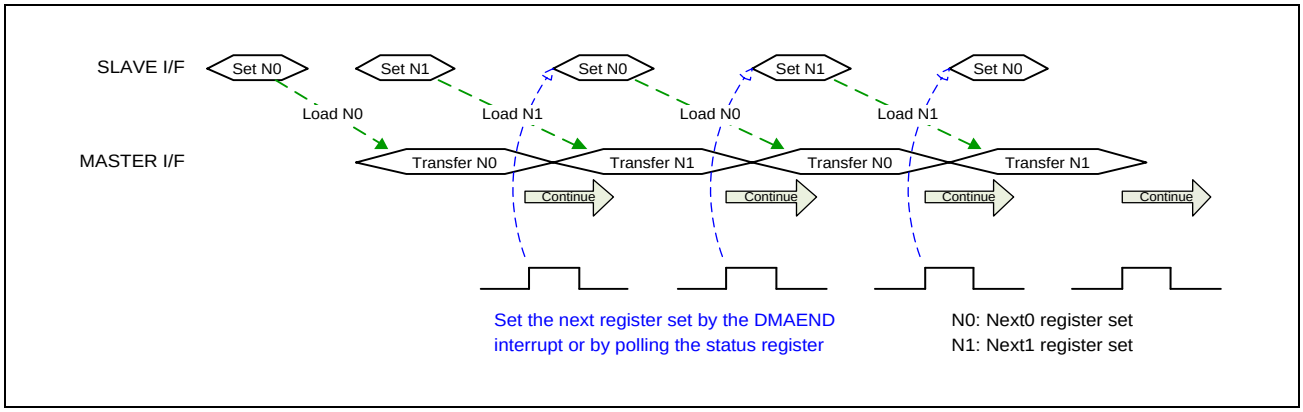


Figure 4.7-42 Schematic View of Next Register Continuous Execution

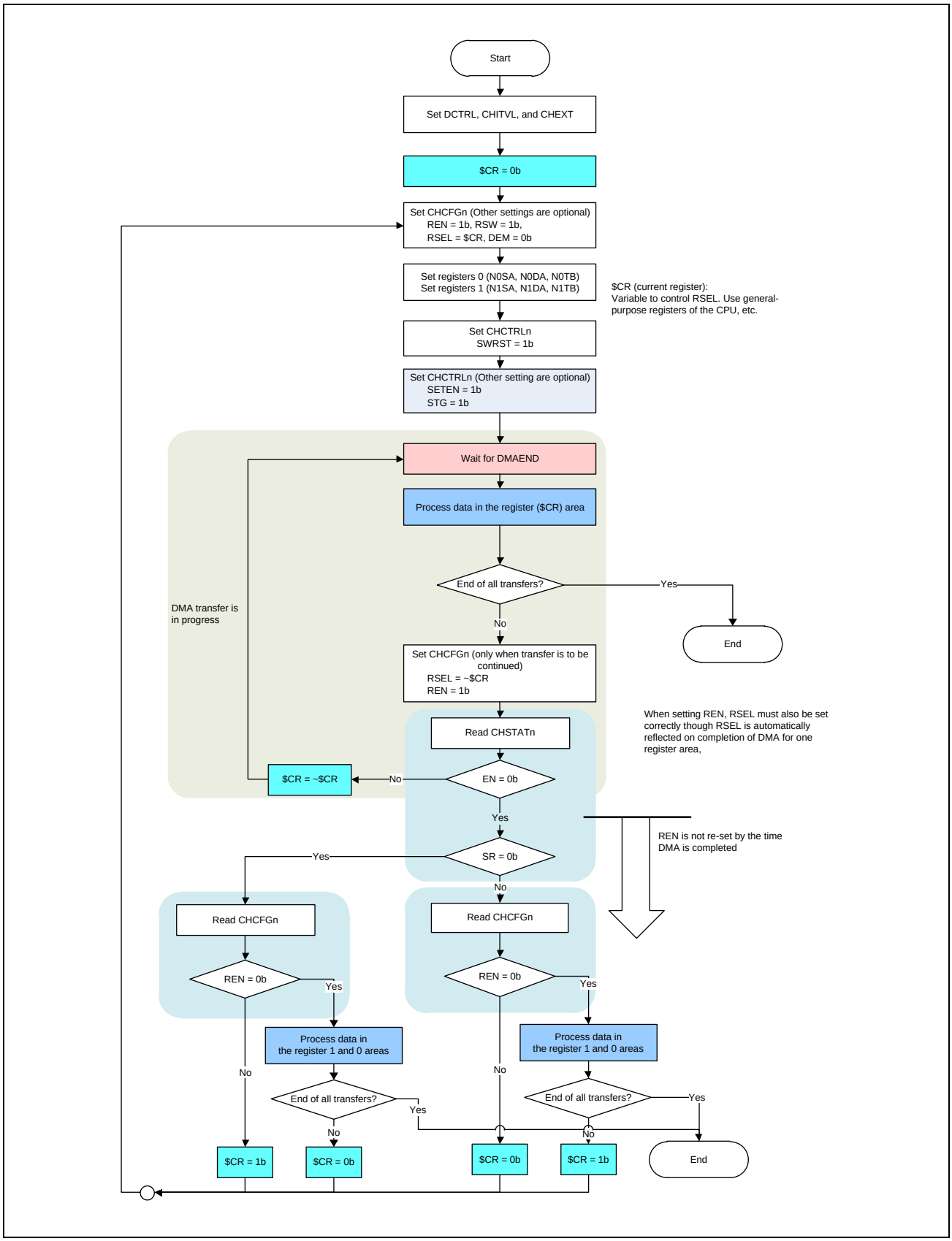


Figure 4.7-43 Example Setting 5

Supplementary Information

Store the first register sets (0 (N0SA, N0DA, N0TB) , 1 (N1SA, N1DA, N1TB)) to be transferred in general-purpose registers of the CPU (this register value called $CR for convenience).

Every time DMA transfer of one register set is completed (DMAEND is asserted), REN is automatically cleared to 0b. To continue transfer, REN of the CHCFG_n register must be set every time DMAEND is asserted. The same register also has the setting bit of RSEL, so this value must also be set correctly. To do this, use $CR.

In this mode, two Next registers are executed consecutively. However, if CLREN is not set by the time the DMA transaction is completed (next DMAEND is asserted), continuous execution stops. In such cases, check how far the transfer has proceeded by reading the SR and EN bits of the CHSTATn register and REN of the CHCFG_n register. To resume the transfer, follow the procedure in the above flowchart.

4.7.8 Points for Caution

4.7.8.1 DMA Transfer Requests by External Devices

When connecting the DREQ and DACK pins to external devices and requesting DMA transfers, use the ACK and REQ signals as level outputs.

4.7.8.2 Problem of the Interrupt Signal Overtaking DMA Transfer

4.7.8.2.1 Overview

This unit asserts the DMAINT interrupt signal on completion of transfer and this serves as a trigger for the CPU to start processing the data.

However, in cases such as where the device for writing is on another bus, the final data to be written may not yet have reached the device for writing at the time the DMAINT interrupt signal is asserted.

If the CPU accesses a device that has detected the DMAINT interrupt signal before the final data of the DMA transaction have been written to the device, it may be using some old data from before the writing.

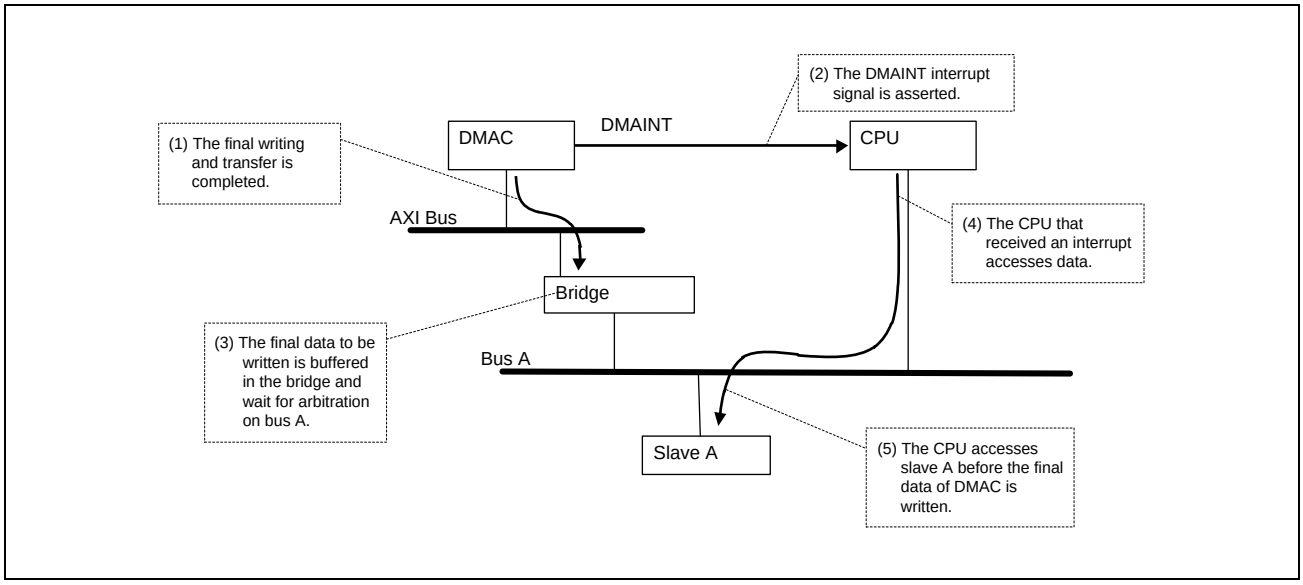


Figure 4.7-44 Example when Access by the CPU Overtakes Writing of the Final Data by the DMAC

4.7.8.2.2 Workaround

The following describes workaround for the above problems.

Setting AWCACHE as non-bufferable

Set the AWCACHE signal as non-bufferable for DMA transactions. By doing this, the above problems can be avoided because the DMAINT interrupt signal will only be asserted after the completion of writing to the device for access.

Setting all transfers to non-bufferable may reduce the efficiency of transfer. In such cases, divide up the register sets or descriptors by setting most of the transactions to bufferable for transfer, and then set only the last transfer to non-bufferable.

Exception)

The effect is not obtained if the device in the access path does not refer to AWCACHE.

SECTION 4 SYSTEM

4.8 Renesas Secure IP (RSIP-E50B)

The Renesas Security IP (RSIP-E50B) consists of the access management circuit, encryption engine, and random number generation circuit. In combination with the RSIP-E50B library, the RSIP-E50B can prevent eavesdropping (to maintain confidentiality), falsification of information (to ensure integrity), and impersonation (to verify authenticity).

Because key information required for encryption and decryption is stored only in the RSIP-E50B and all accesses from the outside can be blocked, RSIP-E50B enables building a more robust security system.

4.8.1 Overview

Table 4.8-1 summarizes the specifications of the RSIP-E50B. **Figure 4.8-1** shows a block diagram of the RSIP-E50B.

Table 4.8-1 Specifications of RSIP-E50B (1/2)

Item	Description
Access control	Access management circuit • In case of irregular access to the RSIP-E50B due to a tampered program or CPU runaway, this circuit blocks all subsequent accesses and stops data output from the RSIP-E50B.
Encryption engine	AES: Compliant with NIST FIPS PUB 197 • Key length: 128, 192, or 256 bits • Data block size: 128 bits • AES supports following block cipher mode – ECB, CBC, CTR: Compliant with NIST SP 800-38A – CCM: Compliant with NIST SP 800-38C – GCM: Compliant with NIST SP 800-38D – XTS: Compliant with IEEE 1619-2007 • AES supports following authentication algorithm – CMAC: Compliant with NIST SP 800-38B – GMAC: Compliant with NIST SP 800-38D AES does not support the KEY size of 192-bit in XTS, CMAC, GMAC, CCM, GCM and CMAC. ChaCha/Ploy • Block size: 512-bit • Key size: 256bit • ChaCha/Poly supports the following algorithms – ChaCha20: Compliant with RFC7539 – ChaCha20-Poly1305: Compliant with RFC7539
Random number generation	128-bit true random number generation circuit
Signature generation and verification	RSA • Maximum number of operable bits: 4224 bits • Signature generation, signature verification, public-key encryption, private-key decryption (Add RSA compliance specification(s)) ECC • Maximum number of operable bits: 576 bits • Signature generation, signature verification, key generation

Table 4.8-1 Specifications of RSIP-E50B (2/2)

Item	Description
Message digest computation	HASH • Block size: – 512-bit (SHA-1, SHA-224, SHA-256) – 1024-bit (SHA-512/224, SHA-512/256, SHA-384, SHA-512) – 1,152-bit (SHA3-224), 1,088-bit (SHA3-256), 832-bit (SHA3-384), 576-bit (SHA3-512), 1,344-bit (SHAKE128), 1,088-bit (SHAKE256) • Key size: 512-bit or less • HASH supports following secure hash algorithms – SHA-1, SHA-224, SHA-256, SHA-512/224, SHA-512/256, SHA-384 and SHA-512: Compliant with FIPS PUB 180-4 – SHA3-224, SHA3-256, SHA3-384, SHA3-512, SHAKE128, SHAKE256: Compliant with FIPS PUB 202 • HASH supports following message authentication algorithm – HMAC: Compliant with FIPS PUB 198
Hardware Unique Key	• Key derivation functions combine the Hardware Unique Key with the key generation information. The derived keys implement the key wrapping for user key secure storage. • The HUK uniqueness prevents the illicit cloning and copying of keys to another LSI of the LSI group. • The HUK itself is stored in wrapped (encrypted, non-plain) format in an isolated memory area. Therefore it is protected from illicit access and copy.
Application Key Management	• Wrapped keys are only valid within the RSIP-E50B.
Unique ID	• An LSI (Unique ID) is accessible from the access management circuit. • Key derivation functions combine the Unique ID with the key generation information. Such derived keys are used to unwrap the HUK within the RSIP-E50B boundary.
SPA/DPA Protections	• SPA/DPA protections can optionally be enabled for crypto engine processing.
Low power consumption	• Setting of the module stop state is possible.

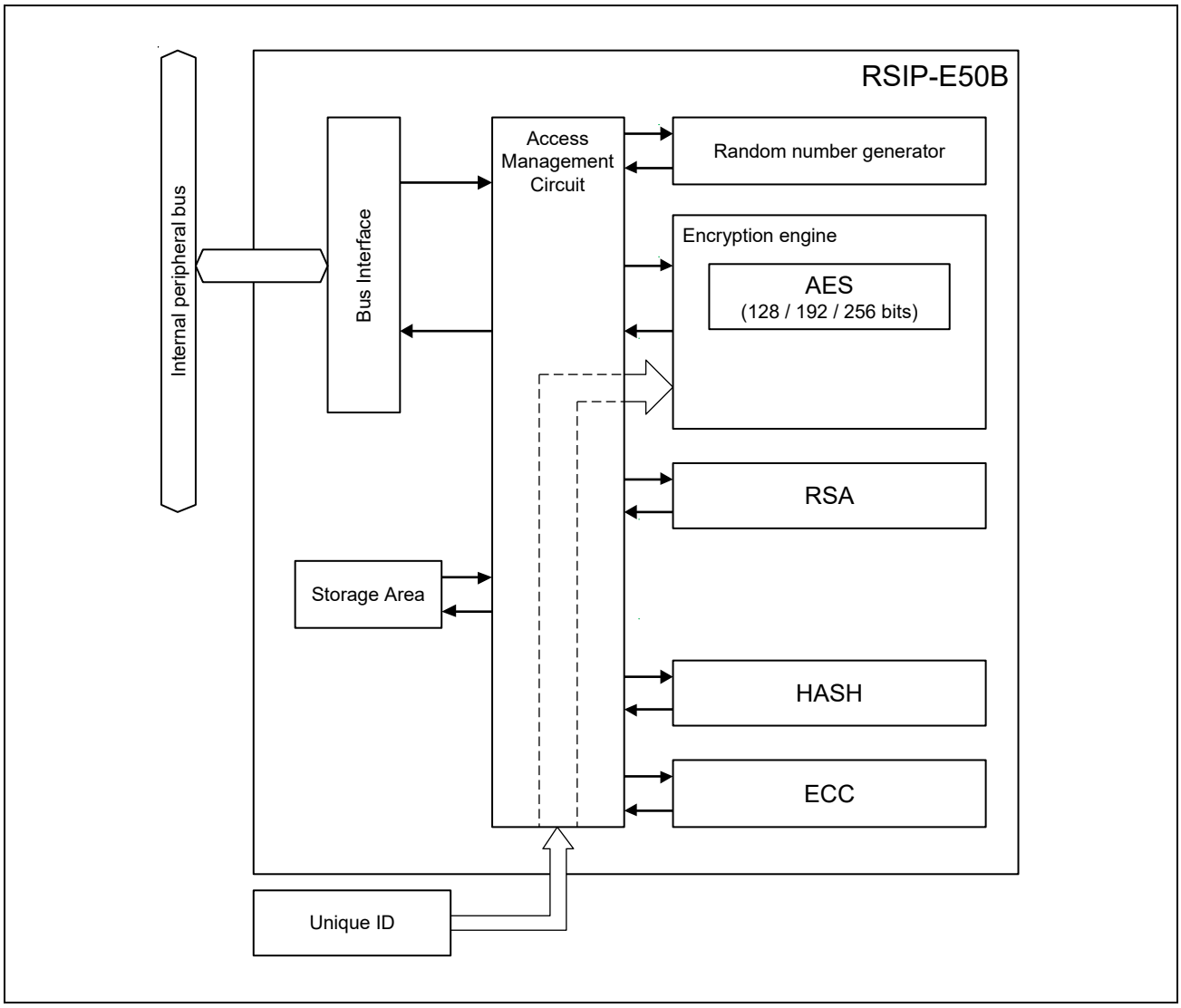


Figure 4.8-1 RSIP-E50B Block Diagram

4.8.2 Operation

4.8.2.1 Encryption Engine

Figure 4.8-2 shows conceptual block diagram of the encryption engine installed in the RSIP-E50B.

The encryption engine uses the input key information, and converts the plaintext data to ciphertext or ciphertext data to plaintext through the hardware.

The encryption/decryption process can be completed without exposing the key data and the process's intermediate data outside of the RSIP-E50B. This process is performed by the secure engine and storage area of the RSIP-E50B internally.

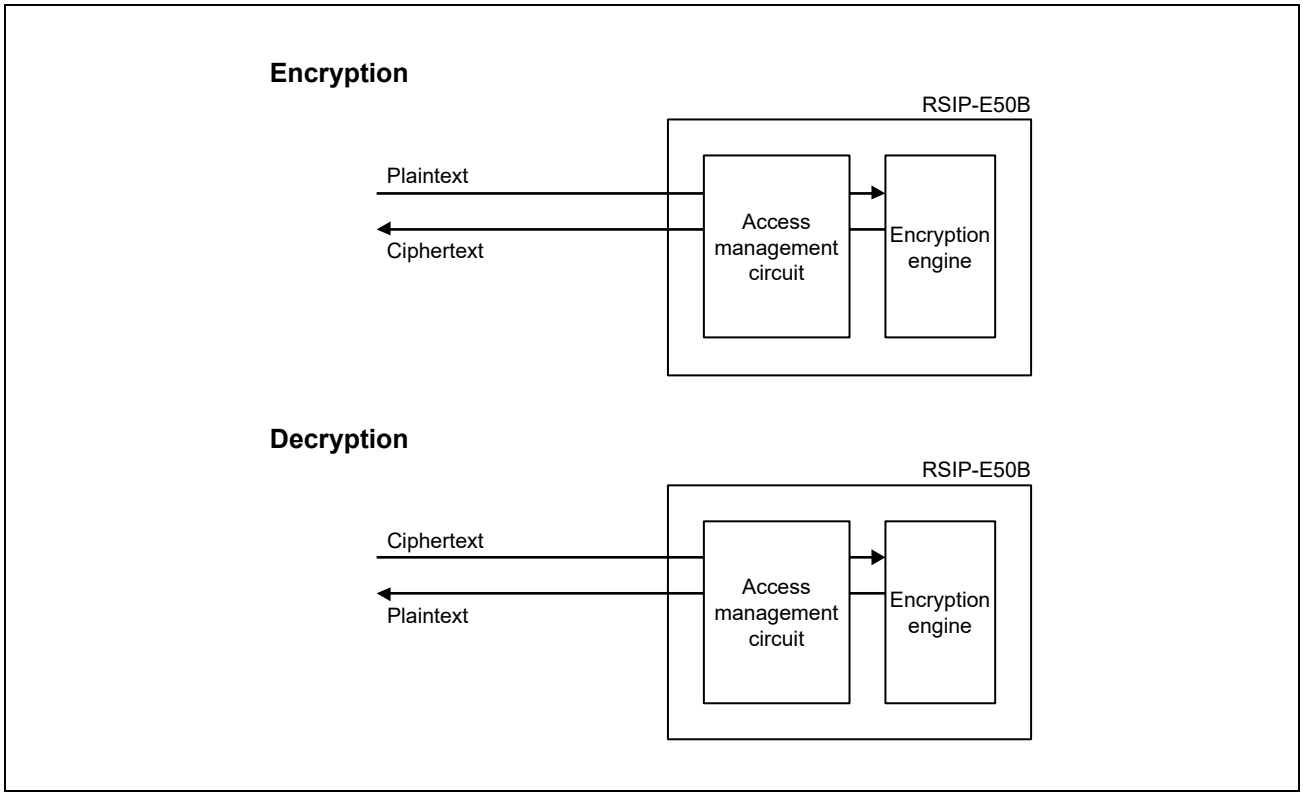


Figure 4.8-2 Encryption and Decryption Processing by Encryption Engine

4.8.2.2 Encryption and Decryption

Follow the procedure below to encrypt and decrypt the data:

(1) Input the key information to the RSIP-E50B.

(2) Input the target data to the RSIP-E50B. Plaintext data is converted to ciphertext and ciphertext data to plaintext.

(3) Read the converted data.

The encryption engine has input and output buffers, and can perform encryption/decryption in parallel with data input/output. **Figure 4.8-3** shows the encryption engine timing.

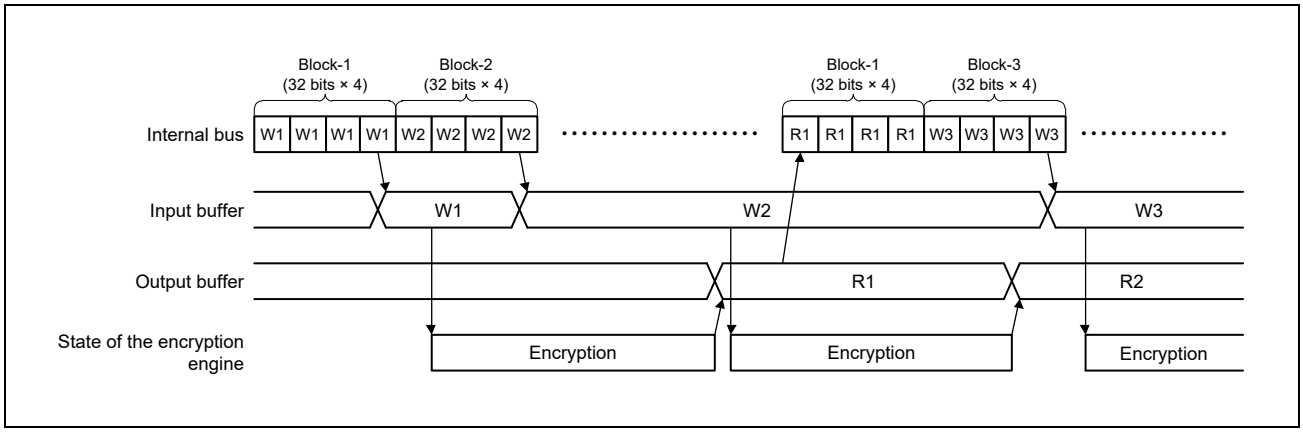


Figure 4.8-3 Encryption and Decryption Timing Diagram

4.8.3 Usage Notes

4.8.3.1 SPA/DPA Protections

SPA/DPA protections can optionally be enabled for crypto engine processing.

4.8.3.2 Renesas Secure IP(RSIP-E50B) Driver

Use of the RSIP-E50B requires the RSIP-E50B driver provided by Renesas Electronics. Please contact our sales office for information regarding the RSIP-E50B driver.

SECTION 4 SYSTEM

4.9 Debug Interface

4.9.1 Overview

This LSI has a debug interface for boundary scan function and debug support for CA55 and CM33.

4.9.1.1 Features

The functions of the debug interface of this LSI are shown below.

- Debug Interface
  - Support JTAG and SWD
- Debugger control function (DAP function)
  - Direct control of IP without going through CPU by switching Access Port (AP)
- Trace data support
  - CA55 trace output
  - CM33 trace output
  - Buffer function of trace data by ETF
  - Trace data output to the system bus by using ETR
- Boundary Scan
  - Connection test between this LSI and other LSIs on the user board
- Other functions
  - Interlocking operation of CA55, CM33, system counter (SYC), watchdog timer (WDT), general timer (GTM), and debug component by cross trigger
  - Support WDT counter stop control function
  - Add time information to trace data by Time Stamp (24MHz operation)

4.9.1.2 Block Diagram

Figure 4.9-1 shows the block diagram of the debug interface, and **Figure 4.9-2** shows the block diagram of the Core Sight System (CST).

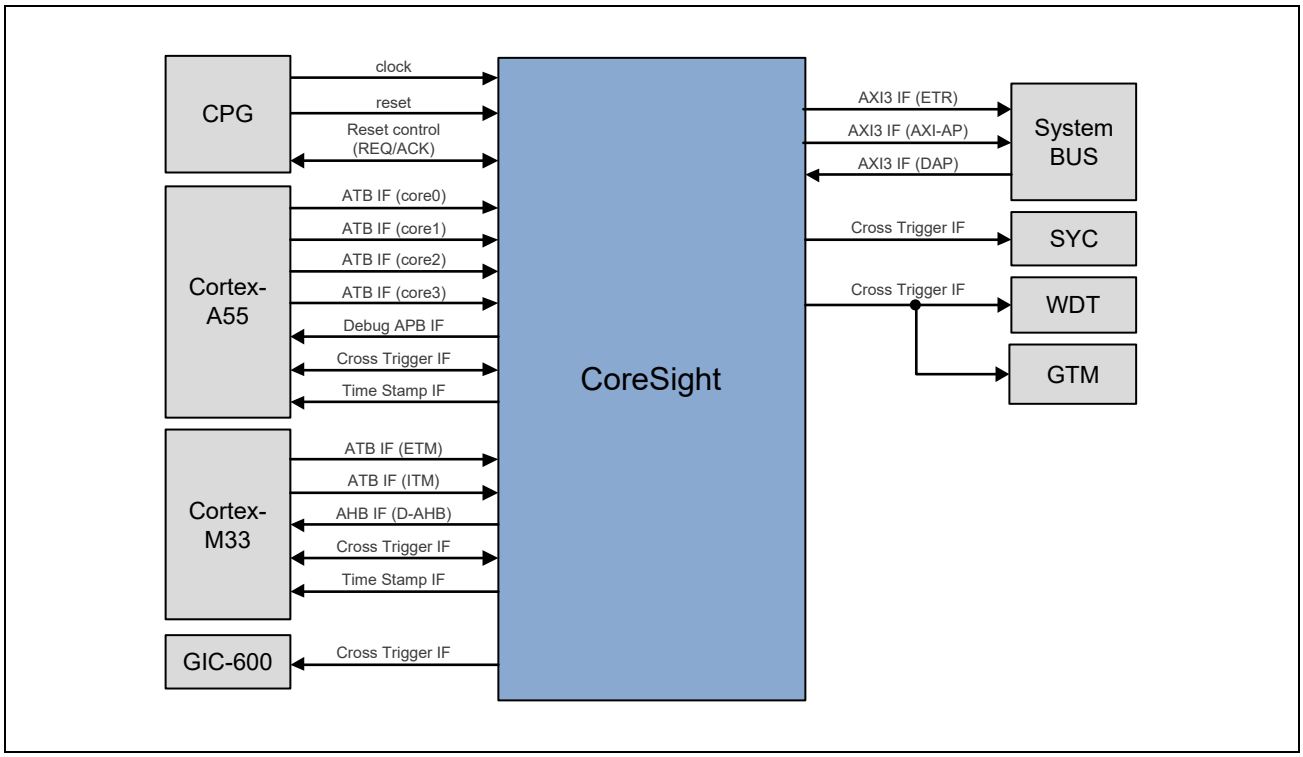


Figure 4.9-1 Debug System Block Diagram

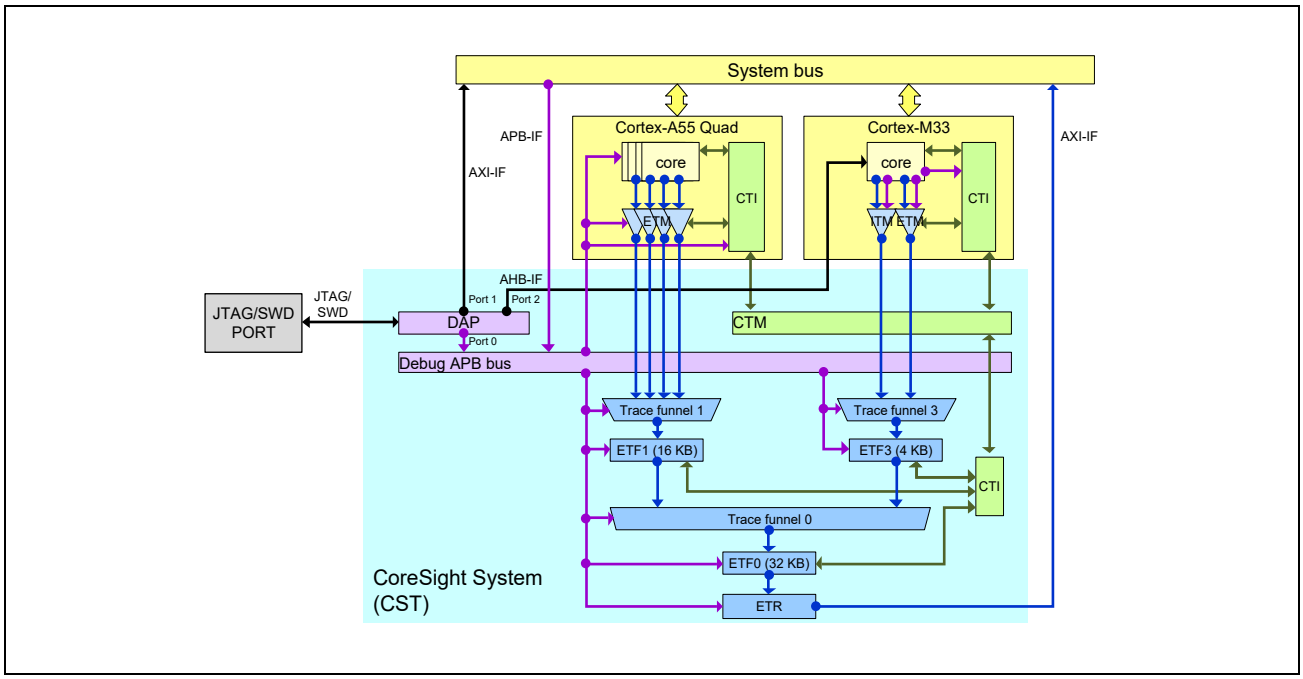


Figure 4.9-2 CoreSight System Block Diagram

- Debug Bus (APB)

  Bus used for controlling debug functions and acquiring debug data (register values and trace data in On-Chip Buffer)

  - DAP: Debug Access Port

- Cross Trigger Matrix (CTM)

  Functions required for co-debugging triggered by mutual events between CPU and Component

  - CTI: Cross Trigger Interface

- Trace Bus (ATB)

  A data bus which the trace data output by ETM flows

  - ETM: Embedded Trace Macrocell
  - ETR: Embedded Trace Router
  - ETF: Embedded Trace FIFO
  - Funnel: Trace data merge function

For details on ETM, ETR, and ETF, refer to “CoreSight Trace Memory Controller Technical Reference Manual”.

4.9.1.3 External Pins

■ **JTAG pins**

The debug interface supports JTAG IF and SWD IF. **Table 4.9-1** shows the JTAG pins and their functions.

Table 4.9-1 JTAG Pins

Pin Name	Input/Output	Function		Name
		JTAG IF	SWD IF	
TCK_SWCLK	Input	TCK	SWCLK	Test clock, Serial wire clock
TMS_SWDIO	Input, I/O	TMS	SWDIO	Test mode select, Serial wire data input/output
TDI	Input	TDI	—	Test data input
TDO	Output	TDO	—	Test data output
TRSTN	Input	TRST	—	Test reset

■ **MD_BOOT3 pins**

Table 4.9-2 shows the functions of the MD_BOOT3 pin.

Table 4.9-2 MD_BOOT3 Pins

Pin Name	Input/Output	Function
MD_BOOT3	Input	Switches between normal operation and debug operation. 0: Normal operation 1: Debug operation

The MD_BOOT3 value must be fixed before releasing the power-on reset (PRST#). **Table 4.9-3** shows the difference in operation between normal operation and debug operation.

Table 4.9-3 Difference in Operation by MD_BOOT3

	Normal Operation (MD_BOOT3 = 0)	Debug Operation (MD_BOOT3 = 1)	Related Module
Debug function module (CoreSight (other than DAP))	Boot impossible (Reset state)	Boot (Refer **4.9.5.1 Debug Related Reset** for reset condition and range)	CPG (Reset control)

■ BSCANP pins

Table 4.9-4 shows the functions of the BSCANP pin.

Contact a Renesas Electronics sales office for information regarding the boundary scan.

Table 4.9-4 BSCANP Pins

Pin Name	Input/Output	Function
BSCANP	Input	Switches between normal operation and boundary scan test mode. 0: Normal operation 1: Boundary scan test mode

Boundary scan test mode is a mode in which a connection test is performed between the terminals of this LSI and other ICs connected on the customer's board and is performed when BSCANP = 1.

The actual access is controlled by the 5 terminals of JTAG when BSCANP = 1.

Table 4.9-5 shows the operation modes of this LSI according to the values of the MD_BOOT3 pin and BSCANP pin.

Table 4.9-5 Operating Mode

Pin Name		
BSCANP	MD_BOOT3	Operating Mode
0	0	Normal operation
0	1	Debug operation
1	*	Boundary scan test mode

4.9.2 CST Registers

The CoreSight of this LSI has a 4 MB debug system address space. The base addresses for the respective channels are as follows.

Table 4.9-6 Address Space of the Debug System (1/2)

Address	Component*1
CoreSight Debug Components (Overall Address Space of this LSI)	
0_1F00_0000h to 0_1F00_FFFFh	CoreSight: ROM table
0_1F01_0000h to 0_1F01_FFFFh	CoreSight: Timestamp generator
0_1F02_0000h to 0_1F02_FFFFh	CoreSight: ETF0
0_1F03_0000h to 0_1F03_FFFFh	CoreSight: ETF1
0_1F04_0000h to 0_1F04_FFFFh	CoreSight: ETF2
0_1F05_0000h to 0_1F05_FFFFh	CoreSight: ETF3
0_1F06_0000h to 0_1F06_FFFFh	CoreSight: ETR
0_1F07_0000h to 0_1F07_FFFFh	CoreSight: Trace Funnel 0
0_1F08_0000h to 0_1F08_FFFFh	CoreSight: Trace Funnel 1
0_1F09_0000h to 0_1F09_FFFFh	CoreSight: Trace Funnel 2
0_1F0A_0000h to 0_1F0A_FFFFh	CoreSight: Trace Funnel 3
0_1F0B_0000h to 0_1F0B_FFFFh	CoreSight: CTI0
0_1F0C_0000h to 0_1F0C_FFFFh	CoreSight: CTI1
0_1F0D_0000h to 0_1F0F_FFFFh	CoreSight: Reserved
0_1F100000h to 0_1F3FFFFFh	Reserved
CA55 Debug Components (Overall Address Space of this LSI)	
0_1F40_0000h to 0_1F40_FFFFh	CA55: ROM table
0_1F41_0000h to 0_1F41_FFFFh	CA55: Core0 Debug
0_1F42_0000h to 0_1F42_FFFFh	CA55: Core0 CTI
0_1F43_0000h to 0_1F43_FFFFh	CA55: Core0 PMU
0_1F44_0000h to 0_1F44_FFFFh	CA55: Core0 ETM
0_1F45_0000h to 0_1F50_FFFFh	Reserved
0_1F51_0000h to 0_1F51_FFFFh	CA55: Core1 Debug
0_1F52_0000h to 0_1F52_FFFFh	CA55: Core1 CTI
0_1F53_0000h to 0_1F53_FFFFh	CA55: Core1 PMU
0_1F54_0000h to 0_1F54_FFFFh	CA55: Core1 ETM
0_1F55_0000h to 0_1F60_FFFFh	Reserved
0_1F61_0000h to 0_1F61_FFFFh	CA55: Core2 Debug
0_1F62_0000h to 0_1F62_FFFFh	CA55: Core2 CTI
0_1F63_0000h to 0_1F63_FFFFh	CA55: Core2 PMU
0_1F64_0000h to 0_1F64_FFFFh	CA55: Core2 ETM
0_1F65_0000h to 0_1F70_FFFFh	Reserved
0_1F71_0000h to 0_1F71_FFFFh	CA55: Core3 Debug
0_1F72_0000h to 0_1F72_FFFFh	CA55: Core3 CTI
0_1F73_0000h to 0_1F73_FFFFh	CA55: Core3 PMU
0_1F74_0000h to 0_1F74_FFFFh	CA55: Core3 ETM
0_1F75_0000h to 0_1F7F_FFFFh	CA55: Reserved

Table 4.9-6 Address Space of the Debug System (2/2)

Address	Component*1
CM33 Debug Components (CM33 Address Space)	
0_E000_0000h to 0_E000_0FFFh	CM33: ITM
0_E000_1000h to 0_E000_1FFFh	CM33: DWT
0_E000_2000h to 0_E000_2FFFh	CM33: FPB
0_E000_3000h to 0_E000_DFFFh	CM33: Reserved
0_E000_E000h to 0_E000_EFFFh	CM33: Secure SCS
0_E000_F000h to 0_E002_DFFFh	CM33: Reserved
0_E002_E000h to 0_E002_EFFFh	CM33: Non-secure SCS
0_E002_F000h to 0_E004_0FFFh	CM33: Reserved
0_E004_1000h to 0_E004_1FFFh	CM33: ETM
0_E004_2000h to 0_E004_2FFFh	CM33: CTI
0_E004_3000h to 0_E00F_EFFFh	CM33: Reserved
0_E00F_F000h to 0_E00F_FFFFh	CM33: ROM table

Note 1. Access to the reserved area is prohibited. Operation is not guaranteed when accessed.

NOTE

- The CoreSight Debug Components area and CA55 Debug Components area can be accessed via the system bus and APB-AP.
  APB-AP is connected to port-0 of DAP. For the port configuration of DAP, refer to **4.9.5.2 DAP**.
- The CM33 Debug Components area is accessible via AHB-AP.
  The AHB-AP for the area access is connected to port-2 of the DAP. For the port configuration of DAP, refer to **4.9.5.2 DAP**.
- Each component is placed on a 64-KB address boundary according to the ARMv8 Debug memory map. As an entity, the first 4 KB is valid.

4.9.2.1 List of Registers

The register is listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
ID Register	CST_BSID	0867_9447h	-	32

4.9.2.2 CST Register Descriptions

4.9.2.2.1 ID Register (CST_BSID)

The BSID register is a 32-bit register that cannot be accessed from the CPU. When the BSCANP pin is 1b, it can be read from the TDO pin by setting the IDCODE command on the TAP controller. It cannot be written.

Access Size : 32 bits
Address : -
Initial Value : x867_9447h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BSID[31:16]															
Initial Value	x	x	x	x	1	0	0	0	0	1	1	0	0	1	1	1
R/W	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSID[15:0]															
Initial Value	1	0	0	1	0	1	0	0	0	1	0	0	0	1	1	1
R/W	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BSID[31:0]	x867_9447h	-	This is the ID register specified in JTAG. The upper 4 bits (bit31-28) indicate the product revision.

4.9.2.2.2 Other Registers

For details on each register other than the BSID register, refer to “ARM CoreSight SoC-400 Technical Reference Manual” and “CoreSight Trace Memory Controller Technical Reference Manual”.

4.9.3 Debug Connection

4.9.3.1 Debug Connection Mode

There are the following three modes for the debug connection method of this LSI.

Authentication Mode	Method
No authentication	Allow access from the debugger without authentication (initial setting)
SEC authentication*1	Allow access from the debugger after authentication is OK. (authentication method is security authentication by security IP)
Authentication refusal*1	Block access from the debugger

Note 1. SEC authentication and Authentication refusal can only be authenticated with secure products.

For SEC authentication modes, enter the authentication code from the debugger is needed.

For information on the authentication function of secure products, contact a Renesas Electronics sales office.

4.9.4 Debug Access Range

The debug access range by CoreSight is for all slaves as well as CA55. For details, refer to **1.7 Internal Bus**.

4.9.5 Operation

4.9.5.1 Debug Related Reset

Refer to **4.4 Clock Pulse Generator (CPG)** for the debug-related resets and reset range.

4.9.5.2 DAP

DAP is equipped with SWJ-DP as Debug Port (DP) and APB-AP, AXI-AP, AHB-AP as Access Port (AP).

The DAP port numbers are shown in **Table 4.9-7** that describes the correspondence between the port number specified in the DP SELECT register and the AP, and its usage.

Table 4.9-7 DAP Port Number

Port number	AP	Usage
Port-0	APB-AP	Debug Component recognition and settings
Port-1	AXI-AP	System IP control (access to system bus)
Port-2	AHB-AP	CM33 settings

4.9.5.3 Trace

4.9.5.3.1 Data flow

In CoreSight, all trace data is aggregated once by Trace Funnel, and stored in ETF, or output to the system bus via ETR in the subsequent stage. Setting this port number to the Trace Funnel according to the route of the trace data is needed. The Trace Funnel port numbers are shown in **Table 4.9-8**.

Table 4.9-8 Trace Funnel Port Number

Trace Funnel Number	Port Number	Source Components
Funnel 0	Port-0	CA55 cluster
	Port-2	CM33 cluster
Funnel 1	Port-0	CA55: Core0 ETM
	Port-1	CA55: Core1 ETM
	Port-2	CA55: Core2 ETM
	Port-3	CA55: Core3 ETM
Funnel 3	Port-0	CM33: ITM
	Port-1	CM33: ETM

4.9.5.3.2 Timestamp

If time information want to be included in the trace data, the Timestamp Generator can be used. The timestamp replicator port numbers are shown in **Table 4.9-9**. The Timestamp Generator runs on a 24-MHz clock.

Table 4.9-9 Timestamp Replicator Port Number

Port Number	Target Components
Port-0	CA55: Core0/1/2/3 ETM
Port-2	CM33

4.9.5.4 Cross Trigger

The CTI (Cross Trigger Interface) implemented in CoreSight, CA55, and CM33 is used to communicate debug events.

4.9.5.4.1 CoreSight cross trigger connection

CoreSight has two CTIs. The connection specifications are shown in **Table 4.9-10** to **Table 4.9-13**.

Table 4.9-10 CoreSight CTI0 Trigger Input

Trigger Input Bit	Source Components	Description
[7]	(Unused)	—
[6]	(Unused)	—
[5]	(Unused)	—
[4]	(Unused)	—
[3]	ETR	ETR Event Output 1 (FULL)
[2]	ETR	ETR Event Output 0 (ACQCOMP)
[1]	ETF0	ETF0 Event Output 1 (FULL)
[0]	ETF0	ETF0 Event Output 0 (ACQCOMP)

Table 4.9-11 CoreSight CTI0 Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	SYC	SYC Event Input 1 (RESTART REQ/ACK)
[6]	SYC	SYC Event Input 0 (HALT REQ)
[5]	WDT	WDT/GTM Event Input 1 (RESTART REQ/ACK)
[4]	WDT	WDT/GTM Event Input 0 (HALT REQ)
[3]	ETR	ETR Event Input 1 (TRIGIN)
[2]	ETR	ETR Event Input 0 (FLUSHIN)
[1]	ETF0	ETF0 Event Input 1 (TRIGIN)
[0]	ETF0	ETF0 Event Input 0 (FLUSHIN)

Table 4.9-12 CoreSight CTI1 Trigger Input

Trigger Input Bit	Source Components	Description
[7]	ETF3	ETF3 Event Output 1 (FULL)
[6]	ETF3	ETF3 Event Output 0 (ACQCOMP)
[5]	(Unused)	—
[4]	(Unused)	—
[3]	ETF1	ETF1 Event Output 1 (FULL)
[2]	ETF1	ETF1 Event Output 0 (ACQCOMP)
[1]	(Unused)	—
[0]	(Unused)	—

Table 4.9-13 CoreSight CTI1 trigger output

Trigger Output Bit	Destination Components	Description
[7]	ETF3	ETF3 Event Input 1 (TRIGIN)
[6]	ETF3	ETF3 Event Input 0 (FLUSHIN)
[5]	(Unused)	—
[4]	(Unused)	—
[3]	ETF1	ETF1 Event Input 1 (TRIGIN)
[2]	ETF1	ETF1 Event Input 0 (FLUSHIN)
[1]	CA55	CA55 Event Input (PMUSNAPSHOTREQ/PMUSNAPSHOTACK)
[0]	GIC-600	GIC-600 Event Input (sample_req/sample_ack)

4.9.5.4.2 CA55 cross trigger connection

The CA55 has four CTIs (CA55 Core0 CTI, CA55 Core1 CTI, CA55 Core2 CTI, and CA55 Core3 CTI). The connection specifications for each CTI are shown in **Table 4.9-14** to **Table 4.9-21**.

Table 4.9-14 CA55 CTI0 trigger input

Trigger Input Bit	Source Components	Description
[7]	CA55: Core0 ETM	ETM Trace Output 3 trigger event
[6]	CA55: Core0 ETM	ETM Trace Output 2 trigger event
[5]	CA55: Core0 ETM	ETM Trace Output 1 trigger event
[4]	CA55: Core0 ETM	ETM Trace Output 0 trigger event
[3]	(Unused)	—
[2]	CA55: Core0 PE	Profiling sample trigger event
[1]	CA55: Core0 PE	Performance Monitors Overflow trigger event
[0]	CA55: Core0 PE	Cross-halt trigger event

Table 4.9-15 CA55 CTI0 Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	CA55: Core0 ETM	Generic Trace External Input 3 trigger event
[6]	CA55: Core0 ETM	Generic Trace External Input 2 trigger event
[5]	CA55: Core0 ETM	Generic Trace External Input 1 trigger event
[4]	CA55: Core0 ETM	Generic Trace External Input 0 trigger event
[3]	(Unused)	—
[2]	External (GIC-600)	Generic CTI Interrupt trigger event
[1]	CA55: Core0 PE	Restart Request trigger event
[0]	CA55: Core0 PE	Debug Request trigger event

Table 4.9-16 CA55 CTI1 Trigger Input

Trigger Input Bit	Source Components	Description
[7]	CA55: Core1 ETM	ETM Trace Output 3 trigger event
[6]	CA55: Core1 ETM	ETM Trace Output 2 trigger event
[5]	CA55: Core1 ETM	ETM Trace Output 1 trigger event
[4]	CA55: Core1 ETM	ETM Trace Output 0 trigger event
[3]	(Unused)	—
[2]	CA55: Core1 PE	Profiling sample trigger event
[1]	CA55: Core1 PE	Performance Monitors Overflow trigger event
[0]	CA55: Core1 PE	Cross-halt trigger event

Table 4.9-17 CA55 CTI1 Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	CA55: Core1 ETM	Generic Trace External Input 3 trigger event
[6]	CA55: Core1 ETM	Generic Trace External Input 2 trigger event
[5]	CA55: Core1 ETM	Generic Trace External Input 1 trigger event
[4]	CA55: Core1 ETM	Generic Trace External Input 0 trigger event
[3]	(Unused)	—
[2]	External (GIC-600)	Generic CTI Interrupt trigger event
[1]	CA55: Core1 PE	Restart Request trigger event
[0]	CA55: Core1 PE	Debug Request trigger event

Table 4.9-18 CA55 CTI2 Trigger Input

Trigger Input Bit	Source Components	Description
[7]	CA55: Core2 ETM	ETM Trace Output 3 trigger event
[6]	CA55: Core2 ETM	ETM Trace Output 2 trigger event
[5]	CA55: Core2 ETM	ETM Trace Output 1 trigger event
[4]	CA55: Core2 ETM	ETM Trace Output 0 trigger event
[3]	(Unused)	—
[2]	CA55: Core2 PE	Profiling sample trigger event
[1]	CA55: Core2 PE	Performance Monitors Overflow trigger event
[0]	CA55: Core2 PE	Cross-halt trigger event

Table 4.9-19 CA55 CTI2 Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	CA55: Core2 ETM	Generic Trace External Input 3 trigger event
[6]	CA55: Core2 ETM	Generic Trace External Input 2 trigger event
[5]	CA55: Core2 ETM	Generic Trace External Input 1 trigger event
[4]	CA55: Core2 ETM	Generic Trace External Input 0 trigger event
[3]	(Unused)	—
[2]	External (GIC-600)	Generic CTI Interrupt trigger event
[1]	CA55: Core2 PE	Restart Request trigger event
[0]	CA55: Core2 PE	Debug Request trigger event

Table 4.9-20 CA55 CTI3 Trigger Input

Trigger Input Bit	Source Components	Description
[7]	CA55: Core3 ETM	ETM Trace Output 3 trigger event
[6]	CA55: Core3 ETM	ETM Trace Output 2 trigger event
[5]	CA55: Core3 ETM	ETM Trace Output 1 trigger event
[4]	CA55: Core3 ETM	ETM Trace Output 0 trigger event
[3]	(Unused)	—
[2]	CA55: Core3 PE	Profiling sample trigger event
[1]	CA55: Core3 PE	Performance Monitors Overflow trigger event
[0]	CA55: Core3 PE	Cross-halt trigger event

Table 4.9-21 CA55 CTI3 Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	CA55: Core3 ETM	Generic Trace External Input 3 trigger event
[6]	CA55: Core3 ETM	Generic Trace External Input 2 trigger event
[5]	CA55: Core3 ETM	Generic Trace External Input 1 trigger event
[4]	CA55: Core3 ETM	Generic Trace External Input 0 trigger event
[3]	(Unused)	—
[2]	External (GIC-600)	Generic CTI Interrupt trigger event
[1]	CA55: Core3 PE	Restart Request trigger event
[0]	CA55: Core3 PE	Debug Request trigger event

4.9.5.4.3 CM33 cross trigger connection

The CM33 has one CTI (CM33 CTI). The connection specifications are shown in **Table 4.9-22** and **Table 4.9-23**.

Table 4.9-22 CM33 CTI Trigger Input

Trigger Input Bit	Source Components	Description
[7]	(Unused)	—
[6]	(Unused)	—
[5]	CM33: ETM	ETM Event Output 1
[4]	CM33: ETM/Processor	ETM Event Output 0 or Comparator Output 3
[3]	CM33: Processor	DWT Comparator Output 2
[2]	CM33: Processor	DWT Comparator Output 1
[1]	CM33: Processor	DWT Comparator Output 0
[0]	CM33: Processor	Processor Halted

Table 4.9-23 CM33 CTI Trigger Output

Trigger Output Bit	Destination Components	Description
[7]	CM33: ETM	ETM Event Input 3
[6]	CM33: ETM	ETM Event Input 2
[5]	CM33: ETM	ETM Event Input 1
[4]	CM33: ETM	ETM Event Input 0
[3]	CM33: System	Interrupt request 1
[2]	CM33: System	Interrupt request 0
[1]	CM33: Processor	Processor Restart
[0]	CM33: Processor	Processor debug request

4.9.5.5 WDT Counter Stop Control

This LSI has a WDT counter stop control function linked with a debug event to prevent an unintended reset by WDT during debug operation. When a debug event occurs, CoreSight asserts the WDT counter stop request signal, controls the WDT counter stop signal (CNTSTOP), and stops counting. The target WDT is all WDT. For details on the WDT count stop function, refer to **4.3 System Controller (SYS)**.

4.9.5.6 SYC Counter Stop Control

This LSI has a SYC counter stop control function linked with debug events to prevent unintended SYC counts during debug operation. When a debug event occurs, CoreSight asserts a SYC counter stop request signal to stop the SYC count.

SECTION 5 TIMER

5.1 Timer Overview

This sections describes the timer unit of this LSI. For details, refer to the section of each unit.

■ System Counter (SYC) (See 5.2)

SYC provides count values for CA55 and GE3D.

■ Realtime Clock (RTC) (See 5.3)

■ Watchdog Timer (WDT) (See 5.4)

This LSI equips 4ch of WDT. Available for CA55 (All core), and CM33 respectively.

■ General Timer (GTM) (See 5.5)

This LSI equips 8ch of GTM.

■ Compare Match Timer W (CMTW) (See 5.6)

This LSI equips 8ch of CMTW.

This unit can input/ouput interrupt signals to/from external devices using external pins.

■ General Purpose Timer (GPT) (See 5.7)

This LSI equips 16ch of GPT. This unit can output PWM waveform in PWM mode.

Enabling control of each GPT external pin is possible by using POEG. For details on POEG, see **5.8 Port Output Enable for GPT (POEG)**.

SECTION 5 TIMER

5.2 System Counter (SYC)

The SYC generates the count value that is used by the GE3D and the generic timer built into the Cortex-A55.

It uses the timestamp generator, which is part of the Arm CoreSight SoC-400, to generate a 64-bit count value. For more information on the timestamp generator, refer to *the ARM CoreSight SoC-400 Technical Reference Manual*.

5.2.1 Overview

5.2.1.1 Features

- Count by 24 MHz clock (SYC_0_CNT_CLK)
- 64-bit gray code counter value generation
  - Convert the value generated by timestamp generator to Gray code and output it.
- Access control
  - Access control for the two interfaces of the timestamp generator
  - Control secure access/non-secure access
- Halt on debug function
  - Counting can be stopped/restarted during debugging as requested by CoreSight

5.2.1.2 Block Diagram of SYC

The SYC connection diagram is shown in **Figure 5.2-1** and the SYC block diagram is shown in **Figure 5.2-2**.

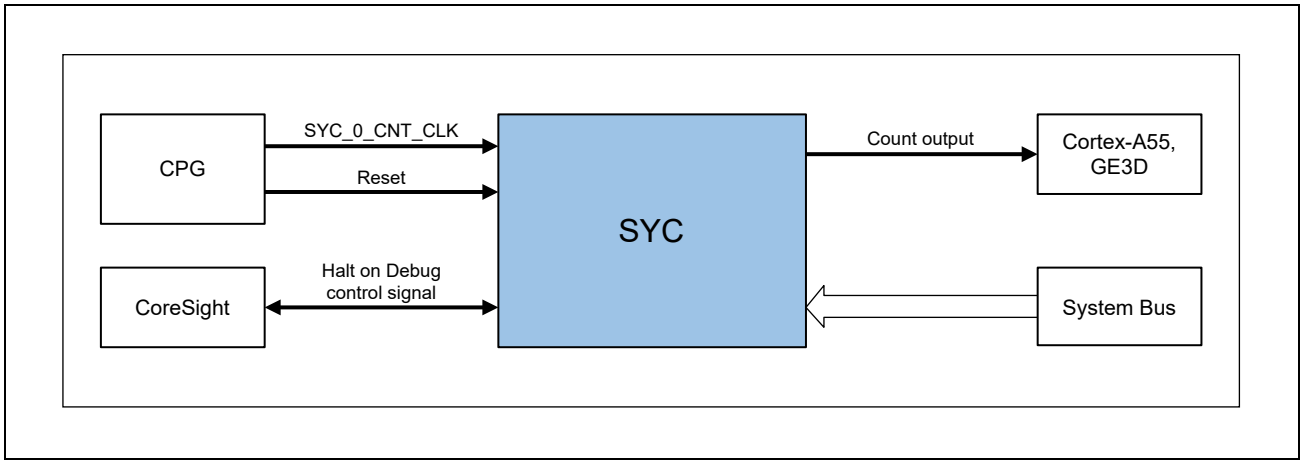


Figure 5.2-1 SYC Connection Diagram

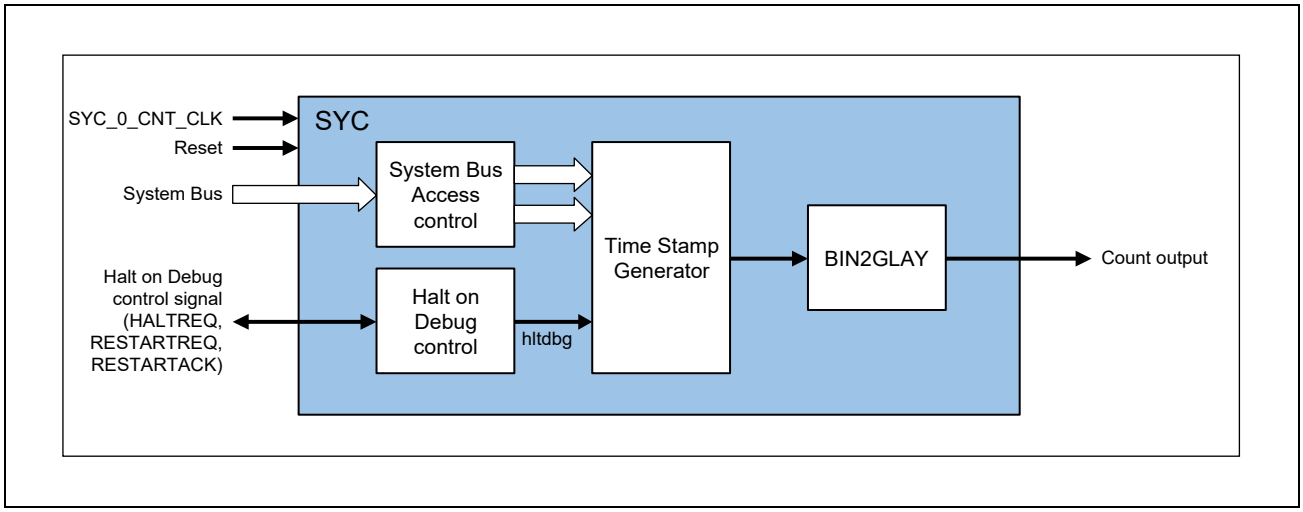


Figure 5.2-2 SYC Block Diagram

5.2.2 Registers

Table 5.2-2 shows the SYC address space. SYC has an address space of 8 KB. The SYC address space is offset from the base address. The base address of SYC is as follows.

Table 5.2-1 Register Base Addresses

Base Address Name	Base Address
<SYC0_base>	0_1401_0000h (5401_0000h*1, 4401_0C00h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

Table 5.2-2 SYC Address Space

Offset Address	Region Name	Access Target
0000h to 0FFFh	PSELCTRL region	PSELCTRL region & PSELCTRL region Management registers
1000h to 1FFFh	PSELREAD region	PSELREAD region & PSELREAD region Management registers

Note: Secure access and non-secure access are possible for the PSELCTRL region and PSELREAD region.

For more information on SYC registers, refer ARM CoreSight SoC-400 Technical Reference Manual 3.10 Timestamp generator.

5.2.3 Operation

5.2.3.1 System Bus Access Control

Access control is performed from a system bus interface of the SYC to two APB interfaces of the timestamp generator.

The PSELCTRL area can be accessed when accessing the first half 4 KB of the 8 KB address area of SYC, and the PSELREAD area can be accessed when accessing the last 4 KB.

The PSELCTRL area and PSELREAD area accept secure access and non-secure access.

5.2.3.2 Halt on Debug Function

SYC supports the Halt on Debug function included in timestamp generator. When HALTREQ is asserted from CoreSight, the counting operation of timestamp generator is stopped.

When RESTARTREQ is asserted from CoreSight, the counting operation of timestamp generator is restarted.

HALTREQ and RESTARTREQ are controlled by CoreSight's CTI (Cross Trigger Interface).

NOTE

This LSI provides the common count value generated by SYC to Cortex-A55 and GE3D. The stop/restart control of the counter by the Halt on Debug function is also performed simultaneously for Cortex-A55 and GE3D.

SECTION 5 TIMER

5.3 Realtime Clock (RTC)

In this section, “PCLK” is used to refer to RTC_0_clk_rtc (clk_rtc).

5.3.1 Overview

The RTC (Real time Clock) has two types of counting mode: calendar count mode and binary count mode. They are used by switching the register settings.

In calendar count mode, the RTC uses a 100-year calendar from 2000 to 2099 and automatically adjusts dates for leap years.

In binary count mode, the RTC does not count in terms of years, months, dates, day-of-week, hours, or minutes; it counts seconds, and retains the information as a serial value. This mode can be used for calendars other than the Gregorian calendar.

The RTC uses the 128-Hz clock which is acquired by the count source divided by the prescaler as the reference clock.

Year, month, date, day-of-week, a.m./p.m. (in 12-hour mode), hour, minute, second, or 32-bit binary is counted in 1/128 second units.

Table 5.3-1 lists the specifications of the RTC, **Figure 5.3-1** shows a block diagram of the RTC, and **Table 5.3-2** shows the pin configuration of the RTC.

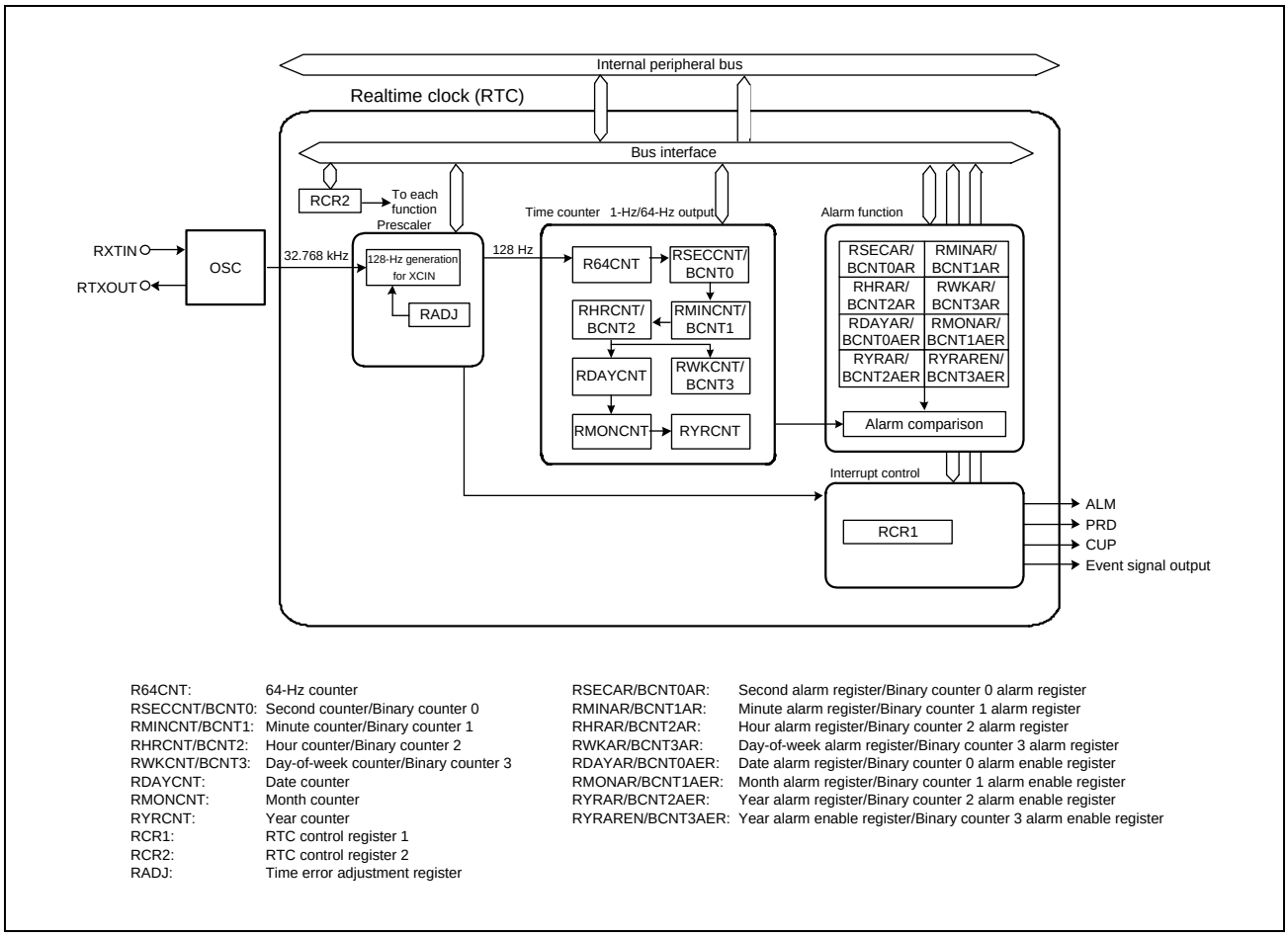
Table 5.3-1 RTC Specifications (1/2)

Item	Description
Count mode	Calendar count mode/binary count mode
Count source*1	Real time Clock (RXTIN)
Clock and calendar functions	• Calendar count mode – Year, month, date, day-of-week, hour, minute, second are counted, BCD display – 12 hours/24 hours mode switching function – 30 seconds adjustment function (a number less than 30 is rounded down to 00 seconds, and 30 seconds or more are rounded up to one minute) – Automatic adjustment function for leap years • Binary count mode – Count seconds in 32 bits, binary display • Common to both modes – Start/stop function – The sub-second digit is displayed in binary units (1 Hz, 2 Hz, 4 Hz, 8 Hz, 16 Hz, 32 Hz, or 64 Hz). – Clock error correction function

Table 5.3-1 RTC Specifications (2/2)

Item	Description
Interrupts	• Alarm interrupt (ALM) As an alarm interrupt condition, selectable which of the below is compared with: – Calendar count mode: Year, month, date, day-of-week, hour, minute, or second can be selected – Binary count mode: Each bit of the 32-bit binary counter • Periodic interrupt (PRD) – 2 seconds, 1 second, 1/2 seconds, 1/4 seconds, 1/8 seconds, 1/16 seconds, 1/32 seconds, 1/64 seconds, or 1/128 seconds can be selected as an interrupt period. • Carry interrupt (CUP) An interrupt is generated at either of the following timings: – When a carry from the 64-Hz counter to the second counter is generated. – When the 64-Hz counter is changed and the R64CNT register is read at the same time.
Event link function	• Alarm event output • Periodic event output • Carry event output

Note 1. Satisfy the frequency of the peripheral module clock (PCLK) ≥ the frequency of the count source.



R64CNT: 64-Hz counter
RSECCNT/BCNT0: Second counter/Binary counter 0
RMINCNT/BCNT1: Minute counter/Binary counter 1
RHRCNT/BCNT2: Hour counter/Binary counter 2
RWKCNT/BCNT3: Day-of-week counter/Binary counter 3
RDAYCNT: Date counter
RMONCNT: Month counter
RYRCNT: Year counter
RCR1: RTC control register 1
RCR2: RTC control register 2
RADJ: Time error adjustment register

RSECAR/BCNT0AR: Second alarm register/Binary counter 0 alarm register
RMINAR/BCNT1AR: Minute alarm register/Binary counter 1 alarm register
RHRAR/BCNT2AR: Hour alarm register/Binary counter 2 alarm register
RWKAR/BCNT3AR: Day-of-week alarm register/Binary counter 3 alarm register
RDAYAR/BCNT0AER: Date alarm register/Binary counter 0 alarm enable register
RMONAR/BCNT1AER: Month alarm register/Binary counter 1 alarm enable register
RYRAR/BCNT2AER: Year alarm register/Binary counter 2 alarm enable register
RYRAREN/BCNT3AER: Year alarm enable register/Binary counter 3 alarm enable register

Figure 5.3-1 Block Diagram of RTC

Table 5.3-2 Pin Configuration of RTC

Pin Name	Input/Output	Function
RTXIN	Input	Connect a 32.768-kHz crystal to these pins.
RTXOUT	Output	

5.3.2 Registers

When writing to or reading from RTC registers, do so in accordance with **5.3.6.4 Notes on Writing to and Reading from Registers**.

If the value in an RTC register after a reset is given as x (undefined bits) in the list, it is not initialized by a reset. When RTC enters the reset state or a low power consumption state during counting operations (i.e. while the RCR2.START bit is 1), the year, month, day of the week, date, hours, minutes, seconds, and 64-Hz counters continue to operate. Note that a reset generated during writing to or updating of a register might destroy the register value. In addition, do not allow the chip to enter software standby mode immediately after setting any of these registers. For details, refer to **5.3.6.3 Transitions to Low Power Consumption Modes after Setting Registers**.

The register addresses of RTC are given as offsets from the individual base addresses <RTC_base> and <RTC_Read_Only_base>. The register base addresses of each RTC are listed in the following table.

Table 5.3-3 Register Base Addresses

Base Address Name	Base Address
<RTC_base>	0_11C0_0800 (51C0_0800*1, 41C0_0800*2)
<RTC_Read_Only_base>*3	0_11C0_0C00 (51C0_0C00*1, 41C0_0C00*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

Note 3. The following registers are only readable.
R64CNT, RSECCNT/BCNT0, RMINCNT/BCNT1, RHRCNT/BCNT2, RWKCNT/BCNT3, RDAYCNT, RMONCNT,RYRCNT

5.3.2.1 List of Registers

The RTC registers and the memory addresses are listed in the following table.

For the actual addresses, the offset values indicated in this table are added to the base addresses.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
64-Hz Counter	RTC_R64CNT	xxh	00h	8
Reserve	-	-	01h	-
Second Counter/Binary Counter 0	RTC_RSECCNT/RTC_BCNT0	xxh	02h	8
Reserve	-	-	03h	-
Minute Counter/Binary Counter 1	RTC_RMINCNT/RTC_BCNT1	xxh	04h	8
Reserve	-	-	05h	-
Hour Counter/Binary Counter 2	RTC_RHRCNT/RTC_BCNT2	xxh	06h	8
Reserve	-	-	07h	-
Day-of-Week Counter/Binary Counter 3	RTC_RWKCNT/RTC_BCNT3	0xh	08h	8
Reserve	-	-	09h	-
Date Counter	RTC_RDAYCNT	xxh	0Ah	8
Reserve	-	-	0Bh	-
Month Counter	RTC_RMONCNT	xxh	0Ch	8
Reserve	-	-	0Dh	-
Year Counter	RTC_RYRCNT	00xxh	0Eh	16
Second Alarm Register /Binary Counter 0 Alarm Register	RTC_RSECAR /RTC_BCNT0AR	xxh	10h	8
Reserve	-	-	11h	-
Minute Alarm Register /Binary Counter 1 Alarm Register	RTC_RMINAR /RTC_BCNT1AR	xxh	12h	8
Reserve	-	-	13h	-
Hour Alarm Register /Binary Counter 2 Alarm Register	RTC_RHRAR /RTC_BCNT2AR	xxh	14h	8
Reserve	-	-	15h	-
Day-of-Week Alarm Register /Binary Counter 3 Alarm Register	RTC_RWKAR /RTC_BCNT3AR	xxh	16h	8
Reserve	-	-	17h	-
Date Alarm Register /Binary Counter 0 Alarm Enable Register	RTC_RDAYAR /RTC_BCNT0AER	xxh	18h	8
Reserve	-	-	19h	-
Month Alarm Register /Binary Counter 1 Alarm Enable Register	RTC_RMONAR /RTC_BCNT1AER	xxh	1Ah	8
Reserve	-	-	1Bh	-
Year Alarm Register /Binary Counter 2 Alarm Enable Register	RTC_RYRAR /RTC_BCNT2AER	00xxh	1Ch	16
Year Alarm Enable Register /Binary Counter 3 Alarm Enable Register	RTC_RYRAREN /RTC_BCNT3AER	x0h	1Eh	8
Reserve	-	-	1Fh	-
RTC Statas Register	RTC_RSR	0xh	20h	8
Reserve	-	-	21h	-
RTC Control Register 1	RTC_RCR1	xxh	22h	8
Reserve	-	-	23h	-
RTC Control Register 2	RTC_RCR2	xxh	24h	8
Reserve	-	-	25h to 2Dh	-
Time Error Adjustment Register	RTC_RADJ	xxh	2Eh	8

5.3.2.2 Register Description

The prefix (RTC_) of the register names is omitted in this and subsequent sections.

5.3.2.2.1 64-Hz Counter (RTC_R64CNT)

The R64CNT counter is used in both calendar count mode and in binary count mode.

The 64-Hz counter (R64CNT) generates a period of one second by counting the 128-Hz reference clock.

The state in the sub-second range can be confirmed by reading this counter.

This counter is set to 00h by an RTC software reset or executing 30-second adjustment.

To read this counter, follow the procedure in **5.3.3.5 Reading 64-Hz Counter and Time**.

Access Size : 8 bits

Address : <RTC_base> + 00h
<RTC_Read_Only_base> + 00h

Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	-	F1HZ	F2HZ	F4HZ	F8HZ	F16HZ	F32HZ	F64HZ
Initial Value	0	x	x	x	x	x	x	x
RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	RW	Description
7	-	0h	R	Reserved This bit is always read as 0b. The written value will be ignored.
6	F1HZ	x	R	Indicate the state between 1 Hz and 64 Hz of the sub-second digit.
5	F2HZ	x	R	Indicate the state between 1 Hz and 64 Hz of the sub-second digit.
4	F4HZ	x	R	Indicate the state between 1 Hz and 64 Hz of the sub-second digit.
3	F8HZ	x	R	Indicate the state between 1 Hz and 64 Hz of the sub-second digit.
2	F16HZ	x	R	Indicate the state between 1 Hz and 64 Hz of the sub-second digit.
1	F32HZ	x	R	Indicate the state between 1 Hz and 64 Hz of the sub-second digit.
0	F64HZ	x	R	Indicate the state between 1 Hz and 64 Hz of the sub-second digit.

x: Undefined value

5.3.2.2.2 Second Counter (RTC_RSECCNT)/Binary Counter 0 (RTC_BCNT0)

a. In calendar count mode

The RSECCNT counter is used for setting and counting the BCD-coded second value. It counts carries generated once per second in the 64-Hz counter.

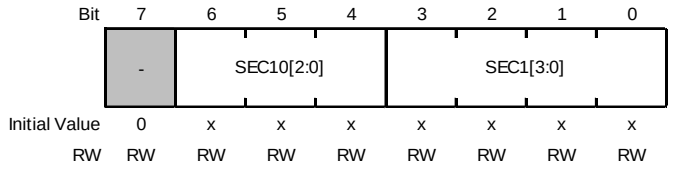
The setting range is decimal 00 to 59. The RTC will not operate normally if any other value is set. Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

After writing to the RSECCNT register, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

Access Size : 8 bits

Address : <RTC_base> + 02h
<RTC_Read_Only_base> + 02h

Initial Value : xxh



Bit	Bit Name	Initial Value	RW	Description
7	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6 to 4	SEC10[2:0]	x	RW	Counts from 0 to 5 for 60-second counting.
3 to 0	SEC1[3:0]	x	RW	Counts from 0 to 9 every second. When a carry is generated, 1 is added to the tens place.

x: Undefined value

b. In binary count mode

The BCNT0 counter is a readable/writable 32-bit binary counter b7 to b0.

The 32-bit binary counter performs count operation by a carry generating for each second of the 64-Hz counter.

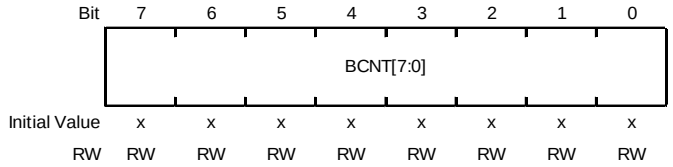
Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

To read this counter, follow the procedure in **5.3.3.5 Reading 64-Hz Counter and Time**.

Access Size : 8 bits

Offset Address : <RTC_base> + 02h
<RTC_Read_Only_base> + 02h

Initial Value : xxh



5.3.2.2.3 Minute Counter (RTC_RMINCNT)/Binary Counter 1 (RTC_BCNT1)

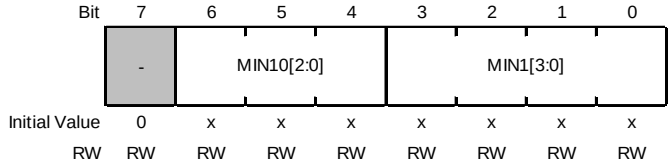
a. In calendar count mode

The RMINCNT counter is used for setting and counting the BCD-coded minute value. It counts carries generated once per minute in the second counter.

A value from 00 through 59 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

After writing to the RMINCNT counter, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

Access Size : 8 bits
Address : <RTC_base> + 04h
<RTC_Read_Only_base> + 04h
Initial Value : xxh



Bit	Bit Name	Initial Value	RW	Description
7	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6 to 4	MIN10[2:0]	x	RW	Counts from 0 to 5 for 60-minute counting.
3 to 0	MIN1[3:0]	x	RW	Counts from 0 to 9 every minute. When a carry is generated, 1 is added to the tens place.

x: Undefined value

b. In binary count mode

The BCNT1 counter is a readable/writable 32-bit binary counter b15 to b8.

The 32-bit binary counter performs count operation by a carry generating for each second of the 64-Hz counter.

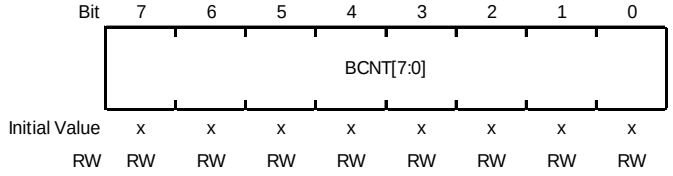
Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

To read this counter, follow the procedure in **5.3.3.5 Reading 64-Hz Counter and Time**.

Access Size : 8 bits

Offset Address : <RTC_base> + 04h
<RTC_Read_Only_base> + 04h

Initial Value : xxh



5.3.2.2.4 Hour Counter (RTC_RHRCNT)/Binary Counter 2 (RTC_BCNT2)

a. In calendar count mode

The RHRCNT counter is used for setting and counting the BCD-coded hour value. It counts carries generated once per hour in the minute counter.

The specifiable time differs according to the setting in the hours mode bit (RCR2.HR24).

When the RCR2.HR24 bit is 0: From 00 to 11 (in BCD)

When the RCR2.HR24 bit is 1: From 00 to 23 (in BCD)

If a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

The PM bit is only enabled when the RCR2.HR24 bit is 0. Otherwise, the setting in the PM bit has no effect.

After writing to the RHRCNT counter, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

Access Size : 8 bits

Address : <RTC_base> + 06h
<RTC_Read_Only_base> + 06h

Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	-	PM	HR10[1:0]		HR1[3:0]			
Initial Value	0	x	x	x	x	x	x	x
RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	RW	Description
7	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6	PM	x	RW	Time Counter Setting for a.m./p.m. 0b: a.m. 1b: p.m.
5 to 4	HR10[1:0]	x	RW	Counts from 0 to 2 once per carry from the ones place.
3 to 0	HR1[3:0]	x	RW	Counts from 0 to 9 once per hour. When a carry is generated, 1 is added to the tens place.

x: Undefined value

b. In binary count mode

The BCNT2 counter is a readable/writable 32-bit binary counter b23 to b16.

The 32-bit binary counter performs count operation by a carry generating for each second of the 64-Hz counter.

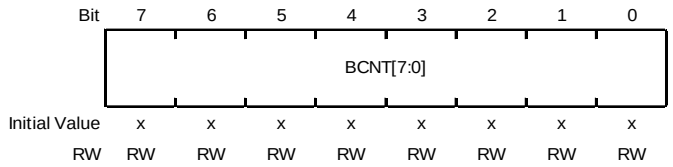
Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

To read this counter, follow the procedure in **5.3.3.5 Reading 64-Hz Counter and Time**.

Access Size : 8 bits

Offset Address : <RTC_base> + 06h
<RTC_Read_Only_base> + 06h

Initial Value : xxh



5.3.2.2.5 Day-of-Week Counter (RTC_RWKCNT)/Binary Counter 3 (RTC_BCNT3)

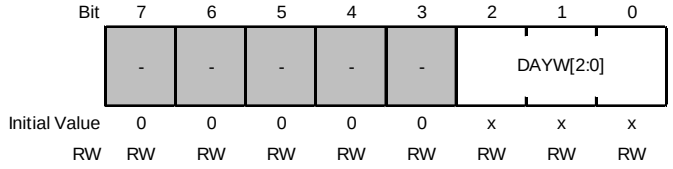
a. In calendar count mode

The RWKCNT counter is used for setting and counting in the coded day-of-week value. It counts carries generated once per day in the hour counter.

A value from 0 through 6 can be specified; if a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

Access Size : 8 bits
Address : <RTC_base> + 08h
<RTC_Read_Only_base> + 08h
Initial Value : 0xh



Bit	Bit Name	Initial Value	RW	Description
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	DAYW[2:0]	x	RW	000b: Sunday 001b: Monday 010b: Tuesday 011b: Wednesday 100b: Thursday 101b: Friday 110b: Saturday 111b: Setting prohibited

x: Undefined value

b. In binary count mode

The BCNT3 counter is a readable/writable 32-bit binary counter b31 to b24.

The 32-bit binary counter performs count operation by a carry generating for each second of the 64-Hz counter.

Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

To read this counter, follow the procedure in **5.3.3.5 Reading 64-Hz Counter and Time**.

Access Size : 8 bits

Offset Address : <RTC_base> + 08h
<RTC_Read_Only_base> + 08h

Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	BCNT[7:0]							
Initial Value	x	x	x	x	x	x	x	x
	RW	RW	RW	RW	RW	RW	RW	RW

5.3.2.2.6 Date Counter (RTC_RDAYCNT)

The RDAYCNT counter is used in calendar count mode.

The RDAYCNT counter is used for setting and counting the BCD-coded date value. It counts carries generated once per day in the hour counter. The count operation depends on the month and whether the year is a leap year.

Leap years are determined according to whether the year counter (RYRCNT) value is divisible by 400, 100, and 4.

A value from 01 through 31 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly. (When specifying a value, note that the range of specifiable days depends on the month and whether the year is a leap year.) Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

After writing to the RDAYCNT counter, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

Access Size : 8 bits

Address : <RTC_base> + 0Ah
<RTC_Read_Only_base> + 0Ah

Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	-	-	DATE10[1:0]		DATE1[3:0]			
Initial Value	0	0	x	x	x	x	x	x
RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	RW	Description
7, 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5, 4	DATE10[1:0]	x	RW	Counts from 0 to 3 once per carry from the ones place.
3 to 0	DATE1[3:0]	x	RW	Counts from 0 to 9 once per day. When a carry is generated, 1 is added to the tens place.

x: Undefined value

5.3.2.2.7 Month Counter (RTC_RMONCNT)

The RMONCNT counter is used in calendar count mode.

The RMONCNT counter is used for setting and counting the BCD-coded month value. It counts carries generated once per month in the date counter.

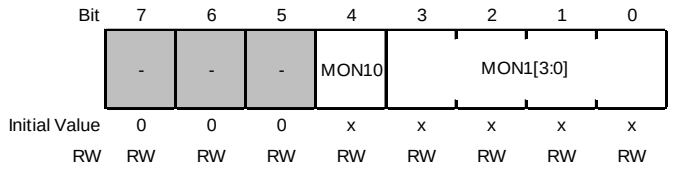
A value from 01 through 12 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

After writing to the RMONCNT counter, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

Access Size : 8 bits

Address : <RTC_base> + 0Ch
<RTC_Read_Only_base> + 0Ch

Initial Value : xxh



Bit	Bit Name	Initial Value	RW	Description
7 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	MON10	x	RW	Counts from 0 to 1 once per carry from the ones place.
3 to 0	MON1[3:0]	x	RW	Counts from 0 to 9 once per month. When a carry is generated, 1 is added to the tens place.

x: Undefined value

5.3.2.2.8 Year Counter (RTC_RYRCNT)

The RYRCNT counter is used in calendar count mode.

The RYRCNT counter is used for setting and counting the BCD-coded year value. It counts carries generated once per year in the month counter.

A value from 00 through 99 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly. Before writing to this register, be sure to stop the count operation through the setting of the START bit in RCR2.

After writing to the RYRCNT counter, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

Access Size : 16 bits

Address : <RTC_base> + 0Eh
<RTC_Read_Only_base> + 0Eh

Initial Value : 00xxh

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	YR10[3:0]				YR1[3:0]			
Initial Value	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	RW	Description
15 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 4	YR10[3:0]	x	RW	Counts from 0 to 9 once per carry from the ones place. When a carry is generated in the tens place, 1 is added to the hundreds place.
3 to 0	YR1[3:0]	x	RW	Counts from 0 to 9 once per year. When a carry is generated, 1 is added to the tens place.

x: Undefined value

5.3.2.2.9 Second Alarm Register (RTC_RSECAR)/Binary Counter 0 Alarm Register (RTC_BCNT0AR)

a. In calendar count mode

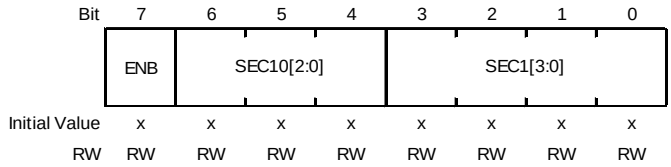
The RSECAR register is an alarm register corresponding to the BCD-coded second counter RSECCNT. When the ENB bit is set to 1, the RSECAR value is compared with the RSECCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR corresponding to the ALM interrupt is set to 1.

RSECAR values from 00 through 59 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly.

After writing to the RSECAR register, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

This register is set to 00h by an RTC software reset.

Access Size : 8 bits
Address : <RTC_base> + 10h
Initial Value : xxh



Bit	Bit Name	Initial Value	RW	Description
7	ENB	x	RW	0b: The register value is not compared with the RSECCNT counter value. 1b: The register value is compared with the RSECCNT counter value.
6 to 4	SEC10[2:0]	x	RW	Value for the tens place of seconds
3 to 0	SEC1[3:0]	x	RW	Value for the ones place of seconds

x: Undefined value

b. In binary count mode

The BCNT0AR counter is a readable/writable alarm register corresponding to 32-bit binary counter b7 to b0.

This register is set to 00h by an RTC software reset.

Access Size : 8 bits
Offset Address : <RTC_base> + 10h
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	BCNTAR[7:0]							
Initial Value	x	x	x	x	x	x	x	x
RW	RW	RW	RW	RW	RW	RW	RW	RW

5.3.2.2.10 Minute Alarm Register (RTC_RMINAR)/Binary Counter 1 Alarm Register (RTC_BCNT1AR)

a. In calendar count mode

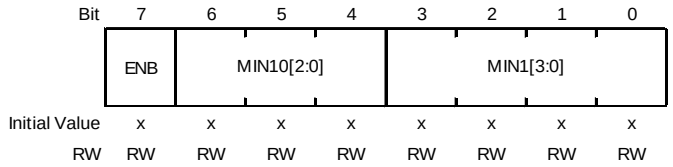
The RMINAR register is an alarm register corresponding to the BCD-coded minute counter RMINCNT. When the ENB bit is set to 1, the RMINAR value is compared with the RMINCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR corresponding to the ALM interrupt is set to 1.

RMINAR values from 00 through 59 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly.

After writing to the RMINAR register, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

This register is set to 00h by an RTC software reset.

Access Size : 8 bits
Address : <RTC_base> + 12h
Initial Value : xxh



Bit	Bit Name	Initial Value	RW	Description
7	ENB	x	RW	0b: The register value is not compared with the RMINCNT counter value. 1b: The register value is compared with the RMINCNT counter value.
6 to 4	MIN10[2:0]	x	RW	Value for the tens place of minutes
3 to 0	MIN1[3:0]	x	RW	Value for the ones place of minutes

x: Undefined value

b. In binary count mode

The BCNT1AR counter is a readable/writable alarm register corresponding to 32-bit binary counter b15 to b8.

This register is set to 00h by an RTC software reset.

Access Size : 8 bits
Offset Address : <RTC_base> + 12h
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	BCNTAR[7:0]							
Initial Value	x	x	x	x	x	x	x	x
RW	RW	RW	RW	RW	RW	RW	RW	RW

5.3.2.2.11 Hour Alarm Register (RTC_RHRAR)/Binary Counter 2 Alarm Register (RTC_BCNT2AR)

a. In calendar count mode

The RHRAR register is an alarm register corresponding to the BCD-coded hour counter RHRCNT. When the ENB bit is set to 1, the RHRAR value is compared with the RHRCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR corresponding to the ALM interrupt is set to 1.

The specifiable time differs according to the setting in the hours mode bit (RCR2.HR24).

- When the RCR2.HR24 bit is 0: From 00 to 11 (in BCD)
- When the RCR2.HR24 bit is 1: From 00 to 23 (in BCD)

If a value outside of this range is specified, the RTC does not operate correctly.

- When the RCR2.HR24 bit is 0, be sure to set the PM bit.
- When the RCR2.HR24 bit is 1, the setting in the PM bit has no effect.

After writing to the RHRAR register, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

This register is set to 00h by an RTC software reset.

Access Size : 8 bits
Address : <RTC_base> + 14h
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	ENB	PM	HR10[1:0]		HR1[3:0]			
Initial Value	x	x	x	x	x	x	x	x
RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	RW	Description
7	ENB	x	RW	0b: The register value is not compared with the RHRCNT counter value. 1b: The register value is compared with the RHRCNT counter value.
6	PM	x	RW	Time Alarm Setting for a.m./p.m. 0b: a.m. 1b: p.m.
5, 4	HR10[1:0]	x	RW	Value for the tens place of hours
3 to 0	HR1[3:0]	x	RW	Value for the ones place of hours

x: Undefined value

b. In binary count mode

The BCNT2AR counter is a readable/writable alarm register corresponding to 32-bit binary counter b23 to b16.

This register is set to 00h by an RTC software reset.

Access Size : 8 bits
Offset Address : <RTC_base> + 14h
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	BCNTAR[7:0]							
Initial Value	x	x	x	x	x	x	x	x
RW	RW	RW	RW	RW	RW	RW	RW	RW

5.3.2.2.12 Day-of-Week Alarm Register (RTC_RWKAR)/Binary Counter 3 Alarm Register (RTC_BCNT3AR)

a. In calendar count mode

The RWKAR register is an alarm register corresponding to the coded day-of-week counter RWKCNT. When the ENB bit is set to 1, the RWKAR value is compared with the RWKCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR corresponding to the ALM interrupt is set to 1.

RWKAR values from 0 through 6 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly.

After writing to the RWKAR register, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

This register is set to 00h by an RTC software reset.

Access Size : 8 bits
Address : <RTC_base> + 16h
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	ENB	-	-	-	-	DAYW[2:0]		
Initial Value	x	0	0	0	0	x	x	x
RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	RW	Description
7	ENB	x	RW	0b: The register value is not compared with the RWKCNT counter value. 1b: The register value is compared with the RWKCNT counter value.
6 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	DAYW[2:0]	x	RW	000b: Sunday 001b: Monday 010b: Tuesday 011b: Wednesday 100b: Thursday 101b: Friday 110b: Saturday 111b: Setting prohibited

x: Undefined value

b. In binary count mode

The BCNT3AR counter is a readable/writable alarm register corresponding to 32-bit binary counter b31 to b24.

This register is set to 00h by an RTC software reset.

Access Size : 8 bits
Offset Address : <RTC_base> + 16h
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	BCNTAR[7:0]							
Initial Value	x	x	x	x	x	x	x	x
RW	RW	RW	RW	RW	RW	RW	RW	RW

5.3.2.2.13 Date Alarm Register (RTC_RDAYAR)/Binary Counter 0 Alarm Enable Register (RTC_BCNT0AER)

a. In calendar count mode

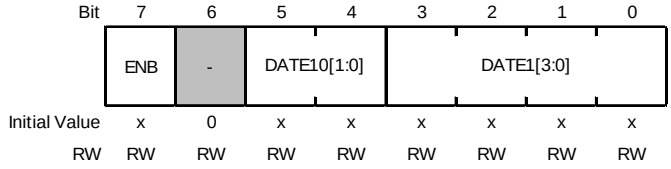
The RDAYAR register is an alarm register corresponding to the BCD-coded date counter RDAYCNT. When the ENB bit is set to 1, the RDAYAR value is compared with the RDAYCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR corresponding to the ALM interrupt is set to 1.

RDAYAR values from 01 through 31 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly.

After writing to the RDAYAR register, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

This register is set to 00h by an RTC software reset.

Access Size : 8 bits
Address : <RTC_base> + 18h
Initial Value : xxh



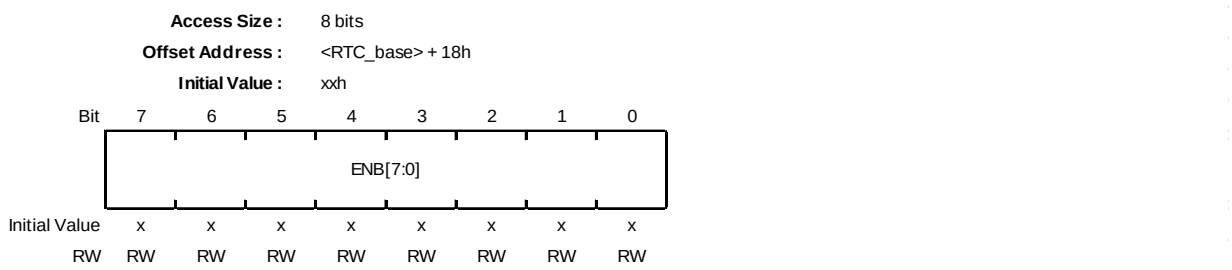
Bit	Bit Name	Initial Value	RW	Description
7	ENB	x	RW	0b: The register value is not compared with the RDAYCNT counter value. 1b: The register value is compared with the RDAYCNT counter value.
6	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5, 4	DATE10[1:0]	x	RW	Value for the tens place of days
3 to 0	DATE1[3:0]	x	RW	Value for the ones place of days

x: Undefined value

b. In binary count mode

The BCNT0AER register is a readable/writable register for setting the alarm enable corresponding to 32-bit binary counter b7 to b0. Among the ENB[31:0] bits, the binary counter (BCNT[31:0]) corresponding to the bits which are set to 1 and the binary alarm register (BCNTAR[31:0]) are compared, and when all match, the IR (alarm of RSR) flag corresponding to the ALM interrupt becomes 1.

This register is set to 00h by an RTC software reset.



5.3.2.2.14 Month Alarm Register (RTC_RMONAR)/Binary Counter 1 Alarm Enable Register (RTC_BCNT1AER)

a. In calendar count mode

The RMONAR register is an alarm register corresponding to the BCD-coded month counter RMONCNT. When the ENB bit is set to 1, the RMONAR value is compared with the RMONCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the alarm flag of RSR corresponding to the ALM interrupt is set to 1.

RMONAR values from 01 through 12 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly.

After writing to the RMONAR register, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

This register is set to 00h by an RTC software reset.

Access Size : 8 bits
Address : <RTC_base> + 1Ah
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	ENB	-	-	MON10	MON1[3:0]			
Initial Value	x	0	0	x	x	x	x	x
RW	RW	RW	RW	RW	RW	RW	RW	RW

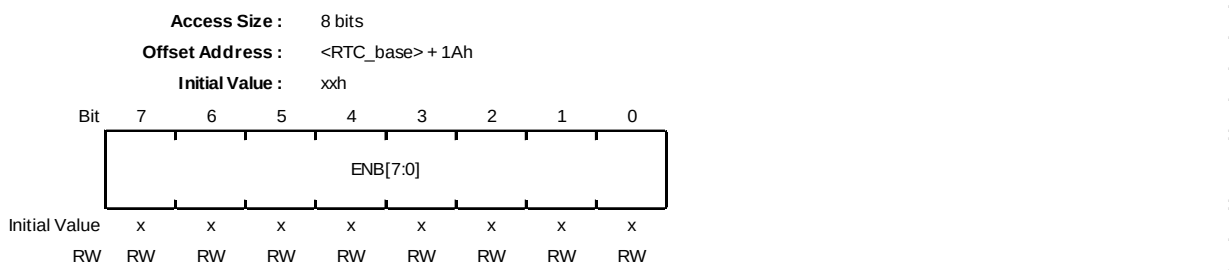
Bit	Bit Name	Initial Value	RW	Description
7	ENB	x	RW	0b: The register value is not compared with the RMONCNT counter value. 1b: The register value is compared with the RMONCNT counter value.
6, 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	MON10	x	RW	Value for the tens place of months
3 to 0	MON1[3:0]	x	RW	Value for the ones place of months

x: Undefined value

b. In binary count mode

The BCNT1AER register is a readable/writable register for setting the alarm enable corresponding to 32-bit binary counter b15 to b8. Among the ENB[31:0] bits, the binary counter (BCNT[31:0]) corresponding to the bits which are set to 1 and the binary alarm register (BCNTAR[31:0]) are compared, and when all match, the IR (alarm of RSR) flag corresponding to the ALM interrupt becomes 1.

This register is set to 00h by an RTC software reset.



5.3.2.2.15 Year Alarm Register (RTC_RYRAR)/Binary Counter 2 Alarm Enable Register (RTC_BCNT2AER)

a. In calendar count mode

The RYRAR register is an alarm register corresponding to the BCD-coded year counter RYRCNT.

RYRAR values from 00 through 99 (in BCD) can be specified; if a value outside of this range is specified, the RTC does not operate correctly.

After writing to the RYRAR register, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** for notes on accessing registers.

This register is set to 0000h by an RTC software reset.

Access Size : 16 bits
Address : <RTC_base> + 1Ch
Initial Value : 00xxh

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	YR10[3:0]				YR1[3:0]			
Initial Value	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	RW	Description
15 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 4	YR10[3:0]	x	RW	Value for the tens place of years
3 to 0	YR1[3:0]	x	RW	Value for the ones place of years

x: Undefined value

b. In binary count mode

The BCNT2AER register is a readable/writable register for setting the alarm enable corresponding to 32-bit binary counter b23 to b16. Among the ENB[31:0] bits, the binary counter (BCNT[31:0]) corresponding to the bits which are set to 1 and the binary alarm register (BCNTAR[31:0]) are compared, and when all match, the IR (alarm of RSR) flag corresponding to the ALM interrupt becomes 1.

This register is set to 0000h by an RTC software reset.

Access Size : 16 bits
Offset Address : <RTC_base> + 1Ch
Initial Value : 00xxh

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ENB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

5.3.2.2.16 Year Alarm Enable Register (RTC_RYRAREN)/Binary Counter 3 Alarm Enable Register (RTC_BCNT3AER)

a. In calendar count mode

When the ENB bit in the RYRAREN register is set to 1, the RYRAR value is compared with the RYRCNT value. From among the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, and RYRAREN), only those selected with the ENB bits set to 1 are compared with the corresponding counters. When the respective values all match, the IR (alarm of RSR) flag corresponding to the ALM interrupt is set to 1.

This register is set to 00h by an RTC software reset.

Access Size : 8 bits
Address : <RTC_base> + 1Eh
Initial Value : x0h

Bit	7	6	5	4	3	2	1	0
	ENB	-	-	-	-	-	-	-
Initial Value	x	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW	RW

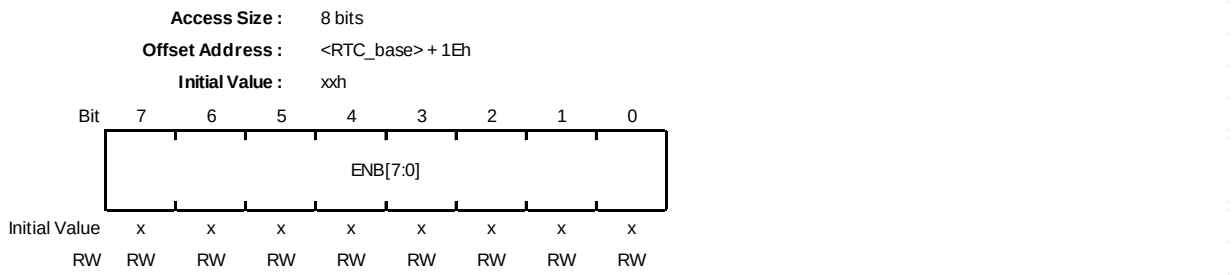
Bit	Bit Name	Initial Value	RW	Description
7	ENB	x	RW	0b: The register value is not compared with the RYRCNT counter value. 1b: The register value is compared with the RYRCNT counter value.
6 to 0	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

x: Undefined value

b. In binary count mode

The BCNT3AER register is a readable/writable register for setting the alarm enable corresponding to 32-bit binary counter b31 to b24. Among the ENB[31:0] bits, the binary counter (BCNT[31:0]) corresponding to the bits which are set to 1 and the binary alarm register (BCNTAR[31:0]) are compared, and when all match, the alarm flag of RSR corresponding to the ALM interrupt becomes 1.

This register is set to 00h by an RTC software reset.



5.3.2.2.17 RTC Status Register (RTC_RSR)

RSR is a flag register for a periodic interrupt, carry, and alarm. This register is common to calendar count mode and binary count mode.

Each flag is set to 1 when the prescaler or clock counter matches the given interrupt setting condition. Since the prescaler, clock counter, and the setting register for each interrupt are not reset, the flag may remain set until it is read.

This register is cleared to 00h by an RTC software reset.

Access Size : 8 bits
Address : <RTC_base> + 20h
Initial Value : 0xh

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	PF	CF	AF
Initial Value	0	0	0	0	0	x	x	x
RW	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	RW	Description
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	PF	x*1	RW	This flag indicates the generation of an interrupt at the interval set by the RCR1.PES[3:0] bits. A periodic interrupt is generated when this flag is set to 1. Clearing condition: 0 is written to the PF flag. Setting condition: An interrupt is generated at the interval set by the RCR1.PES[3:0] bits. 0b: No interrupt is generated at the interval set by the RCR1.PES[3:0] bits. 1b: An interrupt is generated at the interval set by the RCR1.PES[3:0] bits.*1
1	CF	x*1	RW	The setting of this flag being 1 indicates the generation of a carry to the second counter/binary counter 0 or a carry to the 64-Hz counter during reading of the 64-Hz counter. A value of the counter read at this point is not guaranteed. Be sure to read the counter again in such cases. Clearing condition: 0 is written to the CF flag. Setting conditions: • A carry to the second counter/binary counter 0 or a carry to the 64-Hz counter during reading of the 64-Hz counter is generated. • 1 is written to the CF flag. 0b: A carry to the second counter/binary counter 0 or a carry to the 64-Hz counter while reading the 64-Hz counter has not occurred. 1b: A carry to the second counter/binary counter 0 or a carry to the 64-Hz counter while reading the 64-Hz counter has occurred.
0	AF	x*1	RW	This flag is set to 1 when the counter matches the alarm time set by the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR. RDAYAR, RMONAR, and RYRAR in calendar count mode, and BCNT0AR, BCNT1AR, BCNT2AR, and BCNT3AR in binary count mode: only those in which the ENB bit is set to 1). Clearing condition: 0 is written to the AF flag. Setting condition: The counter matches the alarm registers (only those in which the ENB bit is set to 1). 0b: The alarm register and the counter do not match. 1b: The alarm register and the counter match.*1

x: Undefined value

Note 1. Writing 1 has no effect.

5.3.2.2.18 RTC Control Register 1 (RTC_RCR1)

The RCR1 register is used in both calendar count mode and in binary count mode.

Bits AIE, PIE, and PES[3:0] are updated synchronously with the count source. When the RCR1 register is modified, check that all the bits have been updated before proceeding to the next processing.

Access Size : 8 bits
Address : <RTC_base> + 22h
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	PES[3:0]				-	PIE	CIE	AIE
Initial Value	x	x	x	x	0	x	0	x
RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	RW	Description
7 to 4	PES[3:0]	x	RW	0111b: A periodic interrupt is generated every 1/128 seconds. 1000b: A periodic interrupt is generated every 1/64 seconds. 1001b: A periodic interrupt is generated every 1/32 seconds. 1010b: A periodic interrupt is generated every 1/16 seconds. 1011b: A periodic interrupt is generated every 1/8 seconds. 1100b: A periodic interrupt is generated every 1/4 seconds. 1101b: A periodic interrupt is generated every 1/2 seconds. 1110b: A periodic interrupt is generated every 1 second. 1111b: A periodic interrupt is generated every 2 seconds. Other than above: No periodic interrupts are generated.
3	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	PIE	x	RW	0b: A periodic interrupt request is disabled. 1b: A periodic interrupt request is enabled.
1	CIE	0h	RW	0b: A carry interrupt request is disabled. 1b: A carry interrupt request is enabled.
0	AIE	x	RW	0b: An alarm interrupt request is disabled. 1b: An alarm interrupt request is enabled.

x: Undefined value

AIE Bit (Alarm Interrupt Enable)

This bit enables or disables alarm interrupt requests.

CIE Bit (Carry Interrupt Enable)

This bit enables and disables interrupt requests when a carry to the RSECCNT/BCNT0 register occurs, or when a carry to the 64-Hz counter (R64CNT) occurs while reading the 64-Hz counter.

PIE Bit (Periodic Interrupt Enable)

This bit enables or disabled a periodic interrupt.

PES[3:0] Bits (Periodic Interrupt Select)

These bits specify the period for the periodic interrupt. A periodic interrupt is generated with the period specified by these bits.

5.3.2.2.19 RTC Control Register 2 (RTC_RCR2)

The RCR2 register is related to hours mode, automatic adjustment function, 30-second adjustment, RTC software reset, and controlling count operation.

Access Size : 8 bits
Address : <RTC_base> + 24h
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	CNTMD	HR24	AADJP	AADJE	-	ADJ30	RESET	START
Initial Value	x	x	x	x	0	0	0	x
RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	RW	Description
7	CNTMD*3	x	RW	0b: Calendar count mode 1b: Binary count mode
6	HR24*2,*3	x	RW	0b: The RTC operates in 12-hour mode. 1b: The RTC operates in 24-hour mode.
5	AADJP*3	x	RW	0b: The RADJ.ADJ[5:0] setting value is adjusted from the count value of the prescaler every minute (every 32 seconds in binary counter mode). 1b: The RADJ.ADJ[5:0] setting value is adjusted from the count value of the prescaler every 10 seconds (every 8 seconds in binary counter mode).
4	AADJE*3	x	RW	0b: Automatic adjustment is disabled. 1b: Automatic adjustment is enabled.
3	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	ADJ30*2	0h	RW	Writing 0b: Writing is invalid. 1b: 30-second adjustment is executed. Reading 0b: In normal time operation, or 30-second adjustment has completed. 1b: During 30-second adjustment
1	RESET	0h	RW	Writing 0b: Writing is invalid. 1b: The prescaler and the target registers for RTC software reset*1 are initialized. Reading 0b: In normal time operation, or an RTC software reset has completed. 1b: During an RTC software reset
0	START*3	x	RW	0b: The prescaler and counter are stopped. 1b: The prescaler and counter operate normally.

x: Undefined value

Note 1. R64CNT, RSECAR/BCNT0AR, RMINAR/BCNT1AR, RHRAR/BCNT2AR, RWKAR/BCNT3AR, RDAYAR/BCNT0AER, RMONAR/BCNT1AER, RYRAR/BCNT2AER, RYRAREN/BCNT3AER, RADJ, RTCCRn, RSECCPn/BCNT0CPn, RMINCPn/BCNT1CPn, RHRCPn/BCNT2CPn, RDAYCPn/BCNT3CPn, RMONCPn, RSR, RCR2.ADJ30, RCR2.AADJE, RCR2.AADJP

Note 2. This bit is reserved in binary counter mode. When writing to this bit, write 0.

Note 3. After writing to this bit, confirm that its value has actually changed before proceeding with further processing. Refer to **5.3.6.4 Notes on Writing to and Reading from Registers** regarding changes to the values of the AADJE, AADJP, and HR24 bits.

START Bit (Start)

This bit stops or restarts the prescaler or counter (clock) operation.

The START bit is updated in synchronization with the count source. When the START bit is modified, check that the bit is updated before proceeding to the next processing.

RESET Bit (RTC Software Reset)

This bit initializes the prescaler and registers to be reset by RTC software reset.

When 1 is written to the RESET bit, the initialization starts in synchronization with the count source. When the initialization is completed, the RESET bit is automatically set to 0.

When 1 is written to the RESET bit, check that the bit is set to 0, and then make next settings.

ADJ30 Bit (30-Second Adjustment)

This bit is for 30-second adjustment.

When 1 is written to the ADJ30 bit, the RSECCNT value of 30 seconds or less is rounded down to 00 second and the value of 30 seconds or more is rounded up to 1 minute.

The 30-second adjustment is performed in synchronization with the count source. When 1 is written to this bit, the ADJ30 bit is automatically set to 0 after the 30-second adjustment is completed. In case when 1 is written to the ADJ30 bit, check that the bit is set to 0, and then make next settings.

When the 30-second adjustment is performed, the prescaler and R64CNT are also reset.

The ADJ30 bit is set to 0 by an RTC software reset.

This bit is reserved in binary counter mode. The write value should be 0.

AADJE Bit (Automatic Adjustment Enable)

This bit controls (enables or disables) automatic adjustment.

Set the plus–minus bits (RADJ.PMADJ[1:0]) to 00b (adjustment is not performed) before changing the value of the AADJE bit.

The AADJE bit is set to 0 by an RTC software reset.

AADJP Bit (Automatic Adjustment Period Select)

This bit selects the automatic-adjustment period.

Set the plus–minus bits (RADJ.PMADJ[1:0]) to 00b (adjustment is not performed) before changing the value of the AADJP bit.

The AADJP bit is set to 0 by an RTC software reset.

HR24 Bit (Hours Mode)

This bit specifies whether the RTC will operate in 12- or 24-hour mode.

Use the START bit to stop counting by the counters before changing the value of the HR24 bit. Do not stop counting (write 0 to the START bit) and change the value of the HR24 bit at the same time.

This bit is reserved in binary counter mode. The write value should be 0.

CNTMD Bit (Count Mode Select)

This bit specifies whether the RTC count mode is operated in calendar count mode or in binary count mode.

After setting the count mode, execute an RTC software reset and start again from the initial settings.

The CNTMD bit is updated in synchronization with the count source, so when the value of the CNTMD bit has been changed, check that the value of the bit has actually been updated before applying the RTC software reset. The count mode changes to that which was specified beforehand in the CNTMD bit after the RTC software reset is applied.

For details on initial settings, refer to **5.3.3.1 Outline of Initial Settings of Registers after Power On**.

5.3.2.2.20 Time Error Adjustment Register (RTC_RADJ)

Adjustment is performed by the addition to or subtraction from the prescaler.

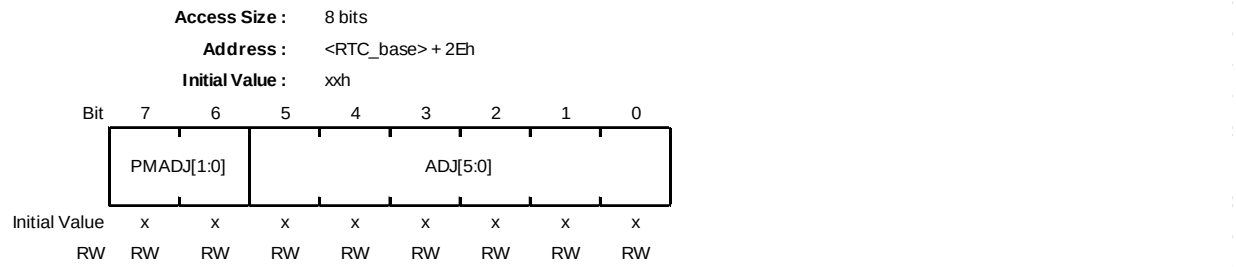
In case when the automatic adjustment enable (RCR2.AADJE) bit is 0, adjustment is performed when writing to the RADJ.

In case when the RCR2.AADJE bit is 1, adjustment is performed in the interval specified by the automatic adjustment period select (RCR2.AADJP) bit.

The current adjustment by software (disabling automatic adjustment) may be invalid if the following adjustment value is specified within 320 cycles of the count source after the register setting. To perform adjustment consecutively, wait for 320 cycles or more of the count source after the register setting and then specify the next adjustment value.

RADJ is updated in synchronization with the count source. When RADJ is modified, check that all the bits have been updated before continuing with further processing.

This register is set to 00h by an RTC software reset.



Bit	Bit Name	Initial Value	RW	Description
7, 6	PMADJ[1:0]	x	RW	00b: Adjustment is not performed. 01b: Adjustment is performed by addition to the prescaler. 10b: Adjustment is performed by subtraction from the prescaler. 11b: Setting prohibited
5 to 0	ADJ[5:0]	x	RW	These bits specify the adjustment value from the prescaler.

x: Undefined value

ADJ[5:0] Bits (Adjustment Value)

These bits specify the adjustment value (the number of sub-clock cycles) from the prescaler.

PMADJ[1:0] Bits (Plus–Minus)

These bits select whether the clock is set ahead or back depending on the error-adjustment value set in the ADJ[5:0] bits.

5.3.3 Operation

5.3.3.1 Outline of Initial Settings of Registers after Power On

After the power is turned on, the initial settings for the clock setting, count mode setting, time error adjustment, time setting, alarm, interrupt, and time capture control register should be performed.

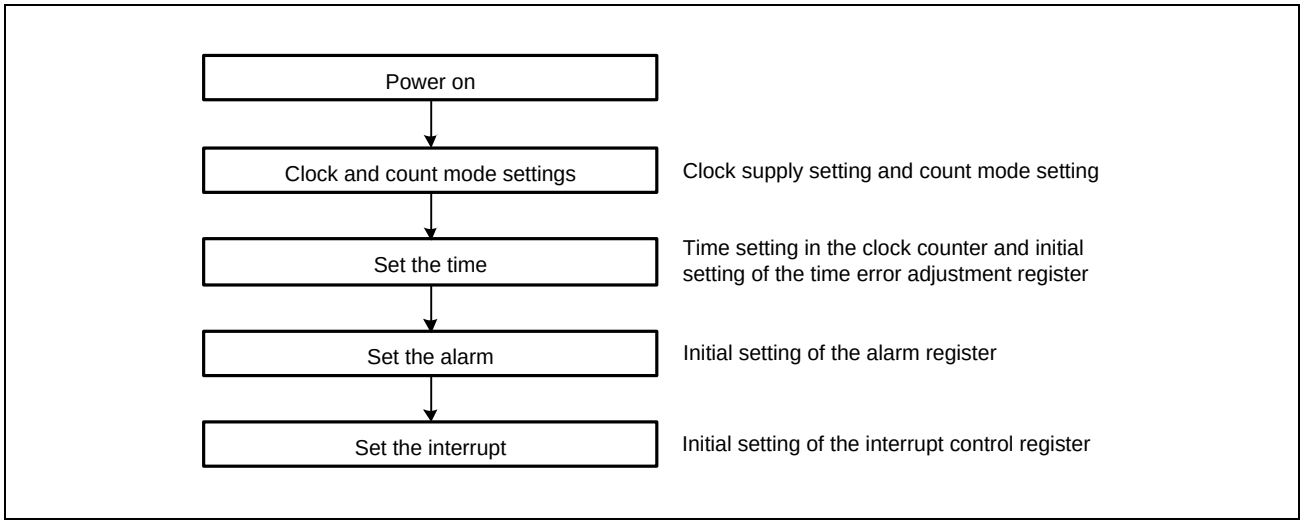
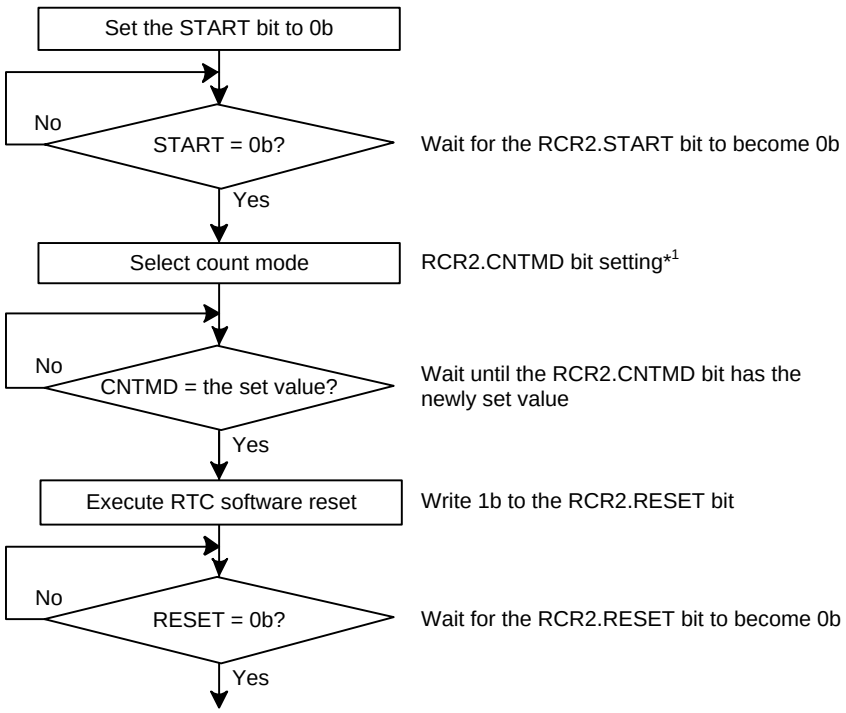


Figure 5.3-2 Outline of Initial Settings after Power On

5.3.3.2 Clock and Count Mode Setting Procedure

Figure 5.3-3 shows how to set the clock and the count mode.

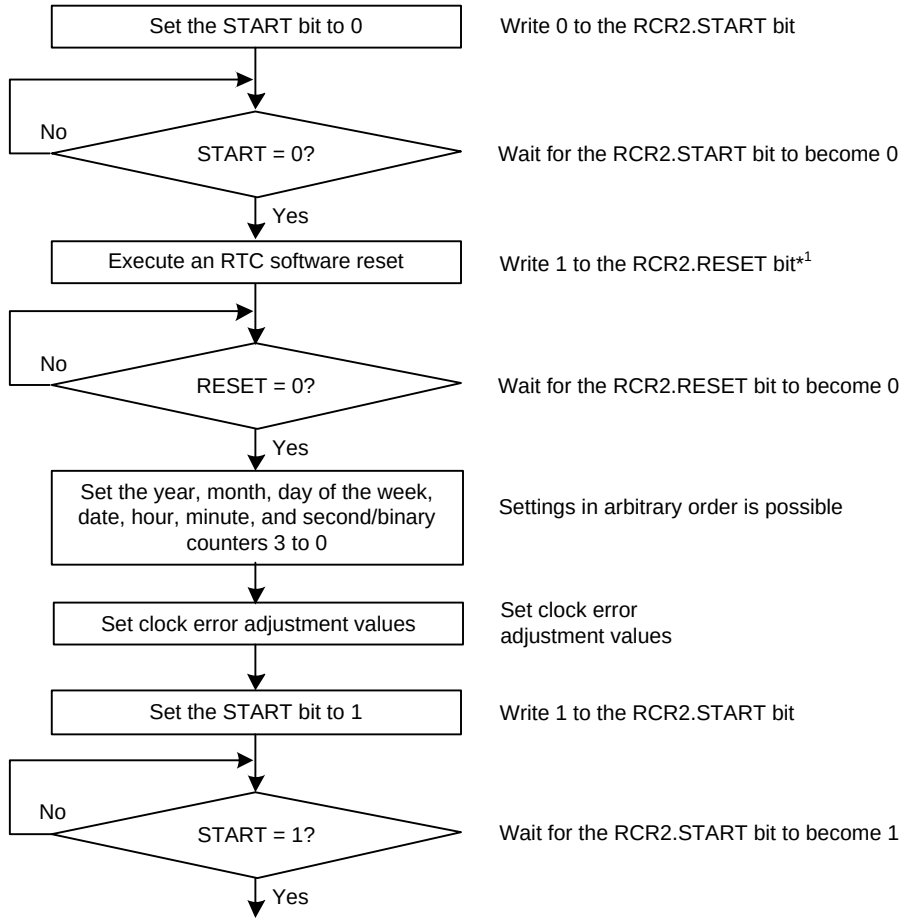


Note 1. This step is not necessary if the count mode has been set concurrently with setting the START bit to 0. A value corresponding to the count mode setting must be written to the RCR2.CNTMD bit.

Figure 5.3-3 Clock and Count Mode Setting Procedure

5.3.3.3 Setting the Time

Figure 5.3-4 shows how to set the time.



Note 1. This step is not necessary for the time-setting procedure because an RTC software reset is executed in the clock setting procedure of the initial settings for the power supply.

Figure 5.3-4 Setting the Time

5.3.3.4 30-Second Adjustment

Figure 5.3-5 shows how to execute 30-second adjustment.

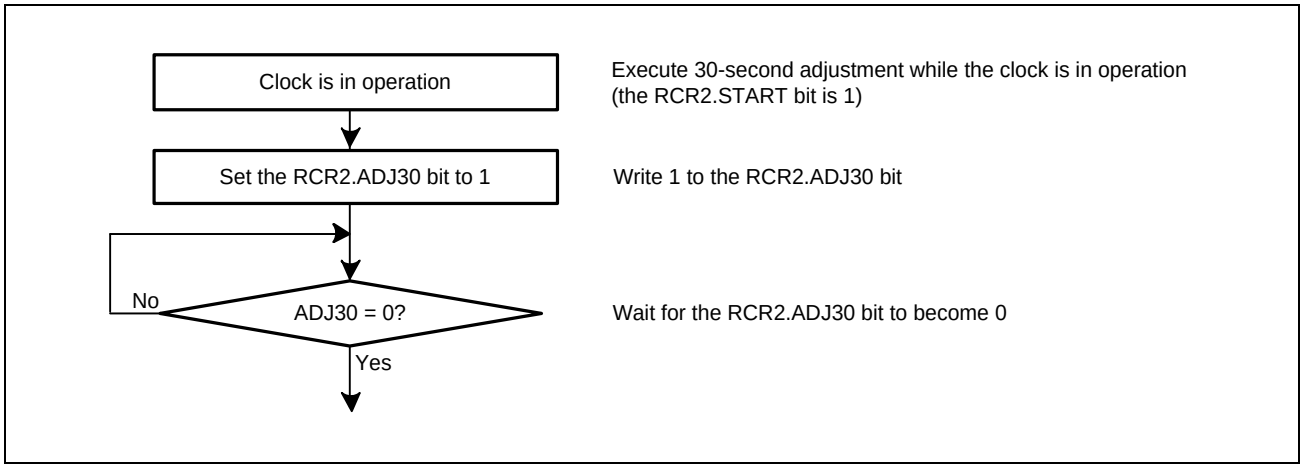
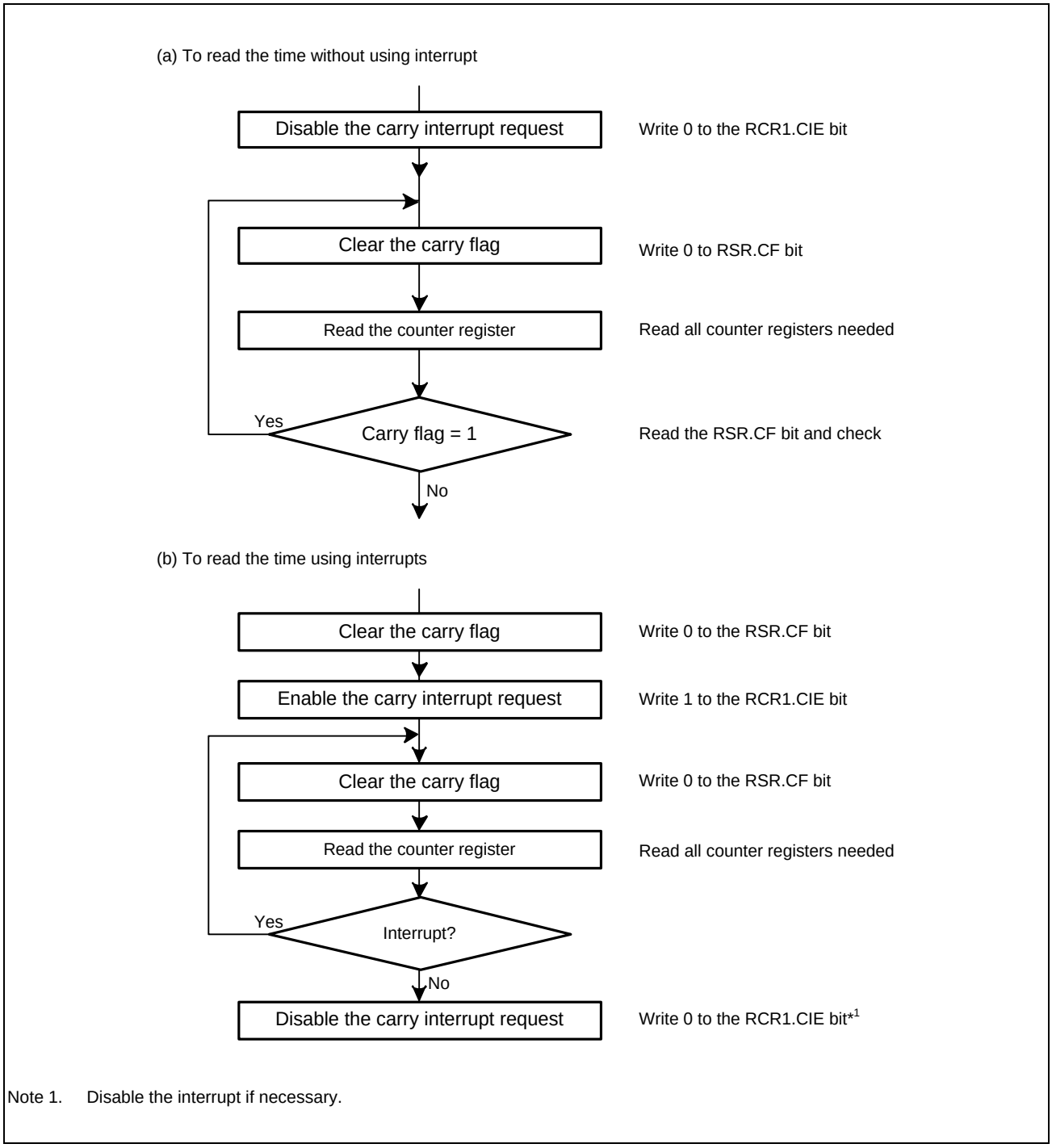


Figure 5.3-5 30-Second Adjustment

5.3.3.5 Reading 64-Hz Counter and Time

Figure 5.3-6 shows how to read the 64-Hz counter and time.



Note 1. Disable the interrupt if necessary.

Figure 5.3-6 Reading Time

If a carry occurs while the 64-Hz counter and time are being read, the correct time will not be obtained, so they must be read again. The procedure for reading the time without using interrupts is shown in (a) in **Figure 5.3-6**, and the procedure using carry interrupts in (b). To keep the program simple, method (a) should be used in most cases.

5.3.3.6 Alarm Function

Figure 5.3-7 shows how to use the alarm function.

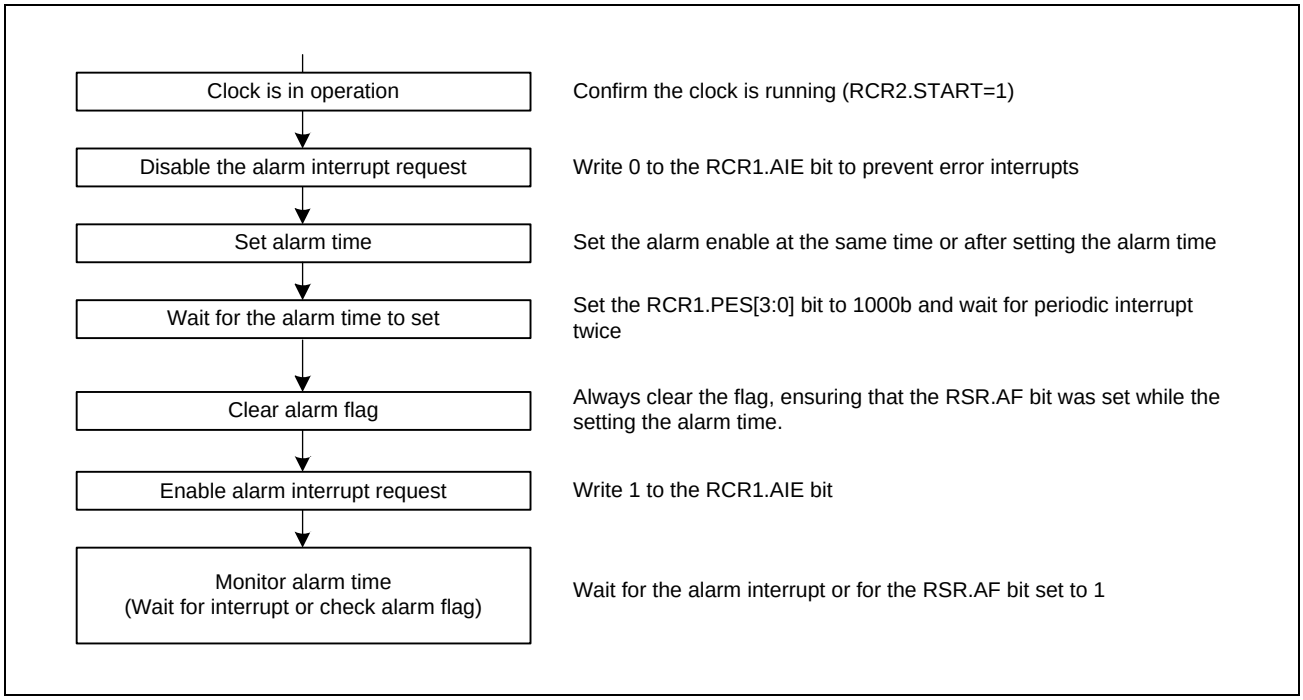


Figure 5.3-7 Using Alarm Function

In calendar count mode, an alarm can be generated by any one of year, month, date, day-of-week, hour, minute or second, or any combination of those. Write 1 to the ENB bit in the alarm registers involved in the alarm setting, and set the alarm time in the lower bits. Write 0b to the ENB bit in registers not involved in the alarm setting.

In binary count mode, an alarm can be generated in any bit combination of 32 bits. Write 1b to the ENB bit of the alarm enable register corresponding to the target bit of the alarm, and set the alarm time to the alarm register. For bits that are not target of the alarm, write 0b to the ENB bit of the alarm enable register.

When the counter and the alarm time match, the alarm flag of RSR is set to 1. Alarm detection can be confirmed by reading this flag, but an interrupt should be used in most cases. If 1 has been written to the RCR1.AIE bit, an alarm interrupt is generated in the event of alarm, enabling the alarm to be detected.

Writing 0b to the alarm flag of RSR clears it to 0b.

5.3.3.7 Time Error Adjustment Function

The time error adjustment function is used to correct errors (running fast or slow) in the time due to the precision of oscillation by the RTC.

Since 32,768 cycles of the RTC constitute 1 second of operation, the clock runs fast if the RTC frequency is high and slow if the RTC frequency is low. This function can be used to correct errors by advancing or delaying the clock.

Two types of time error adjustment functions are provided: automatic adjustment and adjustment by software.

Use the RCR2.AADJE bit to select automatic adjustment or adjustment by software.

5.3.3.7.1 Automatic adjustment

Enable automatic adjustment by setting the RCR2.AADJE bit to 1.

Automatic adjustment is the addition or subtraction of the value counted by the prescaler to or from the value in the RADJ register every time the adjustment period selected by the RCR2.AADJE bit elapses.

Examples are shown below.

[*Example 1*] RTC running at 32.769 kHz

Adjustment procedure:
When the RTC is running at 32.769 kHz, 1 second elapses every 32,769 clock cycles. The RTC is meant to run at 32,768 clock cycles, so the clock runs fast by one clock cycle every second. The time on the clock is fast by 60 clock cycles per minute, so adjustment can take the form of setting the clock back by 60 cycles every minute.

Register settings: (when RCR2.CNTMD = 0)

- RCR2.AADJP = 0b (adjustment every minute)
- RADJ.PMADJ[1:0] = 10b (adjustment is performed by the subtraction from the prescaler.)
- RADJ.ADJ[5:0] = 60 (3Ch)

[*Example 2*] RTC running at 32.766 kHz

Adjustment procedure:
When the RTC is running at 32.766 kHz, 1 second elapses every 32,766 clock cycles. The RTC is meant to run at 32,768 clock cycles, so the clock runs slow by two clock cycles every second. The time on the clock is slow by 20 clock cycles every 10 seconds, so adjustment can take the form of setting the clock forward by 20 cycles every 10 seconds.

Register settings: (when RCR2.CNTMD = 0)

- RCR2.AADJP = 1b (adjustment every 10 seconds)
- RADJ.PMADJ[1:0] = 01b (adjustment is performed by the addition to the prescaler.)
- RADJ.ADJ[5:0] = 20 (14h)

[*Example 3*] RTC running at 32.764 kHz

Adjustment procedure:
When the RTC is running at 32.764 kHz, 1 second elapses on 32,764 clock cycles. Since the RTC operates for 32,768 clock cycles as 1 second, the clock is delayed for four clock cycles per second. In 8 seconds, the delay is 32 clock cycles, so correction can be made by proceeding the clock for 32 clock cycles every 8 seconds.

Register settings when the RCR2.CNTMD bit is 1

- RCR2.AADJP = 1b (adjustment every 8 seconds)
- RADJ.PMADJ[1:0] = 01b (adjustment is performed by the addition to the prescaler.)
- RADJ.ADJ[5:0] = 32 (20h)

5.3.3.7.2 Adjustment by software

Enable adjustment by software by setting the RCR2.AADJE bit to 0b.

Adjustment by software is the addition or subtraction of the value counted by the prescaler to or from the value in the RADJ register at the time of execution of an instruction for writing to the RADJ register.

An example is shown below.

[*Example 1*] RTC running at 32.769 kHz

Adjustment procedure:
When the RTC is running at 32.769 kHz, 1 second elapses every 32,769 clock cycles. The RTC is meant to run at 32,768 clock cycles, so the clock runs fast by one clock cycle every second. The time on the clock is fast by one clock cycle per second, so adjustment can take the form of setting the clock back by one cycle every second.

Register settings:

- RADJ.PMADJ[1:0] = 10b (adjustment is performed by the subtraction from the prescaler.)
- RADJ.ADJ[5:0] = 1 (01h)
  This is written to the RADJ register once per 1-second interrupt.

5.3.3.7.3 Procedure for changing the mode of adjustment

When changing the mode of adjustment, change the value of the AADJE bit in RCR2 after setting the RADJ.PMADJ[1:0] bits to 00b (adjustment is not performed).

Changing from adjustment by software to automatic adjustment:

(1) Set the RADJ.PMADJ[1:0] bits to 00b (adjustment is not performed).

(2) Set the RCR2.AADJE bit to 1b (automatic adjustment is enabled).

(3) Use the RCR2.AADJP bit to select the period of adjustment.

(4) In RADJ, set the PMADJ[1:0] bits for addition or subtraction and the ADJ[5:0] bits to the value for use in time error adjustment.

Changing from adjustment by software to automatic adjustment:

(1) Set the RADJ.PMADJ[1:0] bits to 00b (adjustment is not performed).

(2) Set the RCR2.AADJE bit to 0b (adjustment by software is enabled).

(3) Proceed with adjustment by setting the RADJ.PMADJ[1:0] bits for addition or subtraction and the RADJ.ADJ[5:0] bits to the value for use in time error adjustment at the desired time. After that, the time is adjusted every time a value is written to the RADJ register.

5.3.3.7.4 Procedure for stopping adjustment

To stop adjustment, set the RADJ.PMADJ[1:0] bits to 00b (adjustment is not performed).

5.3.4 Interrupt Sources

There are three interrupt sources in the RTC. **Table 5.3-4** lists interrupt sources for the RTC.

Table 5.3-4 RTC Interrupt Sources

Name	Interrupt Sources
ALM	Alarm interrupt
PRD	Periodic interrupt
CUP	Carry interrupt

5.3.4.1 Alarm Interrupt (ALM)

This interrupt is generated according to the result of comparison between the alarm registers and RTC counters (for details, refer to **5.3.3.6 Alarm Function**).

Since the interrupt flag may be set to 1 when the settings of the alarm registers match the clock counters, wait for the alarm time settings to be confirmed and set the alarm flag of RSR to 0 again after changing the values of the alarm registers. Once the interrupt flag for the alarm interrupt has been set to 1 and the state has returned to non-matching of the alarm registers and clock counters, the flag will not be set again until there is a further match or the values of the alarm registers are modified again.

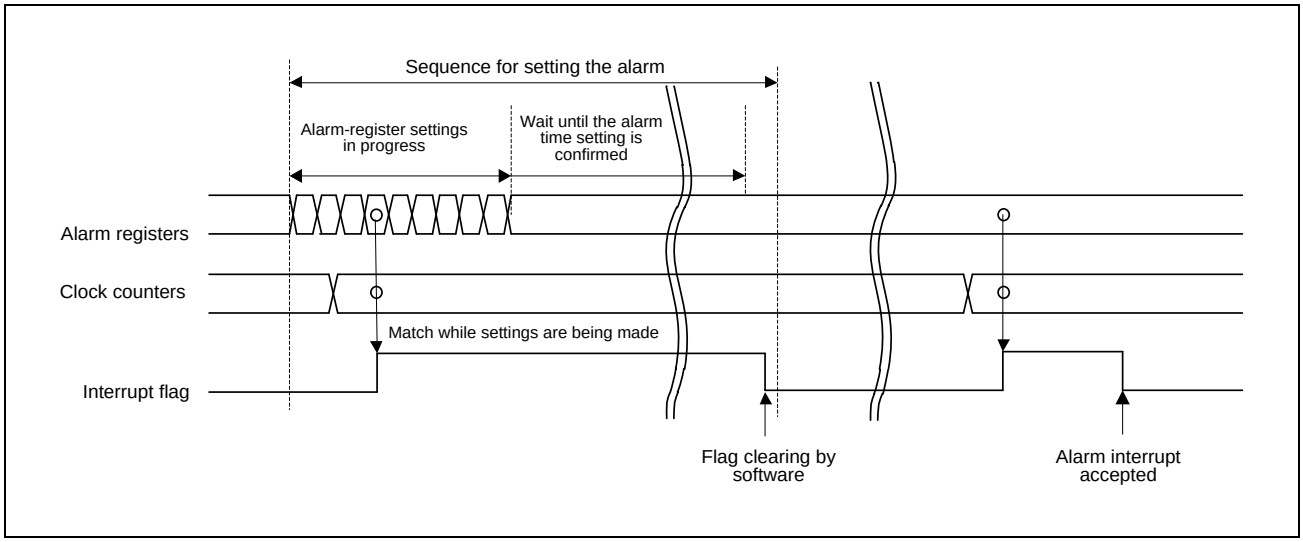


Figure 5.3-8 Timing Chart for the Alarm Interrupt (ALM)

5.3.4.2 Periodic Interrupt (PRD)

This interrupt is generated at intervals of 2 seconds, 1 second, 1/2 seconds, 1/4 seconds, 1/8 seconds, 1/16 seconds, 1/32 seconds, 1/64 seconds, or 1/128 seconds. The interrupt interval can be selected through the RCR1.PES[3:0] bits.

5.3.4.3 Carry Interrupt (CUP)

This interrupt is generated when a carry to the second counter/binary counter 0 occurred or a carry to the R64CNT counter occurred during read access to the 64-Hz counter.

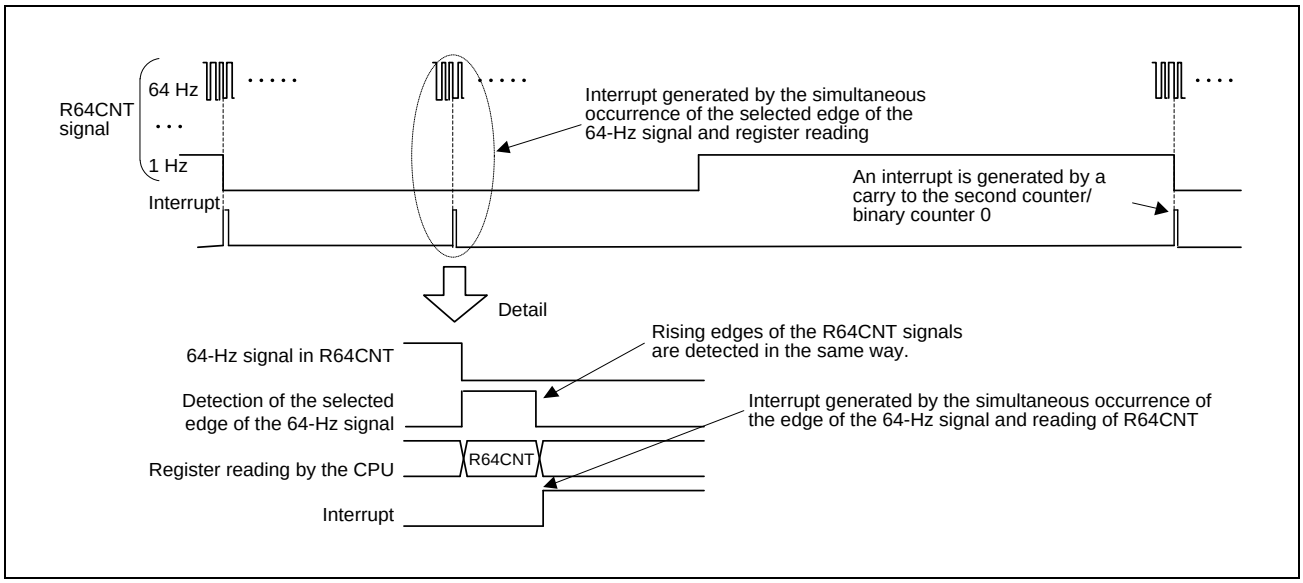


Figure 5.3-9 Carry Interrupt (CUP) Timing Chart

5.3.5 Event Link Output

This module is able to output interrupt source signals as event signals to the event link controller (ELC) to operate modules selected in advance.

(1) Alarm event output

The alarm registers (the year, month, day of the week, date, hour, minute, second/binary counters 3 to 0) selected with the ENB bits set to 1 are compared with the corresponding clock counters. When all values match, an alarm event is output.

Since unintended events as well as interrupts may be generated while the alarm registers are being set, wait for the alarm time settings to be confirmed before making the ELC settings.

(2) Periodic event output

The periodic event signal is output at the interval selected by the setting of the RCR1.PES[3:0] bits (2, 1, 1/2, 1/4, 1/8, 1/16, 1/32, 1/64, or 1/128 seconds).

Since unintended events as well as interrupts may be generated while the event generation interval is selected, select the interval before making the ELC settings. The event generation interval immediately after it has been selected is not guaranteed.

(3) Carry event output

A carry event is output when a carry to the second counter/binary counter 0 has occurred or reading of the 64-Hz counter has coincided with a carry to the R64CNT counter.

NOTE

If event linking from the RTC is to be used, only make the ELC settings after making the RTC settings (initialization, time settings, etc.). Making the RTC settings after the ELC settings can lead to the output of unexpected event signals.

5.3.5.1 Interrupt Handling and Event Linking

Each of the alarm, periodic, and carry interrupts of the RTC has an enable bit to control enabling or disabling of the interrupt signal. An interrupt request signal is output for the CPU when an interrupt source condition is satisfied while the setting of the corresponding enable bit is enabled.

In contrast, an event link output signal is sent to other modules as an event signal via the ELC when an interrupt source is generated, regardless of the setting of the corresponding interrupt enable bit.

NOTE

Although alarm and periodic interrupts can still be output during software standby, the periodic event signals for the ELC are not output.

5.3.6 Usage Notes

5.3.6.1 Register Writing during Counting

The following registers should not be written to during counting (while the RCR2.START bit = 1).

RSECCNT/BCNT0, RMINCNT/BCNT1, RHRCNT/BCNT2, RDAYCNT, RWKCNT/BCNT3, RMONCNT, RYRCNT, RCR1.RTCOS, RCR2.RTCOE, RCR2.HR24

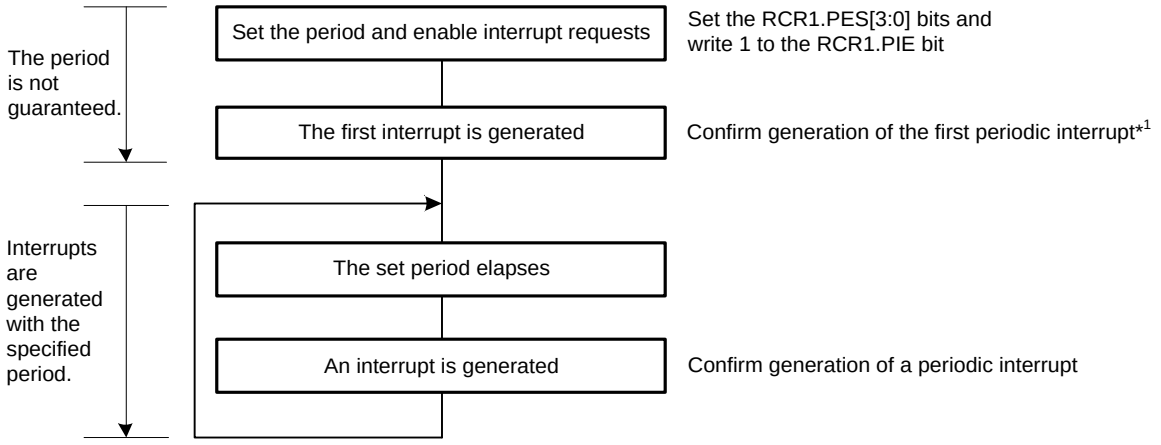
The counter must be stopped before writing to any of the above registers.

5.3.6.2 Use of Periodic Interrupts

The procedure for using periodic interrupts is shown in **Figure 5.3-10**.

The generation and period of the periodic interrupt can be changed by the setting of the RCR1.PES[3:0] bits. However, since the prescaler, R64CNT, and RSECCNT/BCNT0 are used to generate interrupts, the interrupt period is not guaranteed immediately after setting of the RCR1.PES[3:0] bits.

Furthermore, stopping/restarting or resetting counter operation, reset by RTC software, and the 30-second adjustment by changing the RCR2 value affects the interrupt period. When the time error adjustment function is used, the interrupt generation period after adjustment is added or subtracted according to the adjustment value.



Note 1. When an interrupt generation period is changed while the periodic interrupt is used, an interrupt may be generated at the completion of the setting. If the interrupt is generated immediately after the setting, the period is not guaranteed for two interrupts including the current interrupt.

Figure 5.3-10 Using Periodic Interrupt Function

5.3.6.3 Transitions to Low Power Consumption Modes after Setting Registers

A transition to a low power consumption state (software standby mode) during writing to or updating of an RTC register might destroy the register's value. After setting a register, confirm that the setting is in place before initiating a transition to a low power consumption state.

5.3.6.4 Notes on Writing to and Reading from Registers

- When reading a counter register such as the second counter after having written to the counter register, follow the procedure in **5.3.3.5 Reading 64-Hz Counter and Time**.
- The value written to the count registers, alarm registers, year alarm enable register, bits RCR2.AADJE, AADJP, and HR24, r is reflected when four read operations are performed after writing.
- The values written to the RTC Statas Register and RTC Control Register 1.CIE bits can be read immediately after writing.
- To read the value from the timer counter after return from a reset, or software standby mode, wait for 1/128 second while the clock is operating (RCR2.START bit = 1b).
- After a reset is generated, write to the RTC register when six cycles of the count source have elapsed.

5.3.6.5 Changing the Count Mode

When changing the count mode (calendar/binary), set the RCR2.START bit to 0b, stop counting operation, then start again from the initial settings. For details on the initial settings, refer to **5.3.3.1 Outline of Initial Settings of Registers after Power On**.

5.3.6.6 Initialization Procedure When the RTC is Not to be Used

Registers in the RTC are not initialized by a reset. Accordingly, depending on the initial state, the generation of an unintentional interrupt request or operation of the counter may lead to increased power consumption.

For products that do not require a RTC, initialize the registers by following the initialization procedure shown in **Figure 5.3-11**.

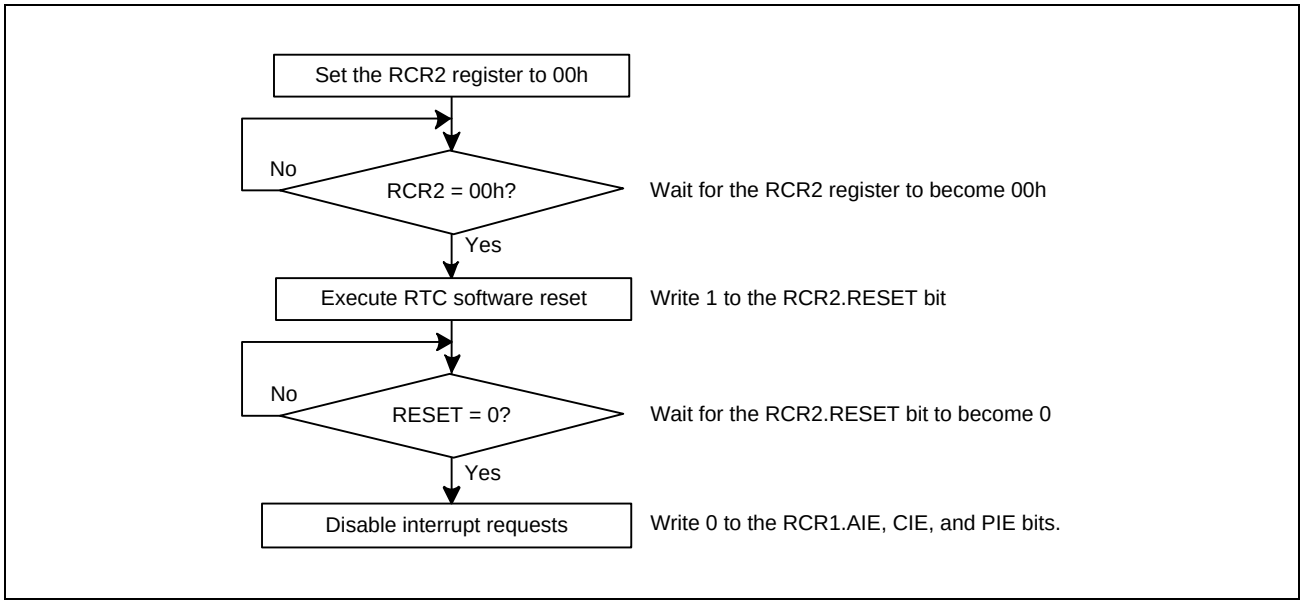


Figure 5.3-11 Initialization Procedure

SECTION 5 TIMER

5.4 Watchdog Timer (WDT)

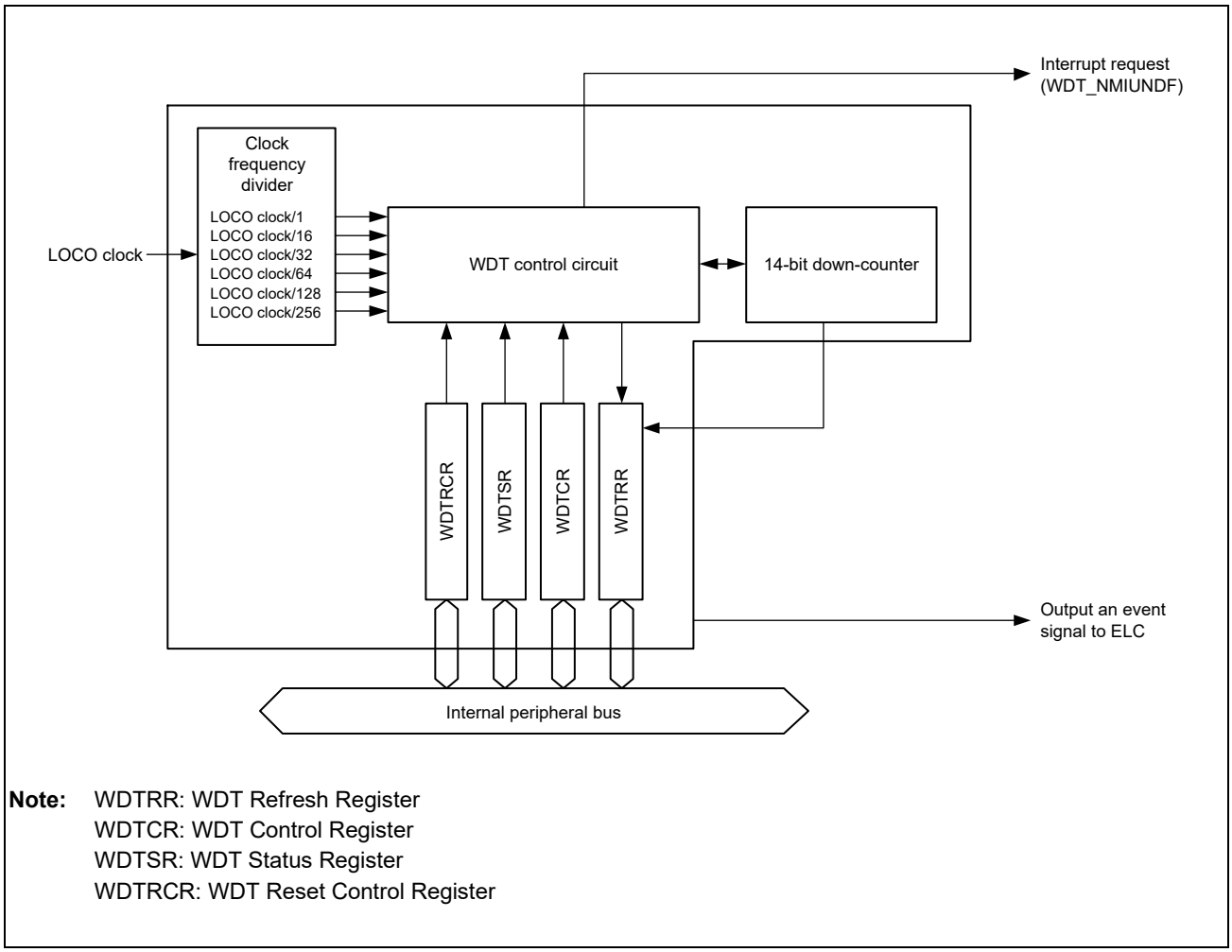
5.4.1 Overview

The Watchdog Timer (WDT) is a 14-bit down counter that can be used to reset this LSI when the counter underflows because the system has run out of control and is unable to refresh the WDT. In addition, the WDT can be used to generate a non-maskable interrupt or an underflow interrupt.

Table 5.4-1 lists the WDT specifications and **Figure 5.4-1** shows a block diagram.

Table 5.4-1 WDT Specifications

Parameter	Specifications
Number of channels	4 channels
Count source	LOCO clock (WDT_0_clk_loco, WDT_1_clk_loco, WDT_2_clk_loco, WDT_3_clk_loco)
Clock division ratio	Division by 1, 16, 32, 64, 128, or 256
Counter operation	Counting down using a 14-bit down-counter
Condition for starting the counter	Counting is started with a refresh by writing to the WDTRR register
Conditions for stopping the counter	• Reset (the down-counter and other registers return to their initial values) • A counter underflows or a refresh error is generated • CA55, CM33 debug state
Window function	Window start and end positions can be specified (refresh-permitted and refresh-prohibited periods)
Interrupt sources	• Down-counter underflows • Refreshing outside the refresh-permitted period (refresh error)
Reading of the counter value	The down-counter value can be read by the WDTSR register
Event link function (output)	• Down-counter underflow event output • Refresh error event output
Output signal (internal signal)	• Interrupt request output



Note: WDTRR: WDT Refresh Register
WDTCR: WDT Control Register
WDTSR: WDT Status Register
WDTRCR: WDT Reset Control Register

Figure 5.4-1 WDT Block Diagram

5.4.2 Registers

The register addresses of WDT are given as offsets from the individual base addresses <WDTn_base>. The register base addresses of each WDT are listed in the following table.

Table 5.4-2 Register Base Addresses

Base Address Name	Base Address	Function
<WDT0_base>	0_11C0_0400 (51C0_0400*1, 41C0_0400*2)	WDT for CM33
<WDT1_base>	0_1440_0000 (5440_0000*1, 4440_0000*2)	WDT for CA55
<WDT2_base>	0_1300_0000 (5300_0000*1, 4300_0000*2)	WDT for Other 0
<WDT3_base>	0_1300_0400 (5300_0400*1, 4300_0400*2)	WDT for Other 1

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

5.4.2.1 List of Registers (m = 0 to 3)

The WDT registers and the memory addresses are listed in the following table.

For the actual addresses, the offset values indicated in this table are added to the base addresses.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
WDT Refresh Register	WDTm_WDTRR	FFh	00h	8
WDT Control Register	WDTm_WDTCR	33F3h	02h	16
WDT Status Register	WDTm_WDTSR	0000h	04h	16
WDT Reset Control Register	WDTm_WDTRCR	80h	06h	8

5.4.2.2 Register Description

The prefix (WDTm_) of the register names is omitted in this and subsequent sections.

5.4.2.2.1 WDT Refresh Register (WDTm_WDTRR)

The WDTRR register refreshes the down-counter of the WDT.

The down-counter of the WDT is refreshed by writing 00h and then writing FFh to WDTRR register (refresh operation) within the refresh-permitted period.

Counting down starts from the value selected by setting the Timeout Period Select bits (WDTCR.TOPS[1:0]) in the WDT Control Register.

When 00h is written, the read value is 00h. When a value other than 00h is written, the read value is FFh. For details of the refresh operation, see **5.4.3.3 Refresh Operation**.

Access Size : 8 bits
Address : <WDTm_base> + 00h
Initial Value : FFh

Bit	7	6	5	4	3	2	1	0
	REFRESH[7:0]							
Initial Value	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	REFRESH[7:0]	FFh	RW	The down-counter is refreshed by writing 00h and then writing FFh to this register.

5.4.2.2.2 WDT Control Register (WDTm_WDTCR)

The WDTCR register is used to set the clock division ratio, and window start and end positions for refresh, and the timeout period until the down-counter underflows in register start mode.

Some constraints apply to writes to the WDTCR register. For details, see **5.4.3.2 Controlling Writes to the WDTCR and WDTRCR Registers**.

Access Size : 16 bits
Address : <WDTm_base> + 02h
Initial Value : 33F3h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	RPSS[1:0]		-	-	RPES[1:0]		CKS[3:0]				-	-	TOPS[1:0]	
Initial Value	0	0	1	1	0	0	1	1	1	1	1	1	0	0	1	1
R/W	R	R	RW	RW	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15, 14	-	All 0	R	Reserved These bits are always read as 0b. The written value will be ignored.
13, 12	RPSS[1:0]	3h	RW	Window Start Position Select 00b: 25% 01b: 50% 10b: 75% 11b: 100% (the window start position is not specified)
11, 10	-	All 0	R	Reserved These bits are always read as 0b. The written value will be ignored.
9, 8	RPES[1:0]	3h	RW	Window End Position Select 00b: 75% 01b: 50% 10b: 25% 11b: 0% (the window end position is not specified)
7 to 4	CKS[3:0]	Fh	RW	Clock Division Ratio Select 0000b: LOCO clock 0010b: LOCO clock/16 0011b: LOCO clock/32 0100b: LOCO clock/64 1111b: LOCO clock/128 0101b: LOCO clock/256 Others: Setting prohibited
3, 2	-	All 0	R	Reserved These bits are always read as 0b. The written value will be ignored.
1, 0	TOPS[1:0]	3h	RW	Timeout Period Select 00b: 1024 cycles (03FFh) 01b: 4096 cycles (0FFFh) 10b: 8192 cycles (1FFFh) 11b: 16384 cycles (3FFFh)

TOPS[1:0] bits (Timeout Period Select)

The TOPS[1:0] bits select the timeout period, the period until the down-counter underflows, from 1024, 4096, 8192, and 16384 cycles, taking the divided clock specified in the CKS[3:0] bits as 1 cycle. After the down-counter is refreshed, the combination of the CKS[3:0] and TOPS[1:0] bits determines the number of LOCO clock cycles until the counter underflows.

Table 5.4-3 lists the relationship between the CKS[3:0] and TOPS[1:0] bit settings, the timeout period, and the number of LOCO clock cycles.

Table 5.4-3 Timeout Period Settings

CKS[3:0] Bits	TOPS[1:0] Bits	Clock Division Ratio	Timeout Period (Number of Cycles)	LOCO Clock Cycles
0000b	00b	LOCO clock	1024	1024
	01b		4096	4096
	10b		8192	8192
	11b		16384	16384
0010b	00b	LOCO clock/16	1024	16384
	01b		4096	65536
	10b		8192	131072
	11b		16384	262144
0011b	00b	LOCO clock/32	1024	32768
	01b		4096	131072
	10b		8192	262144
	11b		16384	524288
0100b	00b	LOCO clock/64	1024	65536
	01b		4096	262144
	10b		8192	524288
	11b		16384	1048576
1111b	00b	LOCO clock/128	1024	131072
	01b		4096	524288
	10b		8192	1048576
	11b		16384	2097152
0101b	00b	LOCO clock/256	1024	262144
	01b		4096	1048576
	10b		8192	2097152
	11b		16384	4194304

CKS[3:0] bits (Clock Division Ratio Select)

The CKS[3:0] bits specify the division ratio of the clock used for the down-counter. The division ratio can be selected from the LOCO clock divided by 1, 16, 32, 64, 128, and 256. Combined with the TOPS[1:0] bit setting, this allows the WDT to be configured to a count period between 1024 and 4194304 LOCO clock cycles.

RPES[1:0] bits (Window End Position Select)

The RPES[1:0] bits specify the window end position that indicates the refresh-permitted period. 75%, 50%, 25%, or 0% of the timeout period can be selected for the window end position. Set the window end position to a value less than the value for the window start position (window start position > window end position). If the window end position is greater than the window start position, only the window start position setting is enabled and the window end position is set to 0%.

RPSS[1:0] bits (Window Start Position Select)

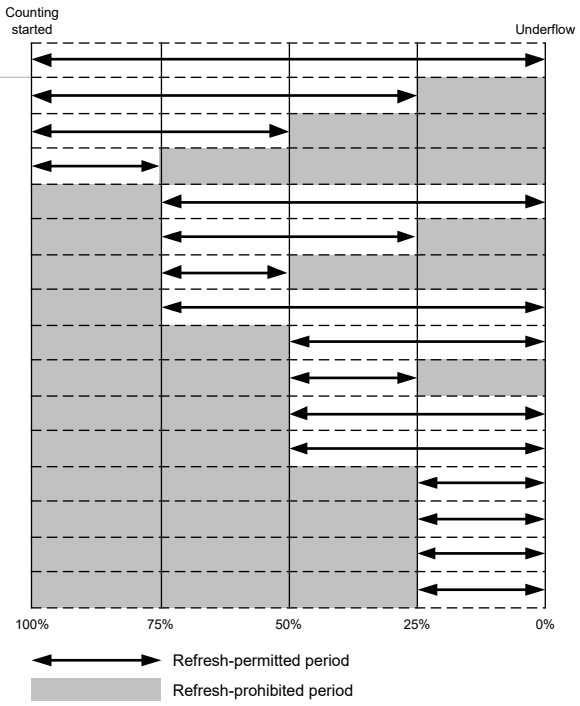
The RPSS[1:0] bits specify the window start position that indicates the refresh-permitted period. 100%, 75%, 50%, or 25% of the timeout period can be selected for the window start position. Set the window start position to a value greater than the value for the window end position. If the window start position is set to a value less than or equal to the window end position, the window start position setting is enabled and the window end position is set to 0%.

Table 5.4-4 lists the counter values for the window start and end positions, and **Figure 5.4-2** shows the refresh-permitted period set in the RPSS[1:0], RPES[1:0], and TOPS[1:0] bits.

Table 5.4-4 Relationship between the Timeout Period and Window Start and End Counter Values

	Timeout Period		Window Start and End Counter Value			
TOPS[1:0]	Cycles	Counter Value	100%	75%	50%	25%
00b	1024	03FFh	03FFh	02FFh	01FFh	00FFh
01b	4096	0FFFh	0FFFh	0BFFh	07FFh	03FFh
10b	8192	1FFFh	1FFFh	17FFh	0FFFh	07FFh
11b	16384	3FFFh	3FFFh	2FFFh	1FFFh	0FFFh

RPSS[1:0] bits		RPES[1:0] bits		Window	
b13	b12	b9	b8	Start (%)	End (%)
1	1	1	1	100	0
		1	0		25
		0	1		50
		0	0		75
1	0	1	1	75	0
		1	0		25
		0	1		50
		0	0		75
0	1	1	1	50	0
		1	0		25
		0	1		50
		0	0		75
0	0	1	1	25	0
		1	0		25
		0	1		50
		0	0		75



Note: If window end setting ≥ window start setting, the window end setting is set to 0%.

Figure 5.4-2 RPSS[1:0] and RPES[1:0] Bits Settings and Refresh-permitted Period

5.4.2.2.3 WDT Status Register (WDTm_WDTSR)

The WDTSR register indicates the counter value of the down-counter and the status of whether an underflow or refresh error occurred in the down-counter.

Access Size : 16 bits
Address : <WDTm_base> + 04h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	REFEF	UNDFF	CNTVAL[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	REFEF	0h	RW	Refresh Error Flag 0b: No refresh error occurred 1b: Refresh error occurred
14	UNDFF	0h	RW	Underflow Flag 0b: No underflow occurred 1b: Underflow occurred
13 to 0	CNTVAL[13:0]	00h	R	Down-Counter Value Value counted by the down-counter

CNTVAL[13:0] bits (Down-Counter Value)

Read the CNTVAL[13:0] bits to confirm the value of the down-counter. The read value might differ from the actual count by 1.

UNDFF flag (Underflow Flag)

Read the UNDFF flag to confirm whether an underflow occurred in the counter. A value of 1 indicates that the down counter underflowed. Write 0 to the flag to set the value to 0. Writing 1 has no effect.

Clearing the UNDFF flag requires (N + 2) LOCO clock cycles and two WDT_n_clkp (n = 0 to 3) cycles. Furthermore, the clearing of this flag is ignored for (N + 2) LOCO clock cycles from an underflow. N is specified in the WDTCR.CKS[3:0] bits as follows:

- When WDTCR.CKS[3:0] = 0h, N = 1
- When WDTCR.CKS[3:0] = 2h, N = 16
- When WDTCR.CKS[3:0] = 3h, N = 32
- When WDTCR.CKS[3:0] = 4h, N = 64
- When WDTCR.CKS[3:0] = Fh, N = 128
- When WDTCR.CKS[3:0] = 5h, N = 256

REFEF flag (Refresh Error Flag)

Read the REFEF flag to confirm whether a refresh error occurred, indicating that a refresh operation was performed during a prohibited period. A value of 1 indicates that a refresh error occurred. Write 0 to the flag to set the value to 0. Writing 1 has no effect.

Clearing the REFEF flag requires (N + 2) LOCO clock cycles and two WDT_n_clkp (n = 0 to 3) cycles. Furthermore, the clearing of this flag is ignored for (N + 2) LOCO clock cycles from a refresh error. N is specified in the WDTCR.CKS[3:0] bits as follows:

- When WDTCR.CKS[3:0] = 0h, N = 1

- When WDTCR.CKS[3:0] = 2h, N = 16
- When WDTCR.CKS[3:0] = 3h, N = 32
- When WDTCR.CKS[3:0] = 4h, N = 64
- When WDTCR.CKS[3:0] = Fh, N = 128
- When WDTCR.CKS[3:0] = 5h, N = 256

5.4.2.2.4 WDT Reset Control Register (WDTm_WDTRCR)

The WDTRCR register controls interrupt request output by a WDT down-counter underflow.

Some constraints apply to writes to the WDTRCR register. For details, see **5.4.3.2 Controlling Writes to the WDTCR and WDTRCR Registers**.

Access Size : 8 bits
Address : <WDTm_base> + 06h
Initial Value : 80h

Bit	7	6	5	4	3	2	1	0
	RSTIRQS	-	-	-	-	-	-	-
Initial Value	1	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	RSTIRQS	1h	RW	Reset Interrupt Request Select 0b: Error notification to the ICU is permitted 1b: Error notification to the ICU is not performed
6 to 0	-	All 0	R	Reserved These bits are always read as 0b. The written value will be ignored.

CAUTION

In this LSI, always set RSTIRQS = 0 when using the WDT.

5.4.3 Operation

5.4.3.1 Register Start Mode

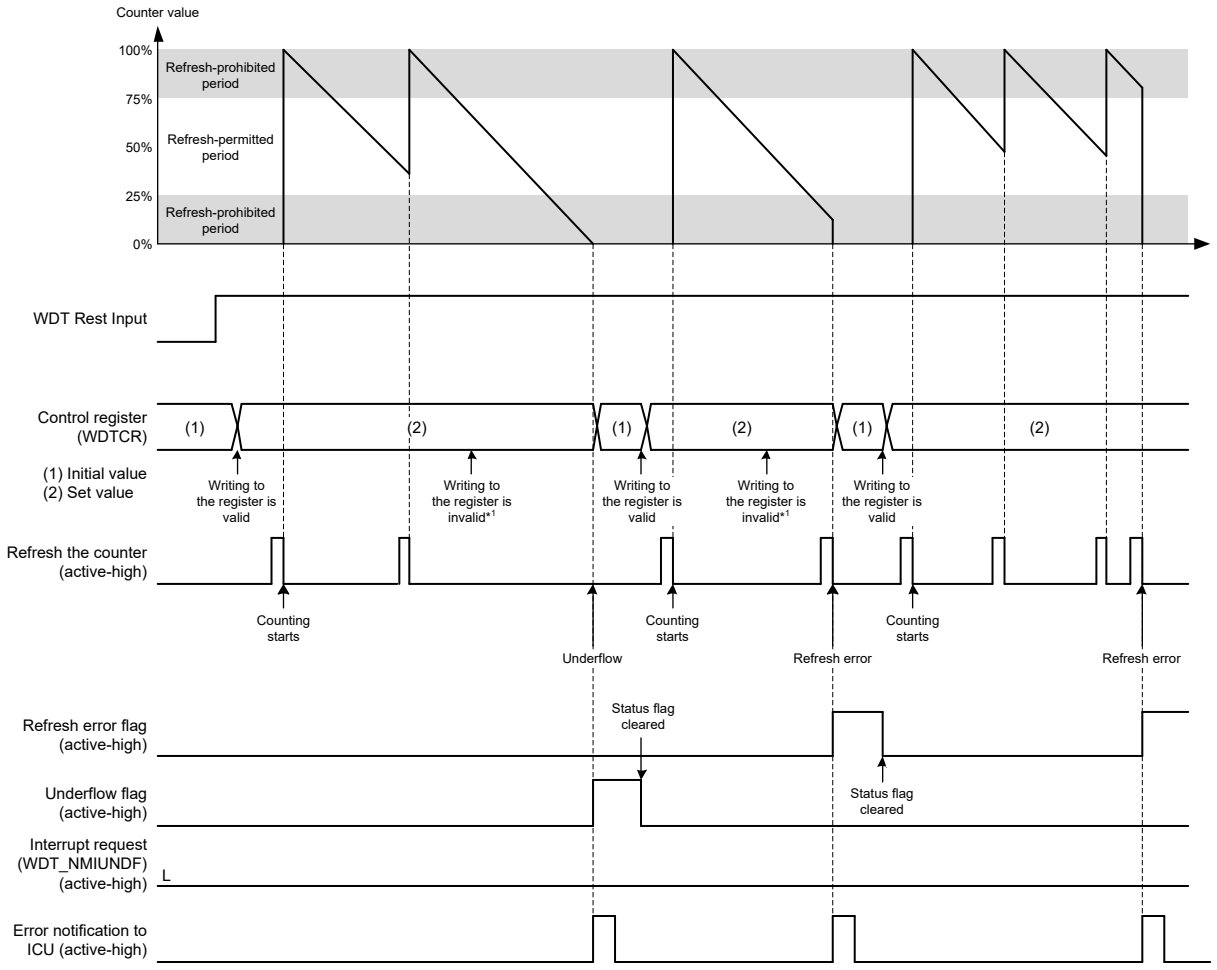
- Clock division ratio
- Window start and end positions
- Timeout period in the WDTCR register
- Interrupt request output in the WDTRCR register

The WDT refresh register (WDTRR) refreshes the down counter. As a result, the down count starts at the value set by the timeout period selection bit (WDTCR.TOPS[1:0]).

As long as the counter is refreshed in the refresh-permitted period, the value in the counter is reset each time the counter is refreshed, and the counter continues to count down. The WDT does not output the non-maskable interrupt request/interrupt request as long as the down-counter continues to count. However, if the down-counter underflows because the down-counter cannot be refreshed due to a program runaway, or if a refresh error occurs because the counter was refreshed outside the refresh-permitted period, the WDT outputs a non-maskable interrupt request/interrupt request (WDT_NMIUNDF). The interrupt request output can be enabled in the WDT Reset Interrupt Request Select bit (WDTRCR.RSTIRQS).

Figure 5.4-3 shows an example of operation under the following conditions:

- Reset Interrupt Request bit (RSTIRQS): 0b
- The window start position is 75% (WDTCR.RPSS[1:0] = 10b)
- The window end position is 25% (WDTCR.RPES[1:0] = 10b)



Note: See **5.4.3.2 Controlling Writes to the WDTCR and WDTRCR Registers**.
Reset pin = QRESN

Figure 5.4-3 Operation Example in Register Start Mode

5.4.3.2 Controlling Writes to the WDTCR and WDTRCR Registers

Writing to the WDT Control Register (WDTCR) or WDT Reset Control Register (WDTRCR) is possible once between the release from the reset state and the first refresh operation.

After a refresh (counting starts) or a write to the WDTCR or WDTRCR register, the protection signal in the WDT becomes 1 to protect the WDTCR and WDTRCR registers against subsequent write attempts. This protection is released by the reset source of the WDT. With other reset sources, the protection is not released.

Figure 5.4-4 shows control waveforms produced in response to writing to the WDTCR.

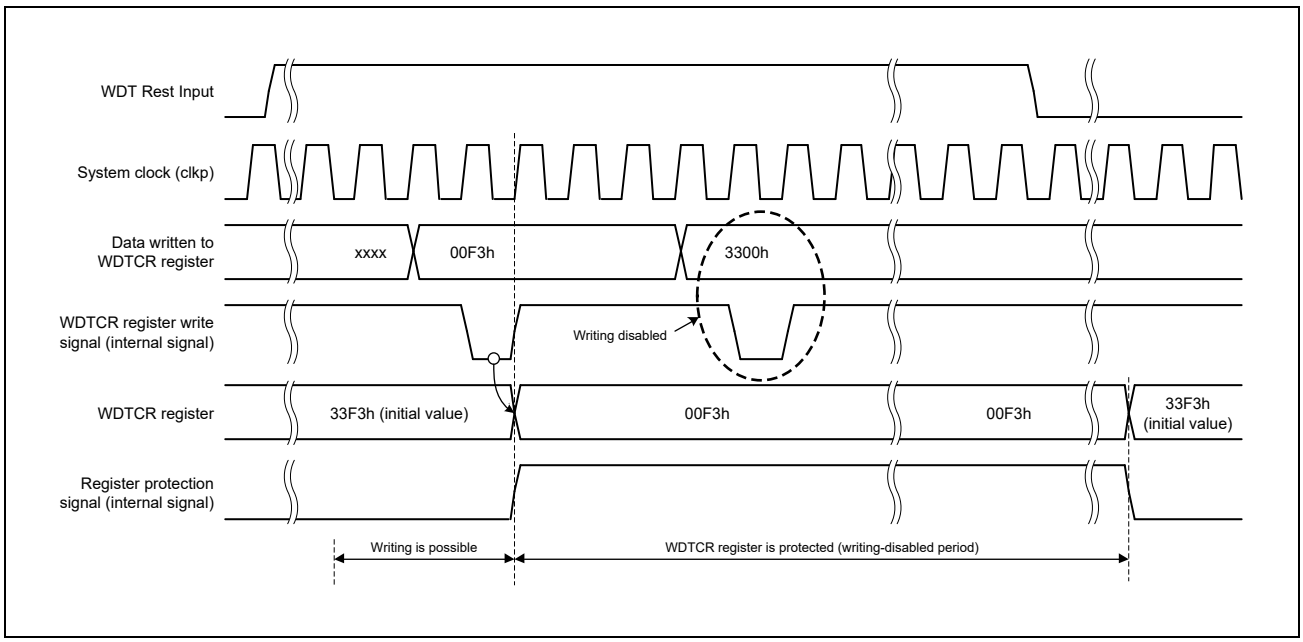


Figure 5.4-4 Control Waveforms Produced in Response to Writes to the WDTCR Register

5.4.3.3 Refresh Operation

To refresh the down-counter and start the counting operation, write to the WDT Refresh Register (WDTRR) in the order of values from 00h to FFh. If a value other than FFh is written after 00h, the down-counter is not refreshed. If an invalid value is written, refreshing is performed normally by writing to the WDTRR register in the order of values from 00h to FFh.

Correct refreshing is also performed when a register other than WDTRR is accessed or WDTRR is read between writing 00h and writing FFh to WDTRR. Writes to refresh the counter must be made within the refresh-permitted period, and this is determined by the FFh write. For this reason, correct refreshing is performed even when 00h is written outside the refresh-permitted period.

[Example write sequences that are valid for refreshing the counter]

00h → FFh

00h ((n − 1)th time) → 00h (nth time) → FFh

00h → access to another register or read from WDTRR → FFh

[Example write sequences that are invalid for refreshing the counter]

23h (a value other than 00h) → FFh

00h → 54h (a value other than FFh)

00h → AAh (00h and a value other than FFh) → FFh

After FFh is written to the WDT Refresh Register (WDTRR), refreshing the down-counter requires up to 4 cycles of the signal for counting. To meet this requirement, complete writing FFh to WDTRR 4 count cycles before the ms.

Figure 5.4-5 shows the WDT refresh-operation waveforms when the clock division ratio is LOCO clock/1.

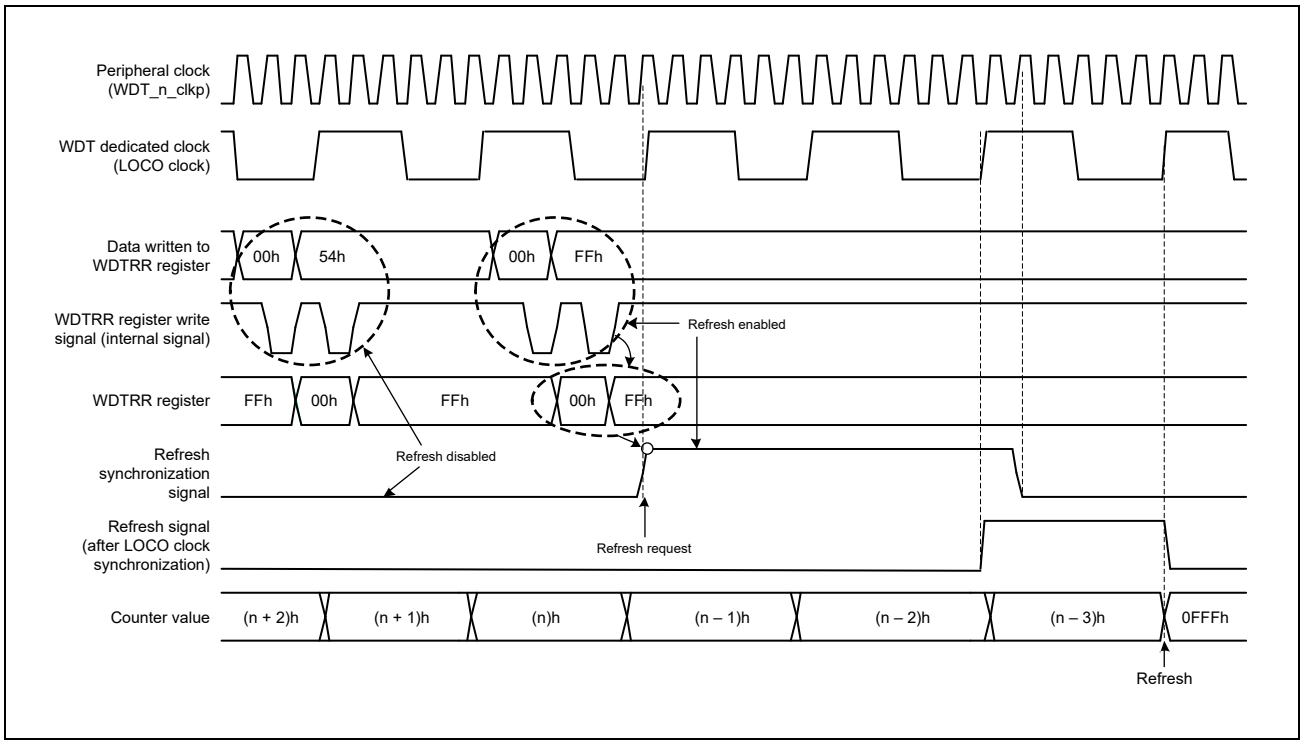


Figure 5.4-5 WDT Refresh Operation Waveforms when WDTCR.CKS[3:0] = 4h and WDTCR.TOPS[1:0] = 01b

5.4.3.4 Status Flags

The refresh error (WDTSR.REFEF) and underflow (WDTSR.UNDFF) flags retain the source of the interrupt request from the WDT. After a release from the interrupt request generation, read the WDTSR.REFEF and WDTSR.UNDFF flags to check for the interrupt source. For each flag, writing 0 clears the bit. Writing 1 has no effect. Leaving the status flags unchanged does not affect operation. If the flags are not cleared at the next interrupt request from the WDT, the earlier interrupt source is cleared and the new interrupt source is written. For the time period between when 0 is written in each flag and when its value is reflected, see **5.4.2.2.3 WDT Status Register (WDTm_WDTSR)**.

5.4.3.5 Interrupt Sources

When the Reset Interrupt Request Select bit (WDTRCR.RSTIRQS) is set to 0, an interrupt signal (WDT_NMIUNDF) is generated upon a counter underflow or generation of a refresh error. For details, see **4.6.1 Interrupt Controller Unit (ICU)**.

Table 5.4-5 WDT Interrupts

Interrupt name	Signal (WDTn)	Function	Interrupt to CPU	Error Interrupt
WDT_NMIUNDF	WDT_CM33_iwdt_nmiundf_n (WDT0)	WDT0 interrupt	Not possible	Possible
	WDT_CA55_iwdt_nmiundf_n (WDT1)	WDT1 interrupt	Not possible	Possible
	WDT_Other_0_iwdt_nmiundf_n (WDT2)	WDT2 interrupt	Possible	Possible
	WDT_Other_1_iwdt_nmiundf_n (WDT3)	WDT3 interrupt	Possible	Possible

5.4.3.6 Reading the Down-Counter Value

The WDT stores the counter value in the down-counter value bits (WDTSR.CNTVAL[13:0]) of the WDT Status Register. Check these bits to obtain the counter value. The read value of the down-counter might differ from the actual count by one.

Figure 5.4-6 shows the processing for reading the WDT down-counter value when the clock division ratio is LOCO clock/1.

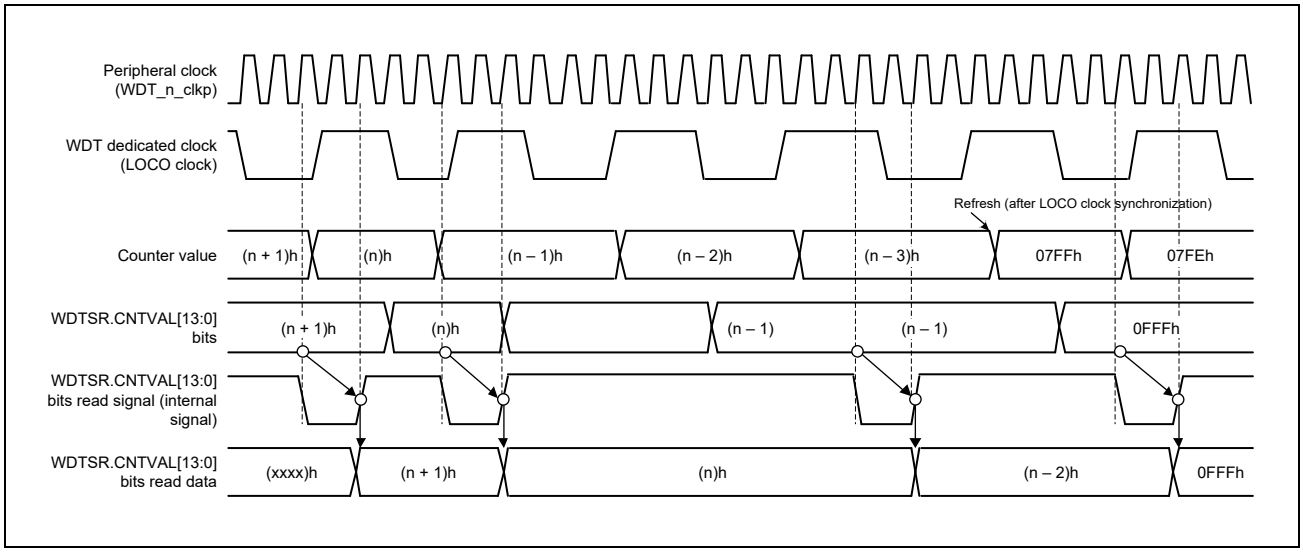


Figure 5.4-6 Processing for Reading WDT Down-Counter Value when WDTCR.CKS[3:0] = 4h and WDTCR.TOPS[1:0] = 01b

5.4.4 Output to the Event Link Controller (ELC)

The WDT is capable of a link operation for the previously specified module when the interrupt request signal is used as an event signal by the ELC. An event signal is output by a counter underflow or refresh error.

An event signal is output regardless of the setting of the WDTRCR.RSTIRQS bit.

An event signal can also be output when the next interrupt source is generated while the Refresh Error flag (WDTSR.REFEF) or Underflow flag (WDTSR.UNDFF) is 1.

5.4.5 Example WDT1 Setting for Error System Reset

A setting example for WDT1 is shown below.

Table 5.4-6 WDT1 Setting Procedure for Error System Reset

Step	Description	Register	Setting Value
0	Confirm the error status, and if an error factor is applied, perform the appropriate processing to suit the system.	—	—
1	Clear the error factor.*1	ICU_ERINTA55CLR0	1000_0000h
2	Set the reset vector address.	SYS_ACPU_CFG_RVAL0	8000_0000h
3	When the WDT generates a CA55 cold reset request • A CA55 cold reset is applied. • The WDTUDFCA signal is asserted.	CPG_ERRORRST_SEL1	***A_***Ah
4	The error system reset is executed when WDT_CA55_iwdt_nmiundf_n is asserted.	CPG_ERRORRST_SEL2	0002_0002h
5	Clear the error flag of WDT_CA55_iwdt_nmiundf_n.	CPG_ERROR_RST2	0002_0002h
6	• Timeout period: 16384 cycles • Clock division ratio: LOCO clock /128 • Window end position: 0% • Window start position: 100%	WDT1_WDTCR*2	33F3h
7	Enable WDT1 interrupt output to the ICU.	WDT1_WDTRCR*2	00h
8	Clear an underflow flag and a refresh error flag.	WDT1_WDTSR	0000h

Remarks: Writing to bits with * is prohibited (read-modify-write operations are required).

Note 1. The error factors are retained by default. Be sure to clear the factors before setting WDT1. Interrupts should be set after clearing the factors. See **4.6 Interrupt Controller** for details.

Note 2. These registers have write restrictions. For details, see **5.4.3.2 Controlling Writes to the WDTCR and WDTRCR Registers**.

SECTION 5 TIMER

5.5 General Timer (GTM)

5.5.1 Functional Overview

The general timer has the following features.

- Two operating modes
  - Interval timer mode
  - Free-running comparison mode
- Choice between startup of the ELC by a compare match and generation of an interrupt

5.5.1.1 Features of GTM

■ **Channels**

This product has the following number of channels of the General timer.

Table 5.5-1 Channels of General Timer

General Timer	
Number of channels	8
Name	OSTMn

Note: n = 0 to 7

■ **Meaning of n**

Throughout this section, the individual channels of the General timer are identified by the index “n” (n = 0 to 7), for example OSTMnCMP for the General timer n output register.

■ **Register address**

The register addresses of the General timer are given as offsets from the individual base addresses <GTMn_base>. The register base addresses of each GTM are listed in the following table.

■ Interrupts

The General timers can generate the following interrupt requests (GTMTINT).

Table 5.5-2 GTM Interrupt Requests

GTM Interrupt Signal	Function	Startup of Direct Memory Access Controller	Function Startup of Event Link Controller
GTM0_GTMTINT	GTM0 interrupt	Not possible	Possible
GTM1_GTMTINT	GTM1 interrupt	Not possible	Possible
GTM2_GTMTINT	GTM2 interrupt	Not possible	Possible
GTM3_GTMTINT	GTM3 interrupt	Not possible	Possible
GTM4_GTMTINT	GTM4 interrupt	Not possible	Possible
GTM5_GTMTINT	GTM5 interrupt	Not possible	Possible
GTM6_GTMTINT	GTM6 interrupt	Not possible	Possible
GTM7_GTMTINT	GTM7 interrupt	Not possible	Possible

5.5.2 Registers

The General timer is controlled by the following registers.

Table 5.5-3 Register Base Addresses

Base Address Name	Base Address
<GTM0_base>	0_1180_0000h (5180_0000h*1, 4180_0000h*2)
<GTM1_base>	0_1180_1000h (5180_1000h*1, 4180_1000h*2)
<GTM2_base>	0_1400_0000h (5400_0000h*1, 4400_0000h*2)
<GTM3_base>	0_1400_1000h (5400_1000h*1, 4400_1000h*2)
<GTM4_base>	0_12C0_0000h (52C0_0000h*1, 42C0_0000h*2)
<GTM5_base>	0_12C0_1000h (52C0_1000h*1, 42C0_1000h*2)
<GTM6_base>	0_12C0_2000h (52C0_2000h*1, 42C0_2000h*2)
<GTM7_base>	0_12C0_3000h (52C0_3000h*1, 42C0_3000h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

5.5.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
OSTM Compare register	OSTMnCMP	0000_0000h	0000h	32
OSTM Counter Register	OSTMnCNT	FFFF_FFFFh	0004h	32
Reserve	-	-	0008h to 000Fh	-
OSTM Count Enable Status Register	OSTMnTE	00h	0010h	8
Reserve	-	-	0011h to 0013h	-
OSTM Count Start Trigger Register	OSTMnTS	00h	0014h	8
Reserve	-	-	0015h to 0017h	-
OSTM Count Stop Trigger Register	OSTMnTT	00h	0018h	8
Reserve	-	-	0019h to 001Fh	-
OSTM Control Register	OSTMnCTL	00h	0020h	8

5.5.2.2 Register Description

5.5.2.2.1 OSTM Compare Register (OSTMnCMP)

Depending on the mode of operation, this register holds the start value of the down-counter or the value for comparison with that of the counter.

Access Size : 32 bits
Address : <GTMm_base> + 00h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CMP[31:0]	0h	RW	In interval timer mode: the start value of the down-counter In free-running comparison mode: the value for comparison

5.5.2.2.2 OSTM Counter Register (OSTMnCNT)

This register indicates the counter value of the timer.

Access Size : 32 bits
Address : <GTMm_base> + 04h
Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNT[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNT[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CNT[31:0]	FFFF_FFFFh*1	R	32-bit counter value

Note 1. The initial value depends on the operating mode of the General timer. Refer to **Table 5.5-4**.

The following table shows the correspondence between operating mode, counting direction and initial value. The initial value is the value read from the counter after a change to the operating mode.

Table 5.5-4 Correspondence between Operating Mode, Counting Direction and Initial Value

Timer Operating Mode	OSTMnCTL.OSTMnMD1	Counting Direction	Initial value
Interval timer mode	0*1	Down	FFFF_FFFFh
Free-running comparison mode	1	Up	0000_0000h

Note 1. Value after reset

5.5.2.2.3 OSTM Count Enable Status Register (OSTMnTE)

This register indicates whether the counter is enabled or disabled.

Access Size : 8 bits
Address : <GTMm_base> + 10h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TE
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	-	All 0	R	Reserved These bits are always read as 0b. The write value should always be 0b.
0	TE	0h	R	This bit indicates whether the counter is enabled or disabled. 0b: Counter disabled 1b: Counter enabled This bit is set to 1b in response to OSTMnTS.OSTMnTS being set to 1b. This bit is reset to 0b in response to OSTMnTT.OSTMnTT being set to 1b.

Note: When OSTMnTE = 0, the counter retains its value.
If the counter is restarted, it restarts counting down from the value in the OSTMnCMP register if it is in interval timer mode, or restarts counting up from the counter value 0000_0000h if it is in free-running comparison mode.

Note: For debugging, the GTM counter can be paused by the counter stop request.

5.5.2.2.4 OSTM Count Start Trigger Register (OSTMnTS)

This register starts the counter.

Access Size : 8 bits
Address : <GTMm_base> + 14h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TS
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	-	All 0	R	Reserved These bits are always read as 0b. The write value should always be 0b.
0	TS	0h	W	The read value is undefined This bit starts the counter. 0b: This setting has no effect. 1b: Starts the counter and sets OSTMnTE.OSTMnTE = 1b. In interval timer mode, a forced restart is executed if this bit is set while OSTMnTE.OSTMnTE = 1b. In free-running comparison mode, setting this bit is ignored as long as OSTMnTE.OSTMnTE = 1b.

5.5.2.2.5 OSTM Count Stop Trigger Register (OSTMnTT)

This register stops the counter.

Access Size : 8 bits
Address : <GTMm_base> + 18h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TT
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	-	All 0	R	Reserved These bits are always read as 0b. The write value should always be 0b.
0	TT	0h	W	The read value is undefined Stops the counter. 0b: This setting has no effect. 1b: Stops the counter and clears the OSTMnTE.OSTMnTE bit.

5.5.2.2.6 OSTM Control Register (OSTMnCTL)

This register specifies the operating mode for the counter and controls enabling/disabling of GTMTINT interrupt requests when counting starts.

Access Size : 8 bits
Address : <GTMm_base> + 20h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	MD1	MD0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	-	All 0	R	Reserved These bits are always read as 0b. The write value should always be 0b.
1	MD1	0h	RW	Specifies the operating mode for the counter. 0b: Interval timer mode 1b: Free-running comparison mode
0	MD0	0h	RW	Controls enabling/disabling of OSTMnTINT interrupt requests when counting starts. 0b: Disables the interrupts when counting starts. 1b: Enables the interrupts when counting starts.

5.5.3 Functional Description

Each General timer is a 32-bit timer/counter.

The settings for operating mode specify the direction of counting (up or down) and the generation of interrupt requests.

5.5.3.1 Block Diagram

The following block diagram shows the main components of GTM.

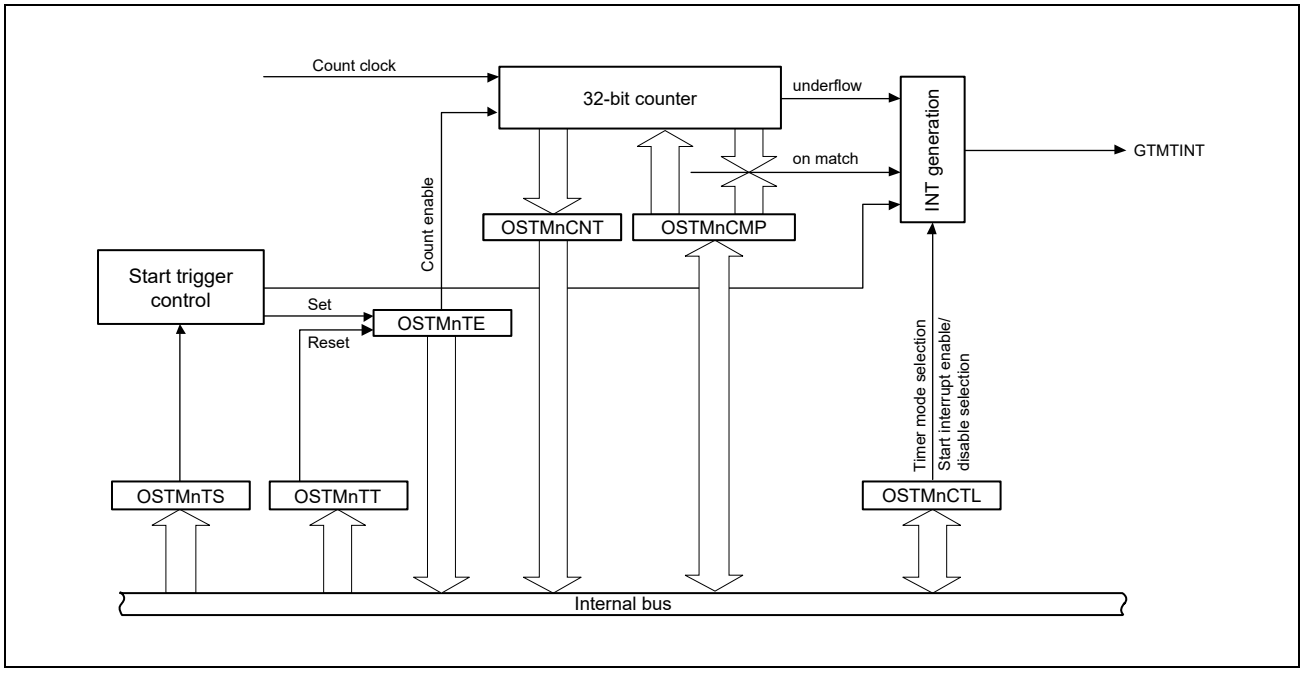


Figure 5.5-1 Block Diagram of GTM

5.5.3.2 Count Clock

The count clocks of GTMn (n = 0 to 7) are listed below.

Table 5.5-5 GTM Clock Signals

GTM Clock Signal	Function
GTM_0_PCLK	GTM0 PCLK
GTM_1_PCLK	GTM1 PCLK
GTM_2_PCLK	GTM2 PCLK
GTM_3_PCLK	GTM3 PCLK
GTM_4_PCLK	GTM4 PCLK
GTM_5_PCLK	GTM5 PCLK
GTM_6_PCLK	GTM6 PCLK
GTM_7_PCLK	GTM7 PCLK

5.5.3.3 Generation of Interrupt Request

A GTMTINT interrupt request is generated whenever the counter reaches 0000_0000h (in interval timer mode) or matches the comparison value (in free-running comparison mode).

An interrupt request can also be generated on starting and restarting of the counter. This is controlled by the OSTMnCTL.OSTMnMD0 bit.

This operation is shown in the following figure.

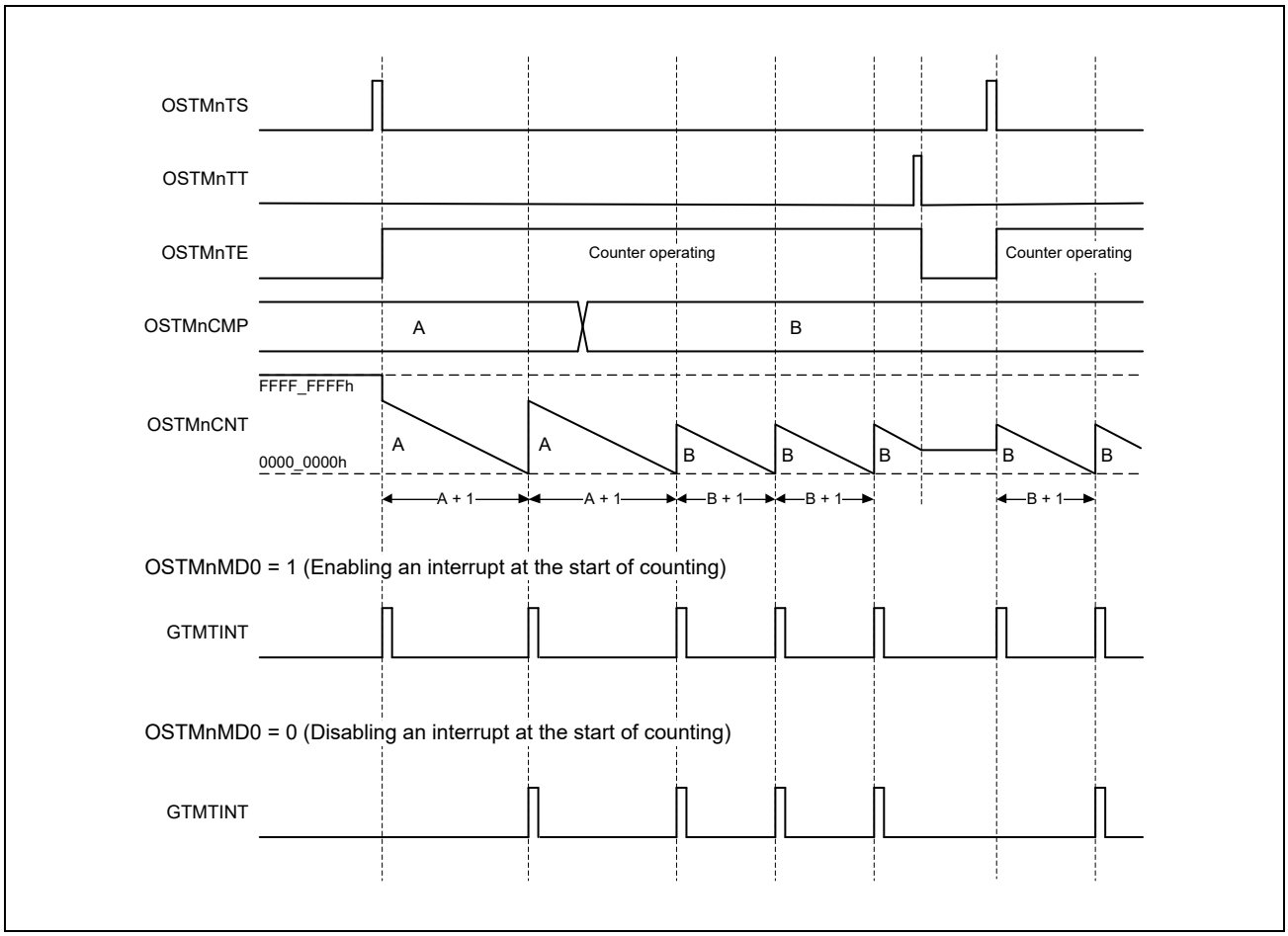


Figure 5.5-2 Generating an Interrupt when Counting Starts (in Interval Timer Mode)

5.5.3.4 Starting and Stopping the Timer

The General timer is started and stopped as follows.

Starting the timer

The timer is started in either of the following way:

- Setting the OSTMnTS.OSTMnTSF bit to 1.
- Status bit OSTMnTE.OSTMnTE is set to 1.

The counter starts to count up or down in accord with the settings for operating mode.

Stopping the timer

Setting the OSTMnTT.OSTMnTT bit to 1 stops the timer.

This also clears the OSTMnTE.OSTMnTE status flag.

5.5.3.5 Interval Timer Mode

Select the interval timer mode when a General timer is to be used as a reference timer for generating interrupt requests at a fixed interval.

5.5.3.5.1 Basic Operation in Interval Timer Mode

In interval timer mode, the timer counts down from the value specified in the OSTMnCMP register. An GTMTINT interrupt request is generated when the counter reaches 0000_0000h.

Select interval timer mode by setting OSTMnCTL.OSTMnMD1 = 0.

New values can be written to the OSTMnCMP register at any time. If it is rewritten during count operation, the counter loads the new OSTMnCMP value when the next 0000_0000h is reached.

Cycles of GTMTINT output

The cycle of GTMTINT output is as follows.

- GTMTINT generation cycle = counter-clock cycle × (OSTMnCMP + 1)

The following figure shows the basic operation of GTM when counter-start interrupts is enabled in interval timer mode.

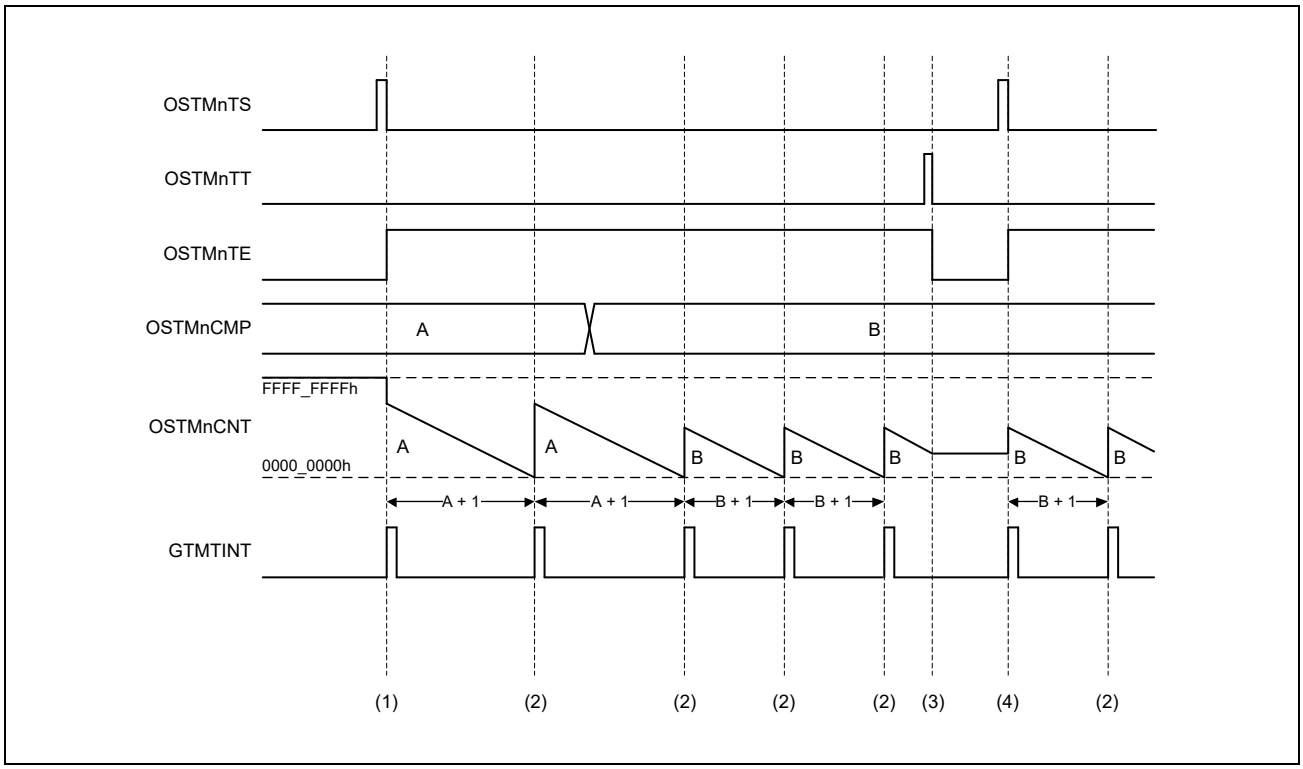


Figure 5.5-3 Timing Diagram of GTM in Interval Timer Mode

The timing diagram above shows the following:

(1) The counter starts counting when OSTMnTS.OSTMnTS = 1. The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter.
The counter starts counting down from the value of OSTMnCMP.

If OSTMnCTL.OSTMnMD0 is 1, GTMTINT interrupt requests are generated at the start of counting. The OSTMnCNT register contains the current value as the counter.

(2) When the counter reaches 0000_0000h, an GTMTINT interrupt request is generated. The counter loads the new start value from OSTMnCMP and continues counting down.

(3) When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter. The counter retains its current value until it is restarted.

(4) When counting is restarted (OSTMnTS.OSTMnTS = 1), the counter loads the new start value from OSTMnCMP and starts counting down.

Forced restart

The counter is forcibly restarted by setting OSTMnTS.OSTMnTS = 1 during counting.

The counter loads the start value from the OSTMnCMP register and continues to count down.

The following figure shows the forced restart of the General timer in interval timer mode, with counter-start interrupts enabled (OSTMnCTL.OSTMnMD0 = 1).

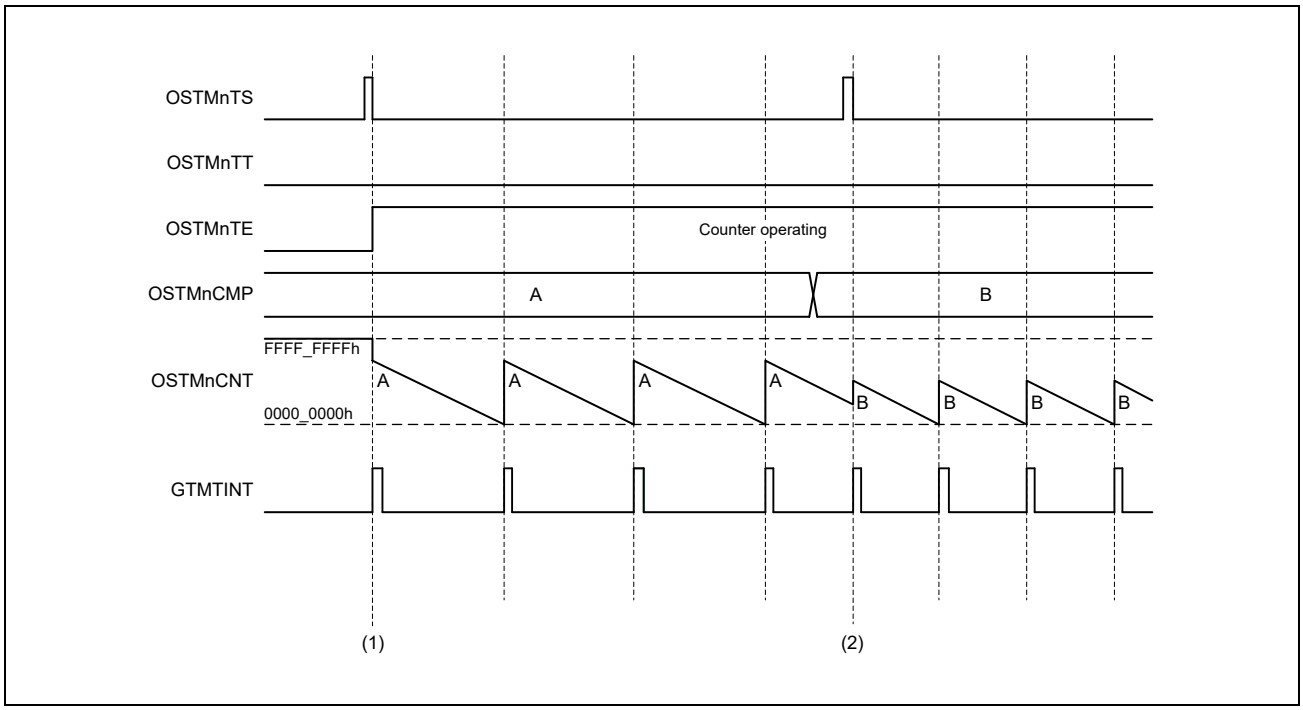


Figure 5.5-4 Timing Diagram of Forced Restart in Interval Timer Mode

Operations shown in the above timing diagram are as follows.

(1) The counter is started and stopped as described under **Figure 5.5-3**.

(2) Setting OSTMnTS.OSTMnTS = 1 restarts the counter while counting is in progress (i.e. while OSTMnTE.OSTMnTE = 1).

The counter immediately restarts counting down, starting with the current value of OSTMnCMP.

When OSTMnCTL.OSTMnMD0 = 1, an GTMTINT interrupt request is generated when counting starts.

5.5.3.5.2 Operation when OSTMnCMP = 0000_0000h

When OSTMnCMP = 0000_0000h, GTM behaves as follows.

- When the counter is enabled, the GTMTINT interrupt request is always set to 1.

The following figure shows operations of GTM when OSTMnCMP = 0000_0000h, and counter-start interrupts are enabled.

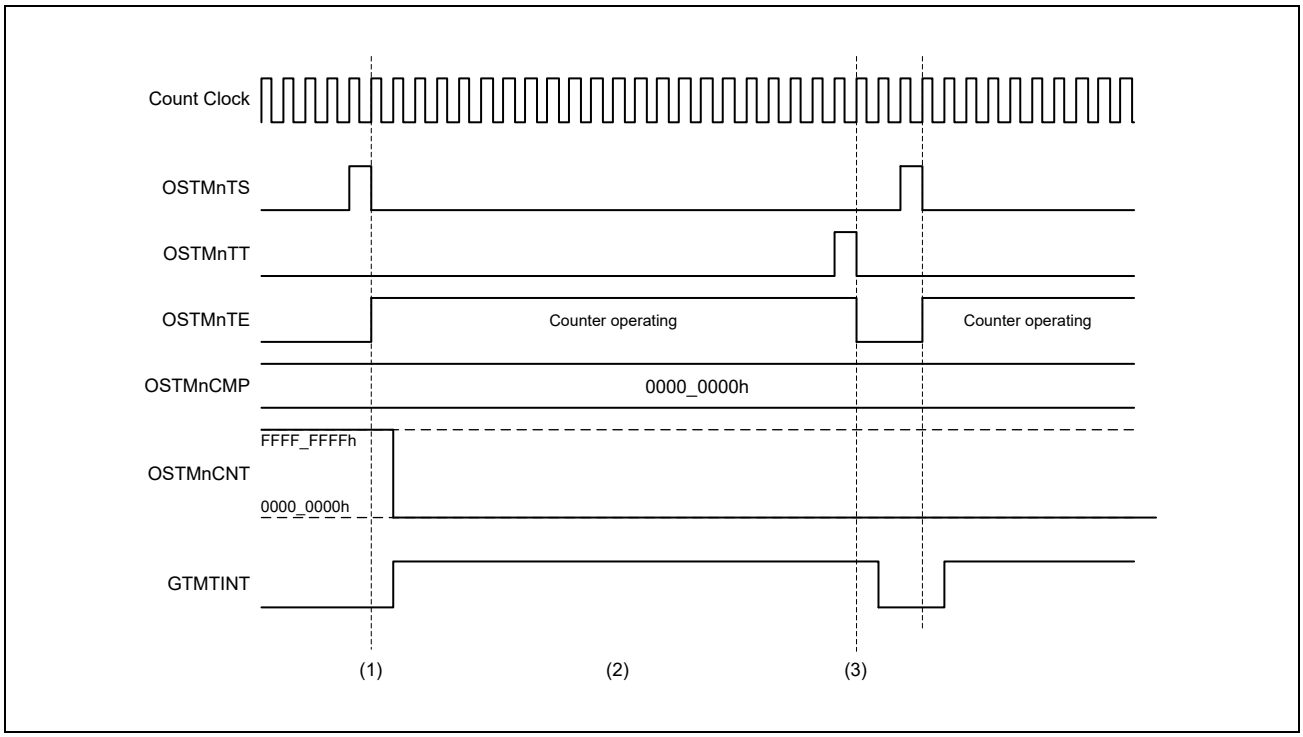


Figure 5.5-5 Timing Diagram when OSTMnCMP = 0000_0000h in Interval Timer Mode

The timing diagram above shows the following operations:

(1) The counter is reloaded with the value in OSTMnCMP as soon as it starts counting, so the value 0000_0000h is retained in OSTMnCMP.

(2) The GTMTINT interrupt request is continuously asserted.

(3) After the counter stops, the GTMTINT interrupt request signal is deasserted.

(4) When interrupts on starting of the counter are disabled, no interrupt is generated when counting starts.

5.5.3.6 Free-Running Comparison Mode

5.5.3.6.1 Basic Operation in Free-Running Comparison Mode

In free-running comparison mode, the counter counts up from 0000_0000h to FFFF_FFFFh. An GTMTINT interrupt request is output when the current value of the counter matches the value of the OSTMnCMP register. The free-running comparison mode is selected by setting the OSTMnCTL.OSTMnMD1 bit to 1.

New values can be written to the OSTMnCMP register at any time.

The following figure shows the basic operation of GTM in free-run compare mode with the start of counting enabled (OSTMnCTL.OSTMnMD0 = 1).

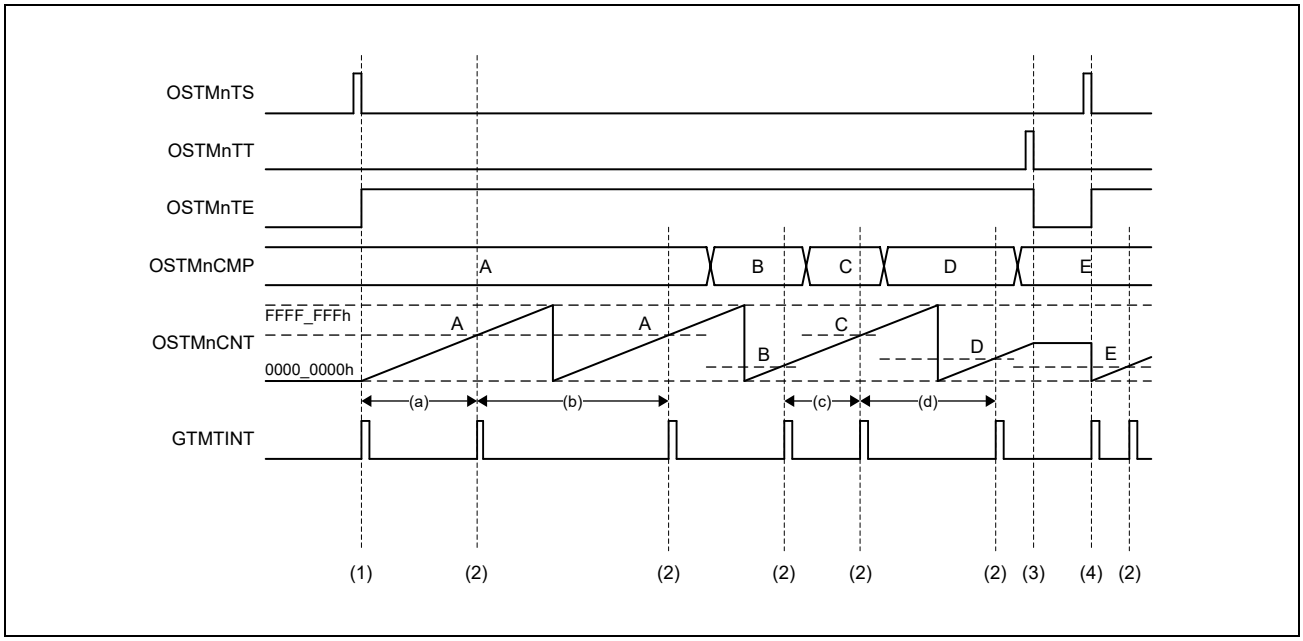


Figure 5.5-6 Timing Diagram of GTM in Free-Run Compare Mode

The timing diagram above shows the following:

(1) The counter starts counting when OSTMnTS.OSTMnTS = 1.
The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter. The counter counts up from 0000_0000h to FFFF_FFFFh. The OSTMnCNT register is the counter, so it contains the current value.
When OSTMnCTL.OSTMnMD0 = 1, an GTMTINT interrupt request is generated at the start of counting.

(2) When the current counter value matches the value in the OSTMnCMP register, an GTMTINT interrupt request is generated.

(3) When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter.
The counter retains its current value until it is restarted.

(4) Counting by the counter restarts from 0000_0000h when OSTMnTS.OSTMnTS = 1.

GTMTINT period

The GTMTINT generation period is different at the start of counting and depends on the old and new compare values if OSTMnCMP is rewritten during operation.

Table 5.5-6 GTMTINT Generation Timing

Old Value for Comparison	New Value for Comparison	Counter Value at Time of Rewriting	Period of GTMTINT Generation	Label in Timing Diagram
Counter starts			(A + 1) × counter clock period	(a)
A	A	No rewriting	(FFFF_FFFFh + 1) x counter clock period	(b)
B	C > B	B < counter value < C	(C – B) × counter clock period	(c)
C	D < C	Counter value > D,C	(FFFF_FFFFh – C + D + 1) x counter clock period	(d)

Forced restart

Forced restarting does not proceed during counting even if the OSTMnTS.OSTMnTS bit is set. The counter ignores the attempted setting and continues counting.

5.5.3.6.2 Operation when OSTMnCMP = 0000_0000h

The following figure shows the operation of GTM when OSTMnCMP = 0000_0000h, and counter-start interrupts are enabled (OSTMnCTL.OSTMnMD0 = 1).

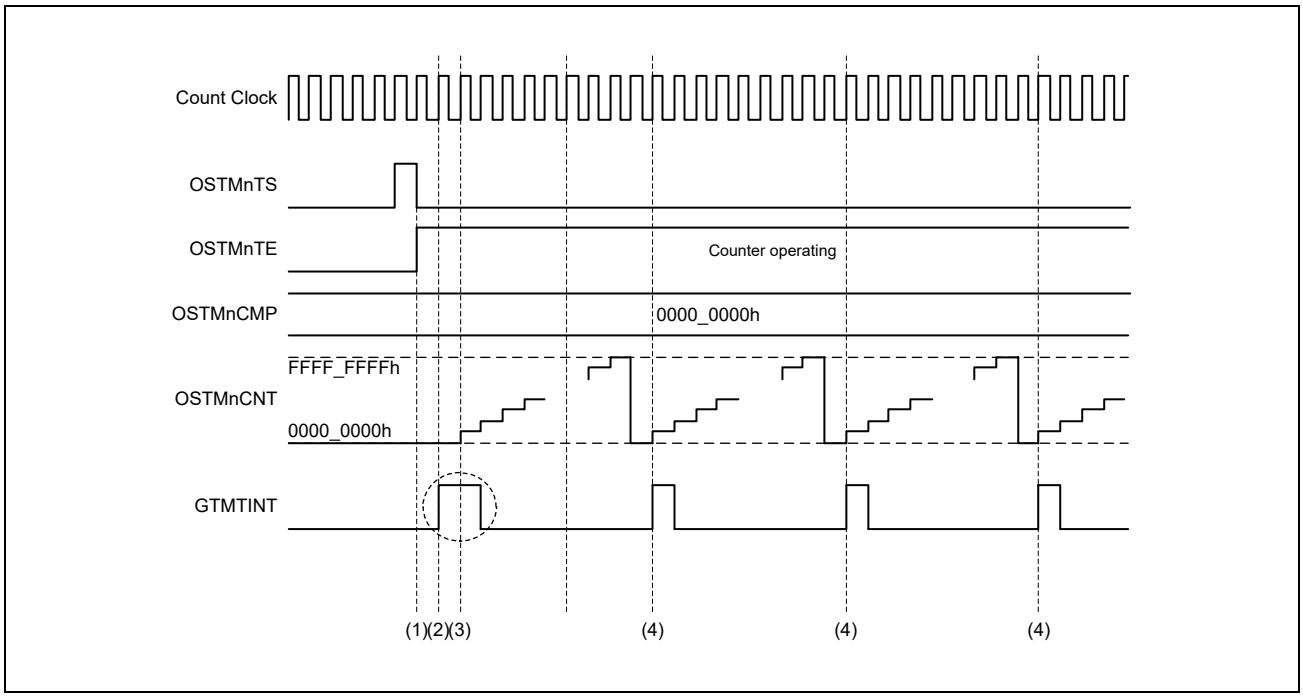


Figure 5.5-7 Timing Diagram when OSTMnCMP = 0000_0000h in Free-Run Compare Mode

The timing diagram above shows the following operations.

(1) Once the counter starts, it counts up from 0000_0000h to FFFF_FFFFh.

(2) An GTMTINT interrupt request is generated when counting starts.

(3) If the current counter value matches OSTMnCMP, an GTMTINT interrupt request is generated. If OSTMnCMP = 0000_0000h in the above case, GTMTINT is generated over two clock cycles.

(4) Every (FFFF_FFFFh + 1) clock cycles the GTMTINT interrupt request is asserted.

When interrupts on starting of the counter are disabled, no interrupt is generated when counting starts.

SECTION 5 TIMER

5.6 Compare Match Timer W (CMTW)

5.6.1 Overview

The LSI has eight channels (4 channels × 2 units) of the 32-bit compare match timer W (CMTW). The CMTW has a 32-bit counter and can generate interrupts each time a set period elapses.

Table 5.6-1 shows the specifications of the CMTW and **Figure 5.6-1** shows a block diagram of the CMTW.

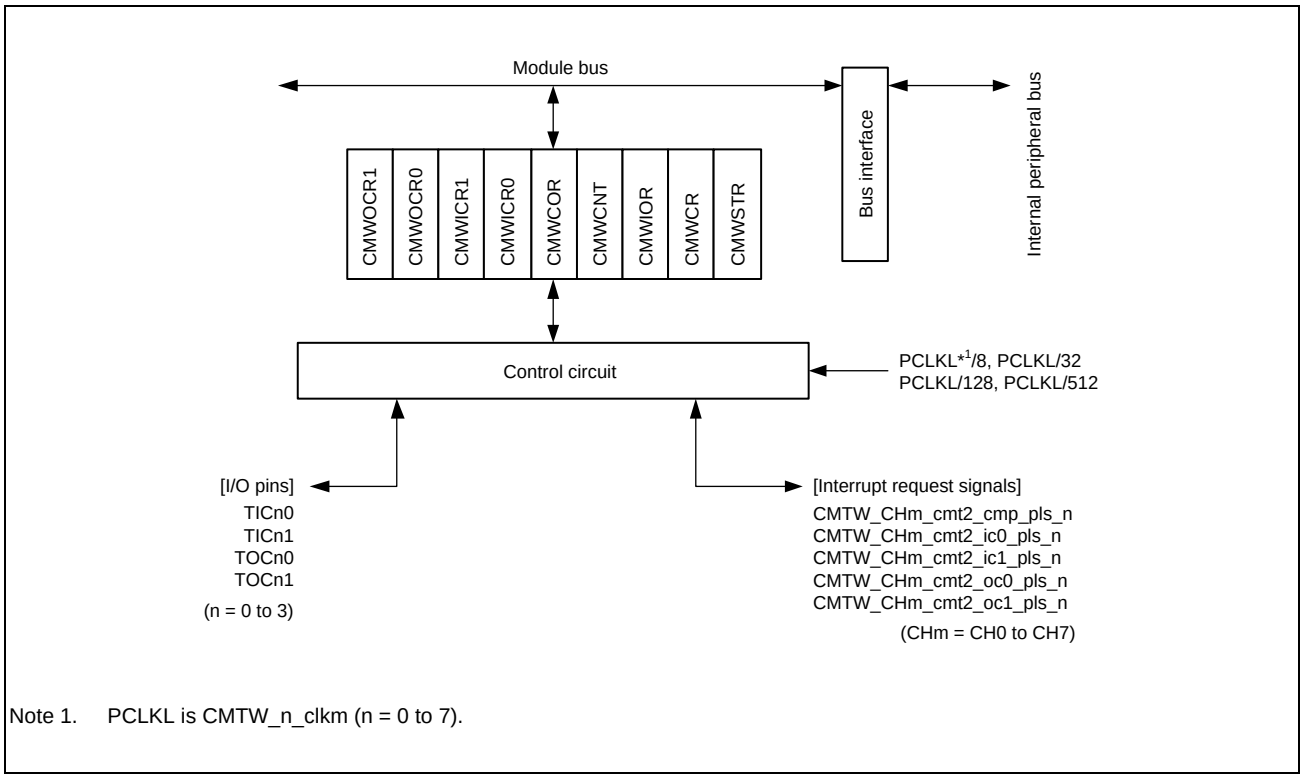
Table 5.6-1 CMTW Specifications

Item	Description
Number of channels	8 channels (4 channels x 2 units)
Timer counter	• 16-bit/32-bit selectable up-counter. • The counter returns to 0000_0000h after a compare match.
Prescaler	• Four dividing clocks are output. • Selectable from any of PCLKL*1/8, PCLKL/32, PCLKL/128, or PCLKL/512.
Input capture	Up to two input capture input signals available per channel
Output compare	Up to two output compare output signals available per channel
Compare match	One compare match available (no output compare output pin used)
Interrupt	• Compare match interrupt • Input capture 0 and 1 interrupts per channel • Output compare 0 and 1 interrupts per channel
Three event outputs available	• Compare match event • Output compare 0 and 1 event

Note 1. PCLKL is CMTW_n_clkm (n = 0 to 7).

Table 5.6-2 CMTW Clock Signals

Clock Signal Name	Description
CMTW_0_clkm	CMTW0 unit clock
CMTW_1_clkm	CMTW1 unit clock
CMTW_2_clkm	CMTW2 unit clock
CMTW_3_clkm	CMTW3 unit clock
CMTW_4_clkm	CMTW4 unit clock
CMTW_5_clkm	CMTW5 unit clock
CMTW_6_clkm	CMTW6 unit clock
CMTW_7_clkm	CMTW7 unit clock



Note 1. PCLKL is CMTW_n_clkm (n = 0 to 7).

Figure 5.6-1 Block Diagram of CMTW

Table 5.6-3 shows the CMTW pin configuration.

Table 5.6-3 Input/Output pins of CMTW

Unit	Pin Name	Input/Output	Description
CMTWn (n = 0 to 3)	TICn0	Input	CMTWn input capture input 0
	TICn1	Input	CMTWn input capture input 1
	TOCn0	Output	CMTWn output compare output 0
	TOCn1	Output	CMTWn output compare output 1

Note: CMTW4 to CMTW7 do not have any I/O pins.

5.6.2 Registers

The register addresses of CMTW are given as offsets from the individual base addresses <CMTWn_base>. The register base addresses of each CMTW are listed in the following table.

Table 5.6-4 Register Base Addresses

Base Address Name	Base Address
<CMTW0_base>	0_11C0_1800h (51C0_1800h*1, 41C0_1800h*2)
<CMTW1_base>	0_11C0_1C00h (51C0_1C00h*1, 41C0_1C00h*2)
<CMTW2_base>	0_11C0_2000h (51C0_2000h*1, 41C0_2000h*2)
<CMTW3_base>	0_11C0_2400h (51C0_2400h*1, 41C0_2400h*2)
<CMTW4_base>	0_1300_0C00h (5300_0C00h*1, 4300_0C00h*2)
<CMTW5_base>	0_1300_1000h (5300_1000h*1, 4300_1000h*2)
<CMTW6_base>	0_1300_1400h (5300_1400h*1, 4300_1400h*2)
<CMTW7_base>	0_1300_1800h (5300_1800h*1, 4300_1800h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

5.6.2.1 List of Registers

The CMTW registers and the memory addresses are listed in the following table. For the actual addresses, the offset values indicated in this table are added to the base addresses.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Timer Start Register	CMTWm_CMWSTR	0000h	00h	16, 32
Timer Control Register	CMTWm_CMWCR	0000h	04h	16, 32
Timer I/O Control Register	CMTWm_CMWIOR	0000h	08h	16, 32
Reserve	-	-	0Ch to 0Fh	-
Timer Counter	CMTWm_CMWCNT	0000_0000h	10h	32
Compare Match Constant Register	CMTWm_CMWCOR	FFFF_FFFFh	14h	32
Input Capture Registers 0	CMTWm_CMWICR0	0000_0000h	18h	32
Input Capture Registers 1	CMTWm_CMWICR1	0000_0000h	1Ch	32
Output Compare Registers 0	CMTWm_CMWOCR0	FFFF_FFFFh	20h	32
Output Compare Registers 1	CMTWn_CMWOCR1	FFFF_FFFFh	24h	32

5.6.2.2 Register Description

The prefix (CMTWm_) of the register names is omitted in this and subsequent sections.

5.6.2.2.1 Timer Start Register (CMTWm_CMWSTR)

The CMWSTR register is used to start or stop the CMWCNT counter.

Access Size : 16, 32 bits
Address : <CMTWm_base> + 00h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 1	-	All 0	R	Reserved These bits are read as 0b. The write value should be 0b.
0	STR	0h	RW	Counter Start 0b: The CMWCNT counter stops counting. The value immediately before the end of counting is retained and counting is stopped. 1b: The CMWCNT counter starts counting.

STR bit (Counter Start)

Specifies whether the timer counter operates or is stopped. The relevant prescaler operates or is stopped according to the setting of the STR bit.

5.6.2.2.2 Timer Control Register (CMTWm_CMWCR)

The CMWCR register selects the counter clearing source and the counter input clock, and enables or disables interrupts. The CMWCR register should be set while the timer counter (CMWCNT) operation is stopped.

Access Size : 16, 32 bits
Address : <CMTWm_base> + 04h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCLR[2:0]			-	-	-	CMS	-	OC1IE	OC0IE	IC1IE	IC0IE	CMWIE	-	CKS[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	R	R	R	RW	R	RW	RW	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	CCLR[2:0]	0h	RW	Counter Clear 000b: The CMWCNT counter is cleared by CMWCOR register compare match. 001b: The CMWCNT counter is not cleared. 010b: The CMWCNT counter is not cleared. 011b: The CMWCNT counter is not cleared. 100b: The CMWCNT counter is cleared by CMWICR0 register input capture. 101b: The CMWCNT counter is cleared by CMWICR1 register input capture. 110b: The CMWCNT counter is cleared by CMWOCR0 register compare match. 111b: The CMWCNT counter is cleared by CMWOCR1 register compare match.
12 to 10	-	All 0	R	Reserved These bits are read as 0b. The write value should be 0b.
9	CMS	0h	RW	Timer Counter Size 0b: 32 bits 1b: 16 bits
8	-	0h	R	Reserved This bit is read as 0b. The write value should be 0b.
7	OC1IE*2	0h	RW	Output Compare 1 Interrupt Enable 0b: Disables an interrupt request by the output compare 1 (CMTWm_OC1I). 1b: Enables an interrupt request by the output compare 1 (CMTWm_OC1I).
6	OC0IE*2	0h	RW	Output Compare 0 Interrupt Enable 0b: Disables an interrupt request by the output compare 0 (CMTWm_OC0I). 1b: Enables an interrupt request by the output compare 0 (CMTWm_OC0I).
5	IC1IE*2	0h	RW	Input Capture 1 Interrupt Enable 0b: Disables an interrupt request by the input capture 1 (CMTWm_IC1I). 1b: Enables an interrupt request by the input capture 1 (CMTWm_IC1I).
4	IC0IE*2	0h	RW	Input Capture 0 Interrupt Enable 0b: Disables an interrupt request by the input capture 0 (CMTWm_IC0I). 1b: Enables an interrupt request by the input capture 0 (CMTWm_IC0I).
3	CMWIE	0h	RW	Compare Match Interrupt Enable 0b: Disables a compare match interrupt request (CMTWm_CMWI). 1b: Enables a compare match interrupt request (CMTWm_CMWI).
2	-	0h	R	Reserved This bit is read as 0b. The write value should be 0b.
1, 0	CKS[1:0]	0h	RW	Clock Select*1 00b: PCLKL/8 01b: PCLKL/32 10b: PCLKL/128 11b: PCLKL/512

Note 1. PCLKL is CMTW_n_clkm (n = 0 to 7).

Note 2. CMTW4 to 7 are invalid because they do not use input capture and output compare.

CKS[1:0] bits (Clock Select)

Select the clock to be input to the CMWCNT counter from among four internal clocks obtained by dividing the peripheral clock (PCLKL). When the CMWSTR.STR bit is set to 1, the CMWCNT counter starts counting up based on the clock selected with the CMWCR.CKS[1:0] bits.

CMWIE bit (Compare Match Interrupt Enable)

Enables or disables compare match interrupt (CMTWm_CMWI) request generation when the CMWCNT counter and CMWCOR register values match.

IC0IE bit (Input Capture 0 Interrupt Enable)

Enables or disables input capture interrupt 0 (CMWm_IC0I) request generation when input capture is generated in the CMWICR0 register.

IC1IE bit (Input Capture 1 Interrupt Enable)

Enables or disables input capture interrupt 1 (CMWm_IC1I) request generation when input capture is generated in the CMWICR1 register.

OC0IE bit (Output Compare 0 Interrupt Enable)

Enables or disables compare match interrupt 0 (CMWm_OC0I) request generation when the CMWCNT counter and CMWOCR0 register values match.

OC1IE bit (Output Compare 1 Interrupt Enable)

Enables or disables compare match interrupt 1 (CMWm_OC1I) request generation when the CMWCNT counter and CMWOCR1 register values match.

CMS bit (Timer Counter Size)

Selects either 16 or 32 bits as the size of the timer counter (CMWCNT). The size selected with the CMS bit is valid in the compare match constant register (CMWCOR), input capture registers (CMWICR0 and CMWICR1), and output compare registers (CMWOCR0 and CMWOCR1).

CCLR[2:0] bits (Counter Clear)

Selects the CMWCNT counter clearing source.

5.6.2.2.3 Timer I/O Control Register (CMTWm_CMWIOR)

The CMWIOR register controls the CMWCOR, CMWICR0, CMWICR1, CMWOCR0, and CMWOCR1 registers. CMWIOR should be set while the timer counter (CMWCNT) operation is stopped.

External pins can only be used by CMTW0 to CMTW3.

Access Size : 16, 32 bits
Address : <CMTWm_base> + 08h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMWE	-	OC1E	OC0E	OC1[1:0]		OC0[1:0]		-	-	IC1E	IC0E	IC1[1:0]		IC0[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15	CMWE	0h	RW	Compare Match Enable 0b: Disables the compare match operation using the CMWCOR register. 1b: Enables the compare match operation using the CMWCOR register.
14	-	0h	R	Reserved These bits are read as 0b. The write value should be 0b.
13	OC1E	0h	RW	Compare Match Enable 1 0b: Disables the compare match operation using the CMWOCR1 register. 1b: Enables the compare match operation using the CMWOCR1 register.
12	OC0E	0h	RW	Compare Match Enable 0 0b: Disables the compare match operation using the CMWOCR0 register. 1b: Enables the compare match operation using the CMWOCR0 register.
11, 10	OC1[1:0]	0h	RW	Output Compare Control 1 00b: Retains the output value.*1 01b: Initially outputs 0 and toggles the output value upon compare match. 10b: Initially outputs 1 and toggles the output value upon compare match. 11b: Setting prohibited
9, 8	OC0[1:0]	0h	RW	Output Compare Control 0 00b: Retains the output value.*1 01b: Initially outputs 0 and toggles the output value upon compare match. 10b: Initially outputs 1 and toggles the output value upon compare match. 11b: Setting prohibited
7, 6	-	All 0	R	Reserved These bits are read as 0b. The write value should be 0b.
5	IC1E	0h	RW	Input Capture Enable 1 0b: Disables the input capture operation of the CMWICR1 register. 1b: Enables the input capture operation of the CMWICR1 register.
4	IC0E	0h	RW	Input Capture Enable 0 0b: Disables the input capture operation of the CMWICR0 register. 1b: Enables the input capture operation of the CMWICR0 register.
3, 2	IC1[1:0]	0h	RW	Input Capture Control 1 00b: Input capture at the rising edge on the TICn1 pin. 01b: Input capture at the falling edge on the TICn1 pin. 10b: Input capture at both edges on the TICn1 pin. 11b: Setting prohibited
1, 0	IC0[1:0]	0h	RW	Input Capture Control 0 00b: Input capture at the rising edge on the TICn0 pin. 01b: Input capture at the falling edge on the TICn0 pin. 10b: Input capture at both edges on the TICn0 pin. 11b: Setting prohibited

Note 1. After reset, 0 is output until the CMWIOR register is set.

IC0[1:0] bits (Input Capture Control 0)

Selects the input capture operation of the CMWICR0 register.

IC1[1:0] bits (Input Capture Control 1)

Selects the input capture operation of the CMWICR1 register.

IC0E bit (Input Capture Enable 0)

Enables or disables the input capture operation of the CMWICR0 register.

IC1E bit (Input Capture Enable 1)

Enables or disables the input capture operation of the CMWICR1 register.

OC0[1:0] bits (Output Compare Control 0)

Sets the output compare operation using the CMWOCR0 register.

OC1[1:0] bits (Output Compare Control 1)

Sets the output compare operation using the CMWOCR1 register.

OC0E bit (Compare Match Enable 0)

Enables or disables the compare match operation using the CMWOCR0 register.

OC1E bit (Compare Match Enable 1)

Enables or disables the compare match operation using the CMWOCR1 register.

CMWE bit (Compare Match Enable)

Enables or disables the compare match operation using the CMWCOR register.

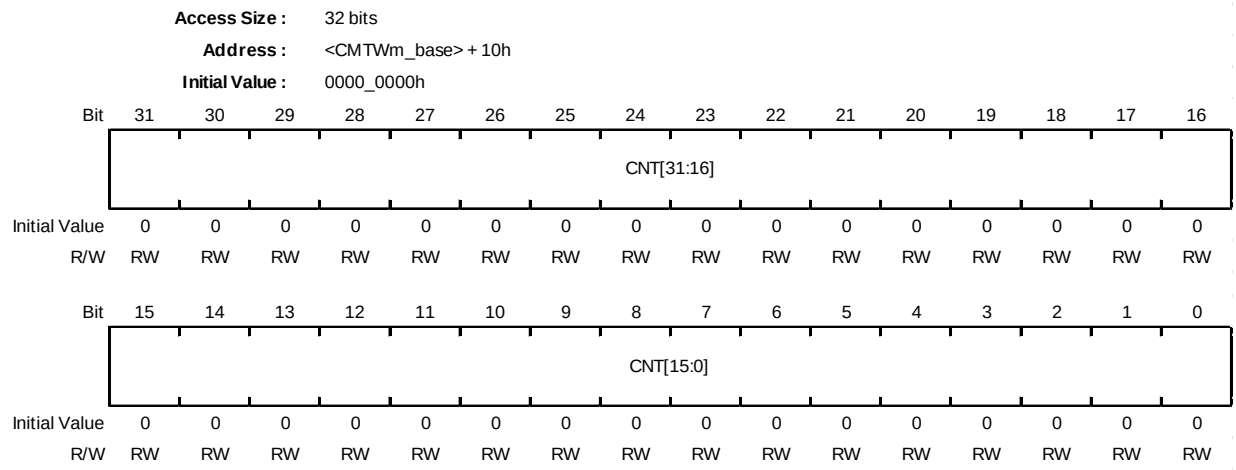
5.6.2.2.4 Timer Counter (CMTWm_CMWCNT)

The CMWCNT counter is used as a readable/writable up-counter.

Before starting counter operation, the timer control register (CMWCR) should be set. When the 16-bit counter size is selected with the CMS bit in the timer control register (CMWCR), bits 15 to 0 in the CMWCNT counter are valid. Since access to this register is in 32-bit units, when writing, write 32-bit values to it with 0000 in the higher-order bits. If a value other than 0000h is set in the higher-order bits, a value greater than 0000_FFFFh may be read when this register is read.

When the STR bit is set to 1, the CMWCNT counter starts counting. When the STR bit is set to 0, the CMWCNT counter retains the value immediately before the end of counting and stops counting.

Access Size : 32 bits
Address : <CMTWm_base> + 10h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CNT[31:0]	0h	RW	Timer Counter

5.6.2.2.5 Compare Match Constant Register (CMTWm_CMWCOR)

The CMWCOR register is a readable/writable register that specifies the time up to a compare match between the timer counter (CMWCNT) value and CMWCOR value.

When the 16-bit counter size is selected with the CMS bit in the timer control register (CMWCR), bits 15 to 0 in this register are valid. Since access to this register is in 32-bit units, when writing, write a 32-bit value to it with 0000h in the higher-order bits.

The cycle for compare matches is as follows:

Compare-match cycle = (setting of the CMWCOR register + 1) × counter-clock cycle*1

Note 1. This is a clock cycle set by the CMWCR.CKS[1:0] bits.

Access Size : 32 bits
Address : <CMTWm_base> + 14h
Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	COR[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COR[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

5.6.2.2.6 Input Capture Registers n (CMTWm_CMWICRn) (n = 0, 1)

The CMWICR0 and CMWICR1 registers are read-only registers in which the CMWCNT counter value is stored when an input capture is generated.
When the 16-bit counter size is selected with the CMS bit in the timer control register (CMWCR), bits 15 to 0 in these registers are valid. Writing to these registers is invalid.

Access Size : 32 bits

Address : <CMTWm_base> + 18h (n=0)
<CMTWm_base> + 1Ch (n=1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ICR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ICR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

5.6.2.2.7 Output Compare Registers n (CMTWm_CMWOCRn) (n = 0, 1)

The CMWOCR0 and CMWOCR1 registers are readable/writable registers that set the value to be compared when an output compare is generated.
When the 16-bit counter size is selected with the CMS bit in the timer control register (CMWCR), bits 15 to 0 in these registers are valid. Since access to these registers is in 32-bit units, when writing, write a 32-bit value to them with 0000_0000h in the higher-order bits.

Access Size : 32 bits

Address : <CMTWm_base> + 20h (n=0)
<CMTWm_base> + 24h (n=1)

Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OCR[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OCR[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

5.6.3 Operation

When the CMWCR register is set and then the STR bit in CMWCR is set to 1, the CMTW starts counter operation.

Setting the CMWSTR.STR bit to 0 enables the CMWCNT counter to retain the value immediately before the end of counting and stop counting. Setting CMWIOR register enables using the compare match function, input capture input function, and output compare output function.

5.6.3.1 Period Counting Operation

When the internal clock is selected by using the CMWCNT.CKS[1:0] bits and the CMWSTR.STR bit is set to 1, the CMWCNT counter starts counting up cycles of the selected clock. When the CCLR[2:0] bits in CMWCR are set so that CMWCNT should be cleared by a specific counter clearing source and the counter clearing source is generated, the CMWCNT counter is cleared to 0000_0000h and continues incrementing. When the CCLR[2:0] bits are set so that CMWCNT should not be cleared by any specific counter clearing source, the CMWCNT counter is cleared to 0000_0000h only when an overflow is generated (FFFF_FFFFh → 0000_0000h (when the counter size is 32 bits) or 0000_FFFFh → 0000_0000h (when the counter size is 16 bits)) and continues incrementing.

5.6.3.2 Compare Match Function

When the values of the CMWCNT counter and CMWCOR register match, a compare match interrupt (CMTWm_CMWI) is generated. The CMWCNT counter operates as follows according to the setting of CMWCR.CCLR[1:0].

1. When CMWCR.CCLR[2:0] = 000b
   When the values of the CMWCNT counter and CMWCOR register match, the CMWCNT counter is cleared to 0000_0000h. The CMWCNT counter then restarts counting up from 0000_0000h.
2. When CMWCR.CCLR[2:0] = a value other than 000b
   Even when the values of the CMWCNT counter and CMWCOR register match, the CMWCNT counter is not cleared to 0000_0000h but continues counting up until the clearing condition set in CMWCR.CCLR[1:0] is satisfied or the value of the counter reaches FFFF_FFFFh (when the size of the counter is 32 bits) or 0000_FFFFh (when the size of the counter is 16 bits).
   The CMWCNT counter is then cleared to 0000_0000h and restarts counting up from 0000_0000h.

Figure 5.6-2 shows an example of procedure for setting compare match operation.

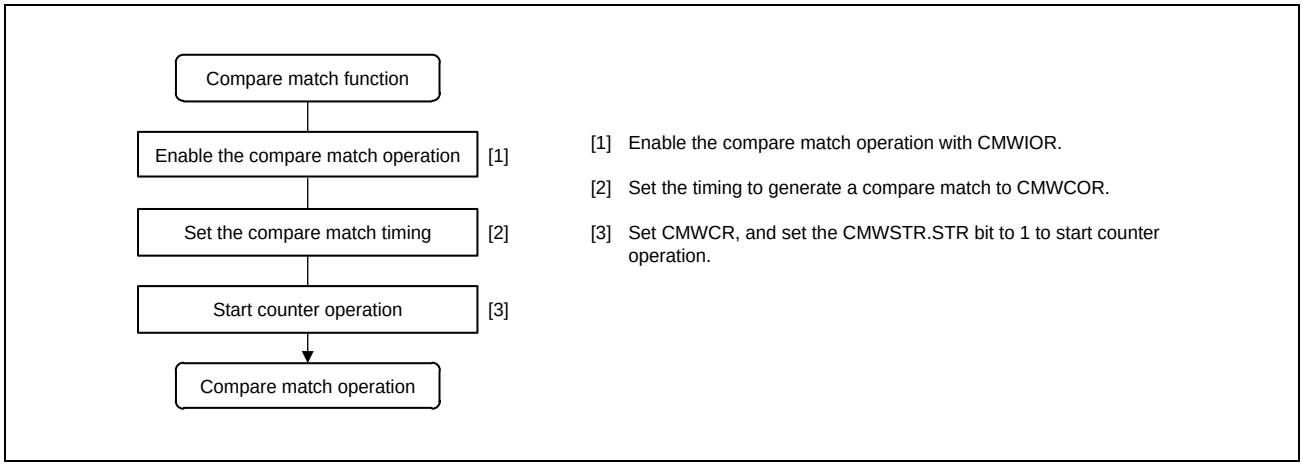


Figure 5.6-2 Procedure for Setting Compare Match Operation

Figure 5.6-3 shows an example when compare match with CMWCOR is set as a counter clearing source.

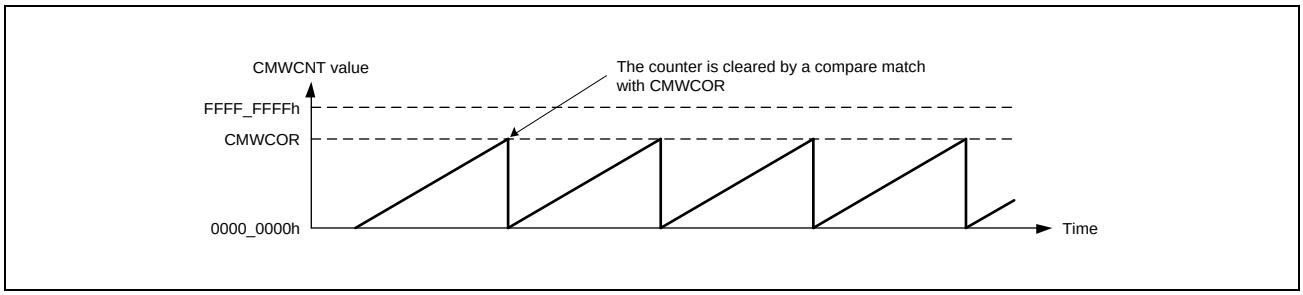


Figure 5.6-3 Example of Compare Match Operation

Figure 5.6-4 shows an example when CMWCOR is set to FFFF_FFFFh and an overflow is detected.

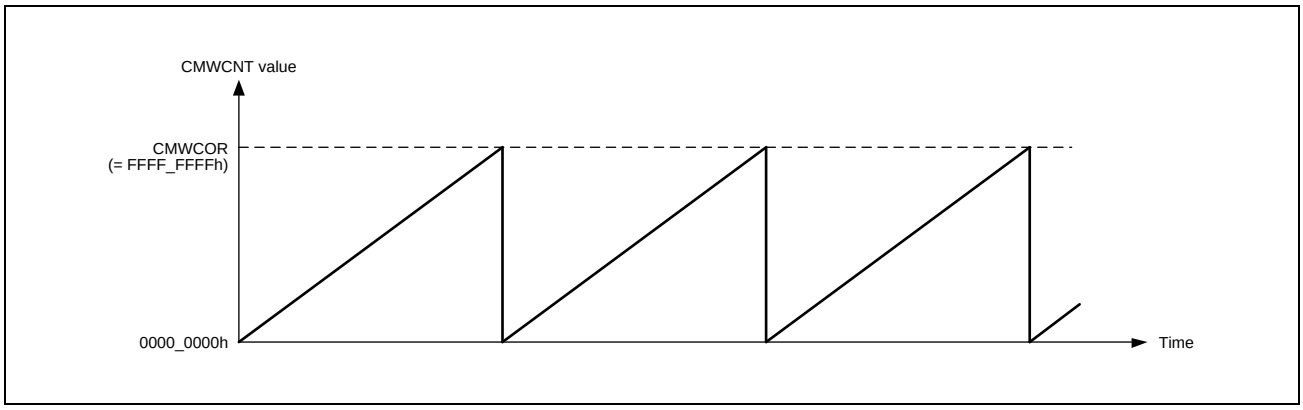


Figure 5.6-4 Example of Compare Match Operation (Overflow Detected)

5.6.3.3 Output Compare Function

Using the output compare function, toggle output from the relevant output pins can be provided. When the CMWCNT counter value matches the value of the CMWOCR0 or CMWOCR1 register, the output compare interrupt (CMTWm_OC0I or CMTWm_OC1I) is generated. The CMWCNT counter operates as follows according to the setting of CMWCR.CCLR[2:0].

1. When CMWCR.CCLR[2:0] = 110b
   When the values of the CMWCNT counter and CMWOCR0 register match, the CMWCNT counter is cleared to 0000_0000h. The CMWCNT counter then restarts counting up from 0000_0000h.
2. When CMWCR.CCLR[2:0] = 111b
   When the values of the CMWCNT counter and CMWOCR1 register match, the CMWCNT counter is cleared to 0000_0000h. The CMWCNT counter then restarts counting up from 0000_0000h.
3. When CMWCR.CCLR[2:0] = a value other than 110b and 111b
   Even when the values of the CMWCNT counter and CMWOCR0 or CMWOCR1 register match, the CMWCNT counter is not cleared to 0000_0000h but continues counting up until the clearing condition set in CMWCR.CCLR[2:0] is satisfied or the value of the counter reaches FFFF_FFFFh (when the size of the counter is 32 bits) or 0000_FFFFh (when the size of the counter is 16 bits). The CMWCNT counter is then cleared to 0000_0000h and restarts counting up from 0000_0000h.

Figure 5.6-5 shows an example of procedure for setting output compare operation.

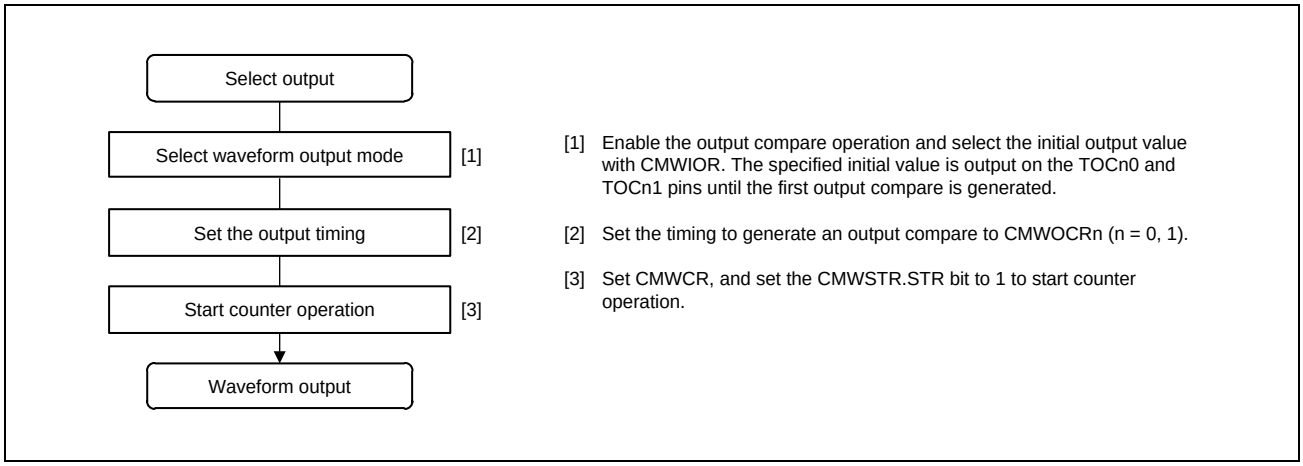


Figure 5.6-5 Procedure for Setting Output Compare Operation

Figure 5.6-6 shows an example when the counter is cleared upon compare match with the CMWOCR1 register and toggle outputs are provided from the TOCn0 and TOCn1 pins.

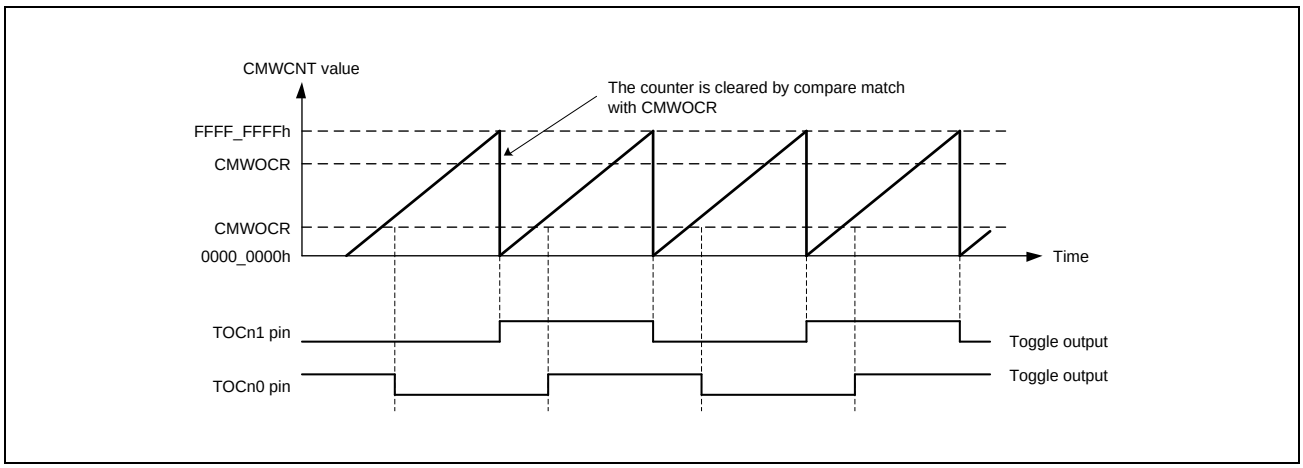


Figure 5.6-6 Example of Output Compare Operation

5.6.3.4 Input Capture Function

Through detecting the edge on the TICn0 and TICn1 pin input, the CMWCNT counter value can be transferred to the CMWICR0 and CMWICR1 registers, respectively. The edges to be detected can be selected from among the rising edge alone, falling edge alone, and both the rising and falling edges. When the CMWCNT counter value is transferred to the CMWICR0 or CMWICR1 register using the input capture function, an input capture interrupt (CMTWm_IC0I or CMTWm_IC1I) is generated. The CMWCNT counter operates as follows according to the setting of CMWCR.CCLR[2:0].

1. When CMWCR.CCLR[2:0] = 100b
   When the CMWCNT counter value is transferred to CMWICR using the input capture operation, the CMWCNT counter is cleared to 0000_0000h.
   The CMWCNT counter then restarts counting up from 0000_0000h.
2. When CMWCR.CCLR[2:0] = 101b
   When the CMWCNT counter value is transferred to CMWICR using the input capture operation, the CMWCNT counter is cleared to 0000_0000h.
   The CMWCNT counter then restarts counting up from 0000_0000h.
3. When CMWCR.CCLR[2:0] = a value other than 100b and 101b
   Even when the CMWCNT counter value is transferred to CMWICR0 or CMWICR1 using the input capture operation, the CMWCNT counter is not cleared to 0000_0000h but continues counting up until the clearing condition set in CMWCR.CCLR[2:0] is satisfied or the value of the counter reaches FFFF_FFFFh (when the size of the counter is 32 bits) or 0000_FFFFh (when the size of the counter is 16 bits). The CMWCNT counter is then cleared to 0000_0000h and restarts counting up from 0000_0000h.

Figure 5.6-7 shows an example of procedure for setting input capture operation.

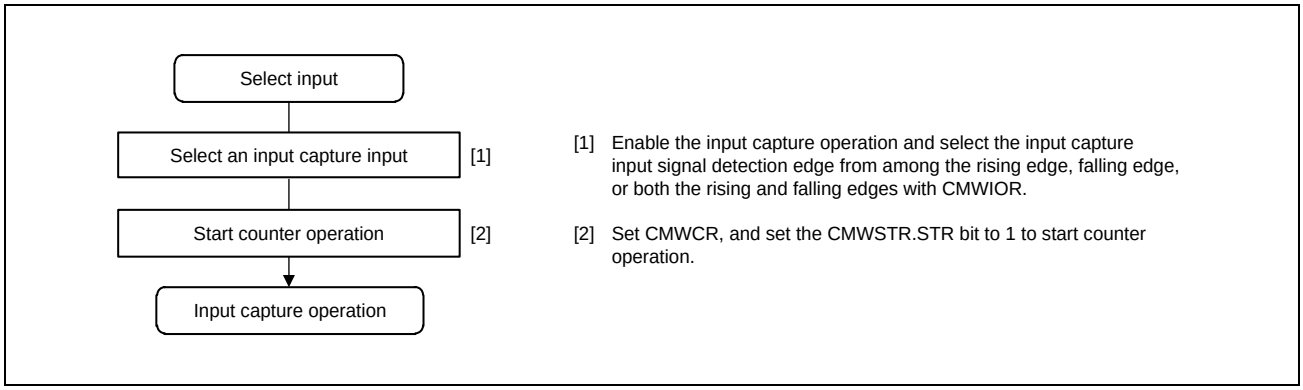


Figure 5.6-7 Procedure for Setting Input Capture Operation

Figure 5.6-8 shows an example in which both the rising and falling edges are selected for the TICn0 pin input capture input edge and the falling edge for the TICn1 pin, and the CMWCNT counter is cleared by a CMWICR1 register input capture.

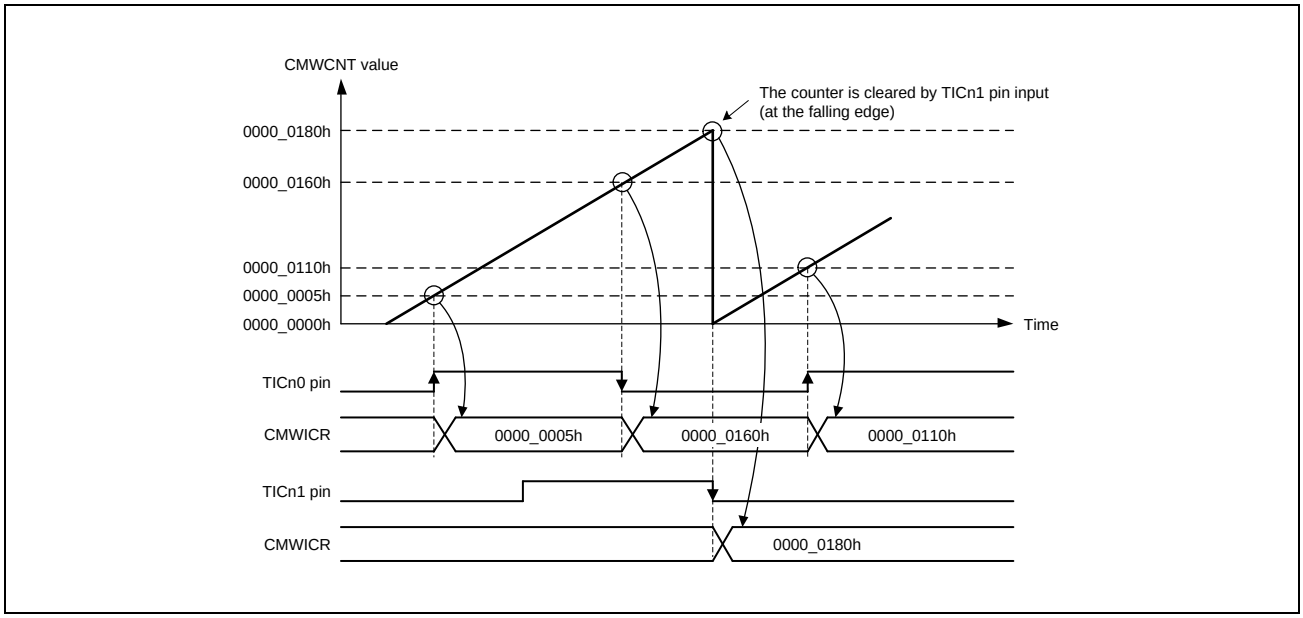


Figure 5.6-8 Example of Input Capture Operation

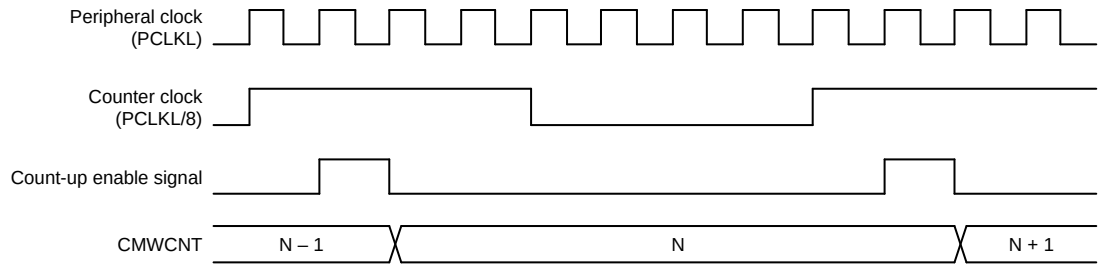
5.6.3.5 Counter Size

With the CMTW, either 16 or 32 bits can be selected as the counter size by using the CMWCR.CMS bit. When the counter is used as a 32-bit counter, set the CMWCOR, CMWOCR0, or CMWOCR1 register to the desired values in 32-bit units. In reading, all 32 bits of CMWICR0 and CMWICR1 are valid. When the counter is used as a 16-bit counter, a 32-bit value should be set to the CMWCOR register with 0000h in the higher-order bits. Similarly, a 32-bit value should be set to the CMWOCR0 and CMWOCR1 registers with 0000h in the higher-order bits.

A 32-bit value with 0000h in the higher-order bits is read from the CMWICR0 and CMWICR1 registers.

5.6.3.6 Count Timing Based on CMWCNT

One of four clocks (PCLKL/8, PCLKL/32, PCLKL/128, and PCLKL/512) obtained by dividing the peripheral clock (PCLKL) can be selected with the CMWCR.CKS[1:0] bits. **Figure 5.6-9** shows the timing.



Note: PCLKL is CMTW_n_clkm (n = 0 to 7).

Figure 5.6-9 Count Timing (PCLKL/8)

5.6.3.7 Output Compare Output Timing

A compare match signal is generated in the last state in which the CMWOCRn (n = 0, 1) register and CMWCNT counter values match (the CMWCNT counter value is updated immediately after the state). That is, the compare match signal is not generated if the CMWCNT counter clock is not input after a match between the CMWOCR register and CMWCNT counter values. When a compare match signal is generated, the output compare output pin (TOC) changes in accord with the setting of the OC0[1:0] or OC1[1:0] bits in the CMWIOR register. **Figure 5.6-10** shows the output compare output timing.

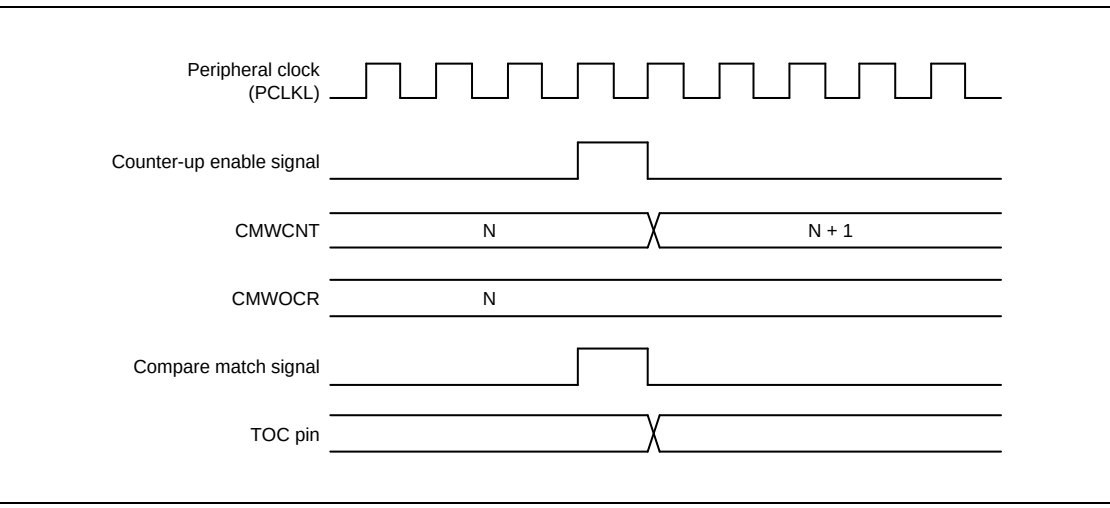


Figure 5.6-10 Output Compare Output Timing

5.6.3.8 Input Capture Timing

Figure 5.6-11 shows the input capture timing.

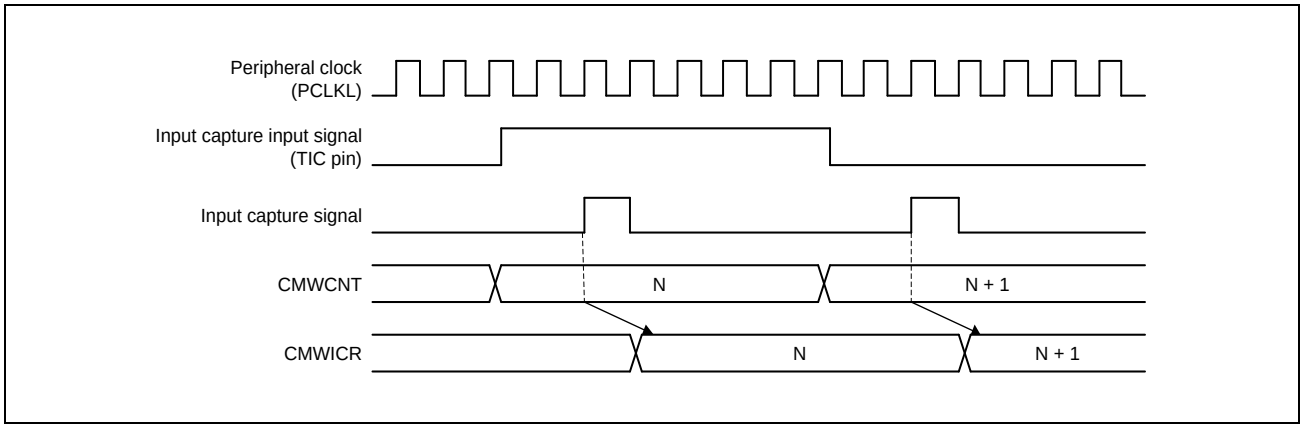


Figure 5.6-11 Input Capture Timing

5.6.4 Interrupts

5.6.4.1 CMTW Interrupt Sources and DMAC Transfer Requests

The CMTW has five interrupt sources: two input capture interrupt requests (CMTWm_IC0I and CMTWm_IC1I), two output compare interrupt requests (CMTWm_OC0I and CMTWm_OC1I), and a compare match interrupt request (CMTWm_CMWI).

Table 5.6-5 shows the interrupt sources and priority. The interrupt sources can be enabled or disabled using the IC0IE, IC1IE, OC0IE, OC1IE, and CMWIE bits in CMWCR and are separately issued to the interrupt controller.

Table 5.6-5 CMTW Interrupt Sources

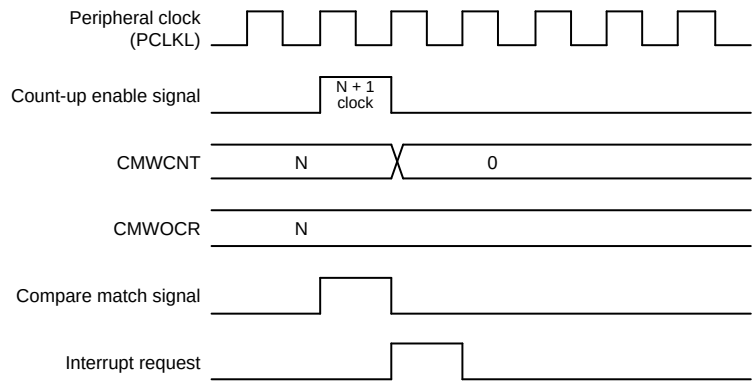
Interrupt source	Interrupt signal name	Interrupt	Interrupt enable bit	DMAC activation	Priority
CMTWm_CMWI	CMTW_CHm_cmt2_cmp_pls_n (“CHm” = CH0 to CH7)	Interrupt caused by compare match	CMWIE	Possible	High
CMTWm_IC0I*1	CMTW_CHm_cmt2_ic0_pls_n (“CHm” = CH0 to CH7)	Interrupt caused by input capture 0	IC0IE	Possible	
CMTWm_IC1I*1	CMTW_CHm_cmt2_ic1_pls_n (“CHm” = CH0 to CH7)	Interrupt caused by input capture 1	IC1IE	Possible	
CMTWm_OC0I*1	CMTW_CHm_cmt2_oc0_pls_n (“CHm” = CH0 to CH7)	Interrupt caused by output compare 0	OC0IE	Possible	
CMTWm_OC1I*1	CMTW_CHm_cmt2_oc1_pls_n (“CHm” = CH0 to CH7)	Interrupt caused by output compare 1	OC1IE	Possible	Low

Note: m = 0 to 7

Note 1. CMTW4 to 7 cannot be used because they do not have external pins.

5.6.4.2 Timing of Compare Match Interrupt Generation

When the values of the CMWCNT counter and CMWCOR register match, a compare match interrupt (CMTWm_CMWI) is generated. The compare match signal is generated at the end of the cycle where the values matched (i.e. when the CMWCNT counter is updated from the matching counter value). The compare match signal, therefore, is not generated until a further cycle of the input clock (PCLKL/8, PCLKL/32, PCLKL/128, or PCLKL/512) for the CMWCNT counter arrives after the values of the CMWCNT counter and CMWCOR register have matched.
Figure 5.6-12 shows the timing of compare match interrupt generation.



Note: PCLKL is CMTW_n_clkm (n = 0 to 7).

Figure 5.6-12 Timing of Compare Match Interrupt Generation

5.6.4.3 Timing of Output Compare Interrupt Generation

Figure 5.6-13 shows the timing of output compare interrupt generation.

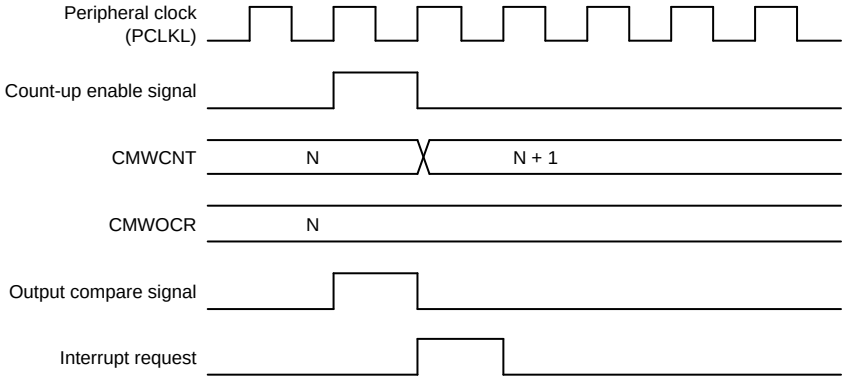


Figure 5.6-13 Timing of Output Compare Interrupt Generation

5.6.4.4 Timing of Input Capture Interrupt Generation

Figure 5.6-14 shows the timing of input capture interrupt generation.

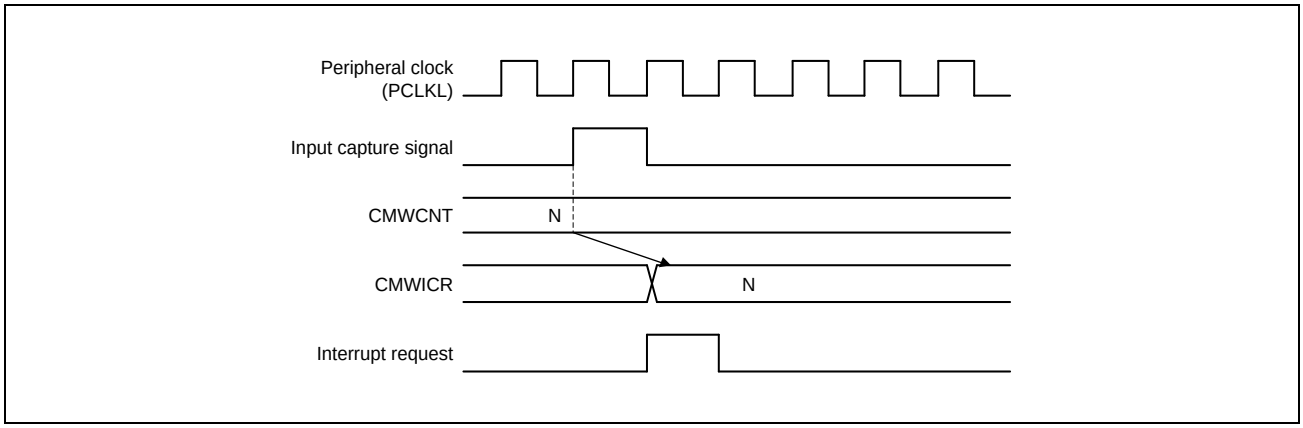


Figure 5.6-14 Timing of Input Capture Interrupt Generation

5.6.5 Event Link Operation

5.6.5.1 Issuing Event Signals to the ELC

The CMTW can issue the following event signals to the event link controller (ELC).

5.6.5.1.1 Compare match event

Upon generation of a compare match signal, the CMTW simultaneously issues an interrupt request and a compare match event signal to the ELC. The event signal is issued regardless of the setting of the corresponding interrupt request enable bit (CMWCR.CMWIE bit).

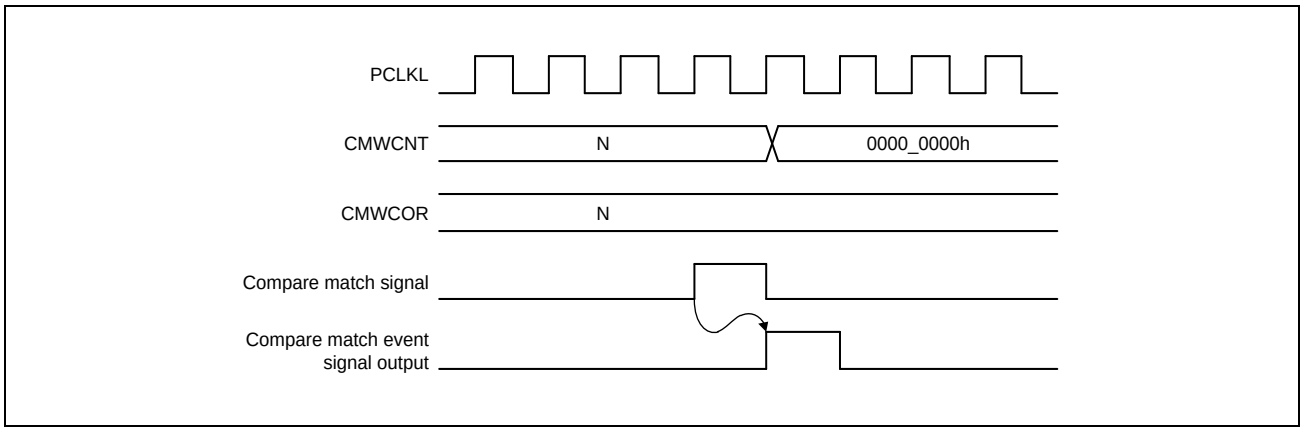


Figure 5.6-15 Timing of Issuing Compare Match Event

5.6.5.1.2 Output compare event

Upon generation of an output compare signal, the CMTW simultaneously issues an interrupt request and an output compare event signal to the ELC. The event signal is issued regardless of the setting of the corresponding interrupt request enable bit (CMWCR. OC0IE or CMWCR. OC1IE bit).

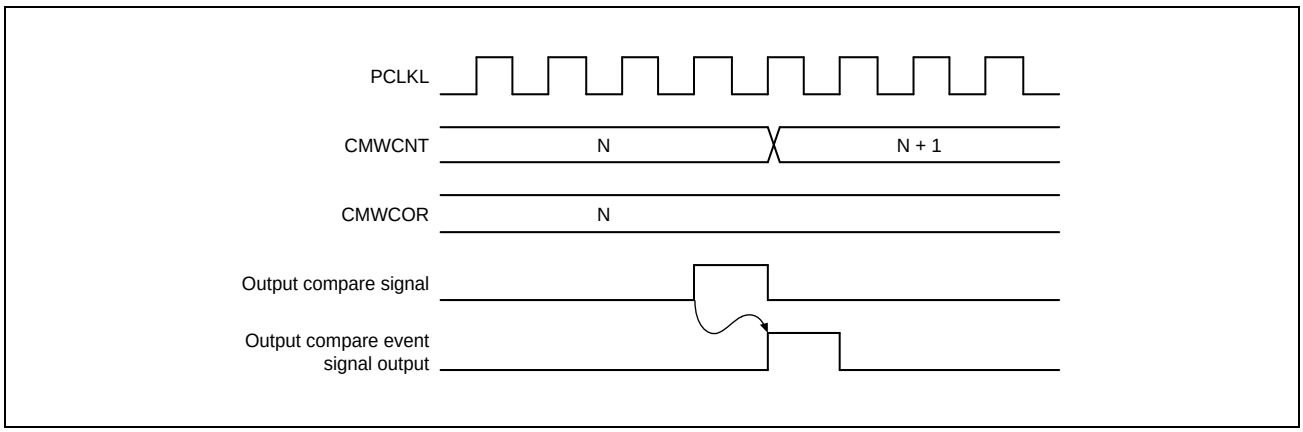


Figure 5.6-16 Timing of Issuing Output Compare Event

5.6.6 Usage Notes

5.6.6.1 Contention between CMWCNT Counter Writing and Compare Match

If the compare match signal is generated during CPU writing to the CMWCNT counter, the compare match interrupt request is output but the counter is not cleared since the CPU writing to the counter is given priority.

Figure 5.6-17 shows the timing of contention between CMWCNT counter writing and compare match.

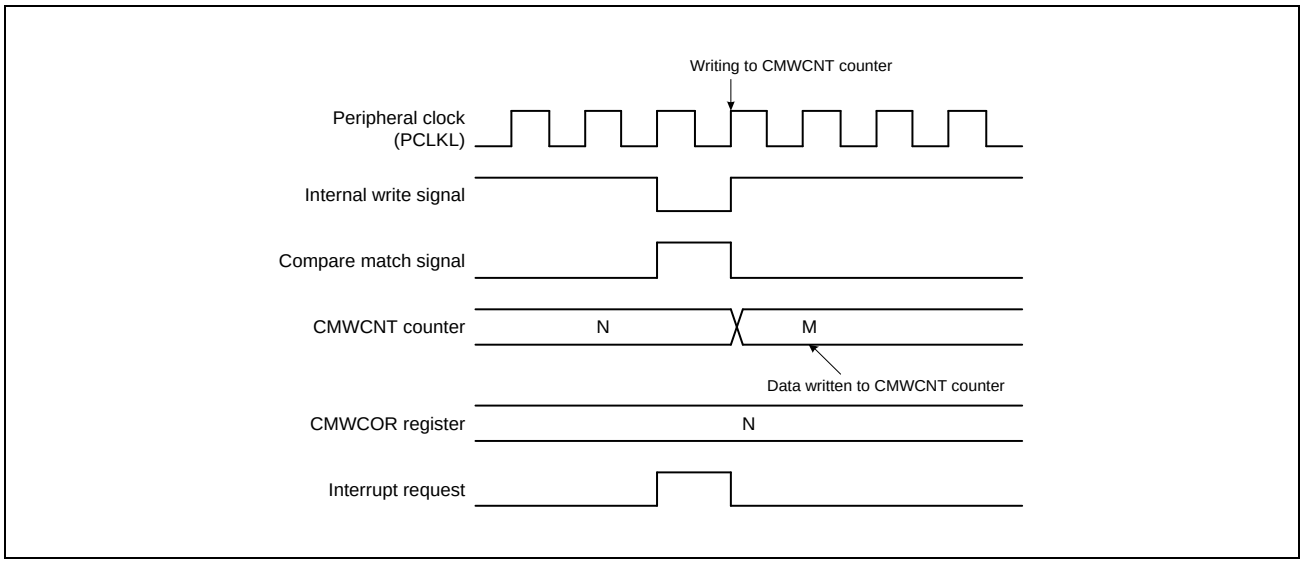


Figure 5.6-17 Contention between CMWCNT Counter Writing and Compare Match

5.6.6.2 Contention between CMWCNT Counter Writing and Incrementing or Clearing

In case of contention between incrementation or clearing of the CMWCNT counter and CPU writing to the counter, the counter is not actually incremented or cleared since the CPU writing to the CMWCNT counter is given priority.

Figure 5.6-18 shows the timing in the case of contention between writing to the CMWCNT counter and incrementation or clearing.

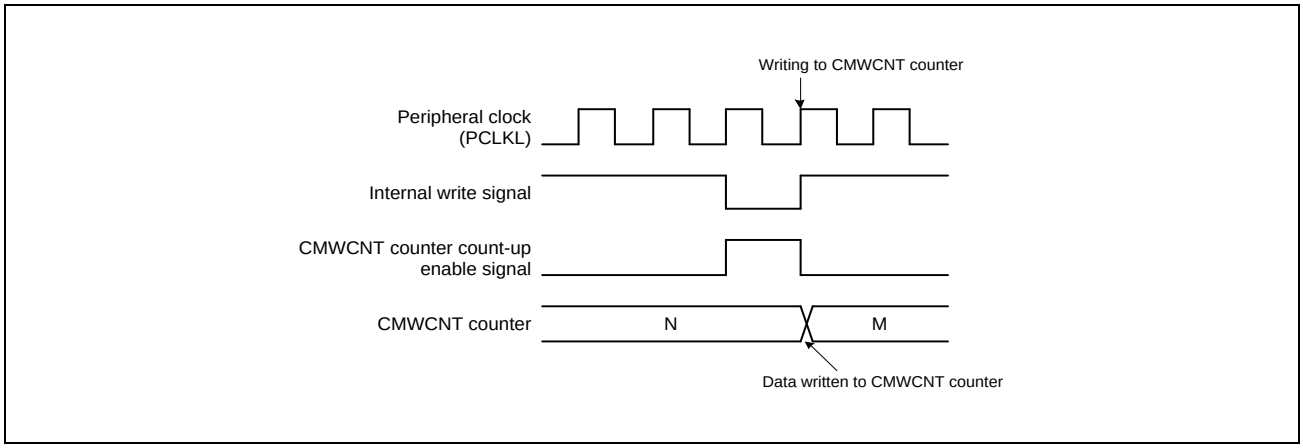


Figure 5.6-18 Contention between CMWCNT Counter Writing and Incrementing

5.6.6.3 Contention between CMWCOR Register Writing and Compare Match

If the compare match is generated during CPU writing to the CMWCOR register, the CPU writing to the CMWCOR register proceeds and also the compare match signal is output.

Figure 5.6-19 shows the timing of contention between CMWCOR register writing and compare match.

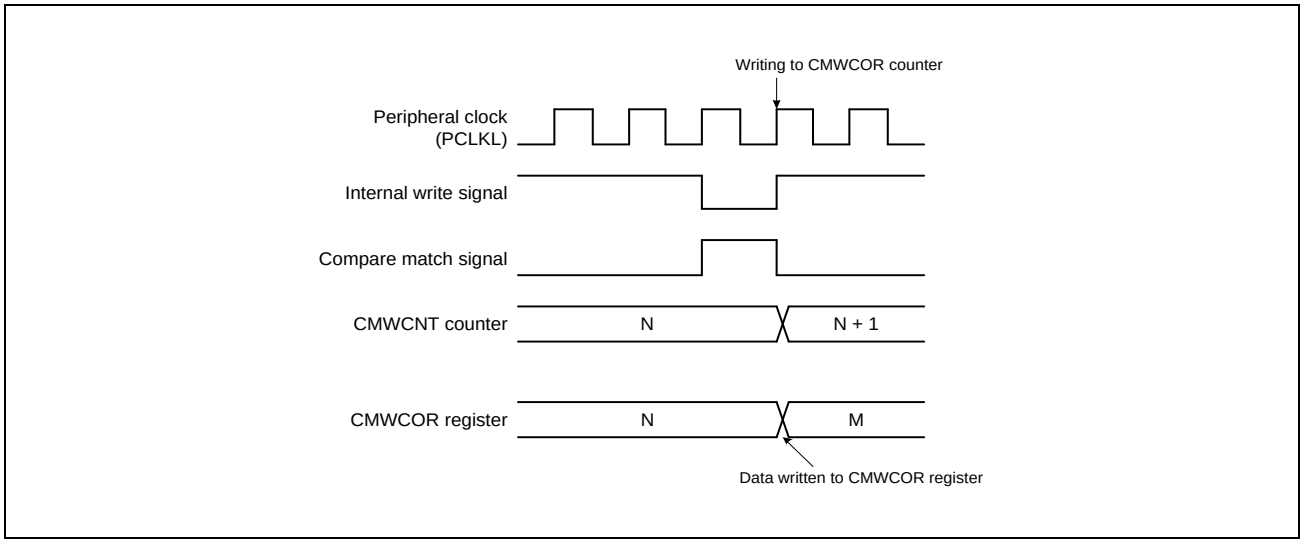


Figure 5.6-19 Contention between CMWCOR Register Writing and Compare Match

5.6.6.4 Contention between CMWOCRn Register Writing and Compare Match

If the compare match is generated during CPU writing to the CMWOCRn register, the CPU writing to the CMWOCRn register proceeds and also the compare match signal is output (n = 0, 1).

Figure 5.6-20 shows the timing of contention between CMWOCRn register writing and compare match.

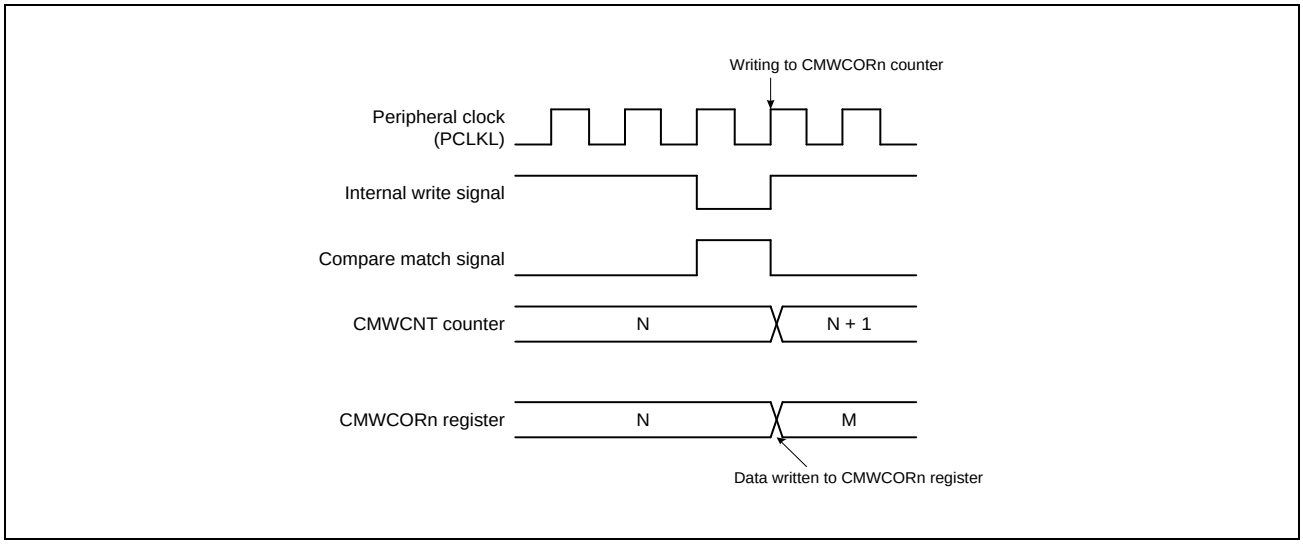


Figure 5.6-20 Contention between CMWOCRn Register Writing and Compare Match

5.6.6.5 Contention between CMWCNT Counter Reading and Incrementing or Clearing

If the CMWCNT counter incrementing or clearing process occurs at the same time that the data of the CMWCNT counter is read, the value having been in the CMWCNT counter before incremented or cleared is read.

Figure 5.6-21 shows the timing of contention between the CMWCNT counter reading and incrementing.

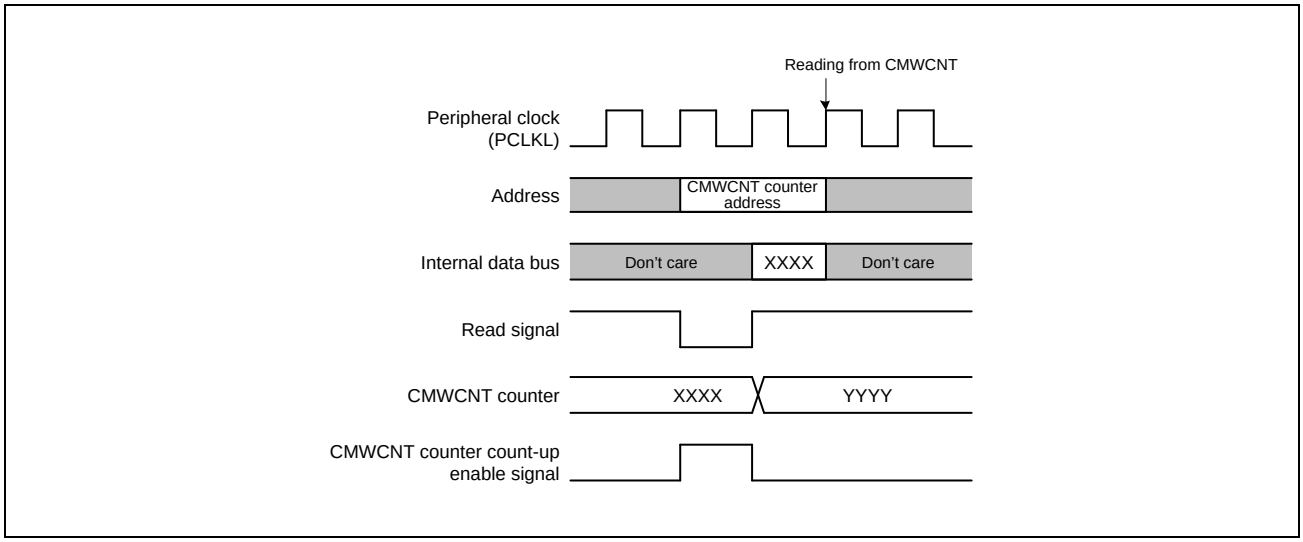


Figure 5.6-21 Contention between CMWCNT Counter Reading and Incrementing (When the Data Reading and Incrementing Process Occur Simultaneously)

5.6.6.6 Contention Between CMWICRn Register Reading and Input Capture

If the input capture signal is generated at the same time that the data of CMWICRn register is read, the value having been in CMWICRn register before updated by input capture transfer is read (n = 0, 1).

Figure 5.6-22 shows the timing of contention between the CMWICRn register reading and input capture.

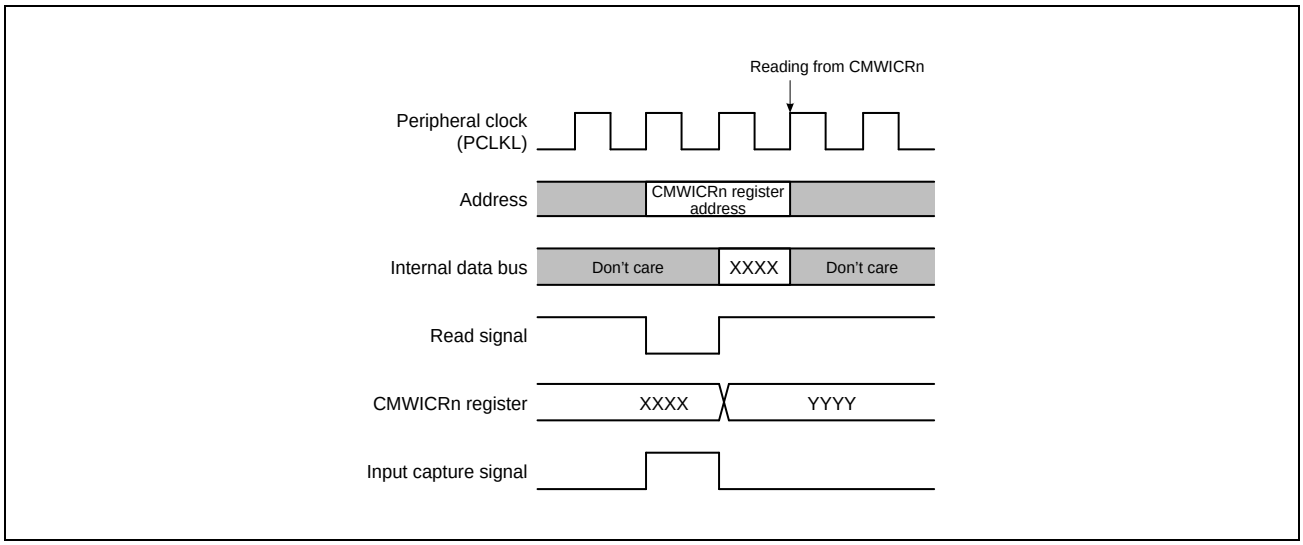


Figure 5.6-22 Contention between CMWICRn Register Reading and Input Capture (When the Input Capture Signal and Read Signal are Generated Simultaneously)

Table 5.6-6 Summary of Contention between Operations Due to the Access to Registers and Changes to the Counter's State

Register Access	CMWCNT State	CMWICR0/1 State	Operation to be Performed
Writing to CMWCNT	Compare match	—	Output of compare match interrupt request/writing to CMWCNT
	Counting up	—	Writing to CMWCNT
Writing to CMWCOR	Compare match	—	Compare match
Writing to CMWOCR0	Output compare 0	—	Output compare 0
Writing to CMWOCR1	Output compare 1	—	Output compare 1
Reading from CMWCNT	Counting up	—	Counting up and reading of the previous value
Reading from CMWICR0	—	Input capture 0	Input capture 0 and reading of the value before transfer
Reading from CMWICR1	—	Input capture 1	Input capture 1 and reading of the value before transfer

SECTION 5 TIMER

5.7 General-Purpose Timer (GPT)

5.7.1 Overview

The GPT is a 32-bit timer with 16 channels. PWM waveforms can be generated by controlling the up-counter, down-counter, or up- and down-counter.

Table 5.7-1 lists the GPT specifications, **Table 5.7-2** shows the GPT functions, and **Figure 5.7-1** shows a block diagram.

Table 5.7-1 GPT Specifications

Parameter	Description
Functions	• 32 bits × 16 channels • Up-counting or down-counting (saw waves) or up/down-counting (triangle waves) for each counter • Clock sources independently selectable for each channel • Four input/output pins per channel • Two output compare / input capture registers per channel • For the two output compare/input capture registers of each channel, four registers are provided as buffer registers and are capable of operating as comparison registers when buffering is not in use • In output compare operation, buffer switching can be at crests or troughs, enabling the generation of laterally asymmetric PWM waveforms • Registers for setting up frame cycles in each channel with capability for generating interrupts at overflow or underflow • Generation of dead times in PWM operation • Synchronous starting, stopping and clearing counters for arbitrary channels • Count start, count stop, count clear, up-count, down-count, or input capture operation in response to a maximum of 8 ELC events • Count start, count stop, count clear, up-count, down-count, or input capture operation in response to the status of two input pins • Count start, count stop, count clear, up-count, down-count, or input capture operation in response to a maximum of 4 external triggers • Output pin disable function by detected short-circuits between output pins • A/D converter start triggers can be generated • Compare match A to F event and overflow/underflow event can be output to the ELC • Enables the noise filter for input capture • Logical operation between the channel output • Bus clock: GPT_0_pclk_sfr, GPT_1_pclk_sfr Core clock: GPT_0_clks_gpt, GPT_1_clks_gpt

Table 5.7-2 GPT Functions (1/2)

Parameter		Description
Count clock (PCLKD)		GPT_0_clks_gpt, GPT_1_clks_gpt GPT_0_clks_gpt/2, GPT_1_clks_gpt/2 GPT_0_clks_gpt/4, GPT_1_clks_gpt/4 GPT_0_clks_gpt/8, GPT_1_clks_gpt/8 GPT_0_clks_gpt/16, GPT_1_clks_gpt/16 GPT_0_clks_gpt/32, GPT_1_clks_gpt/32 GPT_0_clks_gpt/64, GPT_1_clks_gpt/64 GPT_0_clks_gpt/256, GPT_1_clks_gpt/256 GPT_0_clks_gpt/1024, GPT_1_clks_gpt/1024 GTETRGA, GTETRGB, GTETRGC, GTETRGD
Output compare/input capture registers (GTCCR)		GTCCRA GTCCRB
Compare/buffer registers		GTCCRC GTCCRD GTCCRE GTCCRF
Cycle setting register		GTPR
Cycle setting buffer register		GTPBR GTPDBR
I/O pins		GTIOCnA GTIOCnB GTIOCnAN GTIOCnBN (n = 0 to 15)
External trigger input pin		GTETRGA, GTETRGB, GTETRGC, GTETRGD (for GPT0n, n = 0 to 7) GTETRGE, GTETRGF, GTETRGG, GTETRGH (for GPT1n, n = 0 to 7)
Counter clear sources		GTPR register compare match Input capture Input pin status ELC event input GTETRGx (x = A to H) pin input
Compare match output	Low-level output	Available
	High-level output	Available
	Toggle output	Available
Input capture function		Available
Automatic addition of dead time		Available
PWM mode		Available
Phase count function		Available
Buffer operation		Double buffer Simultaneous operation disable control for multiple channels
One-shot operation		Available
DMAC activation		All the interrupt sources
A/D converter start triggers		GTADTRA and GTADTRB compare match

Table 5.7-2 GPT Functions (2/2)

Parameter	Description*2
Interrupt sources	13 sources • GTCCRA compare match/input capture (GPT_Ux_gpt_gtcia_n_m) • GTCCRB compare match/input capture (GPT_Ux_gpt_gtcib_n_m) • GTCCRC compare match (GPT_Ux_gpt_gtcib_n_m) • GTCCRD compare match (GPT_Ux_gpt_gtcid_n_m) • GTCCRE compare match (GPT_Ux_gpt_gtcie_n_m) • GTCCRF compare match (GPT_Ux_gpt_gtcif_n_m) • GTCNT overflow (GTPR compare match) (GPT_Ux_gpt_gtciv_n_m) • GTCNT underflow (GPT_Ux_gpt_gtciu_n_m) • Dead time error (GPT_Ux_gpt_gtdei_n_m) • A and B both high interrupt (GPT_Ux_gpt_gtcih_n_m) • A and B both low interrupt (GPT_Ux_gpt_gtcil_n_m) • GTADTRA compare match (GPT_Ux_gpt_gtciada_n_m) • GTADTRB compare match (GPT_Ux_gpt_gtciadb_n_m)
Interrupt skipping	Skipping of GTCNT counter overflow (GTPR register compare match) (GPT_Ux_gpt_gtciv_n_m GTCIVn)/GTCNT counter underflow (GPT_Ux_gpt_gtciu_n_m GTCIUn) interrupts (interlinked operation with other interrupts and A/D converter start requests is available) Buffer transfer skipping function
Event linking (ELC) function	Available*1
Noise filtering function	Available
Logical operation between the channel output	Available

Note 1. See **5.7.6 Operations Linked by ELC**.

Note 2. x: 0, 1
m: 0 to 7

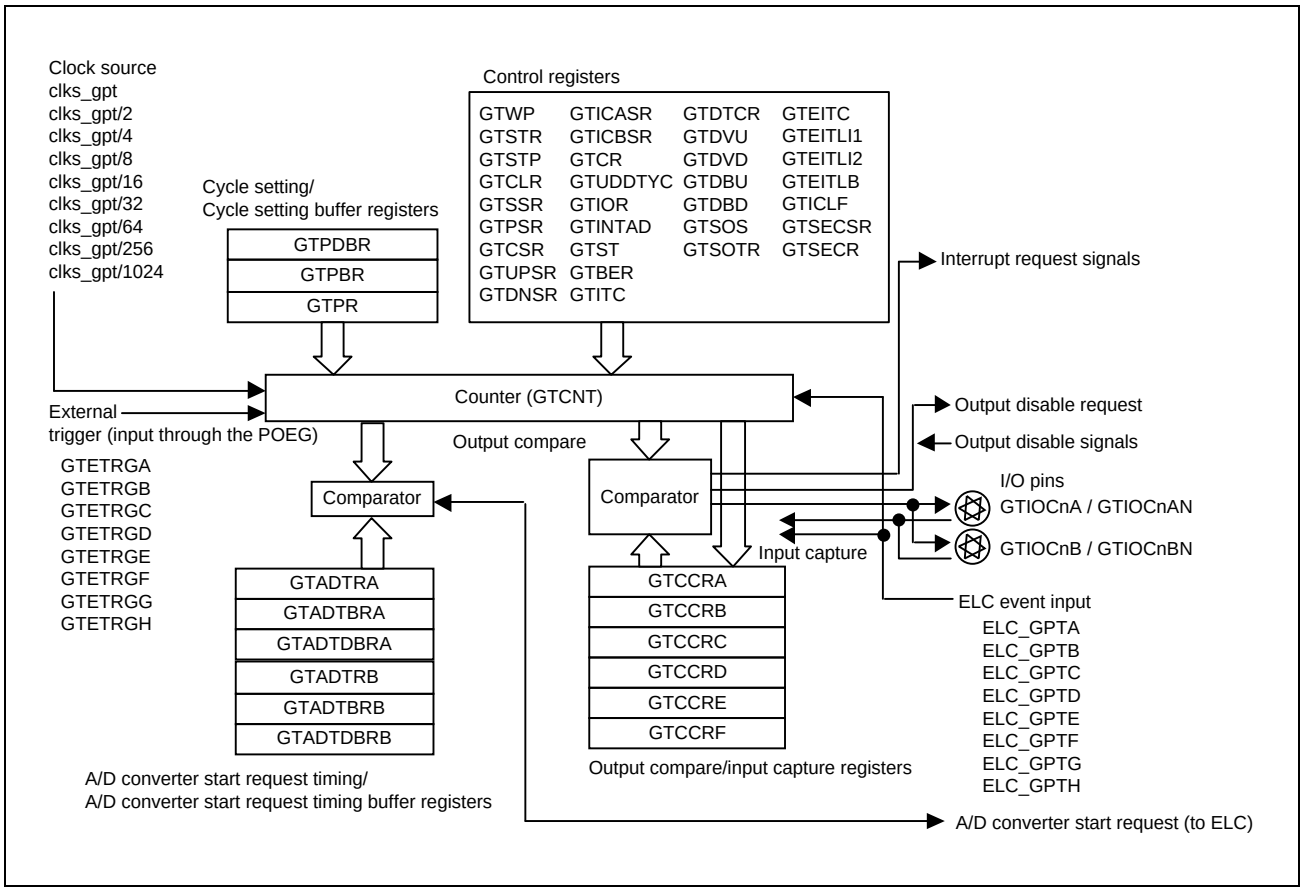


Figure 5.7-1 GPT Block Diagram

Figure 5.7-2 shows an example using multiple GPTs.

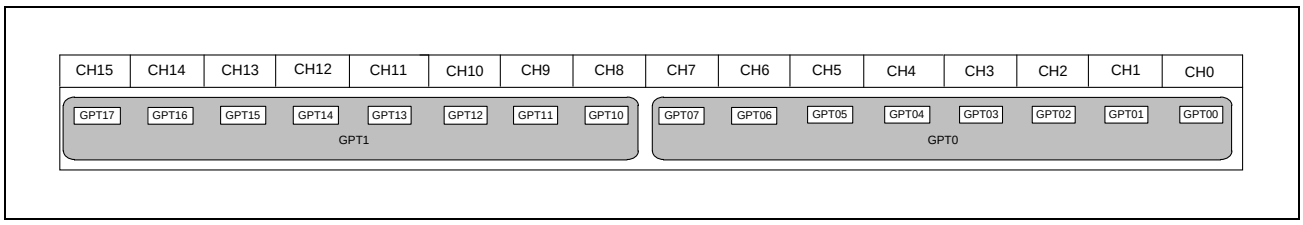


Figure 5.7-2 Association between GPT Channels and Module Names

Table 5.7-3 lists the I/O pins.

Table 5.7-3 GPT I/O Pins

Channel	Pin Name	Input/Output	Function
Common	GTETRGx	Input	External trigger input pin x (input through the POEG) x = A to D: For GPT0 x = E to H: For GPT1
GPT0	GTIOCnA	I/O	GTCCRA register input capture / output timer compare / PWM output pin
	GTIOCnB	I/O	GTCCRB register input capture / output timer compare / PWM output pin
	GTIOCnAN	I/O	Anti-phase signal with GPTIOnA
	GTIOCnBN	I/O	Anti-phase signal with GPTIOnB
GPT1	GTIOCmA	I/O	GTCCRA register input capture / output timer compare / PWM output pin
	GTIOCmB	I/O	GTCCRB register input capture / output timer compare / PWM output pin
	GTIOCmAN	I/O	Anti-phase signal with GPTIOmA
	GTIOCmBN	I/O	Anti-phase signal with GPTIOmB

Note: n: 0 to 7
m: 8 to 15

5.7.2 Registers

Table 5.7-4 Register Base Addresses

Base Address Name	Base Address	Target Module
<GPT0_base>	0_1301_0000h (4301_0000h*2, 5301_0000h*1)	GPT0n (n = 0 to 7)
<GPT1_base>	0_1302_0000h (4302_0000h*2, 5302_0000h*1)	GPT1n (n = 0 to 7)

Note 1. Cortex-M33 address space (non-secure)

Note 2. Cortex-M33 address space (secure)

5.7.2.1 List of Registers

The table below lists the registers.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
General Purpose Timer Write-Protection Register	GPTm_n_GTWP	0000_0000h	0000h + n x 0100h	32
General Purpose Timer Software Start Register	GPTm_n_GTSTR	0000_0000h	0004h + n x 0100h	32
General Purpose Timer Software Stop Register	GPTm_n_GTSTP	FFFF_FFFFh	0008h + n x 0100h	32
General Purpose Timer Software Clear Register	GPTm_n_GTCLR	0000_0000h	000Ch + n x 0100h	32
General Purpose Timer Start Source Select Register	GPTm_n_GTSSR	0000_0000h	0010h + n x 0100h	32
General Purpose Timer Stop Source Select Register	GPTm_n_GTPSR	0000_0000h	0014h + n x 0100h	32
General Purpose Timer Clear Source Select Register	GPTm_n_GTCSR	0000_0000h	0018h + n x 0100h	32
General Purpose Timer Up Count Source Select Register	GPTm_n_GTUPSR	0000_0000h	001Ch + n x 0100h	32
General Purpose Timer Down Count Source Select Register	GPTm_n_GTDNSR	0000_0000h	0020h + n x 0100h	32
General Purpose Timer Input Capture Source Select Register A	GPTm_n_GTICASR	0000_0000h	0024h + n x 0100h	32
General Purpose Timer Input Capture Source Select Register B	GPTm_n_GTICBSR	0000_0000h	0028h + n x 0100h	32
General Purpose Timer Control Register	GPTm_n_GTCR	0000_0000h	002Ch + n x 0100h	32
General Purpose Timer Count Direction and Duty Setting Register	GPTm_n_GTUDDTYC	0000_0001h	0030h + n x 0100h	32
General Purpose Timer I/O Control Register	GPTm_n_GTIOR	0000_0000h	0034h + n x 0100h	32
General Purpose Timer Interrupt Output Setting Register	GPTm_n_GTINTAD	0000_0000h	0038h + n x 0100h	32
General Purpose Timer Status Register	GPTm_n_GTST	0000_0000h	003Ch + n x 0100h	32
General Purpose Timer Buffer Enable Register	GPTm_n_GTBER	0000_0000h	0040h + n x 0100h	32
General-Purpose Timer Interrupt and A/D Converter Start Request Setting Register	GPTm_n_GTITC	0000_0000h	0044h + n x 0100h	32
General Purpose Timer Counter	GPTm_n_GTCNT	0000_0000h	0048h + n x 0100h	32
General Purpose Timer Compare Capture Register A	GPTm_n_GTCCRA	FFFF_FFFFh	004Ch + n x 0100h	32
General PWM Timer Compare Capture Register B	GPTm_n_GTCCRB	FFFF_FFFFh	0050h + n x 0100h	32
General PWM Timer Compare Capture Register C	GPTm_n_GTCCRC	FFFF_FFFFh	0054h + n x 0100h	32
General PWM Timer Compare Capture Register E	GPTm_n_GTCCRE	FFFF_FFFFh	0058h + n x 0100h	32
General PWM Timer Compare Capture Register D	GPTm_n_GTCCRD	FFFF_FFFFh	005Ch + n x 0100h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
General PWM Timer Compare Capture Register F	GPTm_n_GTCCRF	FFFF_FFFFh	0060h + n x 0100h	32
General Purpose Timer Cycle Setting Register	GPTm_n_GTPR	FFFF_FFFFh	0064h + n x 0100h	32
General Purpose Timer Cycle Setting Buffer Register	GPTm_n_GTPBR	FFFF_FFFFh	0068h + n x 0100h	32
General-Purpose Timer Cycle Setting Double-Buffer Register	GPTm_n_GTPDBR	FFFF_FFFFh	006Ch + n x 0100h	32
A/D Converter Start Request Timing Register A	GPTm_n_GTADTRA	FFFF_FFFFh	0070h + n x 0100h	32
A/D Converter Start Request Timing Buffer Register A	GPTm_n_GTADTBRA	FFFF_FFFFh	0074h + n x 0100h	32
A/D Converter Start Request Timing Double-Buffer Register A	GPTm_n_GTADTDBRA	FFFF_FFFFh	0078h + n x 0100h	32
A/D Converter Start Request Timing Register B	GPTm_n_GTADTRB	FFFF_FFFFh	007Ch + n x 0100h	32
A/D Converter Start Request Timing Buffer Register B	GPTm_n_GTADTBRB	FFFF_FFFFh	0080h + n x 0100h	32
A/D Converter Start Request Timing Double-Buffer Register B	GPTm_n_GTADTDBRB	FFFF_FFFFh	0084h + n x 0100h	32
General Purpose Timer Dead Time Control Register	GPTm_n_GTDTCR	0000_0000h	0088h + n x 0100h	32
General Purpose Timer Dead Time Value Register U	GPTm_n_GTDVU	FFFF_FFFFh	008Ch + n x 0100h	32
General PWM Timer Dead Time Value Register D	GPTm_n_GTDVD	FFFF_FFFFh	0090h + n x 0100h	32
General-Purpose Timer Dead Time Buffer Register U	GPTm_n_GTDBU	FFFF_FFFFh	0094h + n x 0100h	32
General-Purpose Timer Dead Time Buffer Register D	GPTm_n_GTDBD	FFFF_FFFFh	0098h + n x 0100h	32
General-Purpose Timer Output Protection Function Status Register	GPTm_n_GTSOS	0000_0000h	009Ch + n x 0100h	32
General-Purpose Timer Output Protection Function Temporary Release Register	GPTm_n_GTSOTR	0000_0000h	00A0h + n x 0100h	32
General-Purpose Timer Extended Interrupt Skipping Counter Control Register	GPTm_n_GTEITC	0000_0000h	00A8h + n x 0100h	32
General-Purpose Timer Extended Interrupt Skipping Setting Register 1	GPTm_n_GTEITLI1	0000_0000h	00ACh + n x 0100h	32
General-Purpose Timer Extended Interrupt Skipping Setting Register 2	GPTm_n_GTEITLI2	0000_0000h	00B0h + n x 0100h	32
General-Purpose Timer Extended Buffer Transfer Skipping Setting Register	GPTm_n_GTEITLB	0000_0000h	00B4h + n x 0100h	32
General Purpose Timer Inter Channel Logical Operation Function Setting Register	GPTm_n_GTICLF	0000_0000h	00B8h + n x 0100h	32
Reserve	-	-	00C0h + n x 0100h to 00CFh + n x 0100h	-
General Purpose Timer Operation Enable Bit Simultaneous Control Channel Select Register	GPTm_n_GTSECSR	0000_0000h	00D0h + n x 0100h	32
General Purpose Timer Operation Enable Bit Simultaneous Control Register	GPTm_n_GTSECR	0000_0000h	00D4h + n x 0100h	32

Note: m = 0, 1
n = 0 to 7

5.7.2.2 Register Description

The prefix (GPTm_n_) of the register names is omitted in this and subsequent sections.

5.7.2.2.1 General-Purpose Timer Write-Protection Register (GPTm_n_GTWP)

GTWP enables or disables writing to registers to prevent accidental modification. Protection by the GTWP register is only for the writes by the CPU. GTWP does not protect registers from updates that occur in association with CPU writes.

Access Size : 32 bits
Address : <GPTm_base> + 0000h + n x 0100h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PRKEY[7:0]								-	-	-	CMNWP	CLRWP	STPWP	STRWP	WP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 8	PRKEY[7:0]	-	W	GTWP Key Code When A5h is written to these bits, writing to the WP, STRWP, STPWP, CLRWP, and CMNWP bits is permitted. These bits are read as 0b.
7 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	CMNWP	0h	RW	Common Register Write Disable 0b: Writing to the registers is enabled 1b: Writing to the registers is disabled
3	CLRWP	0h	RW	GTCLR.CCLR Bit Write Disable 0b: Writing to the bit is enabled 1b: Writing to the bit is disabled
2	STPWP	0h	RW	GTSTP.CSTOP Bit Write Disable 0b: Writing to the bit is enabled 1b: Writing to the bit is disabled
1	STRWP	0h	RW	GTSTR.CSTRT Bit Write Disable 0b: Writing to the bit is enabled 1b: Writing to the bit is disabled
0	WP	0h	RW	Register Write Disable 0b: Writing to the register enabled 1b: Writing to the register disabled

WP bit (Register Write Disable)

The following is a list of write enabled or disabled registers:

GTSSR, GTPSR, GTCSR, GTUPSR, GTDNSR, GTICASR, GTICBSR, GTCR, GTUDDTYC, GTIOR, GTINTAD, GTST, GTBER, GTITC, GTCNT, GTCCRA, GTCCRB, GTCCRC, GTCCRD, GTCCRE, GTCCRF, GTPR, GTPBR, GTPDBR, GTADTRA, GTADTBRA, GTADTDBRA, GTADTRB, GTADTBRB, GTADTDBRB, GTDTCR, GTDVU, GTDVD, GTDBU, GTDBD, GTSOS, GTSOTR, GTEITC, GTEITLI1, GTEITLI2, GTEITLB, GTICLF.

STRWP bit (GTSTR.CSTRT Bit Write Disable)

The STRWP bit enables or disables starting the updating of counter values by writing to the CSTRTn bit (n = 0 to 15) corresponding to a channel number in the GTSTR register.

The bit position of each CSTRTn bit in the GTSTR register is allocated to the channel with the corresponding number, and writing to the GTSTR register for any channel results in writing to the registers of all channels. The STRWP bit for each channel does not control writing but only controls updating of the CSTRT bit for the corresponding channel when simultaneously writing to all channels.

Therefore, when writing to the CSTRT bits of a channel for which the setting of the STRWP bit is 1b (disabling writing), the CSTRT bit for the given channel is not updated, but the CSTRT bits corresponding to channel for which the setting of the STRWP bit is 0b (enabling writing) are updated. For example, when the setting of the GPT00.GTWP.STRWP bit is 0b (enabling writing), writing 1b to the GPT01.GTSTR.CSTRT0 bit when its current setting is 0b causes the value to be updated, and the GPT00.GTCNT counter starts to run. When the setting of the GPT00.GTWP.STRWP bit is 1b (disabling writing), writing 1b to the GPT01.GTSTR.CSTRT0 bit when its current setting is 0b leaves the bit with the value 0b, and the GPT00.GTCNT counter does not run.

If you want to protect all bits in the GTSTR register from being updated, set the STRWP bits of all channels to 1b.

STPWP bit (GTSTP.CSTOP Bit Write Disable)

The STPWP bit enables or disables starting the updating of counter values by writing to the CSTOPn bit (n = 0 to 15) corresponding to a channel number in the GTSTP register.

The bit position of each CSTOPn bit in the GTSTP registers is allocated to the channel with the corresponding number, and the writing to the GTSTP register for any channel results in writing to the registers of all channels. The STPWP bit for each channel does not control writing but only controls updating of the CSTOP bit for the corresponding channel when simultaneously writing to all channels.

Therefore, when writing to the CSTOP bits of a channel for which the setting of the STPWP bit is 1b(disabling writing), the CSTOP bit for the given channel is not updated, but the CSTOP bits corresponding to channel for which the setting of the STPWP bit is 0b (enabling writing) are updated. For example, when the setting of the GPT00.GTWP.STPWP bit is 0b (enabling writing), writing 1b to the GPT01.GTSTP.CSTOP0 bit when its current setting is 0b causes the value to be updated, and the GPT00.GTCNT counter is stopped. When the setting of the GPT00.GTWP.STPWP bit is 1b (disabling writing), writing 1b to the GPT01.GTSTP.CSTOP0 bit when its current setting is 0b leaves the bit with the value 0b, and the GPT00.GTCNT counter is not stopped.

If you want to protect all bits in the GTSTP register from being updated, set the STPWP bits of all channels to 1b.

CLRWP bit (GTCLR.CCLR Bit Write Disable)

CLRWP bit enables or disables starting the updating of counter values by writing to the CCLRn bit (n = 0 to 7) corresponding to a channel number in the GTCLR register.

The bit position of each CCLRn bit in the GTCLR registers is allocated to the channel with the corresponding number, and the writing to the GTCLR register for any channel results in writing to the registers of all channels. The CLRWP bit for each channel does not control writing but only controls updating of the CCLR bit for the corresponding channel when simultaneously writing to all channels.

Therefore, when writing to the CCLR bits of a channel for which the setting of the CLRWP bit is 1b (disabling writing), the CCLR bit for the given channel is not updated, but the CCLR bits corresponding to channel for which the setting of the CLRWP bit is 0b (enabling writing) are updated. For example, when the setting of the GPT00.GTWP.CLRWP bit is 0b (enabling writing), writing 1b to the GPT01.GTCLR.CCLR0 bit when its current setting is 0b causes the value to be updated, and the GPT00.GTCNT counter is cleared. When the setting of the GPT00.GTWP.CLRWP bit is 1b (disabling writing), writing 1b to the GPT01.GTCLR.CCLR0 bit when its current setting is 0b leaves the bit with the value 0b, and the GPT00.GTCNT counter is not cleared.

If you want to protect all bits in the GTCLR register from being updated, set the CLRWP bits of all channels to 1b.

CMNWP bit (Common Register Write Disable)

CMNWP bit enables or disables starting the updating of counter values by writing to the SECSELn bit (n = 0 to 15) corresponding to a channel number in the GTSECSR register or to the GTSECR register.

The bit position of each SECSEL bit in the GTSECSR registers is allocated to the channel with the corresponding number, and the writing to the GTSECSR register for any channel results in writing to the registers of all channels. Writing to the GTSECR register of any channel leads to writing to the registers of all channels. The CMNWP bit for each channel does not control writing but only controls updating of the SECSEL bit and the GTSECR register value for the corresponding channel when simultaneously writing to all channels.

Therefore, when writing to the SECSEL bit and the GTSECR register value of a channel for which the setting of the CMNWP bit is 1b(disabling writing), the SECSEL bit and the GTSECR register value for the given channel is not updated, but the SECSEL bit and the GTSECR register value corresponding to channel for which the setting of the CMNWP bit is 0 (enabling writing) are updated.

For example, when the setting of the GPT00.GTWP.CMNWP bit is 0 (enabling writing), writing to the GPT01.GTSECSR.SECSEL0 bit causes the value of the GPT00.GTSECSR.SECSEL0 bit to be updated. In the same way, writing to the GPT01.GTSECR register updates the value of the GPT00.GTSECR register. When the setting of the GPT00.GTWP.CMNWP bit is 1b (disabling writing), writing to the GPT01.GTSECSR.SECSEL0 bit does not cause the value of the GPT00.GTSECSR.SECSEL0 bit to be updated. In the same way, writing to the GPT01.GTSECR register does not update the value of the GPT00.GTSECR register.

If you want to protect all bits in the GTSECSR and GTSECR registers from being updated, set the CMNWP bits of all channels to 1b.

PRKEY[7:0] bits (GTWP Key Code)

These bits control whether the WP, STRWP, STPWP, CLRWP, and CMNWP bits can be overwritten.

5.7.2.2.2 General-Purpose Timer Software Start Register (GPTm_n_GTSTR)

The GTSTR starts operation of the GTCNT counter of channel n (n = 0 to 15).

The GTSTR bit number represents the channel number. The GTSTR register of each channel is shared by all of the channels. The GTCNT counter starts for the channel corresponding to the GTSTR bit number where 1b is written. Writing 0b has no effect on the status of GTCNT counter and the value of GTSTR register.

Access Size : 32 bits
Address : <GPTm_base> + 0004h + n x 0100h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSTRT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 0	CSTRT[15:0]	0h	RW (R)	GPTmn GTCNT Count Start 0b: GTCNT counter is not started 1b: GTCNT counter is started

CSTRTn bits (Channel n GTCNT Count Start (n = 0 to 15))

The CSTRTn bits start operation of the GTCNT counter of channel n. Writing to the GTSTR.CSTRTn bit (n = 0 to 15) has no effect unless the GTSSR.CSTRT bit is set to 1b.

The read data shows the counter status of each channel (GTCR.CST bit). A value of 0 means the counter is stopped and 1 means the counter is running.

Note that the definitions of the bits differ between GPT0 and GPT1. For details, see the table below.

Table 5.7-5 Definitions of Count Start Bits

	GPT0		GPT1	
Bit	Bit Description	Bit Attribute	Bit Description	Bit Attribute
0	GPT00 GTCNT Count Start	RW	GPT10 GTCNT Count Start	RW
1	GPT01 GTCNT Count Start	RW	GPT11 GTCNT Count Start	RW
2	GPT02 GTCNT Count Start	RW	GPT12 GTCNT Count Start	RW
3	GPT03 GTCNT Count Start	RW	GPT13 GTCNT Count Start	RW
4	GPT04 GTCNT Count Start	RW	GPT14 GTCNT Count Start	RW
5	GPT05 GTCNT Count Start	RW	GPT15 GTCNT Count Start	RW
6	GPT06 GTCNT Count Start	RW	GPT16 GTCNT Count Start	RW
7	GPT07 GTCNT Count Start	RW	GPT17 GTCNT Count Start	RW
8	GPT10 GTCNT Count Start	R	GPT00 GTCNT Count Start	R
9	GPT11 GTCNT Count Start	R	GPT01 GTCNT Count Start	R
10	GPT12 GTCNT Count Start	R	GPT02 GTCNT Count Start	R
11	GPT13 GTCNT Count Start	R	GPT03 GTCNT Count Start	R
12	GPT14 GTCNT Count Start	R	GPT04 GTCNT Count Start	R
13	GPT15 GTCNT Count Start	R	GPT05 GTCNT Count Start	R
14	GPT16 GTCNT Count Start	R	GPT06 GTCNT Count Start	R
15	GPT17 GTCNT Count Start	R	GPT07 GTCNT Count Start	R
31 - 16	Reserved	RW	Reserved	RW

5.7.2.2.3 General-Purpose Timer Software Stop Register (GPTm_n_GTSTP)

The GTSTP stops operation of the GTCNT counter of channel n (n = 0 to 15).

The GTSTP bit number represents the channel number. The GTSTP register of each channel is shared by all the channels. The GTCNT counter stops for the channel corresponding to the GTSTP bit number where 1b is written. Writing 0b has no effect on the status of the GTCNT counter and the value of GTSTP register.

Access Size : 32 bits
Address : <GPTm_base> + 0008h + n x 0100h
Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSTOP[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 1	RW	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
15 to 0	CSTOP[15:0]	FFFFh	RW (R)	GPTmn GTCNT Count Stop 0b: GTCNT counter is not stopped 1b: GTCNT counter is stopped

CSTOPn bits (Channel n GTCNT Count Stop (n = 0 to 15))

The CSTOPn bits stop operation of the GTCNT counter of channel n. Writing to the GTSTP.CSTOPn bit (n = 0 to 15) has no effect unless the GTPSR.CSTOP bit is set to 1b. The read data shows the counter status of each channel (invert of GTCR.CST bit). A value of 0b means the counter is running and 1b means the counter is stopped.

Note that the definitions of the bits differ between GPT0 and GPT1. For details, see the table below.

Table 5.7-6 Definitions of Count Stop Bits

	GPT0		GPT1	
Bit	Bit Description	Bit Attribute	Bit Description	Bit Attribute
0	GPT00 GTCNT Count Stop	RW	GPT10 GTCNT Count Stop	RW
1	GPT01 GTCNT Count Stop	RW	GPT11 GTCNT Count Stop	RW
2	GPT02 GTCNT Count Stop	RW	GPT12 GTCNT Count Stop	RW
3	GPT03 GTCNT Count Stop	RW	GPT13 GTCNT Count Stop	RW
4	GPT04 GTCNT Count Stop	RW	GPT14 GTCNT Count Stop	RW
5	GPT05 GTCNT Count Stop	RW	GPT15 GTCNT Count Stop	RW
6	GPT06 GTCNT Count Stop	RW	GPT16 GTCNT Count Stop	RW
7	GPT07 GTCNT Count Stop	RW	GPT17 GTCNT Count Stop	RW
8	GPT10 GTCNT Count Stop	R	GPT00 GTCNT Count Stop	R
9	GPT11 GTCNT Count Stop	R	GPT01 GTCNT Count Stop	R
10	GPT12 GTCNT Count Stop	R	GPT02 GTCNT Count Stop	R
11	GPT13 GTCNT Count Stop	R	GPT03 GTCNT Count Stop	R
12	GPT14 GTCNT Count Stop	R	GPT04 GTCNT Count Stop	R
13	GPT15 GTCNT Count Stop	R	GPT05 GTCNT Count Stop	R
14	GPT16 GTCNT Count Stop	R	GPT06 GTCNT Count Stop	R
15	GPT17 GTCNT Count Stop	R	GPT07 GTCNT Count Stop	R
31 - 16	Reserved	RW	Reserved	RW

5.7.2.2.4 General-Purpose Timer Software Clear Register (GPTm_n_GTCLR)

The GTCLR is a write-only register that clears operation of the GTCNT counter of channel n (n = 0 to 15).

The GTCLR bit number represents the channel number. The GTCLR register of each channel is shared by all the channels. The GTCNT counter is cleared for the channel corresponding to the GTCLR bit number where 1b is written. Writing 0b has no effect on the status of GTCNT counter.

Access Size : 32 bits
Address : <GPTm_base> + 000Ch + n x 0100h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CCLR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	W	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 0	CCLR[7:0]	0h	W	GPTmn GTCNT Count Clear 0b: GTCNT counter is not cleared 1b: GTCNT counter is cleared

CCLRn bits (Channel n GTCNT Count Clear (n = 0 to 15))

Writing 1 to these bits clears the value of the GTCNT counter of channel 0. These bits are read as 0.

Note that the definitions of the bits differ between GPT0 and GPT1. For details, see the table below.

Table 5.7-7 Definitions of Count Clear Bits

	GPT0		GPT1	
Bit	Bit Description	Bit Attribute	Bit Description	Bit Attribute
0	GPT00 GTCNT Count Clear	W	GPT10 GTCNT Count Clear	W
1	GPT01 GTCNT Count Clear	W	GPT11 GTCNT Count Clear	W
2	GPT02 GTCNT Count Clear	W	GPT12 GTCNT Count Clear	W
3	GPT03 GTCNT Count Clear	W	GPT13 GTCNT Count Clear	W
4	GPT04 GTCNT Count Clear	W	GPT14 GTCNT Count Clear	W
5	GPT05 GTCNT Count Clear	W	GPT15 GTCNT Count Clear	W
6	GPT06 GTCNT Count Clear	W	GPT16 GTCNT Count Clear	W
7	GPT07 GTCNT Count Clear	W	GPT17 GTCNT Count Clear	W
31 - 8	Reserved	W	Reserved	W

5.7.2.2.5 General-Purpose Timer Start Source Select Register (GPTm_n_GTSSR)

The GTSSR sets the source to start the GTCNT counter.

Input from GTETRGn (n = A to H) pins are input to the GPT via the POEG. Set the polarity of these signals with the POEG.

Access Size : 32 bits

Address : <GPTm_base> + 0010h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSTRT	-	-	-	-	-	-	-	SSELCH	SSELCG	SSELCF	SSELCE	SSELCD	SSELCC	SSELCB	SSELCA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SSCBFAH	SSCBFAL	SSCBRAH	SSCBRAL	SSCAFBH	SSCAFBL	SSCARBH	SSCARBL	SSGTRGDF	SSGTRGDR	SSGTRGCF	SSGTRGCR	SSGTRGBF	SSGTRGBR	SSGTRGAF	SSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CSTRT	0h	RW	Software Source Counter Start Enable 0b: Counter start disabled by the GTSTR register 1b: Counter start enabled by the GTSTR register
30 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23	SSELCH	0h	RW	ELC_GPTH Event Source Counter Start Enable 0b: Counter start disabled at the ELC_GPTH input 1b: Counter start enabled at the ELC_GPTH input
22	SSELCG	0h	RW	ELC_GPTG Event Source Counter Start Enable 0b: Counter start disabled at the ELC_GPTG input 1b: Counter start enabled at the ELC_GPTG input
21	SSELCF	0h	RW	ELC_GPTF Event Source Counter Start Enable 0b: Counter start disabled at the ELC_GPTF input 1b: Counter start enabled at the ELC_GPTF input
20	SSELCE	0h	RW	ELC_GPTE Event Source Counter Start Enable 0b: Counter start disabled at the ELC_GPTE input 1b: Counter start enabled at the ELC_GPTE input
19	SSELCD	0h	RW	ELC_GPTD Event Source Counter Start Enable 0b: Counter start disabled at the ELC_GPTD input 1b: Counter start enabled at the ELC_GPTD input
18	SSELCC	0h	RW	ELC_GPTC Event Source Counter Start Enable 0b: Counter start disabled at the ELC_GPTC input 1b: Counter start enabled at the ELC_GPTC input
17	SSELCB	0h	RW	ELC_GPTB Event Source Counter Start Enable 0b: Counter start disabled at the ELC_GPTB input 1b: Counter start enabled at the ELC_GPTB input
16	SSELCA	0h	RW	ELC_GPTA Event Source Counter Start Enable 0b: Counter start disabled at the ELC_GPTA input 1b: Counter start enabled at the ELC_GPTA input
15	SSCBFAH	0h	RW	GTIOCnB Pin Fall Input Source Counter Start Enable when GTIOCnA Value is High 0b: Counter start disabled on the falling edge of GTIOCnB input when GTIOCnA input is 1 1b: Counter start enabled on the falling edge of GTIOCnB input when GTIOCnA input is 1
14	SSCBFAL	0h	RW	GTIOCnB Pin Fall Input Source Counter Start Enable when GTIOCnA Value is Low 0b: Counter start disabled on the falling edge of GTIOCnB input when GTIOCnA input is 0 1b: Counter start enabled on the falling edge of GTIOCnB input when GTIOCnA input is 0
13	SSCBRAH	0h	RW	GTIOCnB Pin Rise Input Source Counter Start Enable when GTIOCnA Value is High 0b: Counter start disabled on the rising edge of GTIOCnB input when GTIOCnA input is 1 1b: Counter start enabled on the rising edge of GTIOCnB input when GTIOCnA input is 1

Bit	Bit Name	Initial Value	R/W	Description
12	SSCBRAL	0h	RW	GTIOCnB Pin Rise Input Source Counter Start Enable when GTIOCnA Value is Low 0b: Counter start disabled on the rising edge of GTIOCnB input when GTIOCnA input is 0 1b: Counter start enabled on the rising edge of GTIOCnB input when GTIOCnA input is 0
11	SSCAFBH	0h	RW	GTIOCnA Pin Fall Input Source Counter Start Enable when GTIOCnB Value is High 0b: Counter start disabled on the falling edge of GTIOCnA input when GTIOCnB input is 1 1b: Counter start enabled on the falling edge of GTIOCnA input when GTIOCnB input is 1
10	SSCAFBL	0h	RW	GTIOCnA Pin Fall Input Source Counter Start Enable when GTIOCnB Value is Low 0b: Counter start disabled on the falling edge of GTIOCnA input when GTIOCnB input is 0 1b: Counter start enabled on the falling edge of GTIOCnA input when GTIOCnB input is 0
9	SSCARBH	0h	RW	GTIOCnA Pin Rise Input Source Counter Start Enable when GTIOCnB Value is High 0b: Counter start disabled on the rising edge of GTIOCnA input when GTIOCnB input is 1 1b: Counter start enabled on the rising edge of GTIOCnA input when GTIOCnB input is 1
8	SSCARBL	0h	RW	GTIOCnA Pin Rise Input Source Counter Start Enable when GTIOCnB Value is Low 0b: Counter start disabled on the rising edge of GTIOCnA input when GTIOCnB input is 0 1b: Counter start enabled on the rising edge of GTIOCnA input when GTIOCnB input is 0
7	SSGTRGDF	0h	RW	GTETRGn Pin Fall Input Source Counter Start Enable (n = D, H) 0b: Counter start disabled on the falling edge of GTETRGn input 1b: Counter start enabled on the falling edge of GTETRGn input
6	SSGTRGDR	0h	RW	GTETRGn Pin Rise Input Source Counter Start Enable (n = D, H) 0b: Counter start disabled on the rising edge of GTETRGn input 1b: Counter start enabled on the rising edge of GTETRGn input
5	SSGTRGCF	0h	RW	GTETRGn Pin Fall Input Source Counter Start Enable (n = C, G) 0b: Counter start disabled on the falling edge of GTETRGn input 1b: Counter start enabled on the falling edge of GTETRGn input
4	SSGTRGCR	0h	RW	GTETRGn Pin Rise Input Source Counter Start Enable (n = C, G) 0b: Counter start disabled on the rising edge of GTETRGn input 1b: Counter start enabled on the rising edge of GTETRGn input
3	SSGTRGBF	0h	RW	GTETRGn Pin Fall Input Source Counter Start Enable (n = B, F) 0b: Counter start disabled on the falling edge of GTETRGn input 1b: Counter start enabled on the falling edge of GTETRGn input
2	SSGTRGBR	0h	RW	GTETRGn Pin Rise Input Source Counter Start Enable (n = B, F) 0b: Counter start disabled on the rising edge of GTETRGn input 1b: Counter start enabled on the rising edge of GTETRGn input
1	SSGTRGAF	0h	RW	GTETRGn Pin Fall Input Source Counter Start Enable (n = A, E) 0b: Counter start disabled on the falling edge of GTETRGn input 1b: Counter start enabled on the falling edge of GTETRGn input
0	SSGTRGAR	0h	RW	GTETRGn Pin Rise Input Source Counter Start Enable (n = A, E) 0b: Counter start disabled on the rising edge of GTETRGn input 1b: Counter start enabled on the rising edge of GTETRGn input

SSGTRGAR bit (GTETRGA/GTETRGE Pin Rise Input Source Counter Start Enable)

The SSGTRGAR bit enables or disables starting of the GTCNT counter on the rising edge of the GTETRGA/GTETRGE pin input.

SSGTRGAF bit (GTETRGA/GTETRGE Pin Fall Input Source Counter Start Enable)

The SSGTRGAF bit enables or disables starting of the GTCNT counter on the falling edge of the GTETRGA/GTETRGE pin input.

SSGTRGBR bit (GTETRGB/GTETRGF Pin Rise Input Source Counter Start Enable)

The SSGTRGBR bit enables or disables starting of the GTCNT counter on the rising edge of the GTETRGB/GTETRGF pin input.

SSGTRGBF bit (GTETRGB/GTETRGF Pin Fall Input Source Counter Start Enable)

The SSGTRGBF bit enables or disables starting of the GTCNT counter on the falling edge of the GTETRGB/GTETRGF pin input.

SSGTRGCR bit (GTETRGC/GTETRGG Pin Rise Input Source Counter Start Enable)

The SSGTRGCR bit enables or disables starting of the GTCNT counter on the rising edge of the GTETRGC/GTETRGG pin input.

SSGTRGCF bit (GTETRGC/GTETRGG Pin Fall Input Source Counter Start Enable)

The SSGTRGCF bit enables or disables starting of the GTCNT counter on the falling edge of the GTETRGC/GTETRGG pin input.

SSGTRGDR bit (GTETRGD/GTETRGH Pin Rise Input Source Counter Start Enable)

The SSGTRGDR bit enables or disables starting of the GTCNT counter on the rising edge of the GTETRGD/GTETRGH pin input.

SSGTRGDF bit (GTETRGD/GTETRGH Pin Fall Input Source Counter Start Enable)

The SSGTRGDF bit enables or disables starting of the GTCNT counter on the falling edge of the GTETRGD/GTETRGH pin input.

SSCARBL bit (GTIOCnA Pin Rise Input Source Counter Start Enable when GTIOCnB Value is Low)

The SSCARBL bit enables or disables starting of the GTCNT counter on the rising edge of the GTIOCnA pin input, when GTIOCnB input is 0.

SSCARBH bit (GTIOCnA Pin Rise Input Source Counter Start Enable when GTIOCnB Value is High)

The SSCARBH bit enables or disables starting of the GTCNT counter on the rising edge of the GTIOCnA pin input, when GTIOCnB input is 1.

SSCAFBL bit (GTIOCnA Pin Fall Input Source Counter Start Enable when GTIOCnB Value is Low)

The SSCAFBL bit enables or disables starting of the GTCNT counter on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 0.

SSCAFBH bit (GTIOCnA Pin Fall Input Source Counter Start Enable when GTIOCnB Value is High)

The SSCAFBH bit enables or disables starting of the GTCNT counter on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 1.

SSCBRAL bit (GTIOCnB Pin Rise Input Source Counter Start Enable when GTIOCnA Value is Low)

The SSCBRAL bit enables or disables starting of the GTCNT counter on the rising edge of the GTIOCnB pin input, when GTIOCnA input is 0.

SSCBRAH bit (GTIOCnB Pin Rise Input Source Counter Start Enable when GTIOCnA Value is High)

The SSCBRAH bit enables or disables starting of the GTCNT counter on the rising edge of the GTIOCnB pin input, when GTIOCnA input is 1.

SSCBFAL bit (GTIOCnB Pin Fall Input Source Counter Start Enable when GTIOCnA Value is Low)

The SSCBFAL bit enables or disables starting of the GTCNT counter on the falling edge of the GTIOCnB pin input, when GTIOCnA input is 0.

SSCBFAH bit (GTIOCnB Pin Fall Input Source Counter Start Enable when GTIOCnA Value is High)

The SSCBFAH bit enables or disables starting of the GTCNT counter on the falling edge of the GTIOCnB pin input, when GTIOCnA input is 1.

SSELCm bit (ELC_GPTm Event Source Counter Start Enable) (m = A to H)

The SSELCm bit enables or disables starting of the GTCNT counter at the ELC_GPTm event input.

CSTRT bit (Software Source Counter Start Enable)

The CSTRT bit enables or disables starting of the GTCNT counter by the GTSTR register.

5.7.2.2.6 General-Purpose Timer Stop Source Select Register (GPTm_n_GTPSR)

The GTPSR sets the source to stop the GTCNT counter.

Inputs from GTETRGn (n = A to H) pins are input to the GPT via the POEG. Set the polarity of these signals with the POEG.

Access Size : 32 bits

Address : <GPTm_base> + 0014h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSTOP	-	-	-	-	-	-	-	PSELCH	PSELCG	PSELCF	PSELCE	PSELCD	PSELCC	PSELCB	PSELCA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PSCBFAH	PSCBFAL	PSCBRAH	PSCBRAL	PSCAFBH	PSCAFBL	PSCARBH	PSCARBL	PSGTRGDF	PSGTRGDR	PSGTRGCF	PSGTRGCR	PSGTRGBF	PSGTRGBR	PSGTRGAF	PSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CSTOP	0h	RW	Software Source Counter Stop Enable 0b: Counter stop disabled by the GTSTP register 1b: Counter stop enabled by the GTSTP register
30 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23	PSELCH	0h	RW	ELC_GPTH Event Source Counter Stop Enable 0b: Counter stop disabled at the ELC_GPTH input 1b: Counter stop enabled at the ELC_GPTH input
22	PSELCG	0h	RW	ELC_GPTG Event Source Counter Stop Enable 0b: Counter stop disabled at the ELC_GPTG input 1b: Counter stop enabled at the ELC_GPTG input
21	PSELCF	0h	RW	ELC_GPTF Event Source Counter Stop Enable 0b: Counter stop disabled at the ELC_GPTF input 1b: Counter stop enabled at the ELC_GPTF input
20	PSELCE	0h	RW	ELC_GPTE Event Source Counter Stop Enable 0b: Counter stop disabled at the ELC_GPTE input 1b: Counter stop enabled at the ELC_GPTE input
19	PSELCD	0h	RW	ELC_GPTD Event Source Counter Stop Enable 0b: Counter stop disabled at the ELC_GPTD input 1b: Counter stop enabled at the ELC_GPTD input
18	PSELCC	0h	RW	ELC_GPTC Event Source Counter Stop Enable 0b: Counter stop disabled at the ELC_GPTC input 1b: Counter stop enabled at the ELC_GPTC input
17	PSELCB	0h	RW	ELC_GPTB Event Source Counter Stop Enable 0b: Counter stop disabled at the ELC_GPTB input 1b: Counter stop enabled at the ELC_GPTB input
16	PSELCA	0h	RW	ELC_GPTA Event Source Counter Stop Enable 0b: Counter stop disabled at the ELC_GPTA input 1b: Counter stop enabled at the ELC_GPTA input
15	PSCBFAH	0h	RW	GTIOCnB Pin Fall Input Source Counter Stop Enable when GTIOCnA Value is High 0b: Counter stop disabled on the falling edge of GTIOCnB input when GTIOCnA input is 1 1b: Counter stop enabled on the falling edge of GTIOCnB input when GTIOCnA input is 1
14	PSCBFAL	0h	RW	GTIOCnB Pin Fall Input Source Counter Stop Enable when GTIOCnA Value is Low 0b: Counter stop disabled on the falling edge of GTIOCnB input when GTIOCnA input is 0 1b: Counter stop enabled on the falling edge of GTIOCnB input when GTIOCnA input is 0
13	PSCBRAH	0h	RW	GTIOCnB Pin Rise Input Source Counter Stop Enable when GTIOCnA Value is High 0b: Counter stop disabled on the rising edge of GTIOCnB input when GTIOCnA input is 1 1b: Counter stop enabled on the rising edge of GTIOCnB input when GTIOCnA input is 1

Bit	Bit Name	Initial Value	R/W	Description
12	PSCBRAL	0h	RW	GTIOCnB Pin Rise Input Source Counter Stop Enable when GTIOCnA Value is Low 0b: Counter stop disabled on the rising edge of GTIOCnB input when GTIOCnA input is 0 1b: Counter stop enabled on the rising edge of GTIOCnB input when GTIOCnA input is 0
11	PSCAFBH	0h	RW	GTIOCnA Pin Fall Input Source Counter Stop Enable when GTIOCnB Value is High 0b: Counter stop disabled on the falling edge of GTIOCnA input when GTIOCnB input is 1 1b: Counter stop enabled on the falling edge of GTIOCnA input when GTIOCnB input is 1
10	PSCAFBL	0h	RW	GTIOCnA Pin Fall Input Source Counter Stop Enable when GTIOCnB Value is Low 0b: Counter stop disabled on the falling edge of GTIOCnA input when GTIOCnB input is 0 1b: Counter stop enabled on the falling edge of GTIOCnA input when GTIOCnB input is 0
9	PSCARBH	0h	RW	GTIOCnA Pin Rise Input Source Counter Stop Enable when GTIOCnB Value is High 0b: Counter stop disabled on the rising edge of GTIOCnA input when GTIOCnB input is 1 1b: Counter stop enabled on the rising edge of GTIOCnA input when GTIOCnB input is 1
8	PSCARBL	0h	RW	GTIOCnA Pin Rise Input Source Counter Stop Enable when GTIOCnB Value is Low 0b: Counter stop disabled on the rising edge of GTIOCnA input when GTIOCnB input is 0 1b: Counter stop enabled on the rising edge of GTIOCnA input when GTIOCnB input is 0
7	PSGTRGDF	0h	RW	GTETRGn Pin Fall Input Source Counter Stop Enable (n = D, H) 0b: Counter stop disabled on the falling edge of GTETRGn input 1b: Counter stop enabled on the falling edge of GTETRGn input
6	PSGTRGDR	0h	RW	GTETRGn Pin Rise Input Source Counter Stop Enable (n = D, H) 0b: Counter stop disabled on the rising edge of GTETRGn input 1b: Counter stop enabled on the rising edge of GTETRGn input
5	PSGTRGCF	0h	RW	GTETRGn Pin Fall Input Source Counter Stop Enable (n = C, G) 0b: Counter stop disabled on the falling edge of GTETRGn input 1b: Counter stop enabled on the falling edge of GTETRGn input
4	PSGTRGCR	0h	RW	GTETRGn Pin Rise Input Source Counter Stop Enable (n = C, G) 0b: Counter stop disabled on the rising edge of GTETRGn input 1b: Counter stop enabled on the rising edge of GTETRGn input
3	PSGTRGBF	0h	RW	GTETRGn Pin Fall Input Source Counter Stop Enable (n = B, F) 0b: Counter stop disabled on the falling edge of GTETRGn input 1b: Counter stop enabled on the falling edge of GTETRGn input
2	PSGTRGBR	0h	RW	GTETRGn Pin Rise Input Source Counter Stop Enable (n = B, F) 0b: Counter stop disabled on the rising edge of GTETRGn input 1b: Counter stop enabled on the rising edge of GTETRGn input
1	PSGTRGAF	0h	RW	GTETRGn Pin Fall Input Source Counter Stop Enable (n = A, E) 0b: Counter stop disabled on the falling edge of GTETRGn input 1b: Counter stop enabled on the falling edge of GTETRGn input
0	PSGTRGAR	0h	RW	GTETRGn Pin Rise Input Source Counter Stop Enable (n = A, E) 0b: Counter stop disabled on the rising edge of GTETRGn input 1b: Counter stop enabled on the rising edge of GTETRGn input

PSGTRGAR bit (GTETRGA/GTETRGE Pin Rise Input Source Counter Stop Enable)

The PSGTRGAR bit enables or disables stopping of the GTCNT counter on the rising edge of the GTETRGA/GTETRGE pin input.

PSGTRGAF bit (GTETRGA/GTETRGE Pin Fall Input Source Counter Stop Enable)

The PSGTRGAF bit enables or disables stopping of the GTCNT counter on the falling edge of the GTETRGA/GTETRGE pin input.

PSGTRGBR bit (GTETRGB/GTETRGF Pin Rise Input Source Counter Stop Enable)

PSGTRGBR bit enables or disables stopping of the GTCNT counter on the rising edge of the GTETRGB/GTETRGF pin input.

PSGTRGBF bit (GTETRGB/GTETRGF Pin Fall Input Source Counter Stop Enable)

The PSGTRGBF bit enables or disables stopping of the GTCNT counter on the falling edge of the GTETRGB/GTETRGF pin input.

PSGTRGCR bit (GTETRGC/GTETRGG Pin Rise Input Source Counter Stop Enable)

PSGTRGCR bit enables or disables stopping of the GTCNT counter on the rising edge of the GTETRGC/GTETRGG pin input.

PSGTRGCF bit (GTETRGC/GTETRGG Pin Fall Input Source Counter Stop Enable)

The PSGTRGCF bit enables or disables stopping of the GTCNT counter on the falling edge of the GTETRGC/GTETRGG pin input.

PSGTRGDR bit (GTETRGD/GTETRGH Pin Rise Input Source Counter Stop Enable)

PSGTRGDR bit enables or disables stopping of the GTCNT counter on the rising edge of the GTETRGD/GTETRGH pin input.

PSGTRGDF bit (GTETRGD/GTETRGH Pin Fall Input Source Counter Stop Enable)

The PSGTRGDF bit enables or disables stopping of the GTCNT counter on the falling edge of the GTETRGD/GTETRGH pin input.

PSCARBL bit (GTIOCnA Pin Rise Input Source Counter Stop Enable when GTIOCnB Value is Low)

The PSCARBL bit enables or disables stopping of the GTCNT counter on the rising edge of the GTIOCnA pin input, when GTIOCnB input is 0.

PSCARBH bit (GTIOCnA Pin Rise Input Source Counter Stop Enable when GTIOCnB Value is High)

The PSCARBH bit enables or disables stopping of the GTCNT counter on the rising edge of the GTIOCnA pin input, when GTIOCnB input is 1.

PSCAFBL bit (GTIOCnA Pin Fall Input Source Counter Stop Enable when GTIOCnB Value is Low)

The PSCAFBL bit enables or disables stopping of the GTCNT counter on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 0.

PSCAFBH bit (GTIOCnA Pin Fall Input Source Counter Stop Enable when GTIOCnB Value is High)

The PSCAFBH bit enables or disables stopping of the GTCNT counter on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 1.

PSCBRAL bit (GTIOCnB Pin Rise Input Source Counter Stop Enable when GTIOCnA Value is Low)

The PSCBRAL bit enables or disables stopping of the GTCNT counter on the rising edge of the GTIOCnB pin input, when GTIOCnA input is 0.

PSCBRAH bit (GTIOCnB Pin Rise Input Source Counter Stop Enable when GTIOCnA Value is High)

The PSCBRAH bit enables or disables stopping of the GTCNT counter on the rising edge of the GTIOCnB pin input, when GTIOCnA input is 1.

PSCBFAL bit (GTIOCnB Pin Fall Input Source Counter Stop Enable when GTIOCnA Value is Low)

The PSCBFAL bit enables or disables stopping of the GTCNT counter on the falling edge of the GTIOCnB pin input, when GTIOCnA input is 0.

PSCBFAH bit (GTIOCnB Pin Fall Input Source Counter Stop Enable when GTIOCnA Value is High)

The PSCBFAH bit enables or disables stopping of the GTCNT counter on the falling edge of the GTIOCnB pin input, when GTIOCnA input is 1.

PSELCm bit (ELCm Event Source Counter Stop Enable) (m = A to H)

The PSELCm bit enables or disables stopping of the GTCNT counter at the ELC_GPTm event input.

CSTOP bit (Software Source Counter Stop Enable)

The CSTOP bit enables or disables stopping of the GTCNT counter by the GTSTP register.

5.7.2.2.7 General-Purpose Timer Clear Source Select Register (GPTm_n_GTCSR)

The GTCSR sets the source to clear the GTCNT counter.

Counter clearing can be executed whether the counter is running (GTCR.CST=1) or stopped (GTCR.CST=0).

Inputs from GTETRGn (n = A to H) pins are input to the GPT via the POEG. Set the polarity of these signals with the POEG.

Access Size : 32 bits

Address : <GPTm_base> + 0018h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CCLR	-	-	-	-	-	-	-	CSELCH	CSELCG	CSELCF	CSELCE	CSELCD	CSELCC	CSELCB	CSELCA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSCBFAH	CSCBFAL	CSCBRAH	CSCBRAL	CSCAFBH	CSCAFBL	CSCARBH	CSCARBL	CSGTRGDF	CSGTRGDR	CSGTRGCF	CSGTRGCR	CSGTRGBF	CSGTRGBR	CSGTRGAF	CSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CCLR	0h	RW	Software Source Counter Clear Enable 0b: Counter clear disabled by the GTCLR register 1b: Counter clear enabled by the GTCLR register
30 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23	CSELCH	0h	RW	ELC_GPTH Event Source Counter Clear Enable 0b: Counter clear disabled at the ELC_GPTH input 1b: Counter clear enabled at the ELC_GPTH input
22	CSELCG	0h	RW	ELC_GPTG Event Source Counter Clear Enable 0b: Counter clear disabled at the ELC_GPTG input 1b: Counter clear enabled at the ELC_GPTG input
21	CSELCF	0h	RW	ELC_GPTF Event Source Counter Clear Enable 0b: Counter clear disabled at the ELC_GPTF input 1b: Counter clear enabled at the ELC_GPTF input
20	CSELCE	0h	RW	ELC_GPTE Event Source Counter Clear Enable 0b: Counter clear disabled at the ELC_GPTE input 1b: Counter clear enabled at the ELC_GPTE input
19	CSELCD	0h	RW	ELC_GPTD Event Source Counter Clear Enable 0b: Counter clear disabled at the ELC_GPTD input 1b: Counter clear enabled at the ELC_GPTD input
18	CSELCC	0h	RW	ELC_GPTC Event Source Counter Clear Enable 0b: Counter clear disabled at the ELC_GPTC input 1b: Counter clear enabled at the ELC_GPTC input
17	CSELCB	0h	RW	ELC_GPTB Event Source Counter Clear Enable 0b: Counter clear disabled at the ELC_GPTB input 1b: Counter clear enabled at the ELC_GPTB input
16	CSELCA	0h	RW	ELC_GPTA Event Source Counter Clear Enable 0b: Counter clear disabled at the ELC_GPTA input 1b: Counter clear enabled at the ELC_GPTA input
15	CSCBFAH	0h	RW	GTIOCnB Pin Fall Input Source Counter Clear Enable when GTIOCnA Value is High 0b: Counter clear disabled on the falling edge of GTIOCnB input when GTIOCnA input is 1 1b: Counter clear enabled on the falling edge of GTIOCnB input when GTIOCnA input is 1
14	CSCBFAL	0h	RW	GTIOCnB Pin Fall Input Source Counter Clear Enable when GTIOCnA Value is Low 0b: Counter clear disabled on the falling edge of GTIOCnB input when GTIOCnA input is 0 1b: Counter clear enabled on the falling edge of GTIOCnB input when GTIOCnA input is 0

Bit	Bit Name	Initial Value	R/W	Description
13	CSCBRAH	0h	RW	GTIOCnB Pin Rise Input Source Counter Clear Enable when GTIOCnA Value is High 0b: Counter clear disabled on the rising edge of GTIOCnB input when GTIOCnA input is 1 1b: Counter clear enabled on the rising edge of GTIOCnB input when GTIOCnA input is 1
12	CSCBRAL	0h	RW	GTIOCnB Pin Rise Input Source Counter Clear Enable when GTIOCnA Value is Low 0b: Counter clear disabled on the rising edge of GTIOCnB input when GTIOCnA input is 0 1b: Counter clear enabled on the rising edge of GTIOCnB input when GTIOCnA input is 0
11	CSCAFBH	0h	RW	GTIOCnA Pin Fall Input Source Counter Clear Enable when GTIOCnB Value is High 0b: Counter clear disabled on the falling edge of GTIOCnA input when GTIOCnB input is 1 1b: Counter clear enabled on the falling edge of GTIOCnA input when GTIOCnB input is 1
10	CSCAFBL	0h	RW	GTIOCnA Pin Fall Input Source Counter Clear Enable when GTIOCnB Value is Low 0b: Counter clear disabled on the falling edge of GTIOCnA input when GTIOCnB input is 0 1b: Counter clear enabled on the falling edge of GTIOCnA input when GTIOCnB input is 0
9	CSCARBH	0h	RW	GTIOCnA Pin Rise Input Source Counter Clear Enable when GTIOCnB Value is High 0b: Counter clear disabled on the rising edge of GTIOCnA input when GTIOCnB input is 1 1b: Counter clear enabled on the rising edge of GTIOCnA input when GTIOCnB input is 1
8	CSCARBL	0h	RW	GTIOCnA Pin Rise Input Source Counter Clear Enable when GTIOCnB Value is Low 0b: Counter clear disabled on the rising edge of GTIOCnA input when GTIOCnB input is 0 1b: Counter clear enabled on the rising edge of GTIOCnA input when GTIOCnB input is 0
7	CSGTRGDF	0h	RW	GTETRGn Pin Fall Input Source Counter Clear Enable (n = D, H) 0b: Counter clear disabled on the falling edge of GTETRGn input 1b: Counter clear enabled on the falling edge of GTETRGn input
6	CSGTRGDR	0h	RW	GTETRGn Pin Rise Input Source Counter Clear Enable (n = D, H) 0b: Disable counter clear on the rising edge of GTETRGn input 1b: Enable counter clear on the rising edge of GTETRGn input
5	CSGTRGCF	0h	RW	GTETRGn Pin Fall Input Source Counter Clear Enable (n = C, G) 0b: Counter clear disabled on the falling edge of GTETRGn input 1b: Counter clear enabled on the falling edge of GTETRGn input
4	CSGTRGCR	0h	RW	GTETRGn Pin Rise Input Source Counter Clear Enable (n = C, G) 0b: Disable counter clear on the rising edge of GTETRGn input 1b: Enable counter clear on the rising edge of GTETRGn input
3	CSGTRGBF	0h	RW	GTETRGn Pin Fall Input Source Counter Clear Enable (n = B, F) 0b: Counter clear disabled on the falling edge of GTETRGn input 1b: Counter clear enabled on the falling edge of GTETRGn input
2	CSGTRGBR	0h	RW	GTETRGn Pin Rise Input Source Counter Clear Enable (n = B, F) 0b: Disable counter clear on the rising edge of GTETRGn input 1b: Enable counter clear on the rising edge of GTETRGn input
1	CSGTRGAF	0h	RW	GTETRGn Pin Fall Input Source Counter Clear Enable (n = A, E) 0b: Counter clear disabled on the falling edge of GTETRGn input 1b: Counter clear enabled on the falling edge of GTETRGn input
0	CSGTRGAR	0h	RW	GTETRGn Pin Rise Input Source Counter Clear Enable (n = A, E) 0b: Counter clear disabled on the rising edge of GTETRGn input 1b: Counter clear enabled on the rising edge of GTETRGn input

CSGTRGAR bit (GTETRGA/GTETRGE Pin Rising Input Source Counter Clear Enable)

The CSGTRGAR bit enables or disables clearing of the GTCNT counter on the rising edge of the GTETRGA/GTETRGE pin input.

CSGTRGAF bit (GTETRGA/GTETRGE Pin Fall Input Source Counter Clear Enable)

The CSGTRGAF bit enables or disables clearing of the GTCNT counter on the falling edge of the GTETRGA/GTETRGE pin input.

CSGTRGBR bit (GTETRGB/GTETRGF Pin Rise Input Source Counter Clear Enable)

The CSGTRGBR bit enables or disables clearing of the GTCNT counter on the rising edge of the GTETRGB/GTETRGF pin input.

CSGTRGBF bit (GTETRGB/GTETRGF Pin Fall Input Source Counter Clear Enable)

The CSGTRGBF bit enables or disables clearing of the GTCNT counter on the falling edge of the GTETRGB/GTETRGF pin input.

CSGTRGCR bit (GTETRGC/GTETRGG Pin Rise Input Source Counter Clear Enable)

The CSGTRGCR bit enables or disables clearing of the GTCNT counter on the rising edge of the GTETRGC/GTETRGG pin input.

CSGTRGCF bit (GTETRGC/GTETRGG Pin Fall Input Source Counter Clear Enable)

The CSGTRGCF bit enables or disables clearing of the GTCNT counter on the falling edge of the GTETRGC/GTETRGG pin input.

CSGTRGDR bit (GTETRGD/GTETRGH Pin Rise Input Source Counter Clear Enable)

The CSGTRGDR bit enables or disables clearing of the GTCNT counter on the rising edge of the GTETRGD/GTETRGH pin input.

CSGTRGDF bit (GTETRGD/GTETRGH Pin Fall Input Source Counter Clear Enable)

The CSGTRGDF bit enables or disables clearing of the GTCNT counter on the falling edge of the GTETRGD/GTETRGH pin input.

CSCARBL bit (GTIOCnA Pin Rise Input Source Counter Clear Enable when GTIOCnB Value is Low)

The CSCARBL bit enables or disables clearing of the GTCNT counter on the rising edge of the GTIOCnA pin input, when GTIOCnB input is 0.

CSCARBH bit (GTIOCnA Pin Rise Input Source Counter Clear Enable when GTIOCnB Value is High)

The CSCARBH bit enables or disables clearing of the GTCNT counter on the rising edge of the GTIOCnA pin input, when GTIOCnB input is 1.

CSCAFBL bit (GTIOCnA Pin Fall Input Source Counter Clear Enable when GTIOCnB Value is Low)

The CSCAFBL bit enables or disables clearing of the GTCNT counter on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 0.

CSCAFBH bit (GTIOCnA Pin Fall Input Source Counter Clear Enable when GTIOCnB Value is High)

The CSCAFBH bit enables or disables clearing of the GTCNT counter on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 1.

CSCBRAL bit (GTIOCnB Pin Rise Input Source Counter Clear Enable when GTIOCnA Value is Low)

The CSCBRAL bit enables or disables clearing of the GTCNT counter on the rising edge of the GTIOCnB pin input, when GTIOCnA input is 0.

CSCBRAH bit (GTIOCnB Pin Rise Input Source Counter Clear Enable when GTIOCnA Value is High)

The CSCBRAH bit enables or disables clearing of the GTCNT counter on the rising edge of the GTIOCnB pin input, when GTIOCnA input is 1.

CSCBFAL bit (GTIOCnB Pin Fall Input Source Counter Clear Enable when GTIOCnA Value is Low)

The CSCBFAL bit enables or disables clearing of the GTCNT counter on the falling edge of the GTIOCnB pin input, when GTIOCnA input is 0.

CSCBFAH bit (GTIOCnB Pin Fall Input Source Counter Clear Enable when GTIOCnA Value is High)

The CSCBFAH bit enables or disables clearing of the GTCNT counter on the falling edge of the GTIOCnB pin input, when GTIOCnA input is 1.

CSELCm bit (ELCm Event Source Counter Clear Enable) (m = A to H)

The CSELCm bit enables or disables clearing of the GTCNT counter at the ELC_GPTm event input.

CCLR bit (Software Source Counter Clear Enable)

The CCLR bit enables or disables clearing of the GTCNT counter by the GTCLR register.

5.7.2.2.8 General-Purpose Timer Up Count Source Select Register (GPTm_n_GTUPSR)

The GTUPSR sets the source to count up the GTCNT counter.

When at least one bit in the GTUPSR register is set to 1b, the GTCNT counter is counted up by the source that is set to 1 in this register. In this case, GTCR.TPCS has no effect.

Number of increments in counting is one even when multiple sources are generated simultaneously.

Inputs from GTETRGn (n = A to H) pins are input to the GPT via the POEG. Set the polarity of these signals with the POEG.

Access Size : 32 bits

Address : <GPTm_base> + 001Ch + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	USELCH	USELCG	USELCF	USELCE	USELCD	USELCC	USELCB	USELCA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	USCBFAH	USCBFAL	USCBRAH	USCBRAL	USCAFBH	USCAFBL	USCARBH	USCARBL	USGTRGDF	USGTRGDR	USGTRGCF	USGTRGCR	USGTRGBF	USGTRGBR	USGTRGAF	USGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23	USELCH	0h	RW	ELC_GPTH Event Source Counter Count Up Enable 0b: Counter count up disabled at the ELC_GPTH input 1b: Counter count up enabled at the ELC_GPTH input
22	USELCG	0h	RW	ELC_GPTG Event Source Counter Count Up Enable 0b: Counter count up disabled at the ELC_GPTG input 1b: Counter count up enabled at the ELC_GPTG input
21	USELCF	0h	RW	ELC_GPTF Event Source Counter Count Up Enable 0b: Counter count up disabled at the ELC_GPTF input 1b: Counter count up enabled at the ELC_GPTF input
20	USELCE	0h	RW	ELC_GPTE Event Source Counter Count Up Enable 0b: Counter count up disabled at the ELC_GPTE input 1b: Counter count up enabled at the ELC_GPTE input
19	USELCD	0h	RW	ELC_GPTD Event Source Counter Count Up Enable 0b: Counter count up disabled at the ELC_GPTD input 1b: Counter count up enabled at the ELC_GPTD input
18	USELCC	0h	RW	ELC_GPTC Event Source Counter Count Up Enable 0b: Counter count up disabled at the ELC_GPTC input 1b: Counter count up enabled at the ELC_GPTC input
17	USELCB	0h	RW	ELC_GPTB Event Source Counter Count Up Enable 0b: Counter count up disabled at the ELC_GPTB input 1b: Counter count up enabled at the ELC_GPTB input
16	USELCA	0h	RW	ELC_GPTA Event Source Counter Count Up Enable 0b: Counter count up disabled at the ELC_GPTA input 1b: Counter count up enabled at the ELC_GPTA input
15	USCBFAH	0h	RW	GTIOCnB Pin Fall Input Source Counter Count Up Enable when GTIOCnA Value is High 0b: Counter count up disabled on the falling edge of GTIOCnB input when GTIOCnA input is 1 1b: Counter count up enabled on the falling edge of GTIOCnB input when GTIOCnA input is 1
14	USCBFAL	0h	RW	GTIOCnB Pin Fall Input Source Counter Count Up Enable when GTIOCnA Value is Low 0b: Counter count up disabled on the falling edge of GTIOCnB input when GTIOCnA input is 0 1b: Counter count up enabled on the falling edge of GTIOCnB input when GTIOCnA input is 0

Bit	Bit Name	Initial Value	R/W	Description
13	USCBRAH	0h	RW	GTIOCnB Pin Rise Input Source Counter Count Up Enable when GTIOCnA Value is High 0b: Counter count up disabled on the rising edge of GTIOCnB input when GTIOCnA input is 1 1b: Counter count up enabled on the rising edge of GTIOCnB input when GTIOCnA input is 1
12	USCBRAL	0h	RW	GTIOCnB Pin Rise Input Source Counter Count Up Enable when GTIOCnA Value is Low 0b: Counter count up disabled on the rising edge of GTIOCnB input when GTIOCnA input is 0 1b: Counter count up enabled on the rising edge of GTIOCnB input when GTIOCnA input is 0
11	USCAFBH	0h	RW	GTIOCnA Pin Fall Input Source Counter Count Up Enable when GTIOCnB Value is High 0b: Counter count up disabled on the falling edge of GTIOCnA input when GTIOCnB input is 1 1b: Counter count up enabled on the falling edge of GTIOCnA input when GTIOCnB input is 1
10	USCAFBL	0h	RW	GTIOCnA Pin Fall Input Source Counter Count Up Enable when GTIOCnB Value is Low 0b: Counter count up disabled on the falling edge of GTIOCnA input when GTIOCnB input is 0 1b: Counter count up enabled on the falling edge of GTIOCnA input when GTIOCnB input is 0
9	USCARBH	0h	RW	GTIOCnA Pin Rise Input Source Counter Count Up Enable when GTIOCnB Value is High 0b: Counter count up disabled on the rising edge of GTIOCnA input when GTIOCnB input is 1 1b: Counter count up enabled on the rising edge of GTIOCnA input when GTIOCnB input is 1
8	USCARBL	0h	RW	GTIOCnA Pin Rise Input Source Counter Count Up Enable when GTIOCnB Value is Low 0b: Counter count up disabled on the rising edge of GTIOCnA input when GTIOCnB input is 0 1b: Counter count up enabled on the rising edge of GTIOCnA input when GTIOCnB input is 0
7	USGTRGDF	0h	RW	GTETRGn Pin Fall Input Source Counter Count Up Enable (n = D, H) 0b: Counter count up disabled on the falling edge of GTETRGn input 1b: Counter count up enabled on the falling edge of GTETRGn input
6	USGTRGDR	0h	RW	GTETRGn Pin Rise Input Source Counter Count Up Enable (n = D, H) 0b: Counter count up disabled on the rising edge of GTETRGn input 1b: Counter count up enabled on the rising edge of GTETRGn input
5	USGTRGCF	0h	RW	GTETRGn Pin Fall Input Source Counter Count Up Enable (n = C, G) 0b: Counter count up disabled on the falling edge of GTETRGn input 1b: Counter count up enabled on the falling edge of GTETRGn input
4	USGTRGCR	0h	RW	GTETRGn Pin Rise Input Source Counter Count Up Enable (n = C, G) 0b: Counter count up disabled on the rising edge of GTETRGn input 1b: Counter count up enabled on the rising edge of GTETRGn input
3	USGTRGBF	0h	RW	GTETRGn Pin Fall Input Source Counter Count Up Enable (n = B, F) 0b: Counter count up disabled on the falling edge of GTETRGn input 1b: Counter count up enabled on the falling edge of GTETRGn input
2	USGTRGBR	0h	RW	GTETRGn Pin Rise Input Source Counter Count Up Enable (n = B, F) 0b: Counter count up disabled on the rising edge of GTETRGn input 1b: Counter count up enabled on the rising edge of GTETRGn input
1	USGTRGAF	0h	RW	GTETRGn Pin Fall Input Source Counter Count Up Enable (n = A, E) 0b: Counter count up disabled on the falling edge of GTETRGn input 1b: Counter count up enabled on the falling edge of GTETRGn input
0	USGTRGAR	0h	RW	GTETRGn Pin Rise Input Source Counter Count Up Enable (n = A, E) 0b: Counter count up disabled on the rising edge of GTETRGn input 1b: Counter count up enabled on the rising edge of GTETRGn input

USGTRGAR bit (GTETRGA/GTETRGE Pin Rise Input Source Counter Count Up Enable)

The USGTRGAR bit enables or disables counting up of the GTCNT counter on the rising edge of the GTETRGA/GTETRGE pin input.

USGTRGAF bit (GTETRGA/GTETRGE Pin Fall Input Source Counter Count Up Enable)

The USGTRGAF bit enables or disables counting up of the GTCNT counter on the falling edge of the GTETRGA/GTETRGE pin input.

USGTRGBR bit (GTETRGB/GTETRGF Pin Rise Input Source Counter Count Up Enable)

The USGTRGBR bit enables or disables counting up of the GTCNT counter on the rising edge of the GTETRGB/GTETRGF pin input.

USGTRGBF bit (GTETRGB/GTETRGF Pin Fall Input Source Counter Count Up Enable)

The USGTRGBF bit enables or disables counting up of the GTCNT counter on the falling edge of the GTETRGB/GTETRGF pin input.

USGTRGCR bit (GTETRGC/GTETRGG Pin Rise Input Source Counter Count Up Enable)

The USGTRGCR bit enables or disables counting up of the GTCNT counter on the rising edge of the GTETRGC/GTETRGG pin input.

USGTRGCF bit (GTETRGC/GTETRGG Pin Fall Input Source Counter Count Up Enable)

The USGTRGCF bit enables or disables counting up of the GTCNT counter on the falling edge of the GTETRGC/GTETRGG pin input.

USGTRGDR bit (GTETRGD/GTETRGH Pin Rise Input Source Counter Count Up Enable)

The USGTRGDR bit enables or disables counting up of the GTCNT counter on the rising edge of the GTETRGD/GTETRGH pin input.

USGTRGDF bit (GTETRGD/GTETRGH Pin Fall Input Source Counter Count Up Enable)

The USGTRGDF bit enables or disables counting up of the GTCNT counter on the falling edge of the GTETRGD/GTETRGH pin input.

USCARBL bit (GTIOCnA Pin Rise Input Source Counter Count Up Enable when GTIOCnB Value is Low)

The USCARBL bit enables or disables GTCNT counter count up on the rising edge of GTIOCnA pin input, when GTIOCnB input is 0.

USCARBH bit (GTIOCnA Pin Rise Input Source Counter Count Up Enable when GTIOCnB Value is High)

The USCARBH bit enables or disables counting up of the GTCNT counter on the rising edge of the GTIOCnA pin input, when GTIOCnB input is 1.

USCAFBL bit (GTIOCnA Pin Fall Input Source Counter Count Up Enable when GTIOCnB Value is Low)

The USCAFBL bit enables or disables counting up of the GTCNT counter on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 0.

USCAFBH bit (GTIOCnA Pin Fall Input Source Counter Count Up Enable when GTIOCnB Value is High)

The USCAFBH bit enables or disables counting up of the GTCNT counter on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 1.

USCBRAL bit (GTIOCnB Pin Rise Input Source Counter Count Up Enable when GTIOCnA Value is Low)

The USCBRAL bit enables or disables counting up of the GTCNT counter on the rising edge of the GTIOCnB pin input, when GTIOCnA input is 0.

USCBRAH bit (GTIOCnB Pin Rise Input Source Counter Count Up Enable when GTIOCnA Value is High)

The USCBRAH bit enables or disables counting up of the GTCNT counter on the rising edge of the GTIOCnB pin input, when the GTIOCnA input is 1.

USCBFAL bit (GTIOCnB Pin Fall Input Source Counter Count Up Enable when GTIOCnA Value is Low)

The USCBFAL bit enables or disables counting up of the GTCNT counter on the falling edge of the GTIOCnB pin input, when the GTIOCnA input is 0.

USCBFAH bit (GTIOCnB Pin Fall Input Source Counter Count Up Enable when GTIOCnA Value is High)

The USCBFAH bit enables or disables counting up of the GTCNT counter on the falling edge of the GTIOCnB pin input, when the GTIOCnA input is 1.

USELCm bit (ELC_GPTm Event Source Counter Count Up Enable) (m = A to H)

The USELCm bit enables or disables counting up of the GTCNT counter at the ELC_GPTm event input.

5.7.2.2.9 General-Purpose Timer Down Count Source Select Register (GPTm_n_GTDNSR)

The GTDNSR sets the source to count down the GTCNT counter.

When at least one bit in the GTDNSR register is set to 1b, the GTCNT counter is counted down by the source that is set to 1b in this register. In this case, GTCR.TPCS has no effect.

Number of decrements in counting is one even when multiple sources are generated simultaneously.

Inputs from GTETRGn (n = A to H) pins are input to the GPT via the POEG. Set the polarity of these signals with the POEG.

Access Size : 32 bits

Address : <GPTm_base> + 0020h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	DSELCH	DSELCG	DSELCF	DSELCE	DSELCD	DSELCC	DSELCB	DSELCA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSCBFAH	DSCBFAL	DSCBRAH	DSCBRAL	DSCAFBH	DSCAFBL	DSCARBH	DSCARBL	DSGTRGDF	DSGTRGDR	DSGTRGCF	DSGTRGCR	DSGTRGBF	DSGTRGBR	DSGTRGAF	DSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23	DSELCH	0h	RW	ELC_GPTH Event Source Counter Count Down Enable 0b: Counter count down disabled at the ELC_GPTH input 1b: Counter count down enabled at the ELC_GPTH input
22	DSELCG	0h	RW	ELC_GPTG Event Source Counter Count Down Enable 0b: Counter count down disabled at the ELC_GPTG input 1b: Counter count down enabled at the ELC_GPTG input
21	DSELCF	0h	RW	ELC_GPTF Event Source Counter Count Down Enable 0b: Counter count down disabled at the ELC_GPTF input 1b: Counter count down enabled at the ELC_GPTF input
20	DSELCE	0h	RW	ELC_GPTE Event Source Counter Count Down Enable 0b: Counter count down disabled at the ELC_GPTE input 1b: Counter count down enabled at the ELC_GPTE input
19	DSELCD	0h	RW	ELC_GPTD Event Source Counter Count Down Enable 0b: Counter count down disabled at the ELC_GPTD input 1b: Counter count down enabled at the ELC_GPTD input
18	DSELCC	0h	RW	ELC_GPTC Event Source Counter Count Down Enable 0b: Counter count down disabled at the ELC_GPTC input 1b: Counter count down enabled at the ELC_GPTC input
17	DSELCB	0h	RW	ELC_GPTB Event Source Counter Count Down Enable 0b: Counter count down disabled at the ELC_GPTB input 1b: Counter count down enabled at the ELC_GPTB input
16	DSELCA	0h	RW	ELC_GPTA Event Source Counter Count Down Enable 0b: Counter count down disabled at the ELC_GPTA input 1b: Counter count down enabled at the ELC_GPTA input
15	DSCBFAH	0h	RW	GTIOCnB Pin Fall Input Source Counter Count Down Enable when GTIOCnA Value is High 0b: Counter count down disabled on the falling edge of GTIOCnB input when GTIOCnA input is 1 1b: Counter count down enabled on the falling edge of GTIOCnB input when GTIOCnA input is 1
14	DSCBFAL	0h	RW	GTIOCnB Pin Fall Input Source Counter Count Down Enable when GTIOCnA Value is Low 0b: Counter count down disabled on the falling edge of GTIOCnB input when GTIOCnA input is 0 1b: Counter count down enabled on the falling edge of GTIOCnB input when GTIOCnA input is 0

Bit	Bit Name	Initial Value	R/W	Description
13	DSCBRAH	0h	RW	GTIOCnB Pin Rise Input Source Counter Count Down Enable when GTIOCnA Value is High 0b: Counter count down disabled on the rising edge of GTIOCnB input when GTIOCnA input is 1 1b: Counter count down enabled on the rising edge of GTIOCnB input when GTIOCnA input is 1
12	DSCBRAL	0h	RW	GTIOCnB Pin Rise Input Source Counter Count Down Enable when GTIOCnA Value is Low 0b: Counter count down disabled on the rising edge of GTIOCnB input when GTIOCnA input is 0 1b: Counter count down enabled on the rising edge of GTIOCnB input when GTIOCnA input is 0
11	DSCAFBH	0h	RW	GTIOCnA Pin Fall Input Source Counter Count Down Enable when GTIOCnB Value is High 0b: Counter count down disabled on the falling edge of GTIOCnA input when GTIOCnB input is 1 1b: Counter count down enabled on the falling edge of GTIOCnA input when GTIOCnB input is 1
10	DSCAFBL	0h	RW	GTIOCnA Pin Fall Input Source Counter Count Down Enable when GTIOCnB Value is Low 0b: Counter count down disabled on the falling edge of GTIOCnA input when GTIOCnB input is 0 1b: Counter count down enabled on the falling edge of GTIOCnA input when GTIOCnB input is 0
9	DSCARBH	0h	RW	GTIOCnA Pin Rise Input Source Counter Count Down Enable when GTIOCnB Value is High 0b: Counter count down disabled on the rising edge of GTIOCnA input when GTIOCnB input is 1 1b: Counter count down enabled on the rising edge of GTIOCnA input when GTIOCnB input is 1
8	DSCARBL	0h	RW	GTIOCnA Pin Rise Input Source Counter Count Down Enable when GTIOCnB Value is Low 0b: Counter count down disabled on the rising edge of GTIOCnA input when GTIOCnB input is 0 1b: Counter count down enabled on the rising edge of GTIOCnA input when GTIOCnB input is 0
7	DSGTRGDF	0h	RW	GTETRGn Pin Fall Input Source Counter Count Down Enable (n = D, H) 0b: Counter count down disabled on the falling edge of GTETRGD input 1b: Counter count down enabled on the falling edge of GTETRGD input
6	DSGTRGDR	0h	RW	GTETRGn Pin Rise Input Source Counter Count Down Enable (n = D, H) 0b: Counter count down disabled on the rising edge of GTETRGn input 1b: Counter count down enabled on the rising edge of GTETRGn input
5	DSGTRGCF	0h	RW	GTETRGn Pin Fall Input Source Counter Count Down Enable (n = C, G) 0b: Counter count down disabled on the falling edge of GTETRGD input 1b: Counter count down enabled on the falling edge of GTETRGD input
4	DSGTRGCR	0h	RW	GTETRGn Pin Rise Input Source Counter Count Down Enable (n = C, G) 0b: Counter count down disabled on the rising edge of GTETRGn input 1b: Counter count down enabled on the rising edge of GTETRGn input
3	DSGTRGBF	0h	RW	GTETRGn Pin Fall Input Source Counter Count Down Enable (n = B, F) 0b: Counter count down disabled on the falling edge of GTETRGD input 1b: Counter count down enabled on the falling edge of GTETRGD input
2	DSGTRGBR	0h	RW	GTETRGn Pin Rise Input Source Counter Count Down Enable (n = B, F) 0b: Counter count down disabled on the rising edge of GTETRGn input 1b: Counter count down enabled on the rising edge of GTETRGn input
1	DSGTRGAF	0h	RW	GTETRGn Pin Fall Input Source Counter Count Down Enable (n = A, E) 0b: Counter count down disabled on the falling edge of GTETRGD input 1b: Counter count down enabled on the falling edge of GTETRGD input
0	DSGTRGAR	0h	RW	GTETRGn Pin Rise Input Source Counter Count Down Enable (n = A, E) 0b: Counter count down disabled on the rising edge of GTETRGn input 1b: Counter count down enabled on the rising edge of GTETRGn input

DSGTRGAR bit (GTETRGA/GTETRGE Pin Rise Input Source Counter Count Down Enable)

The DSGTRGAR bit enables or disables counting down of the GTCNT counter on the rising edge of the GTETRGA/GTETRGE pin input.

DSGTRGAF bit (GTETRGA/GTETRGE Pin Fall Input Source Counter Count Down Enable)

The DSGTRGAF bit enables or disables counting down of the GTCNT counter on the falling edge of the GTETRGA/GTETRGE pin input.

DSGTRGBR bit (GTETRGB/GTETRGF Pin Rise Input Source Counter Count Down Enable)

The DSGTRGBR bit enables or disables counting down of the GTCNT counter on the rising edge of the GTETRGB/GTETRGF pin input.

DSGTRGBF bit (GTETRGB/GTETRGF Pin Fall Input Source Counter Count Down Enable)

The DSGTRGBF bit enables or disables counting down of the GTCNT counter on the falling edge of the GTETRGB/GTETRGF pin input.

DSGTRGCR bit (GTETRGC/GTETRGG Pin Rise Input Source Counter Count Down Enable)

The DSGTRGCR bit enables or disables counting down of the GTCNT counter on the rising edge of the GTETRGC/GTETRGG pin input.

DSGTRGCF bit (GTETRGC/GTETRGG Pin Fall Input Source Counter Count Down Enable)

The DSGTRGCF bit enables or disables counting down of the GTCNT counter on the falling edge of the GTETRGC/GTETRGG pin input.

DSGTRGDR bit (GTETRGD/GTETRGH Pin Rise Input Source Counter Count Down Enable)

The DSGTRGDR bit enables or disables counting down of the GTCNT counter on the rising edge of the GTETRGD/GTETRGH pin input.

DSGTRGDF bit (GTETRGD/GTETRGH Pin Fall Input Source Counter Count Down Enable)

The DSGTRGDF bit enables or disables counting down of the GTCNT counter on the falling edge of the GTETRGD/GTETRGH pin input.

DSCARBL bit (GTIOCnA Pin Rise Input Source Counter Count Down Enable when GTIOCnB Value is Low)

The DSCARBL bit enables or disables counting down of the GTCNT counter on the rising edge of the GTIOCnA pin input, when the GTIOCnB input is 0.

DSCARBH bit (GTIOCnA Pin Rise Input Source Counter Count Down Enable when GTIOCnB Value is High)

The DSCARBH bit enables or disables counting down of the GTCNT counter on the rising edge of the GTIOCnA pin input, when GTIOCnB input is 1.

DSCAFBL bit (GTIOCnA Pin Fall Input Source Counter Count Down Enable when GTIOCnB Value is Low)

The DSCAFBL bit enables or disables counting down of the GTCNT counter on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 0.

DSCAFBH bit (GTIOCnA Pin Fall Input Source Counter Count Down Enable when GTIOCnB Value is High)

The DSCAFBH bit enables or disables counting down of the GTCNT counter on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 1.

DSCBRAL bit (GTIOCnB Pin Rise Input Source Counter Count Down Enable when GTIOCnA Value is Low)

The DSCBRAL bit enables or disables counting down of the GTCNT counter on the rising edge of the GTIOCnB pin input, when GTIOCnA input is 0.

DSCBRAH bit (GTIOCnB Pin Rise Input Source Counter Count Down Enable when GTIOCnA Value is High)

The DSCBRAH bit enables or disables counting down of the GTCNT counter on the rising edge of GTIOCnB pin input, when GTIOCnA input is 1.

DSCBFAL bit (GTIOCnB Pin Fall Input Source Counter Count Down Enable when GTIOCnA Value is Low)

The DSCBFAL bit enables or disables counting down of the GTCNT counter on the falling edge of the GTIOCnB pin input, when GTIOCnA input is 0.

DSCBFAH bit (GTIOCnB Pin Fall Input Source Counter Count Down Enable when GTIOCnA Value is High)

The DSCBFAH bit enables or disables counting down of the GTCNT counter on the falling edge of the GTIOCnB pin input, when GTIOCnA input is 1.

DSELCm bit (ELC_GPTm Event Source Counter Count Down Enable) (m = A to H)

The DSELCm bit enables or disables counting down of the GTCNT counter at the ELC_GPTm event input.

5.7.2.2.10 General-Purpose Timer Input Capture Source Select Register A (GPTm_n_GTICASR)

The GTICASR sets the source of input capture for GTCCRA.

When at least one bit among bits in the GTICASR register is set to 1b, input capture operation making the GTCCRA register as an input capture register is performed.

Inputs from GTETRGn (n = A to H) pins are input to the GPT via the POEG. Set the polarity of these signals with the POEG.

Access Size : 32 bits

Address : <GPTm_base> + 0024h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	ASELCH	ASELCG	ASELCF	ASELCE	ASELCD	ASELCC	ASELCB	ASELCA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ASCBFAH	ASCBFAL	ASCBRAH	ASCBRAL	ASCAFBH	ASCAFBL	ASCARBH	ASCARBL	ASGTRGDF	ASGTRGDR	ASGTRGCF	ASGTRGCR	ASGTRGBF	ASGTRGBR	ASGTRGAF	ASGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23	ASELCH	0h	RW	ELC_GPTH Event Source GTCCRA Input Capture Enable 0b: GTCCRA input capture disabled at the ELC_GPTH input 1b: GTCCRA input capture enabled at the ELC_GPTH input
22	ASELCG	0h	RW	ELC_GPTG Event Source GTCCRA Input Capture Enable 0b: GTCCRA input capture disabled at the ELC_GPTG input 1b: GTCCRA input capture enabled at the ELC_GPTG input
21	ASELCF	0h	RW	ELC_GPTF Event Source GTCCRA Input Capture Enable 0b: GTCCRA input capture disabled at the ELC_GPTF input 1b: GTCCRA input capture enabled at the ELC_GPTF input
20	ASELCE	0h	RW	ELC_GPTE Event Source GTCCRA Input Capture Enable 0b: GTCCRA input capture disabled at the ELC_GPTE input 1b: GTCCRA input capture enabled at the ELC_GPTE input
19	ASELCD	0h	RW	ELC_GPTD Event Source GTCCRA Input Capture Enable 0b: GTCCRA input capture disabled at the ELC_GPTD input 1b: GTCCRA input capture enabled at the ELC_GPTD input
18	ASELCC	0h	RW	ELC_GPTC Event Source GTCCRA Input Capture Enable 0b: GTCCRA input capture disabled at the ELC_GPTC input 1b: GTCCRA input capture enabled at the ELC_GPTC input
17	ASELCB	0h	RW	ELC_GPTB Event Source GTCCRA Input Capture Enable 0b: GTCCRA input capture disabled at the ELC_GPTB input 1b: GTCCRA input capture enabled at the ELC_GPTB input
16	ASELCA	0h	RW	ELC_GPTA Event Source GTCCRA Input Capture Enable 0b: GTCCRA input capture disabled at the ELC_GPTA input 1b: GTCCRA input capture enabled at the ELC_GPTA input
15	ASCBFAH	0h	RW	GTIOCnB Pin Fall Input Source GTCCRA Input Capture Enable when GTIOCnA Value is High 0b: GTCCRA input capture disabled on the falling edge of GTIOCnB input when GTIOCnA input is 1 1b: GTCCRA input capture enabled on the falling edge of GTIOCnB input when GTIOCnA input is 1

Bit	Bit Name	Initial Value	R/W	Description
14	ASCBFAL	0h	RW	GTIOCnB Pin Fall Input Source GTCCRA Input Capture Enable when GTIOCnA Value is Low 0b: GTCCRA input capture disabled on the falling edge of GTIOCnB input when GTIOCnA input is 0 1b: GTCCRA input capture enabled on the falling edge of GTIOCnB input when GTIOCnA input is 0
13	ASCBRAH	0h	RW	GTIOCnB Pin Rise Input Source GTCCRA Input Capture Enable when GTIOCnA Value is High 0b: GTCCRA input capture disabled on the rising edge of GTIOCnB input when GTIOCnA input is 1 1b: GTCCRA input capture enabled on the rising edge of GTIOCnB input when GTIOCnA input is 1
12	ASCBRAL	0h	RW	GTIOCnB Pin Rise Input Source GTCCRA Input Capture Enable when GTIOCnA Value is Low 0b: GTCCRA input capture disabled on the rising edge of GTIOCnB input when GTIOCnA input is 0 1b: GTCCRA input capture enabled on the rising edge of GTIOCnB input when GTIOCnA input is 0
11	ASCAFBH	0h	RW	GTIOCnA Pin Fall Input Source GTCCRA Input Capture Enable when GTIOCnB Value is High 0b: GTCCRA input capture disabled on the falling edge of GTIOCnA input when GTIOCnB input is 1 1b: GTCCRA input capture enabled on the falling edge of GTIOCnA input when GTIOCnB input is 1
10	ASCAFBL	0h	RW	GTIOCnA Pin Fall Input Source GTCCRA Input Capture Enable when GTIOCnB Value is Low 0b: GTCCRA input capture disabled on the falling edge of GTIOCnA input when GTIOCnB input is 0 1b: GTCCRA input capture enabled on the falling edge of GTIOCnA input when GTIOCnB input is 0
9	ASCARBH	0h	RW	GTIOCnA Pin Rise Input Source GTCCRA Input Capture Enable when GTIOCnB Value is High 0b: GTCCRA input capture disabled on the rising edge of GTIOCnA input when GTIOCnB input is 1 1b: GTCCRA input capture enabled on the rising edge of GTIOCnA input when GTIOCnB input is 1
8	ASCARBL	0h	RW	GTIOCnA Pin Rise Input Source GTCCRA Input Capture Enable when GTIOCnB Value is Low 0b: GTCCRA input capture disabled on the rising edge of GTIOCnA input when GTIOCnB input is 0 1b: GTCCRA input capture enabled on the rising edge of GTIOCnA input when GTIOCnB input is 0
7	ASGTRGDF	0h	RW	GTETRGn Pin Fall Input Source GTCCRA Input Capture Enable (n = D, H) 0b: GTCCRA input capture disabled on the falling edge of GTETRGn input 1b: GTCCRA input capture enabled on the falling edge of GTETRGn input
6	ASGTRGDR	0h	RW	GTETRGn Pin Rise Input Source GTCCRA Input Capture Enable (n = D, H) 0b: GTCCRA input capture disabled on the rising edge of GTETRGn input 1b: GTCCRA input capture enabled on the rising edge of GTETRGn input
5	ASGTRGCF	0h	RW	GTETRGn Pin Fall Input Source GTCCRA Input Capture Enable (n = C, G) 0b: GTCCRA input capture disabled on the falling edge of GTETRGn input 1b: GTCCRA input capture enabled on the falling edge of GTETRGn input
4	ASGTRGCR	0h	RW	GTETRGn Pin Rise Input Source GTCCRA Input Capture Enable (n = C, G) 0b: GTCCRA input capture disabled on the rising edge of GTETRGn input 1b: GTCCRA input capture enabled on the rising edge of GTETRGn input
3	ASGTRGBF	0h	RW	GTETRGn Pin Fall Input Source GTCCRA Input Capture Enable (n = B, F) 0b: GTCCRA input capture disabled on the falling edge of GTETRGn input 1b: GTCCRA input capture enabled on the falling edge of GTETRGn input
2	ASGTRGBR	0h	RW	GTETRGn Pin Rise Input Source GTCCRA Input Capture Enable (n = B, F) 0b: GTCCRA input capture disabled on the rising edge of GTETRGn input 1b: GTCCRA input capture enabled on the rising edge of GTETRGn input
1	ASGTRGAF	0h	RW	GTETRGn Pin Fall Input Source GTCCRA Input Capture Enable (n = A, E) 0b: GTCCRA input capture disabled on the falling edge of GTETRGn input 1b: GTCCRA input capture enabled on the falling edge of GTETRGn input
0	ASGTRGAR	0h	RW	GTETRGn Pin Rise Input Source GTCCRA Input Capture Enable (n = A, E) 0b: GTCCRA input capture disabled on the rising edge of GTETRGn input 1b: GTCCRA input capture enabled on the rising edge of GTETRGn input

ASGTRGAR bit (GTETRGA/GTETRGE Pin Rise Input Source GTCCRA Input Capture Enable)

The ASGTRGAR bit enables or disables input capture for GTCCRA on the rising edge of the GTETRGA/GTETRGE pin input.

ASGTRGAF bit (GTETRGA/GTETRGE Pin Fall Input Source GTCCRA Input Capture Enable)

The ASGTRGAF bit enables or disables input capture for GTCCRA on the falling edge of the GTETRGA/GTETRGE pin input.

ASGTRGBR bit (GTETRGB/GTETRGF Pin Rise Input Source GTCCRA Input Capture Enable)

The ASGTRGBR bit enables or disables input capture for GTCCRA on the rising edge of the GTETRGB/GTETRGF pin input.

ASGTRGBF bit (GTETRGB/GTETRGF Pin Fall Input Source GTCCRA Input Capture Enable)

The ASGTRGBF bit enables or disables input capture for GTCCRA on the falling edge of the GTETRGB/GTETRGF pin input.

ASGTRGCR bit (GTETRGC/GTETRGG Pin Rise Input Source GTCCRA Input Capture Enable)

The ASGTRGCR bit enables or disables input capture for GTCCRA on the rising edge of the GTETRGC/GTETRGG pin input.

ASGTRGCF bit (GTETRGC/GTETRGG Pin Fall Input Source GTCCRA Input Capture Enable)

The ASGTRGCF bit enables or disables input capture for GTCCRA on the falling edge of the GTETRGC/GTETRGG pin input.

ASGTRGDR bit (GTETRGD/GTETRGH Pin Rise Input Source GTCCRA Input Capture Enable)

The ASGTRGDR bit enables or disables input capture for GTCCRA on the rising edge of the GTETRGD/GTETRGH pin input.

ASGTRGDF bit (GTETRGD/GTETRGH Pin Fall Input Source GTCCRA Input Capture Enable)

The ASGTRGDF bit enables or disables input capture for GTCCRA on the falling edge of the GTETRGD/GTETRGH pin input.

ASCARBL bit (GTIOCnA Pin Rise Input Source GTCCRA Input Capture Enable when GTIOCnB Value is Low)

The ASCARBL bit enables or disables input capture for GTCCRA on the rising edge of the GTIOCnA pin input, when GTIOCnB input is 0b.

ASCARBH bit (GTIOCnA Pin Rise Input Source GTCCRA Input Capture Enable when GTIOCnB Value is High)

The ASCARBH bit enables or disables input capture for GTCCRA on the rising edge of the GTIOCnA pin input, when GTIOCnB input is 1b.

ASCAFBL bit (GTIOCnA Pin Fall Input Source GTCCRA Input Capture Enable when GTIOCnB Value is Low)

The ASCAFBL bit enables or disables input capture for GTCCRA on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 0b.

ASCAFBH bit (GTIOCnA Pin Fall Input Source GTCCRA Input Capture Enable when GTIOCnB Value is High)

The ASCAFBH bit enables or disables input capture for GTCCRA on the falling edge of the GTIOCnA pin input, when the GTIOCnB input is 1b.

ASCBRAL bit (GTIOCnB Pin Rise Input Source GTCCRA Input Capture Enable when GTIOCnA Value is Low)

The ASCBRAL bit enables or disables input capture for GTCCRA on the rising edge of the GTIOCnB pin input, when the GTIOCnA input is 0b.

ASCBRAH bit (GTIOCnB Pin Rise Input Source GTCCRA Input Capture Enable when GTIOCnA Value is High)

The ASCBRAH bit enables or disables input capture for GTCCRA on the rising edge of the GTIOCnB pin input, when GTIOCnA input is 1b.

ASCBFAL bit (GTIOCnB Pin Fall Input Source GTCCRA Input Capture Enable when GTIOCnA Value is Low)

The ASCBFAL bit enables or disables input capture for GTCCRA on the falling edge of the GTIOCnB pin input, when GTIOCnA input is 0b.

ASCBFAH bit (GTIOCnB Pin Fall Input Source GTCCRA Input Capture Enable when GTIOCnA Value is High)

The ASCBFAH bit enables or disables input capture for GTCCRA on the falling edge of the GTIOCnB pin input, when GTIOCnA input is 1b.

ASELCm bit (ELC_GPTm Event Source Counter GTCCRA Input Capture Enable) (m = A to H)

The ASELCm bit enables or disables input capture for GTCCRA at the ELC_GPTm event input.

5.7.2.2.11 General-Purpose Timer Input Capture Source Select Register B (GPTm_n_GTICBSR)

The GTICBSR sets the source of input capture for GTCCRB.

When at least one bit among bits in the GTICBSR register is set to 1b, input capture operation making the GTCCRB register as an input capture register is performed.

Inputs from GTETRGn (n = A to H) pins are input to the GPT via the POEG. Set the polarity of these signals with the POEG.

Access Size : 32 bits

Address : <GPTm_base> + 0028h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	BSELCH	BSELCG	BSELCF	BSELCE	BSELCD	BSELCC	BSELCB	BSELCA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSCBFAH	BSCBFAL	BSCBRAH	BSCBRAL	BSCAFBH	BSCAFBL	BSCARBH	BSCARBL	BSGTRGDF	BSGTRGDR	BSGTRGCF	BSGTRGCR	BSGTRGBF	BSGTRGBR	BSGTRGAF	BSGTRGAR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23	BSELCH	0h	RW	ELC_GPTH Event Source GTCCRB Input Capture Enable 0b: GTCCRB input capture disabled at the ELC_GPTH input 1b: GTCCRB input capture enabled at the ELC_GPTH input
22	BSELCG	0h	RW	ELC_GPTG Event Source GTCCRB Input Capture Enable 0b: GTCCRB input capture disabled at the ELC_GPTG input 1b: GTCCRB input capture enabled at the ELC_GPTG input
21	BSELCF	0h	RW	ELC_GPTF Event Source GTCCRB Input Capture Enable 0b: GTCCRB input capture disabled at the ELC_GPTF input 1b: GTCCRB input capture enabled at the ELC_GPTF input
20	BSELCE	0h	RW	ELC_GPTE Event Source GTCCRB Input Capture Enable 0b: GTCCRB input capture disabled at the ELC_GPTE input 1b: GTCCRB input capture enabled at the ELC_GPTE input
19	BSELCD	0h	RW	ELC_GPTD Event Source GTCCRB Input Capture Enable 0b: GTCCRB input capture disabled at the ELC_GPTD input 1b: GTCCRB input capture enabled at the ELC_GPTD input
18	BSELCC	0h	RW	ELC_GPTC Event Source GTCCRB Input Capture Enable 0b: GTCCRB input capture disabled at the ELC_GPTC input 1b: GTCCRB input capture enabled at the ELC_GPTC input
17	BSELCB	0h	RW	ELC_GPTB Event Source GTCCRB Input Capture Enable 0b: GTCCRB input capture disabled at the ELC_GPTB input 1b: GTCCRB input capture enabled at the ELC_GPTB input
16	BSELCA	0h	RW	ELC_GPTA Event Source GTCCRB Input Capture Enable 0b: GTCCRB input capture disabled at the ELC_GPTA input 1b: GTCCRB input capture enabled at the ELC_GPTA input
15	BSCBFAH	0h	RW	GTIOCnB Pin Fall Input Source GTCCRB Input Capture Enable when GTIOCnA Value is High 0b: GTCCRB input capture disabled on the falling edge of GTIOCnB input when GTIOCnA input is 1 1b: GTCCRB input capture enabled on the falling edge of GTIOCnB input when GTIOCnA input is 1

Bit	Bit Name	Initial Value	R/W	Description
14	BSCBFAL	0h	RW	GTIOCnB Pin Fall Input Source GTCCRB Input Capture Enable when GTIOCnA Value is Low 0b: GTCCRB input capture disabled on the falling edge of GTIOCnB input when GTIOCnA input is 0 1b: GTCCRB input capture enabled on the falling edge of GTIOCnB input when GTIOCnA input is 0
13	BSCBRAH	0h	RW	GTIOCnB Pin Rise Input Source GTCCRB Input Capture Enable when GTIOCnA Value is High 0b: GTCCRB input capture disabled on the rising edge of GTIOCnB input when GTIOCnA input is 1 1b: GTCCRB input capture enabled on the rising edge of GTIOCnB input when GTIOCnA input is 1
12	BSCBRAL	0h	RW	GTIOCnB Pin Rise Input Source GTCCRB Input Capture Enable when GTIOCnA Value is Low 0b: GTCCRB input capture disabled on the rising edge of GTIOCnB input when GTIOCnA input is 0 1b: GTCCRB input capture enabled on the rising edge of GTIOCnB input when GTIOCnA input is 0
11	BSCAFBH	0h	RW	GTIOCnA Pin Fall Input Source GTCCRB Input Capture Enable when GTIOCnB Value is High 0b: GTCCRB input capture disabled on the falling edge of GTIOCnA input when GTIOCnB input is 1 1b: GTCCRB input capture enabled on the falling edge of GTIOCnA input when GTIOCnB input is 1
10	BSCAFBL	0h	RW	GTIOCnA Pin Fall Input Source GTCCRB Input Capture Enable when GTIOCnB Value is Low 0b: GTCCRB input capture disabled on the falling edge of GTIOCnA input when GTIOCnB input is 0 1b: GTCCRB input capture enabled on the falling edge of GTIOCnA input when GTIOCnB input is 0
9	BSCARBH	0h	RW	GTIOCnA Pin Rise Input Source GTCCRB Input Capture Enable when GTIOCnB Value is High 0b: GTCCRB input capture disabled on the rising edge of GTIOCnA input when GTIOCnB input is 1 1b: GTCCRB input capture enabled on the rising edge of GTIOCnA input when GTIOCnB input is 1
8	BSCARBL	0h	RW	GTIOCnA Pin Rise Input Source GTCCRB Input Capture Enable when GTIOCnB Value is Low 0b: GTCCRB input capture disabled on the rising edge of GTIOCnA input when GTIOCnB input is 0 1b: GTCCRB input capture enabled on the rising edge of GTIOCnA input when GTIOCnB input is 0
7	BSGTRGDF	0h	RW	GTETRGn Pin Fall Input Source GTCCRB Input Capture Enable (n = D, H) 0b: GTCCRB input capture disabled on the falling edge of GTETRGn input 1b: GTCCRB input capture enabled on the falling edge of GTETRGn input
6	BSGTRGDR	0h	RW	GTETRGn Pin Rise Input Source GTCCRB Input Capture Enable (n = D, H) 0b: GTCCRB input capture disabled on the rising edge of GTETRGn input 1b: GTCCRB input capture enabled on the rising edge of GTETRGn input
5	BSGTRGCF	0h	RW	GTETRGn Pin Fall Input Source GTCCRB Input Capture Enable (n = C, G) 0b: GTCCRB input capture disabled on the falling edge of GTETRGn input 1b: GTCCRB input capture enabled on the falling edge of GTETRGn input
4	BSGTRGCR	0h	RW	GTETRGn Pin Rise Input Source GTCCRB Input Capture Enable (n = C, G) 0b: GTCCRB input capture disabled on the rising edge of GTETRGn input 1b: GTCCRB input capture enabled on the rising edge of GTETRGn input
3	BSGTRGBF	0h	RW	GTETRGn Pin Fall Input Source GTCCRB Input Capture Enable (n = B, F) 0b: GTCCRB input capture disabled on the falling edge of GTETRGn input 1b: GTCCRB input capture enabled on the falling edge of GTETRGn input
2	BSGTRGBR	0h	RW	GTETRGn Pin Rise Input Source GTCCRB Input Capture Enable (n = B, F) 0b: GTCCRB input capture disabled on the rising edge of GTETRGn input 1b: GTCCRB input capture enabled on the rising edge of GTETRGn input
1	BSGTRGAF	0h	RW	GTETRGn Pin Fall Input Source GTCCRB Input Capture Enable (n = A, E) 0b: GTCCRB input capture disabled on the falling edge of GTETRGn input 1b: GTCCRB input capture enabled on the falling edge of GTETRGn input
0	BSGTRGAR	0h	RW	GTETRGn Pin Rise Input Source GTCCRB Input Capture Enable (n = A, E) 0b: GTCCRB input capture disabled on the rising edge of GTETRGn input 1b: GTCCRB input capture enabled on the rising edge of GTETRGn input

BSGTRGAR bit (GTETRGA/GTETRGE Pin Rise Input Source GTCCRB Input Capture Enable)

The BSGTRGAR bit enables or disables input capture for GTCCRB on the rising edge of the GTETRGA/GTETRGE pin input.

BSGTRGAF bit (GTETRGA/GTETRGE Pin Fall Input Source GTCCRB Input Capture Enable)

The BSGTRGAF bit enables or disables input capture for GTCCRB on the falling edge of the GTETRGA/GTETRGE pin input.

BSGTRGBR bit (GTETRGB/GTETRGF Pin Rise Input Source GTCCRB Input Capture Enable)

The BSGTRGBR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGB/GTETRGF pin input.

BSGTRGBF bit (GTETRGB/GTETRGF Pin Fall Input Source GTCCRB Input Capture Enable)

The BSGTRGBF bit enables or disables input capture for GTCCRB on the falling edge of the GTETRGB/GTETRGF pin input.

BSGTRGCR bit (GTETRGC/GTETRGG Pin Rise Input Source GTCCRB Input Capture Enable)

The BSGTRGCR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGC/GTETRGG pin input.

BSGTRGCF bit (GTETRGC/GTETRGG Pin Fall Input Source GTCCRB Input Capture Enable)

The BSGTRGCF bit enables or disables input capture for GTCCRB on the falling edge of the GTETRGC/GTETRGG pin input.

BSGTRGDR bit (GTETRGD/GTETRGH Pin Rise Input Source GTCCRB Input Capture Enable)

The BSGTRGDR bit enables or disables input capture for GTCCRB on the rising edge of GTETRGD/GTETRGH pin input.

BSGTRGDF bit (GTETRGD/GTETRGH Pin Fall Input Source GTCCRB Input Capture Enable)

The BSGTRGDF bit enables or disables input capture for GTCCRB on the falling edge of the GTETRGD/GTETRGH pin input.

BSCARBL bit (GTIOCnA Pin Rise Input Source GTCCRB Input Capture Enable when GTIOCnB Value is Low)

The BSCARBL bit enables or disables input capture for GTCCRB on the rising edge of the GTIOCnA pin input, when the GTIOCnB input is 0b.

BSCARBH bit (GTIOCnA Pin Rise Input Source GTCCRB Input Capture Enable when GTIOCnB Value is High)

The BSCARBH bit enables or disables input capture for GTCCRB on the rising edge of the GTIOCnA pin input, when GTIOCnB input is 1b.

BSCAFBL bit (GTIOCnA Pin Fall Input Source GTCCRB Input Capture Enable when GTIOCnB Value is Low)

The BSCAFBL bit enables or disables input capture for GTCCRB on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 0b.

BSCAFBH bit (GTIOCnA Pin Fall Input Source GTCCRB Input Capture Enable when GTIOCnB Value is High)

The BSCAFBH bit enables or disables input capture for GTCCRB on the falling edge of the GTIOCnA pin input, when GTIOCnB input is 1b.

BSCBRAL bit (GTIOCnB Pin Rise Input Source GTCCRB Input Capture Enable when GTIOCnA Value is Low)

The BSCBRAL bit enables or disables input capture for GTCCRB on the rising edge of the GTIOCnB pin input, when GTIOCnA input is 0b.

BSCBRAH bit (GTIOCnB Pin Rise Input Source GTCCRB Input Capture Enable when GTIOCnA Value is High)

The BSCBRAH bit enables or disables input capture for GTCCRB on the rising edge of the GTIOCnB pin input, when GTIOCnA input is 1b.

BSCBFAL bit (GTIOCnB Pin Fall Input Source GTCCRB Input Capture Enable when GTIOCnA Value is Low)

The BSCBFAL bit enables or disables input capture for GTCCRB on the falling edge of the GTIOCnB pin input, when GTIOCnA input is 0b.

BSCBFAH bit (GTIOCnB Pin Fall Input Source GTCCRB Input Capture Enable when GTIOCnA Value is High)

The BSCBFAH bit enables or disables input capture for GTCCRB on the falling edge of the GTIOCnB pin input, when GTIOCnA input is 1b.

BSELCm bit (ELC_GPTm Event Source Counter GTCCRB Input Capture Enable) (m = A to H)

The BSELCm bit enables or disables input capture for GTCCRB at the ELC_GPTm event input.

5.7.2.2.12 General-Purpose Timer Control Register (GPTm_n_GTCR)

The GTCR controls GTCNT.

Access Size : 32 bits

Address : <GPTm_base> + 002Ch + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	TPCS[3:0]				-	-	-	-	MD[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
26 to 23	TPCS[3:0]	0h	RW	Timer Prescaler Select 0000b: PCLKD/1 0001b: PCLKD/2 0010b: PCLKD/4 0011b: PCLKD/8 0100b: PCLKD/16 0101b: PCLKD/32 0110b: PCLKD/64 0111b: Setting prohibited 1000b: PCLKD/256 1001b: Setting prohibited 1010b: PCLKD/1024 1011b: Setting prohibited 1100b: GTETRGA/GTETRGE (via the POEG) 1101b: GTETRGB/GTETRGF (via the POEG) 1110b: GTETRGC/GTETRGG (via the POEG) 1111b: GTETRGD/GTETRGH (via the POEG)
22 to 19	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18 to 16	MD[2:0]	0h	RW	Mode Select 000b: Saw-wave PWM mode (single buffer or double buffer enabled) 001b: Saw-wave one-shot pulse mode (fixed buffer operation) 010b: Setting prohibited 011b: Setting prohibited 100b: Triangle-wave PWM mode 1 (32-bit transfer at trough) (single buffer or double buffer enabled) 101b: Triangle-wave PWM mode 2 (32-bit transfer at crest and trough) (single buffer or double buffer enabled) 110b: Triangle-wave PWM mode 3 (64-bit transfer at trough) (fixed buffer operation) 111b: Setting prohibited
15 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	CST	0h	RW	Count Start 0b: Count operation is stopped 1b: Count operation is performed

CST bit (Count Start)

The CST bit controls starting and stopping of the GTCNT counter.

[Setting conditions]

- The GTSTR value where the channel number corresponding to the bit number is set to 1b with the GTSSR.CSTRT bit at 1b.
- Occurrence of the ELC event input, the external trigger, or the GTIOCnA/GTIOCnB input that are enabled by GTSSR for the starting counter source (n = 0 to 15)
- 1b is written by software directly.

[Clearing conditions]

- The GTSTP value where the channel number corresponding to the bit number is set to 1b with the GTPSR.CSTOP bit at 1b.
- Occurrence of the ELC event input, the external trigger, or the GTIOCnA/GTIOCnB input enabled by GTSSR as the counter stop source (n = 0 to 15)
- 0b is written by software directly.

MD[2:0] bits (Mode Select)

The MD[2:0] bits select the GPT operating mode. The MD[2:0] bits must be set while the GTCNT operation is stopped.

TPCS[3:0] bits (Timer Prescaler Select)

The TPCS[3:0] bits select the clock for GTCNT. A clock prescaler can be selected independently for each channel. The TPCS[3:0] bits must be set while the GTCNT operation is stopped.

5.7.2.2.13 General-Purpose Timer Count Direction and Duty Setting Register (GPTm_n_GTUDDTYC)

The GTUDDTYC sets the direction in which the GTCNT counts (up-counting or down-counting), and sets the duty cycle of the GTIOCnA/GTIOCnB pin output.

Access Size : 32 bits

Address : <GPTm_base> + 0030h + n x 0100h

Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	OBDTYR	OBDTYF	OBDTY[1:0]		-	-	-	-	OADTYR	OADTYF	OADTY[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	UDF	UD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
27	OBDTYR	0h	RW	GTIOCnB Output Value Selection after Release of 0%/100% Duty Setting 0b: The function selected by the GTIOB[3:2] bits is applied to the output value set for a duty cycle of 0% or 100% after release from the 0% or 100% duty-cycle setting. 1b: The function selected by the GTIOB[3:2] bits is applied to the masked compare match output value after release from the 0% or 100% duty-cycle setting.
26	OBDTYF	0h	RW	Forcible GTIOCnB Output Duty Setting 0b: Not forcibly set 1b: Forcibly set
25, 24	OBDTY[1:0]	0h	RW	GTIOCnB Output Duty Setting 00b: The duty cycle of the GTIOCnB pin depends on the compare match 01b: The duty cycle of the GTIOCnB pin depends on the compare match 10b: The duty cycle of the GTIOCnB pin is 0% 11b: The duty cycle of the GTIOCnB pin is 100%
23 to 20	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
19	OADTYR	0h	RW	GTIOCnA Output Value Selecting after Releasing 0%/100% Duty Setting 0b: The function selected by the GTIOA[3:2] bits is applied to the output value when the duty cycle is set after release from the 0 or 100% duty-cycle setting. 1b: The function selected by the GTIOA[3:2] bits is applied to the compare match output value which is masked after release from the 0 or 100% duty-cycle setting.
18	OADTYF	0h	RW	Forcible GTIOCnA Output Duty Setting 0b: Not forcibly set 1b: Forcibly set
17, 16	OADTY[1:0]	0h	RW	GTIOCnA Output Duty Setting 00b: The duty cycle of the GTIOCnA pin depends on the compare match 01b: The duty cycle of the GTIOCnA pin depends on the compare match 10b: The duty cycle of the GTIOCnA pin is 0% 11b: The duty cycle of the GTIOCnA pin is 100%
15 to 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	UDF	0h	RW	Forcible Count Direction Setting 0b: Not forcibly set 1b: Forcibly set
0	UD	1h	RW	Count Direction Setting 0b: GTCNT counts down 1b: GTCNT counts up

The setting is invalid during the event count operation.

[Count Direction]

- In saw-wave mode

  When the UD value is set to 0b during up-counting, the count direction changes on overflow (the timing synchronous with count clock after the GTCNT value becomes the GTPR value). When the UD value is set to 1b during down-counting, the count direction changes on underflow (the timing synchronous with count clock after the GTCNT value becomes 0b).

  When the UD value changes from 1b to 0b with the UDF bit being 0b and while counting stops, the counter starts up-counting and the count direction changes on overflow (the timing synchronous with count clock after the GTCNT value becomes the GTPR value). When the UD value changes from 0b to 1b with the UDF bit being 0b and while counting stops, the counter starts down-counting and the count direction changes on underflow (the timing synchronous with count clock after the GTCNT value becomes 0b).

  When the UDF bit is set to 1b while counting stops, the UD bit value is reflected in the count direction when counting starts.

- In triangle-wave mode

  When the UD value changes during counting, the count direction does not change. When the UD value changes while the UDF bit is 0b and counting stops, the change is not reflected in the count direction when counting starts.

  When the UDF bit is set to 1b while counting is stopped, the UD value is reflected in the count direction when counting starts.

UD bit (Count Direction Setting)

The UD bit sets the count direction (up-counting or down-counting) for GTCNT.

UDF bit (Forcible Count Direction Setting)

The UDF bit forcibly sets the count direction when GTCNT starts operation as the UD value. Only 0b should be written to this bit during counter operation. When 1b is written to this bit while counting stops, return this bit to 0 before counting starts.

[Output duty cycle]

- In saw-wave mode

  When the OADTY/OBDTY value changes during up-counting, the duty cycle is reflected on overflow (GTCNT = GTPR). When the OADTY/OBDTY value is changed during down-counting, the duty cycle is reflected on underflow (GTCNT = 0b).

  When the OADTY/OBDTY value is changed with the OADTYF/OBDTYF bit being 0b and while counting stops the output duty cycle is not reflected at the starting counter operation. When the count direction is up, the output duty cycle is reflected on overflow (GTCNT = GTPR). When the count direction is down, the output duty cycle is reflected on underflow (GTCNT = 0b).

  When the OADTY/OBDTY value is changed with the OADTYF/OBDTYF bit being 1b and while counting stops, the output duty cycle is reflected at the start of counter operation.

- In triangle-wave mode

  When the OADTY/OBDTY value changes during counting, the duty cycle is reflected on underflow.

  When the OADTY/OBDTY value is changed with the OADTYF/OBDTYF bit being 0b and while counting stops, the output duty cycle is not reflected at the start of counter operation. The output duty cycle is reflected on underflow.

When the OADTY/OBDTY value is changed with the OADTYF/OBDTYF bit being 1b and while counting stops, the output duty cycle is reflected at the start of counter operation.

In both saw-wave mode and triangle-wave mode, when the OADTYF/OBDTYF bit is set back to 0b and the OADTY[1:0]/ OBDTY[1:0] bits are set after setting the OADTYF/OBDTYF bit to 1b and setting the OADTY[1:0]/OBDTY[1:0] bits for the duty cycle of the first cycle while counting is stopped, these duty-cycle settings during stopping count operation are reflected in the first cycle and the second cycle after starting count operation.

OmDTY[1:0] bits (GTIOCnm Output Duty Setting) (m = A, B)

The OmDTY[1:0] bits set the output duty cycle (0%, 100% or compare match control) of the GTIOCm pin.

OmDTYF bit (Forcible GTIOCnm Output Duty Setting) (m = A, B)

The OmDTYF bit forcibly sets the output duty cycle to the OmDTY setting. Set this bit to 0b during counter operation.

OmDTYR bit (GTIOCnm Output Value Selection after Release from 0%/100% Duty Setting) (m = A, B)

The OmDTYR bit selects the target value for output retained or toggled at the cycle end, when the control changes from 0% or 100% duty-cycle setting to compare match for the GTIOCm pin and GTIOR.GTIOm[3:2] bits are set to 00b (output retained at the cycle end) or the GTIOR.GTIOm[3:2] bits are set to 11b (output toggled at the cycle end).

The GPT internally continues to perform compare match operation during duty-cycle 0% or 100% operation. When the OmDTYR bit is 1b, the value after the period has elapsed from this compare match operation is a target for the GTIOm[3:2] bits.

5.7.2.2.14 General-Purpose Timer I/O Control Register (GPTm_n_GTIOR)

The GTIOR sets the functions of the GTIOCnA and GTIOCnB pins.

Access Size : 32 bits

Address : <GPTm_base> + 0034h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NFCSB[1:0]		NFBEN	-	-	OBDF[1:0]		OBE	OBHLD	OBDFLT	-	GTIOB[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NFCSA[1:0]		NFAEN	-	-	OADF[1:0]		OAE	OAHLD	OADFLT	-	GTIOA[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 30	NFCSB[1:0]	0h	RW	Noise Filter B Sampling Clock Select 00b: PCLKD/1 01b: PCLKD/4 10b: PCLKD/16 11b: PCLKD/64
29	NFBEN	0h	RW	Noise Filter B Enable 0b: The noise filter for the GTIOCnB pin is disabled 1b: The noise filter for the GTIOCnB pin is enabled
28, 27	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
26, 25	OBDF[1:0]	0h	RW	GTIOCnB Pin Disable Value Setting 00b: None of the following sources is specified 01b: GTIOCnB pin is set to Hi-Z in response to control for output negation 10b: GTIOCnB pin is set to 0 in response to control for output negation 11b: GTIOCnB pin is set to 1 in response to control for output negation
24	OBE	0h	RW	GTIOCnB Pin Output Enable 0b: Output is disabled 1b: Output is enabled
23	OBHLD	0h	RW	GTIOCnB Pin Output Setting at Count Start/Stop 0b: The GTIOCnB pin output level at the start/stop of counting depends on the register setting 1b: The GTIOCnB pin output level is retained at the start/stop of counting
22	OBDFLT	0h	RW	GTIOCnB Pin Output Value Setting at Count Stop 0b: The GTIOCnB pin outputs low when counting stops 1b: The GTIOCnB pin outputs high when counting stops
21	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20 to 16	GTIOB[4:0]	0h	RW	GTIOCnB Pin Function Select See **Table 5.7-8**.
15, 14	NFCSA[1:0]	0h	RW	Noise Filter A Sampling Clock Select 00b: PCLKD/1 01b: PCLKD/4 10b: PCLKD/16 11b: PCLKD/64
13	NFAEN	0h	RW	Noise Filter A Enable 0b: The noise filter for the GTIOCnA pin is disabled 1b: The noise filter for the GTIOCnA pin is enabled
12, 11	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
10, 9	OADF[1:0]	0h	RW	GTIOCnA Pin Disable Value Setting 00b: None of the following sources is specified 01b: GTIOCnA pin is set to Hi-Z in response to control for output negation 10b: GTIOCnA pin is set to 0 in response to control for output negation 11b: GTIOCnA pin is set to 1 in response to control for output negation
8	OAE	0h	RW	GTIOCnA Pin Output Enable 0b: Output is disabled 1b: Output is enabled
7	OAHLD	0h	RW	GTIOCnA Pin Output Setting at Count Start/Stop 0b: The GTIOCnA pin output level at the start or stop of counting depends on the register setting 1b: The GTIOCnA pin output level is retained at the start or stop of counting
6	OADFLT	0h	RW	GTIOCnA Pin Output Value Setting at Count Stop 0b: The GTIOCnA pin outputs low when counting stops 1b: The GTIOCnA pin outputs high when counting stops
5	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4 to 0	GTIOA[4:0]	0h	RW	GTIOCnA Pin Function Select See **Table 5.7-8**.

GTIOA[4:0] bits (GTIOCnA Pin Function Select)

The GTIOA[4:0] bits select the GTIOCnA pin function. For details, see **Table 5.7-8**.

OADFLT bit (GTIOCnA Pin Output Value Setting at Count Stop)

The OADFLT bit sets whether the GTIOCnA pin outputs the high or low level when counting stops.

OAHLD bit (GTIOCnA Pin Output Setting at Count Start/Stop)

The OAHLD bit specifies whether the GTIOCnA pin output level is retained or it is in accord with the register setting at the start or stop of counting.

When the OAHLD bit is set to 0b:

- The value specified in bit [4] of the GTIOA[4:0] bits is output when counting starts
- The value specified in the OADFLT bit is output when counting stops
- If the OADFLT bit is modified while counting stops, the new value is immediately reflected in the output.

When the OAHLD bit is set to 1b:

- The output is retained when counting starts or stops.

OAE bit (GTIOCnA Pin Output Enable)

The OAE bit disables or enables the GTIOCnA pin output.

When GTCCRA register is used as the input capture register (at least one bit in the GTICASR register is set to 1b), the GTIOCnA pin does not output regardless of the OAE bit value.

OADF[1:0] bits (GTIOCnA Pin Disable Value Setting)

The OADF[1:0] bits select the output value of the GTIOCnA pin in response to a request to disable output from the POEG.

NFAEN bit (Noise Filter A Enable)

The NFAEN bit disables or enables the noise filter for input from the GTIOCnA pin. Because changing the value of the bit might lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the GTIOR register before doing so.

NFCSA[1:0] bits (Noise Filter A Sampling Clock Select)

The NFCSA[1:0] bits set the sampling interval for the noise filter of the GTIOCnA pin. When setting these bits, wait for 2 cycles of the selected sampling interval before setting the input capture function.

GTIOB[4:0] bits (GTIOCnB Pin Function Select)

The GTIOB[4:0] bits select the GTIOCnB pin function. For details, see **Table 5.7-8**.

OBDFLT bit (GTIOCnB Pin Output Value Setting at Count Stop)

The OBDFLT bit sets whether the GTIOCnB pin outputs the high or low level when counting stops.

OBHLD bit (GTIOCnB Pin Output Setting at Count Start/Stop)

The OBHLD bit specifies whether the GTIOCnB pin output level is retained or it is in accord with the register setting at the start or stop of counting.

When the OBHLD bit is set to 0b:

- The value specified in bit [4] of the GTIOB[4:0] bits is output when counting starts
- The value specified in the OBDFLT bit is output when counting stops
- If the OBDFLT bit is modified while counting stops, the new value is immediately reflected in the output.

When the OBHLD bit is set to 1b:

- The output is retained when counting starts or stops.

OBE bit (GTIOCnB Pin Output Enable)

The OBE bit disables or enables the GTIOCnB pin output.

When GTCCRB register is used as the input capture register (at least one bit in the GTICBSR register is set to 1b), the GTIOCnB pin does not output regardless of the OBE bit value.

OBDF[1:0] bits (GTIOCnB Pin Disable Value Setting)

The OBDF[1:0] bits select the output value of the GTIOCnB pin in response to a request to disable output from the POEG.

NFBEN bit (Noise Filter B Enable)

The NFBEN bit disables or enables the noise filter for input from the GTIOCnB pin. Because changing the value of the bit might lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the GTIOR register before doing so.

NFCSB[1:0] bits (Noise Filter B Sampling Clock Select)

The NFCSB[1:0] bits set the sampling interval for the noise filter of the GTIOCnB pin. When setting these bits, wait for 2 cycles of the selected sampling interval before setting the input capture function.

Table 5.7-8 Settings of GTIOA[4:0] and GTIOB[4:0] Bits

GTIOA/GTIOB[4:0] Bits					Function		
b4	b3	b2	b1	b0	b4	b3, b2*1,*2,*3	b1, b0*2
0	0	0	0	0	Initial output is low	Output retained at cycle end	Output retained at GTCCRA/GTCCRB compare match
0	0	0	0	1			Low-level output at GTCCRA/GTCCRB compare match
0	0	0	1	0			High-level output at GTCCRA/GTCCRB compare match
0	0	0	1	1			Output toggled at GTCCRA/GTCCRB compare match
0	0	1	0	0		Low-level output at cycle end	Output retained at GTCCRA/GTCCRB compare match
0	0	1	0	1			Low-level output at GTCCRA/GTCCRB compare match
0	0	1	1	0			High-level output at GTCCRA/GTCCRB compare match
0	0	1	1	1			Output toggled at GTCCRA/GTCCRB compare match
0	1	0	0	0		High-level output at cycle end	Output retained at GTCCRA/GTCCRB compare match
0	1	0	0	1			Low-level output at GTCCRA/GTCCRB compare match
0	1	0	1	0			High-level output at GTCCRA/GTCCRB compare match
0	1	0	1	1			Output toggled at GTCCRA/GTCCRB compare match
0	1	1	0	0		Output toggled at cycle end	Output retained at GTCCRA/GTCCRB compare match
0	1	1	0	1			Low-level output at GTCCRA/GTCCRB compare match
0	1	1	1	0			High-level output at GTCCRA/GTCCRB compare match
0	1	1	1	1			Output toggled at GTCCRA/GTCCRB compare match
1	0	0	0	0	Initial output is high	Output retained at cycle end	Output retained at GTCCRA/GTCCRB compare match
1	0	0	0	1			Low-level output at GTCCRA/GTCCRB compare match
1	0	0	1	0			High-level output at GTCCRA/GTCCRB compare match
1	0	0	1	1			Output toggled at GTCCRA/GTCCRB compare match
1	0	1	0	0		Low-level output at cycle end	Output retained at GTCCRA/GTCCRB compare match
1	0	1	0	1			Low-level output at GTCCRA/GTCCRB compare match
1	0	1	1	0			High-level output at GTCCRA/GTCCRB compare match
1	0	1	1	1			Output toggled at GTCCRA/GTCCRB compare match
1	1	0	0	0		High-level output at cycle end	Output retained at GTCCRA/GTCCRB compare match
1	1	0	0	1			Low-level output at GTCCRA/GTCCRB compare match
1	1	0	1	0			High-level output at GTCCRA/GTCCRB compare match
1	1	0	1	1			Output toggled at GTCCRA/GTCCRB compare match
1	1	1	0	0		Output toggled at cycle end	Output retained at GTCCRA/GTCCRB compare match
1	1	1	0	1			Low-level output at GTCCRA/GTCCRB compare match
1	1	1	1	0			High-level output at GTCCRA/GTCCRB compare match
1	1	1	1	1			Output toggled at GTCCRA/GTCCRB compare match

Note 1. The cycle end means an overflow (GTCNT changes from GTPR to 0 in up-counting), an underflow (GTCNT changes from 0 to GTPR in down-counting), or counter clearing for saw-wave mode, and a trough (GTCNT changes from 0b to 1b) for triangle-wave mode.

Note 2. When the timing of a cycle end and the timing of a GTCCRA/GTCCRB compare match are the same in a compare-match operation, the b3 and b2 settings are given priority in saw-wave PWM mode, and the b1 and b0 settings are given priority in any other mode.

Note 3. In event count operation where at least one bit in GTUPSR or GTDNSR is set to 1b, the setting of b3 and b2 is ignored.

5.7.2.2.15 General-Purpose Timer Interrupt Output Setting Register (GPTm_n_GTINTAD)

The GTINTAD enables or disables interrupt requests, A/D converter start requests and output disable requests.

Access Size : 32 bits

Address : <GPTm_base> + 0038h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	GRPABL	GRPABH	GRPDTE	-	-	GRP[1:0]		-	-	-	-	ADTRBDEN	ADTRBUEN	ADTRADEN	ADTRAUEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	GTINTPR[1:0]		GTINTF	GTINTE	GTINTD	GTINTC	GTINTB	GTINTA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
30	GRPABL	0h	RW	Same Time Output Level Low Disable Request Enable 0b: Same time output level low disable request disabled 1b: Same time output level low disable request enabled
29	GRPABH	0h	RW	Same Time Output Level High Disable Request Enable 0b: Same time output level high disable request disabled 1b: Same time output level high disable request enabled
28	GRPDTE	0h	RW	Dead Time Error Output Disable Request Enable 0b: Dead time error output disable request disabled 1b: Dead time error output disable request enabled
27, 26	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
25, 24	GRP[1:0]	0h	RW	Output Disable Source Select 00b: Group A / Group E output disable source is selected 01b: Group B / Group F output disable source is selected 10b: Group C / Group G output disable source is selected 11b: Group D / Group H output disable source is selected
23 to 20	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
19	ADTRBDEN	0h	RW	GTADTRB Compare Match (Down-Counting) A/D Converter Start Request Enable 0b: Disable A/D conversion start requests 1b: Enable A/D conversion start requests
18	ADTRBUEN	0h	RW	GTADTRB Compare Match (Up-Counting) A/D Converter Start Request Enable 0b: Disable A/D conversion start requests 1b: Enable A/D conversion start requests
17	ADTRADEN	0h	RW	GTADTRA Compare Match (Down-Counting) A/D Converter Start Request Enable 0b: Disable A/D conversion start requests 1b: Enable A/D conversion start requests
16	ADTRAUEN	0h	RW	GTADTRA Compare Match (Up-Counting) A/D Converter Start Request Enable 0b: Disable A/D conversion start requests 1b: Enable A/D conversion start requests
15 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7, 6	GTINTPR[1:0]	0h	RW	GTPR Compare Match Interrupt Enable 00b: Disable interrupt requests 01b: In saw-wave mode, interrupt requests are enabled on overflow. In triangle-wave mode, interrupt requests are enabled at crests. 10b: In saw-wave mode, interrupt requests are enabled on underflow. In triangle-wave mode, interrupt requests are enabled at troughs. 11b: In saw-wave mode, interrupt requests are enabled at both overflows and underflows. In triangle-wave mode, interrupt requests are enabled at both crests and troughs.

Bit	Bit Name	Initial Value	R/W	Description
5	GTINTF	0h	RW	GTCCRF Compare Match Interrupt Enable 0b: Disable interrupt requests 1b: Enable interrupt requests
4	GTINTE	0h	RW	GTCCRE Compare Match Interrupt Enable 0b: Disable interrupt requests 1b: Enable interrupt requests
3	GTINTD	0h	RW	GTCCRD Compare Match Interrupt Enable 0b: Disable interrupt requests 1b: Enable interrupt requests
2	GTINTC	0h	RW	GTCCRC Compare Match Interrupt Enable 0b: Disable interrupt requests 1b: Enable interrupt requests
1	GTINTB	0h	RW	GTCCRB Compare Match/Input Capture Interrupt Enable 0b: Disable interrupt requests 1b: Enable interrupt requests
0	GTINTA	0h	RW	GTCCRA Compare Match/Input Capture Interrupt Enable 0b: Disable interrupt requests 1b: Enable interrupt requests

GTINTA bit (GTCCRA Compare Match/Input Capture Interrupt Enable)

The GTINTA bit enables and disables the interrupt request (GPT_Ux_gpt_gtcia_n_m) in response to compare matches with or input capture by the GTCCRA register.

GTINTB bit (GTCCRB Compare Match/Input Capture Interrupt Enable)

The GTINTB bit enables and disables the interrupt request (GPT_Ux_gpt_gtcib_n_m) in response to compare matches with or input capture by the GTCCRB register.

GTINTC bit (GTCCRC Compare Match Interrupt Enable)

The GTINTC bit enables and disables the interrupt request (GPT_Ux_gpt_gtcic_n_m) in response to compare matches with the GTCCRC register.

GTINTD bit (GTCCRD Compare Match Interrupt Enable)

The GTINTD bit enables and disables the interrupt request (GPT_Ux_gpt_gtcid_n_m) in response to compare matches with the GTCCRD register.

GTINTE bit (GTCCRE Compare Match Interrupt Enable)

The GTINTE bit enables and disables the interrupt request (GPT_Ux_gpt_gtcie_n_m) in response to compare matches with the GTCCRE register.

GTINTF bit (GTCCRF Compare Match Interrupt Enable)

The GTINTF bit enables and disables the interrupt request (GPT_Ux_gpt_gtcif_n_m) in response to compare matches with the GTCCRF register.

GTINTPR[1:0] bits (GTPR Compare Match Interrupt Enable)

The GTINTPR[1:0] bits enable and disable the interrupt requests (GPT_Ux_gpt_gtciv_n_m and GPT_Ux_gpt_gciu_n_m) in response to compare matches with the GTPR register (and overflows of the GTCNT counter) and underflows of the GTCNT counter.

ADTRAUEN bit (GTADTRA Compare Match (Up-Counting) A/D Converter Start Request Enable)

The ADTRAUEN bit enables or disables A/D converter start requests generated by GTADTRA compare matches during GTCNT up-counting. The setting of this bit has no effect during event counting and A/D converter start requests are not generated.

ADTRADEN bit (GTADTRA Compare Match (Down-Counting) A/D Converter Start Request Enable)

The ADTRADEN bit enables or disables A/D converter start requests generated by GTADTRA compare matches during GTCNT down-counting. The setting of this bit has no effect during event counting and A/D converter start requests are not generated.

ADTRBUEN bit (GTADTRB Compare Match (Up-Counting) A/D Converter Start Request Enable)

The ADTRBUEN bit enables or disables A/D converter start requests generated by GTADTRB compare matches during GTCNT up-counting. The setting of this bit has no effect during event counting and A/D converter start requests are not generated.

ADTRBDEN bit (GTADTRB Compare Match (Down-Counting) A/D Converter Start Request Enable)

The ADTRBDEN bit enables or disables A/D converter start requests generated by GTADTRB compare matches during GTCNT down-counting. The setting of this bit has no effect during event counting and A/D converter start requests are not generated.

GRP[1:0] bits (Output Disable Source Select)

These bits select the group of output disable requests from GPT to POEG and the group of output disable sources for the GTIOCnA and GTIOCnB pins from POEG to GPT.

The output disable request to POEG is output to the group selected in the GRP[1:0] bits, with dead-time errors, simultaneous high-level output, and simultaneous low-level output sources following their respective disable request enable bits.

GTST.ODF shows the request of the output disable source group that is selected with the GRP[1:0] bits. Set the GRP[1:0] bits when both GTIOR.OAE and GTIOR.OBE bits are 0b.

GRPDTE bit (Dead Time Error Output Disable Request Enable)

The GRPDTE bit enables or disables dead time error output disable requests. The setting of this bit has no effect during event counting.

GRPABH bit (Simultaneous Output Level High Disable Request Enable)

The GRPABH bit enables or disables the output disable request when the GTIOCnA and GTIOCnB pins output 1 at the same time.

GRPABL bit (Simultaneous Output Level Low Disable Request Enable)

The GRPABL bit enables or disables the output disable request when the GTIOCnA and GTIOCnB pins output 0 at the same time.

5.7.2.2.16 General-Purpose Timer Status Register (GPTm_n_GTST)

The GTST indicates the status of the GPT.

Access Size : 32 bits

Address : <GPTm_base> + 003Ch + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	OABLF	OABHF	DTEF	-	-	-	ODF	-	-	-	-	ADTRBDF	ADTRBUF	ADTRADF	ADTRAUF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW0	R	R	R	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TUCF	-	-	-	-	ITCNT[2:0]			TCFPU	TCFPO	TCFF	TCFE	TCFD	TCFC	TCFB	TCFA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW0	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
30	OABLF	0h	R	Simultaneous Output Level Low Flag 0b: No simultaneous generation of 0 both for the GTIOCnA and GTIOCnB pins has occurred. 1b: A simultaneous generation of 0 both for the GTIOCnA and GTIOCnB pins has occurred.
29	OABHF	0h	R	Simultaneous Output Level High Flag 0b: No simultaneous generation of 1 both for the GTIOCnA and GTIOCnB pins has occurred. 1b: A simultaneous generation of 1 both for the GTIOCnA and GTIOCnB pins has occurred.
28	DTEF	0h	R	Dead Time Error Flag 0b: No dead time error is generated 1b: A dead time error is generated
27 to 25	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
24	ODF	0h	R	Output Disable Flag 0b: No output disable request is generated 1b: An output disable request is generated
23 to 20	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
19	ADTRBDF	0h	RW	GTADTRB Compare Match (Down-Counting) A/D Converter Start Request Flag 0b: No GTADTRB register compare match has occurred during counting down 1b: A GTADTRB register compare match has occurred during counting down Only "0" can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, write "0" to clear and write "1" to keep the flag.
18	ADTRBUF	0h	RW	GTADTRB Compare Match (Up-Counting) A/D Converter Start Request Flag 0b: No GTADTRB register compare match has occurred during counting up 1b: A GTADTRB register compare match has occurred during counting up Only "0" can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, write "0" to clear and write "1" to keep the flag.
17	ADTRADF	0h	RW	GTADTRA Compare Match (Down-Counting) A/D Converter Start Request Flag 0b: No GTADTRA register compare match has occurred during counting down 1b: A GTADTRA register compare match has occurred during counting down Only "0" can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, write "0" to clear and write "1" to keep the flag.
16	ADTRAUF	0h	RW	GTADTRA Compare Match (Up-Counting) A/D Converter Start Request Flag 0b: No GTADTRA register compare match has occurred during counting up 1b: A GTADTRA register compare match has occurred during counting up Only "0" can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, write "0" to clear and write "1" to keep the flag.
15	TUCF	0h	R	Count Direction Flag 0b: GTCNT counter counts downward 1b: GTCNT counter counts upward

Bit	Bit Name	Initial Value	R/W	Description
14 to 11	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10 to 8	ITCNT[2:0]	0h	R	GTCIV/GTCIU Interrupt Skipping Count Counter Timer interrupt skipping count counter
7	TCFPU	0h	RW	Underflow Flag 0b: No underflow (trough) occurred 1b: An underflow (trough) occurred Only "0" can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, write "0" to clear and write "1" to keep the flag.
6	TCFPO	0h	RW	Overflow Flag 0b: No overflow (crest) occurred 1b: An overflow (crest) occurred Only "0" can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, write "0" to clear and write "1" to keep the flag.
5	TCFF	0h	RW	Input Compare Match Flag F 0b: No compare match of GTCCRF is generated 1b: A compare match of GTCCRF is generated Only "0" can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, write "0" to clear and write "1" to keep the flag.
4	TCFE	0h	RW	Input Compare Match Flag E 0b: No compare match of GTCCRE is generated 1b: A compare match of GTCCRE is generated Only "0" can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, write "0" to clear and write "1" to keep the flag.
3	TCFD	0h	RW	Input Compare Match Flag D 0b: No compare match of GTCCRD is generated 1b: A compare match of GTCCRD is generated Only "0" can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, write "0" to clear and write "1" to keep the flag.
2	TCFC	0h	RW	Input Compare Match Flag C 0b: No compare match of GTCCRC is generated 1b: A compare match of GTCCRC is generated Only "0" can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, write "0" to clear and write "1" to keep the flag.
1	TCFB	0h	RW	Input Capture/Compare Match Flag B 0b: No input capture/compare match of GTCCRB is generated 1b: An input capture/compare match of GTCCRB is generated Only "0" can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, write "0" to clear and write "1" to keep the flag.
0	TCFA	0h	RW	Input Capture/Compare Match Flag A 0b: No input capture/compare match of GTCCRA is generated 1b: An input capture/compare match of GTCCRA is generated Only "0" can be written to clear the flag. When clearing the ADTRAUF, ADTRADF, ADTRBUF, or ADTRBDF flag, write "0" to clear and write "1" to keep the flag.

TCFA flag (Input Capture/Compare Match Flag A)

The TCFA flag indicates the status of the input capture or compare match with GTCCRA.

[Setting conditions]

- GTCNT = GTCCRA, when the GTCCRA register functions as a compare match register
- The GTCNT counter value is transferred to GTCCRA by the input capture signal when the GTCCRA register functions as an input capture register.

[Clearing condition]

- 0b is written to this flag.

TCFB flag (Input Capture/Compare Match Flag B)

The TCFB flag indicates the status of the input capture or compare match with GTCCRB.

[Setting conditions]

- GTCNT = GTCCRB, when the GTCCRB register functions as a compare match register
- The GTCNT counter value is transferred to GTCCRB by the input capture signal when the GTCCRB register functions as an input capture register.

[Clearing condition]

- 0b is written to this flag.

TCFC flag (Input Compare Match Flag C)

The TCFC flag indicates the status of the compare match with GTCCRC.

When GTCCRC performs buffer operation, it does not perform compare match.

[Setting condition]

- GTCNT = GTCCRC.

[Clearing condition]

- 0b is written to this flag.

[Conditions for no comparison]

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 01b, 10b, 11b (GTCCRC performs buffer operation).

TCFD flag (Input Compare Match Flag D)

The TCFD flag indicates the status of the compare match with GTCCRD.

When GTCCRD performs buffer operation, it does not perform compare match.

[Setting condition]

- GTCNT = GTCCRD.

[Clearing condition]

- 0b is written to this flag.

[Conditions for no comparison]

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 10b, 11b (GTCCRD performs buffer operation).

TCFE flag (Input Compare Match Flag E)

The TCFE flag indicates the status of the compare match with GTCCRE.

When GTCCRE performs buffer operation, it does not perform compare match.

[Setting condition]

- GTCNT = GTCCRE.

[Clearing condition]

- 0b is written to this flag.

[Conditions for no comparison]

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 01b, 10b, 11b (GTCCRE performs buffer operation).

TCFF flag (Input Compare Match Flag F)

The TCFF flag indicates the status of the compare match with GTCCRF.

When GTCCRF performs buffer operation, it does not perform compare match.

[Setting condition]

- GTCNT = GTCCRF.

[Clearing condition]

- 0b is written to this flag.

[Conditions for no comparison]

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 10b, 11b (GTCCRF performs buffer operation).

TCFPO flag (Overflow Flag)

The TCFPO flag indicates when an overflow or crest has occurred.

[Setting conditions]

- In saw-wave mode, an overflow (GTCNT changes from GTPR to 0b in up-counting) has occurred
- In triangle-wave mode, a crest (GTCNT changes from GTPR to GTPR - 1) has occurred
- In counting by hardware sources, an overflow (GTCNT changes from GTPR to 0b in up-counting) has occurred.

[Clearing condition]

- 0b is written to this flag.

TCFPU flag (Underflow Flag)

The TCFPU flag indicates when an underflow or trough has occurred.

[Setting conditions]

- In saw-wave mode, an underflow (GTCNT changes from 0 to GTPR in down-counting) has occurred
- In triangle-wave mode, a trough (GTCNT changes from 0 to 1) has occurred
- In counting by hardware sources, an underflow (GTCNT changes from 0 to GTPR in down-counting) has occurred.

[Clearing condition]

- 0b is written to this bit.

ITCNT[2:0] bits (GPT_Ux_gpt_gtciv_n_m/GPT_Ux_gpt_gciu_n_m Interrupt Skipping Count Counter)

When the GPT_Ux_gpt_gtciv_n_m/GPT_Ux_gpt_gciu_n_m (x = 0 or 1, m = 0 to 7) interrupt skipping function is used (the GTITC.IVTC[1:0] bits are set to a value other than 00b), the counter in the ITCNT[2:0] bits is incremented by 1 every time the GPT_Ux_gpt_gtciv_n_m/GPT_Ux_gpt_gciu_n_m interrupt source that is selected in GTITC.IVTC[1:0] is generated.

[Clearing conditions]

- The GPT_Ux_gpt_gtciv_n_m/GPT_Ux_gpt_gciu_n_m interrupt skipping function is not used (GTITC.IVTT[2:0] is 000b when GTITC.IVTC[1:0] is 00b)
- The GPT_Ux_gpt_gtciv_n_m/GPT_Ux_gpt_gciu_n_m interrupt skipping count matches the specified count (ITCNT[2:0] matches the skipping count specified in GTITC.IVTT[2:0]).

TUCF flag (Count Direction Flag)

The TUCF flag indicates the count direction of GTCNT. In event count operation, this flag is set to 1b in up-counting and to 0b in down-counting.

ADTRAUF flag (GTADTRA Compare Match (Up-Counting) A/D Converter Start Request Flag)

The ADTRAUF flag indicates whether or not a compare match with the GTADTRA register has occurred during counting up.

[Setting condition]

- GTCNT = GTADTRA during counting up

[Clearing condition]

- 0b is written to this flag.

ADTRADF flag (GTADTRA Compare Match (Down-Counting) A/D Converter Start Request Flag)

The ADTRADF flag indicates whether or not a compare match with the GTADTRA register has occurred during counting down.

[Setting condition]

- GTCNT = GTADTRA during counting down

[Clearing condition]

- 0b is written to this flag.

ADTRBUF flag (GTADTRB Compare Match (Up-Counting) A/D Converter Start Request Flag)

The ADTRBUF flag indicates whether or not a compare match with the GTADTRB register has occurred during counting up.

[Setting condition]

- GTCNT = GTADTRB during counting up

[Clearing condition]

- 0b is written to this flag.

ADTRBDF flag (GTADTRB Compare Match (Down-Counting) A/D Converter Start Request Flag)

The ADTRBDF flag indicates whether or not a compare match with the GTADTRB register has occurred during counting down.

[Setting condition]

- GTCNT = GTADTRB during counting down

[Clearing condition]

- 0b is written to this flag.

ODF flag (Output Disable Flag)

The ODF flag shows the request of the output disable source group selected in the GRP[1:0] bits.

When output is disabled, an output disable control is not released within the same cycle in which an output disable request is negated. It is released in the next cycle.

DTEF flag (Dead Time Error Flag)

The DTEF flag indicates that the timer output toggle point after the automatic addition of dead time has exceeded the timer cycle. This flag returns to 0b when the timer output toggle point after the automatic addition of dead time is returned to the cycle. This flag is read only. Writing 0b to clear the flag is not allowed.

When an interrupt by the DTEF flag is enabled (GTINTAD.GRPDTE = 1b), the DTEF flag is output to POEG as an output disable request each time the DTEF flag changes from 0b to 1b.

[Setting condition]

- The timer output toggle point after the automatic addition of dead time has exceeded the timer cycle.

  For triangle wave in up-counting: GTCCRA – GTDVU ≤ 0

  For triangle wave in down-counting: GTCCRA – GTDVD < 0

  For saw wave 1 shot pulse mode in up-counting: GTCCRA – GTDVU < 0 or GTCCRA + GTDVD > GTPR

  For saw wave 1 shot pulse mode in down-counting: GTCCRA + GTDVU > GTPR or GTCCRA - GTDVD < 0

[Clearing condition]

- The timer output toggle point after the automatic addition of dead time is within the timer cycle.

OABHF flag (Simultaneous Output Level High Flag)

The OABHF flag indicates that the GTIOCnA and GTIOCnB pins output 1 at the same time.

When the GTIOCnA or GTIOCnB pin outputs 0, this flag returns to 0b. This flag is read only. Writing 0 to clear the flag is prohibited.

When the output disable request by the OABHF flag is enabled (GTINTAD.GRPABH = 1b), the OABHF flag is output to POEG as an output disable request. The GPT does not have an interrupt to indicate that outputs have been simultaneous driven to the high level. Use the interrupt function in the POEG if this is necessary.

[Setting condition]

- The GTIOCnA and GTIOCnB pins output 1 at the same time when both OAE and OBE bits are set to 1b.

[Clearing conditions]

- The GTIOCnA pin output value is different from the GTIOCnB pin output value when both OAE and OBE bits are set to 1b
- The GTIOCnA and GTIOCnB pins output 0 at the same time when both OAE and OBE bits are set to 1b
- Either the OAE bit or OBE bit is set to 0b.

OABLF flag (Simultaneous Output Level Low Flag)

The OABLF flag indicates that the GTIOCnA and GTIOCnB pins output 0 at the same time.

When the GTIOCnA pin or GTIOCnB pin outputs 1, this flag returns to 0b. This flag is read only. Writing 0b to clear the flag is prohibited.

When the output disable request by the OABLF flag is enabled (GTINTAD.GRPABL = 1b), the OABLF flag is output to POEG as an output disable request. The GPT does not have an interrupt to indicate that outputs have been simultaneous driven to the low level. Use the interrupt function in the POEG if this is necessary.

[Setting condition]

- The GTIOCnA and GTIOCnB pins output 0 at the same time when both OAE and OBE bits are set to 1b.

[Clearing conditions]

- The GTIOCnA pin output value is different from the GTIOCnB pin output value when both OAE and OBE bits are set to 1b
- The GTIOCnA and GTIOCnB pins output 1 at the same time when both OAE and OBE bits are set to 1b
- Either the OAE bit or the OBE bit is set to 0b.

The compare-target signals to generate the OABHF/OABLF flag are the compare match outputs (PWM outputs) signals before they are masked by the output disable function. Even during the output disable condition, compare match operation continues internally, where the OABHF or OABLF flag is updated based on the operation results.

5.7.2.2.17 General-Purpose Timer Buffer Enable Register (GPTm_n_GTBER)

The GTBER register provides settings for buffer operation. Set the GTBER register while the GTCNT counter is stopped.

Access Size : 32 bits
Address : <GPTm_base> + 0040h + n x 0100h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	ADTDB	ADTTB[1:0]		-	ADTDA	ADTTA[1:0]		-	CCRSWT	PR[1:0]		CCRB[1:0]		CCRA[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	W	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DBRTSADB	DBRTEADB	DBRTSADA	DBRTEADA	DBRTSCB	DBRTECB	DBRTSCA	DBRTECA	-	-	-	-	BD3	BD2	BD1	BD0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
30	ADTDB	0h	RW	GTADTRB Double Buffer Operation 0b: Operates as a single buffer (GTADTBRB register → GTADTRB register) 1b: Operates as a double buffer (GTADDTDBRB register → GTADTRB register)
29, 28	ADTTB[1:0]	0h	RW	GTADTRB Buffer Transfer Timing Select In triangle-wave mode 00b: No transfer 01b: Transfer at (crest) 11b: Transfer at (crest/trough) In saw-wave mode 00b: No transfer Other than 00b: Transfer on underflow (when counting down), overflow (when counting up), and counter clearing
27	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
26	ADTDA	0h	RW	GTADTRA Double Buffer Operation 0b: Operates as a single buffer (GTADTBRA register → GTADTRA register) 1b: Operates as a double buffer (GTADDTDBRA register → GTADTRA register)
25, 24	ADTTA[1:0]	0h	RW	GTADTRA Buffer Transfer Timing Select In triangle-wave mode 00b: No transfer 01b: Transfer at (crest) 11b: Transfer at both (crest/trough) In saw-wave mode 00b: No transfer Other than 00b: Transfer on underflow (when counting down), overflow (when counting up), and counter clearing
23	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
22	CCRSWT	-	W	The read value is undefined GTCCRA and GTCCRB Forcible Buffer Operation Writing 1b to this bit forces buffer transfer of GTCCRA and GTCCRB. This bit automatically returns to 0b after 1b is written. This bit is read as 0b.
21, 20	PR[1:0]	0h	RW	GTPR Buffer Operation 00b: No buffer operation 01b: Single buffer operation (GTPBR → GTPR) 0xb: Double buffer operation (GTPDBR → GTPBR → GTPR)

Bit	Bit Name	Initial Value	R/W	Description
19, 18	CCRB[1:0]	0h	RW	GTCCRB Buffer Operation 00b: No buffer operation 01b: Single buffer operation (GTCCRB ↔ GTCCRE) Others: Double buffer operation (GTCCRB ↔ GTCCRE ↔ GTCCRF)
17, 16	CCRA[1:0]	0h	RW	GTCCRA Buffer Operation 00b: No buffer operation 01b: Single buffer operation (GTCCRA ↔GTCCRC) Others: Double buffer operation (GTCCRA ↔ GTCCRC ↔ GTCCRD)
15	DBRTSADB	0h	RW	GTADTRB Register Double Buffer Repeat Operation Period Select 0b: Repeat transfer from an intermediate buffer to the GTADTRB register with the trough as a cycle 1b: Repeat transfer from an intermediate buffer to the GTADTRB register with the crest as a cycle
14	DBRTEADB	0h	RW	GTADTRB Register Double Buffer Repeat Operation Enable 0b: Disable double buffer repeat operation of the GTADTRB register 1b: Enable double buffer repeat operation of the GTADTRB register
13	DBRTSADA	0h	RW	GTADTRA Register Double Buffer Repeat Operation Period Select 0b: Repeat the transfer from an intermediate buffer to the GTADTRA register with the trough as a cycle 1b: Repeat the transfer from an intermediate buffer to the GTADTRA register with crest as a cycle
12	DBRTEADA	0h	RW	GTADTRA Register Double Buffer Repeat Operation Enable 0b: Disable double buffer repeat operation of the GTADTRA register 1b: Enable double buffer repeat operation of the GTADTRA register
11	DBRTSCB	0h	RW	GTCCRB Register Double Buffer Repeat Operation Period Select 0b: Repeat the transfer from an intermediate buffer to the GTCCRB register with the trough as a cycle 1b: Repeat the transfer from an intermediate buffer to the GTCCRB register with the crest as a cycle
10	DBRTECB	0h	RW	GTCCRB Register Double Buffer Repeat Operation Enable 0b: Disable double buffer repeat operation of the GTCCRB register 1b: Enable double buffer repeat operation of the GTCCRB register
9	DBRTSCA	0h	RW	GTCCRA Register Double Buffer Repeat Operation Period Select 0b: Repeat the transfer from an intermediate buffer to the GTCCRA register with the trough as a cycle 1b: Repeat the transfer from an intermediate buffer to the GTCCRA register with the crest as a cycle
8	DBRTECA	0h	RW	GTCCRA Register Double Buffer Repeat Operation Enable 0b: Disable double buffer repeat operation of the GTCCRA register 1b: Enable double buffer repeat operation of the GTCCRA register
7 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3	BD3	0h	RW	GTDVU/GTDVD Register Buffer Operation Disable 0b: Buffer operation is enabled 1b: Buffer operation is disabled
2	BD2	0h	RW	GTADTRA/GTADTRB Register Buffer Operation Disable 0b: Buffer operation is enabled 1b: Buffer operation is disabled
1	BD1	0h	RW	GTPR Buffer Operation Disable 0b: Buffer operation is enabled 1b: Buffer operation is disabled
0	BD0	0h	RW	GTCCR Buffer Operation Disable 0b: Buffer operation is enabled 1b: Buffer operation is disabled

BD[0] bit (GTCCR Buffer Operation Disable)

The BD[0] bit disables buffer operation in combination with GTCCRA, GTCCRB, GTCCRC, GTCCRD, GTCCRE, and GTCCRF.

When GTDTCR.TDE is 1b and when BD0 is set to 0b, GTCCRB does not perform buffer operation. The GTCCRB register is automatically set to a compare match value for negative-phase waveform with dead time.

A value for the BD0 bit in the channel related to the position of the bit written with 1b by the GTSECSR register can be set when 1b is written to the GTSECR.SBDCE or GTSECR.SBDCD.

If the DBRTECm (m = A, B) bit is 1b, when a double buffer is used in saw-wave one-shot pulse mode, triangle-wave PWM mode 3, or triangle-wave PWM mode 2, buffer operation from an intermediate buffer to the GTCCRm register is performed even if the BD[0] bit is set to 1b.

BD[1] bit (GTPR Buffer Operation Disable)

The BD[1] bit disables buffer operation in combination with GTPR and GTPBR.

A value of the BD1 bit in the channel related to the position of the bit written with 1b by the GTSECSR register can be set when 1b is written to the GTSECR.SBDPE or GTSECR.SBDPD.

BD[2] bit (GTADTR Buffer Operation Disable)

The BD[2] bit disables buffer operation in combination with GTADTRA, GTADTRB, GTADTBRA, GTADTBRB, GTADTDBRA, and GTADTDBRB of GPT. In event count operation, this bit is not available and the GTADTR buffer operation is not performed. The setting of this bit has no effect during event counting and the GTADTRA and GTADTRB registers do not perform buffer operation.

By writing 1b to the SBDAE or SBDAD bit of the GTSECR register, a value can be set in the BD[2] bit of the corresponding channel at the bit position which is set to 1b in the GTSECSR register.

If the DBRTEADm (m = A, B) bit is 1b, when the GTADTRm register is made to operate as a double buffer with triangle waves and the timing of buffer transfer is set for both troughs and crests (the GTBER.ADTTm[1:0] bits are set to 11b), buffer operation from an intermediate buffer to the GTADTRm register is performed even if the BD[2] bit is set to 1b.

BD[3] bit (GTDV Buffer Operation Disable)

The BD[3] bit disables buffer operation in combination with GTDVU, GTDVD, GTDBD, and GTDBU of GPT. When the GTDTCR.TDFER bit is set to 1b and when BD[3] is set to 0b, buffer operation is not performed and the GTDVD value is set as a value of GTDVU automatically. In event count operation, this bit is not available and the GTDV buffer operation is not performed. The setting of this bit has no effect during event counting and the GTDVU and GTDVD registers do not perform buffer operation.

By writing 1b to the SBDDE or SBDDD bit of the GTSECR register, a value can be set in the BD[3] bit of the channels in the bit position which is set to 1b in the GTSECSR register.

DBRTECm bit (GTCCRm Register Double Buffer Repeat Operation Enable) (m = A, B)

To make the GTCCRm register operate as a double buffer, this bit enables operation to repeat transfer from an intermediate buffer to the GTCCRm register on a cyclic basis over the period where buffer transfer is disabled.

This bit is valid in saw-wave one-shot pulse mode, triangle-wave PWM mode 3, and triangle-wave PWM mode 2 that makes the GTCCRm register operate as a double buffer.

In the period where buffer transfer is disabled, setting the BD[0] bit (CPU writing or simultaneous buffer operation control by the GTSECSR register) and buffer transfer extended skipping (except for skipping by counting both crests and troughs with triangle waves) by the GTEITLB register are both possible.

When the DBRTECm bit is 1b, writing by the CPU to the GTCCRm register sets the same value in temporary register x (x = C, E). Also, the value of the GTCCRx (x = C, E) register is transferred to temporary register x (x = C, E) by forcible buffer transfer.

DBRTSCm bit (GTCCRm Register Double Buffer Repeat Operation Period Select) (m = A, B)

This bit selects the target period for repeated operation when GTCCRm double buffer repeat operation is enabled. This bit is valid in triangle-wave PWM mode 2.

When disabling or enabling of buffer transfer is to be changed during counting by GTCNT, change it with the period on the basis of the same cyclic period as that set by the DBRTSCm bit.

DBRTEADm bit (GTADTRm Register Double Buffer Repeat Operation Enable) (m = A, B)

To make the GTADTRm register operate as a double buffer, this bit enables operation to repeat transfer from an intermediate buffer to the GTADTRm register on a cyclic basis over the period where buffer transfer is disabled.

This bit is valid when the GTADTRm register is made to operate as a double buffer (the GTBER.ADTDm is 1b) with triangle waves and the timing of buffer transfer is set for both troughs and crests (the GTBER.ADTTm[1:0] bits are 11b).

In the period where buffer transfer is disabled, setting the BD[2] bit (CPU writing or simultaneous buffer operation control by the GTSECSR register) and buffer transfer extended skipping (except for skipping by counting both crests and troughs with triangle waves) by the GTEITLB register are both possible.

When the ADTDm bit is 1b and the DBRTEADm bit is 1b, writing by the CPU to the GTADTRm register sets the same value in temporary register ADm (m = A, B).

DBRTSADm bit (GTADTRm Register Double Buffer Repeat Operation Period Select) (m = A, B)

This bit selects the target period for repeated operation when GTADTRm double buffer repeat operation is enabled.

When disabling or enabling of buffer transfer is to be changed during counting by GTCNT, change it with the period on the basis of the same cyclic period set by the DBRTSADm bit.

CCRA[1:0] bits (GTCCRA Buffer Operation)

The CCRA[1:0] bits set buffer operation in combination with GTCCRA, GTCCRC, and GTCCRD. When the buffer operation is restricted by the operating mode set in GTCR, the GTCR setting is given priority.

The buffer operation mode is fixed in saw-wave one-shot pulse mode or triangle-wave PWM mode 3 (64-bit transfer at trough).

CCRB[1:0] bits (GTCCRB Buffer Operation)

The CCRB[1:0] bits set buffer operation in combination with GTCCRB, GTCCRE, and GTCCRF. When the buffer operation is restricted by the operating mode set in GTCR, the GTCR setting is given priority.

The buffer operation mode is fixed in saw-wave one-shot pulse mode or triangle-wave PWM mode 3 (64-bit transfer at trough).

PR[1:0] bits (GTPR Buffer Operation)

The PR[1:0] bits set buffer operation in combination with GTPR, GTPBR and GTPDBR.

CCRSWT bit (GTCCRA and GTCCRB Forcible Buffer Operation)

Writing 1b to the CCRSWT bit forces a buffer transfer of GTCCRA and GTCCRB. This bit automatically returns to 0b after 1b has been written. This bit is read as 0b, and is valid only when counting is stopped with a compare match operation specified.

ADTTA[1:0] bits (GTADTRA Buffer Transfer Timing Select)

The ADTTA[1:0] bits set the transfer timing for buffer operation of GTADTRA, GTADTBRA, and GTADTDBRA.

These bits are not available in event count operation.

ADTDA bit (GTADTRA Double Buffer Operation)

The ADTDA bit sets buffer operation in combination with GTADTRA, GTADTBRA, and GTADTDBRA. This bit is not available in event count operation.

ADTTB[1:0] bits (GTADTRB Buffer Transfer Timing Select)

The ADTTB[1:0] bits set the transfer timing for buffer operation of GTADTRB, GTADTBRB, and GTADTDBRB. This bit is not available in event count operation.

ADTDB bit (GTADTRB Double Buffer Operation)

The ADTDB bit sets buffer operation in combination with GTADTRB, GTADTBRB, and GTADTDBRB of GPT. This bit is not available in event count operation.

5.7.2.2.18 General-Purpose Timer Interrupt and A/D Converter Start Request Setting Register (GPTm_n_GTITC)

GTITC sets the skipping function for the GTCNT counter overflow (GTPR compare match) interrupt (OVFn) and underflow interrupt (UNFn). It also specifies whether to link other interrupts and A/D converter start requests with the OVFn/UNFn interrupt skipping function. An output disable request to the POEG cannot be linked with the OVFn/UNFn interrupt skipping function. This bit is not available in event count operation.

Access Size : 32 bits

Address : <GPTm_base> + 0044h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	ADTBL	-	ADTAL	-	IVTT[2:0]			IVTC[1:0]		ITLF	ITLE	ITLD	ITLC	ITLB	ITLA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
14	ADTBL	0h	RW	GTADTRB register A/D conversion start request interlink bit 0b: Not interlinked with GTCIV/GTCIU interrupt skipping function 1b: Interlinked with GTCIV/GTCIU interrupt skipping function
13	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12	ADTAL	0h	RW	GTADTRA register A/D conversion start request interlink bit 0b: Not interlinked with GTCIV/GTCIU interrupt skipping function 1b: Interlinked with GTCIV/GTCIU interrupt skipping function
11	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10 to 8	IVTT[2:0]	0h	RW	GTCIV/GTCIU interrupt skipping count selection bit 000b: No skipping 001b: Skipping count: 1 010b: Skipping count: 2 011b: Skipping count: 3 100b: Skipping count: 4 101b: Skipping count: 5 110b: Skipping count: 6 111b: Skipping count: 7
7, 6	IVTC[1:0]	0h	RW	GTCIV/GTCIU interrupt skipping function selection bit 00b: No skipping 01b: Skipping by counting both overflow/underflow in saw-wave mode and (crests) in triangle-wave mode 10b: Skipping by counting both overflow/underflow in saw-wave mode and (troughs) in triangle-wave mode 11b: Skipping by counting both overflow/underflow in saw-wave mode and both (troughs/crests) in triangle-wave mode
5	ITLF	0h	RW	GTCCRF register compare match interrupt interlink bit 0b: Not interlinked with GTCIV/GTCIU interrupt skipping function 1b: Interlinked with GTCIV/GTCIU interrupt skipping function
4	ITLE	0h	RW	GTCCRE register compare match interrupt interlink bit 0b: Not interlinked with GTCIV/GTCIU interrupt skipping function 1b: Interlinked with GTCIV/GTCIU interrupt skipping function

Bit	Bit Name	Initial Value	R/W	Description
3	ITLD	0h	RW	GTCCRD register compare match interrupt interlink bit 0b: Not interlinked with GTCIV/GTCIU interrupt skipping function 1b: Interlinked with GTCIV/GTCIU interrupt skipping function
2	ITLC	0h	RW	GTCCRC register compare match interrupt interlink bit 0b: Not interlinked with GTCIV/GTCIU interrupt skipping function 1b: Interlinked with GTCIV/GTCIU interrupt skipping function
1	ITLB	0h	RW	GTCCRB register compare match/input capture interrupt interlink bit 0b: Not interlinked with GTCIV/GTCIU interrupt skipping function 1b: Interlinked with GTCIV/GTCIU interrupt skipping function
0	ITLA	0h	RW	GTCCRA register compare match/input capture interrupt interlink bit 0b: Not interlinked with GTCIV/GTCIU interrupt skipping function 1b: Interlinked with GTCIV/GTCIU interrupt skipping function

ITLA bit (GTCCRA Compare Match/Input Capture Interrupt Link)

The ITLA bit specifies whether to link the GTCCRA compare match/input capture interrupt (GTCIA) with the OVFn/UNFn interrupt skipping function.

ITLB bit (GTCCRB Compare Match/Input Capture Interrupt Link)

The ITLB bit specifies whether to link the GTCCRB compare match/input capture interrupt (GTCIB) with the OVFn/UNFn interrupt skipping function.

ITLC bit (GTCCRC Compare Match Interrupt Link)

The ITLC bit specifies whether to link the GTCCRC compare match interrupt (GTCIC) with the OVFn/UNFn interrupt skipping function.

ITLD bit (GTCCRD Compare Match Interrupt Link)

The ITLD bit specifies whether to link the GTCCRD compare match interrupt (GTCID) with the OVFn/UNFn interrupt skipping function.

ITLE bit (GTCCRE Compare Match Interrupt Link)

The ITLE bit specifies whether to link the GTCCRE compare match interrupt (GTCIE) with the OVFn/UNFn interrupt skipping function.

ITLF bit (GTCCRF Compare Match Interrupt Link)

The ITLF bit specifies whether to link the GTCCRF compare match interrupt (GTCIF) with the OVFn/UNFn interrupt skipping function.

IVTC[1:0] bits (OVFn/UNFn Interrupt Skipping Function Select)

The IVTC[1:0] bits set the skipping function for the GTPR compare match (GTCNT overflow) interrupt (OVFn) and GTCNT counter underflow interrupt (UNFn).

IVTT[2:0] bits (OVFn/UNFn Interrupt Skipping Count Select)

The IVTT[2:0] bits set the skipping count for the GTPR compare match (GTCNT overflow) interrupt (OVFn) and GTCNT counter underflow interrupt (UNFn). When modifying the IVTT[2:0] bits, first set the IVTC[1:0] bits to 00b.

ADTAL bit (GTADTRA A/D Converter Start Request Link)

The ADTAL bit specifies whether to link the GTADTRA A/D converter start request with OVFn/UNFn interrupt skipping function.

ADTBL bit (GTADTRB A/D Converter Start Request Link)

The ADTBL bit specifies whether to link the GTADTRB A/D converter start request with OVFn/UNFn interrupt skipping function.

5.7.2.2.19 General-Purpose Timer Counter (GPTm_n_GTCNT)

Access Size : 32 bits

Address : <GPTm_base> + 0048h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GTCNT[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GTCNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GTCNT[31:0]	0h	RW	GTCNT is a 32-bit readable/writable counter. GTCNT can only be written to after counting stops. GTCNT must be set within the range of 0 ≤ GTCNT ≤ GTPR.

5.7.2.2.20 General-Purpose Timer Compare Capture Register k (GPTm_n_GTCCRk) (k = A to F)

The notations of the addresses are as follows.

p = 0 when k = A
p = 1 when k = B
p = 2 when k = C
p = 3 when k = E
p = 4 when k = D
p = 5 when k = F

Access Size : 32 bits
Address : <GPTm_base> + 004Ch + n x 0100h + 04h x p
Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GTCCR[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GTCCR[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GTCCR[31:0]	FFFF_FFFFh	RW	GTCCRk registers are readable/writable registers. GTCCRA and GTCCRB are registers used for both output compare and input capture. GTCCRC and GTCCRE are compare match registers, and can also function as buffer registers for GTCCRA and GTCCRB. GTCCRD and GTCCRF are compare match registers, and can also function as buffer registers for GTCCRC and GTCCRE (double-buffer registers for GTCCRA and GTCCRB).

5.7.2.2.21 General-Purpose Timer Cycle Setting Register (GPTm_n_GTPR)

Access Size : 32 bits

Address : <GPTm_base> + 0064h + n x 0100h

Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GTPR[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GTPR[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GTPR[31:0]	FFFF_FFFFh	RW	GTPR is a readable/writable register that sets the maximum count value of GTCNT. In saw-wave mode, the value of (GTPR + 1) is the cycle. In triangle-wave mode, the value of (GTPR value x 2) is the cycle.

5.7.2.2.22 General-Purpose Timer Cycle Setting Buffer Register (GPTm_n_GTPBR)

Access Size : 32 bits

Address : <GPTm_base> + 0068h + n x 0100h

Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GTPBR[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GTPBR15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GTPBR[31:0]	FFFF_FFFFh	RW	GTPBR is a readable/writable register that operates as a buffer register for GTPR.

5.7.2.2.23 General-Purpose Timer Cycle Setting Double-Buffer Register (GPTm_n_GTPDBR)

Access Size : 32 bits

Address : <GPTm_base> + 006Ch + n x 0100h

Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GTPDBR[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GTPDBR[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GTPDBR[31:0]	FFFF_FFFFh	RW	The GTPDBR register is a readable/writable register and operates as a buffer register for the GTPBR register (GTPR double buffer register).

5.7.2.2.24 A/D Converter Start Request Timing Register k (GPTm_n_GTADTRk) (k = A, B)

Access Size : 32 bits

Address : <GPTm_base> + 0070h + n x 0100h (k = A)
<GPTm_base> + 007Ch + n x 0100h (k = B)

Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GTADTR[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GTADTR[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GTADTR[31:0]	FFFF_FFFFh	RW	The GTADTRk register is a readable/writable register that sets the timing of the A/D conversion start request. When the GTADTRk register value matches the GTCNT counter value, an A/D converter start request is generated.

5.7.2.2.25 A/D Converter Start Request Timing Buffer Register k (GPTm_n_GTADTBRk) (k = A, B)

Access Size : 32 bits

Address : <GPTm_base> + 0074h + n x 0100h (k = A)
<GPTm_base> + 0080h + n x 0100h (k = B)

Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GTADTBR[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GTADTBR[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GTADTBR [31:0]	FFFF_FFFFh	RW	The GTADTBRk register is a readable/writable register and operates as a buffer register for the GTADTRk register.

5.7.2.2.26 A/D Converter Start Request Timing Double-Buffer Register k (GPTm_n_GTADTDBRk) (k = A, B)

Access Size : 32 bits

Address : <GPTm_base> + 0078h + n x 0100h (k = A)
<GPTm_base> + 0084h + n x 0100h (k = B)

Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GTADTDBR[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GTADTDBR[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GTADTDBR [31:0]	FFFF_FFFFh	RW	The GTADTDBRk register is a readable/writable register that operates as a buffer register for the GTADTBRk register (double buffer register for GTADTRk).

5.7.2.2.27 General-Purpose Timer Dead Time Control Register (GPTm_n_GTDTCR)

GTDTCR enables automatic setting of the compare match value for a negative-phase waveform with dead time. GPT has a dead time control function and the GTDVU register is used for setting the dead time value.

Access Size : 32 bits
Address : <GPTm_base> + 0088h + n x 0100h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TDFER	-	-	TDBDE	TDBUE	-	-	-	TDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8	TDFER	0h	RW	GTDVD Register Setting 0b: Sets GTDVU and GTDVD registers individually 1b: The value written to the GTDVU register is automatically set in the GTDVD register
7, 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	TDBDE	0h	RW	GTDVD Register Buffer Operation Enable 0b: Disable GTDVD register buffer operation 1b: Enable GTDVD register buffer operation
4	TDBUE	0h	RW	GTDVU Register Buffer Operation Enable 0b: Disable GTDVU register buffer operation 1b: Enable GTDVU register buffer operation
3 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	TDE	0h	RW	Negative-Phase Waveform Setting 0b: GTCCRB is set without using GTDVU/GTDVD 1b: GTDVU/GTDVD is used to automatically set the compare match value for negative-phase waveform with dead time in GTCCRB

TDE bit (Negative-Phase Waveform Setting)

The TDE bit specifies whether to use GTDVU/GTDVD. When GTDVU is used, the compare match value for a negative-phase waveform with dead time obtained by the compare match value of a positive-phase waveform (GTCCRA) and the dead time value (GTDVU/GTDVD) is automatically set in GTCCRB.

The TDE bit setting is ignored in saw-wave PWM mode, and the GTCCRB is not automatically set.

The GTCCRB value is automatically set and has the following upper and lower limit values. If the obtained GTCCRB value is not within the upper or lower limit, the following limit value is set in GTCCRB.

- Triangle waves:
  Upper limit value: GTPR – 1
  Lower limit value: 1 in up-counting, 0 in down-counting
- Saw-wave one-shot pulse mode:
  Upper limit value:
  GTPR Lower limit value: 0.

TDBUE bit (GTDVU Buffer Operation Enable)

The TDBUE bit enables buffer operation in combination with GTDVU and GTDBU. The buffer transfer timing is the trough for triangle waves, and an overflow or underflow for saw waves.

TDBDE bit (GTDVD Buffer Operation Enable)

The TDBDE bit enables buffer operation in combination with GTDVD and GTDBD. The buffer transfer timing is the trough for triangle waves, and an overflow or underflow for saw waves. When this bit and the TDFER bit are set to 1 simultaneously, the TDFER bit setting is given priority.

TDFER bit (GTDVD Setting)

The TDFER bits sets whether or not the value written to GTDVU is also set to GTDVD automatically.

5.7.2.2.28 General-Purpose Timer Dead Time Value Register k (GPTm_n_GTDVk) (k = U, D)

Access Size : 32 bits

Address : <GPTm_base> + 008Ch + n x 0100h (k = U)
<GPTm_base> + 0090h + n x 0100h (k = D)

Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GTDV[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GTDV[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GTDV[31:0]	FFFF_FFFFh	RW	GTDVk is a 32-bit read/write register that sets the dead time for generating PWM waveforms with dead time. GTDVU is used for up-counting and GTDVD is used for down-counting. It is prohibited to set the GTDVk register with a value larger than or equal to that of the GTPR register. Do not set the change point of waveform beyond the count cycle period, when the automatic dead time setting function is used. By reading the GTCCRB register, the change point for the reverse-phase waveform after the dead time added, which is set in the automatic dead time setting function. When GTDVk is used, writing to GTCCRB is not allowed. When this register is set to 0, waveforms without dead time are output. GTDVn must be accessed in 32-bit units. Access in 8-bit/16-bit units is prohibited. When GTDVk buffer operation is enabled, GTDBk can be written anytime. The value of GTDBk is transferred to GTDVk at the end of the count cycle period. When the GTDVn buffer operation is disabled, set the CST bit in the GTCR register to stop the GPT with before changing GTDVn to a new value. When GTDVk buffer operation is disabled, to change GTDVk to a new value, stop the GPT with the CST bit in the GTCR register.

5.7.2.2.29 General-Purpose Timer Dead Time Buffer Register k (GPTm_n_GTDBk) (k = U, D)

Access Size : 32 bits

Address : <GPTm_base> + 0094h + n x 0100h (k = U)
<GPTm_base> + 0098h + n x 0100h (k = D)

Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GTDB[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GTDB[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GTDB[31:0]	FFFF_FFFFh	RW	The GTDB register is a readable/writable register that operates as a buffer register for the GTDVm register

5.7.2.2.30 General-Purpose Timer Output Protection Function Status Register (GPTm_n_GTSOS)

GTSOS is a status register that indicates the status of the output protection function. The output protection function is enabled only when the dead time is automatically set (GTDTCR.TDE bit = 1) in triangle-wave mode.

Access Size : 32 bits

Address : <GPTm_base> + 009Ch + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	OBMON	OAMON	-	-	-	-	-	-	SOS[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	OBMON	0h	R	GTIOCnB Pin Monitoring
8	OAMON	0h	R	GTIOCnA Pin Monitoring
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	SOS[1:0]	0h	R	Output Protection Function Status 00b: Normal operation 01b: Protected state (GTCCRA register = 0b is set in transfer at troughs or crests) 10b: Protected state (GTCCRA register ≥ GTPR register is set in transfer at troughs) 11b: Protected state (GTCCRA register ≥ GTPR register is set in transfer at crests)

SOS bits (Output Protection Function Status)

The SOS[1:0] bits indicate the status of the output protection function in triangle-wave PWM mode. For details of the output protection function, see **5.7.8.4 GTIOCnm Protection Function for GTIOCnm Pin Output (n = 0 to 15; m = A, B)**.

OAMON bit (GTIOCnA Pin Monitoring)

In input capture operation, the state of input capture input on the GTIOCnA pin via the synchronization circuit is monitored. When the noise filter is enabled, the state of input capture input on the GTIOCnA pin via the noise filter is monitored. In output compare/PWM operation, the state of output compare output on the GTIOCnA pin to which output disable control has not been applied is monitored.

OBMON bit (GTIOCnB Pin Monitoring)

In input capture operation, the state of input capture input on the GTIOCnB pin via the synchronization circuit is monitored. When the noise filter is enabled, the state of input capture input on the GTIOCnB pin via the noise filter is monitored. In output compare/PWM operation, the state of output compare output on the GTIOCnB pin to which output disable control has not been applied is monitored.

5.7.2.2.31 General-Purpose Timer Output Protection Function Temporary Release Register (GPTm_n_GTSOTR)

GTSOTR temporarily releases the protected state of GTIOCnB pin output when output protection is set. The protected state can be released only when GTSOS.SOS[1:0] bits = 10b (protected state in which GTCCRA ≥ GTPR has occurred during transfer at trough). The protected state cannot be released in any other case.

Access Size : 32 bits
Address : <GPTm_base> + 00A0h + n x 0100h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SOTR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	SOTR	0h	RW	Output Protection Function Temporary Release 0b: The protected state is not released 1b: The protected state is released

SOTR bit (Output Protection Function Temporary Release)

The SOTR bit specifies whether to temporarily release the protected state of the GTIOCnB pin output in an output protected state. When the SOTR bit is set to 1b, the output protection function is canceled from the first trough. When the SOTR bit is set to 0b, output protection resumes from the first trough.

5.7.2.2.32 General-Purpose Timer Extended Interrupt Skipping Counter Control Register (GPTm_n_GTEITC)

The GTEITC register controls the skipping counter for the extended interrupt skipping function to skip interrupts, A/D converter start requests, and buffer transfers independently by counting the overflow and underflow of the GTCNT counter.

It operates independently of interrupt skipping by the GTITC register. The setting of this register has no effect during event counting.

Access Size : 32 bits

Address : <GPTm_base> + 00A8h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EITCNT2[3:0]				EITCNT2IV[3:0]				EIVTT2[3:0]				-	-	EIVTC2[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EITCNT1[3:0]				-	-	-	-	EIVTT1[3:0]				-	-	EIVTC1[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	EITCNT2[3:0]	0h	R	Extended Interrupt Skipping Counter 2
27 to 24	EITCNT2IV [3:0]	0h	RW	Extended Interrupt Skipping Counter 2 Initial Value Note: Writable only when the EIVTC2[1:0] bits are “00b” and a value other than “00b” is written to the EIVTC2[1:0] bits.
23 to 20	EIVTT2[3:0]	0h	RW	Extended Interrupt Skipping 2 Skipping Count Setting
19, 18	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
17, 16	EIVTC2[1:0]	0h	RW	Extended Interrupt Skipping Counter 2 Count Source Select 00b: Do not count (no skipping) 01b: Count both overflow/underflow in saw-wave mode and (crests) in triangle-wave mode 10b: Count both overflow/underflow in saw-wave mode and (troughs) in triangle-wave mode 11b: Counts both overflows/underflows in saw-wave mode and both (troughs/crests) in triangle-wave mode
15 to 12	EITCNT1[3:0]	0h	R	Extended Interrupt Skipping Counter 1
11 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 4	EIVTT1[3:0]	0h	RW	Extended Interrupt Skipping 1 Skipping Count Setting
3, 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1, 0	EIVTC1[1:0]	0h	RW	Extended Interrupt Skipping Counter 1 Count Source Select 00b: Do not count (no skipping) 01b: Count both overflow/underflow in saw-wave mode and (crests) in triangle-wave mode 10b: Count both overflow/underflow in saw-wave mode and (troughs) in triangle-wave mode 11b: Counts both overflows/underflows in saw-wave mode and both (troughs/crests) in triangle-wave mode

EIVTCk[1:0] bits (Extended Interrupt Skipping Counter k Count Source Select) (k = 1, 2)

These bits select the way of counting by extended interrupt skipping counter k.

Only setting these bits does not skip the interrupts, A/D converter start requests, and buffer transfers. Use the GTEITLI1, GTEITLI2, and GTEITLB registers to set the skipping function for the interrupt, A/D converter start request, and buffer transfer individually as targets for skipping.

EIVTTk[3:0] bits (Extended Interrupt Skipping k Skipping Count Setting) (k = 1, 2)

These bits set the count for the period for continuous skipping as the skipping count with generation of the count source selected by the EIVTCk[1:0] bits as an interval.

When the count source is generated while the value of the EIVTTk[3:0] bits matches the EITCNTk[3:0] value, the EITCNTk[3:0] bits return to 0b.

When these bits are 0000b, skipping is not performed.

EITCNT1[3:0] bits (Extended Interrupt Skipping Counter 1)

This counter counts up by 1 every time the count source (overflow/underflow/crest/trough) selected by the EIVTC1[1:0] bits is generated.

Counting is performed periodically within the range between 0 and the EIVTT1[3:0] bits.

The value is not cleared even when the GTCNT counter is stopped, and the value when the GTCNT counter is stopped is retained.

[Clearing conditions]

- 00b is written to the EIVTC1[1:0] bits.
- 0000b is written to the EIVTT1[3:0] bits.
- The count source (overflow/underflow/crest/trough) selected by the EIVTC1[1:0] bits is generated while the extended interrupt skipping 1 skipping count set by the EIVTT1[3:0] bits matches the value of the EITCNT1[3:0].

EITCNT2IV[3:0] bits (Extended Interrupt Skipping Counter 2 Initial Value)

The value of these bits is the initial value of extended interrupt skipping counter 2.

Writing to the EITCNT2[3:0] bits only proceeds when the GTEITC register is written by access in 32-bit units and the value written to the EIVTC2[1:0] bits is other than 00b while the setting of the EITCNT2[3:0] bits is for no counting (EIVTC2[1:0] bits are 00b). When the EITCNT2IV[3:0] bits are written, the value written to the EITCNT2IV[3:0] bits is also written to the EITCNT2[3:0] bits simultaneously.

While the setting of the EITCNT2[3:0] bits is for counting (the EIVTC2[1:0] bits are other than 00b) or when these bits are set for no counting (00b is written to the EIVTC2[1:0] bits), writing to the EITCNT2IV[3:0] bits is ignored.

The EITCNT2IV[3:0] bits are not reset by writing 00b to the EIVTC2[1:0] bits.

EITCNT2[3:0] bits (Extended Interrupt Skipping Counter 2)

This counter counts up by 1 every time the count source (overflow/underflow/crest/trough) selected by the EIVTC2[1:0] bits is generated.

Counting is performed periodically within the range between 0 and the EIVTT2[3:0] bits.

The value is not cleared even when the GTCNT counter is stopped, and the value when the GTCNT counter is stopped is retained.

The initial value of the EITCNT2[3:0] bits is only set when the GTEITC register is written by access to the higher-order 16 or 32 bits and the value written to the EIVTC2[1:0] bits is other than 00b while the setting of extended interrupt skipping counter 2 is for no counting (the EIVTC2[1:0] bits are 00b).

When the initial value is written, the value written to the EITCNT2IV[3:0] bits is written to the EITCNT2[3:0] bits as the initial value.

[Clearing conditions]

- 00b is written to the EIVTC2[1:0] bits.

- 0000b is written to the EIVTT2[3:0] bits.
- The value other than 00b is written to the EIVTC2[1:0] bits and 0000b is written to the EITCNT2IV[3:0] bits simultaneously while the setting of the EIVTC2[1:0] bits is 00b.
- The count source (overflow/underflow/crest/trough) selected by the EIVTC2[1:0] bits is generated when the extended interrupt skipping 2 skipping count set by the EIVTT2[3:0] bits matches the value of the EITCNT2[3:0] bits.

5.7.2.2.33 General-Purpose Timer Extended Interrupt Skipping Setting Register 1 (GPTm_n_GTEITLI1)

The GTEITLI1 register sets the extended skipping function for the interrupt by compare match/input capture, overflow, and underflow. When the extended interrupt skipping function is set, a change in the status flag is also skipped.

Only the setting of this register does not lead to skipping. Set the GTEITC register such that the corresponding extended interrupt skipping counter operates counting.

It operates independently of interrupt skipping by the GTITC register. The setting of this register has no effect during event counting.

Access Size : 32 bits

Address : <GPTm_base> + 00ACh + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EITLPU	EITLU[2:0]			EITLPV	EITLV[2:0]			EITLPF	EITLF[2:0]			EITLPE	EITLE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EITLPD	EITLD[2:0]			EITLPC	EITLC[2:0]			EITLPB	EITLB[2:0]			EITLPA	EITLA[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	EITLPU	0h	RW	Underflow Interrupt Extended Skipping Period Select 0b: OR of the underflow interrupt skipping period set by the GTITC.IVTC[1:0] bits and the extended interrupt skipping period set by the EITLU[2:0] bits 1b: AND of the underflow interrupt skipping period set by the GTITC.IVTC[1:0] bits and the extended interrupt skipping period set by the EITLU[2:0] bits
30 to 28	EITLU[2:0]	0h	RW	Underflow Interrupt Extended Skipping Function Select See Table **5.7-9 Settings of GTEITLI1 Register Function Select Bits**
27	EITLPV	0h	RW	Overflow Interrupt Extended Skipping Period Select 0b: OR of the overflow interrupt skipping period set by the GTITC.IVTC[1:0] bits and the extended interrupt skipping period set by the EITLV[2:0] bits 1b: AND of the overflow interrupt skipping period set by the GTITC.IVTC[1:0] bits and the extended interrupt skipping period set by the EITLV[2:0] bits
26 to 24	EITLV[2:0]	0h	RW	Overflow Interrupt Extended Skipping Function Select See **Table 5.7-9 Settings of GTEITLI1 Register Function Select Bits**
23	EITLPF	0h	RW	GTCCRF Register Compare Match Interrupt Extended Skipping Period Select 0b: OR of the skipping period by the GTITC.ITLF bit and the extended interrupt skipping period by the EITLF[2:0] bits 1b: AND of the skipping period set by the GTITC.ITLF bit and the extended interrupt skipping period set by the EITLF[2:0] bits
22 to 20	EITLF[2:0]	0h	RW	GTCCRF Register Compare Match Interrupt Extended Skipping Function Select See **Table 5.7-9 Settings of GTEITLI1 Register Function Select Bits**
19	EITLPE	0h	RW	GTCCRE Register Compare Match Interrupt Extended Skipping Period Select 0b: OR of the skipping period by the GTITC.ITLE bit and the extended interrupt skipping period by the EITLE[2:0] bits 1b: AND of the skipping period set by the GTITC.ITLE bit and the extended interrupt skipping period set by the EITLE[2:0] bits
18 to 16	EITLE[2:0]	0h	RW	GTCCRE Register Compare Match Interrupt Extended Skipping Function Select See **Table 5.7-9 Settings of GTEITLI1 Register Function Select Bits**
15	EITLPD	0h	RW	GTCCRD Register Compare Match Interrupt Extended Skipping Period Select 0b: OR of the skipping period by the GTITC.ITLD bit and the extended interrupt skipping period by the EITLD[2:0] bits 1b: AND of the skipping period set by the GTITC.ITLD bit and the extended interrupt skipping period set by the EITLD[2:0] bits

Bit	Bit Name	Initial Value	R/W	Description
14 to 12	EITLD[2:0]	0h	RW	GTCCRD Register Compare Match Interrupt Extended Skipping Function Select See **Table 5.7-9 Settings of GTEITLI1 Register Function Select Bits**
11	EITLPC	0h	RW	GTCCRC Register Compare Match Interrupt Extended Skipping Period Select 0b: OR of the skipping period by the GTITC.ITLC bit and the extended interrupt skipping period by the EITLC[2:0] bits 1b: AND of the skipping period set by the GTITC.ITLC bit and the extended interrupt skipping period set by the EITLC[2:0] bits
10 to 8	EITLC[2:0]	0h	RW	GTCCRC Register Compare Match Interrupt Extended Skipping Function Select See **Table 5.7-9 Settings of GTEITLI1 Register Function Select Bits**
7	EITLPB	0h	RW	GTCCRB Register Compare Match/Input Capture Interrupt Extended Skipping Period Select 0b: OR of the skipping period by the GTITC.ITLB bit and the extended interrupt skipping period by the EITLB[2:0] bits 1b: AND of the skipping period set by the GTITC.ITLB bit and the extended interrupt skipping period set by the EITLB[2:0] bits
6 to 4	EITLB[2:0]	0h	RW	GTCCRB Register Compare Match/Input Capture Interrupt Extended Skipping Function Select See **Table 5.7-9 Settings of GTEITLI1 Register Function Select Bits**
3	EITLPA	0h	RW	GTCCRA Register Compare Match/Input Capture Interrupt Extended Skipping Period Select 0b: OR of the skipping period by the GTITC.ITLA bit and the extended interrupt skipping period by the EITLA[2:0] bits 1b: AND of the skipping period set by the GTITC.ITLA bit and the extended interrupt skipping period set by the EITLA[2:0] bits
2 to 0	EITLA[2:0]	0h	RW	GTCCRA Register Compare Match/Input Capture Interrupt Extended Skipping Function Select See **Table 5.7-9 Settings of GTEITLI1 Register Function Select Bits**

EITLm[2:0] bits (GTCCRm Register Compare Match/Input Capture Interrupt Extended Skipping Function Select) (m = A, B)

These bits select the extended interrupt skipping function to skip the compare match/input capture interrupt (GTCIm) from the GTCCRm register. For details, see **Table 5.7-9**.

EITLPm bit (GTCCRm Register Compare Match/Input Capture Interrupt Extended Skipping Period Select) (m = A, B)

This bit selects the skipping period when skipping by the GTITC.ITLm bit and skipping by the EITLm[2:0] bits are performed simultaneously in response to the compare match/input capture interrupt (GTCIm) from the GTCCRm register.

EITLx[2:0] bits (GTCCRx Register Compare Match Interrupt Extended Skipping Function Select) (x = C, D, E, F)

These bits select the extended interrupt skipping function to skip compare match interrupts (GTCIx) from the GTCCRx register. For details, see **Table 5.7-9**.

EITLPx bit (GTCCRx Register Compare Match Interrupt Extended Skipping Period Select) (x= C, D, E, F)

This bit selects the skipping period when skipping by the GTITC.ITLx bit and skipping by the EITLx[2:0] bits are performed simultaneously in response to the compare match interrupt (GTCIx) from the GTCCRx register.

EITLV[2:0] bits (Overflow Interrupt Extended Skipping Function Select)

These bits select the extended interrupt skipping function to skip overflow interrupts. For details, see **Table 5.7-9**.

EITLPV bit (Overflow Interrupt Extended Skipping Period Select)

This bit selects the skipping period when overflow interrupt skipping by the GTITC.IVTC[1:0] bits and skipping by the EITLV[2:0] bits are performed simultaneously in response to the overflow interrupt (GTCIV).

EITLU[2:0] bits (Underflow Interrupt Extended Skipping Function Select)

These bits select the extended interrupt skipping function to skip underflow interrupts. For details, see **Table 5.7-9**.

EITLPU bit (Underflow Interrupt Extended Skipping Period Select)

This bit selects the skipping period when underflow interrupt skipping by the GTITC.IVTC[1:0] bits and skipping by the EITLV[2:0] bits are performed simultaneously in response to the underflow interrupt (GTCIU).

Table 5.7-9 Settings of GTEITLI1 Register Function Select Bits

EITLy[2:0]	Function
000b	Extended interrupt skipping is not performed.
001b	Interrupts are skipped for the period in which the value of extended interrupt skipping counter 1 is other than 0 (interrupts are output for the period in which EITCNT1[3:0] bits = 0).
010b	Interrupts are skipped for the period in which the value of extended interrupt skipping counter 2 is other than 0 (interrupts are output for the period in which EITCNT2[3:0] bits = 0).
011b	Interrupts are skipped for the period in which the value of extended interrupt skipping counter 1 or 2 is other than 0 (interrupts are output for the period in which EITCNT1[3:0] bits = EITCNT2[3:0] bits = 0).
100b	Reserved
101b	Interrupts are skipped for the period in which the value of extended interrupt skipping counter 1 is other than the skipping count (interrupts are output for the period in which EITCNT1[3:0] bits = EIVTT1[3:0] bits).
110b	Interrupts are skipped for the period in which the value of extended interrupt skipping counter 2 is other than the skipping count (interrupts are output for the period in which EITCNT2[3:0] bits = EIVTT2[3:0] bits).
111b	Interrupts are skipped for the period in which the value of extended interrupt skipping counter 1 or 2 is other than the skipping count (interrupts are output for the period in which EITCNT1[3:0] bits = EIVTT1[3:0] bits and EITCNT2[3:0] bits = EIVTT2[3:0] bits).

Remarks: y = A, B, C, D, E, F, V, U

Note: Skipping is not performed if the skipping counter to be used is set for no counting (EIVTCk[1:0] bits = 00b or EIVTTk[3:0] bits = 0000b: k = 1, 2).

Note: Skipping is not performed when the EITLy[2:0] bits are set to 011b or 111b, or when either of the skipping counters is set for no counting.

5.7.2.2.34 General-Purpose Timer Extended Interrupt Skipping Setting Register 2 (GPTm_n_GTEITLI2)

The GTEITLI2 register selects the extended skipping function for A/D converter start requests. When the extended interrupt skipping function is set, a change in the status flag is also skipped.

Only the setting of this register does not lead to skipping. Set the GTEITC register such that the corresponding extended interrupt skipping counter operates counting.

It operates independently of interrupt skipping by the GTITC register. The setting of this register has no effect during event counting.

Access Size : 32 bits

Address : <GPTm_base> + 00B0h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	EADTPBL	EADTBL[2:0]			EADTPAL	EADTAL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7	EADTPBL	0h	RW	GTADTRB Register Compare Match/Input Capture Interrupt Extended Skipping Period Select 0b: OR of the skipping period set by the GTITC.ADTBL bit and the extended interrupt skipping period set by the EADTBL[2:0] bits 1b: AND of the skipping period set by the GTITC.ADTBL bit and the extended interrupt skipping period set by the EADTBL[2:0] bits
6 to 4	EADTBL[2:0]	0h	RW	GTADTRB Register A/D Converter Start Request Extended Skipping Function Select See **Table 5.7-10 Settings of GTEITLI2 Register Function Select Bits**
3	EADTPAL	0h	RW	GTADTRA Register A/D Converter Start Request Extended Skipping Period Select 0b: OR of the skipping period by the GTITC.ADTAL bit and the extended interrupt skipping period by the EADTAL[2:0] bits 1b: AND of the skipping period set by the GTITC.ADTAL bit and the extended interrupt skipping period set by the EADTAL[2:0] bits
2 to 0	EADTAL[2:0]	0h	RW	GTADTRA Register A/D Converter Start Request Extended Skipping Function Select See **Table 5.7-10 Settings of GTEITLI2 Register Function Select Bits**

EADTmL[2:0] bits (GTADTRm Register A/D Converter Start Request Extended Skipping Function Select) (m = A, B)

These bits select the extended interrupt skipping function to skip A/D converter start requests on compare match with the GTADTRm register. For details, see **Table 5.7-10**.

EADTPmL bit (GTADTRm Register A/D Converter Start Request Extended Skipping Period Select) (m = A, B)

This bit selects the skipping period when skipping by the GTITC.ADTmL bit and skipping by the EADTmL[2:0] bits are performed simultaneously in response to the A/D converter start request on compare match with the GTADTRm register.

Table 5.7-10 Settings of GTEITLI2 Register Function Select Bits

EADTmL[2:0]	Function
000b	Extended interrupt skipping is not performed.
001b	A/D converter start requests are skipped for the period in which the value of extended interrupt skipping counter 1 is other than 0 (A/D converter start requests are output for the period in which EITCNT1[3:0] bits = 0).
010b	A/D converter start requests are skipped for the period in which the value of extended interrupt skipping counter 2 is other than 0 (A/D converter start requests are output for the period in which EITCNT2[3:0] bits = 0).
011b	A/D converter start requests are skipped for the period in which the value of extended interrupt skipping counter 1 or 2 is other than 0 (A/D converter start requests are output for the period in which EITCNT1[3:0] bits = 0 and EITCNT2[3:0] bits = 0).
100b	Reserved
101b	A/D converter start requests are skipped for the period in which the value of extended interrupt skipping counter 1 is other than the skipping count (A/D converter start requests are output for the period in which EITCNT1[3:0] bits = EIVTT1[3:0] bits).
110b	A/D converter start requests are skipped for the period in which the value of extended interrupt skipping counter 2 is other than the skipping count (A/D converter start requests are output for the period in which EITCNT2[3:0] bits = EIVTT2[3:0] bits).
111b	A/D converter start requests are skipped for the period in which the value of extended interrupt skipping counter 1 or 2 is other than the skipping count (A/D converter start requests are output for the period in which EITCNT1[3:0] bits = EIVTT1[3:0] bits and EITCNT2[3:0] bits = EIVTT2[3:0] bits).

Remarks: m = A, B

Note: Skipping is not performed if the skipping counter to be used is set for no counting (EIVTCk[1:0] bits = 00b or EIVTTk[3:0] bits = 0000b: k = 1, 2).

Note: Skipping is not performed when the EADTmL[2:0] bits are set to 011b or 111b, or when either of the skipping counters is set for no counting.

5.7.2.2.35 General-Purpose Timer Extended Buffer Transfer Skipping Setting Register (GPTm_n_GTEITLB)

The GTEITLB register selects the extended skipping function for buffer transfer.

Only the setting of this register does not lead to skipping. Set the GTEITC register such that the corresponding extended interrupt skipping counter operates counting.

It operates independently of interrupt skipping by the GTITC register. The setting of this register has no effect during event counting.

Access Size : 32 bits

Address : <GPTm_base> + 00B4h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	EBTLDVD[2:0]			-	EBTLDVU[2:0]			-	EBTLADB[2:0]			-	EBTLADA[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	EBTLPR[2:0]			-	EBTLCB[2:0]			-	EBTLCA[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
30 to 28	EBTLDVD[2:0]	0h	RW	GTDVD Register Buffer Transfer Extended Skipping Function Select See **Table 5.7-11 Settings of GTEITLB Register Function Select Bits**
27	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
26 to 24	EBTLDVU[2:0]	0h	RW	GTDVU Register Buffer Transfer Extended Skipping Function Select See **Table 5.7-11 Settings of GTEITLB Register Function Select Bits**
23	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
22 to 20	EBTLADB[2:0]	0h	RW	GTADTRB Register Buffer Transfer Extended Skipping Function Select See **Table 5.7-11 Settings of GTEITLB Register Function Select Bits**
19	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18 to 16	EBTLADA[2:0]	0h	RW	GTADTRA Register Buffer Transfer Extended Skipping Function Select See **Table 5.7-11 Settings of GTEITLB Register Function Select Bits**
15 to 11	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10 to 8	EBTLPR[2:0]	0h	RW	GTPR Register Buffer Transfer Extended Skipping Function Select See **Table 5.7-11 Settings of GTEITLB Register Function Select Bits**
7	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6 to 4	EBTLCB[2:0]	0h	RW	GTCCRB Register Buffer Transfer Extended Skipping Function Select See **Table 5.7-11 Settings of GTEITLB Register Function Select Bits**
3	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	EBTLCA[2:0]	0h	RW	GTCCRA Register Buffer Transfer Extended Skipping Function Select See **Table 5.7-11 Settings of GTEITLB Register Function Select Bits**

EBTLCA[2:0] bits (GTCCRA Register Buffer Transfer Extended Skipping Function Select)

These bits select the extended buffer transfer skipping function to skip buffer transfer from the GTCCRA register (transfer from the GTCCRA, GTCCRC, or GTCCRD register to temporary register A). For details, see **Table 5.7-11**.

Extended skipping of buffer transfer from the GTCCRA register is also valid for forcible buffer transfer by the GTBER.CCRSWT bit while counting is stopped. When forcible buffer transfer from the GTCCRA register is to be performed, buffer transfer should be performed with no extended skipping of buffer transfer.

EBTLCB[2:0] bits (GTCCRB Register Buffer Transfer Extended Skipping Function Select)

These bits select the extended buffer transfer skipping function to skip buffer transfer from the GTCCRB register (transfer from the GTCCRB, GTCCRE, or GTCCRF register to temporary register B). For details, see **Table 5.7-11**.

Extended skipping of buffer transfer from the GTCCRB register is also valid for forcible buffer transfer by the GTBER.CCRSWT bit while counting is stopped. When forcible buffer transfer from the GTCCRB register is to be performed, buffer transfer should be performed with no extended skipping of buffer transfer.

EBTLPR[2:0] bits (GTPR Register Buffer Transfer Extended Skipping Function Select)

These bits select the extended buffer transfer skipping function to skip buffer transfer from the GTPR register (transfer from the GTPR register to the GTPBR or GTPDBR register). For details, see **Table 5.7-11**.

EBTLADm[2:0] bits (GTADTRm Register Buffer Transfer Extended Skipping Function Select) (m = A, B)

These bits select the extended buffer transfer skipping function to skip buffer transfer from the GTADTRm register (transfer from the GTADTRm register to the GTADTBRm or GTADTDBRm register). For details, see **Table 5.7-11**.

EBTLDVm[2:0] bits (GTDVm Register Buffer Transfer Extended Skipping Function Select) (m = U, D)

These bits select the extended buffer transfer skipping function to skip buffer transfer from the GTDVm register (transfer from the GTDVm register to the GTDBm register). For details, see **Table 5.7-11**.

Table 5.7-11 Settings of GTEITLB Register Function Select Bits

EBTLx[2:0]	Function
000b	Extended buffer transfer skipping is not performed.
001b	Buffer transfer is skipped for the period in which the value of extended interrupt skipping counter 1 is other than 0 (buffer transfer is performed for the period in which EITCNT1[3:0] bits = 0).
010b	Buffer transfer is skipped for the period in which the value of extended interrupt skipping counter 2 is other than 0 (buffer transfer is performed for the period in which EITCNT2[3:0] bits = 0).
011b	Buffer transfer is skipped for the period in which the value of extended interrupt skipping counter 1 or 2 is other than 0 (buffer transfer is performed for the period in which EITCNT1[3:0] bits = EITCNT2[3:0] bits = 0).
100b	Reserved
101b	Buffer transfer is skipped for the period in which the value of extended interrupt skipping counter 1 is other than the skipping count (buffer transfer is performed for the period in which EITCNT1[3:0] bits = EIVTT1[3:0] bits).
110b	Buffer transfer is skipped for the period in which the value of extended interrupt skipping counter 2 is other than the skipping count (buffer transfer is performed for the period in which EITCNT2[3:0] bits = EIVTT2[3:0] bits).
111b	Buffer transfer is skipped for the period in which the value of extended interrupt skipping counter 1 or 2 is other than the skipping count (buffer transfer is performed for the period in which EITCNT1[3:0] bits = EIVTT1[3:0] bits and EITCNT2[3:0] bits = EIVTT2[3:0] bits).

Remarks: x = CA, CB, PR, ADA, ADB, DVU, DVD

Note: Skipping is not performed if the skipping counter to be used is set for no counting (EIVTCk[1:0] bits = 00b or EIVTTk[3:0] bits = 0000b: k = 1, 2).

Note: Skipping is not performed when the EBTLx[2:0] bits are set to 011b or 111b, or when either of the skipping counters is set for no counting.

5.7.2.2.36 General-Purpose Timer Inter Channel Logical Operation Function Setting Register (GPTm_n_GTICLF)

The GTICLF register sets the logical operation function between compare match outputs. The logical operation is performed with the signals that the duty cycle 0%/100% control is performed after compare match control. (The output disable control is performed with the signal after logical operation.)

Access in 8-bit units to GTICLF is prohibited.

Access Size : 32 bits

Address : <GPTm_base> + 00B8h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	ICLFSELD[5:0]						-	ICLFB[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	ICLFSELC[5:0]						-	ICLFA[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
25 to 20	ICLFSELD [5:0]	0h	RW	Inter Channel Signal D Select*1,*3 Refer to **Table 5.7-12**.
19	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18 to 16	ICLFB[2:0]	0h	RW	GTIOCnB Output Logical Operation Function Select 000b: B (no delay) 001b: NOT B (no delay) 010b: D (1 PCLKD delay) 011b: NOT D (1 PCLKD delay) 100b: B AND D (1 PCLKD delay)*3 101b: B OR D (1 PCLKD delay)*3 110b: B EXOR D (1 PCLKD delay)*3 111b: B NOR D (1 PCLKD delay)*3
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9 to 4	ICLFSELC [5:0]	0h	RW	Inter Channel Signal C Select*1,*2 Refer to **Table 5.7-12**.
3	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	ICLFA[2:0]	0h	RW	GTIOCnA Output Logical Operation Function Select 000b: A (no delay) 001b: NOT A (no delay) 010b: C (1 PCLKD delay) 011b: NOT C (1 PCLKD delay) 100b: A AND C (1 PCLKD delay)*2 101b: A OR C (1 PCLKD delay)*2 110b: A EXOR C (1 PCLKD delay)*2 111b: A NOR C (1 PCLKD delay)*2

Note 1. The signal before performing output disable control is selected.

Note 2. When channel's own GTIOCnA is selected, C is treated as "1".

Note 3. When channel's own GTIOCnB is selected, D is treated as "1".

ICLFm[2:0] bits (GTIOCnm Output Logical Operation Function Select) (m = A, B)

These bits select the logical operation function between signals before performing output disable control for GTIOCnm. To prevent hazard to the GPT output, the signal after logical operation is latched with PCLKD. After latching, the output disable control is performed. When the logical operation function which causes the delay of 1 PCLKD is selected, the output enable signal is also delayed with 1 PCLKD and input to the output disable control.

When the same signal to operate logical function AND, OR, EXOR and NOR is selected, one signal is treated as “1”.

ICLFSELk[5:0] bits (Inter Channel Signal k Select) (k = C, D)

These bits select the signal k to which the logical operation is performed with the signal before performing output disable control for GTIOCnm.

The pins to be selected differ between GPT0n and GPT1n. For details, see the table below.

Table 5.7-12 ICLFSELk Bit Settings and Corresponding Pins

Setting	GPT0n Select Pin	GPT1n Select Pin
00_0000b	GTIOC0A	GTIOC8A
00_0001b	GTIOC0B	GTIOC8B
00_0010b	GTIOC1A	GTIOC9A
00_0011b	GTIOC1B	GTIOC9B
00_0100b	GTIOC2A	GTIOC10A
00_0101b	GTIOC2B	GTIOC10B
00_0110b	GTIOC3A	GTIOC11A
00_0111b	GTIOC3B	GTIOC11B
00_1000b	GTIOC4A	GTIOC12A
00_1001b	GTIOC4B	GTIOC12B
00_1010b	GTIOC5A	GTIOC13A
00_1011b	GTIOC5B	GTIOC13B
00_1100b	GTIOC6A	GTIOC14A
00_1101b	GTIOC6B	GTIOC14B
00_1110b	GTIOC7A	GTIOC15A
00_1111b	GTIOC7B	GTIOC15B
01_0000b	GTIOC8A	GTIOC0A
01_0001b	GTIOC8B	GTIOC0B
01_0010b	GTIOC9A	GTIOC1A
01_0011b	GTIOC9B	GTIOC1B
01_0100b	GTIOC10A	GTIOC2A
01_0101b	GTIOC10B	GTIOC2B
01_0110b	GTIOC11A	GTIOC3A
01_0111b	GTIOC11B	GTIOC3B
01_1000b	GTIOC12A	GTIOC4A
01_1001b	GTIOC12B	GTIOC4B
01_1010b	GTIOC13A	GTIOC5A
01_1011b	GTIOC13B	GTIOC5B
01_1100b	GTIOC14A	GTIOC6A
01_1101b	GTIOC14B	GTIOC6B
01_1110b	GTIOC15A	GTIOC7A
01_1111b	GTIOC15B	GTIOC7B

5.7.2.2.37 General-Purpose Timer Operation Enable Bit Simultaneous Control Channel Select Register (GPTm_n_GTSECSR)

The GTSECSR register selects desired channel n (n = 0 to 15) for updating an operation enable bit by the GTSECR register. A bit position for the GTSECSR register indicates a channel number. The GTSECSR register of each channel is a common register, and writing 1b to a bit in the GTSECSR register in any channel and updating it changes a channel, related to the position of the bit written with 1 by the GTSECSR register, to be simultaneously controlled of the operation enable bit by the GTSECR register.

Access Size : 32 bits

Address : <GPTm_base> + 00D0h + n x 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SECSEL15	SECSEL14	SECSEL13	SECSEL12	SECSEL11	SECSEL10	SECSEL9	SECSEL8	SECSEL7	SECSEL6	SECSEL5	SECSEL4	SECSEL3	SECSEL2	SECSEL1	SECSEL0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15	SECSEL15	0h	R	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
14	SECSEL14	0h	R	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
13	SECSEL13	0h	R	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
12	SECSEL12	0h	R	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
11	SECSEL11	0h	R	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
10	SECSEL10	0h	R	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
9	SECSEL9	0h	R	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
8	SECSEL8	0h	R	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
7	SECSEL7	0h	RW	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
6	SECSEL6	0h	RW	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
5	SECSEL5	0h	RW	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control

Bit	Bit Name	Initial Value	R/W	Description
4	SECSEL4	0h	RW	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
3	SECSEL3	0h	RW	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
2	SECSEL2	0h	RW	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
1	SECSEL1	0h	RW	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control
0	SECSEL0	0h	RW	GPTmn Operation Enable Bit Simultaneous Control Channel Select 0b: Disable simultaneous control 1b: Enable simultaneous control

SECSELn bit (Operation Enable Bit Simultaneous Control Channel Select) (n = 0 to 15)

This bit enables or disables simultaneous control of operation enable in channel n.

Simultaneous control is enabled when the bit is set to 1b and disabled when the bit is 0b.

Note that the definitions of the bits differ between GPT0n and GPT1n. For details, see the table below.

Table 5.7-13 Definitions of Timer Operation Enable Bit Simultaneous Control Channel Select Bits

	GPT0		GPT1	
Bit	Bit Description	Bit Attribute	Bit Description	Bit Attribute
0	GPT00 Simultaneous Control Enable	RW	GPT10 Simultaneous Control Enable	RW
1	GPT01 Simultaneous Control Enable	RW	GPT11 Simultaneous Control Enable	RW
2	GPT02 Simultaneous Control Enable	RW	GPT12 Simultaneous Control Enable	RW
3	GPT03 Simultaneous Control Enable	RW	GPT13 Simultaneous Control Enable	RW
4	GPT04 Simultaneous Control Enable	RW	GPT14 Simultaneous Control Enable	RW
5	GPT05 Simultaneous Control Enable	RW	GPT15 Simultaneous Control Enable	RW
6	GPT06 Simultaneous Control Enable	RW	GPT16 Simultaneous Control Enable	RW
7	GPT07 Simultaneous Control Enable	RW	GPT17 Simultaneous Control Enable	RW
8	GPT10 Simultaneous Control Enable	R	GPT00 Simultaneous Control Enable	R
9	GPT11 Simultaneous Control Enable	R	GPT01 Simultaneous Control Enable	R
10	GPT12 Simultaneous Control Enable	R	GPT02 Simultaneous Control Enable	R
11	GPT13 Simultaneous Control Enable	R	GPT03 Simultaneous Control Enable	R
12	GPT14 Simultaneous Control Enable	R	GPT04 Simultaneous Control Enable	R
13	GPT15 Simultaneous Control Enable	R	GPT05 Simultaneous Control Enable	R
14	GPT16 Simultaneous Control Enable	R	GPT06 Simultaneous Control Enable	R
15	GPT17 Simultaneous Control Enable	R	GPT07 Simultaneous Control Enable	R
31 - 16	Reserved	RW	Reserved	RW

5.7.2.2.38 General-Purpose Timer Operation Enable Bit Simultaneous Control Register (GPTm_n_GTSECR)

The GTSECR register simultaneously updates the value for operation enable bits of a channel set by the GTSECSR register.

Writing 1b to a bit in the GTSECR register in any channel and updating it updates an operation enable bit for all channels, related to the position of the bit written with 1b by the all GTSECSR registers.

Setting enable and disable bits for the same operation enable bit to 1b in the GTSECR is prohibited.

A bit to which 1b has been written is automatically cleared. When the GTSECR is read, 0b is read.

Access Size : 32 bits
Address : <GPTm_base> + 00D4h + n x 0100h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	SBDPD	SBDCD	-	-	-	-	-	-	SBDPE	SBDCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	SBDPD	0h	RW	GTPR Register Buffer Operation Simultaneous Disable 0b: Disable simultaneous disabling of GTPR buffer operations 1b: Disable GTPR register buffer operations simultaneously
8	SBDCD	0h	RW	GTCCR Register Buffer Operation Simultaneous Disable 0b: Disable simultaneous disabling of GTCCR buffer operations 1b: Disable GTCCR register buffer operations simultaneously
7 to 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	SBDPE	0h	RW	GTPR Register Buffer Operation Simultaneous Enable 0b: Disable simultaneous enabling of GTPR buffer operations 1b: Enable GTPR register buffer operations simultaneously
0	SBDCE	0h	RW	GTCCR Register Buffer Operation Simultaneous Enable 0b: Disable simultaneous enabling of GTCCR buffer operations 1b: Enable GTCCR register buffer operations simultaneously

SBDCE bit (GTCCR Register Buffer Operation Simultaneous Enable)

When 1b is written to this bit, 0b is simultaneously set to a GTBER.BD[0] bit in the channels, which has been set to 1b by the GTSECSR register, and buffer operations using the GTCCRA, GTCCRC, and GTCCRD registers and using the GTCCRB, GTCCRE, and GTCCRF registers are enabled.

Simultaneous setting of the SBDCE and SBDCD bits to 1b is prohibited.

SBDPE bit (GTPR Register Buffer Operation Simultaneous Enable)

When 1b is written to this bit, 0b is simultaneously set to a GTBER.BD[1] bit in the channels, which has been set to 1b by the GTSECSR register, and buffer operations using the GTPR and GTPBR registers are enabled.

Simultaneous setting of the SBDPE and SBDPD bits to 1b is prohibited.

SBDCD bit (GTCCR Register Buffer Operation Simultaneous Disable)

When 1b is written to this bit, 1b is simultaneously set to a GTBER.BD[0] bit in the channels, which has been set to 1b by the GTSECSR register, and buffer operations using the GTCCRA, GTCCRC, and GTCCRD registers and using the GTCCRB, GTCCRE, and GTCCRF registers are disabled.

Simultaneous setting of the SBDCE and SBDCD bits to 1b is prohibited.

SBDPD bit (GTPR Register Buffer Operation Simultaneous Disable)

When 1b is written to this bit, 1b is simultaneously set to a GTBER.BD[1] bit in the channels, which has been set to 1b by the GTSECSR register, and buffer operations using the GTPR and GTPBR registers are disabled.

Simultaneous setting of the SBDPE and SBDPD bits to 1b is prohibited.

5.7.3 Operation

5.7.3.1 Basic Operation

A timer in each channel performs periodic count operation using the count clock and hardware sources. The counting function provides both up-counting and down-counting. The GTPR controls the counting cycle.

When the GTCNT counter value matches the value in GTCCRA or GTCCRB, the output from the corresponding GTIOCnA or GTIOCnB can be changed (n = 0 to 15). GTCCRA or GTCCRB can be used as an input capture register with hardware sources.

GTCCRC and GTCCRD can function as buffer registers for GTCCRA. GTCCRE and GTCCRF can function as buffer registers for GTCCRB.

5.7.3.1.1 Counter Operation

(1) Counter Starting and Stopping

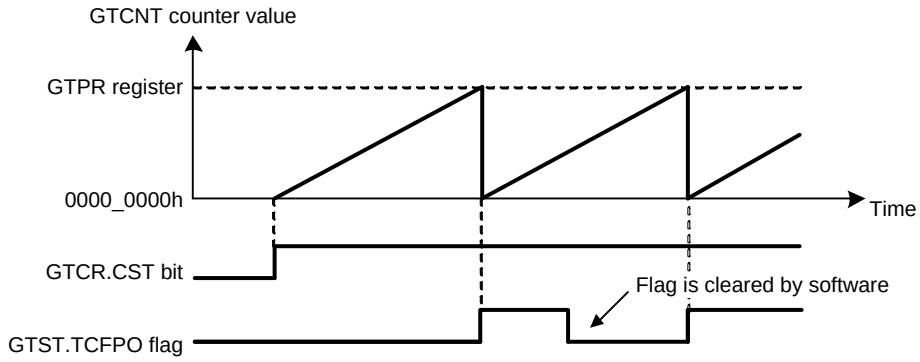
The counter of each channel starts counting when the GTCR.CST bit is set to 1b, and stops counting when the bit is set to 0b. The GTCR.CST bit value is changed by any of the following sources:

- Writing to the GTCR register
- Writing 1b to the bit in GTSTR corresponding to the GPT channel number when the GTSSR.CSTRT bit is 1b
- Writing 1b to the bit in GTSTP corresponding to the GPT channel number when the GTPSR.CSTOP bit is 1b
- The hardware source selected in the GTSSR register
- The hardware source selected in the GTPSR register.

(2) Periodic Count Operation in Up-counting by Count Clock

The GTCNT counter of each channel starts up-counting when the corresponding GTCR.CST bit is set to 1b with GTUPSR and GTDNSR registers set to 0000_0000h. When the GTCNT value changes from the GTPR value to 0 (overflow), the GTST.TCFPO flag is set to 1b, and the overflow interrupt (GPT_Ux_gpt_gtciv_n_m) is also generated. After GTCNT overflows, up-counting resumes from 0000_0000h.

Figure 5.7-3 shows an example of periodic count operation in up-counting by the count clock.



Note: The base addresses of the registers are the same.

Figure 5.7-3 Example of Periodic Count Operation in Up-Counting by the Count Clock

Table 5.7-14 shows an example for setting periodic count operation in up-counting by the count clock.

Table 5.7-14 Example for Setting Periodic Count Operation in Up-Counting by the Count Clock

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-3**, 000b (saw-wave PWM mode) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-3**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter. In **Figure 5.7-3**, 0000_0000h is set.
6	Start count operation	Set the GTCR.CST bit to 1b to start count operation.

(3) Periodic Count Operation in Down-counting by Count Clock

The GTCNT counter of each channel can perform down-counting by setting GTUDDTYC.UD with GTUPSR and GTDNSR registers set to 0000_0000h. When GTCNT changes from 0 to the GTPR value (underflow), GTST.TCFPU is set to 1b, and the underflow interrupt (GPT_Ux_gpt_gtciu_n_m) is also generated. After the GTCNT counter underflows, down-counting resumes from the GTPR value.

Figure 5.7-4 shows an example of periodic count operation in down-counting by the count clock.

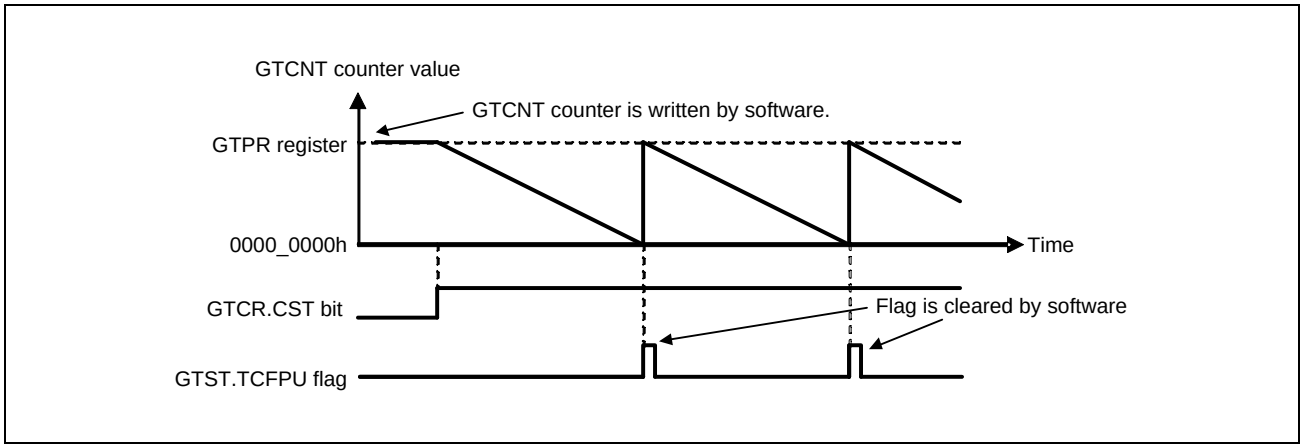


Figure 5.7-4 Example of Periodic Count Operation in Down-Counting by the Count Clock

Table 5.7-15 shows an example for setting periodic count operation in down-counting by the count clock.

Table 5.7-15 Example for Setting Periodic Count Operation in Down-Counting by Count Clock

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-4**, 000b (saw-wave PWM mode) is set.
2	Set count direction	Select the count direction with the GTUDDTYC register. In **Figure 5.7-4**, after 10b is set in the GTUDDTYC[1:0] bits, 00b is set in the GTUDDTYC[1:0] bits (down-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter. In **Figure 5.7-4**, the GTPR register value is set.
6	Start count operation	Set the GTCR.CST bit to 1 to start count operation. In **Figure 5.7-4**, 1 is set in the CST bit.

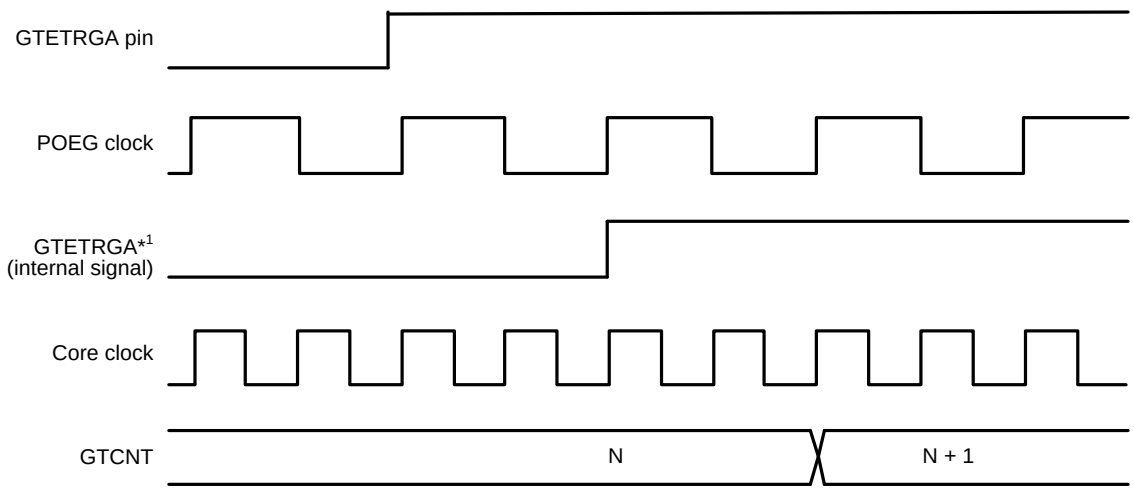
(4) Event Count Operation in Up-counting Using Hardware Sources

The GTCNT counter of each channel can perform up-counting using hardware sources set in GTUPSR.

When GTUPSR is set to enable, the count clock selected in GTCR.TPCS[3:0] and the count direction selected in GTUDDTYC.UD are ignored. If up-counting and down-counting using hardware sources occur at the same time, the GTCNT counter value does not change. The overflow behavior when up-counting using hardware sources is the same as when up-counting by the count clock.

When GTCR.CST bit is set to 1b to count up using hardware sources, counting operation is enabled. After GTCR.CST is set to 1b, the counter cannot count up for 1 clock cycle as specified in GTCR.TPCS[3:0] because counting operation is synchronized by the count clock selected in GTCR.TPCS[3:0]. Set GTCR.TPCS[3:0] to 000b to count up with a 1 PCLKD delay after GTCR.CST is set to 1b.

Figure 5.7-5 shows an example of event count operation in up-counting by a hardware source (the rising edge of the GTETRGA pin input).



Note: POEG clock: GPT_0_pclk_sfr, GPT_1_pclk_sfr
Core clock: GPT_0_clks_gpt, GPT_1_clks_gpt

Note 1. The GTETRGA internal signal reflects the signal being input to the GPT when the digital noise filter for the GTETRGA pin in the POEG is not in use.

Figure 5.7-5 Example of Event Count Operation in Up-Counting Using Hardware Sources

Table 5.7-16 shows an example for setting event count operation in up-counting by a hardware source.

Table 5.7-16 Example for Setting Event Count Operation in Up-Counting Using Hardware Sources

No.	Step Name	Description
1	Set count source	Select the counting-up source with the GTUPSR register.
2	Set cycle	Set the cycle in the GTPR register.
3	Set initial value for counter	Set the initial value in the GTCNT counter.
4	Start count operation	Set the GTCR.CST bit to 1b to start count operation.

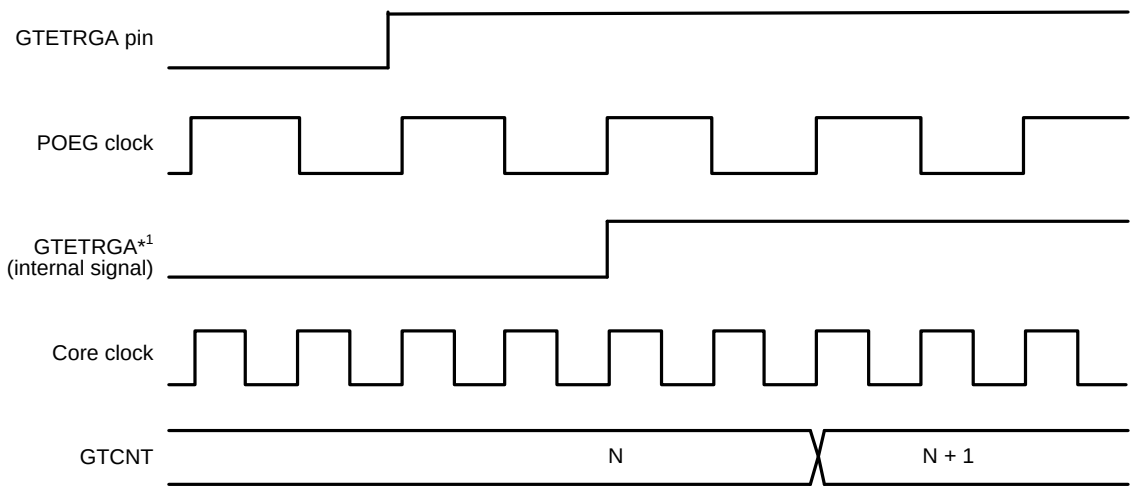
(5) Event Count Operation in Down-counting Using Hardware Sources

The GTCNT counter of each channel can perform down-counting using hardware sources set in GTDNSR.

When GTDNSR is set to enable, the count clock selected in GTCR.TPCS[3:0] and the count direction selected in GTUDDTYC.UD are ignored. If up-counting and down-counting using hardware sources occur at the same time, the GTCNT counter value does not change. The underflow behavior when down-counting using hardware sources is the same as when down-counting by the count clock.

When GTCR.CST bit is set to 1b to count down using hardware sources, counting operation is enabled. After GTCR.CST is set to 1b, the counter cannot count down for 1 clock cycle as specified in GTCR.TPCS[3:0] because counting operation is synchronized with the count clock selected in GTCR.TPCS[3:0]. Set GTCR.TPCS[3:0] to 000b to count down with a 1 PCLKD delay after GTCR.CST is set to 1b.

Figure 5.7-6 shows an example of event count operation in down-counting by a hardware source (the rising edge of the GTETRGA pin input).



Note: POEG clock: GPT_0_pclk_sfr, GPT_1_pclk_sfr
Core clock: GPT_0_clks_gpt, GPT_1_clks_gpt

Note 1. The GTETRGA internal signal reflects the signal being input to the GPT when the digital noise filter for the GTETRGA pin in the POEG is not in use.

Figure 5.7-6 Example of Event Count Operation in Down-Counting Using Hardware Sources

Table 5.7-17 shows an example for setting event count operation in down-counting by a hardware source.

Table 5.7-17 Example for Setting an Event Count Operation in Down-Counting Using Hardware Sources

No.	Step Name	Description
1	Set count source	Select the counting-down source with the GTDNSR register.
2	Set cycle	Set the cycle in the GTPR register.
3	Set initial value for counter	Set the initial value in the GTCNT counter.
4	Start count operation	Set the GTCR.CST bit to 1b to start count operation.

(6) Counter Clearing Operation

The counter of each channel is cleared by any of the following sources:

- Writing 0b to GTCNT register
- Writing 1b to the bit in GTCLR corresponding to the GPT channel number when the GTCSR.CCLR bit is 1b
- The hardware source selected in GTCSR register.

Writing to the GTCNT register is prohibited during counting. The GTCNT counter can be cleared by both writing 1b to the GTCLR and requesting clearing by hardware sources, even when GTCNT is counting (GTCR.CST is 1b) or not counting (GTCR.CST is 0).

When the count direction flag is set as decrement (GTST.TCUF flag = 0b) in saw-wave mode selected with GTCR.MD[2:0] bits, the GTCNT register is set to the value of the GTPR register when writing 1b to the GTCLR register or clearing by hardware sources.

When operation is neither in saw-waves mode nor down-counting, the GTCNT register is set to 0b when writing 1b to the GTCLR register or clearing by hardware sources.

In event count operation when at least 1 bit in the GTUPSR or GTDNSR register is set to 1b, after the generation of clear sources, both writing to GTCLR register and clearing by hardware sources are performed immediately for synchronization with PCLKD. If other settings are used, clearing is synchronized with the counter clock selected in GTCR.TPCS[3:0].

5.7.3.1.2 Waveform Output by Compare Match

Compare match means that the GTCNT counter value matches the value of GTCCRA or GTCCRB. When a compare match occurs, the compare match flag is generated synchronously with the count clock, including the event count. At the same time, the GPT can perform low-level, high-level, or toggled output from the corresponding GTIOCnA or GTIOCnB output pin (n = 0 to 15). In addition, the GTIOCnA or GTIOCnB pin output can be low-level, high-level, or toggled output at the cycle end which is determined by GTPR.

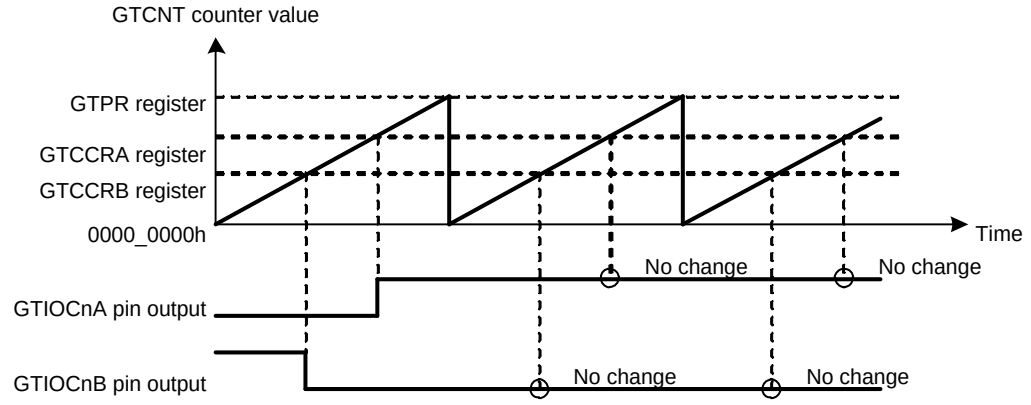
The cycle end is:

- For saw waves in up-counting: When GTCNT changes from the GTPR value to 0b (overflow)
- For saw waves in down-counting: When GTCNT changes from 0b to GTPR value (underflow)
- For saw waves: When the GTCNT counter is cleared
- For triangle waves: When the GTCNT counter changes from 0b to 1b (trough).

(1) Low-level/High-level Output

Figure 5.7-7 shows an example of low-level and high-level output operation by a compare match with GTCCRA and GTCCRB.

In this example, the GTCNT counter performs up-counting, and settings are made so that the high level is output from the GTIOCnA pin by a compare match with GTCCRA, and the low level is output from the GTIOCnB pin by a compare match with GTCCRB. The pin level does not change when the specified level and pin level match.



[Setting examples]
GTIOR.GTIOA[4:0] bits: Initial output is low, high output at compare match, output retained at cycle end
GTIOR.GTIOB[4:0] bits: Initial output is high, low output at compare match, output retained at cycle end

Note: The base addresses of the registers are the same.

Figure 5.7-7 Example of Low-level and High-level Output Operation

Table 5.7-18 shows an example for setting low-level and high-level output operation.

Table 5.7-18 Example for Setting Low-level and High-level Output Operation

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-7**, 000b (saw-wave PWM mode) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-7**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter.
6	Set GTIOCnm pin function	Set the GTIOCnm pin function with the GTIOA[4:0] and GTIOB[4:0] bits in the GTIOR register. In **Figure 5.7-7**, GTIOA[4:0] = 0_0010b, GTIOB[4:0] = 1_0001b.
7	Enable GTIOCnm pin output	Set to enable the GTIOCnm pin output with the OAE and OBE bits in the GTIOR register.
8	Set compare match value	Set compare match values in the GTCCRA and GTCCRB registers.
9	Start count operation	Set the GTCR.CST bit to 1b to start count operation.

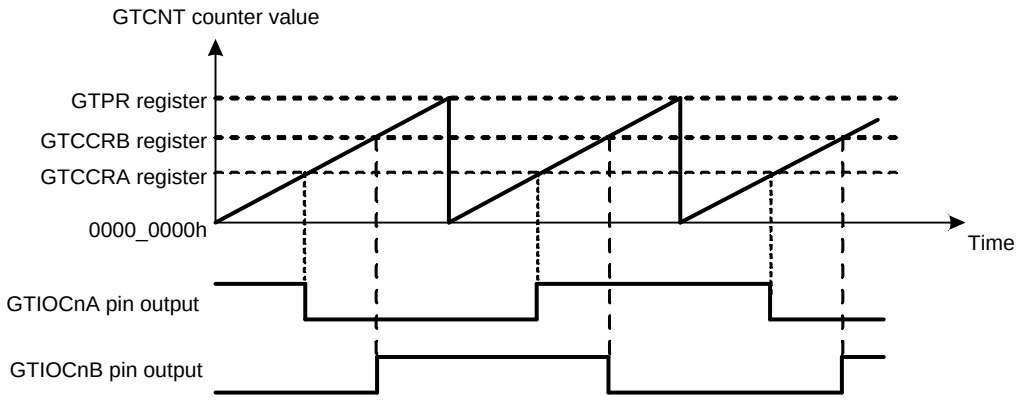
Note: n: 0 to 15
m: A, B

(2) Toggled Output

Figure 5.7-8 and **Figure 5.7-9** show examples of toggled output operation by compare matches with GTCCRA and GTCCRB.

In **Figure 5.7-8**, the GTCNT counter performs up-counting, and settings are made so that the GTIOCnA pin output by a compare match with GTCCRA and GTIOCnB pin output by a compare match with GTCCRB are toggled.

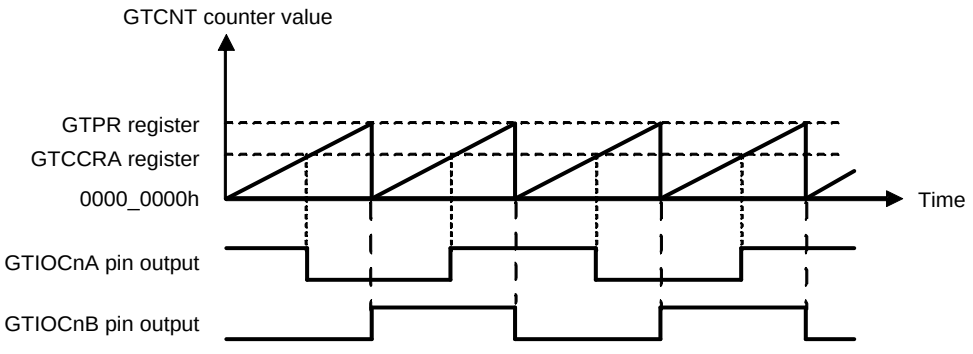
In **Figure 5.7-9**, the GTCNT counter performs up-counting, and settings are made so that a compare match with GTCCRA toggles the GTIOCnA pin output and toggles the GTIOCnB pin output at the cycle end.



[Setting examples]
GTIOR.GTIOA[4:0] bits: Initial output is high, output toggled at compare match, output retained at cycle end
GTIOR.GTIOB[4:0] bits: Initial output is low, output toggled at compare match, output retained at cycle end

Note: The base addresses of the registers are the same.

Figure 5.7-8 Example of Toggled Output Operation (1)



[Setting examples]
GTIOR.GTIOA[4:0] bits: Initial output is high, output toggled at compare match, output retained at cycle end
GTIOR.GTIOB[4:0] bits: Initial output is low, output retained at compare match, output toggled at cycle end

Note: The base addresses of the registers are the same.

Figure 5.7-9 Example of Toggled Output Operation (2)

Table 5.7-19 shows an example for setting toggled output operation.

Table 5.7-19 Example for Setting Toggled Output Operation

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-8** and **Figure 5.7-9**, 000b (saw-wave PWM mode) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-8** and **Figure 5.7-9**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter.
6	Set GTIOCnm pin function	Set the GTIOCnm pin function with the GTIOA[4:0] and GTIOB[4:0] bits in the GTIOR register. In **Figure 5.7-8**, GTIOA[4:0] = 1_0011b, GTIOB[4:0] = 0_0011b, and in **Figure 5.7-9**, GTIOA[4:0] = 1_0011b, GTIOB[4:0] = 0_1100b.
7	Enable GTIOCnm pin output	Set to enable the GTIOCnm pin output with the OAE and OBE bits in the GTIOR register.
8	Set compare match value	Set compare match values in the GTCCRA and GTCCRB registers.
9	Start count operation	Set the GTCR.CST bit to 1b to start count operation.

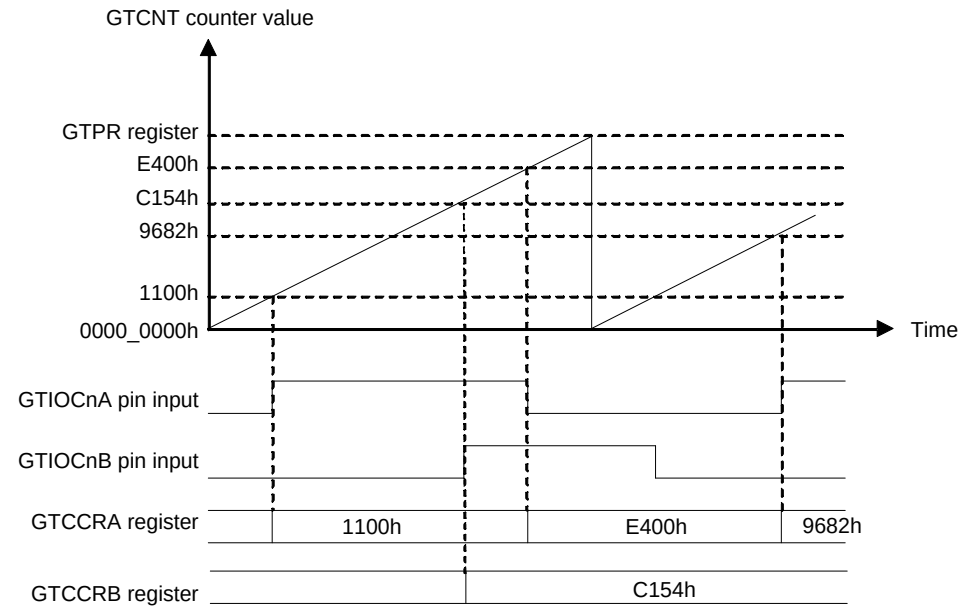
Note: n: 0 to 15
m: A, B

5.7.3.1.3 Input Capture Function

The GTCNT counter value can be transferred to either GTCCRA or GTCCRB on detection of the hardware source that is set in GTICASR and GTICBSR.

Figure 5.7-10 shows an example of the input capture function.

In this example, the GTCNT counter performs up-counting by the count clock, and settings are made so that input capture is performed to GTCCRA at both edges of the GTIOCnA input pin and to GTCCRB on the rising edge of the GTIOCnB input pin.



[Setting examples]
GTICASR setting: Input capture at both edges
GTICBSR setting: Input capture at the rising edge

Note: The base addresses of the registers are the same.

Figure 5.7-10 Example of Input Capture Operation

Table 5.7-20 and **Table 5.7-23** show the example for setting input capture operation with count operation by the count clock.

Table 5.7-20 Example for Setting Input Capture Operation

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-10**, 000b (saw-wave PWM mode) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-10**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter.
6	Select input capture source	Select the input capture source in the GTICASR and GTICBSR registers. In **Figure 5.7-10**, GTICASR = 0000_0F00h, GTICBSR = 0000_3000h.
7	Start count operation	Set the GTCR.CST bit to 1b to start count operation.

5.7.3.2 Buffer Operation

The following buffer operations can be set with GTBER:

- GTPR, GTPBR, and GTPDBR
- GTCCRA, GTCCRC, and GTCCRD
- GTCCRB, GTCCRE, and GTCCRF
- GTADTRA, GTADTBRA, and GTADTDBRA
- GTADTRB, GTADTBRB, and GTADTDBRB
- GTDVU and GTDBU
- GTDVD and GTDBD

5.7.3.2.1 GTPR Register Buffer Operation

GTPBR can function as a buffer register for GTPR, and GTPDBR can function as a buffer register for GTPBR (double buffer register for GTPR). Buffer transfer is performed on overflow (during up-counting) or underflow (during down-counting) in saw-wave mode or in event counting, and at troughs in triangle-wave mode.

In saw-wave mode or in event counting, buffer transfer is performed when any of the following counter clearing operations occurs during counting:

- Clearing by hardware sources (the clear source is selected in GTCSR register)
- Clearing by software (when GTCSR.CCLR bit is 1b and GTCLR.CCLRn bit is set to 1, n = 0 to 15).

To set GTPR to function as a buffer, set the GTBER.PR bit to 1b. To set GTPR not to function as a buffer, set the GTBER.PR bit to 0b.

Figure 5.7-11 to **Figure 5.7-13** show examples of GTPR buffer operation and **Table 5.7-21** shows an example for setting GTPR buffer operation.

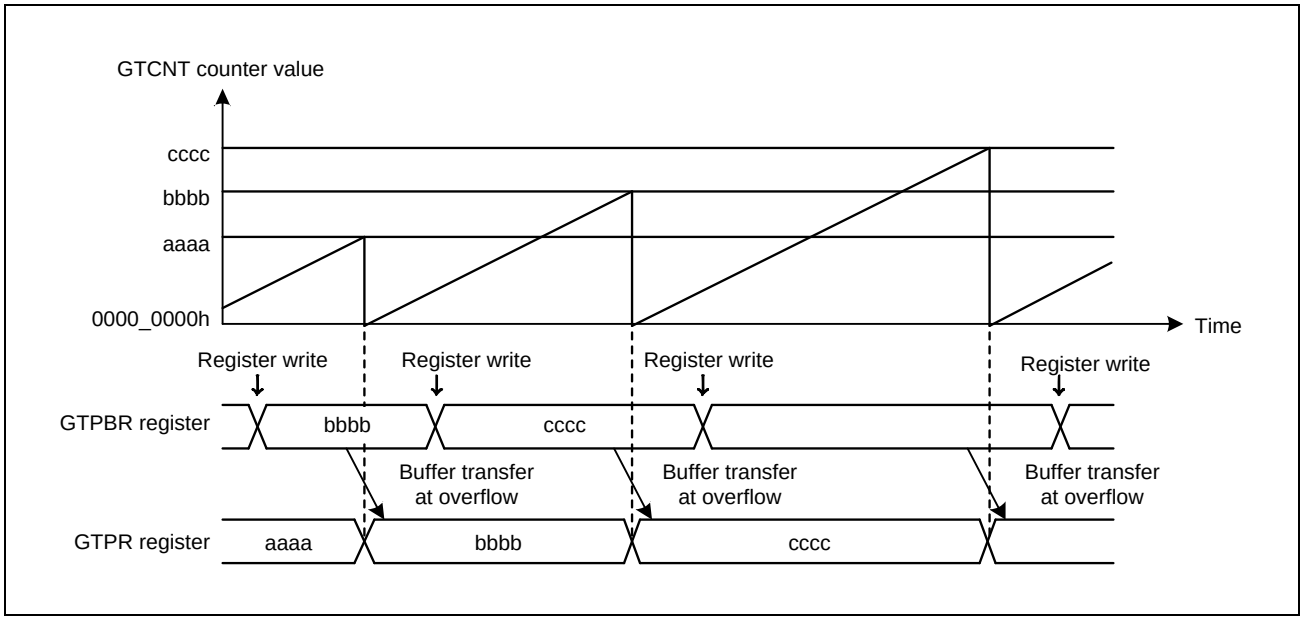


Figure 5.7-11 Example of GTPR Buffer Operation with Saw Waves in Up-Counting

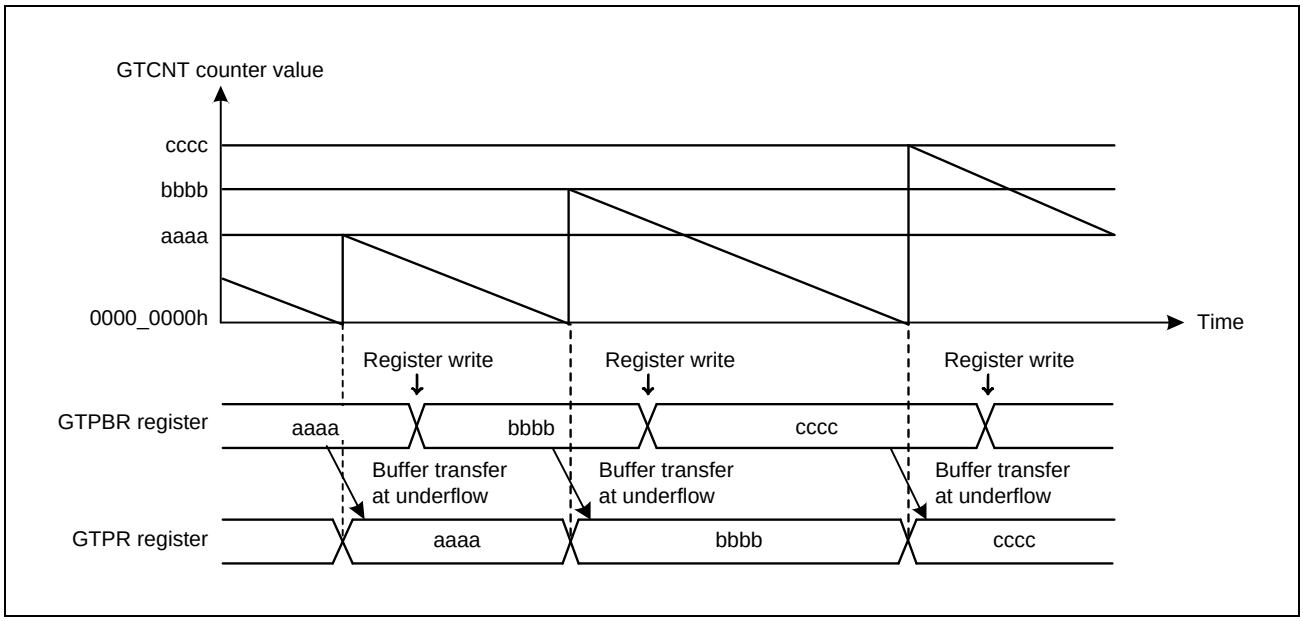


Figure 5.7-12 Example of GTPR Buffer Operation with Saw Waves in Down-Counting

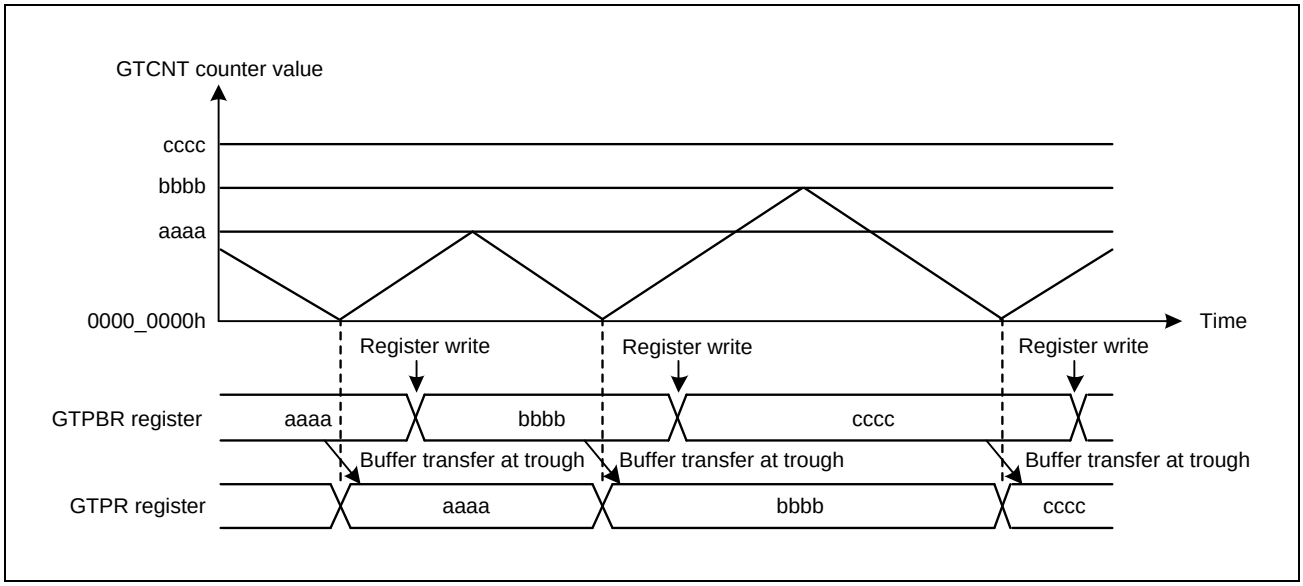


Figure 5.7-13 Example of GTPR Buffer Operation with Triangle Waves

Table 5.7-21 Example for Setting GTPR Register Buffer Operation

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-11** and **Figure 5.7-12**, 000b (saw-wave PWM mode) is set, and in **Figure 5.7-13**, 100b (triangle-wave PWM mode 1) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-11**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting). In **Figure 5.7-12**, after 10b is set in the GTUDDTYC[1:0] bits, 00b is set in the GTUDDTYC[1:0] bits (down-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter.
6	Set buffer operation	Set buffer operation with the GTBER.PR[1:0] bits. In **Figure 5.7-11**, **Figure 5.7-12**, and **Figure 5.7-13**, PR[1:0] = 01b.
7	Set buffer value	For buffer operation, set a value of one cycle after the current cycle in GTPBR. For double buffer operation, also set a value in two cycles after the current cycle in GTPDBR.
8	Start count operation	Set the GTCR.CST bit to 1b to start count operation.
9	Set buffer value for each cycle	For buffer operation, set a value of one cycle after the current cycle in GTPBR. For double buffer operation, also set a value in two cycles after the current cycle in GTPDBR.

5.7.3.2.2 Buffer Operation for GTCCRA and GTCCRB Registers

GTCCRC can function as a buffer register for GTCCRA and GTCCRD can function as a buffer register for GTCCRC (double-buffer register for GTCCRA). Similarly, GTCCRE can function as a buffer register for GTCCRB and GTCCRF can function as a buffer register for GTCCRE (double-buffer register for GTCCRB).

To set GTCCRA or GTCCRB to function as a double buffer, set GTBER.CCRA[1:0] or GTBER.CCRB[1:0] to 10b or 11b. For single buffer operation, set 01b. To set GTCCRA or GTCCRB to not function as a buffer, set 00b.

(1) When GTCCRA or GTCCRB Functions as Output Compare Register

Buffer transfer occurs in the following situations:

- Buffer transfer by overflow or underflow

  Buffer transfer is performed on overflow (during up-counting) or underflow (during down-counting) in saw-wave mode or in event count operation. In triangle-wave mode, buffer transfer is performed at troughs (triangle-wave PWM mode 1) or at crests and troughs (triangle-wave PWM mode 2).

- Buffer transfer by counter clearing

  In saw-wave mode or in event count operation, during counting, buffer transfer (which is the same as an overflow during up-counting or an underflow during down-counting) is performed by the counter clear sources similar to the case shown in **5.7.3.2.1 GTPR Register Buffer Operation**.

  In triangle-wave mode, buffer transfer is not performed by counter clearing.

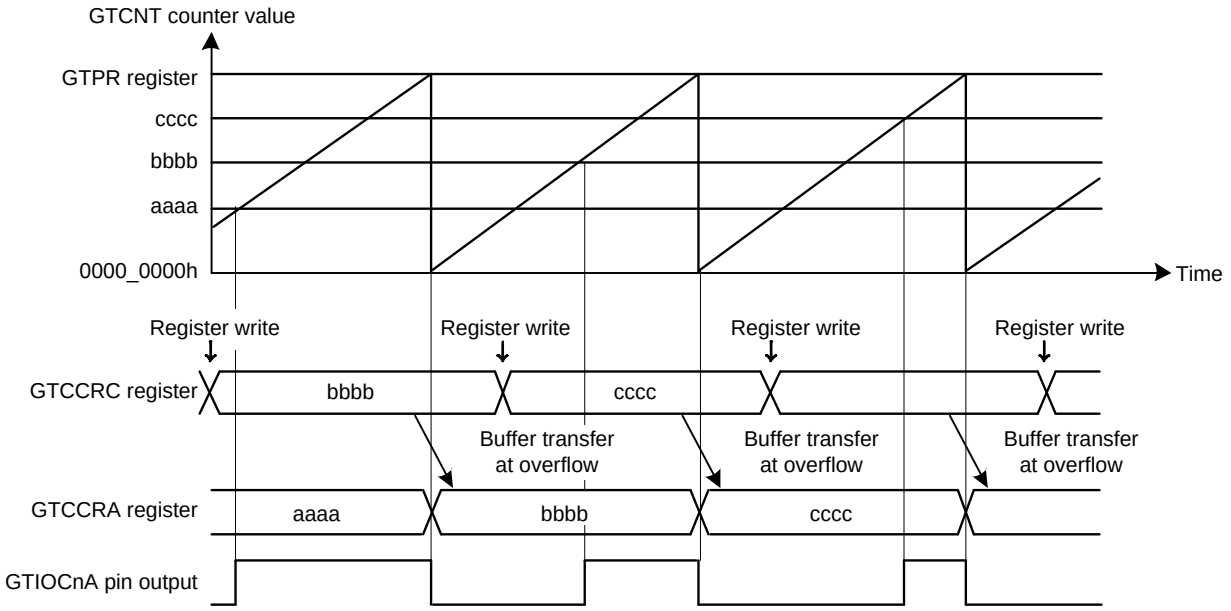
- Forcible buffer transfer

  When GTBER.CCRSWT bit is set to 1 while counting is stopped, buffer transfer of the GTCCRA and GTCCRB registers is performed forcibly in saw-wave mode, in event count operation and in triangle-wave mode.

  Additionally buffer transfer from the GTCCRD register to temporary register A and from the GTCCRF register to temporary register B is performed in saw-wave one-shot pulse mode or in triangle-wave PWM mode 3.

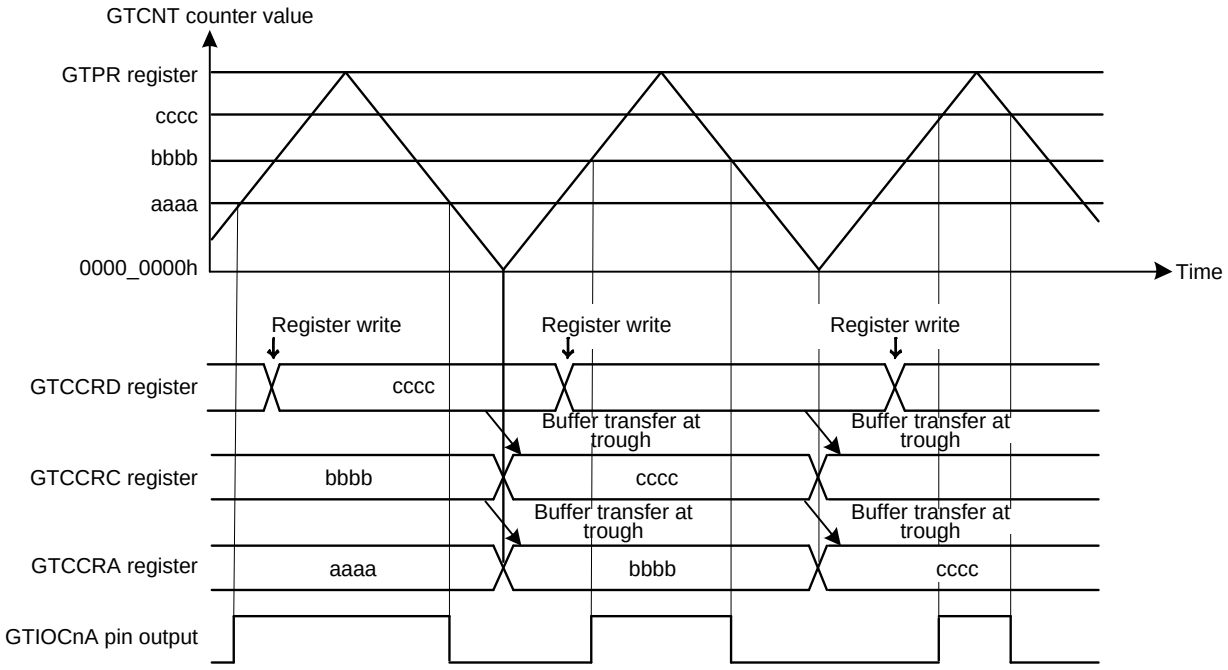
If a double buffer is used in saw-wave one-shot pulse mode, triangle-wave PWM mode 3, or triangle-wave PWM mode 2 by setting the GTBER.DBRTECm (m = A, B) bit to 1, even while buffer transfer is inhibited by the setting of the GTBER.BD[0] bit or buffer transfer extended skipping function, transfer from the intermediate buffer to the GTCCRm (m = A, B) register is repeated on a cyclic basis (the function for repeated double-buffered operation when buffer transfer is inhibited). For details, see **5.7.8.2.2 Repeated Double-Buffered Operation when GTCCR Buffer Transfer is Inhibited**.

Figure 5.7-14 to **Figure 5.7-16** show examples of GTCCRA and GTCCRB buffer operation and **Table 5.7-22** shows an example for setting GTCCRA and GTCCRB buffer operation.



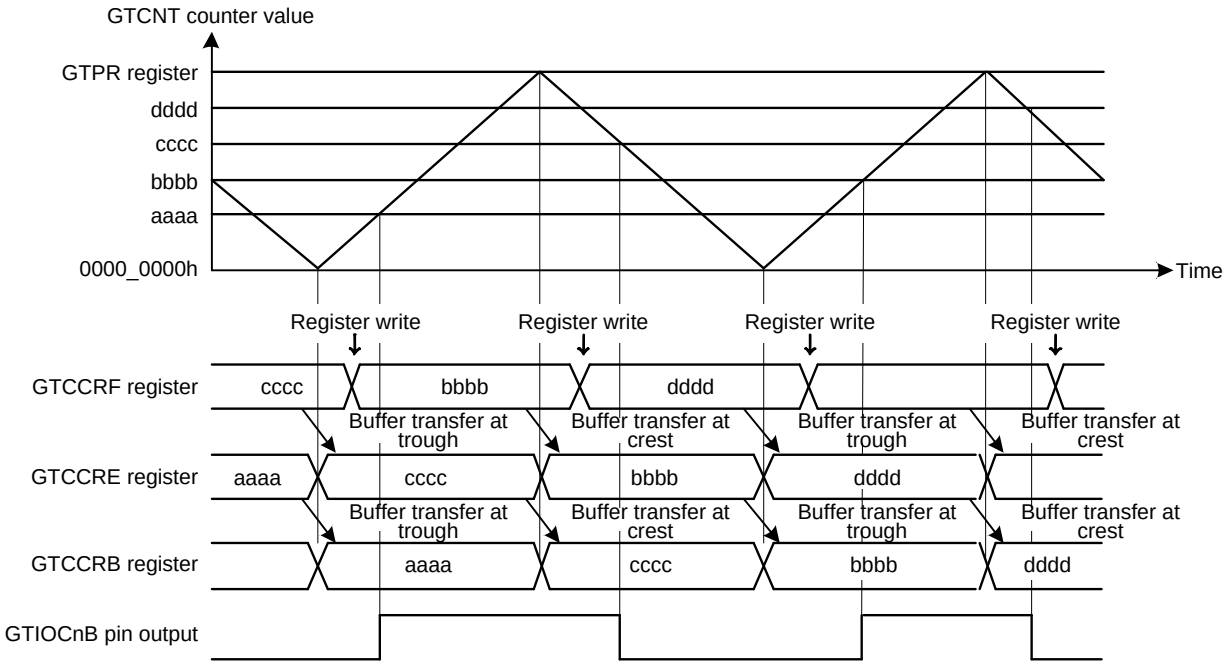
Note: The base addresses of the registers are the same.

Figure 5.7-14 Example of GTCCRA and GTCCRB Buffer Operation with Output Compare, Saw Waves in Up-Counting, High-level Output at GTCCRA Compare Match, and Low-level Output at Cycle End



Note: The base addresses of the registers are the same.

Figure 5.7-15 Example of GTCCRA and GTCCRB Double Buffer Operation with Output Compare, Triangle Waves, Buffer Operation at Trough, Output Toggled at GTCCRA Compare Match, and Output Retained at Cycle End



Note: The base addresses of the registers are the same.

Figure 5.7-16 Example of GTCCRA and GTCCRB Double Buffer Operation with Output Compare, Triangle Waves, Buffer Operation at Both Troughs and Crests, Output Toggled at GTCCRB Compare Match, and Output Retained at Cycle End

Table 5.7-22 Example for Setting GTCCRA and GTCCRB Buffer Operation for Output Compare

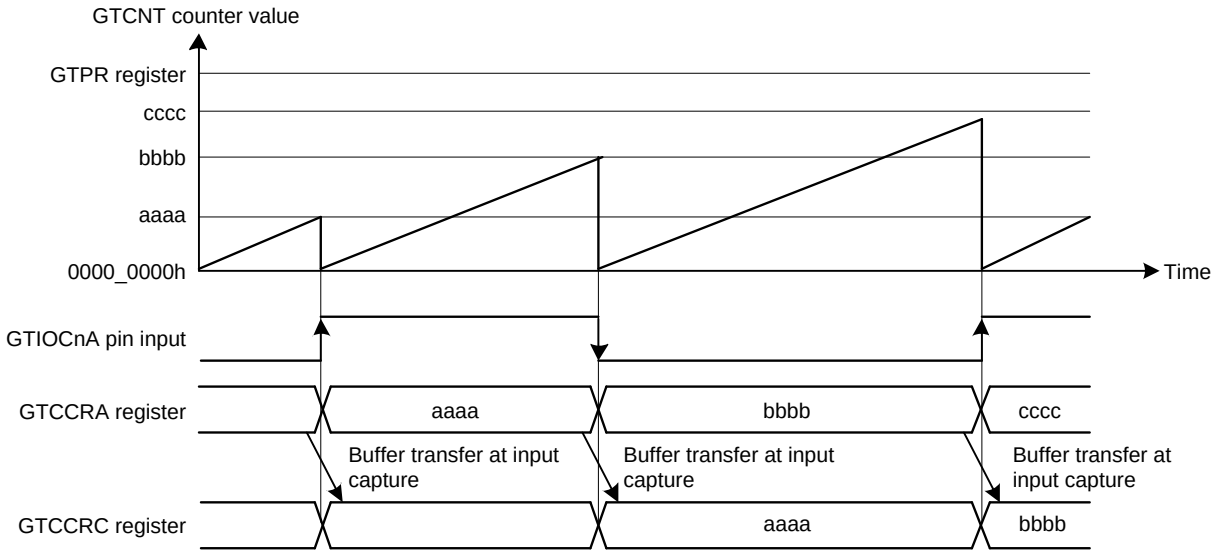
No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-14**, 000b (saw-wave PWM mode) is set, in **Figure 5.7-15**, 100b (triangle-wave PWM mode 1) is set, and in **Figure 5.7-16**, 101b (triangle-wave PWM mode 2) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-14**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter.
6	Set GTIOCnm pin function	Set the GTIOCnm pin function with the GTIOA[4:0] and GTIOB[4:0] bits in the GTIOR register. In **Figure 5.7-14**, GTIOA[4:0] = 0_0110b, in **Figure 5.7-15**, GTIOA[4:0] = 0_0011b, and in **Figure 5.7-16**, GTIOB[4:0] = 0_0011b.
7	Enable GTIOCnm pin output	Set to enable the GTIOCnm pin output with the OAE and OBE bits in the GTIOR register.
8	Set buffer operation	Set buffer operation with the CCRA[1:0] and CCRB[1:0] bits in the GTBER register. In **Figure 5.7-14**, CCRA[1:0] = 01b, in **Figure 5.7-15**, CCRA[1:0] = 1xb, and in **Figure 5.7-16**, CCRB[1:0] = 1xb.
9	Set compare match value	Set the GTIOCnA pin transition in the GTCCRA register and the GTIOCnB pin transition in the GTCCRB register.
10	Set buffer value	For buffer operation, set the GTIOCnA and GTIOCnB pins transitions in 1 cycle after the current cycle (in saw-wave mode or triangle-wave mode with buffer transfer at trough or crest) or half cycle after the current cycle (in triangle-wave mode with buffer transfer at both trough and crest) in the GTCCRC and GTCCRE registers, respectively. For double buffer operation, also set the GTIOCnA and GTIOCnB pins transitions in 2 cycles after the current cycle (in saw-wave mode or triangle-wave mode with buffer transfer at trough or crest) or 1 cycle after the current cycle (in triangle-wave mode with buffer transfer at both trough and crest) in the GTCCRD and GTCCRF registers, respectively.
11	Start count operation	Set the GTCR.CST bit to 1b to start count operation.
12	Set buffer value for each cycle	For buffer operation, set the GTIOCnA and GTIOCnB pins transitions in 1 cycle after the current cycle (in saw-wave mode or triangle-wave mode with buffer transfer at trough or crest) or half cycle after the current cycle (in triangle-wave mode with buffer transfer at both trough and crest) in the GTCCRC and GTCCRE registers, respectively. For double buffer operation, also set the GTIOCnA and GTIOCnB pins transitions in 2 cycles after the current cycle (in saw-wave mode or triangle-wave mode with buffer transfer at trough or crest) or 1 cycle after the current cycle (in triangle-wave mode with buffer transfer at both trough and crest) in the GTCCRD and GTCCRF registers, respectively.

Note: n: 0 to 15
m: A, B

(2) When GTCCRA or GTCCRB Functions as Input Capture Register

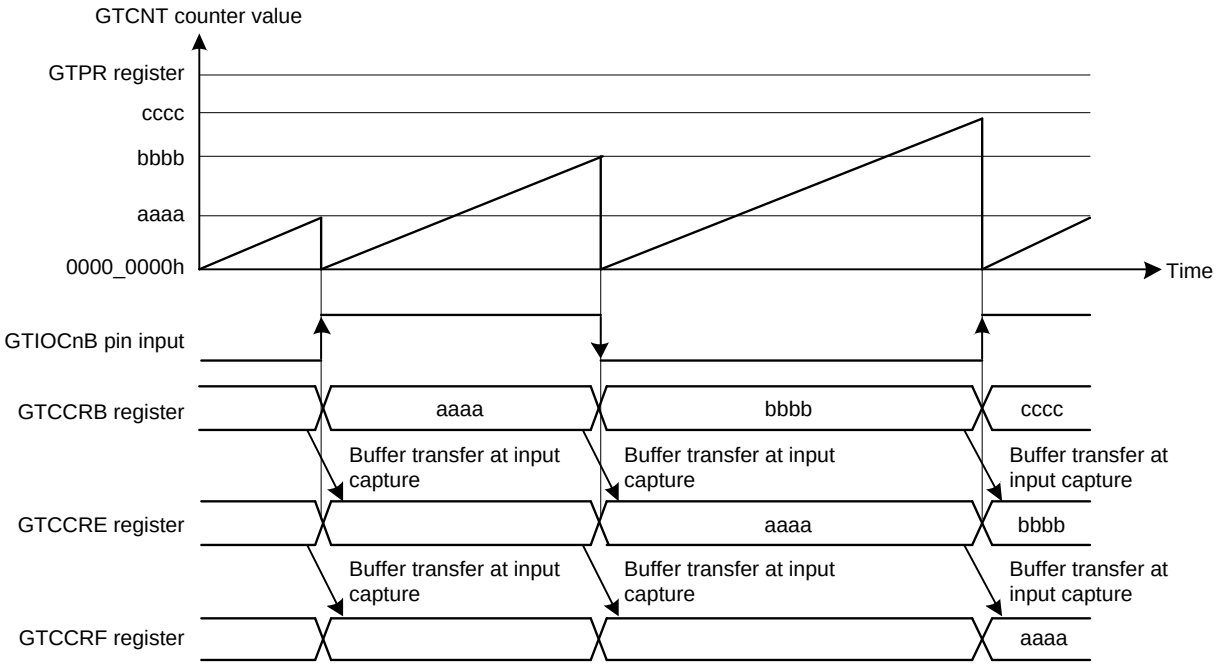
When input capture occurs, the GTCNT counter value is transferred to GTCCRA and GTCCRB and the stored GTCCRA and GTCCRB register values are transferred to the buffer registers. In input capture operation, buffer transfer is not performed by counter clearing.

Figure 5.7-17 and **Figure 5.7-18** show examples of GTCCRA and GTCCRB buffer operation and **Table 5.7-23** shows an example for setting GTCCRA and GTCCRB buffer operation.



Note: The base addresses of the registers are the same.

Figure 5.7-17 Example of GTCCRA and GTCCRB Buffer Operation with Input Capture at Both Edges of GTIOCnA Input, Saw Waves in Up-counting, and GTCNT Counter Cleared at Both Edges of GTIOCnA Input



Note: The base addresses of the registers are the same.

Figure 5.7-18 Example of GTCCRA and GTCCRB Double Buffer Operation with Input Capture at Both Edges of GTIOCnB Input, Saw Waves in Up-counting, and GTCNT Counter Cleared at Both Edges of GTIOCnB Input

Table 5.7-23 Example for Setting GTCCRA and GTCCRB Buffer Operation for Input Capture

No.	Step Name	Description
1	Set operating mode and counter clear sources	Set the operating mode with the GTCR.MD[2:0] bits and count clear source with the GTCSR register. In **Figure 5.7-17**, MD[2:0] = 000b (saw-wave PWM mode) and GTCSR = 0000_0F00h, and in **Figure 5.7-18**, MD[2:0] = 000b (saw-wave PWM mode) and GTCSR = 0000_F000h.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-17**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter.
6	Select input capture source	Select input capture source in the GTICASR register and GTICBSR register. In **Figure 5.7-17**, GTICASR = 0000_0F00h, and in **Figure 5.7-18**, GTICBSR = 0000_F000h.
7	Set buffer operation	Set buffer operation with the CCRA and CCRB bits in the GTBER register. In **Figure 5.7-17**, CCRA[1:0] = 01b, and in **Figure 5.7-18**, CCRB[1:0] = 1xb.
8	Start count operation	Set the GTCR.CST bit to 1b to start count operation.

5.7.3.2.3 Buffer Operation for GTADTRA and GTADTRB

The GTADTBRA register operates as a buffer register for the GTADTRA register and the GTADTDBRA register operates as a buffer register for the GTADTBRA register (a double-buffer register for the GTADTRA register). Similarly, the GTADTBRB register operates as a buffer register for the GTADTRB register and the GTADTDBRB register operates as a buffer register for the GTADTBRB register (a double-buffer register for the GTADTRB register).

To make the GTADTRA and GTADTRB registers operate as double buffers, set the GTBER.ADTDA and ADTDB bits to 1b, respectively. For single buffer operation, set these bits to 0b. To make the GTADTRA and GTADTRB buffers not operate as buffers, set the GTBER.ADTTA[1:0] and ADTTB[1:0] bits to 00b, respectively.

The timing of buffer transfer can be set by using the GTBER.ADTTm[1:0] (m = A, B) bits. Buffer transfer proceeds on overflow (in up-counting) or underflow (in down-counting) in saw-wave mode. In triangle-wave mode, it proceeds at crests when the setting of the GTBER.ADTTm[1:0] (m = A, B) bits is 01b, at troughs when the setting is 10b, and at crests and troughs when the setting is 11b.

In saw-wave mode, when counting is in progress and the setting of the GTBER.ADTTm[1:0] (m = A, B) bits is other than 00b, buffer transfer proceeds in the same way as that on overflow (in up-counting) or underflow (in down-counting) by the same counter clearing source in **5.7.3.2.1 GTPR Register Buffer Operation**.

If the GTADTRm (m = A, B) register is made to operate as a double buffer and triangle-wave mode is used by setting the GTBER.DBRTEADm (m = A, B) bit to 1b, even while buffer transfer is inhibited by the setting of the GTBER.BD[2] bit or buffer transfer extended skipping function, transfer to the GTADTRm (m = A, B) register is repeated on a cyclic basis (the function for repeated double-buffered operation when buffer transfer is inhibited). For details, see **5.7.8.2.3 Repeated Double-Buffered Operation when GTADTR Buffer Transfer is Inhibited**.

Figure 5.7-19 to **Figure 5.7-21** show examples of buffer operation of the GTADTRA and GTADTRB registers, and **Figure 5.7-22** shows an example for setting buffer operation for the GTADTRA and GTADTRB registers.

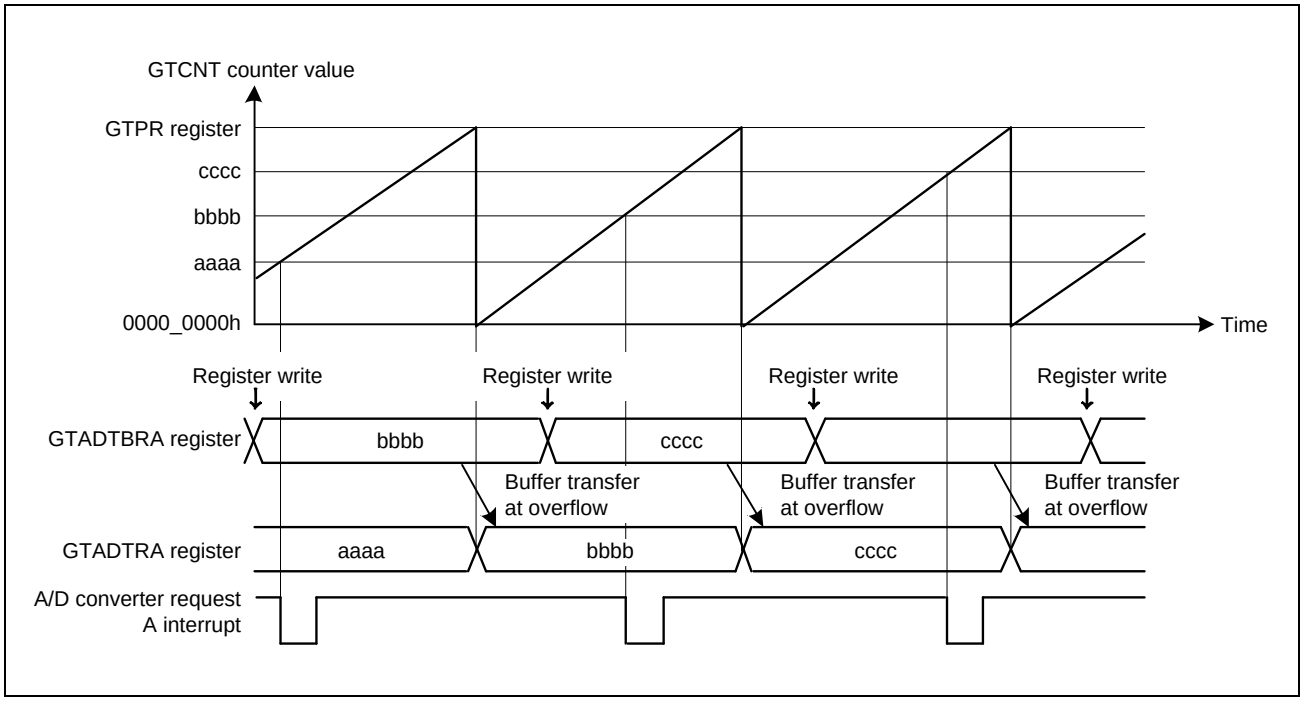


Figure 5.7-19 Example of Buffer Operation of GTADTRA and GTADTRB Registers (Saw-Wave Up-Counting, A/D Converter Start Request Interrupt Generated in Up-Counting)

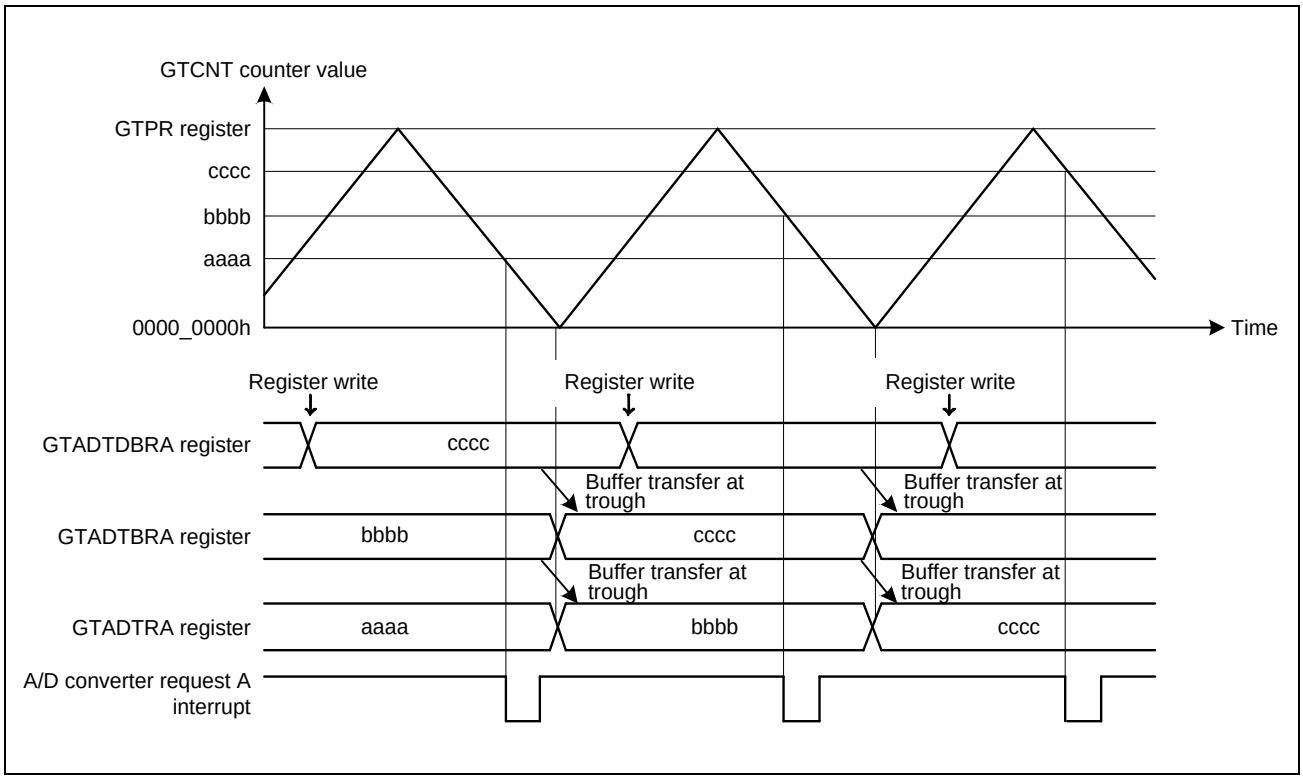


Figure 5.7-20 Example of Double-Buffered Operation of GTADTRA and GTADTRB Registers (Triangle Waves, Buffer Transfer at Troughs, A/D Converter Start Request Interrupt Generated in Down-Counting)

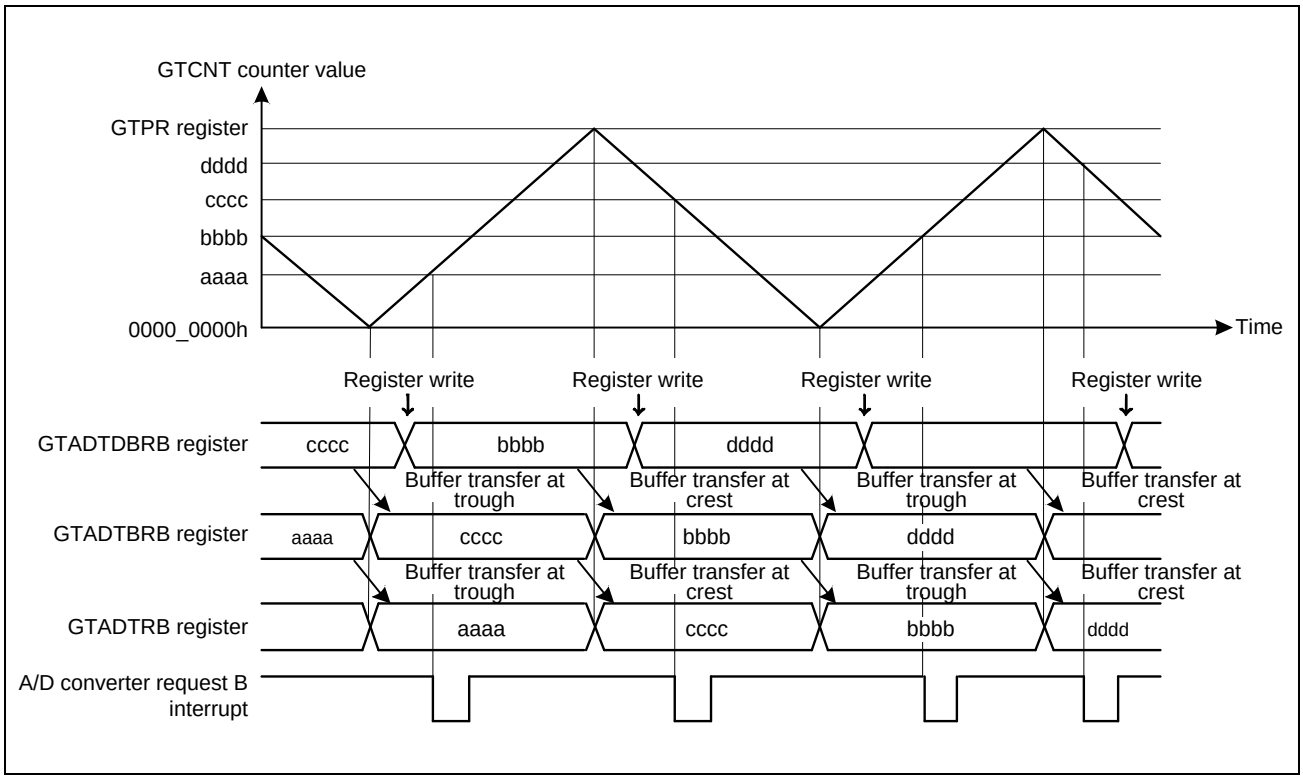


Figure 5.7-21 Example of Double-Buffered Operation of GTADTRA and GTADTRB Registers (Triangle Waves, Buffer Transfer at Both Troughs and Crests, A/D Converter Start Request Interrupt Generated in Both Up-Counting and Down-Counting)

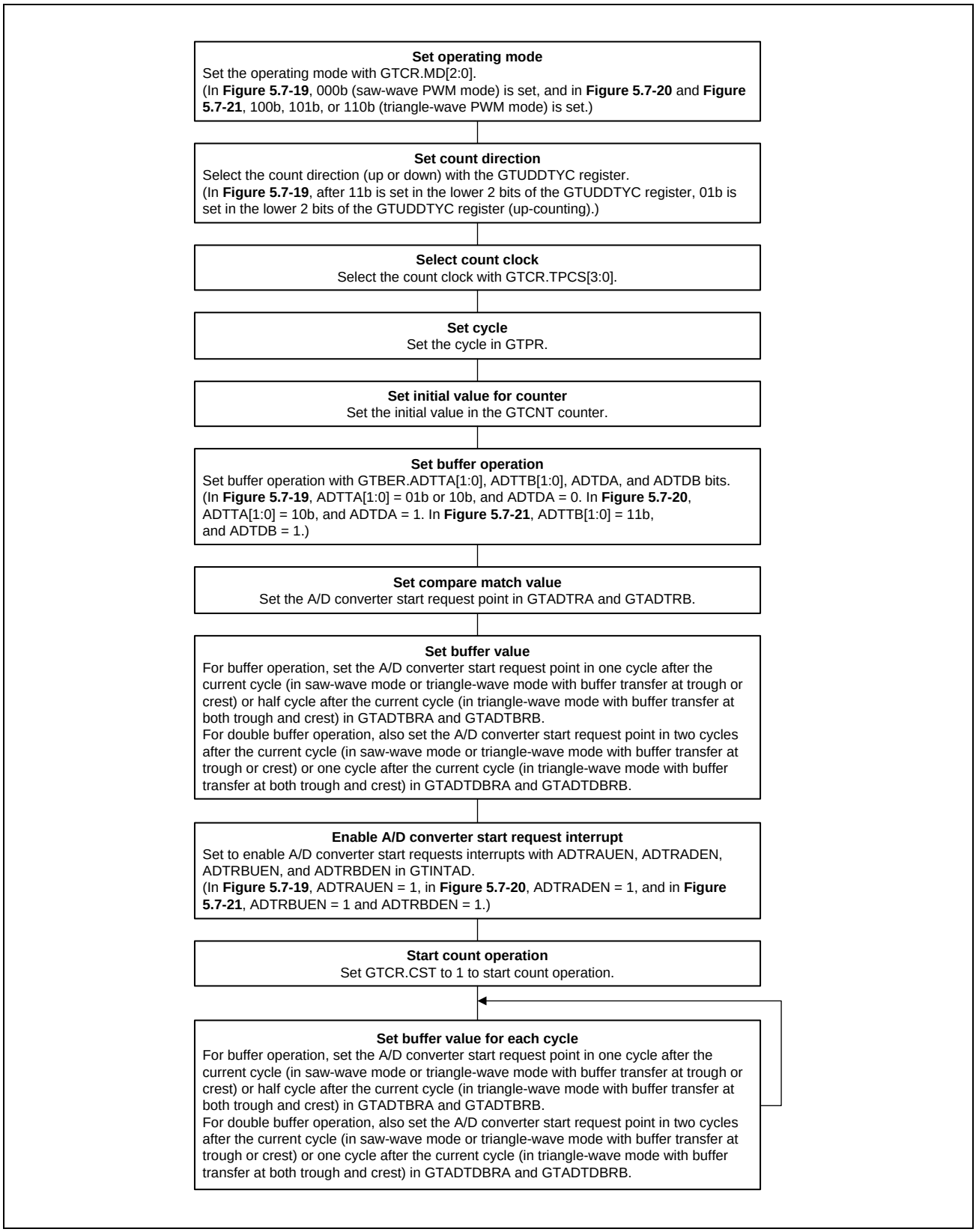


Figure 5.7-22 Example of Setting Buffer Operation for GTADTRA and GTADTRB Registers

5.7.3.3 PWM Output Operating Mode

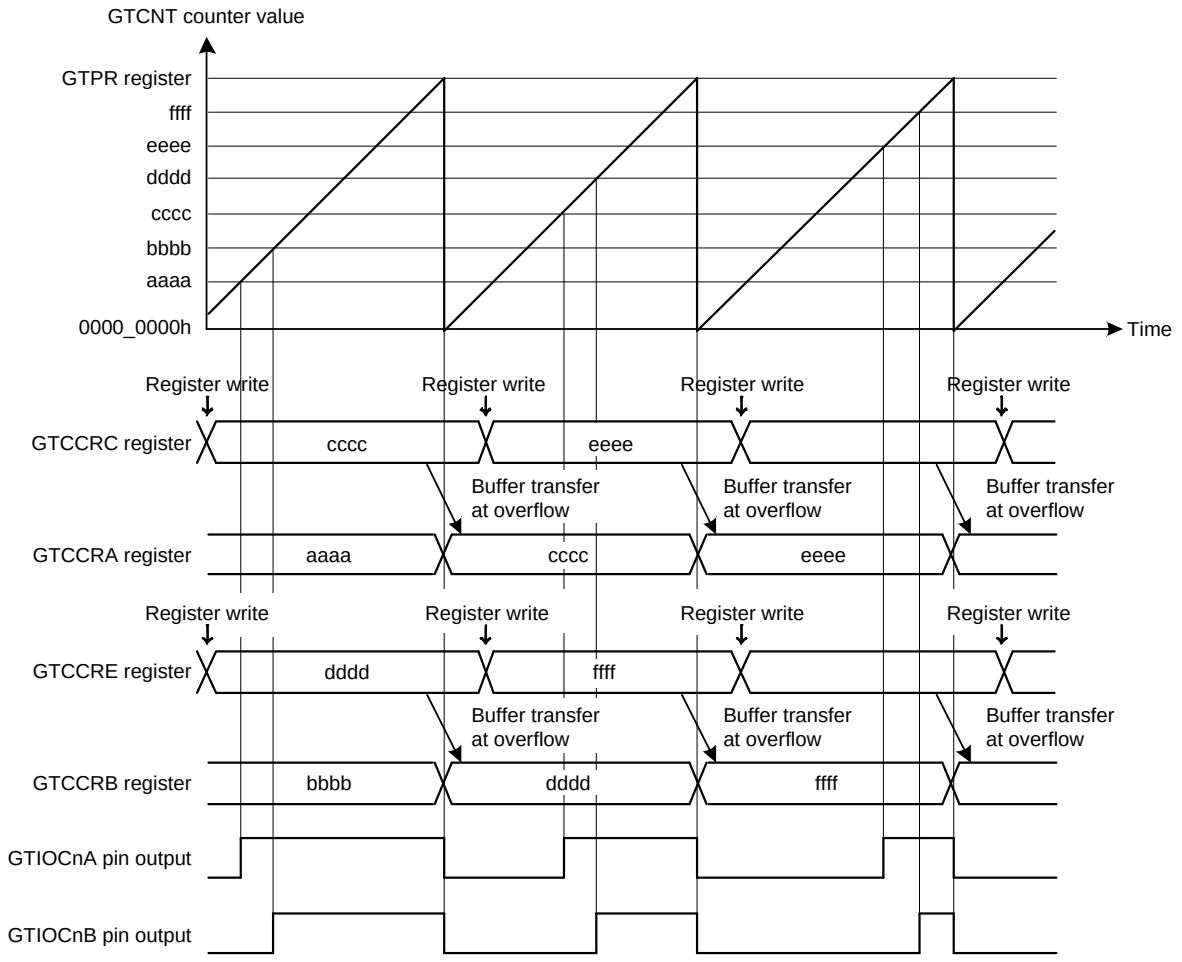
The GPT can output PWM waveforms to the GTIOCnA or GTIOCnB pin (n = 0 to 15) by a compare match between the GTCNT counter and GTCCRA or GTCCRB.

By setting GTDTCR, GTDVU, and GTDVD, the compare match value for a negative-phase waveform with dead time can be automatically set in GTCCRB.

5.7.3.3.1 Saw-wave PWM Mode

In saw-wave PWM mode, GTCNT performs saw-wave (half-wave) operation by setting the cycle in GTPR and a PWM waveform is output to the GTIOCnA or GTIOCnB pin (n = 0 to 15) when a GTCCRA or GTCCRB compare match occurs. The pin output value can be selected from low-level output, high-level output, or toggle output separately for a compare match and for the cycle end according to the GTIOR setting.

Figure 5.7-23 shows an example of saw-wave PWM mode operation, and **Table 5.7-24** shows an example for setting saw-wave PWM mode.



Note: The base addresses of the registers are the same.

Figure 5.7-23 Example of Saw-Wave PWM Mode Operation with Up-Counting, Buffer Operation, High-level Output at GTCCRA/GTCCRB Compare Match, and Low-level Output at Cycle End

Table 5.7-24 Example for Setting Saw-Wave PWM Mode

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-23**, 000b (saw-wave PWM mode) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-23**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter.
6	Set GTIOCnm pin function	Set the GTIOCnm pin function with the GTIOA[4:0] and GTIOB[4:0] bits in the GTIOR register. In **Figure 5.7-23**, GTIOA[4:0] = 0_0110b and GTIOB[4:0] = 0_0110b.
7	Enable GTIOCnm pin output	Set to enable the GTIOCnm pin output with the OAE and OBE bits in the GTIOR register.
8	Set buffer operation	Set buffer operation with the CCRA[1:0] and CCRB[1:0] bits in the GTBER register. In **Figure 5.7-23**, CCRA[1:0] = 01b and CCRB[1:0] = 01b.
9	Set compare match value	Set the GTIOCnA pin transition in the GTCCRA register and the GTIOCnB pin transition in the GTCCRB register.
10	Set buffer value	For buffer operation, set the GTIOCnA and GTIOCnB pins transitions in 1 cycle after the current cycle in the GTCCRC and GTCCRE registers, respectively. For double buffer operation, also set the GTIOCnA and GTIOCnB pins transitions in 2 cycles after the current cycle in the GTCCRD and GTCCRF registers, respectively.
11	Start count operation	Set the GTCR.CST bit to 1b to start count operation.
12	Set buffer value for each cycle	For buffer operation, set the GTIOCnA and GTIOCnB pins transitions in 1 cycle after the current cycle in the GTCCRC and GTCCRE registers, respectively. For double buffer operation, also set the GTIOCnA and GTIOCnB pins transitions in 2 cycles after the current cycle in the GTCCRD and GTCCRF registers, respectively.

Note: n: 0 to 15
m: A, B

5.7.3.3.2 Saw-Wave One-Shot Pulse Mode

The saw-wave one-shot pulse mode is a mode in which the cycle is set in GTPR, the GTCNT counter performs saw-wave (half-wave) operation and a PWM waveform is output to the GTIOCnA or GTIOCnB pin (n = 0 to 15) on a compare match with GTCCRA or GTCCRB with buffer operation fixed.

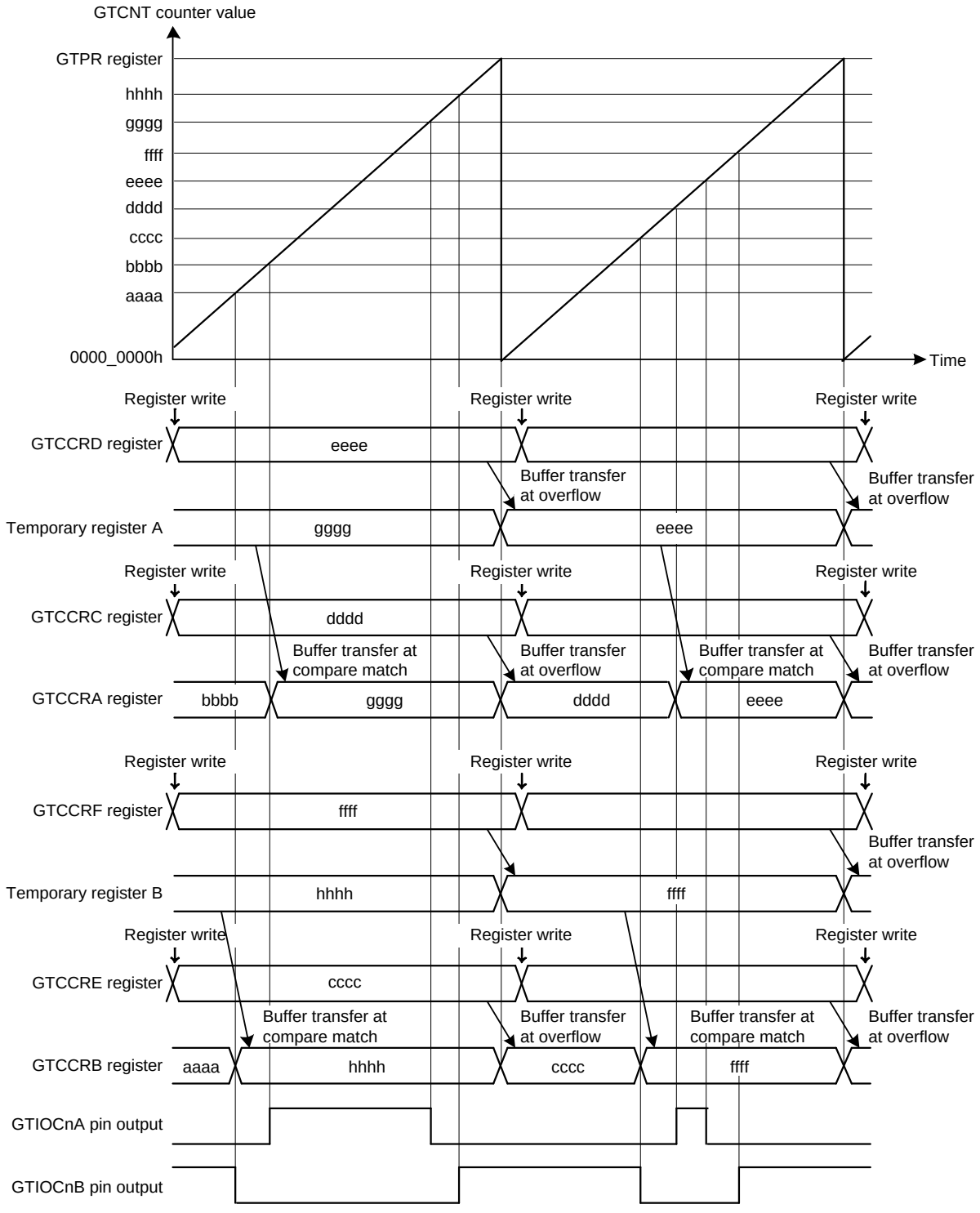
Buffer operation in saw-wave one-shot pulse mode is different from the usual buffer operation. Buffer transfer is performed as follows:

- From GTCCRC to GTCCRA at the cycle end
- From GTCCRE to GTCCRB at the cycle end
- From GTCCRD to temporary register A at the cycle end
- From GTCCRF to temporary register B at the cycle end
- From temporary register A to GTCCRA on a compare match with GTCCRA
- From temporary register B to GTCCRB on a compare match with GTCCRB

The pin output value can be selected from low-level output, high-level output, or toggled output separately for a compare match and the cycle end according to the GTIOR setting. When the GTBER.CCRSWT bit is set to 1 while counting is stopped, buffer transfer from the GTCCRD register to temporary register A and from the GTCCRF register to temporary register B is forcibly performed. By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time is automatically set in GTCCRB.

If the GTBER.DBRTECm (m = A, B) bit is set to 1b, transfer from an intermediate buffer to the GTCCRm (m = A, B) register is repeated on a cyclic basis by using temporary register x (x = C, E) and temporary register m (m = A, B) which serve as intermediate buffers for the GTCCRx (x = C, E) and GTCCRm (m = A, B) registers, respectively, even while buffer transfer is inhibited (the function for repeated double-buffered operation when buffer transfer is inhibited). For details, see **5.7.8.2.2 Repeated Double-Buffered Operation when GTCCR Buffer Transfer is Inhibited**.

Figure 5.7-24 shows an example of saw-wave one-shot pulse mode operation, and **Table 5.7-25** shows an example for setting saw-wave one-shot pulse mode.



Note: The base addresses of the registers are the same.

Figure 5.7-24 Example of Saw-Wave One-Shot Pulse Mode Operation with Up-Counting, Low-level Output from the GTIOCnA Pin and High-level Output from the GTIOCnB Pin at Count Start, Output Toggled at GTCCRA/ GTCCRB Compare Match, and Output Retained at Cycle End

Table 5.7-25 Example Setting for Saw-Wave One-Shot Pulse Mode

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-24**, 001b (saw-wave one-shot pulse mode) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-24**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter.
6	Set GTIOCnm pin function	Set the GTIOCnm pin function with the GTIOA[4:0] and GTIOB[4:0] bits in the GTIOR register. In **Figure 5.7-24**, GTIOA[4:0] = 0_0011b and GTIOB[4:0] = 1_0011b.
7	Enable GTIOCnm pin output	Set to enable the GTIOCnm pin output with the OAE and OBE bits in the GTIOR register.
8	Set compare match value	Set the GTIOCnA pin transition immediately after the count start in the GTCCRC and GTCCRD registers and the GTIOCnB pin transition in the GTCCRE and GTCCRF registers.
9	Set forcible buffer transfer	Set the GTBER.CCRSWT bit to 1b to transfer buffer register data forcibly.
10	Set buffer value	For buffer operation, set the GTIOCnA pin transition in one cycle after the current cycle in the GTCCRC and GTCCRD registers and the GTIOCnB pin transition in the GTCCRE and GTCCRF registers.
11	Start count operation	Set the GTCR.CST bit to 1b to start count operation.
12	Set buffer value for each cycle	For buffer operation, set the GTIOCnA pin transition in one cycle after the current cycle in the GTCCRC and GTCCRD registers and the GTIOCnB pin transition in the GTCCRE and GTCCRF registers.

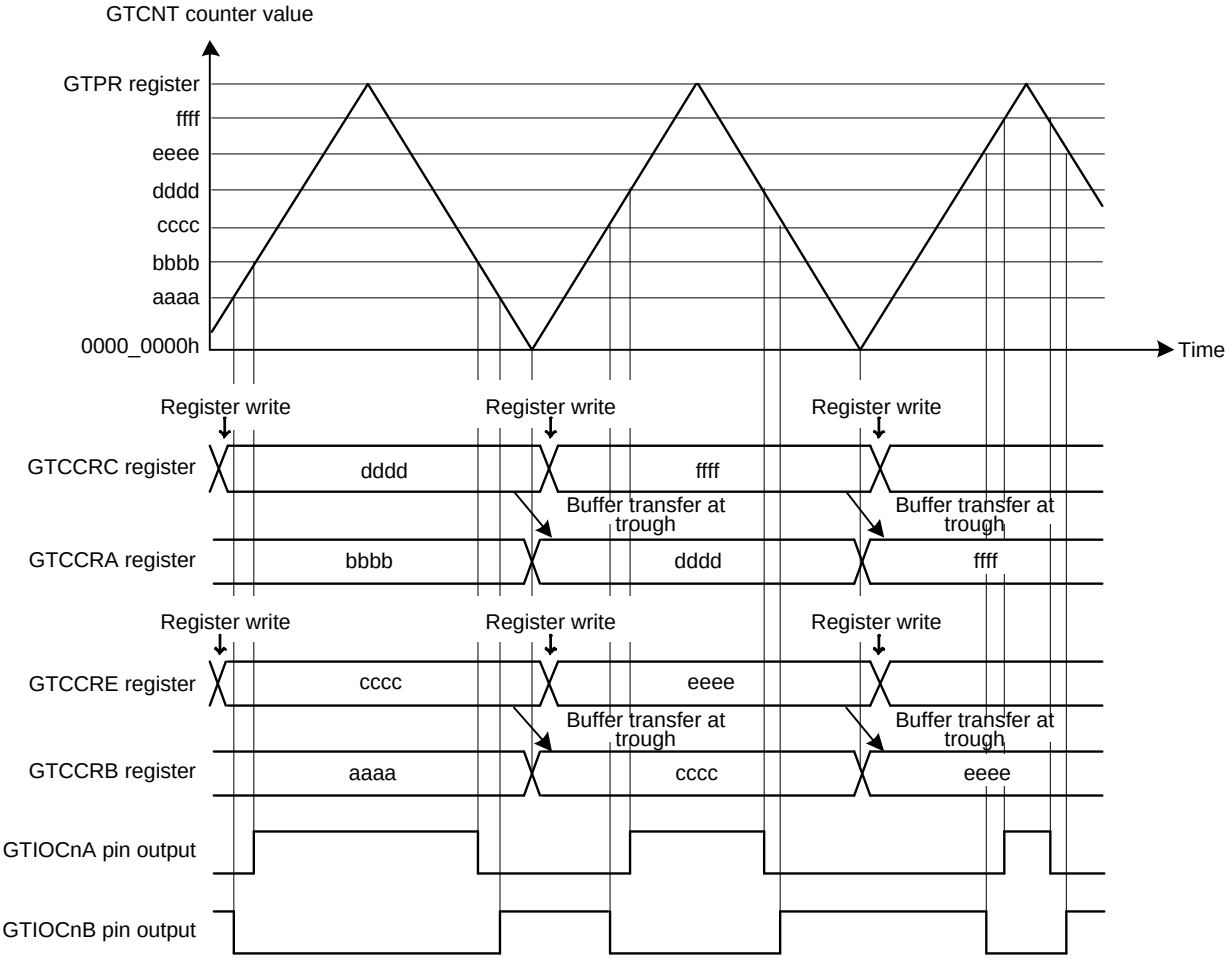
Note: n: 0 to 15
m: A, B

5.7.3.3.3 Triangle-Wave PWM Mode 1 (32-Bit Transfer at Trough)

The triangle-wave PWM mode 1 is a mode in which the cycle is set in GTPR. The GTCNT counter performs triangle-wave (full-wave) operation, and a PWM waveform is output to the GTIOCnA or GTIOCnB pin (n = 0 to 15) when a GTCCRA or GTCCRB compare match occurs. Buffer transfer is performed at troughs. The pin output value can be selected from low-level output, high-level output, or toggled output separately for a compare match and for the cycle end according to the GTIOR setting.

By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can be automatically set in GTCCRB.

Figure 5.7-25 shows an example of a triangle-wave PWM mode 1 operation, and **Table 5.7-26** shows an example for setting a triangle-wave PWM mode 1.



Note: The base addresses of the registers are the same.

Figure 5.7-25 Example of Triangle-Wave PWM Mode 1 Operation with Buffer Operation, Low-level Output from the GTIOCnA Pin and High-level Output from the GTIOCnB Pin at Count Start, Output Toggled at GTCCRA/GTCCRB Register Compare Match, and Output Retained at Cycle End

Table 5.7-26 Example Setting for Triangle-Wave PWM Mode 1

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-25**, 100b (triangle-wave PWM mode 1) is set.
2	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
3	Set cycle	Set the cycle in the GTPR register.
4	Set initial value for counter	Set the initial value in the GTCNT counter.
5	Set GTIOCnm pin function	Set the GTIOCnm pin function with the GTIOA[4:0] and GTIOB[4:0] bits in the GTIOR register. In **Figure 5.7-25**, GTIOA[4:0] = 0_0011b and GTIOB[4:0] = 1_0011b.
6	Enable GTIOCnm pin output	Set to enable the GTIOCnm pin output with the OAE and OBE bits in the GTIOR register.
7	Set buffer operation	Set buffer operation with the CCRA[1:0] and CCRB[1:0] bits in the GTBER register. In **Figure 5.7-25**, CCRA[1:0] = 01b and CCRB[1:0] = 01b.
8	Set compare match value	Set the GTIOCnA and GTIOCnB pins transitions in the GTCCRA and GTCCRB registers, respectively.
9	Set buffer value	For buffer operation, set the GTIOCnA and GTIOCnB pins transitions in 1 cycle after the current cycle in the GTCCRC and GTCCRE registers, respectively. For double buffer operation, also set the GTIOCnA and GTIOCnB pins transitions in 2 cycles after the current cycle in the GTCCRD and GTCCRF registers, respectively.
10	Start count operation	Set the GTCR.CST bit to 1b to start count operation.
11	Set buffer value for each cycle	For buffer operation, set the GTIOCnA and GTIOCnB pins transitions in 1 cycle after the current cycle in the GTCCRC and GTCCRE registers, respectively. For double buffer operation, also set the GTIOCnA and GTIOCnB pins transitions in 2 cycles after the current cycle in the GTCCRD and GTCCRF registers, respectively.

Note: n: 0 to 15
m: A, B

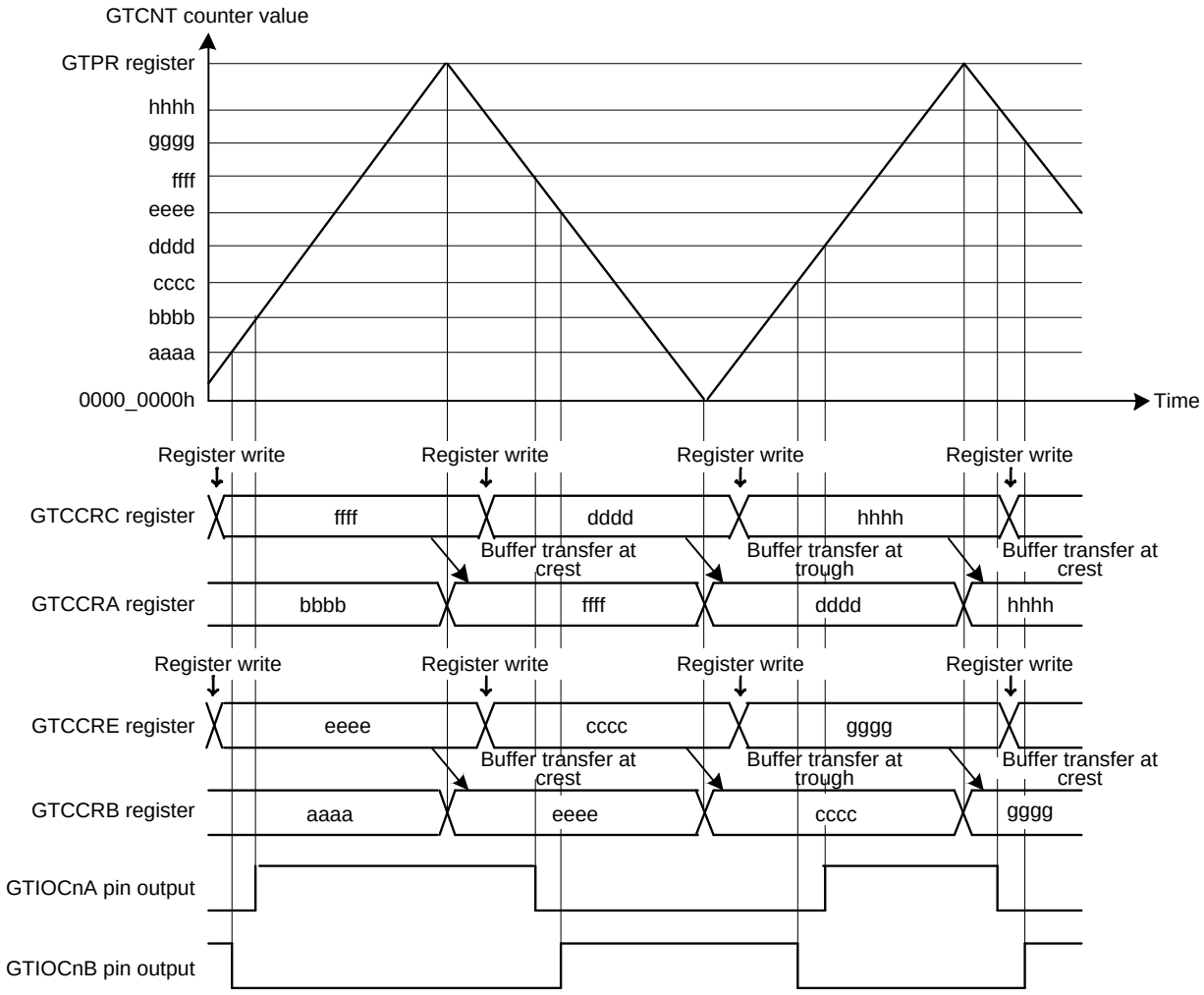
5.7.3.3.4 Triangle-Wave PWM Mode 2 (32-Bit Transfer at Crest and Trough)

Similar to triangle-wave PWM mode 1, in triangle-wave PWM mode 2 the cycle is set in GTPR. The GTCNT counter performs triangle-wave (full-wave) operation, and a PWM waveform is output to the GTIOCnA or GTIOCnB pin (n = 0 to 15) when a GTCCRA or GTCCRB compare match occurs. The buffer transfer is performed at both crests and troughs. The pin output value can be selected from low-level output, high-level output, or toggle output separately for a compare match and for the cycle end according to the GTIOR setting.

By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can be automatically set in GTCCRB.

If the GTCCRm (m = A, B) register is made to operate as a double buffer, when the GTBER.DBRTECm (m = A, B) bit is set to 1b, transfer from an intermediate buffer to the GTCCRm (m = A, B) register is repeated on a cyclic basis by using temporary register x (x = C, E) and the GTCCRx (x = C, E) register which serve as intermediate buffers for the GTCCRx (x = C, E) and GTCCRm (m = A, B) registers, respectively, even while buffer transfer is inhibited (the function for repeated double-buffered operation when buffer transfer is inhibited). For details, see **5.7.8.2.2 Repeated Double-Buffered Operation when GTCCR Buffer Transfer is Inhibited**.

Figure 5.7-26 shows an example of triangle-wave PWM mode 2 operation, and **Table 5.7-27** shows an example for setting triangle-wave PWM mode 2.



Note: The base addresses of the registers are the same.

Figure 5.7-26 Example of Triangle-Wave PWM Mode 2 Operation with Buffer Operation, Low-level Output from the GTIOCnA Pin and High-level Output from the GTIOCnB Pin at Count Start, Output Toggled at GTCCRA/ GTCCRB Compare Match, and Output Retained at Cycle End

Table 5.7-27 Example for Setting Triangle-Wave PWM Mode 2

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-26**, 101b (triangle-wave PWM mode 2) is set.
2	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
3	Set cycle	Set the cycle in the GTPR register.
4	Set initial value for counter	Set the initial value in the GTCNT counter.
5	Set GTIOCnm pin function	Set the GTIOCnm pin function with the GTIOA[4:0] and GTIOB[4:0] bits in the GTIOR register. In **Figure 5.7-26**, GTIOA[4:0] = 0_0011b and GTIOB[4:0] = 1_0011b.
6	Enable GTIOCnm pin output	Set to enable the GTIOCnm pin output with the OAE and OBE bits in the GTIOR register.
7	Set buffer operation	Set buffer operation with the CCRA[1:0] and CCRB[1:0] bits in the GTBER register. In **Figure 5.7-26**, CCRA[1:0] = 01b and CCRB[1:0] = 01b.
8	Set compare match value	Set the GTIOCnA and GTIOCnB pins transitions in the GTCCRA and GTCCRB registers, respectively.
9	Set buffer value	For buffer operation, set the GTIOCnA and GTIOCnB pins transitions in half cycle after the current cycle in the GTCCRC and GTCCRE registers, respectively. For double buffer operation, also set the GTIOCnA and GTIOCnB pins transitions in 1 cycle after the current cycle in the GTCCRD and GTCCRF registers, respectively.
10	Start count operation	Set the GTCR.CST bit to 1b to start count operation.
11	Set buffer value for each half cycle	For buffer operation, set the GTIOCnA and GTIOCnB pins transitions in half cycle after the current cycle in the GTCCRC and GTCCRE registers, respectively. For double buffer operation, also set the GTIOCnA and GTIOCnB pins transitions in 1 cycle after the current cycle in GTCCRD and GTCCRF registers, respectively.

Note: n: 0 to 15
m: A, B

5.7.3.3.5 Triangle-Wave PWM Mode 3 (64-Bit Transfer at Trough)

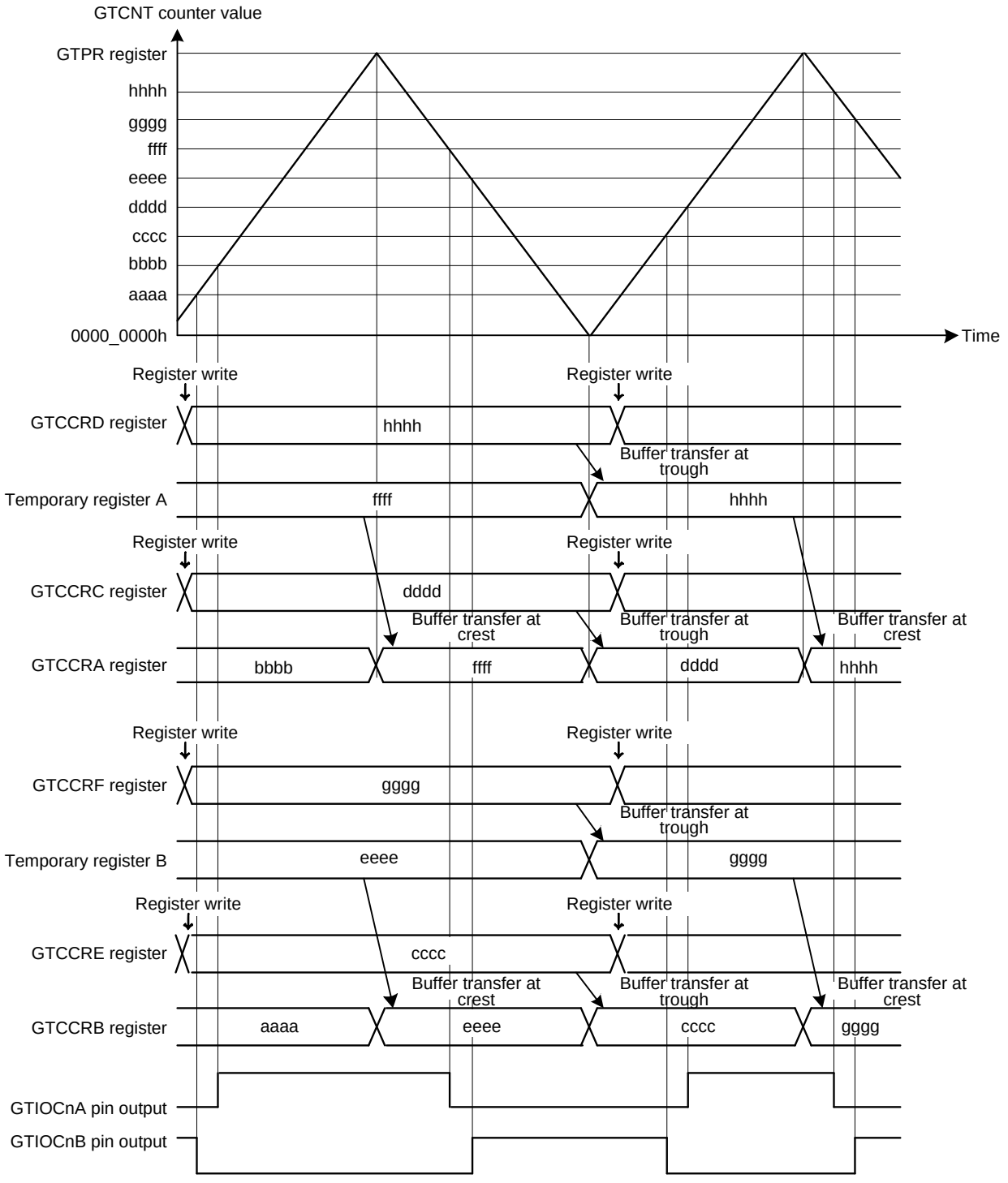
The triangle-wave PWM mode 3 is a mode in which the cycle is set in GTPR. The GTCNT counter performs triangle-wave (full-wave) operation and a PWM waveform is output to the GTIOCnA or GTIOCnB pin (n = 0 to 15) on a compare match with GTCCRA or GTCCRB with buffer operation fixed. Buffer operation in triangle-wave PWM mode 3 is different from usual buffer operation. Buffer transfer is performed as follows:

- From GTCCRC to GTCCRA at troughs
- From GTCCRE to GTCCRB at troughs
- From GTCCRD to temporary register A at troughs
- From GTCCRF to temporary register B at troughs
- From temporary register A to GTCCRA at crests
- From temporary register B to GTCCRB at crests

The pin output value can be selected from low-level output, high-level output, or toggled output separately for a compare match and for the cycle end according to the GTIOR setting.

By setting GTDTCR, GTDVU, and GTDVD, a compare match value for a negative-phase waveform with dead time can be automatically set in GTCCRB.

Figure 5.7-27 shows an example of triangle-wave PWM mode 3 operation, and **Table 5.7-28** shows an example for setting triangle-wave PWM mode 3.



Note: The base addresses of the registers are the same.

Figure 5.7-27 Example of Triangle-Wave PWM Mode 3 Operation with Low-level Output from the GTIOCnA Pin and High-level Output from the GTIOCnB Pin at Count Start, Output Toggled at GTCCRA/GTCCRB Compare Match, and Output Retained at Cycle End

Table 5.7-28 Example Setting for Triangle-Wave PWM Mode 3

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-27**, 110b (triangle-wave PWM mode 3) is set.
2	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
3	Set cycle	Set the cycle in the GTPR register.
4	Set initial value for counter	Set the initial value in the GTCNT counter.
5	Set GTIOCnm pin function	Set the GTIOCnm pin function with the GTIOA[4:0] and GTIOB[4:0] bits in the GTIOR register. In **Figure 5.7-27**, GTIOA[4:0] = 0_0011b and GTIOB[4:0] = 1_0011b.
6	Enable GTIOCnm pin output	Set to enable the GTIOCnm pin output with the OAE and OBE bits in the GTIOR register.
7	Set compare match value	Set the GTIOCnA pin transition immediately after the count start in the GTCCRC and GTCCRD registers and the GTIOCnB pin transition in the GTCCRE and GTCCRF registers.
8	Set forcible buffer transfer	Set the GTBER.CCRSWT bit to 1b to transfer buffer register data forcibly.
9	Set buffer value	Set the GTIOCnA pin transition in 1 cycle after the current cycle in the GTCCRC and GTCCRD registers and the GTIOCnB pin transition in the GTCCRE and GTCCRF registers.
10	Start count operation	Set the GTCR.CST bit to 1b to start count operation.
11	Set buffer value for each cycle	Set the GTIOCnA pin transition in 1 cycle after the current cycle in the GTCCRC and GTCCRD registers and the GTIOCnB pin transition in the GTCCRE and GTCCRF registers.

Note: n: 0 to 15
m: A, B

5.7.3.4 Automatic Dead Time Setting Function

By setting the GTDTCR register, a compare match value for a negative waveform with dead time obtained by a compare match value for a positive waveform (the GTCCRA register value) and specified dead time values (the GTDVU and GTDVD registers values) can be automatically set in the GTCCRB register.

The automatic dead time setting function can be used in saw-wave one-shot pulse mode and all the triangle PWM modes.

Dead time can be separately set for the first half and second half of a waveform. Dead time for the changing point in the first half of a negative waveform is set in the GTDVU register and that in the second half is set in the GTDVD register. The same dead time can also be set for the first and second halves by setting the GTDTCR.TDFER bit to 1b.

The GTDBU register can be used as a buffer register for the GTDVU register, and the GTDBD register can be used as a buffer register for the GTDVD register. Buffer transfer is performed at the end of the cycle (in saw-wave mode: either of an overflow of the GTCNT counter (up-counting), an underflow (down-counting), or the GTCNT counter clearing in triangle-wave mode: a trough).

The change point of the negative-phase waveform, which is automatically calculated, is obtained by reading the GTCCRB register. Writing to the GTCCRB register is prohibited when the automatic dead time setting function is used.

Do not set the dead-time that makes the change point of the waveform exceeding the count period. When a dead-time setting which would generate a dead-time error is made, adjust the change points of the positive- and negative-phase waveforms to generate waveforms with secured dead-time as shown in **Table 5.7-29**. The adjusted change point of the negative-phase waveform is automatically set in the GTCCRB register. An internal signal is used to determine the change point of the positive-phase waveform, thus the value of the GTCCRA register is not updated by the adjusted value.

In saw-wave one-shot pulse mode, if the order of the change point becomes inconsistent by adjustment of the waveform change point due to occurrence of dead time errors, or if the change point exceeds the count period even after the adjustment, the complementary relation between the positive- and negative-phases cannot be guaranteed.

In triangle-wave PWM mode, if dead-time exceeds the count period by setting 0000_0000h or a value greater than or equal to the setting value of the GTPR register is set in the GTCCRA register, output change is controlled by the output protection function (see **5.7.8.4 GTIOCnm Protection Function for GTIOCnm Pin Output (n = 0 to 15; m = A, B)**). When GTCCRA register is greater than or equal to [GTPR register + GTDVm (m = U, D) register], [GTPR register − 1] is set in the GTCCRB register as the upper limit.

Automatic setting for a dead time value to the GTCCRB register is performed at the next count clock after the register value for calculating the automatic setting value is updated. In triangle-wave mode, it can also be done at the next count clock from the current crest.

Table 5.7-29 Adjustment of the waveform change point when a dead-time error occurs

Mode	Count direction	Period	Condition for dead-time error	Change point of the positive-phase waveform after adjustment	Change point of the negative-phase waveform after adjustment
Saw-wave one-shot pulse mode	Up-counting	First half	GTCCRA − GTDVU < 0	GTDVU	0
		Second half	GTCCRA + GTDVD > GTPR	GTPR − GTDVD	GTPR
	Down-counting	First half	GTCCRA + GTDVU > GTPR	GTPR − GTDVU	GTPR
		Second half	GTCCRA − GTDVD < 0	GTDVD	0
Triangle-wave PWM mode 1/2/3	Up-counting	(First half)	GTCCRA − GTDVU ≤ 0	GTDVU + 1	1
	Down-counting	(Second half)	GTCCRA − GTDVD < 0	GTDVD	0

Figure 5.7-28 to **Figure 5.7-30** show examples of automatic dead time setting function operation.

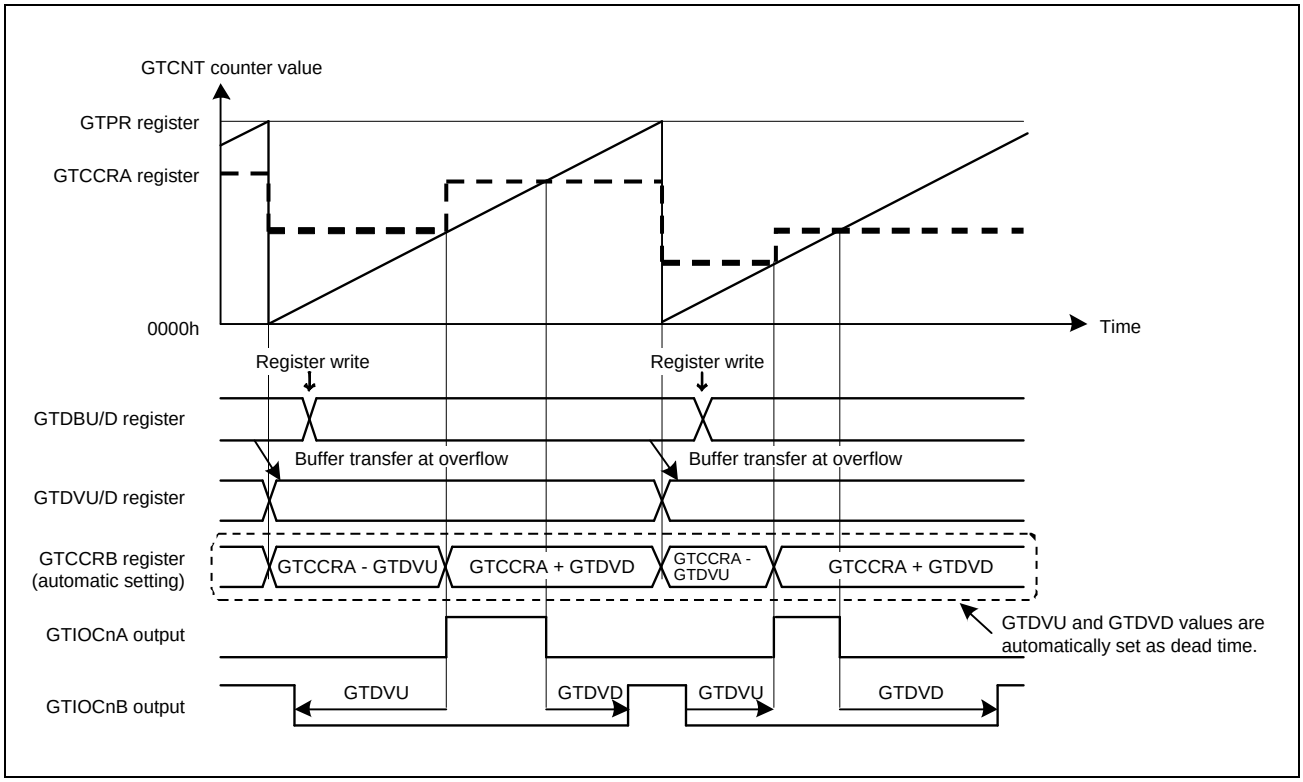


Figure 5.7-28 Example of automatic dead time setting function operation (saw-wave one-shot pulse mode, up-counting, GTDVU and GTDVD registers set to buffer operation, active-level: high) (n = 0 to 15)

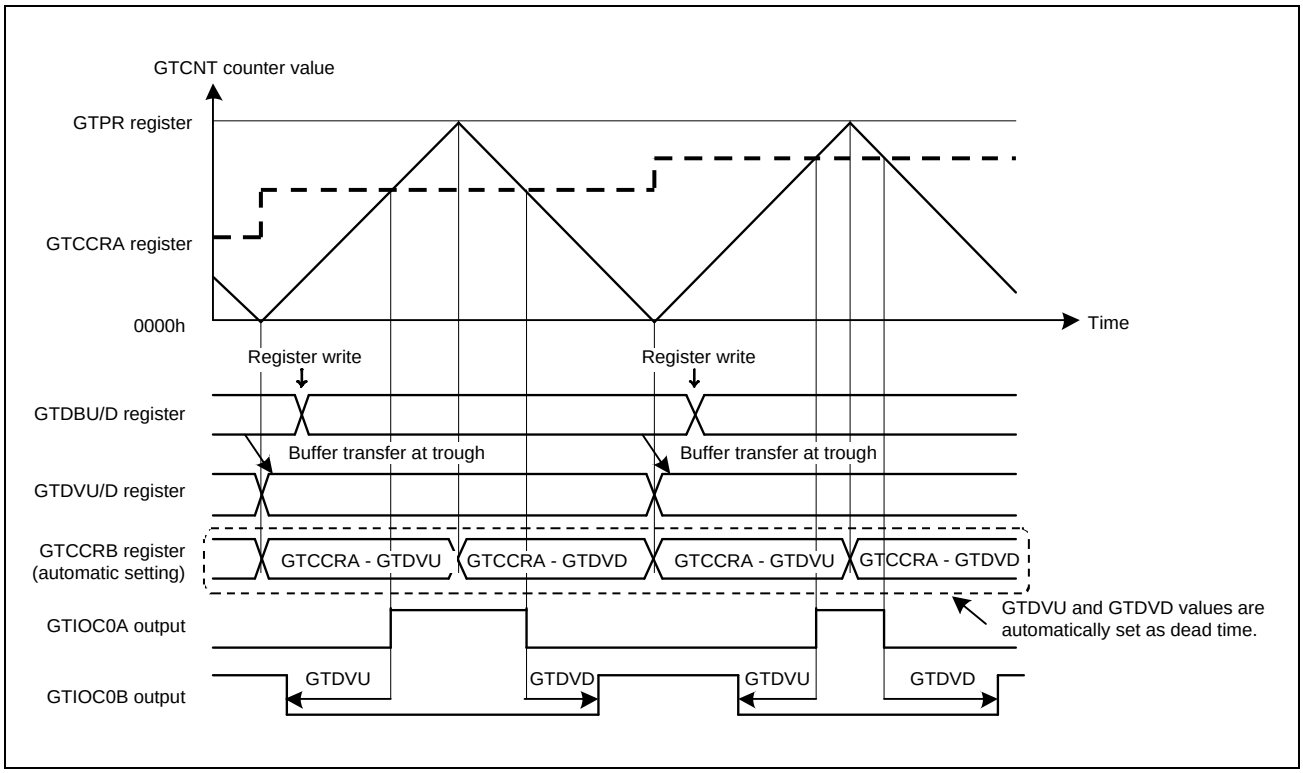


Figure 5.7-29 Example of automatic compare-match value setting function with dead time (triangle-wave PWM mode 1, GTDVU and GTDVD registers set to buffer operation, active-level: high)

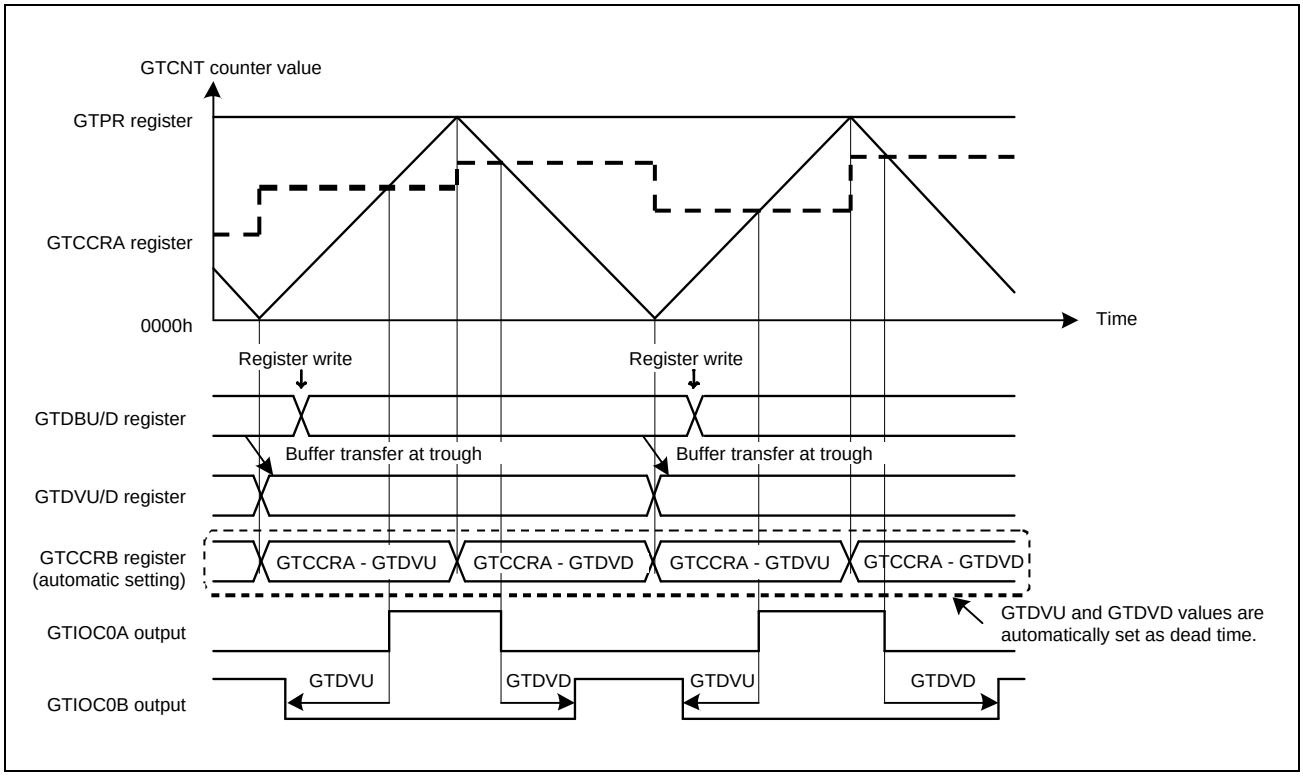


Figure 5.7-30 Example of automatic compare-match value setting function with dead time (triangle-wave PWM mode 2 or 3, GTDVU and GTDVD registers set to buffer operation, active-level: high)

The following example shows the settings for the automatic dead time setting function (saw-wave one-shot pulse mode, triangle-wave PWM mode 3). Note that n = 0 to 15 and m = A, B for the signals below.

1. Set operating mode
   Set the operating mode with the GTCR.MD[2:0] bits.
   In **Figure 5.7-28**, 001b (saw-wave one-shot pulse mode) is set. In **Figure 5.7-30**, 110b (triangle-wave PWM mode 3) is set.
2. Select count direction
   Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-28**, the lower 2 bits of the GTUDDTYC register is set to 11b and then to 01b for up counting.
3. Select count clock
   Select the count clock with GTCR.TPCS[3:0] bits.
4. Period setting
   Set a period in the GTPR register.
5. Set initial value for counter
   Set the initial value in the GTCNT counter.
6. Set GTIOCnm pin function
   Set the GTIOCnm pin function with the GTIOA[4:0] and GTIOB[4:0] bits in the GTIOR register. In **Figure 5.7-28** and **Figure 5.7-30**, GTIOA[4:0] bits = 0_0011b and GTIOB[4:0] bits = 1_0011b.
7. Enable GTIOCnm pin output
   Set to enable the GTIOCnm pin output with the OAE and OBE bits in the GTIOR register.
8. Set buffer value for compare match
   Set the GTIOCnA pin changing point immediately after the count start in the GTCCRC and GTCCRD registers.
9. Set forcible buffer transfer for compare match
   Set the GTBER.CCRSWT bit to 1b to transfer buffer register data forcibly to the GTCCRA register.
10. Set buffer value for compare match
    Set the GTIOCnA pin changing point in one cycle after the count start in the GTCCRC and GTCCRD registers.
11. Set automatic dead time setting function
    Set GTDTCR.TDE to 1b to enable the automatic dead time setting function.
12. Set buffer operation for dead time setting
    Set buffer operation with the TDBUE and TDBDE bits in the GTDTCR register.
13. Set dead time value
    Set the first half dead time value in the GTDVU register and the second half dead time in the GTDVD register. When the GTDVU register is set with the GTDTCR.TDFER bit set to 1b, the same value is also set to the GTDVD register, the same dead time value can be set for the first and second halves.
14. Set buffer value for dead time
    For buffer operation, set the first half dead time in one cycle after the current cycle in the GTDBU register and the second half dead time in the GTDBD register.
15. Set count operation
    Set the GTCR.CST bit to 1b to start count operation.

16. Set buffer value by cycle
    Set the GTIOCnA pin changing point for the next cycle from the current cycle to the GTCCRC and GTCCRD registers. When the dead time register is used for buffer operation, set the dead time value in the first half of the next cycle from the current cycle to the GTDBU register and the dead time value in the second half to the GTDBD register.

The following example shows the settings for the automatic dead time setting function (triangle-wave PWM mode 1 or 2). Note that n = 0 to 15 and m = A, B for the signals below.

1. Set operating mode
   Set the operating mode with the GTCR.MD[2:0] bits.
   In **Figure 5.7-29**, 100b (triangle-wave PWM mode 1) is set.
   In **Figure 5.7-30**, 101b (triangle-wave PWM mode 2) is set.
2. Select count clock
   Select the count clock with GTCR.TPCS[3:0] bits.
3. Period setting
   Set a period in the GTPR register.
4. Set initial value for counter
   Set the initial value in the GTCNT counter.
5. Set GTIOCnm pin function
   Set the GTIOCnm pin function with the GTIOA[4:0] and GTIOB[4:0] bits in the GTIOR register.
   In **Figure 5.7-29** and **Figure 5.7-30**, GTIOA[4:0] bits = 0_0011b and GTIOB[4:0] bits = 1_0011b.
6. Enable GTIOCnm pin output
   Set to enable the GTIOCnm pin output with the OAE and OBE bits in the GTIOR register.
7. Set buffer operation for compare match
   Set buffer operation with the GTBER.CCRA[1:0] bits.
8. Set the value for compare match
   Set the GTIOCnA pin changing point to the GTCCRA register.
9. Set buffer value for compare match
   For buffer operation, set the GTIOCnA pin changing point in one cycle after the current cycle (in triangle-wave PWM mode 1) or half cycle after the current cycle (in triangle-wave PWM mode 2) in the GTCCRC register. For double buffer operation, also set the GTIOCnA pin changing point in two cycles after the current cycle (in triangle-wave PWM mode 1) or one cycle after the current cycle (in triangle-wave PWM mode 2) in the GTCCRD register.
10. Set automatic dead time setting function
    Set GTDTCR.TDE to 1b to enable the automatic dead time setting function.
11. Set buffer operation for dead time setting
    Set buffer operation with the TDBUE and TDBDE bits in the GTDTCR register.
12. Set dead time value
    Set the first half dead time value in the GTDVU register and the second half dead time in the GTDVD register. When the GTDVU register is set with the GTDTCR.TDFER bit set to 1b, the same value is also set to the GTDVD register, and the same dead time value can be set for the first and second halves.

13. Set buffer value for dead time
    For buffer operation, set the first half dead time in one cycle after the current cycle in the GTDBU register and the second half dead time in the GTDBD register.
14. Set count operation
    Set the GTCR.CST bit to 1b to start count operation.
15. Set buffer value by cycle
    For buffer operation of compare match registers, set the GTIOCnA pin changing point in one cycle after the current cycle (in triangle-wave PWM mode 1) or half cycle after the current cycle (in triangle-wave PWM mode 2) in the GTCCRC register. For double buffered operation of compare match registers, set the GTICCnA pin changing point in two cycles after the current cycle (in triangle-wave PWM mode 1) or one cycle after the current cycle (in triangle-wave PWM mode 2) in the GTCCRD register.
    In the same way, set the dead time value in the first half of the cycle after the current cycle in the GTDBU register and the dead time in the second half in the GTDBD register.

5.7.3.5 Count Direction Changing Function

The count direction of the GTCNT counter can be changed by modifying the UD bit in GTUDDTYC.

In saw-wave mode, if the UD bit in GTUDDTYC is modified during counting, the count direction is changed on overflow (when modified during up-counting) or underflow (when modified during down-counting). If the GTUDDTYC.UD bit is modified while counting is stopped and the GTUDDTYC.UDF bit is 0b, the GTUDDTYC.UD bit modification is not reflected at the start of counting and the count direction is changed on overflow or underflow. If the UDF bit is set to 1b while counting is stopped, the GTUDDTYC.UD bit value at that time is reflected at the start of counting.

In triangle-wave mode, the count direction is not changed even though the UD bit in GTUDDTYC is modified during counting. Similarly, even though the GTUDDTYC.UD bit is modified while counting is stopped and GTUDDTYC.UDF bit is 0b, the GTUDDTYC.UD bit value is not reflected in counting operation. If the GTUDDTYC.UDF bit is set to 1b while counting is stopped, the GTUDDTYC.UD bit value at that time is reflected at the start of counting.

If the count direction is changed during a saw-wave count operation, the GTPR value after the start of up-counting is reflected in the counting cycle during up-counting and the GTPR value after the start of down-counting is reflected in the counting cycle during down-counting.

Figure 5.7-31 shows an example of count direction changing function operation.

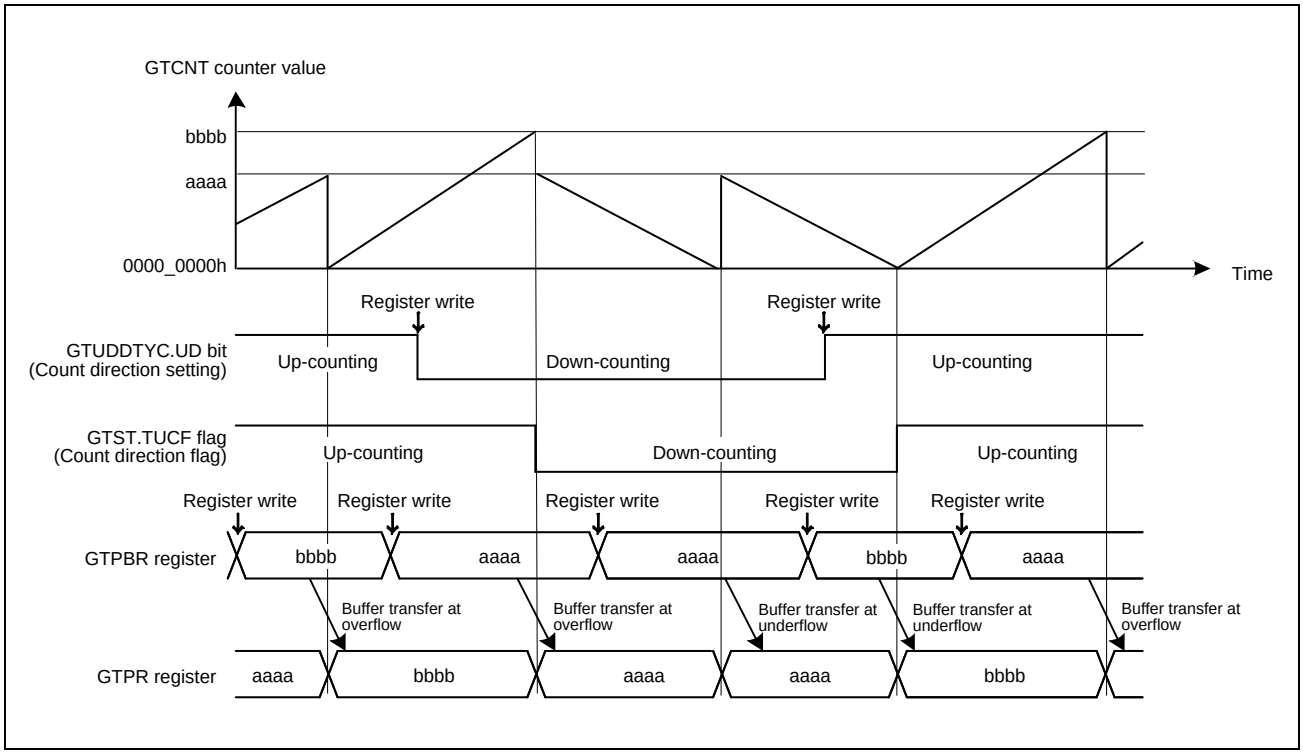


Figure 5.7-31 Example of a Count Direction Changing Function Operation during Buffer Operation

5.7.3.6 Function of Output Duty 0% and 100%

The output duty cycle of the GTIOCnA and GTIOCnB pins (n = 0 to 15) is set to 0% or 100% by changing the GTUDDTYC.OADTY or GTUDDTYC.OBDTY bit.

In saw-wave mode, if the GTUDDTYC.OADTY or GTUDDTYC.OBDTY bit is modified during counting, the output duty-cycle setting is reflected on overflow (when modified during up-counting) or underflow (when modified during down-counting). If the GTUDDTYC.OADTY or GTUDDTYC.OBDTY bit is modified while counting is stopped and the GTUDDTYC.OADTYF or the GTUDDTYC.OBDTYF bit is 0b, the output duty-cycle modification is not reflected at the start of counting. The output duty cycle changes on overflow or underflow. If the GTUDDTYC.OADTY or GTUDDTYC.OBDTY bit is modified while counting is stopped and the GTUDDTYC.OADTYF or GTUDDTYC.OBDTYF bit is 1b, the GTUDDTYC.OADTY or GTUDDTYC.OBDTY bit value at that time is reflected at the start of counting.

In triangle-wave mode, if the GTUDDTYC.OADTY or GTUDDTYC.OBDTY bit is modified during counting, the output duty-cycle setting is reflected on underflow.

If the GTUDDTYC.OADTY or GTUDDTYC.OBDTY bit is modified while counting is stopped and the GTUDDTYC.OADTYF or GTUDDTYC.OBDTYF bit is 0, the output duty-cycle modification is not reflected at the start of counting. The output duty cycle changes on underflow. If the GTUDDTYC.OADTY or GTUDDTYC.OBDTY bit is modified while counting is stopped and the GTUDDTYC.OADTYF or GTUDDTYC.OBDTYF bit is 1b, the output duty-cycle modification is reflected at the start of counting.

In performing 0% or 100% duty-cycle operation, GPT internally continues the following operations:

- Perform compare match operation
- Set compare match flag
- Output interrupts
- Perform buffer operation.

When control is changed from the 0% or 100% duty-cycle setting to the compare match, the output value of GTIOCnA pin at the cycle end is decided by GTIOR.GTIOA[3:2] and GTUDDTYC.OADTYR. The output value of GTIOCnB pin at the cycle end is decided by GTIOR.GTIOB[3:2] and GTUDDTYC.OBDTYR.

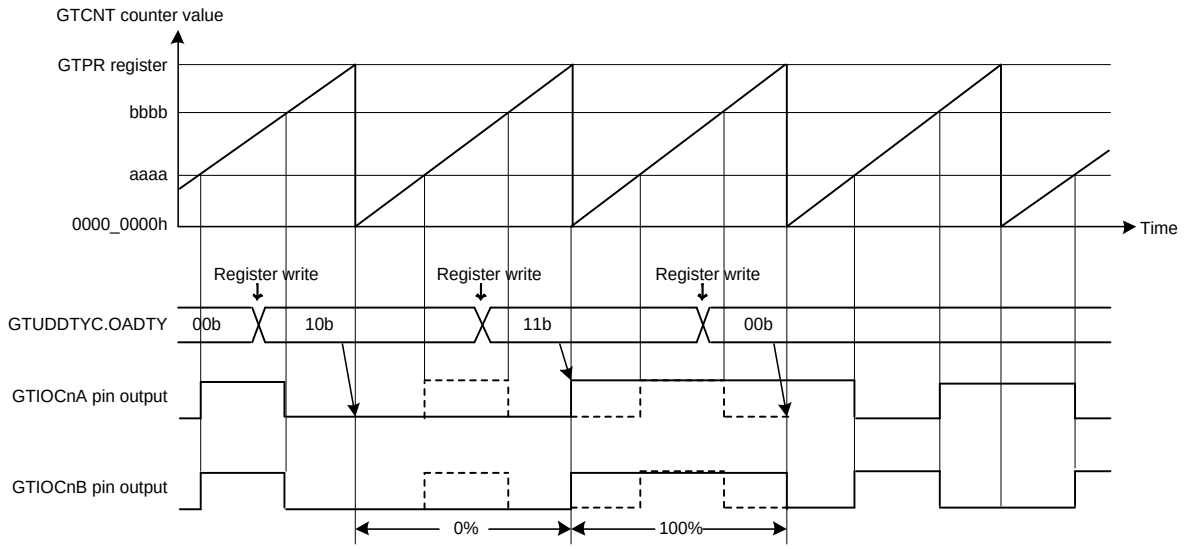
When GTIOR.GTIOA[3:2] and GTIOR.GTIOB[3:2] are set to 01b, the output pins output the low level at the cycle end. When GTIOR.GTIOA[3:2] and GTIOR.GTIOB[3:2] are set to 10b, the output pins output the high level at the cycle end.

GTUDDTYC.OADTYR selects the target value for output retained/toggled at the cycle end, when GTIOR.GTIOm[3:2] are set to 00b (output retained at the cycle end) or when GTIOR.GTIOm[3:2] are set to 11b (output toggled at the cycle end). **Table 5.7-30** shows the output values of the GTIOCnA and GTIOCnB pins at the cycle end.

Table 5.7-30 Output Values after Release from 0% or 100% Duty-Cycle Setting (m = A, B)

GTIOR.GTIOm[3:2]	Compare Match Value at Cycle End Masked by 0% or 100% Duty-Cycle Setting	GTUDDTYC.OmDTYR in Duty 0% Setting		GTUDDTYC.OmDTYR in Duty 100% Setting	
		0	1	0	1
00b (output retained at cycle end)	0	0	0	1	0
	1	0	1	1	1
01b (low-level output at cycle end)	—	0	0	0	0
10b (high-level output at cycle end)	—	1	1	1	1
11b (output toggled at cycle end)	0	1	1	0	1
	1	1	0	0	0

Figure 5.7-32 shows an example of output duty-cycle 0% and 100% function.



[Setting examples]

GTIOR.GTIOA[4:0] bits: 0_0011b
Initial low-level output, output toggled at compare match, output retained at cycle end

GTUDDTYC.OADTYR bit: 0b
The function selected by GTIOA[3:2] is applied to the output value when the duty cycle is set after release from 0% or 100% duty-cycle setting.

GTIOR.GTIOB[4:0] bits: 0_0011b
Initial low-level output, output toggled at compare match, output retained at cycle end

GTUDDTYC.OBDTYR bit: 1
The function of GTIOB[3:2] is applied to the compare match output value which is masked after 0% or 100% duty-cycle setting release.

Note: The base addresses of the registers are the same.

Figure 5.7-32 Example of Output Duty 0% and 100% Function

5.7.3.7 Hardware Count Start/Count Stop and Clear Operation

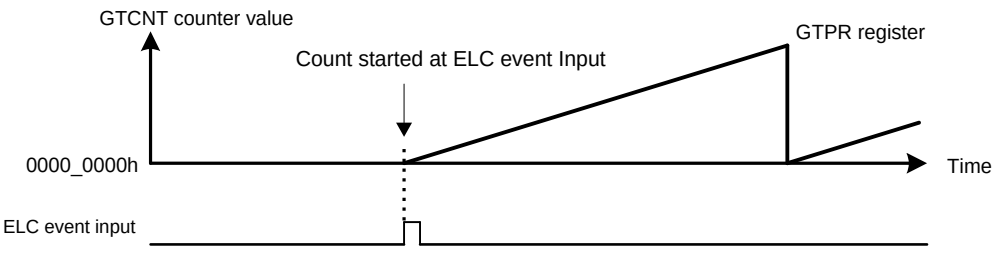
The GTCNT counter can be started, stopped, or cleared by the following hardware sources:

- External trigger input
- ELC event input
- GTIOCnA and GTIOCnB pin input (n = 0 to 15)

5.7.3.7.1 Hardware Start Operation

The GTCNT counter can be started by selecting a hardware source using GTSSR.

Figure 5.7-33 shows an example of count start operation by a hardware source. **Table 5.7-31** shows the setting example.



Note: ELC event input: ELC_GPTA event input

Figure 5.7-33 Example of Count Start Operation by Hardware Source (Starting at the Input of the Signal from the ELC_GPTA Event)

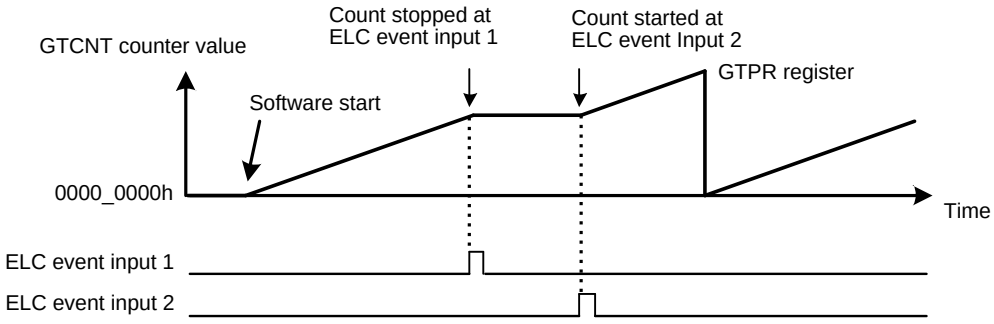
Table 5.7-31 Example Setting for Count Start Operation by Hardware Source

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-33**, 000b (saw-wave PWM mode) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-33**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter. In **Figure 5.7-33**, 0000_0000h is set.
6	Set hardware count start	Select a hardware source for starting count operation in the GTSSR register. In **Figure 5.7-33**, GTSSR.SSELCA = 1
7	Set hardware source operation	Set operation of the hardware source selected by the GTSSR register and start counting. In **Figure 5.7-33**, the ELC_GPTA event input operation is set.

5.7.3.7.2 Hardware Stop Operation

The GTCNT counter can be stopped by selecting a hardware source using GTPSR.

Figure 5.7-34 shows an example of count stop operation by a hardware source. **Table 5.7-32** shows the setting example. In this example, counting stops at the ELC_GPTA event input and restarts at the ELC_GPTB event input.



Note: ELC event input 1: ELC_GPTA event input
ELC event input 2: ELC_GPTB event input

Figure 5.7-34 Example of Count Stop Operation by Hardware Source (Starting by Software, Stopping at ELC_GPTA Input, and Restarting at ELC_GPTB Input)

Table 5.7-32 Example Setting for Count Stop Operation by Hardware Source

No.	Step Name	Description
1	Set operating mode	Set the operating mode with GTCR.MD[2:0]. In **Figure 5.7-34**, 000b (saw-wave PWM mode) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-34**, after 11b is set in GTUDDTYC[1:0], 01b is set in GTUDDTYC[1:0] (up-counting).
3	Select count clock	Select the count clock with GTCR.TPCS[3:0].
4	Set cycle	Set the cycle in GTPR.
5	Set initial value for counter	Set the initial value in the GTCNT counter. In **Figure 5.7-34**, 0000_0000h is set.
6	Set hardware count start	Select a hardware source for starting count operation in GTSSR register, and wait for count start by the hardware source. In **Figure 5.7-34**, GTSSR.SSELCB = 1b.
7	Set hardware count stop	Select a hardware source for stopping count operation in GTPSR register and wait for count stop by the hardware source. In **Figure 5.7-34**, GTPSR.PSELCA = 1b.
8	Set hardware source operation	Set operation of the hardware source selected in GTSSR register or GTPSR register, and start or stop counting. In **Figure 5.7-34**, ELC_GPTA input operation and ELC_GPTB input operation are set.

Figure 5.7-35 shows an example of count start/stop operation by a hardware source. **Table 5.7-33** shows the setting example. In this example, the counter operates during the high-level periods of the external trigger input GTETRGA.

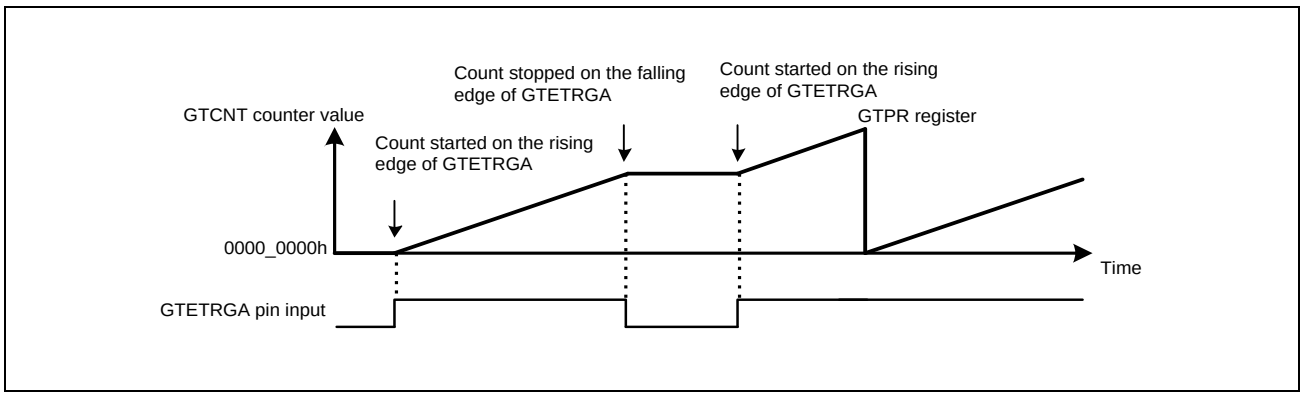


Figure 5.7-35 Example of Count Start/Stop Operation by Hardware Source (Starting on the Rising Edge of GTETRGA Pin Input, and Stopping on the Falling Edge of GTETRGA Pin Input)

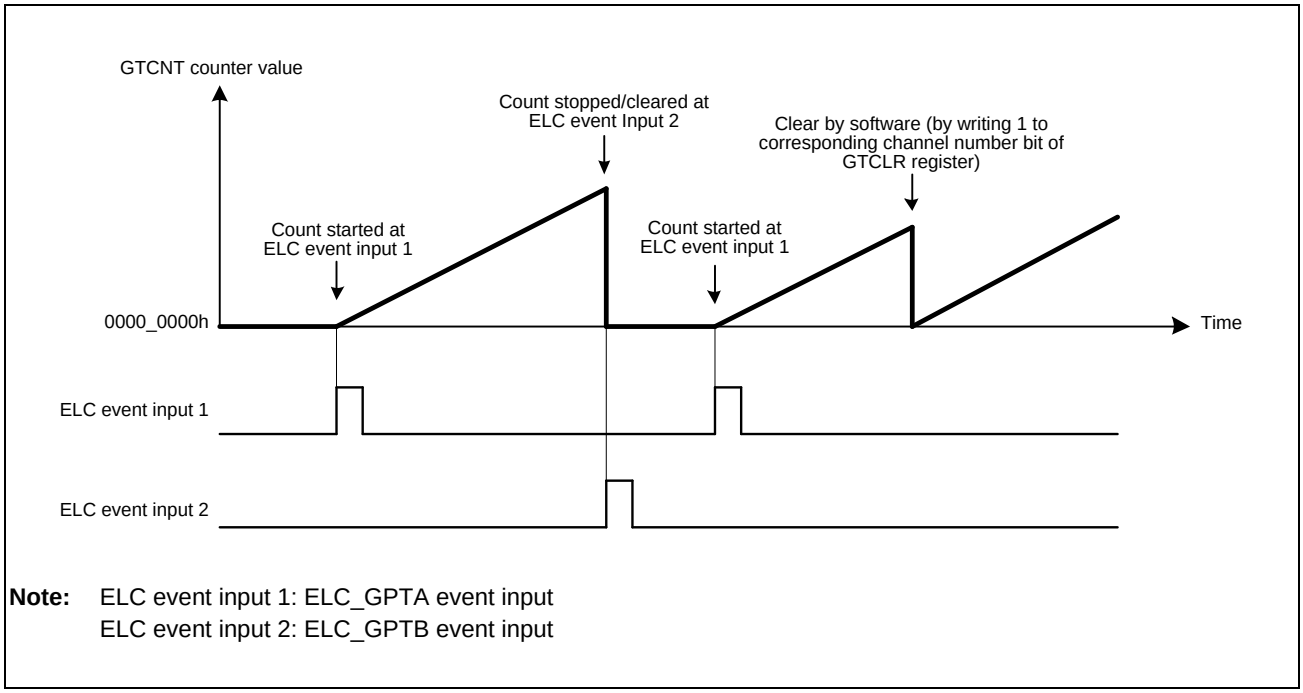
Table 5.7-33 Example Setting for Count Start/Stop Operation by Hardware Source

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-35**, 000b (saw-wave PWM mode) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-35**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter. In **Figure 5.7-35**, 0000_0000h is set.
6	Set hardware count start	Select a hardware source for starting count operation with the GTSSR register, and wait for count start by the hardware source. In **Figure 5.7-35**, GTSSR.SSGTRGAR = 1b.
7	Set hardware count stop	Select a hardware source for stopping count operation with the GTPSR register, and wait for count stop by the hardware source. In **Figure 5.7-35**, GTPSR.PSGTRGAF = 1b.
8	Set hardware source operation	Set operation of the hardware source selected in the GTSSR register or GTPSR register and start or stop counting. In **Figure 5.7-35**, the GTETRGA pin operation is set.

5.7.3.7.3 Hardware Clear Operation

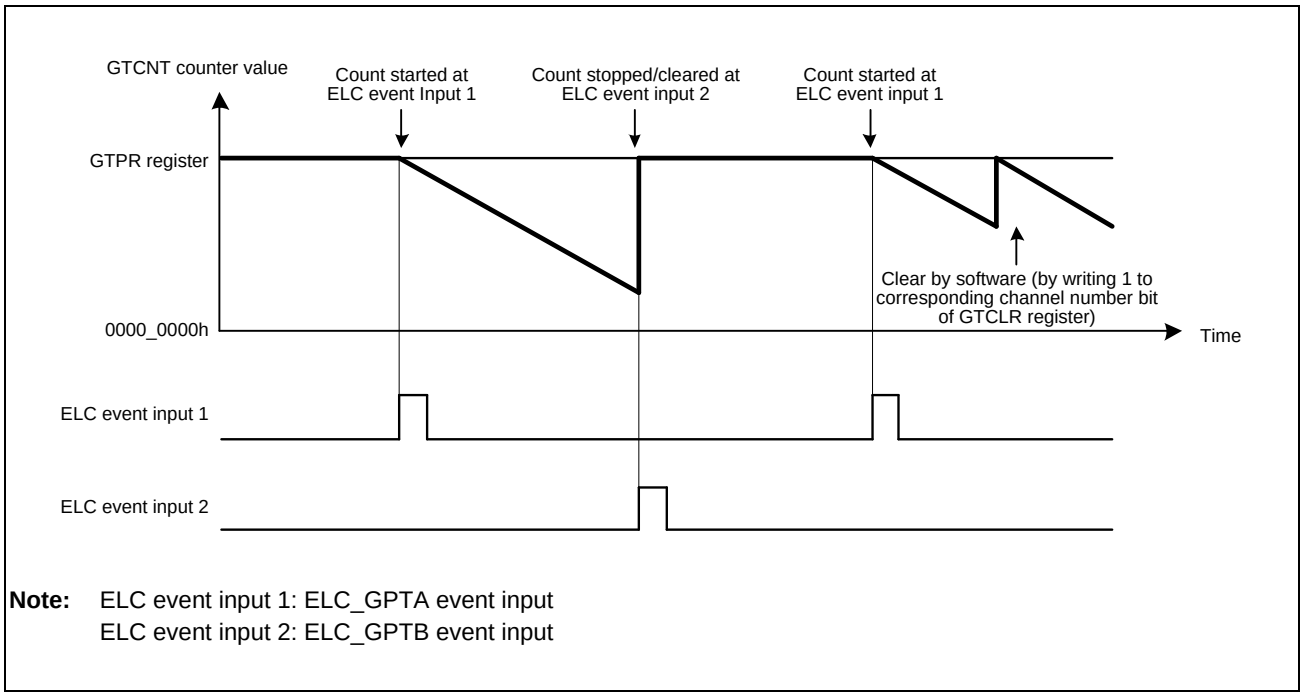
The GTCNT counter can be cleared by selecting a hardware source using GTCSR. The GPT_Ux_gpt_gtciv_n_m/GPT_Ux_gpt_gtciu_n_m (m = 0 to 7) interrupt (overflow/underflow interrupt) is not generated when the GTCNT counter is cleared by a hardware source or by software.

Figure 5.7-36 and **Figure 5.7-37** show examples of clearing the GTCNT counter by a hardware source. **Table 5.7-34** shows the setting example. In this example, the GTCNT counter starts at the ELC_GPTA input, and the counter is stopped and cleared at the ELC_GPTB input.



Note: ELC event input 1: ELC_GPTA event input
ELC event input 2: ELC_GPTB event input

Figure 5.7-36 Examples of Count Clearing Operation by Hardware Source in Saw Wave Up-Counting (Starting at ELC_GPTA Input, and Stopping/Clearing at ELC_GPTB Input)



Note: ELC event input 1: ELC_GPTA event input
ELC event input 2: ELC_GPTB event input

Figure 5.7-37 Examples of Count Clearing Operation by Hardware Source in Saw Wave Down-Counting (Starting at ELC_GPTA Input, and Stopping/Clearing at ELC_GPTB Input)

Table 5.7-34 Example Setting for Count Clearing Operation by Hardware Source

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-36** and **Figure 5.7-37**, 000b (saw-wave PWM mode) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-36**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting). In **Figure 5.7-37**, after 10b is set in the GTUDDTYC[1:0] bits, 00b is set in the GTUDDTYC[1:0] bits (down-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter. In **Figure 5.7-36**, 0000_0000h is set. In **Figure 5.7-37**, the GTPR register value is set.
6	Set hardware count start	Select a hardware source for starting count operation in the GTSSR register, and wait for count start by the hardware source. In **Figure 5.7-36** and **Figure 5.7-37**, GTSSR.SSELCA = 1b.
7	Set hardware count stop	Select a hardware source for stopping count operation in the GTPSR register, and wait for count stop by the hardware source. In **Figure 5.7-36** and **Figure 5.7-37**, GTPSR.PSELCB = 1b.
8	Set hardware count clear	Select a hardware source for clearing count operation in the GTCSR register, and wait for count clear by the hardware source. In **Figure 5.7-36** and **Figure 5.7-37**, GTCSR.CSELCB = 1b.
9	Set hardware source operation	Set operation of the hardware source selected in the GTSSR register, GTPSR register or GTCSR register and start, stop or clear counting. In **Figure 5.7-36** and **Figure 5.7-37**, the ELC_GPTA input and ELC_GPTB input are set.

The GPT_Ux_gpt_gtciv_n_m/GPT_Ux_gpt_gtciu_n_m (m = 0 to 7) interrupt (overflow/underflow interrupt) is not generated when the counter is cleared by a hardware source or by software.

Figure 5.7-38 shows the relationship between counter clearing by a hardware source and the GPT_Ux_gpt_gtciv_n_m interrupt.

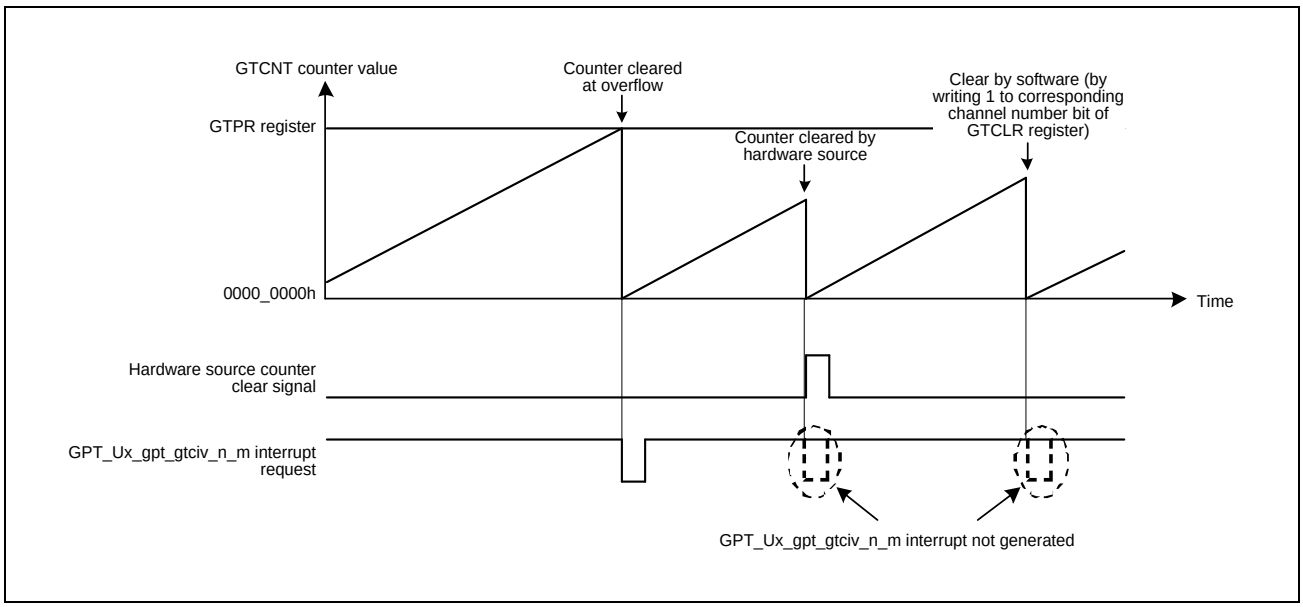


Figure 5.7-38 Relationship between Counter Clearing by Hardware Source and GPT_Ux_gpt_gtciv_n_m Interrupt

5.7.3.8 Synchronized Operation

Synchronized operation on channels such as synchronized starting, stopping, and clearing can be performed.

5.7.3.8.1 Synchronized Operation by Software

The GTCNT counters can be started, stopped, and cleared on multiple channels by setting the corresponding GTSTR, GTSTP, or GTCLR bits simultaneously to 1b.

Count start with a phase difference is possible by setting the initial value in the GTCNT counter and setting the corresponding GTSTR bits simultaneously to 1b.

Figure 5.7-39 shows an example of simultaneous starting, stopping, and clearing by software. **Figure 5.7-40** shows an example of phase start operation by software.

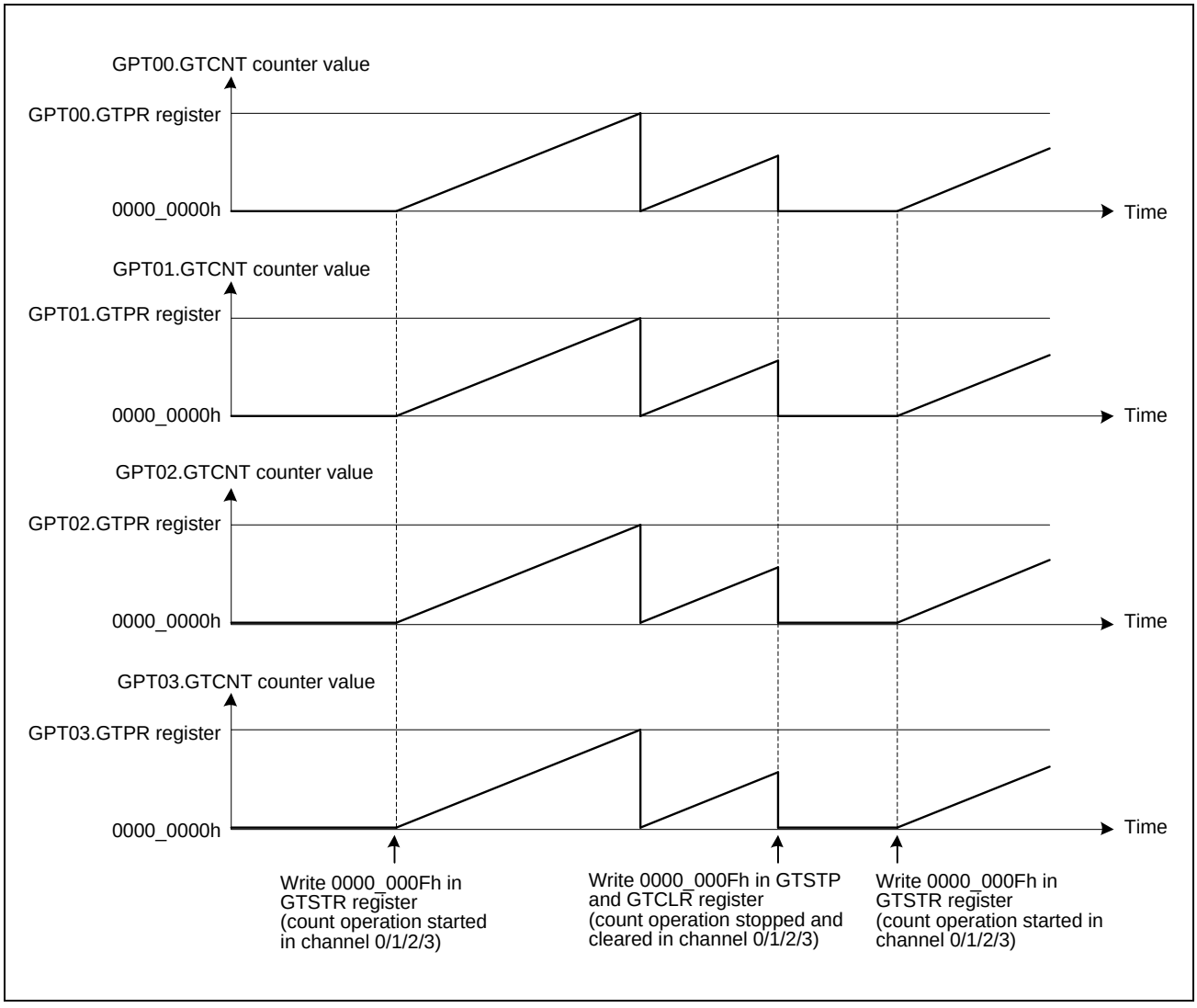


Figure 5.7-39 Example of Simultaneous Starting, Stopping, and Clearing by Software with the Same Counting Cycle (GTPR Register Value)

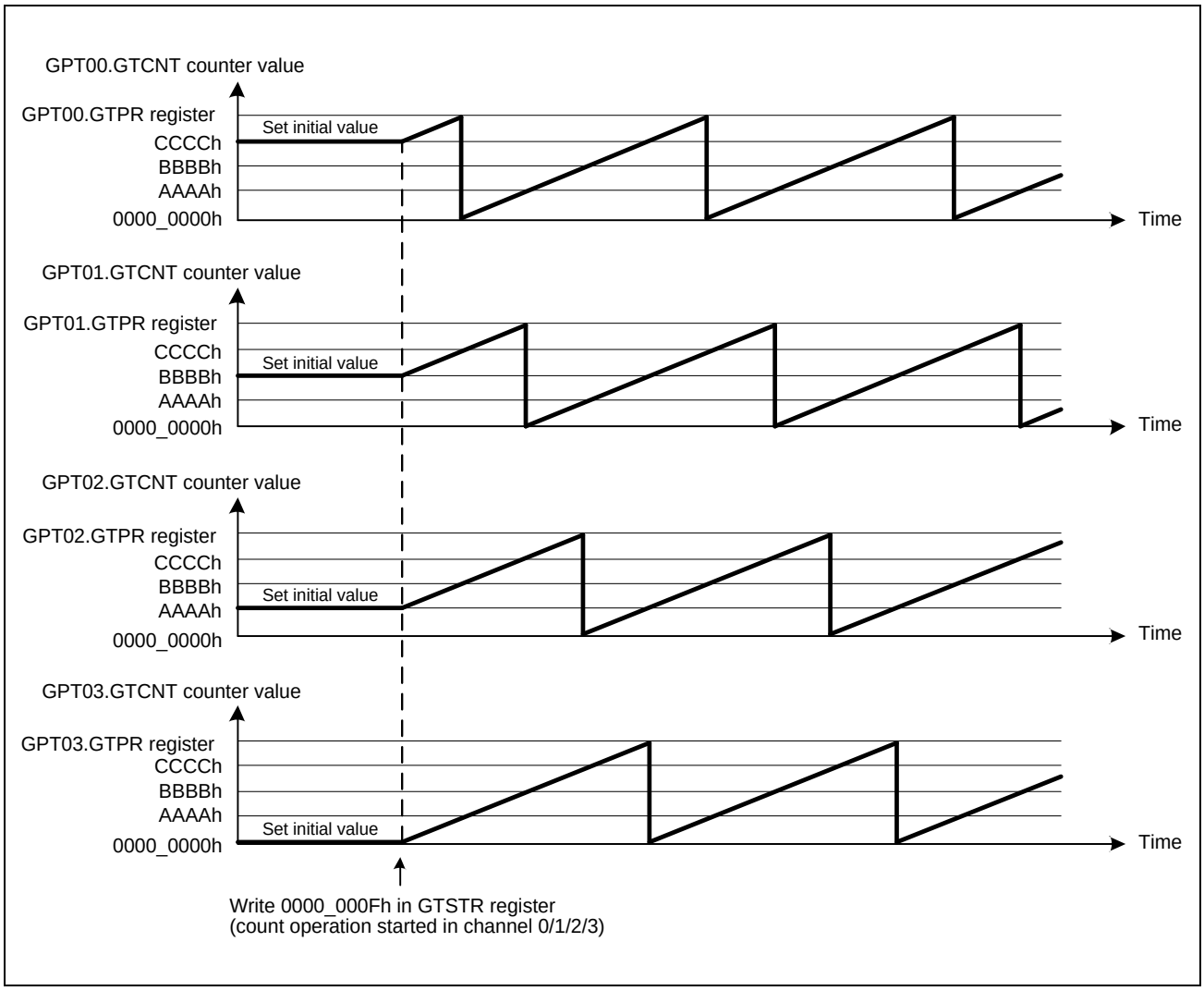
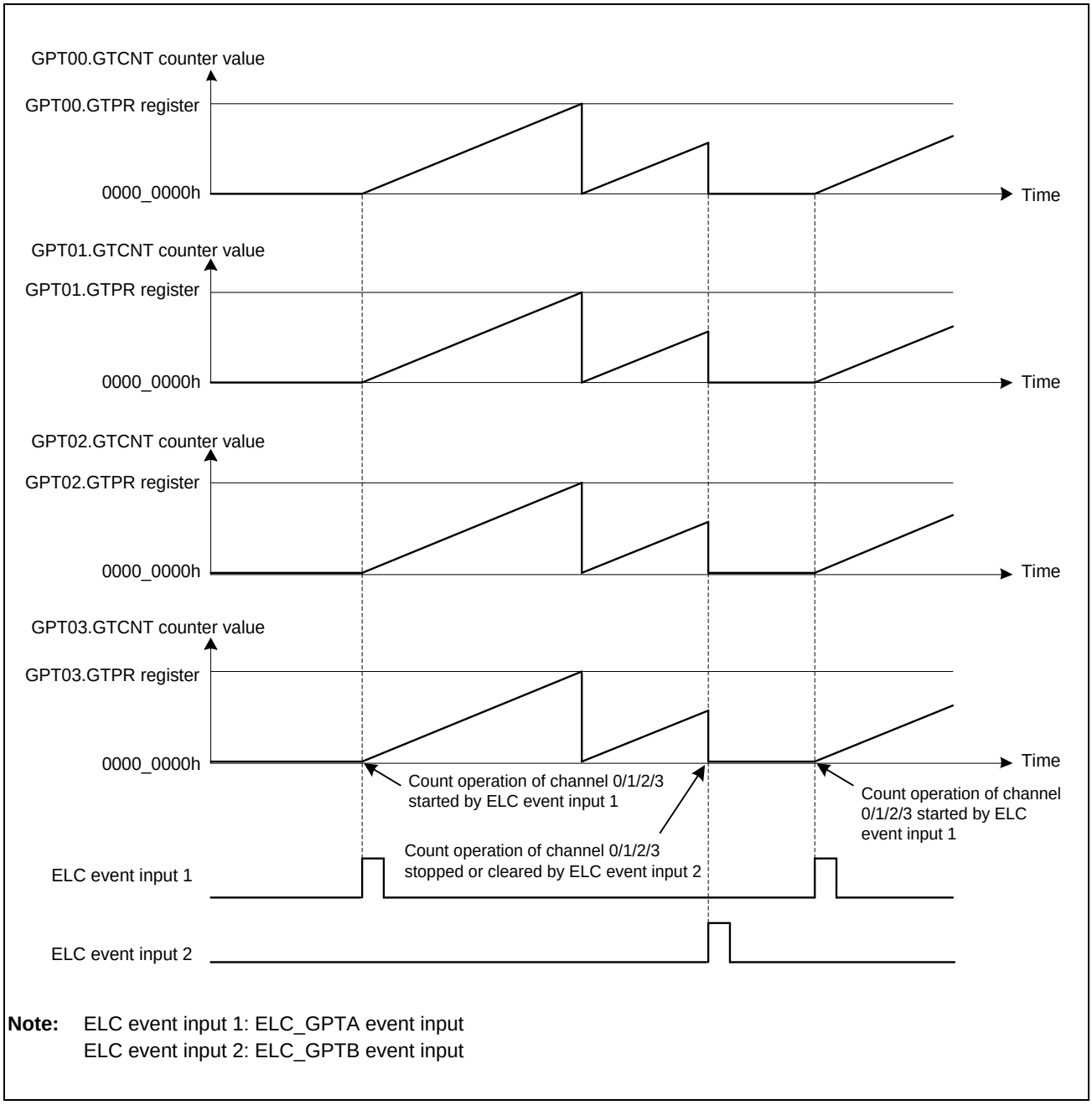


Figure 5.7-40 Example of Phase Start by Software with the Same Counting Cycle (GTPR Register Value)

5.7.3.8.2 Synchronized Operation by Hardware

The counters for multiple channels can be started, stopped, and cleared simultaneously by the following hardware sources. Hardware sources that can cause a synchronized operation are external trigger input and ELC event input. Synchronized operation through the GTIOCnA and GTIOCnB pin inputs is possible by setting an ELC event due to input capture as a hardware source (n = 0 to 15).

Figure 5.7-41 shows an example of simultaneous starting, stopping, and clearing by a hardware source. **Table 5.7-35** shows the setting example.



Note: ELC event input 1: ELC_GPTA event input
ELC event input 2: ELC_GPTB event input

Figure 5.7-41 Example of Simultaneous Starting, Stopping, and Clearing by Hardware Source with the Same Counting Cycle (GTPR Register Value)

Table 5.7-35 Example Setting for Simultaneous Start by Hardware Source

No.	Step Name	Description
1	Set operating mode	Set the operating mode with the GTCR.MD[2:0] bits. In **Figure 5.7-41**, 000b (saw-wave PWM mode) is set.
2	Set count direction	Select the count direction (up or down) with the GTUDDTYC register. In **Figure 5.7-41**, after 11b is set in the GTUDDTYC[1:0] bits, 01b is set in the GTUDDTYC[1:0] bits (up-counting).
3	Select count clock	Select the count clock with the GTCR.TPCS[3:0] bits.
4	Set cycle	Set the cycle in the GTPR register.
5	Set initial value for counter	Set the initial value in the GTCNT counter. In **Figure 5.7-41**, 0000_0000h is set.
6	Set hardware count start	Select a hardware source for starting count operation with the GTSSR register, and wait for count start by the hardware source. In **Figure 5.7-41**, GTSSR.SSELCA = 1b.
7	Set hardware count stop	Select a hardware source for stopping count operation with the GTPSR register, and wait for count stop by the hardware source. In **Figure 5.7-41**, GTPSR.PSELCB = 1b.
8	Set hardware count clear	Select a hardware source for clearing count operation with the GTCSR register, and wait for count clear by the hardware source. In **Figure 5.7-41**, GTCSR.CSELCB = 1b.
9	Set hardware source operation	Set operation of the hardware source selected in the GTSSR, GTPSR, or GTCSR registers, and start, stop, or clear counting. In **Figure 5.7-41**, ELC_GPTA input and ELC_GPTB input are set.

5.7.3.9 PWM Output Operation Examples

5.7.3.9.1 Synchronized PWM Output

The GPT outputs 16 × 2 phases of linked PWM waveforms for a maximum of GPT × 16 channels.

Figure 5.7-42 shows an example in which four channels perform synchronized operation in saw-wave PWM mode and eight phases of PWM waveforms are output. The GTIOCnA is set so that it outputs the low level as the initial value, high level on a compare match with GTCCRA, and low level at the cycle end. The GTIOCnB is set so that it outputs the low level as the initial value, high level on a compare match with GTCCRB, and low level at the cycle end.

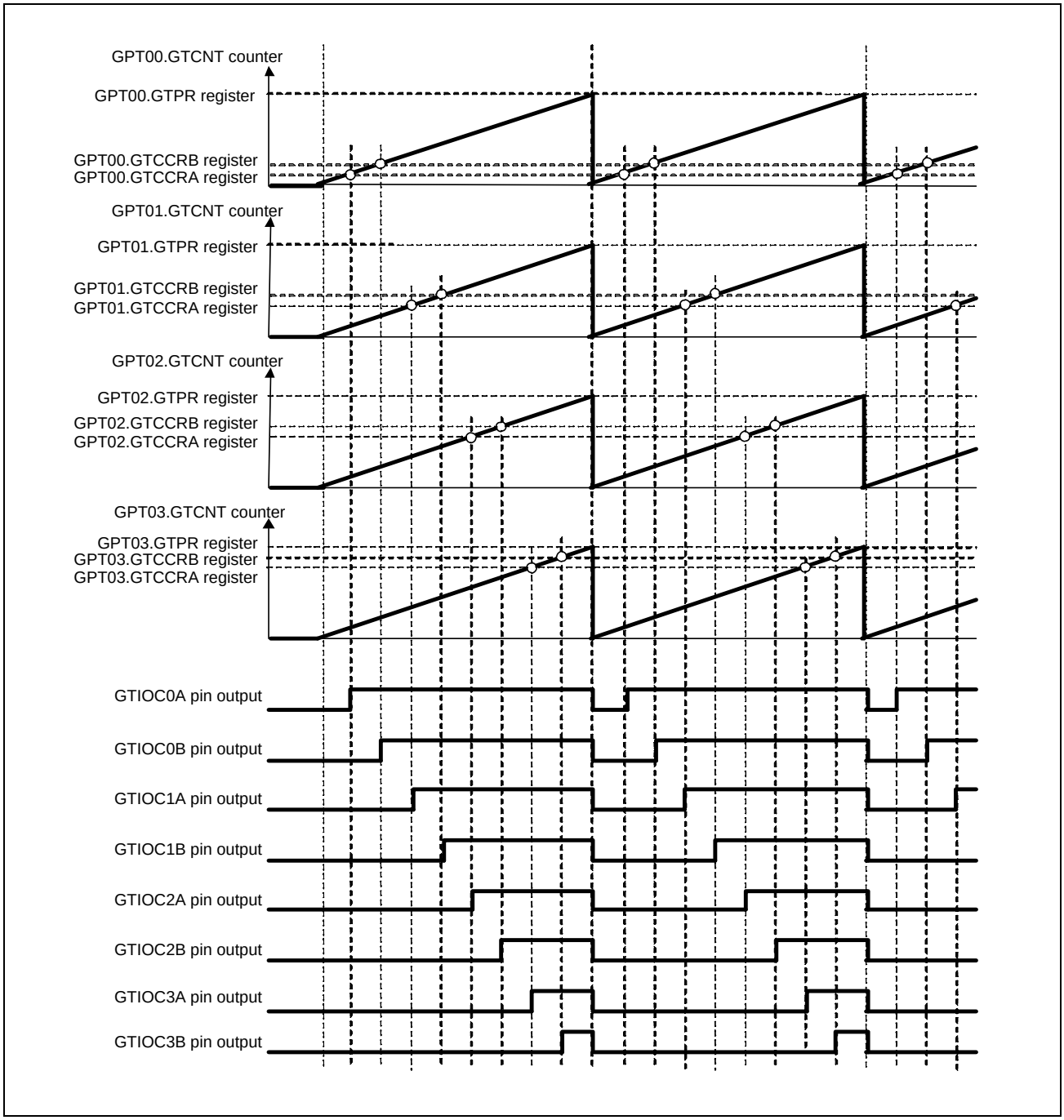


Figure 5.7-42 Example of Synchronized PWM Output

5.7.3.9.2 3-Phase Saw-Wave Complementary PWM Output

Figure 5.7-43 shows an example in which three channels perform synchronized operation in saw-wave PWM mode and 3- phase complementary PWM waveforms are output. The GTIOCnA pin is set so that it outputs the low level as the initial value, high level on a compare match with GTCCRA, and low level at the cycle end. The GTIOCnB pin is set so that it outputs the high level as the initial value, low level on a compare match with GTCCRB, and high level at the cycle end.

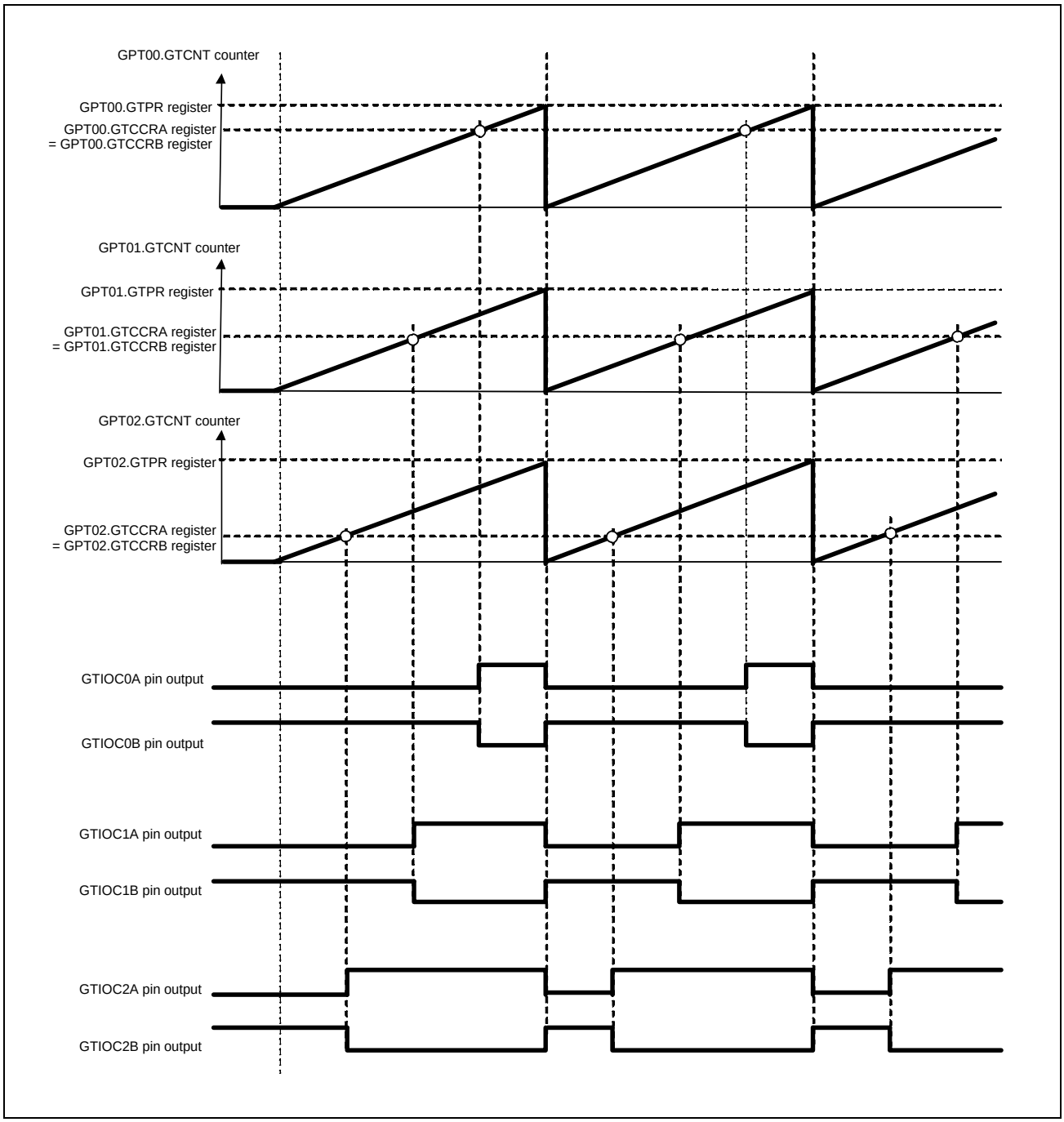


Figure 5.7-43 Example of 3-Phase Saw-Wave Complementary PWM Output

5.7.3.9.3 3-Phase Saw-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 5.7-44 shows an example in which three channels perform synchronized operation in saw-wave one-shot pulse mode with automatic dead time setting and 3-phase complementary PWM waveforms are output. The GTIOCnA pin is set so that it outputs the low level as the initial value, toggles the output on a compare match with GTCCRA, and retains the output at the cycle end. The GTIOCnB pin is set so that it outputs the high level as the initial value, toggles the output on a compare match with GTIOCnB, and retains the output at the cycle end.

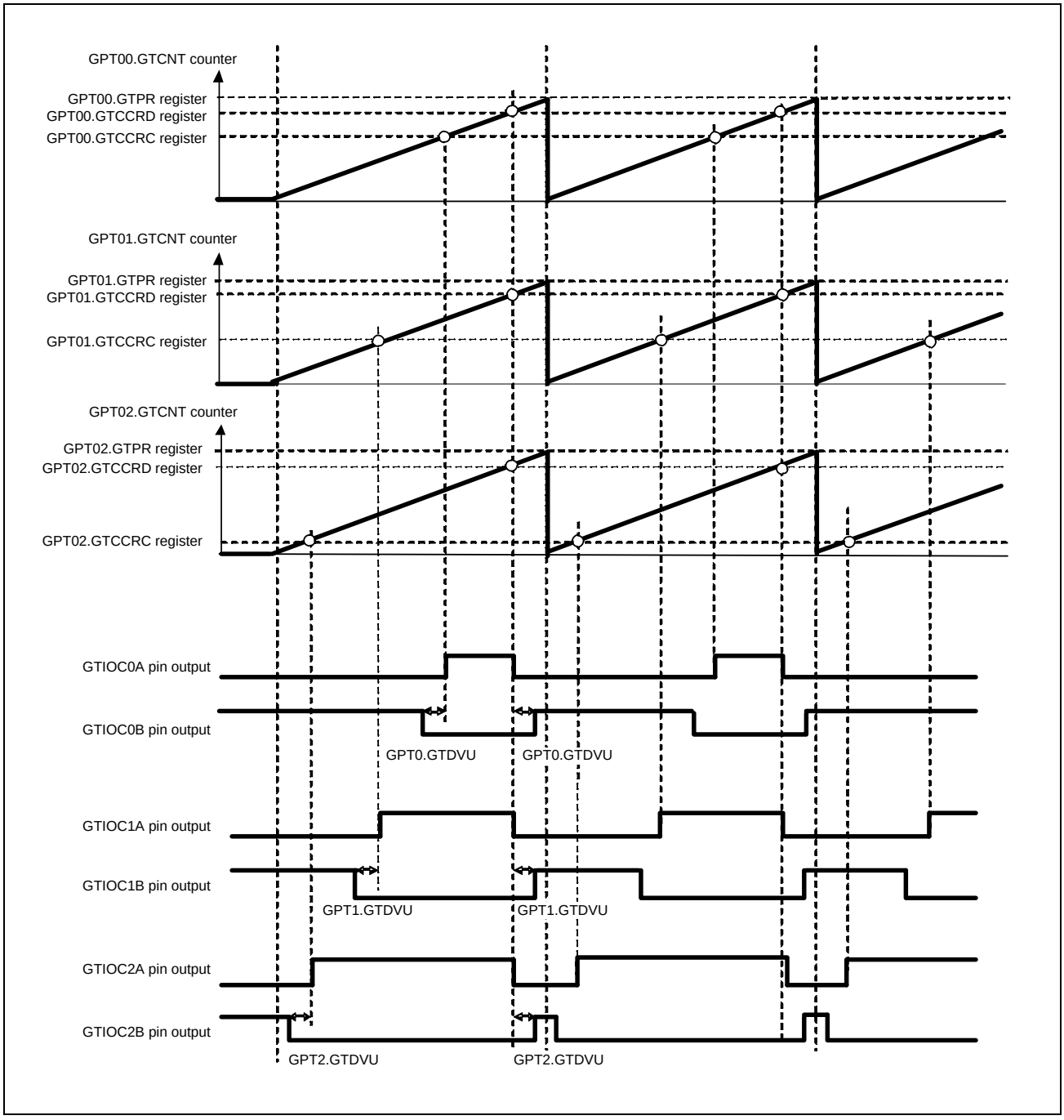


Figure 5.7-44 Example of 3-Phase Saw-Wave Complementary PWM Output with Automatic Dead Time Setting

5.7.3.9.4 3-Phase Triangle-Wave Complementary PWM Output

Figure 5.7-45 shows an example in which three channels perform synchronized operation in triangle-wave PWM mode 1 and 3-phase complementary PWM waveforms are output. The GTIOCnA pin is set so that it outputs the low level as the initial value, toggles the output on a compare match with GTCCRA, and retains the output at the cycle end. The GTIOCnB pin is set so that it outputs the high level as the initial value, toggles the output on a compare match with GTCCRB, and retains the output at the cycle end.

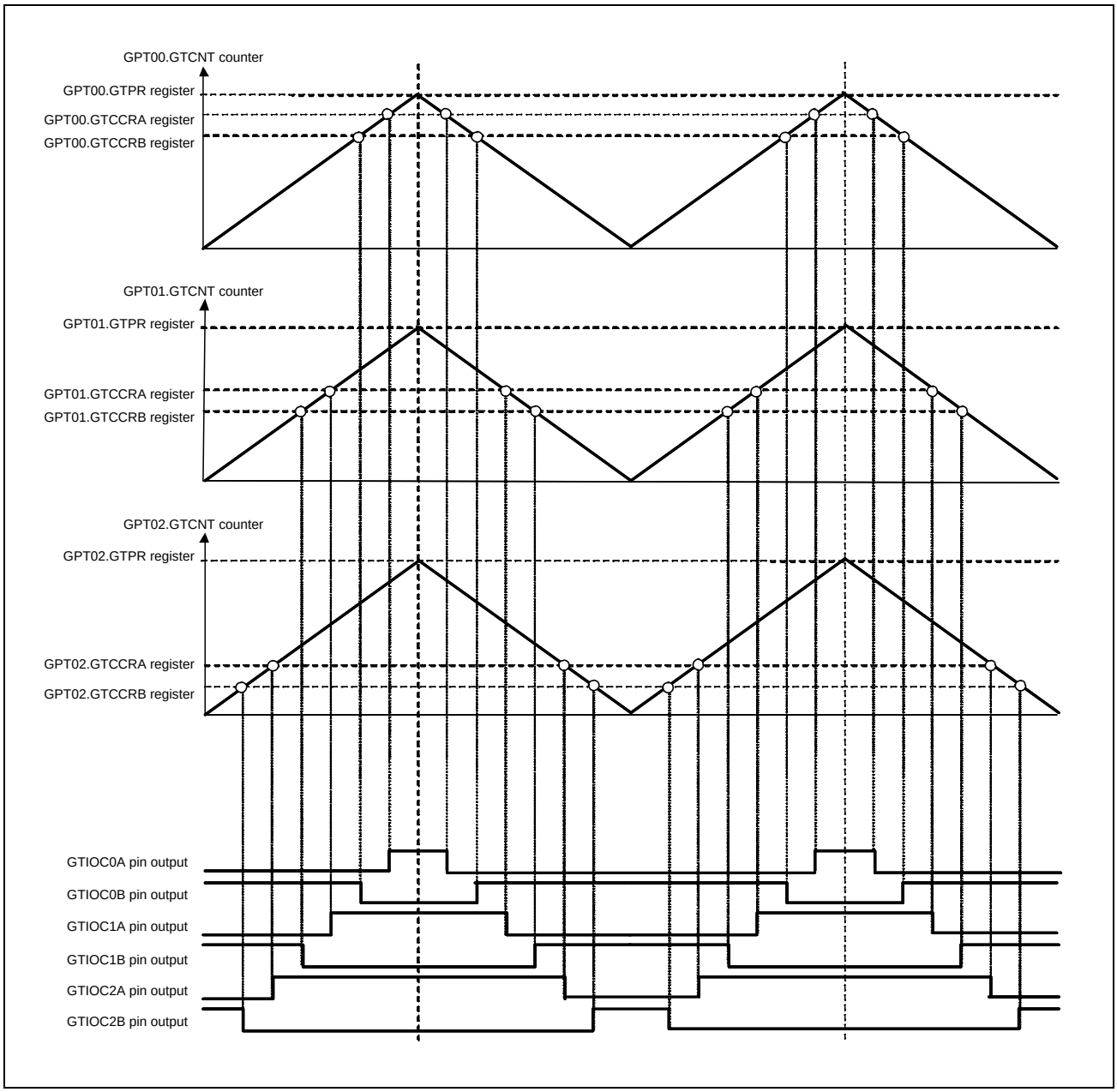


Figure 5.7-45 Example of 3-Phase Triangle-Wave Complementary PWM Output

5.7.3.9.5 3-Phase Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 5.7-46 shows an example in which three channels perform synchronized operation in triangle-wave PWM mode 1 with automatic dead time setting and 3-phase complementary PWM waveforms are output. The GTIOCnA pin is set so that it outputs the low level as the initial value, toggles the output on a compare match with GTCCRA, and retains the output at the cycle end.

The GTIOCnB pin is set so that it outputs the high level as the initial value, toggles the output on a compare match with GTCCRB, and retains the output at the cycle end.

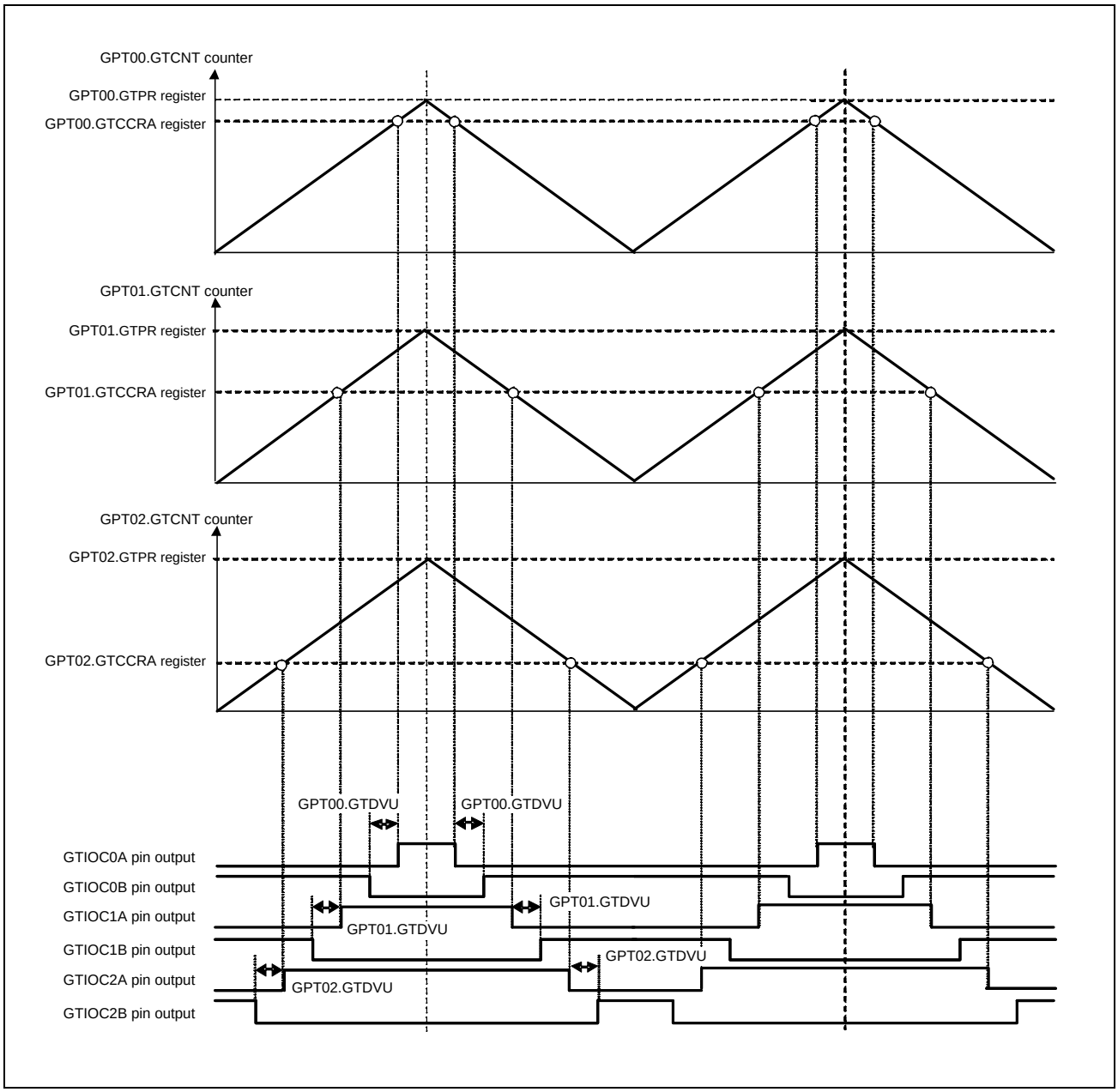


Figure 5.7-46 Example of 3-Phase Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

5.7.3.9.6 3-Phase Asymmetric Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

Figure 5.7-47 shows an example in which three channels perform synchronized operation in triangle-wave PWM mode 3 with automatic dead time setting and 3-phase complementary PWM waveforms are output. The GTIOCnA is set so that it outputs the low level as the initial value, toggles the output on a compare match with GTCCRA, and retains the output at the cycle end.

The GTIOCnB is set so that it outputs the high level as the initial value, toggles the output on a compare match with GTCCRB, and retains the output at the cycle end.

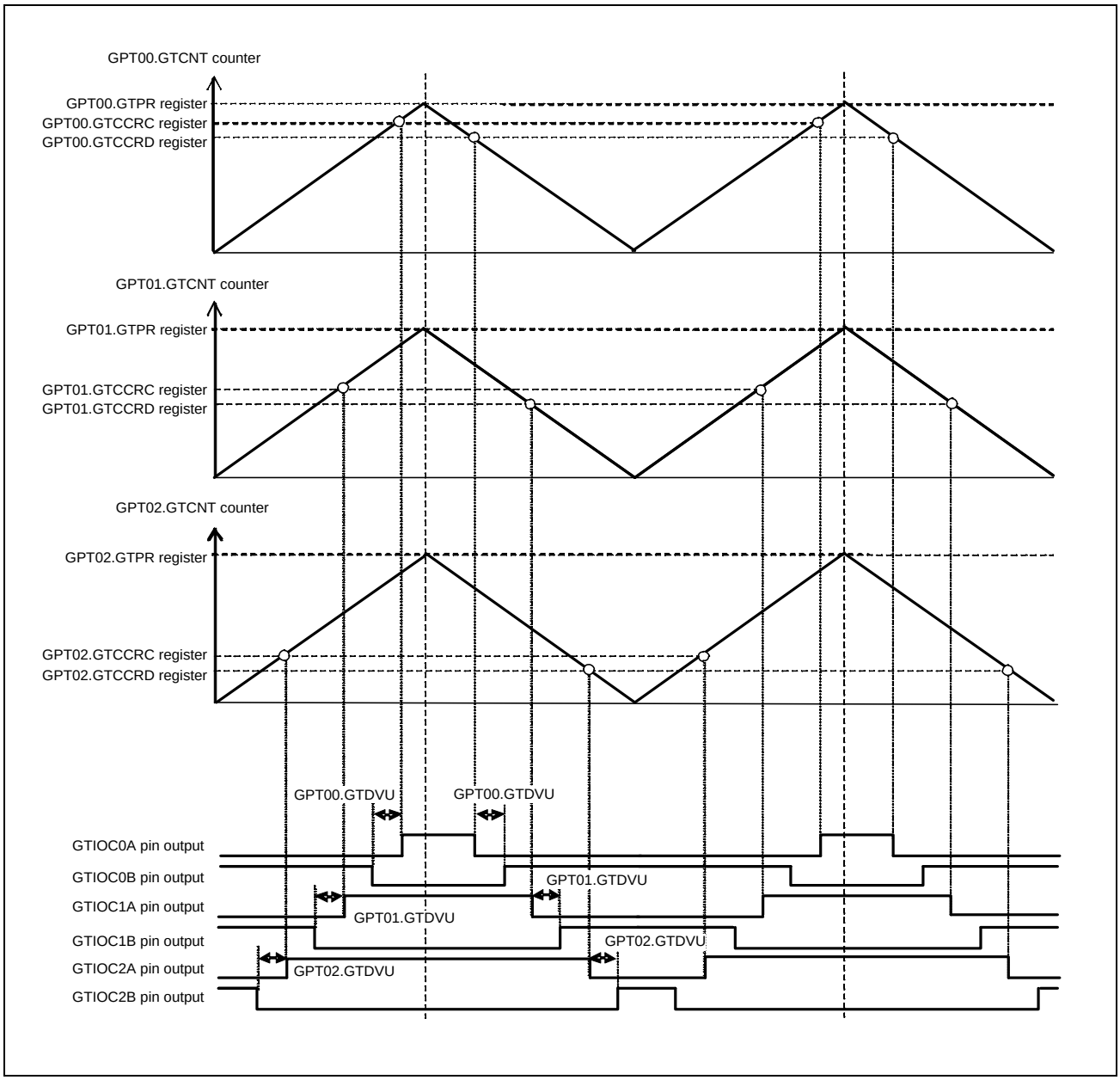


Figure 5.7-47 Example of 3-Phase Asymmetric Triangle-Wave Complementary PWM Output with Automatic Dead Time Setting

5.7.3.10 Phase Counting Function

A phase difference between the GTIOCnA and GTIOCnB pin (n = 0 to 15) inputs is detected and the corresponding GTCNT counts up or counts down. A phase difference is detectable in any combination of the relationship between the edge and the level of GTIOCnA and GTIOCnB pin inputs that are set in the GTUPSR and GTDNSR registers. For details on count operation, see **5.7.3.1.1 Counter Operation**.

Figure 5.7-48 to **Figure 5.7-57** show an example of operation of phase counting modes 1 to 5 when the GTIOCnA, GTIOCnB pins are used. **Table 5.7-36** to **Table 5.7-45** show conditions for up-counting or down-counting and list the settings of the GTUPSR and GTDNSR registers which are corresponding to **Figure 5.7-48** to **Figure 5.7-57**.

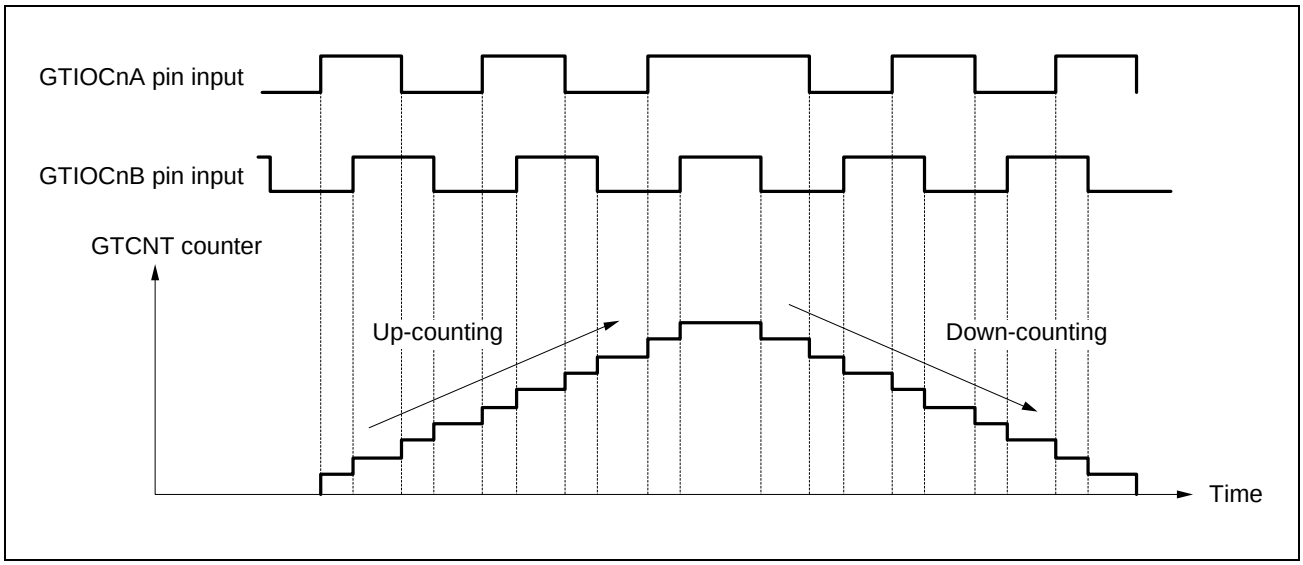


Figure 5.7-48 Example of Phase Counting Mode 1

Table 5.7-36 Conditions of Up-Counting/Down-Counting in Phase Counting Mode 1

↑ : Rising edge
↓ : Falling edge

GTIOCnA Pin Input	GTIOCnB Pin Input	Operation	Register Setting
High	↑	Up-counting	GTUPSR = 0000_6900h GTDNSR = 0000_9600h
Low	↓		
↑	Low		
↓	High		
High	↓	Down-counting	
Low	↑		
↑	High		
↓	Low		

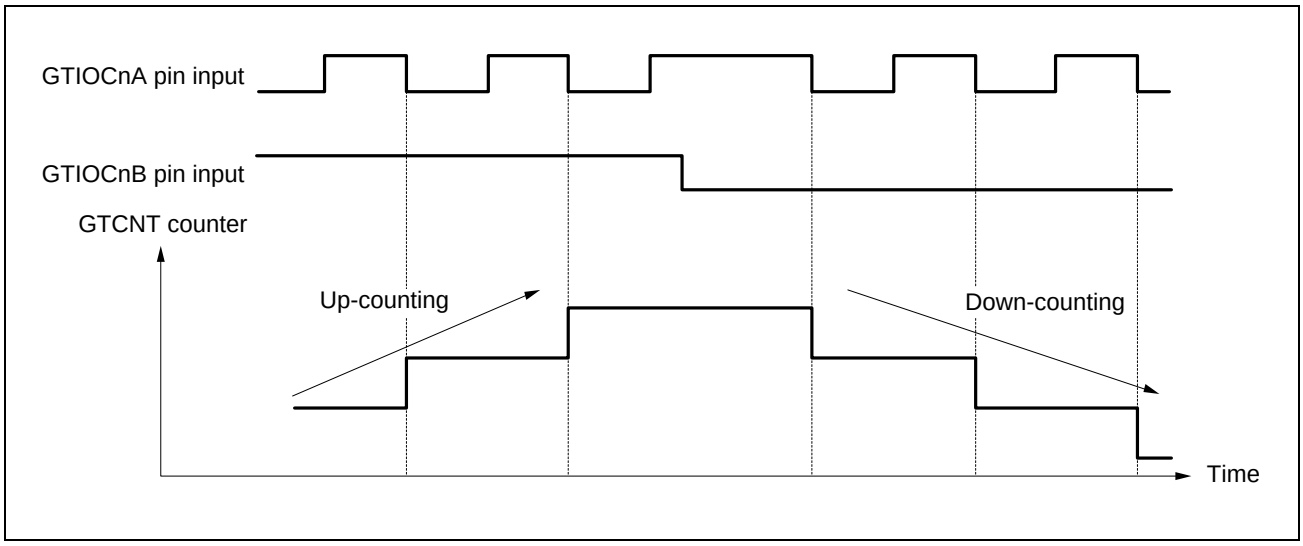


Figure 5.7-49 Example of Phase Counting Mode 2 (A)

Table 5.7-37 Conditions of Up-Counting/Down-Counting in Phase Counting Mode 2 (A)

: Rising edge
: Falling edge

GTIOCnA Pin Input	GTIOCnB Pin Input	Operation	Register Setting
High	↑	No counting	GTUPSR = 0000_0800h GTDNSR = 0000_0400h
Low	↓		
↑	Low		
↓	High	Up-counting	
High	↓	No counting	
Low	↑		
↑	High		
↓	Low	Down-counting	

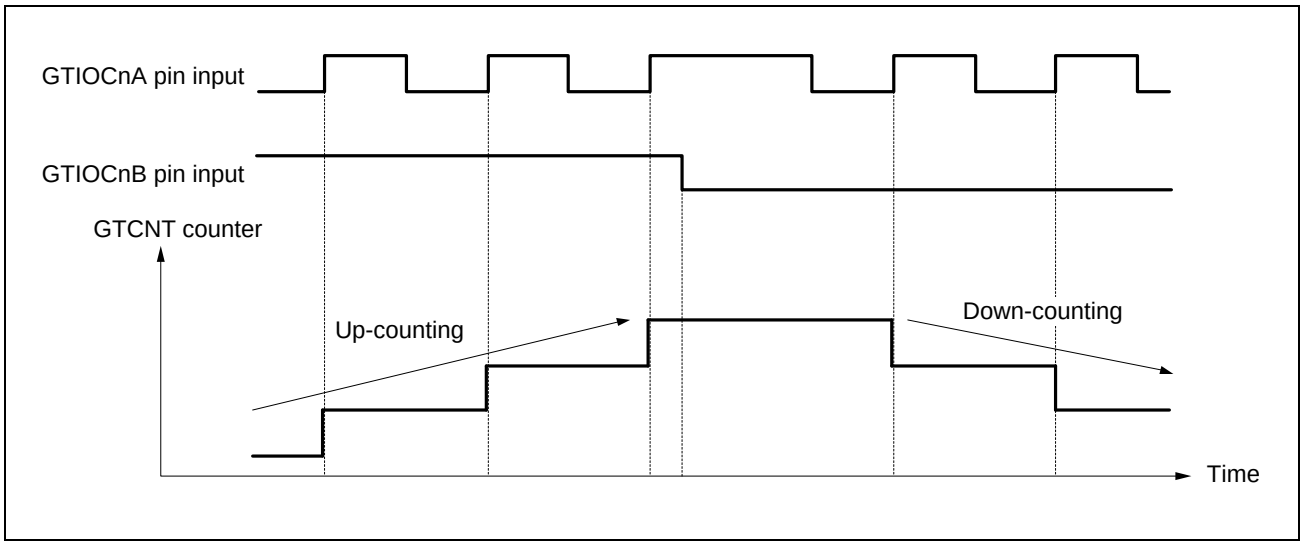


Figure 5.7-50 Example of Phase Counting Mode 2 (B)

Table 5.7-38 Conditions of Up-Counting/Down-Counting in Phase Counting Mode 2 (B)

: Rising edge
: Falling edge

GTIOCnA Pin Input	GTIOCnB Pin Input	Operation	Register Setting
High	↑	No counting	GTUPSR = 0000_0200h GTDNSR = 0000_0100h
Low	↓		
↑	Low	Down-counting	
↓	High	No counting	
High	↓		
Low	↑		
↑	High	Up-counting	
↓	Low	No counting	

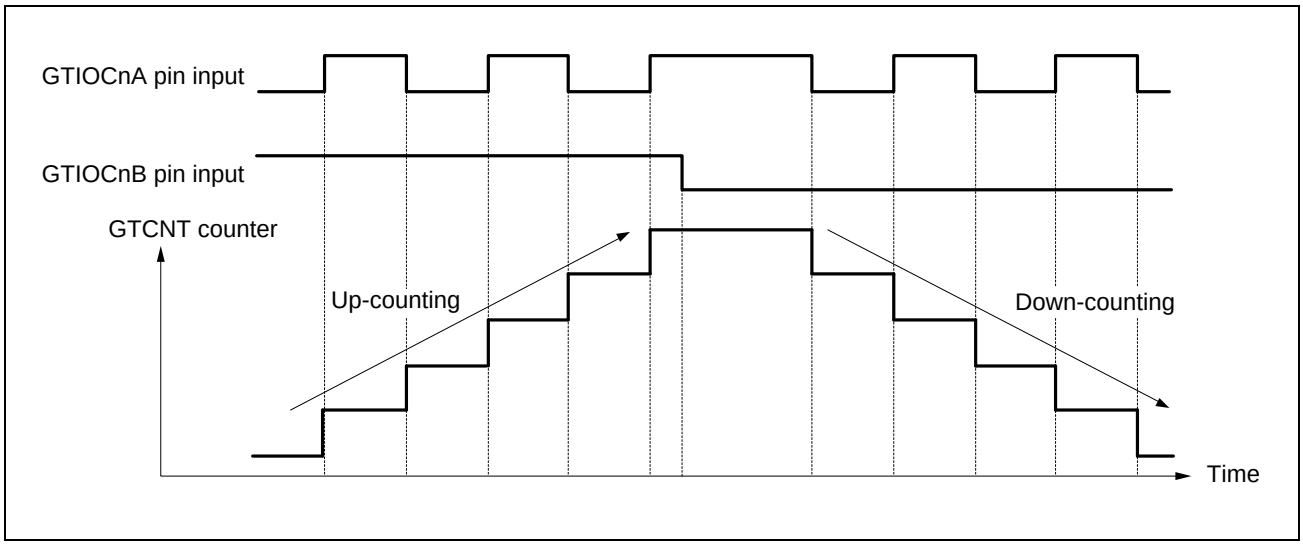


Figure 5.7-51 Example of Phase Counting Mode 2 (C)

Table 5.7-39 Conditions of Up-Counting/Down-Counting in Phase Counting Mode 2 (C)

: Rising edge
: Falling edge

GTIOCnA Pin Input	GTIOCnB Pin Input	Operation	Register Setting
High	↑	No counting	GTUPSR = 0000_0A00h GTDNSR = 0000_0500h
Low	↓		
↑	Low	Down-counting	
↓	High	Up-counting	
High	↓	No counting	
Low	↑		
↑	High	Up-counting	
↓	Low	Down-counting	

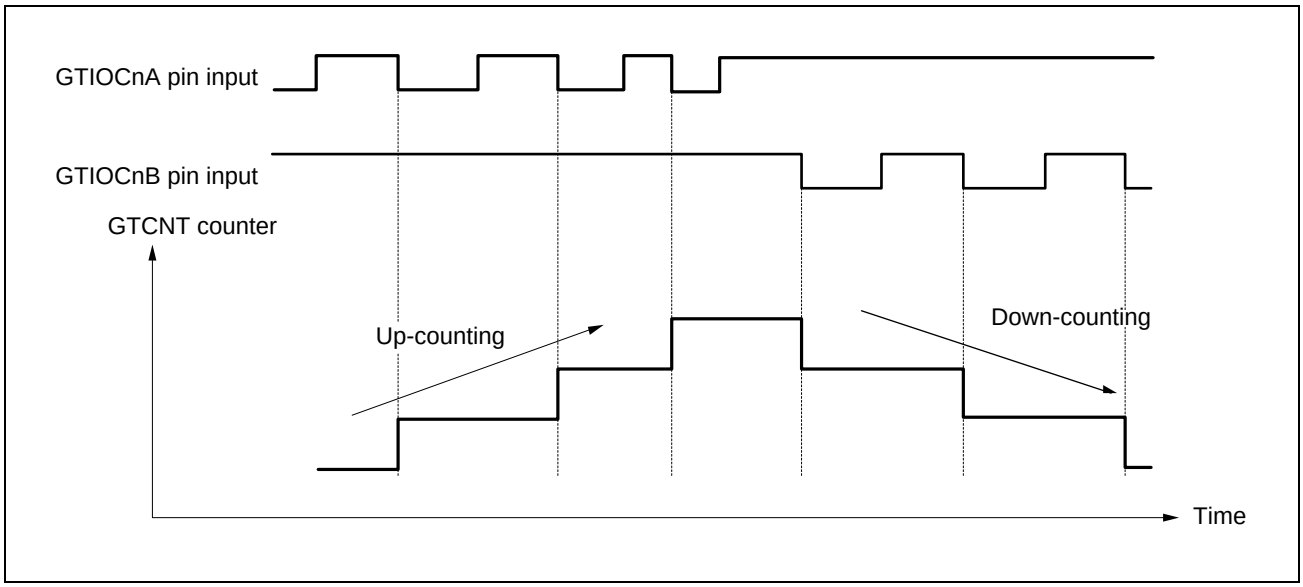


Figure 5.7-52 Example of Phase Counting Mode 3 (A)

Table 5.7-40 Conditions of Up-Counting/Down-Counting in Phase Counting Mode 3 (A)

: Rising edge
: Falling edge

GTIOCnA Pin Input	GTIOCnB Pin Input	Operation	Register Setting
High	↑	No counting	GTUPSR = 0000_0800h GTDNSR = 0000_8000h
Low	↓		
↑	Low		
↓	High	Up-counting	
High	↓	Down-counting	
Low	↑	No counting	
↑	High		
↓	Low		

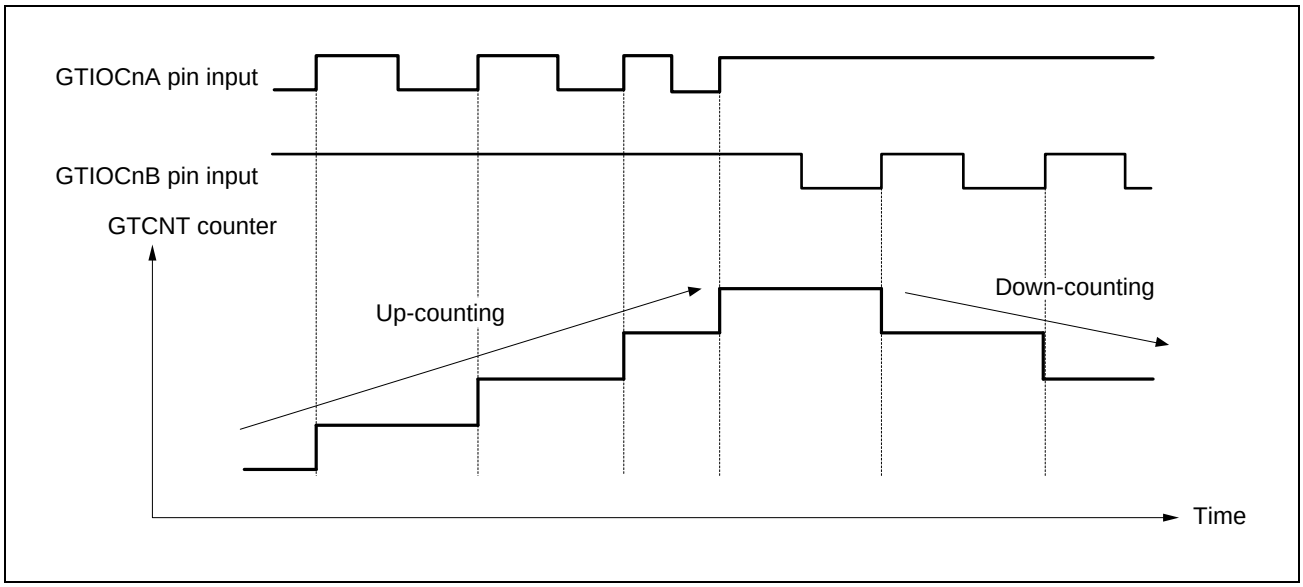


Figure 5.7-53 Example of Phase Counting Mode 3 (B)

Table 5.7-41 Conditions of up-Counting/Down-Counting in Phase Counting Mode 3 (B)

: Rising edge
: Falling edge

GTIOCnA Pin Input	GTIOCnB Pin Input	Operation	Register Setting
High		Down-counting	GTUPSR = 0000_0200h GTDNSR = 0000_2000h
Low		No counting	
	Low		
	High		
High			
Low			
	High	Up-counting	
	Low	No counting	

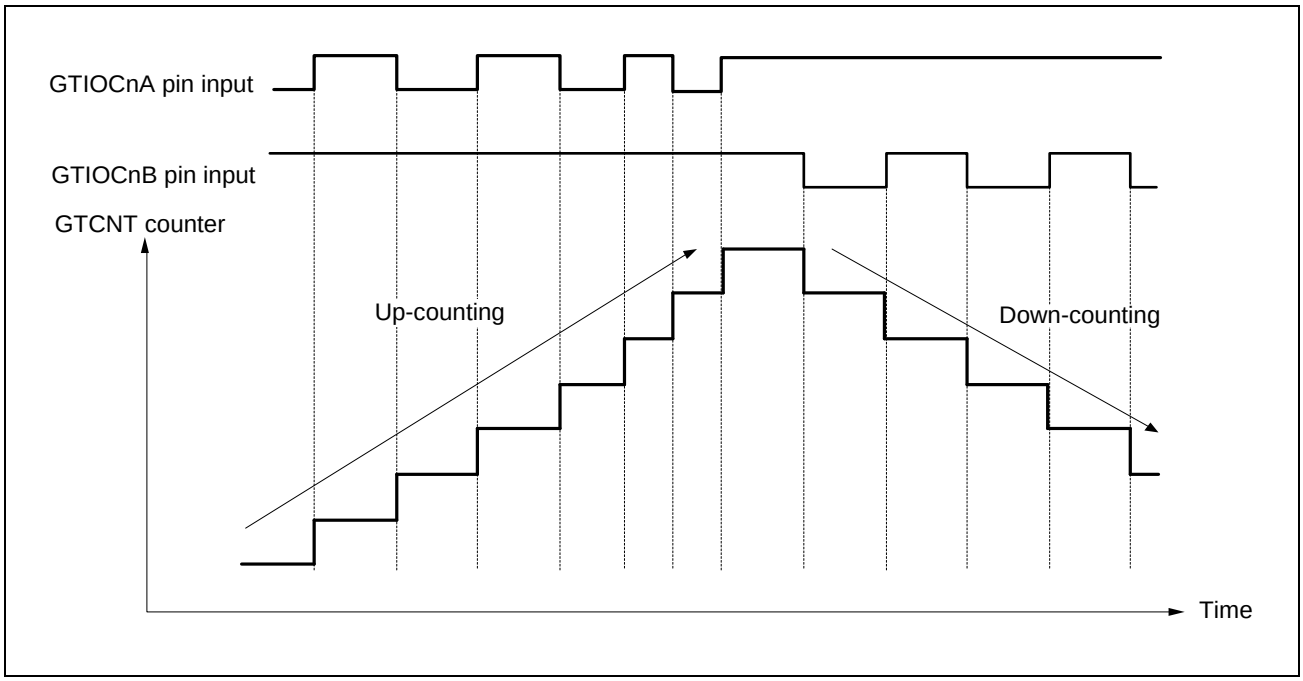


Figure 5.7-54 Example of Phase Counting Mode 3 (C)

Table 5.7-42 Conditions of Up-Counting/Down-Counting in Phase Counting Mode 3 (C)

: Rising edge
: Falling edge

GTIOCnA Pin Input	GTIOCnB Pin Input	Operation	Register Setting
High	↑	Down-counting	GTUPSR = 0000_0A00h GTDNSR = 0000_A000h
Low	↓	No counting	
↑	Low		
↓	High	Up-counting	
High	↓	Down-counting	
Low	↑	No counting	
↑	High	Up-counting	
↓	Low	No counting	

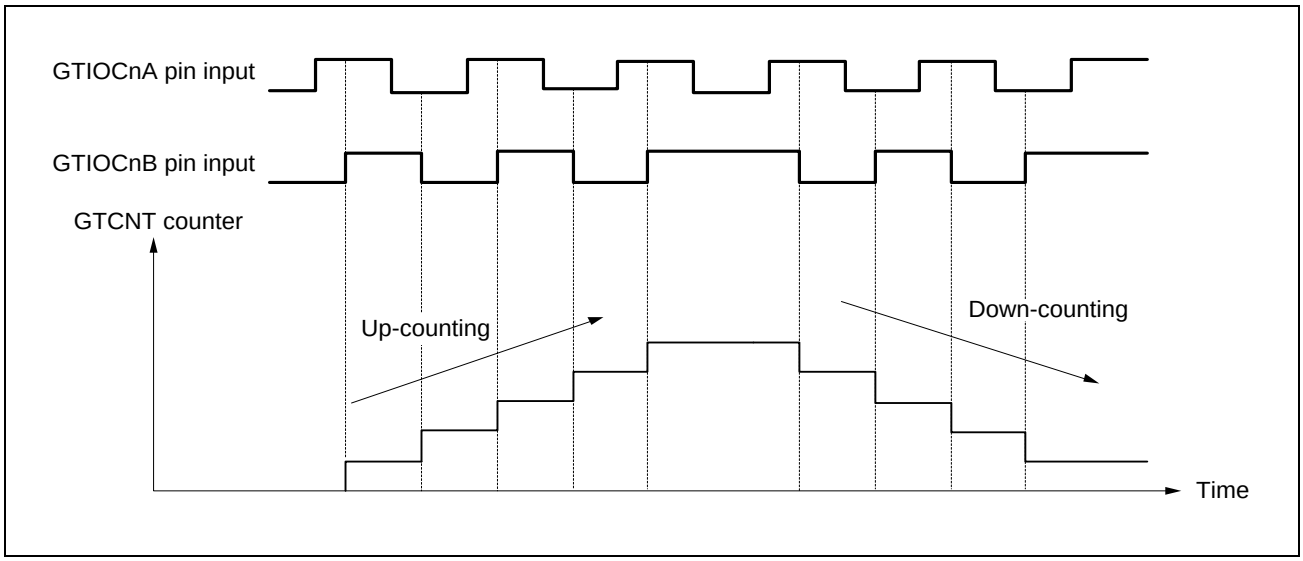


Figure 5.7-55 Example of Phase Counting Mode 4

Table 5.7-43 Conditions of Up-Counting/Down-Counting in Phase Counting Mode 4

: Rising edge
: Falling edge

GTIOCnA pin Input	GTIOCnB pin Input	Operation	Register Setting
High		Up-counting	GTUPSR = 0000_6000h GTDNSR = 0000_9000h
Low			
	Low	No counting	
	High		
High		Down-counting	
Low			
	High	No counting	
	Low		

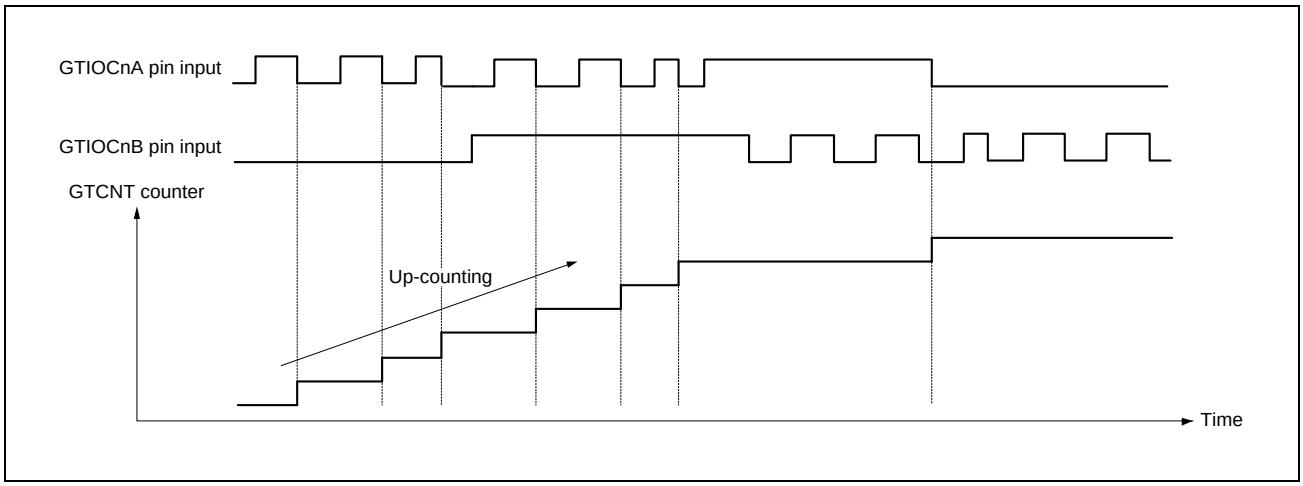


Figure 5.7-56 Example of Phase Counting Mode 5 (A)

Table 5.7-44 Conditions of Up-Counting/Down-Counting in Phase Counting Mode 5 (A)

: Rising edge
: Falling edge

GTIOCnA Pin Input	GTIOCnB Pin Input	Operation	Register Setting
High	↑	No counting	GTUPSR = 0000_0C00h GTDNSR = 0000_0000h
Low	↓		
↑	Low		
↓	High	Up-counting	
High	↓	No counting	
Low	↑		
↑	High		
↓	Low	Up-counting	

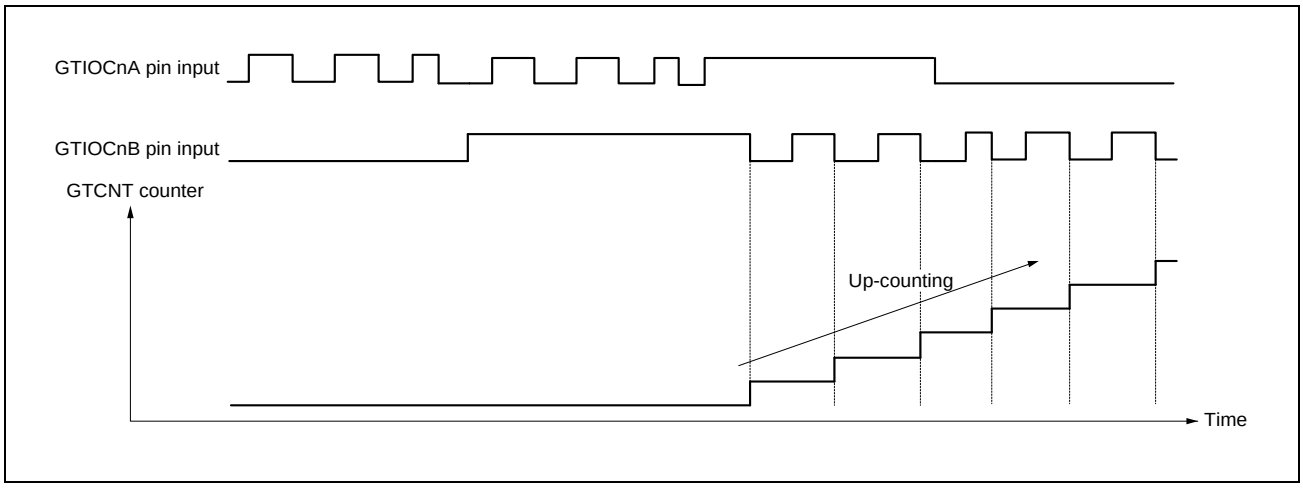


Figure 5.7-57 Example of Phase Counting Mode 5 (B)

Table 5.7-45 Conditions of Up-Counting/Down-Counting in Phase Counting Mode 5 (B)

: Rising edge
: Falling edge

GTIOCnA Pin Input	GTIOCnB Pin Input	Operation	Register Setting
High	↑	No counting	GTUPSR = 0000_C000h GTDNSR = 0000_0000h
Low	↓	Up-counting	
↑	Low	No counting	
↓	High		
High	↓	Up-counting	
Low	↑	No counting	
↑	High		
↓	Low		

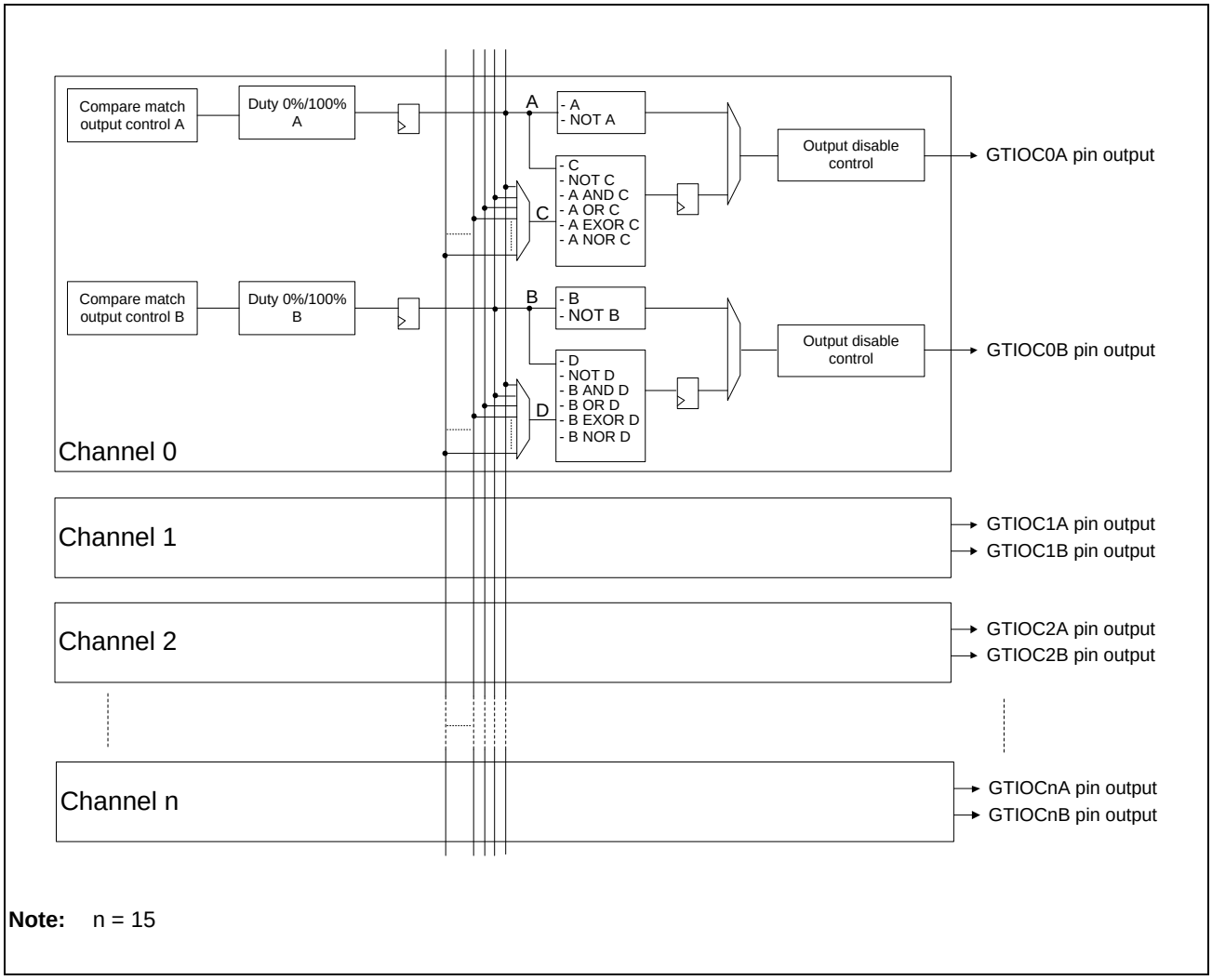
5.7.3.11 Inter Channel Logical Operation Function

The logical operation function between compare match outputs can be performed. **Figure 5.7-58** shows the block diagram of inter channel logical operation.

To prevent hazard to the GPT output, the signal after logical operation is latched with PCLKD. After latching, output disable control is performed.

When the logical operation function which causes the delay of 1 PCLKD is selected, the output enable signal is also delayed with 1 PCLKD and input to the output disable control.

When the same signal (C = A or D = B) to operate logical function AND, OR, EXOR and NOR is selected, C or D is treated as 1. In the case of GTIOCnA pin output, when A of same channel is selected for C, the result of AND is A, the result of OR is 1, the result of EXOR is NOT A, and the result of NOR is 0.



Note: n = 15

Figure 5.7-58 Block Diagram of Inter Channel Logical Operation

Figure 5.7-59 shows an example of inter channel logical operation.

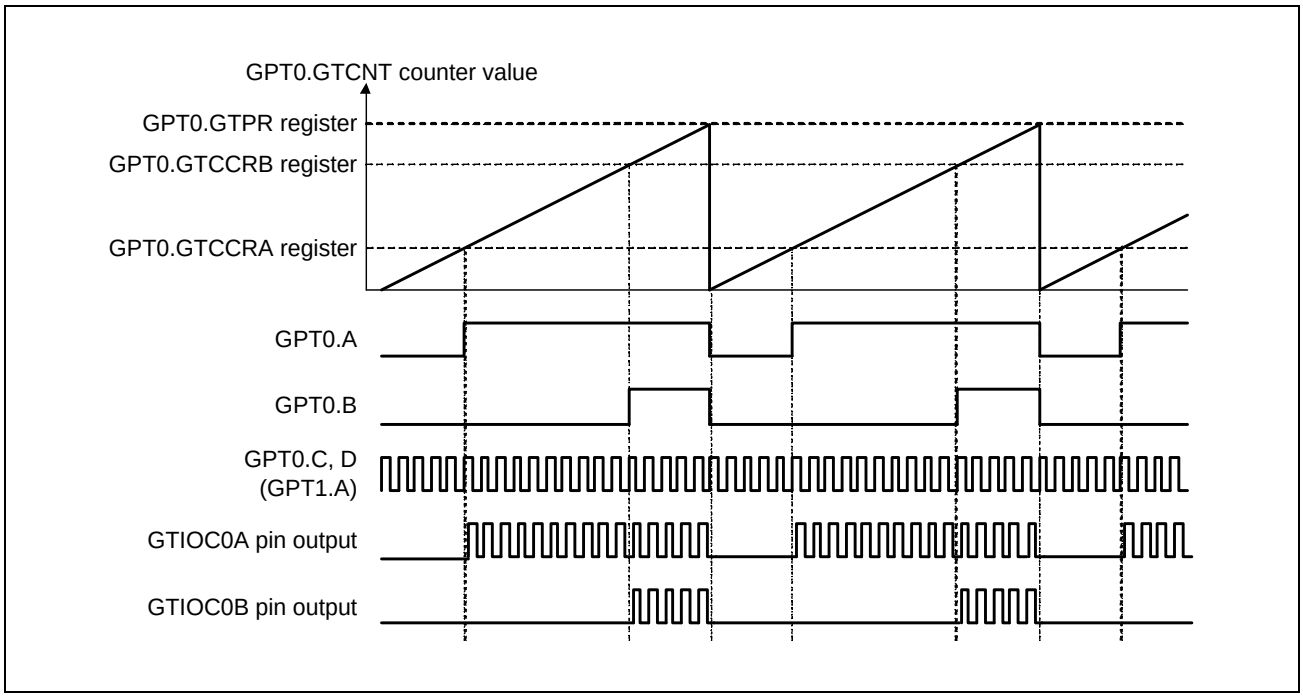


Figure 5.7-59 Example of Inter Channel Logical Operation

5.7.4 Interrupt Sources

5.7.4.1 Interrupt Sources

The GPT provides the following interrupt sources:

- GTCCR input capture/compare match
- GTCNT counter overflow (GTPR compare match)/underflow.
- GTADTR compare match
- Dead time error
- Short-circuited output

Each interrupt source has its own status flag. When an interrupt source signal is generated, the corresponding status flag in GTST is set to 1b. The corresponding status flag in GTST can be cleared by writing 0b to it. If flag setting and flag clearing occur at the same time, flag clearing takes priority over flag setting. These flags are automatically updated according to the internal state. The Interrupt Controller Unit can change the priority between the channels. However, the priority within a channel is fixed. For details, see **4.6.1 Interrupt Control Unit (ICU)**.

Table 5.7-46 lists the GPT interrupt sources.

Table 5.7-46 Interrupt sources

Name	Interrupt Source	Interrupt Flag	DMAC Activation
GPT_Ux_gpt_gtcia_n_m	GTCCRA input capture/compare match	GTST[0] (TCFA)	Possible
GPT_Ux_gpt_gtcib_n_m	GTCCRB input capture/compare match	GTST[1] (TCFB)	Possible
GPT_Ux_gpt_gtcic_n_m	GTCCRC compare match	GTST[2] (TCFC)	Possible
GPT_Ux_gpt_gtcid_n_m	GTCCRD compare match	GTST[3] (TCFD)	Possible
GPT_Ux_gpt_gtcie_n_m	GTCCRE compare match	GTST[4] (TCFE)	Possible
GPT_Ux_gpt_gtcif_n_m	GTCCRF compare match	GTST[5] (TCFF)	Possible
GPT_Ux_gpt_gtciv_n_m	GTCNT overflow (GPTn.GTPR compare match)	GTST[6] (TCFPO)	Possible
GPT_Ux_gpt_gtciu_n_m	GTCNT underflow	GTST[7] (TCFPU)	Possible
GPT_Ux_gpt_gtdei_n_m	Dead time error	GTST[28] (DTEF)	Possible
GPT_Ux_gpt_gtcih_n_m	A and B both high interrupt	GTST[29] (OABHF)	Possible
GPT_Ux_gpt_gtcil_n_m	A and B both low interrupt	GTST[30] (OABLF)	Possible
GPT_Ux_gpt_gtciada_n_m	Compare match with the GTADTRA register	GTST[16] (ADTRAUF) GTST[17] (ADTRADF)	Possible
GPT_Ux_gpt_gtciadb_n_m	Compare match with the GTADTRB register	GTST[18] (ADTRBUF) GTST[19] (ADTRBDF)	Possible

Note: x: 0, 1
m: 0 to 7

5.7.4.1.1 GPT_Ux_gpt_gtcia_n_m

An interrupt request is generated under the following conditions:

- When the GTCCRA register functions as a compare match register, the GTCNT counter value matches with the GTCCRA register
- When the GTCCRA register functions as an input capture register, the input-capture signal causes transfer of the GTCNT counter value to the GTCCRA register.

5.7.4.1.2 GPT_Ux_gpt_gtcib_n_m

An interrupt request is generated under the following conditions:

- When the GTCCRB register functions as a compare match register, the GTCNT counter value matches with the GTCCRB register
- When the GTCCRB register functions as an input capture register, the input-capture signal causes transfer of the GTCNT counter value to the GTCCRB register.

5.7.4.1.3 GPT_Ux_gpt_gtcic_n_m

An interrupt request is generated under the following condition:

- When the GTCCRC register functions as a compare match register, the GTCNT counter value matches with the GTCCRC register.

A compare match is not performed and therefore, an interrupt is not requested in the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 01b, 10b, 11b (the GTCCRC register performs buffer operation).

5.7.4.1.4 GPT_Ux_gpt_gtcid_n_m

An interrupt request is generated under the following condition:

- When the GTCCRD register functions as a compare match register, the GTCNT counter value matches with the GTCCRD register.

A compare match is not performed and therefore, an interrupt is not requested in the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRA[1:0] = 10b, 11b (the GTCCRD register performs buffer operation).

5.7.4.1.5 GPT_Ux_gpt_gtcie_n_m

An interrupt request is generated under the following condition:

- When the GTCCRE register functions as a compare match register, the GTCNT counter value matches with the GTCCRE register.

A compare match is not performed and therefore, an interrupt is not requested in the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 01b, 10b, 11b (the GTCCRE register performs buffer operation).

5.7.4.1.6 GPT_Ux_gpt_gtcif_n_m

An interrupt request is generated under the following condition:

- When the GTCCRF register functions as a compare match register, the GTCNT counter value matches with the GTCCRF register.

A compare match is not performed and therefore, an interrupt is not requested in the following conditions:

- GTCR.MD[2:0] = 001b (saw-wave one-shot pulse mode)
- GTCR.MD[2:0] = 110b (triangle-wave PWM mode 3)
- GTBER.CCRB[1:0] = 10b, 11b (the GTCCRF register performs buffer operation).

5.7.4.1.7 GPT_Ux_gpt_gtciv_n_m

An interrupt request is generated in the following conditions:

- In saw-wave mode, interrupt requests are enabled on overflow (when the GTCNT counter value changes from GTPR to 0b during up-counting)
- In triangle-wave mode, interrupt requests are enabled at crests (the GTCNT changes from GTPR to GTPR-1)
- In counting by hardware sources, an overflow (GTCNT changes from GTPR to 0b in up-counting) has occurred.

5.7.4.1.8 GPT_Ux_gpt_gtciu_n_m

An interrupt request is generated in the following conditions.

- In saw-wave mode, interrupt requests are enabled on underflow (when the GTCNT counter value changes from 0b to GTPR during down-counting)
- In triangle-wave mode, interrupt requests are enabled at troughs (the GTCNT changes from 0b to 1b)
- In counting by hardware sources, underflow (GTCNT changes from 0b to GTPR in down-counting) has occurred.

About Interrupt signals and interrupt status flags, see **5.7.2.2.16 General-Purpose Timer Status Register (GPTm_n_GTST)**.

5.7.4.1.9 GPT_Ux_gpt_gtdei_n_m

When the automatic dead time setting has been made, the GTST.DTEF flag becomes 1b when the change point of the waveform after the automatic setting exceeds the counting cycle. If the GTINTAD.GRPDTE bit is 1 at this time, a dead time error interrupt (GDTE) request is generated.

Note that the GTST.DTEF flag changes from 1b to 0b when the change point of the waveform after the automatic dead time setting has fallen back within the counting cycle.

5.7.4.1.10 GPT_Ux_gpt_gtcih_n_m

If the values on the GTIOCnA and GTIOCnB pins become 1 simultaneously, the GTST.OABHF flag becomes 1b. If the setting of the GTINTAD.GRPABH bit is 1b at this time, a simultaneous high-level output interrupt (GTCIH) is generated.

Note that the GTST.OABHF flag changes from 1b to 0b if the value on either of the GTIOCnA and GTIOCnB pins becomes 0.

5.7.4.1.11 GPT_Ux_gpt_gtcil_n_m

If the values on the GTIOCnA and GTIOCnB pins become 0 simultaneously, the GTST.OABLF flag becomes 1b. If the setting of the GTINTAD.GRPABL bit is 1b at this time, a simultaneous low-level output interrupt (GTCIL) is generated.

Note that the GTST.OABLF flag changes from 1b to 0b if the value on either of the GTIOCnA and GTIOCnB pins becomes 1.

5.7.4.1.12 GPT_Ux_gpt_gtciada_n_m

When the GTCNT counter value matches the value of the GTADTRA register, an interrupt request is generated if any of the following condition is met:

- The GTINTAD.ADTRAUEN bit is 1b in up-counting.
- The GTINTAD.ADTRADEN bit is 1b in down-counting.

An interrupt request is not generated during event counting.

5.7.4.1.13 GPT_Ux_gpt_gtciadb_n_m

When the GTCNT counter value matches the value of the GTADTRB register, an interrupt request is generated if any of the following condition is met:

- The GTINTAD.ADTRBUEN bit is 1b in up-counting.
- The GTINTAD.ADTRBDEN bit is 1b in down-counting.

An interrupt request is not generated during event counting.

5.7.4.2 DMAC Activation

The DMAC can be activated by the interrupt in each channel. For details, see **4.6.1 Interrupt Control Unit (ICU)** and **4.7 DMA Controller (DMAC)**.

5.7.4.3 Interrupt and A/D Converter Start Request Skipping Function

5.7.4.3.1 Interrupt Skipping Function by GTITC Register

By setting the GTITC register, the GTCNT counter overflow (GTPR register compare match)/underflow interrupt (GTCIV/GTCIU) can be skipped. Other interrupts and A/D converter start request signals can be skipped by interlinking with the GTCIV/GTCIU interrupt skipping function. However, the dead time error interrupts cannot be interlinked with the GTCIV/GTCIU skipping function. Simultaneous high-level output and low-level output interrupts cannot be interlinked with the GTCIV/GTCIU interrupt skipping function. When interrupts are skipped, a change in the corresponding status flag is also skipped and the skipping function continues to operate even while the status flag is 1b.

The interrupt skipping function is only dependent on the setting of the GTITC register and is not dependent on the setting of the GTINTAD register interrupt enable bit.

When both troughs and crests are counted and skipped in triangle-wave mode, if the skipping count is an odd number, GTCIV/GTCIU interrupt requests on troughs only or crests only are not generated depending on the timing with which the skipping counter is started. When both troughs and crests are to be counted and skipped in triangle-wave mode and the GTCIV/GTCIU interrupts on troughs only or crests only are to be used, the skipping count setting should be an even number.

Similarly, when both overflows and underflows are counted and skipped with the counting direction changed in saw-wave mode, GTCIV/GTCIU interrupt requests on overflow only or underflow only may not be generated. When both overflows and underflows are to be counted and skipped with the counting direction changed in saw-wave mode and the GTCIV/GTCIU interrupts on overflow only or underflow only, the skipped state should be checked thoroughly before use.

Before changing the skipping count, be sure to release the skipping function (GTITC.IVTC[1:0] bits = 00b).

Figure 5.7-60 to **Figure 5.7-65** show examples of skipping function operation.

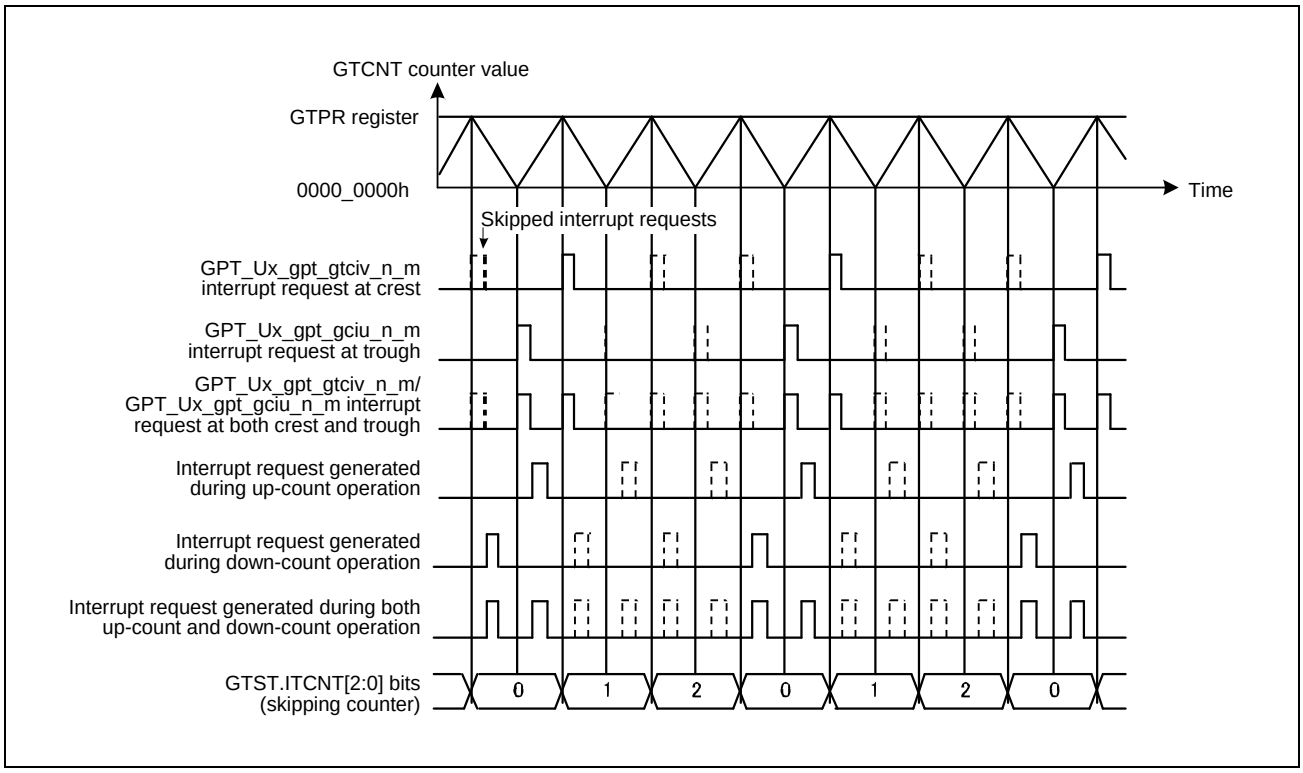


Figure 5.7-60 Example of Interrupt Skipping Function Operation (Triangle Waves, Counting and Skipping Crests, Skipping Count: 2)

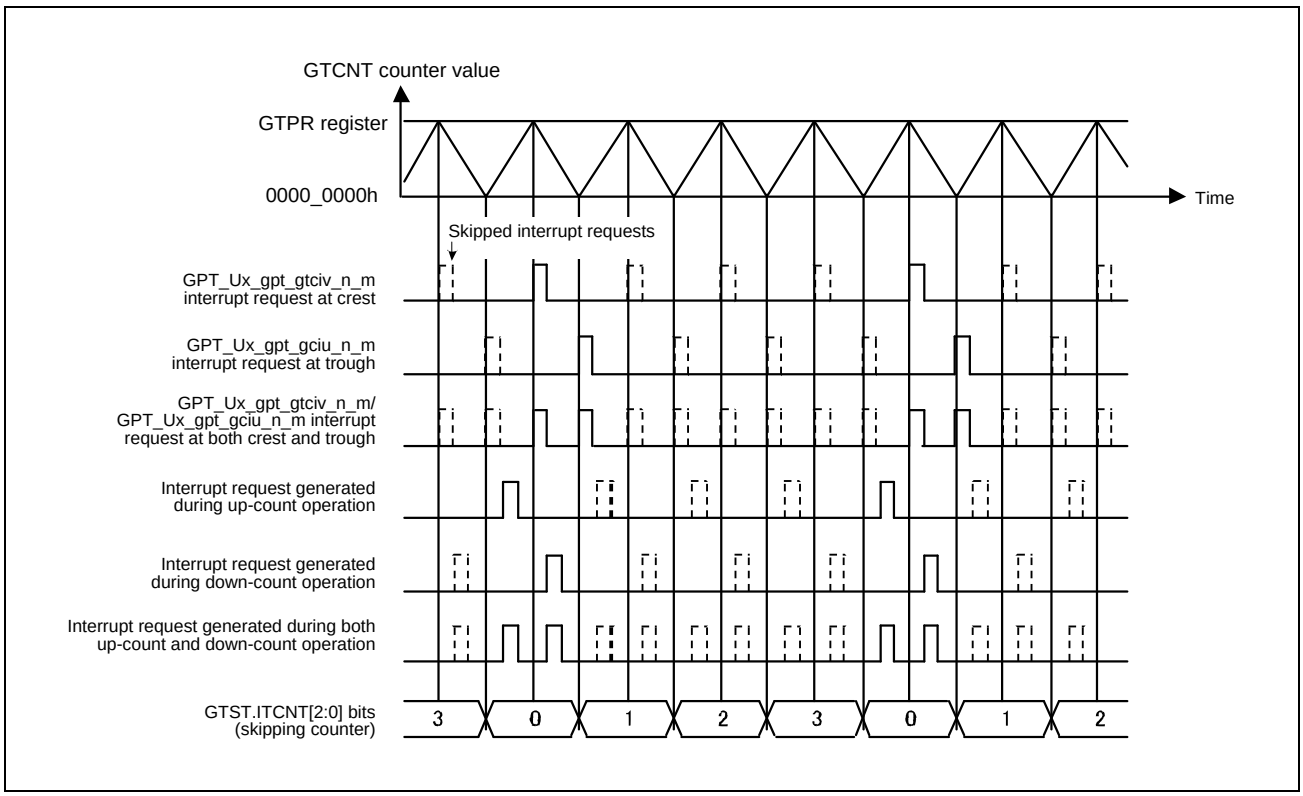


Figure 5.7-61 Example of Interrupt Skipping Function Operation (Triangle Waves, Counting and Skipping Troughs, Skipping Count: 3)

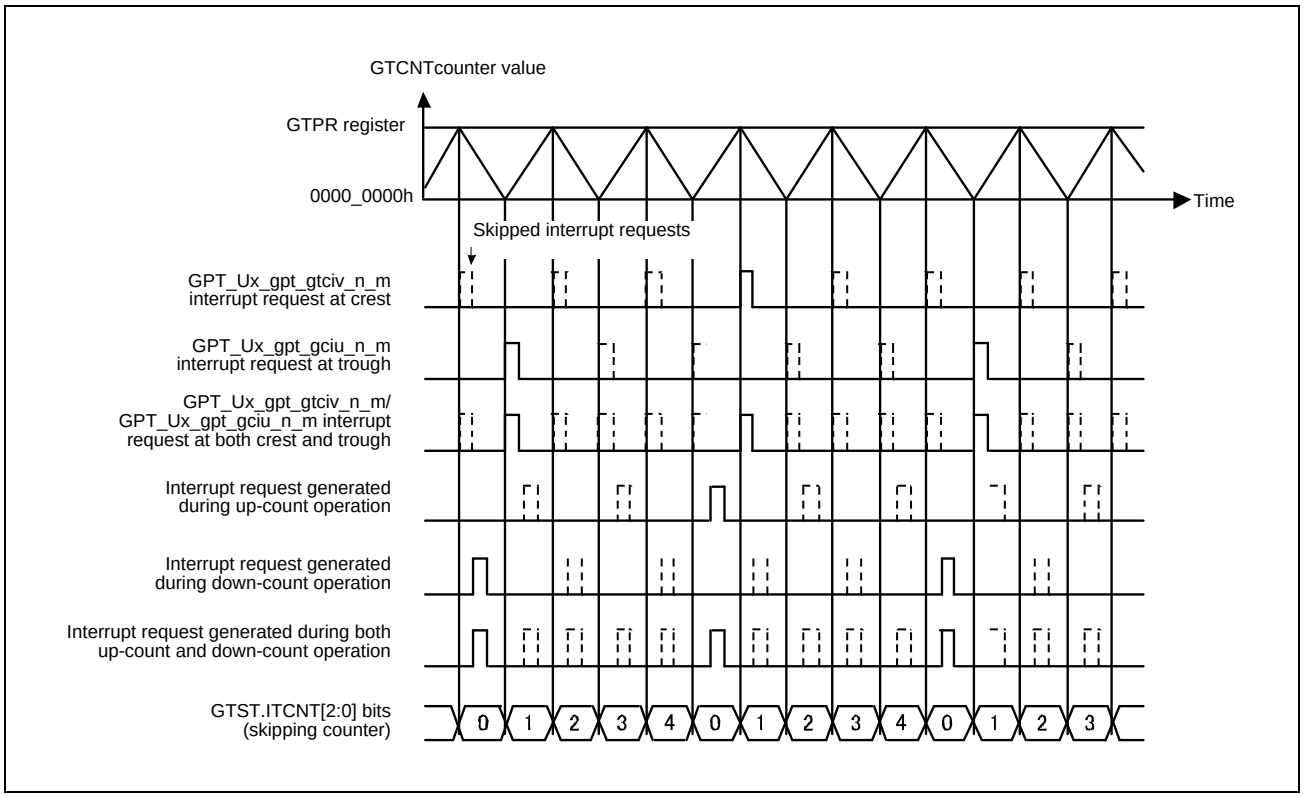


Figure 5.7-62 Example of Interrupt Skipping Function Operation (Triangle Waves, Counting and Skipping Both Troughs and Crests, Skipping Count: 4)

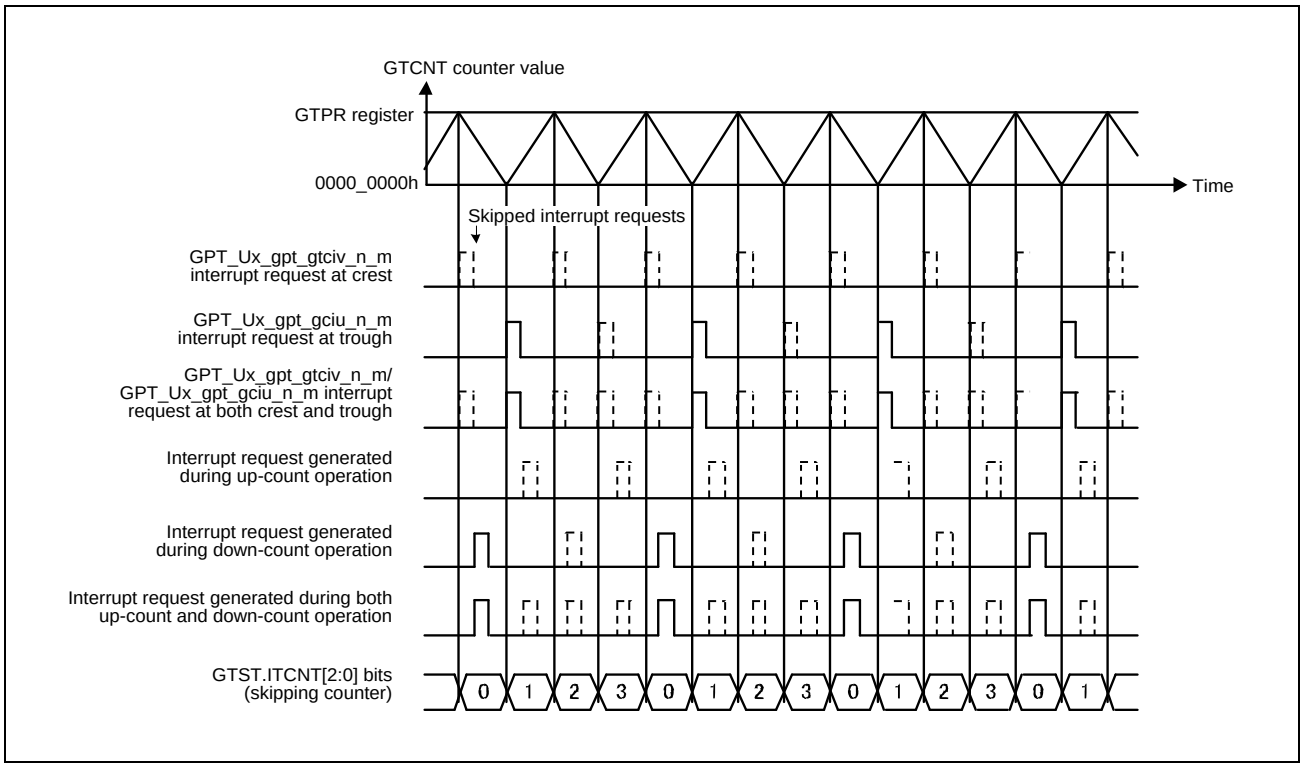


Figure 5.7-63 Example of Interrupt Skipping Function Operation (Triangle Waves, Counting and Skipping Both Troughs and Crests, Skipping Count: 3, Skipping Started at Up-Counting)

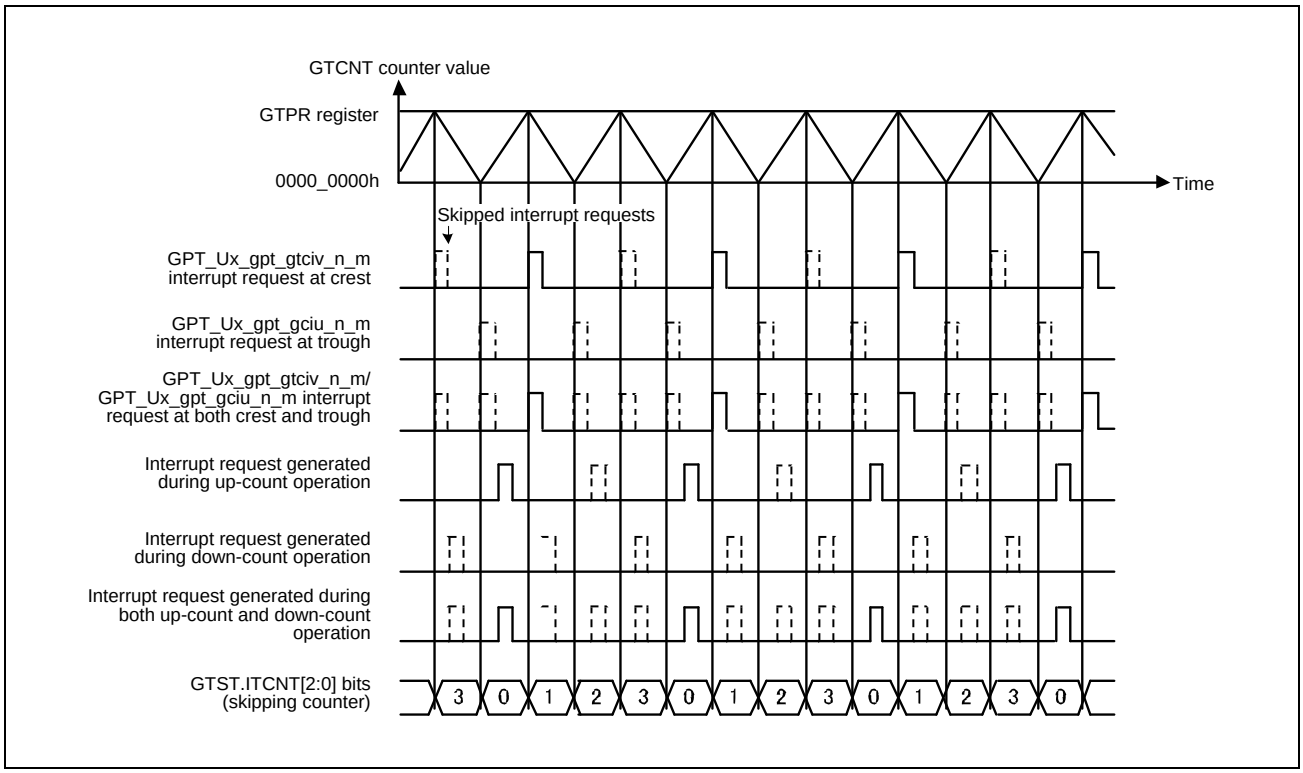


Figure 5.7-64 Example of Interrupt Skipping Function Operation (Triangle Waves, Counting and Skipping Both Troughs and Crests, Skipping Count: 3, Skipping Started at Down-Counting)

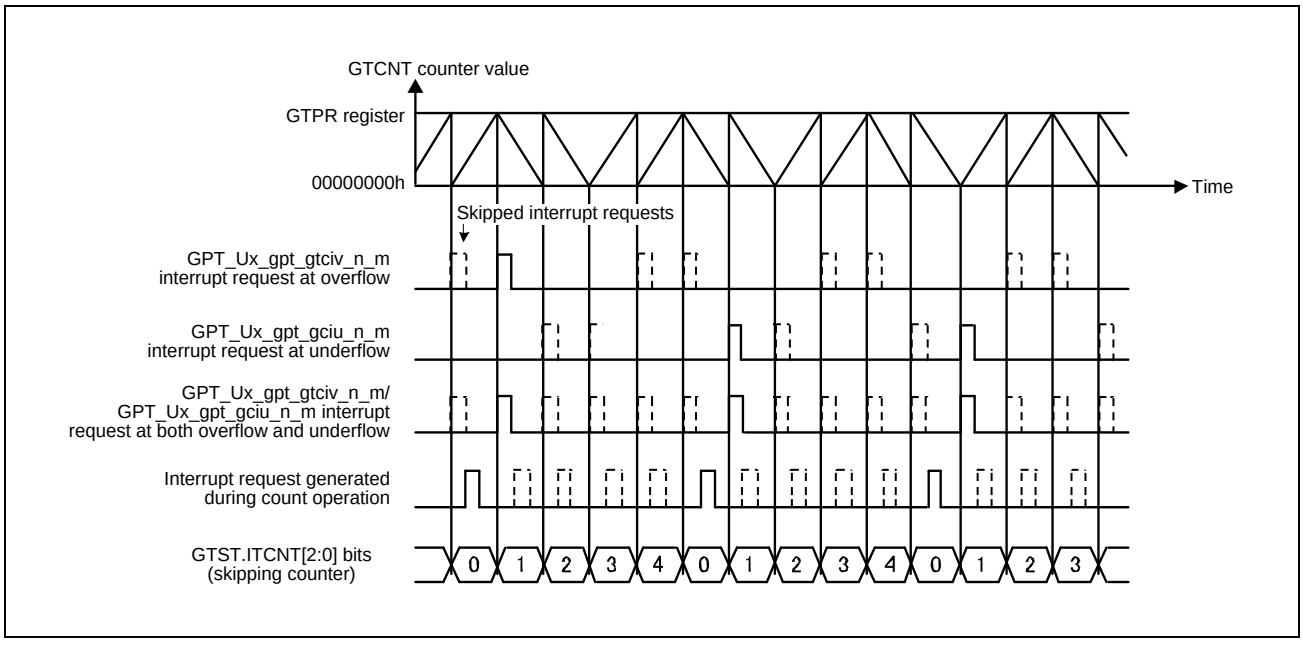


Figure 5.7-65 Example of Interrupt Skipping Function Operation (Saw Waves, Operation with Count Direction Changed, Counting and Skipping Both Overflows and Underflows, Skipping Count: 4)

5.7.4.3.2 Extended Interrupt Skipping Function

An overflow/underflow interrupt, compare match/input capture interrupt, A/D converter start request, and buffer transfer can be skipped by counting the GTCNT counter overflows or underflows through the settings of the GTEITC, GTEITLI1, GTEITLI2, and GTEITLB registers. A dead time error interrupt cannot be skipped. Simultaneous high-level output and low-level output interrupts cannot be skipped.

For ELC event outputs corresponding to interrupts and A/D converter start requests, two signals for skipping and no skipping are output.

Whether or not to perform skipping and the skipping period can be set individually for the overflow/underflow interrupt, compare match/input capture interrupt, A/D converter start request, and buffer transfer by using the GTEITLI1, GTEITLI2, and GTEITLB registers.

The skipping period is set as a period in which the counter value is other than 0 or other than the skipping count, in conjunction with the operation of the two independent extended interrupt skipping counters (GTEITC.EITCNT1[3:0] and EITCNT2[3:0] bits). It can also be selected when not only one skipping counter but two counters are both other than 0 or other than the skipping count.

Figure 5.7-66 shows an example of operation of the counter for interrupt skipping by the GTITC register and the counter for extended interrupt skipping.

The counter operation for extended interrupt skipping is set by the GTEITC register.

The EITCNT1[3:0] bits count the count source (a crest in **Figure 5.7-66**) selected by the extended interrupt skipping counter 1 count source select bits (EIVTC1[1:0] bits) when the initial value is 0b, and repeat counting which returns to 0 when the skipping count (2 in **Figure 5.7-66**) set by the extended interrupt skipping 1 skipping count setting bits (EIVTT1[3:0] bits) is reached.

The EITCNT2[3:0] bits can set the initial value, count the count source (a trough in **Figure 5.7-66**) selected by the extended interrupt skipping counter 2 count source select bits (EIVTC2[1:0] bits), and repeat counting which returns to 0 when the skipping count (2 in **Figure 5.7-66**) set by the EIVTT2[3:0] bits is reached. The initial value is set when the GTEITC register is written by access to the higher-order 16 or 32 bits and only when the value written to the EIVTC2[1:0] bits is other than 00b, while the setting of extended interrupt skipping counter 2 is for no counting (EIVTC2[1:0] bits are 00b). When the initial value is written, the value written to the extended interrupt skipping counter 2 initial value bits (EITCNT2IV[3:0] bits) is set in the EITCNT2[3:0] bits as the initial value.

The extended interrupt skipping counter counts up in the first cycle of the counter clock after the setting has been modified from no counting to counting.

The interrupt skipping count counter (GTST.ITCNT[2:0] bits) for interrupt skipping by the GTITC register is reset to 000b when the GTCNT counter operation is stopped; however, the EITCNT1[3:0] and EITCNT2[3:0] bits for extended interrupt skipping retain their values even after the GTCNT counter operation has been stopped, and counting can be resumed from the value before the counter is stopped. To reset the values of the EITCNT1[3:0] and EITCNT2[3:0] bits (to 0000b), set the EIVTC1[1:0] and EIVTC2[1:0] bits for no counting (no skipping) (00b).

When the skipping count is to be changed, only do so after stopping the skipping counter operation (set the EIVTC1[1:0] bits or EIVTC2[1:0] bits to 00b).

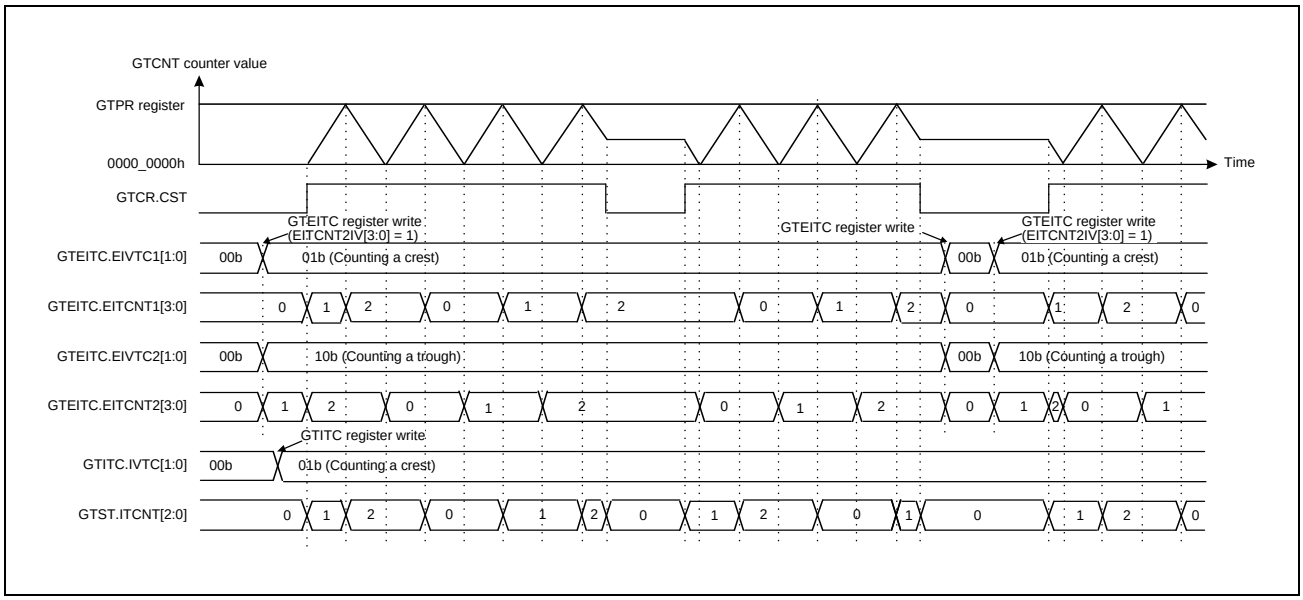


Figure 5.7-66 Example of Counter Operation for Interrupt Skipping

Interrupt skipping by the GTEITLI1 register and A/D converter start request skipping by the GTEITLI2 register can be performed simultaneously with interrupt skipping by the GTITC register. A skipping period at this time is represented by ORed skipping periods of respective registers. When the skipping period select bit is set to 1b, a skipping period is the logical AND of the respective skipping periods. **Figure 5.7-67** shows an example of operation when interrupt skipping operations by different registers are performed simultaneously.

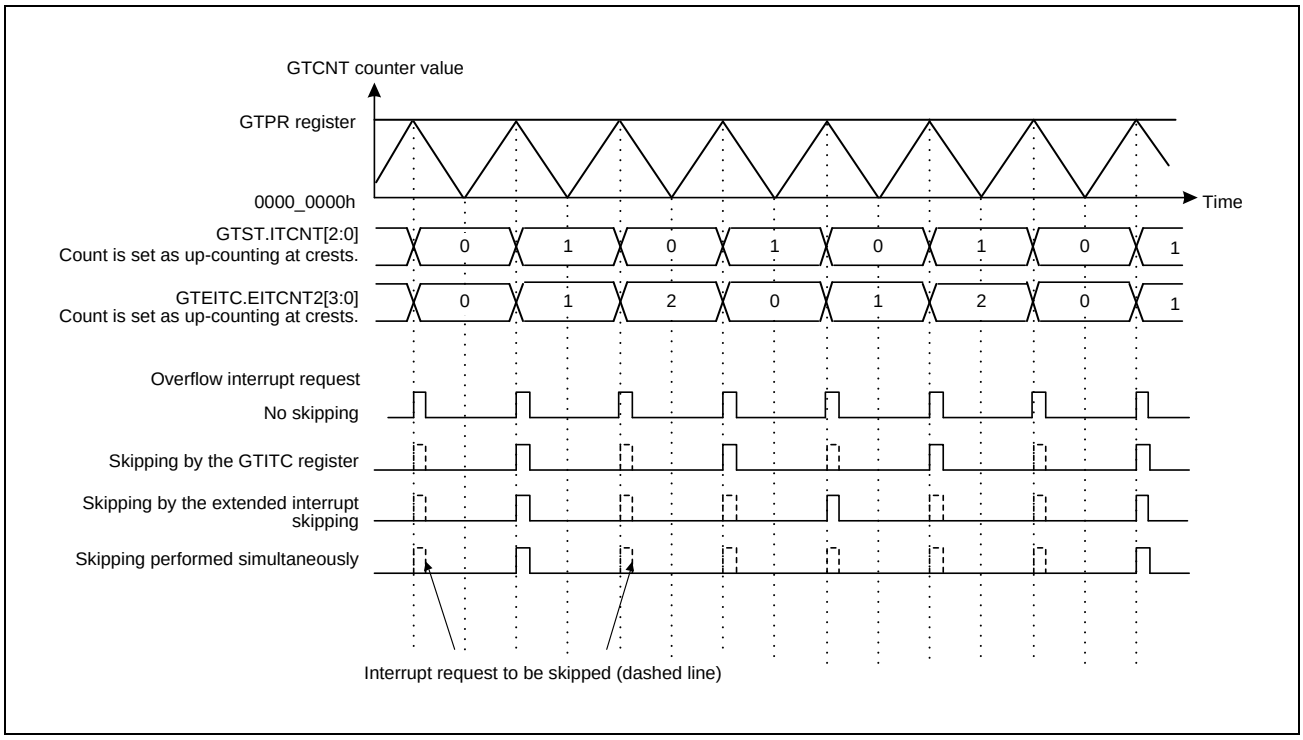


Figure 5.7-67 Example of Interrupt Skipping Operation (Skipping by the GTITC Register: Counting Crests, Extended Interrupt Skipping: EIVTC1[1:0] = 00b, EIVTC2[1:0] = 01b, EITLV[2:0] = 010b)

If interrupt skipping which can be set by the GTEITLI1 register is performed, a change in the corresponding status flag is also skipped and the skipping function continues to operate even while the status flag is 1b. Updating of the status flag corresponding to interrupts for which skipping can be set up in the GTEITLI1 register and skipping of ELC event outputs are only based on the settings of the GTITC register and the extended interrupt skipping register, and has no connection with the setting of the interrupt enable bit in the GTINTAD register. The interrupt enable bit is used only for the interrupt signal output after skipping.

If A/D converter start request skipping which can be set by the GTEITLI2 register is performed, updating of the corresponding status flag and ELC event outputs only depend on the setting the A/D converter start request enable bit in the GTINTAD register. Operations by A/D converter start requests which are set to “disabled” in the GTINTAD register are not all performed.

Buffer transfer skipping by the GTEITLB register is performed for buffer operation which is enabled in the GTBER and GTDTCR registers and for all buffer operations which are performed in saw-wave one-shot pulse mode or triangle-wave PWM mode 3.

Interrupt skipping and buffer transfer skipping operate independently. For buffer transfer which involves output of interrupts, interrupt output without buffer transfer and buffer transfer without interrupt output are both possible.

The extended interrupt skipping counter 2 initial value is set by the value written to the ETICNT2IV[3:0] bits which is applied to change the setting from no counting of the extended interrupt skipping counter 2 count source (GTEITC.EIVTC2[1:0] = 00b) to counting (EIVTC2[1:0] = other than 00b). Writing to the extended interrupt skipping counter 2 initial value bits (ETICNT2IV[3:0]) is performed only when the setting of the above-mentioned extended interrupt skipping counter 2 initial value is written.

1. Set GTCNT counter operation, set buffer operation, and set compare match value
2. Set the extended interrupt skipping function
   Select the skipping counter used for skipping and the skipping period with the skipping function select bit for the interrupt, for the A/D converter start request, and for a buffer transfer, all of which are to be skipped, in the GTEITLI1, GTEITLI2, and GTEITLB registers, respectively.
3. Set the extended skipping counter
   With the GTEITC register, set a count source of the skipping counter used for skipping, skipping count, and the skipping counter 2 initial value in the following order.
   Set the EIVTCk[1:0] bits (k = 1, 2) to a value other than 00b, and set the EIVTTk[3:0] bits to a value other than 0000b. When the skipping counter 2 is used, change the EIVTC2[1:0] bits from 00b to a value other than 00b as well as set the EITCNT2IV[3:0] bits to the skipping counter 2 initial value.
4. Start count operation
   Set the GTCR.CST bit to 1b to start count operation.
5. Set buffer value by cycle

5.7.5 A/D Converter Start Request

An A/D converter start request can be generated by a compare match between the GTCNT counter and the GTADTRA or GTADTRB register. It can be generated in up-counting only, down-counting only, or both up-counting and down-counting by the setting of the GTINTAD register.

An A/D converter start request cannot be generated during event counting.

The A/D converter start request is not output directly to the A/D converter but the interrupt and event signals to the ELC are output.

The GTADTRA and GTADTRB registers each have two buffer registers. Buffer operation of the GTADTRA register in combination with the GTADTBRA and GTADTDBRA registers and buffer operation of the GTADTRB register in combination with the GTADTBRB and GTADTDBRB registers are possible.

The timing of the generation of A/D converter start requests can be monitored by an external pin. When the A/D converter start request signal to be monitored is selected in the GTADSMR.ADSMSk bit (k = 0, 1) and when the output is enabled in the ADSMENk bit, a signal is output synchronized with a cycle frame of the timer used to generate the A/D converter start request signal, of which the output is driven high at the generation of the A/D converter start request signal by the GTADSMk pin, or at the end of the cycle of which the output is driven low. When a signal to request the start of A/D conversion is generated at the end of the cycle, the generation of this signal has priority in terms of monitoring output and the output remains at the high level till the end of the next cycle. The registers (GTADTRA and GTADTRB) that are sources of generating the A/D converter start request signals and their counting directions can be checked by the A/D converter start request flags (ADTRAUF, ADTRADF, ADTRBUF, and ADTRBDF) in the GTST register. When the output of the same A/D converter start request signal monitoring output is enabled for multiple channels, ORed signals will be output from the GPTW.

Figure 5.7-68 shows an example of A/D converter start request operation and **Figure 5.7-69** shows an example of the settings for A/D converter start request operation.

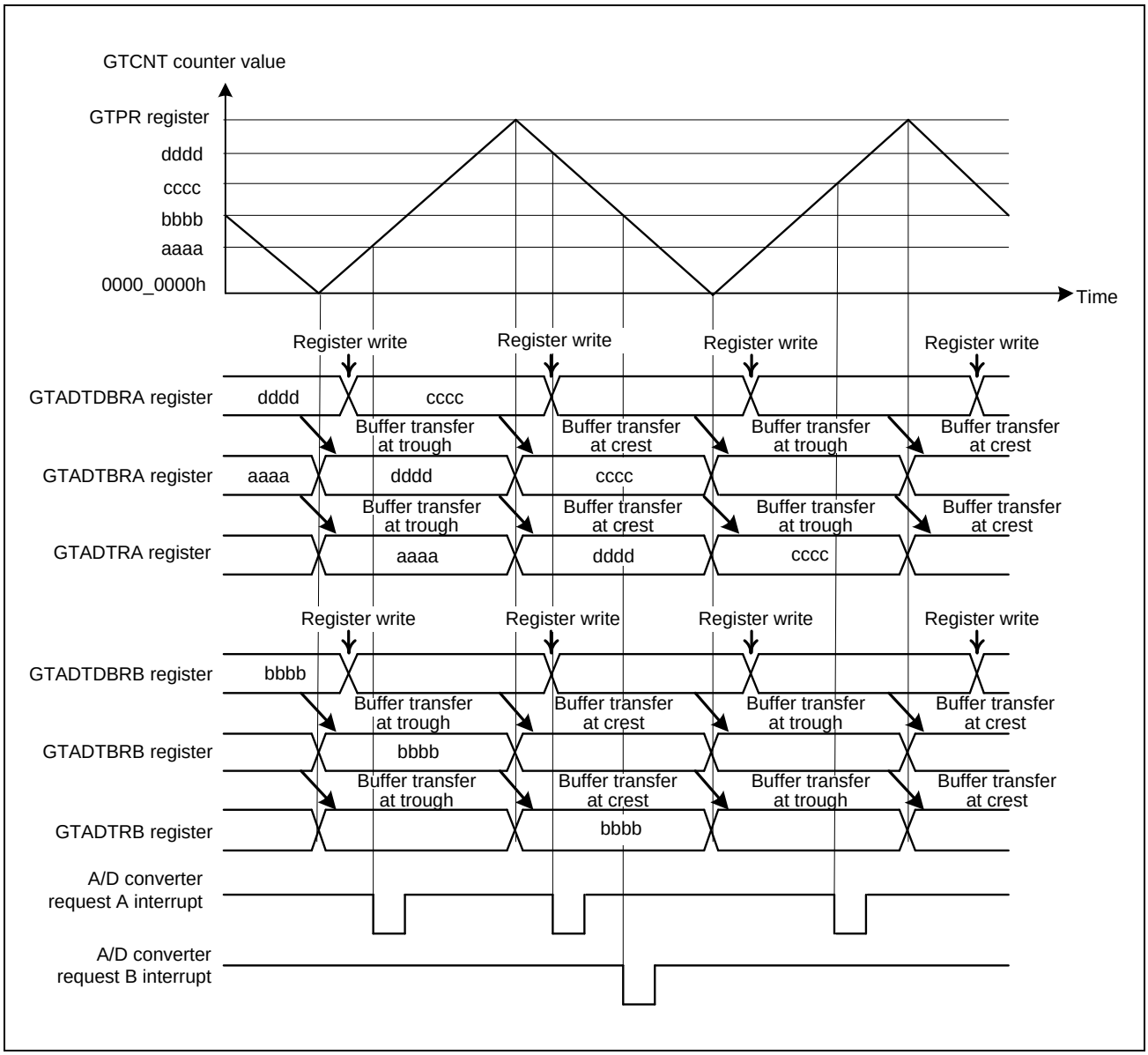


Figure 5.7-68 Example of A/D Converter Start Request Operation (Triangle Waves, Double Buffer Operation, Buffer Transfer at Both Troughs and Crests, A/D Converter Start Request Interrupt Generation by GTADTRA at Both Up-Counting and Down-Counting, A/D Converter Start Request Interrupt Generation by GTADTRB at Down-Counting, Monitoring of GTADTRA Up-Counting by GTADSM0, Monitoring of GTADTRB Down-Counting by GTADSM1)

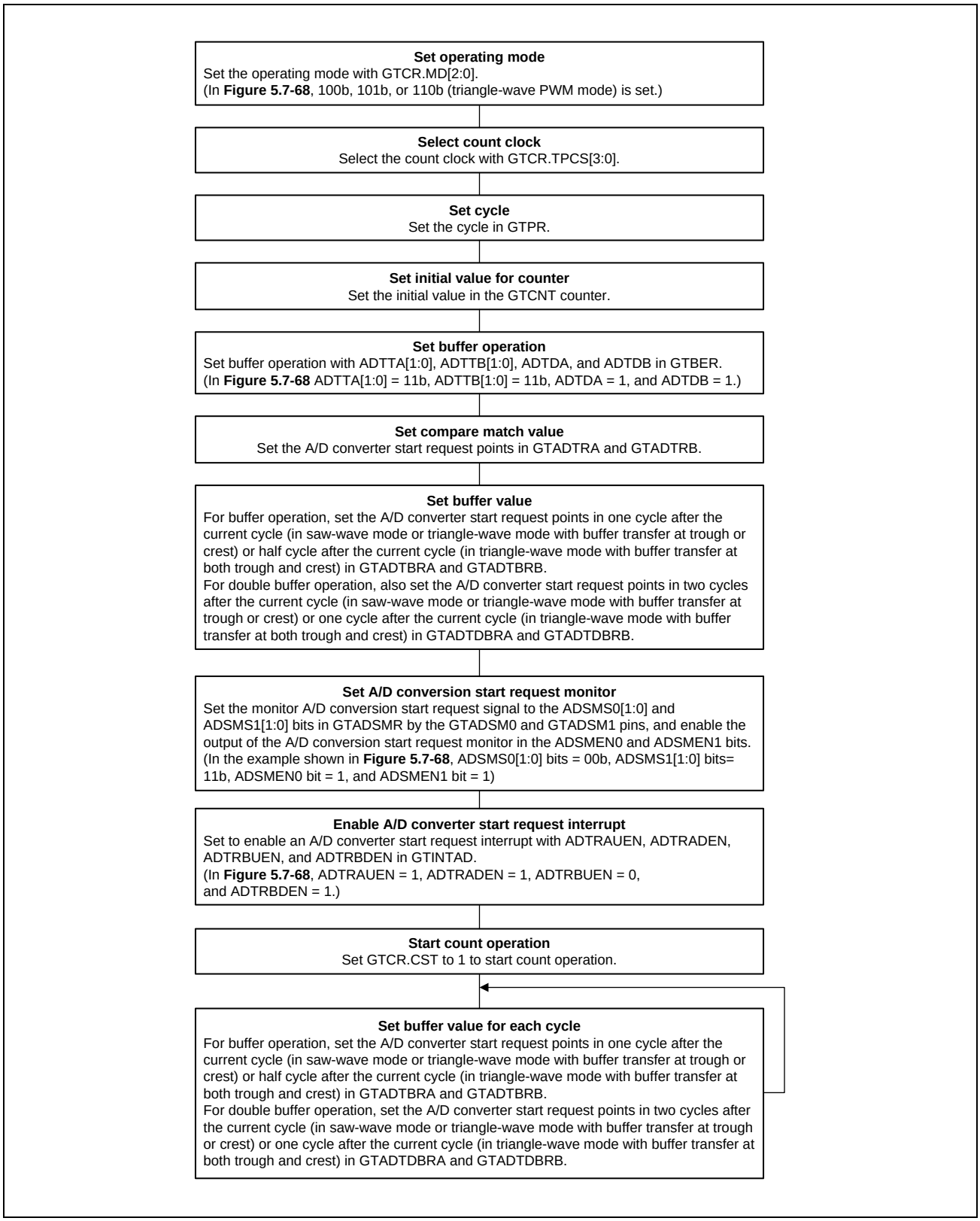


Figure 5.7-69 Example of Settings for A/D Converter Start Request Operation

5.7.6 Operations Linked by ELC

5.7.6.1 Event Signal Output to ELC

The GPT can perform operation linked with another module set in advance when its interrupt request signal is used as an event signal by the Event Link Controller (ELC).

The GPT has the following ELC event signals (“GPT_Ux_” (x = 0, 1) is omitted.):

- Generation of compare match and input capture A interrupt (gpt_elccmpa_m)
- Generation of compare match and input capture B interrupt (gpt_elccmpb_m)
- Generation of compare match C interrupt (gpt_elccmpc_m)
- Generation of compare match D interrupt (gpt_elccmpd_m)
- Generation of compare match E interrupt (gpt_elccmpe_m)
- Generation of compare match F interrupt (gpt_elccmpf_m)
- Generation of overflow interrupt (gpt_elcovf_m)
- Generation of underflow interrupt (gpt_elcudf_m)

Note: m = 0 to 7

5.7.6.2 Event Signal Inputs from ELC

The GPT can perform the following operations in response to a maximum of 8 events from the ELC:

- Start counting, stop counting, clear counting
- Up-counting, down-counting
- Input capture.

See **4.2.1.6 Event Link Controller (ELC)** for the connection between the ELC and the event signal input.

5.7.7 Noise Filter Function

The pin for use in input capture to the GPT is equipped with a noise filter. The noise filter samples input signals at the sampling clock and removes the pulses whose length is less than 3 sampling cycles.

The noise filter functionality includes enabling and disabling the noise filter for each pin and setting of the sampling clock for each channel.

Figure 5.7-70 shows the timing of noise filtering.

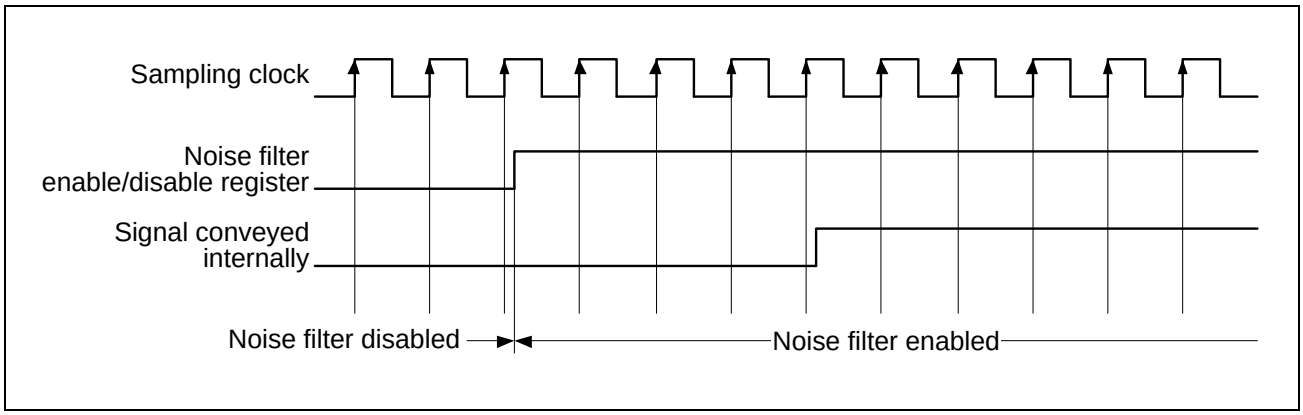


Figure 5.7-70 Timing of Noise Filtering

If noise filtering is enabled, the input capture operation or output phase switching operation is performed on the edges of the noise filtered signal after a delay of (sampling interval × 2 + PCLKD) at the shortest. This is due to the noise filtering for the input capture input or hall sensor input.

5.7.8 Protection Function

5.7.8.1 Write-Protection for Registers

To prevent registers from being accidentally modified, registers can be write-protected in channel units by setting GTWP.WP. Write-protection can be set for the following registers:

GTSSR, GTPSR, GTCSR, GTUPSR, GTDNSR, GTICASR, GTICBSR, GTCR, GTUDDTYC, GTIOR, GTINTAD, GTST, GTBER, GTITC, GTCNT, GTCCRA, GTCCRB, GTCCRC, GTCCRD, GTCCRE, GTCCRF, GTPR, GTPBR, GTPDBR, GTADTRA, GTADTBRA, GTADTDBRA, GTADTRB, GTADTBRB, GTADTDBRB, GTDTCR, GTDVU, GTDVD, GTDBU, GTDBD, GTSOS, GTSOTR, GTEITC, GTEITLI1, GTEITLI2, GTEITLB, GTICLF.

Every bit in registers GTSTR, GTSTP and GTCLR which can update the corresponding registers in other channels and can be updated by any of the corresponding registers in other channels conversely, can be protected by setting the GTWP.STRWP, STPWP, and CLRWP bits, respectively, per channel.

Likewise, writing to the GTSECSR and GTSECR registers, which can control all channels by writing to the GTSECSR and GTSECR registers of a given channel, can be enabled or disabled by the setting of the GTWP.CMNWP bit.

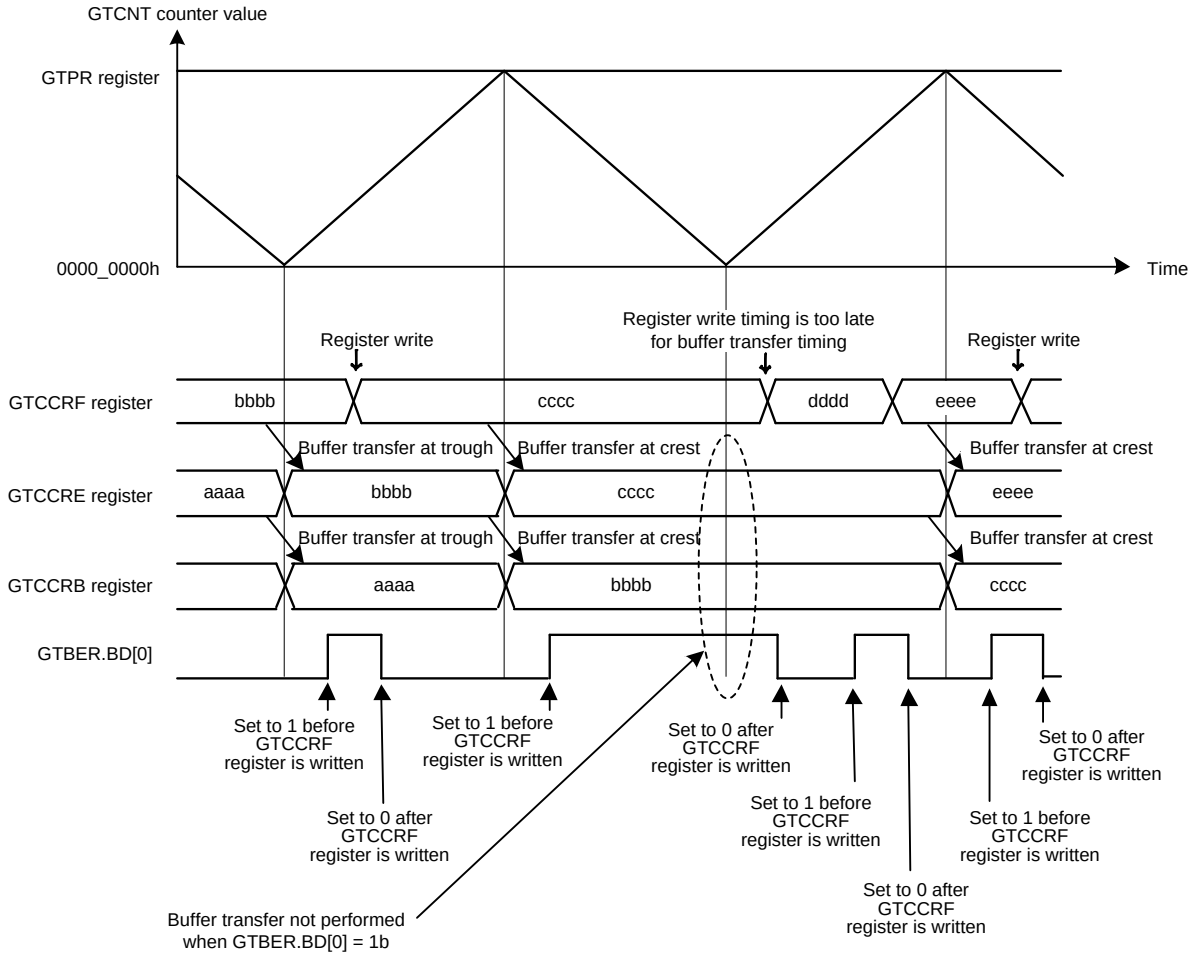
Protection using the GTWP register is only for write operations by the CPU. This protection does not cover updates to registers that occur in association with CPU writes.

5.7.8.2 Disabling of Buffer Operation

If the timing of the buffer register write is delayed relative to the timing for the buffer transfer, buffer operation can be suspended with the GTBER.BD[1] and BD[0] bits settings. Specifically, buffer transfer can be temporarily disabled even though a buffer transfer condition is generated during buffer register write, by setting the BD[1] and BD[0] bits to 1b (buffer operation disabled) before buffer register write, and setting the bits to 0b (buffer operation enabled) after completion of writing to all the buffer registers.

The BD[1] and BD[0] bits can be set on channel basis by writing directly to the GTBER register or it can be set to 0b simultaneously by setting the GTSECR register for multiple channels which were set by the GTSECSR register.

Figure 5.7-71 shows an example of operation for disabling buffer operation by writing to the GTBER register.



Note: The base addresses of the registers are the same.

Figure 5.7-71 Example of Operation for Disabling Buffer Operation with Triangle Waves, Double Buffer Operation, and Buffer Transfer at Both Troughs and Crests

5.7.8.2.1 Simultaneous Control of Buffer Operations of Multiple Channels

The GTBER.BD bit can be set by writing directly to the GTBER register per channel or by making settings in the GTSECR register for multiple channels that have already set in the GTSECSR register.

Follow the procedure below to simultaneously set the GTBER.BD bits of multiple channels.

1. Select the channels for simultaneously setting by the GTSECSR register
   Set the GTSECSR register so that the values at the bit positions for the corresponding channels for simultaneously setting of the GTBER.BD bits become 1b. All GTSECSR registers can be updated by writing to the GTSECSR register of any channel.
2. Simultaneously set the GTBER.BD bits by updating the GTSECR register
   In the GTSECR register, set the operation of the GTBER.BD bits (enabling or disabling of buffer operation) which are to be simultaneously set. Writing to a GTSECR register from any channel updates the GTBER.BD bits in all channels corresponding to the bits set as 1b in the GTSECSR register, in accordance with the value of the GTSECR register.

Figure 5.7-72 shows examples of simultaneously controlling the enabling or disabling of buffer operation for multiple channels.

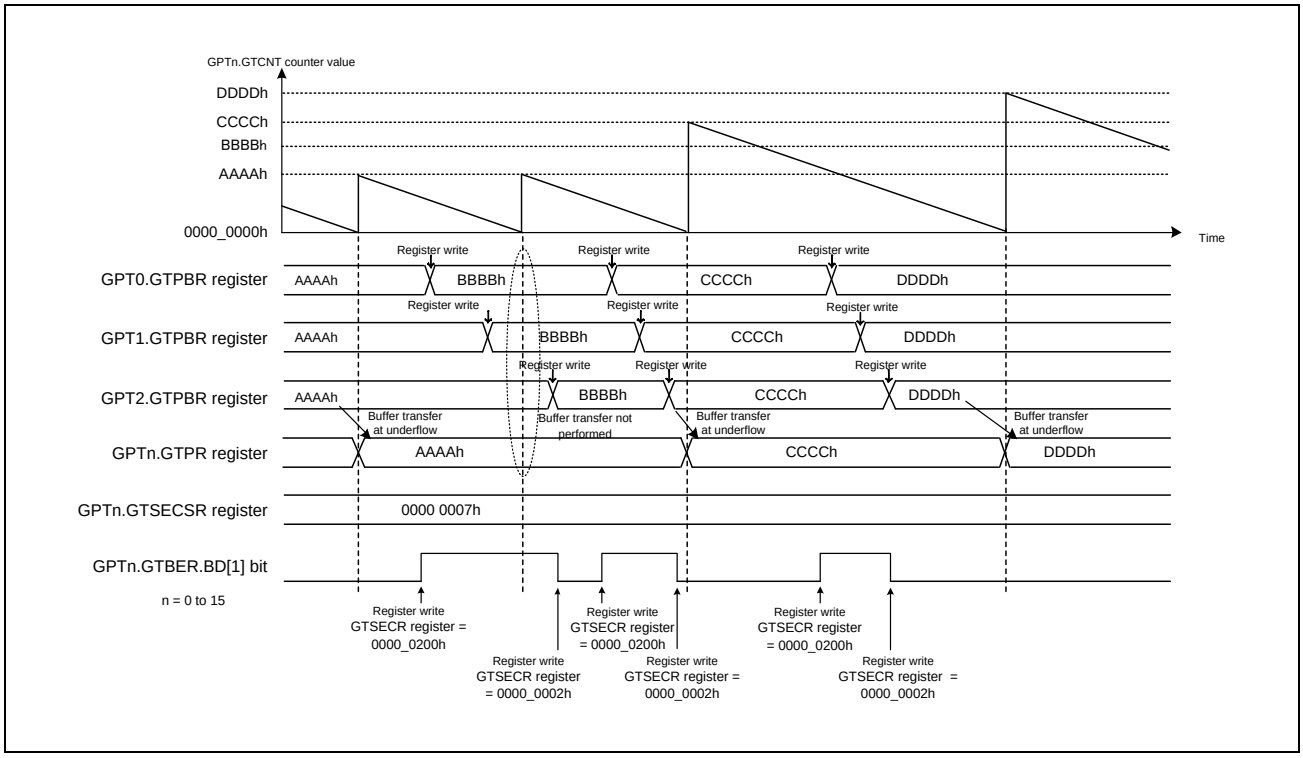


Figure 5.7-72 Example of Multiple Channel Operation for Disabling Buffer Operation (Saw Waves, Single Buffer Operation)

5.7.8.2.2 Repeated Double-Buffered Operation when GTCCR Buffer Transfer is Inhibited

If saw-wave one-shot pulse mode, triangle-wave PWM mode 3, or triangle-wave PWM mode 2 that operates the GTCCRm (m =A, B) register as a double buffer is used, setting the GTBER.DBRTECm (m = A, B) bit to 1b repeats transfer from the intermediate buffer to the GTCCRm (m = A, B) register on a cyclic basis even while buffer transfer is inhibited by the setting of the GTBER.BD[0] bit or by the buffer transfer extended skipping function.

(1) In saw-wave one-shot pulse mode

In saw-wave one-shot pulse mode, the compare match value for the first half of the cycle is stored in temporary register x (x = C, E) as the intermediate buffer for the GTCCRx (x = C, E) register and the compare match value for the second half of the cycle is stored in temporary register m (m = A, B) as the intermediate buffer for the GTCCRy (y = D, F) register, respectively, for compare match values during repeated operation, and the given values are alternately transferred to the GTCCRm (m = A, B) registers. **Table 5.7-47** lists the types of buffer transfer of the GTCCR register during counting in saw-wave one-shot pulse mode.

While counting is stopped, the setting of the value in the temporary register is transferred through forcible buffer transfer. In forcible buffer transfer, the values of the GTCCRy (y = D, F) registers are transferred to temporary registers m (m = A, B), and the values of the GTCCRx (x = C, E) registers are transferred to temporary registers x (x = C, E), when the setting of the corresponding GTBER.DBRTECm (m = A, B) bit is 1b, respectively.

When the setting of the GTBER.DBRTECm (m = A, B) bit is 1b, values written by the CPU to the GTCCRm (m = A, B) registers are reflected as the values of temporary registers x (x = C, E).

Table 5.7-47 GTCCR Buffer Operation in Saw-Wave One-Shot Pulse Mode (during Counting)

GTBER.DBRTECm Bit	Buffer Transfer	Transfer Timing				
		GTCCRx register ↓ GTCCRm register	GTCCRx register ↓ Temporary register x	Temporary register x ↓ GTCCRm register	GTCCRy register ↓ Temporary register m	Temporary register m ↓ GTCCRm register
0b	Transfer-enabled period	Overflow or underflow	No transfer	No transfer	Overflow or underflow	GTCCRm compare match
	Transfer-inhibited period	No transfer	No transfer	No transfer	No transfer	No transfer
1b	Transfer-enabled period	Overflow or underflow	Overflow or underflow	No transfer	Overflow or underflow	GTCCRm compare match
	Transfer-inhibited period	No transfer	No transfer	Overflow or underflow	No transfer	GTCCRm compare match

Note: m: A, B
x: C, E
y: D, F

Figure 5.7-73 shows the case when a transfer-inhibited period is generated by extended buffer transfer skipping, as an example of repeated double-buffered operation when the GTCCR buffer transfer is inhibited in saw-wave one-shot pulse mode. **Figure 5.7-74** shows the case when a transfer-inhibited period is generated by updating the GTBER.BD[0] bit, as an example of repeated double-buffered operation when the GTCCR buffer transfer is inhibited in saw-wave one-shot pulse mode.

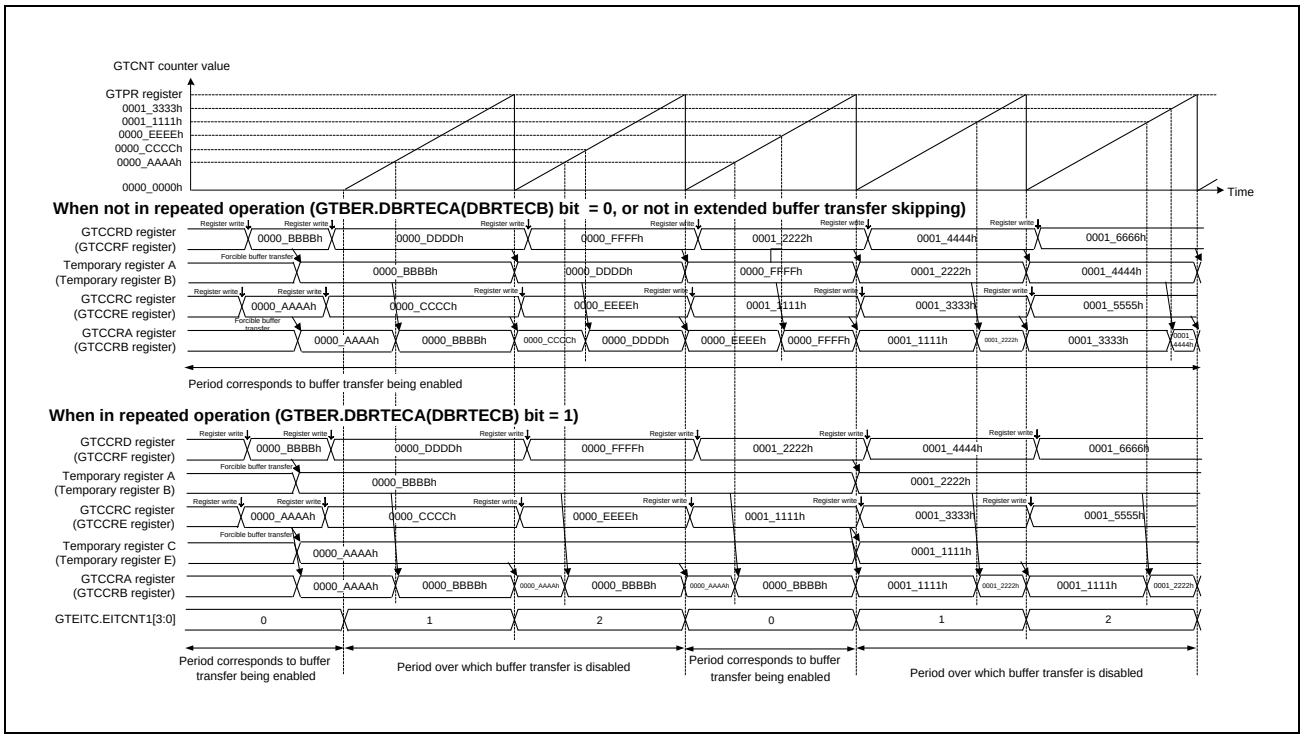


Figure 5.7-73 Example of Repeated Double-Buffered Operation when GTCCR Buffer Transfer is Inhibited (in Saw-Wave One-Shot Pulse Mode, Extended Buffer Transfer Skipping is Used, the GTBER.BD[0] Bit is Always 0)

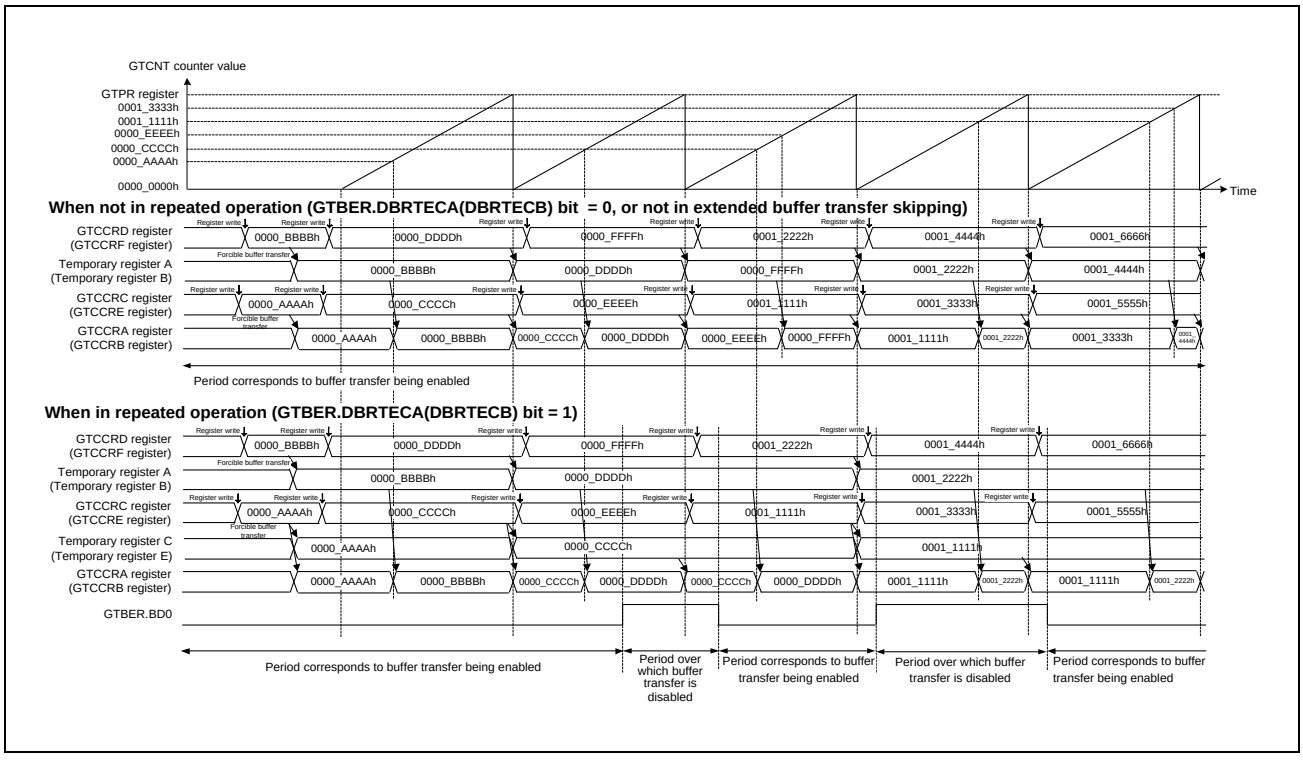


Figure 5.7-74 Example of Repeated Double-Buffered Operation when GTCCR Buffer Transfer is Inhibited (in Saw-Wave One-Shot Pulse Mode, the GTBER.BD[0] Bit is Updated)

(2) In triangle-wave PWM mode 3

In triangle-wave PWM mode 3, the compare match value for the first half of the cycle is stored in temporary register x (x = C, E) as the intermediate buffer for the GTCCRx (x = C, E) register and the compare match value for the second half of the cycle is stored in temporary register m (m = A, B) as the intermediate buffer for the GTCCRy (y = D, F) register, respectively, for compare match values during repeated operation, and the given values are alternately transferred to the GTCCRm (m = A, B) register. **Table 5.7-48** lists the types of buffer transfer of the GTCCR register during counting in triangle-wave PWM mode 3.

While counting is stopped, the setting of the value in the temporary register is transferred through forcible buffer transfer. In forcible buffer transfer, the values of the GTCCRy (y = D, F) registers are transferred to temporary registers m (m = A, B), and the values of the GTCCRx (x = C, E) registers are transferred to temporary registers x (x = C, E), when the setting of the corresponding GTBER.DBRTECm (m = A, B) bit is 1b, respectively.

When the setting of the GTBER.DBRTECm (m = A, B) bit 1b, values written by the CPU to the GTCCRm (m = A, B) registers are reflected as the values of temporary registers x (x = C, E).

Table 5.7-48 GTCCR Buffer Operation in Triangle-Wave PWM Mode 3 (during Counting)

GTBER.DBRTECm Bit	Buffer Transfer	Transfer Timing				
		GTCCRx register ↓ GTCCRm register	GTCCRx register ↓ Temporary register x	Temporary register x ↓ GTCCRm register	GTCCRy register ↓ Temporary register m	Temporary register m ↓ GTCCRm register
0b	Transfer-enabled period	Trough	No transfer	No transfer	Trough	Crest
	Transfer-inhibited period	No transfer	No transfer	No transfer	No transfer	No transfer
1b	Transfer-enabled period	Trough	Trough	No transfer	Trough	Crest
	Transfer-inhibited period	No transfer	No transfer	Trough	No transfer	Crest

Note: m: A, B
x: C, E
y: D, F

Figure 5.7-75 shows the case when a transfer-inhibited period is generated by extended buffer transfer skipping, as an example of repeated double-buffered operation when the GTCCR buffer transfer is inhibited in triangle-wave PWM mode 3. **Figure 5.7-76** shows the case when a transfer-inhibited period is generated by updating the GTBER.BD[0] bit, as an example of repeated double-buffered operation when the GTCCR buffer transfer is inhibited in triangle-wave PWM mode 3.

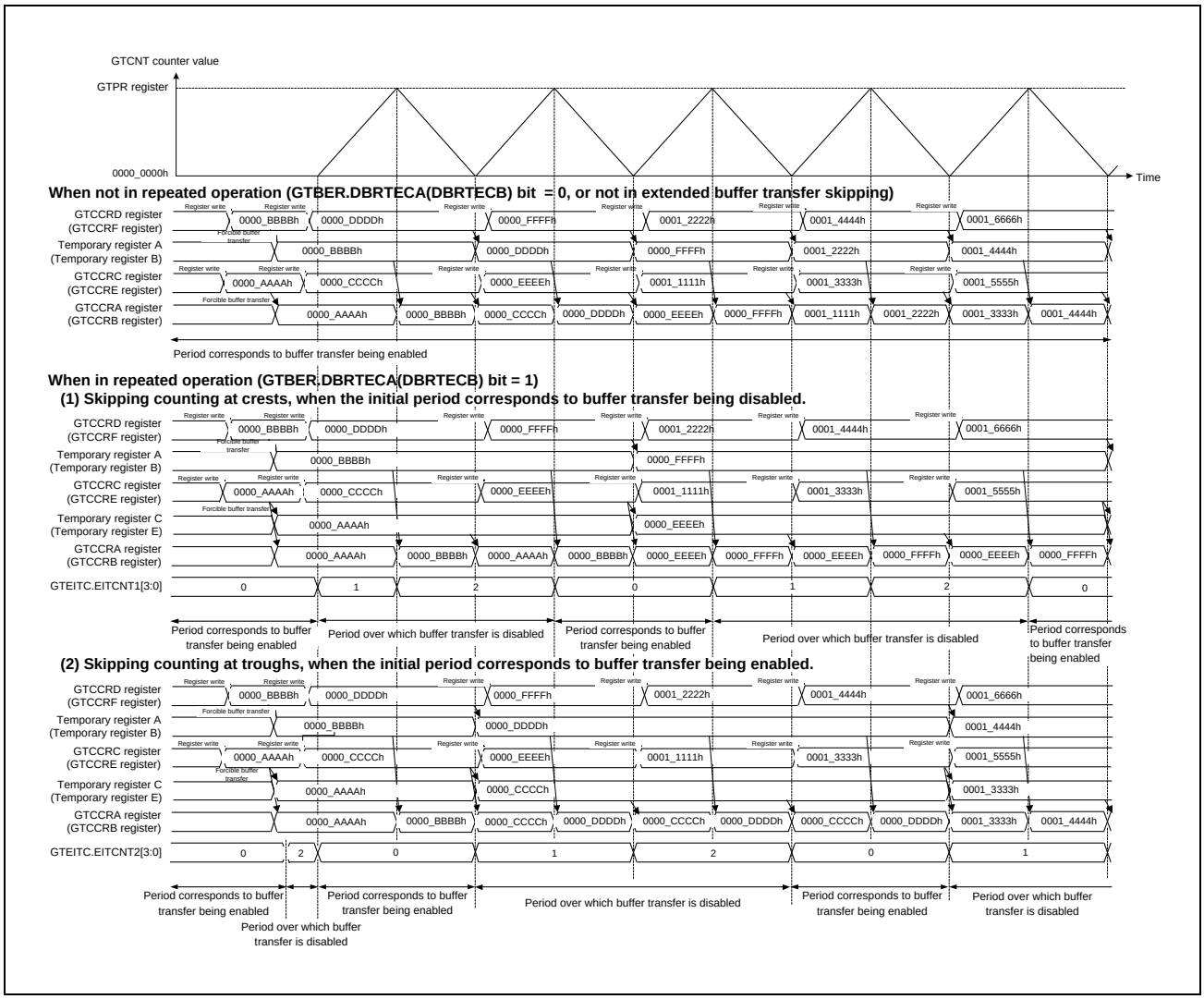


Figure 5.7-75 Example of Repeated Double-Buffered Operation when GTCCR Buffer Transfer is Inhibited (in Triangle-Wave PWM Mode 3, Extended Buffer Transfer Skipping is Used, the GTBER.BD[0] Bit is Always 0)

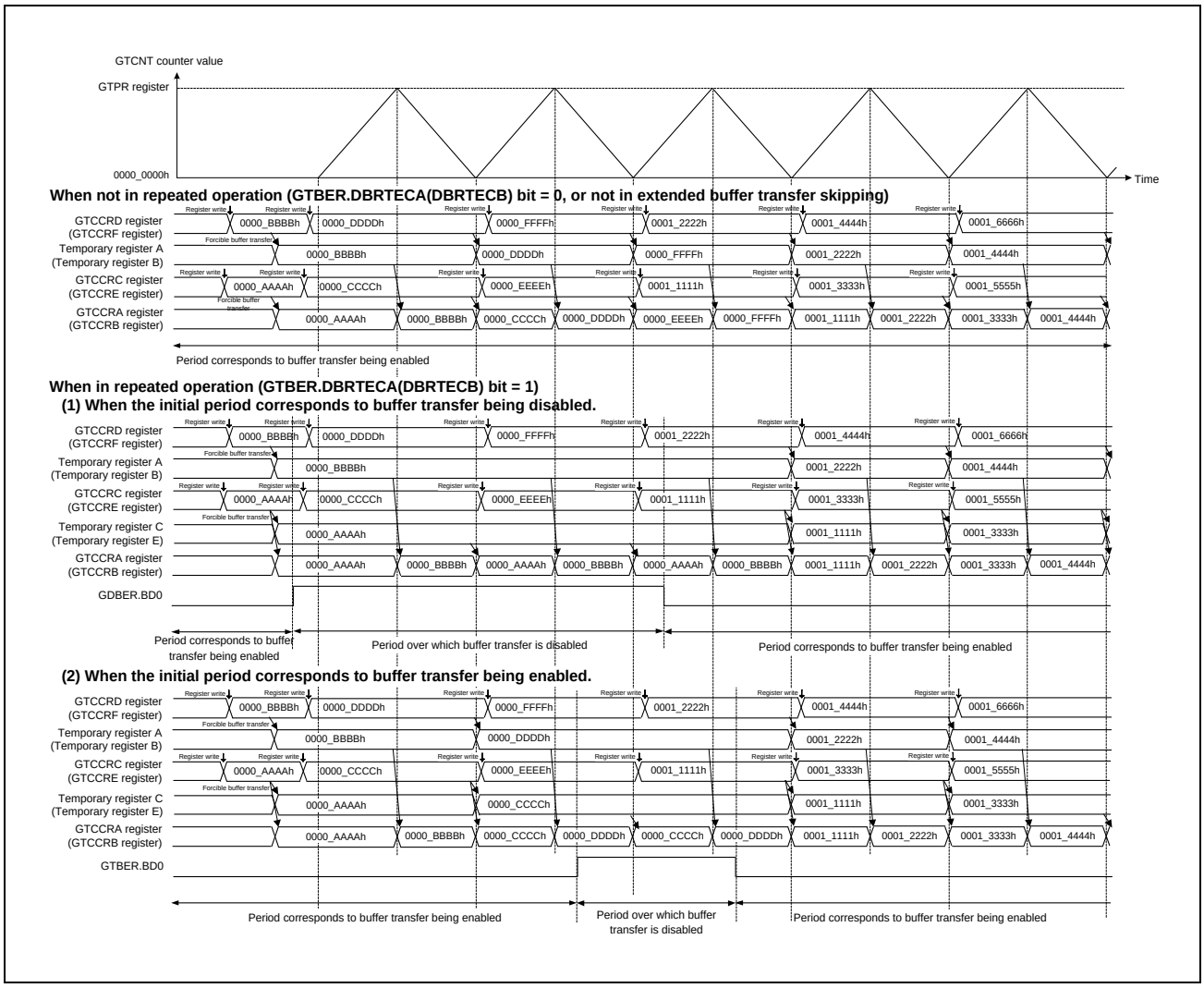


Figure 5.7-76 Example of Repeated Double-Buffered Operation when GTCCR Buffer Transfer is Inhibited (in Triangle-Wave PWM Mode 3, the GTBER.BD[0] Bit is Updated)

(3) In triangle-wave PWM mode 2

In triangle-wave PWM mode 2, the compare match value for the first half of the cycle is stored in temporary register x (x = C, E) as the intermediate buffer for the GTCCRx (x = C, E) register and the compare match value for the second half of the cycle is stored in the GTCCRx (x = C, E) register, respectively, for compare match values during repeated operation, and the given values are alternately transferred to the GTCCRm (m = A, B) register. The first and second halves of a waveform is determined by the setting of the GTCCRm double-buffer repeat operation period select bit GTBER.DBRTSCm (m = A, B). **Table 5.7-49** lists the types of buffer transfer of the GTCCR register during counting in triangle-wave PWM mode 2.

While counting is stopped, set the compare match value for the first half of the cycle following the start of counting in temporary register x (x = C, E) beforehand. When the setting of the GTBER.DBRTECm (m = A, B) bit is 1b, values written by the CPU to the GTCCRm (m = A, B) registers are reflected as the values of temporary registers x (x = C, E).

If the start of counting is at the second half of the cycle, set the value for the second half of the cycle in the GTCCRm (m = A, B) register beforehand when counting is stopped. If the start of counting is at the second half of the cycle and in the buffer transfer inhibited period, set the same value as the GTCCRx (x = C, E) register in the GTCCRm (m = A, B) register. When setting the value for the second half of the cycle in the GTCCRm (m = A, B) register when counting is stopped, set the value for the first half of the cycle in temporary register x (x = C, E), and then return the value of the GTBER.DBRTECm (m = A, B) bit to 0 once before writing by the CPU. After writing, set the GTBER.DBRTECm (m = A, B) bit to 1b before counting starts.

Table 5.7-49 GTCCR Buffer Operation in Triangle-Wave PWM Mode 2 (during Counting)

GTBER.DBRTECm Bit	GTBER.DBRTSCm Bit	Buffer Transfer	Transfer Timing			
			GTCCRy register ↓ GTCCRx register	GTCCRx register ↓ GTCCRm register	GTCCRy register ↓ Temporary register x	Temporary register x ↓ GTCCRm register
0b	—	Transfer-enabled period	Both trough and crest	Both trough and crest	No transfer	No transfer
		Transfer-inhibited period	No transfer	No transfer	No transfer	No transfer
1b	0b (trough cycle)	Transfer-enabled period	Both trough and crest	Both trough and crest	Crest	No transfer
		Transfer-inhibited period	No transfer	Crest	No transfer	Trough
	1b (crest cycle)	Transfer-enabled period	Both trough and crest	Both trough and crest	Trough	No transfer
		Transfer-inhibited period	No transfer	Trough	No transfer	Crest

Note: m: A, B
x: C, E
y: D, F

5.7.8.2.3 Repeated Double-Buffered Operation when GTADTR Buffer Transfer is Inhibited

If the GTADTRm (m = A, B) register is made to operate as a double buffer in triangle-wave mode (the GTBER.ADTDm (m = A, B) bit is 1) and the timing of buffer transfer is set for both troughs and crests (the GTBER.ADTTm (m= A, B)[1:0] bits are 11b), setting the GTBER.DBRTEADm (m = A, B) bit to 1 repeats transfer from the intermediate buffer to the GTADTRm (m = A, B) register on a cyclic basis even while buffer transfer is inhibited by the setting of the GTBER.BD[2] bit or by the buffer transfer extended skipping function.

The A/D converter start request timing value for the first half of the cycle is stored in temporary register ADm (m=A, B) as the intermediate buffer for the GTADTBRm (m = A, B) register and the A/D converter start request timing value for the second half of the cycle is stored in the GTADTBRm (m = A, B) register, respectively, for A/D converter start request timing values during repeated operation, and the given values are alternately transferred to the GTADTRm (m = A, B) register. The first and second halves of the cycle is determined by the setting of the GTADTRm double-buffer repeat operation period select bit GTBER.DBRTSADm (m = A, B). **Table 5.7-50** lists the types of buffer transfer of the GTADTR register during counting in triangle-wave mode.

While counting is stopped, set the A/D converter start request timing value for the first half of the cycle following the start of counting in temporary register ADm (m = A, B) beforehand. When the setting of the GTBER.DBRTEADm (m = A, B) bit is 1b, values written by the CPU to the GTADTRm (m = A, B) registers are reflected as the values of temporary registers ADm (m = A, B).

If the start of counting is at the second half of the cycle, set the value for the second half of the cycle in the GTADTRm (m = A, B) register beforehand when counting is stopped. If the start of counting is at the second half of the cycle and in the buffer transfer inhibited period, set the same value as the GTADTBRm (m = A, B) register in the GTADTRm (m = A, B) register. When setting the value for the second half of the cycle in the GTADTRm (m = A, B) register when counting is stopped, set the value for the first half of the cycle in temporary register ADm (m = A, B), and then return the value of the GTBER.DBRTEADm (m = A, B) bit to 0b once before writing by the CPU. After writing, set the GTBER.DBRTEADm (m = A, B) bit to 1b before counting starts.

Table 5.7-50 GTADTR Buffer Operation in Triangle-Wave Mode (during Counting)

GTBER. DBRTECm Bit	GTBER. DBRTSCm Bit	Buffer Transfer	Transfer Timing			
			GTCCRy register ↓ GTCCRx register	GTCCRx register ↓ GTCCRm register	GTCCRy register ↓ Temporary register x	Temporary register x ↓ GTCCRm register
0b	—	Transfer-enabled period	Both trough and crest	Both trough and crest	No transfer	No transfer
		Transfer-inhibited period	No transfer	No transfer	No transfer	No transfer
1b	0b (trough cycle)	Transfer-enabled period	Both trough and crest	Both trough and crest	Crest	No transfer
		Transfer-inhibited period	No transfer	Crest	No transfer	Trough
	1b (crest cycle)	Transfer-enabled period	Both trough and crest	Both trough and crest	Trough	No transfer
		Transfer-inhibited period	No transfer	Trough	No transfer	Crest

Note: m: A, B

5.7.8.3 GTIOCnm Pin Output Negate Control (n = 0 to 15, m = A, B)

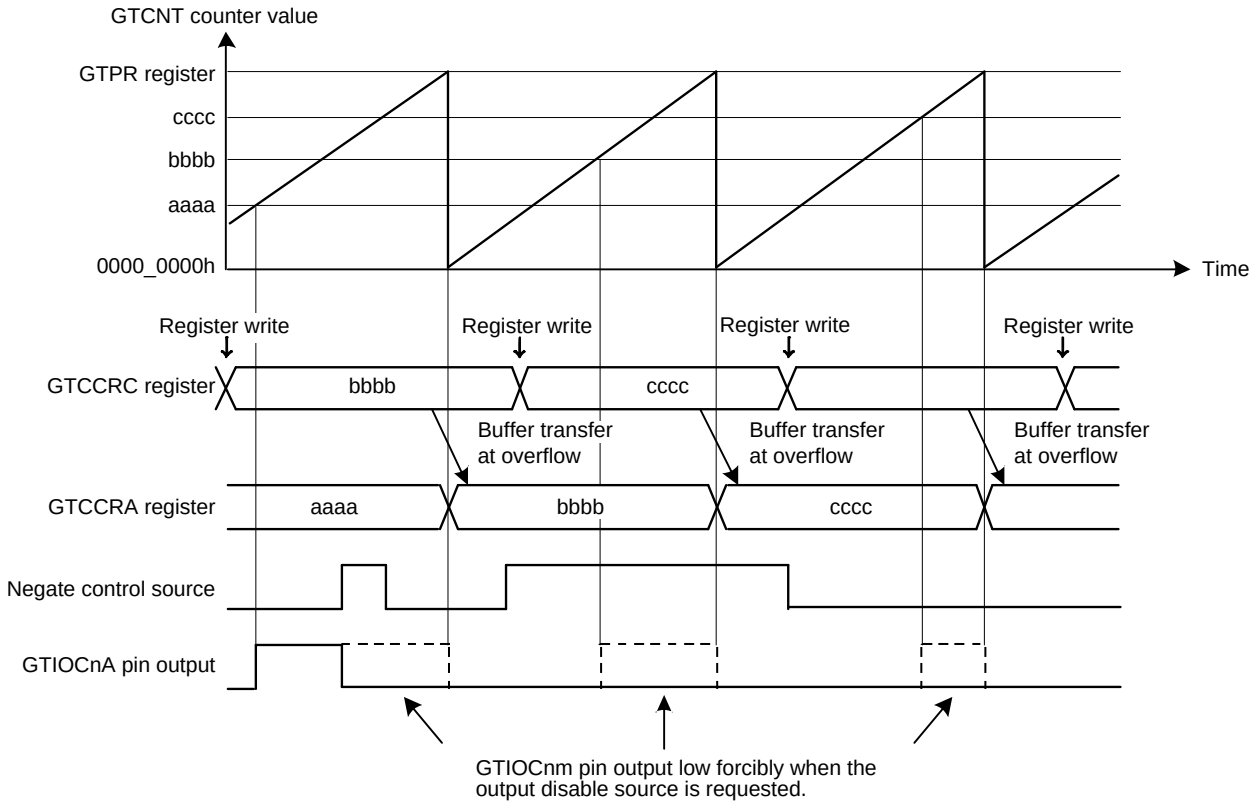
For protection from system failure, the output disable control that changes the GTIOCnm pin output value forcibly is provided for GTIOCnm pin output by the request of output disable from POEG. Output protection is required when the same output level being on the GTIOCnA and GTIOCnB pins is detected. GPT detects this condition and generates output disable requests to POEG according to the setting of the output disable request permission bits, such as GTINTAD.GRPABH, GTINTAD.GRPABL. After the POEG performs the logical OR of the output disable request from each channel and the output disable request from the external input, the POEG generates output disable requests to GPT.

One output disable signal (representing the shared output disable request signal of the GTIOCnA pin and the GTIOCnB pin) out of 4 output disable requests generated by the POEG is selected by setting GTINTAD.GRP[1:0]. The status of the selected output disable request is monitored by reading the GTST.ODF bit. The output level during output disable is set based on the GTIOR.OADF[1:0] bits for the GTIOCnA pin and the GTIOR.OBDF[1:0] setting for the GTIOCnB pin.

The change to the output disable state is performed asynchronously by generating the output disable request from the POEG. The release of the output disable state is performed at end of cycle by terminating the output disable request. It is after 3 PCLKD at shortest when the output disable condition is released after the output disable request becomes no longer satisfied. To reliably control output disabling, clear the flag of POEG for which the condition for the request to disable the output is no longer satisfied after 4 cycles of PCLKD.

When event count is performed or when the output disable state should be released immediately without waiting for end of cycle, GTIOR.OADF[1:0] should be set to 00b (for GTIOCnA pin) or GTIOR.OBDF[1:0] should be set to 00b (for the GTIOCnB pin).

Figure 5.7-77 shows an example of the GTIOCnm pin output disable control operation. (n = 0 to 15, m = A, B)



Note: The base addresses of the registers are the same.

Figure 5.7-77 Example of GTIOCnm Pin Output Disable Control Operation in Saw-Wave Up-Counting, Buffer Operation, Active Level 1, High-level Output at GTCCRA Compare Match, Low-level Output at Cycle End, and Low-level Output at Output Disable (n = 0 to 15, m = A, B)

5.7.8.4 GTIOCnm Protection Function for GTIOCnm Pin Output (n = 0 to 15; m = A, B)

In case an incorrect value (0000_0000h or a value greater than or equal to the GTPR register value) is set in the GTCCRA register, the output protection function (inhibition) for the GTIOCnm pin output operates when the automatic dead time setting is made (the GTDTCR.TDE bit = 1) in triangle-wave PWM mode.

The operating state of the output protection function can be checked by reading the GTSOS.SOS[1:0] bits.

Figure 5.7-78 shows the state transitions of the output protection function.

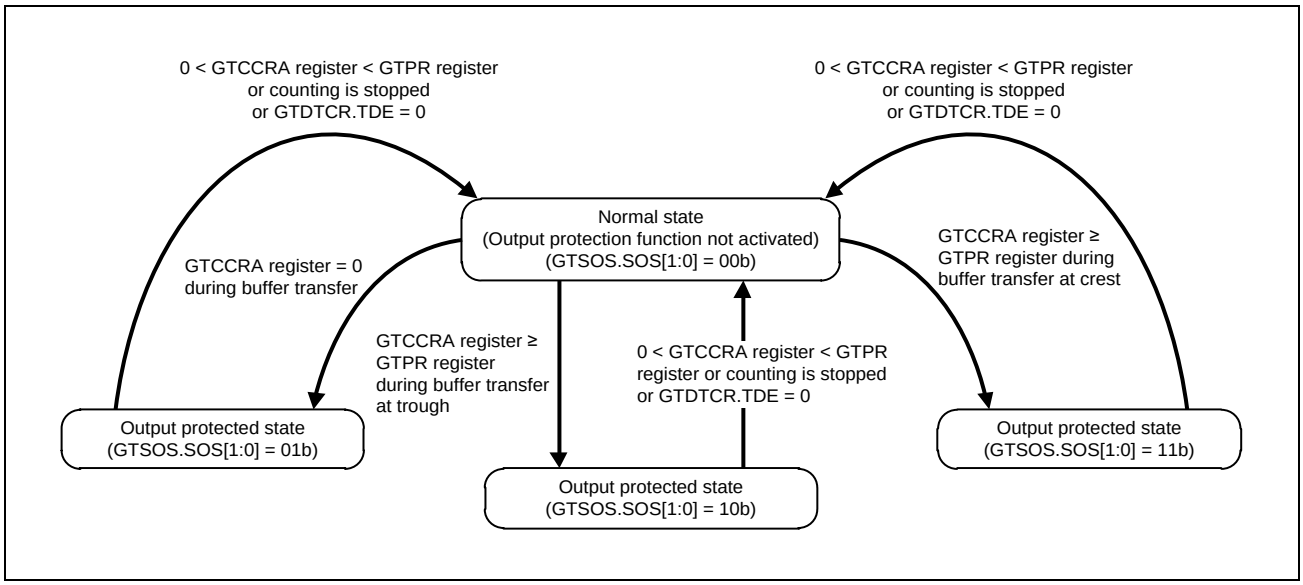
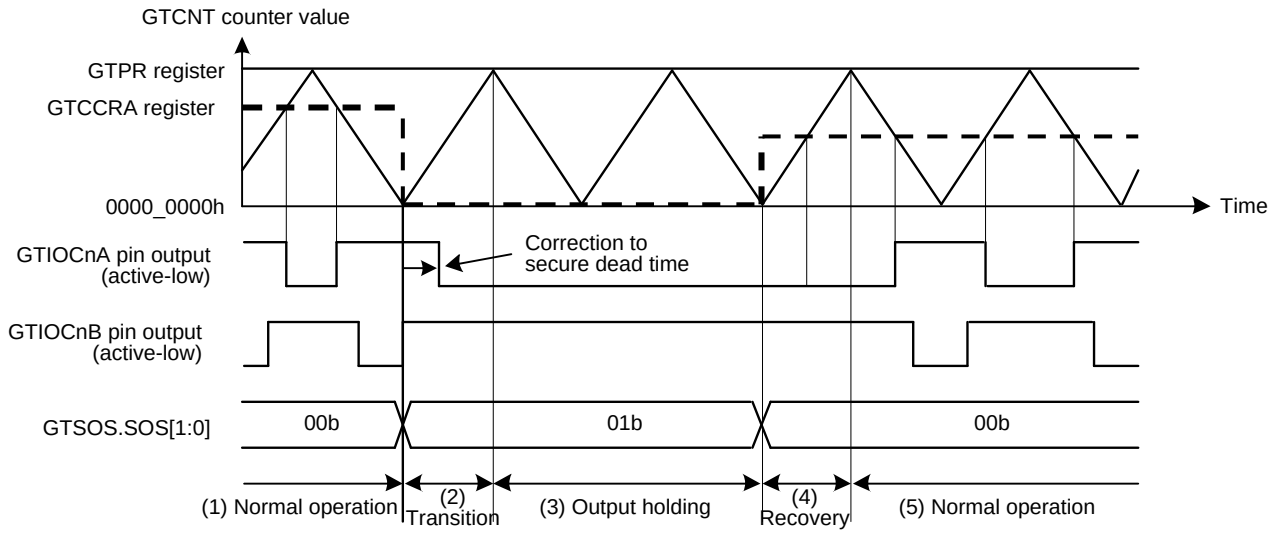


Figure 5.7-78 Output Protection Function

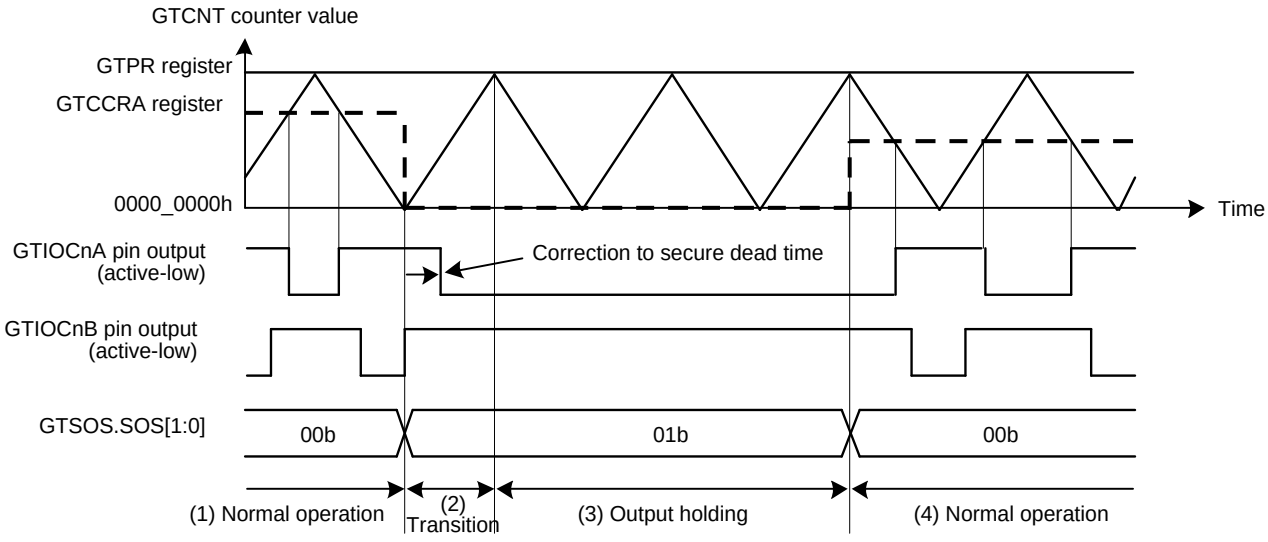
(1) Output Protection Function when the GTCCRA Register Value is 0000_0000h during Buffer Transfer

Figure 5.7-79 and **Figure 5.7-81** show examples of output protection function operation when the GTCCRA register value is 0000_0000h during buffer transfer at troughs, and **Figure 5.7-82** and **Figure 5.7-83** show examples when the GTCCRA register value is 0000_0000h during buffer transfer at crests.



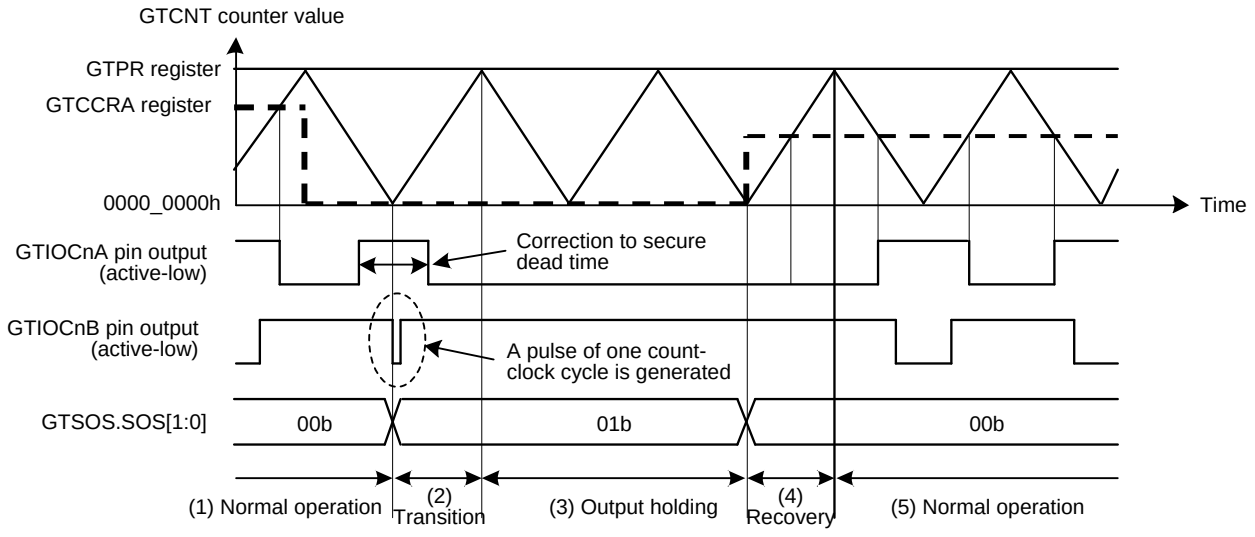
(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Transition period: When GTCCRA = 0 is detected at the trough, the output protection status changes to GTSOS.SOS[1:0] = 01b, and toggle output is continued up to the next crest.
(3) Output holding period: The output is held while GTCCRA = 0.
(4) Recovery period: When 0 < GTCCRA < GTPR is detected at the trough, the output protection status changes to GTSOS.SOS[1:0] = 00b, and holding output is maintained up to the next crest.
(5) Normal operation period: Toggle output due to a compare match is performed normally.

Figure 5.7-79 Example of Output Protection Function Operation when the GTCCRA Register Value is 0000_0000h during Buffer Transfer at Troughs (Restored to 0 < GTCCRA Register < GTPR Register during Buffer Transfer at Troughs, Active Level: Low) (n = 0 to 15)



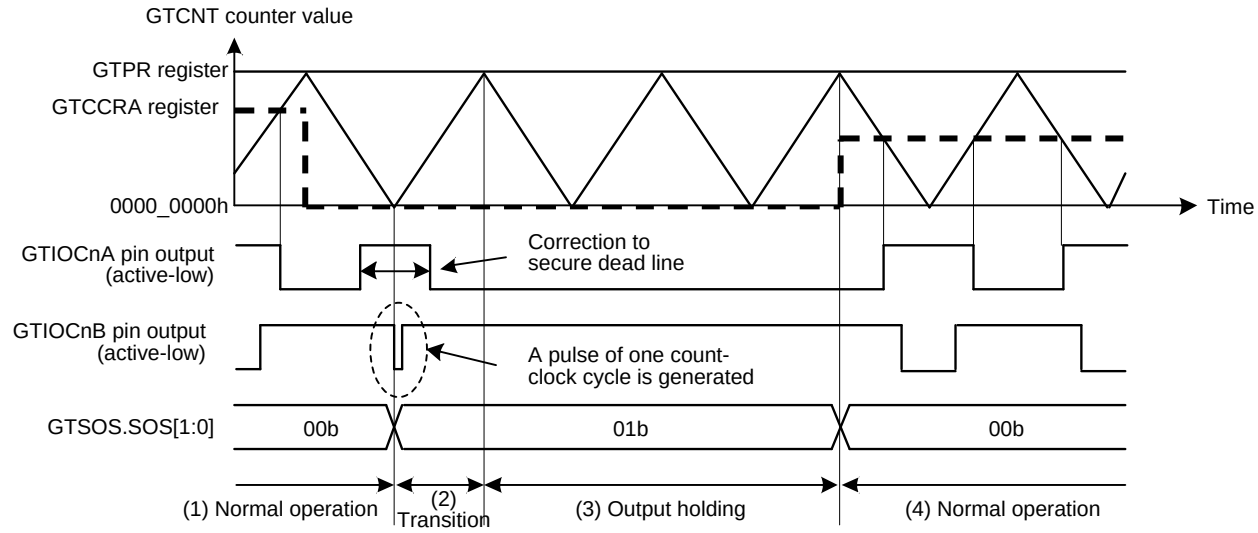
(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Transition period: When GTCCRA = 0 is detected at the trough, the output protection status changes to GTSOS.SOS[1:0] = 01b, and toggle output is continued up to the next crest.
(3) Output holding period: The output is held while GTCCRA = 0.
(4) Normal operation period: When 0 < GTCCRA < GTPR is detected at the crest, the output protection status changes to GTSOS.SOS[1:0] = 00b, and toggle output due to a compare match is performed normally.

Figure 5.7-80 Example of Output Protection Function Operation when the GTCCRA Register Value is 0000_0000h during Buffer Transfer at Troughs (Restored to 0 < GTCCRA Register < GTPR Register during Buffer Transfer at Crests, Active Level: Low) (n = 0 to 15)



(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Transition period: When GTCCRA = 0 is detected at the trough, the output protection status changes to GTSOS.SOS[1:0] = 01b, and toggle output is continued up to the next crest.
(3) Output holding period: The output is held while GTCCRA = 0.
(4) Recovery period: When 0 < GTCCRA < GTPR is detected at the trough, the output protection status changes to GTSOS.SOS[1:0] = 00b, and holding output is maintained up to the next crest.
(5) Normal operation period: Toggle output due to a compare match is performed normally.

Figure 5.7-81 Example of Output Protection Function Operation when the GTCCRA Register Value is 0000_0000h during Buffer Transfer at Crests (Restored to 0 < GTCCRA Register < GTPR Register during Buffer Transfer at Troughs, Active Level: Low) (n = 0 to 15)

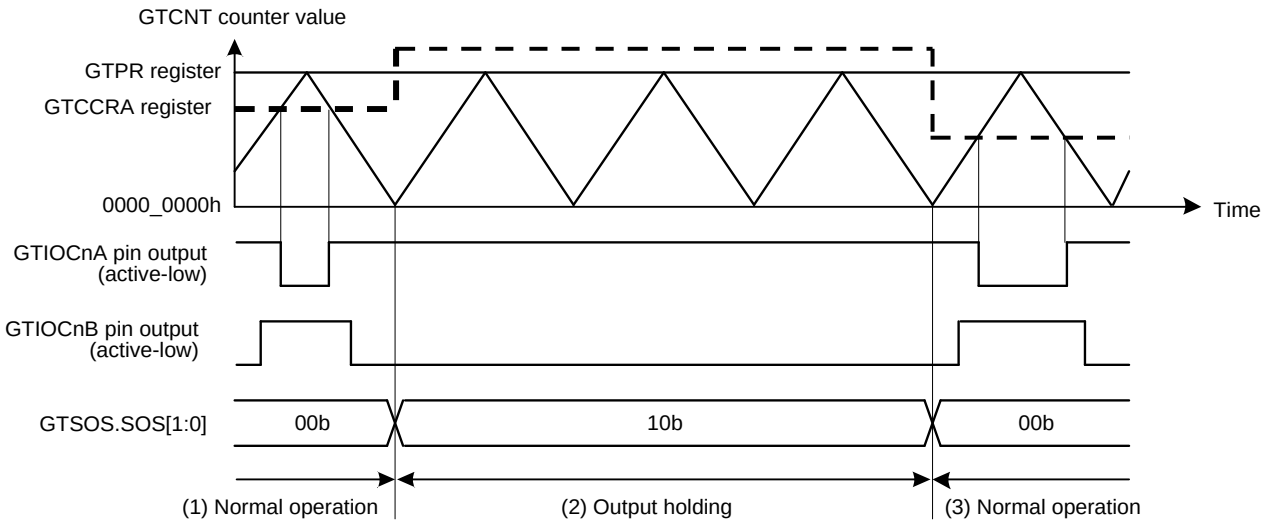


(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Transition period: When GTCCRA = 0 is detected at the trough, the output protection status changes to GTSOS.SOS[1:0] = 01b, and toggle output is continued up to the next crest.
(3) Output holding period: The output is held while GTCCRA = 0.
(4) Recovery period: When 0 < GTCCRA < GTPR is detected at the crest, the output protection status changes to GTSOS.SOS[1:0] = 00b, and toggle output due to a compare match is performed normally.

Figure 5.7-82 Example of Output Protection Function Operation when the GTCCRA Register Value is 0000_0000h during Buffer Transfer at Crests (Restored to 0 < GTCCRA Register < GTPR Register during Buffer Transfer at Crests, Active Level: Low) (n = 0 to 15)

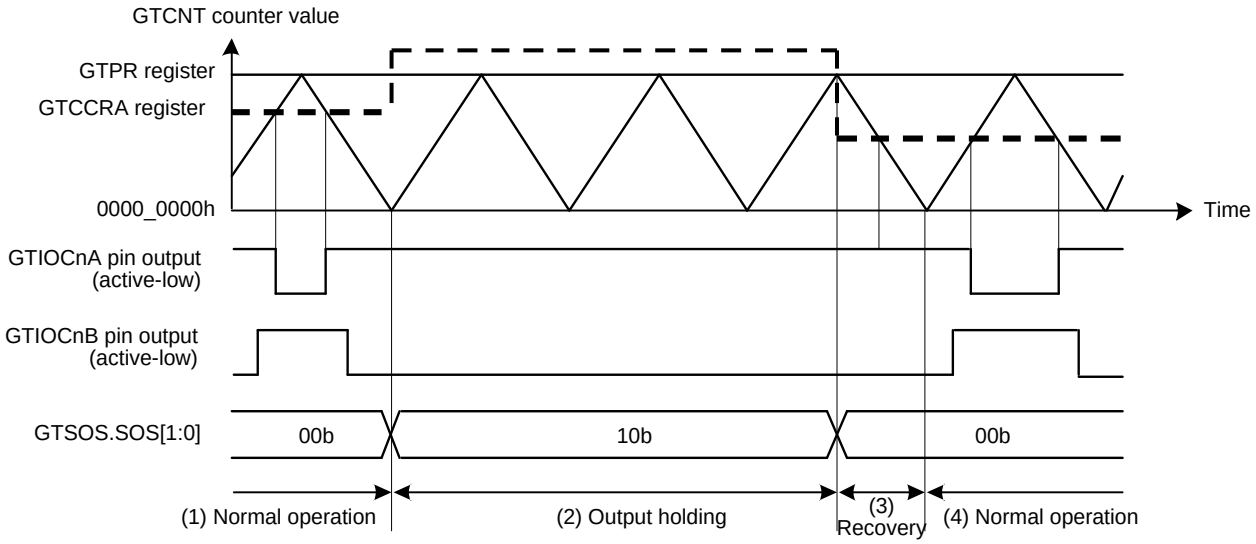
(2) Output Protection Function when GTCCRA Register ≥ GTPR Register is Set during Buffer Transfer at Troughs

Figure 5.7-83 and **Figure 5.7-84** show examples of output protection function operation when GTCCRA register ≥ GTPR register is set during buffer transfer at troughs.



(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Output holding period: When GTCCRA ≥ GTPR is detected at the trough, the output protection status changes to GTSOS[1:0] = 10b, and the output is held while GTCCRA ≥ GTPR.
(3) Normal operation period: When 0 < GTCCRA < GTPR is detected at the trough, the output protection status changes to GTSOS[1:0] = 00b, ant toggle output due to a compare match is performed normally.

Figure 5.7-83 Example of Output Protection Function Operation when GTCCRA Register ≥ GTPR Register is Set during Buffer Transfer at Troughs (Restored to 0 < GTCCRA Register < GTPR Register during Buffer Transfer at Troughs, Active Level: Low) (n = 0 to 15)

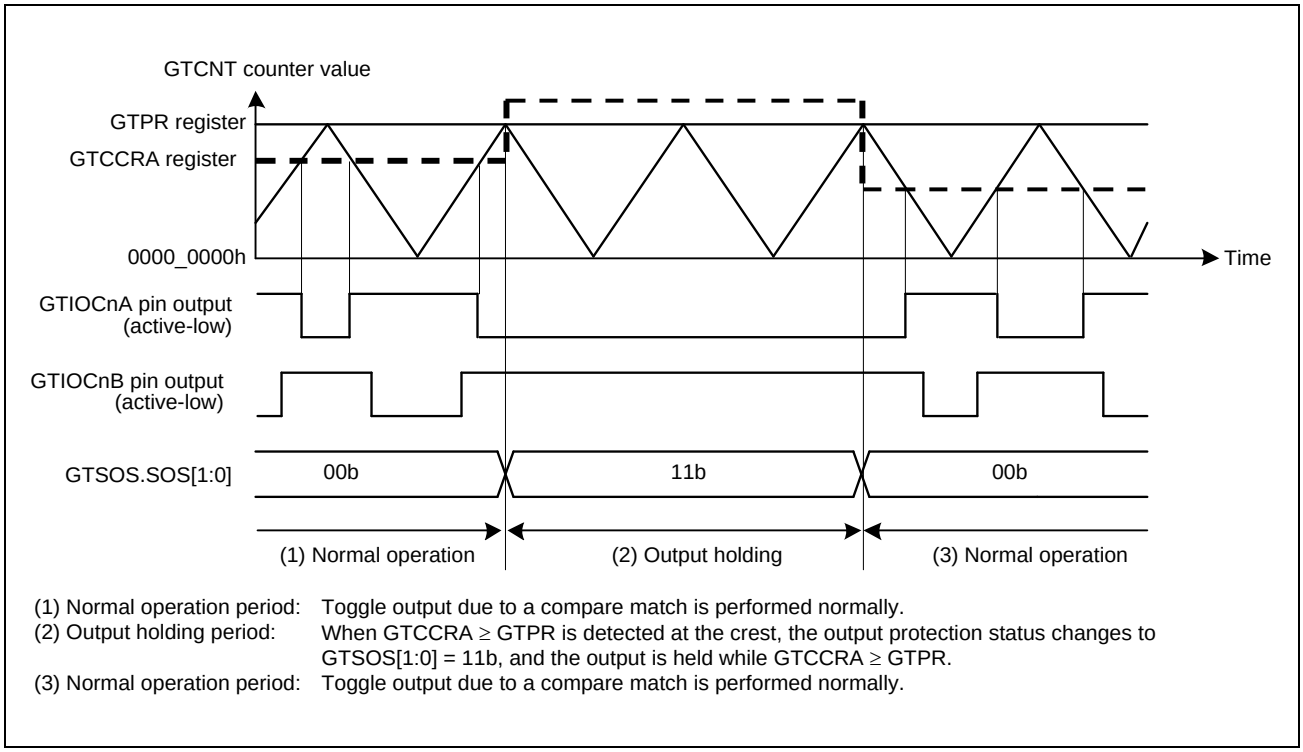


(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Output holding period: When GTCCRA ≥ GTPR is detected at the trough, the output protection status changes to GTSOS[1:0] = 10b, and the output is held while GTCCRA ≥ GTPR.
(3) Recovery period: When 0 < GTCCRA < GTPR is detected at the crest, the output protection status changes to GTSOS.SOS[1:0] = 00b, and holding output is maintained up to the next trough.
(4) Normal operation period: When 0 < GTCCRA register < GTPR register is detected in the crest, the output protection status changes to GTSOS[1:0] = 00b, and toggle output due to a compare match is performed normally.

Figure 5.7-84 Example of Output Protection Function Operation when GTCCRA Register ≥ GTPR Register is Set during Buffer Transfer at Troughs (Restored to 0 < GTCCRA Register < GTPR Register during Buffer Transfer at Crests, Active Level: Low) (n = 0 to 15)

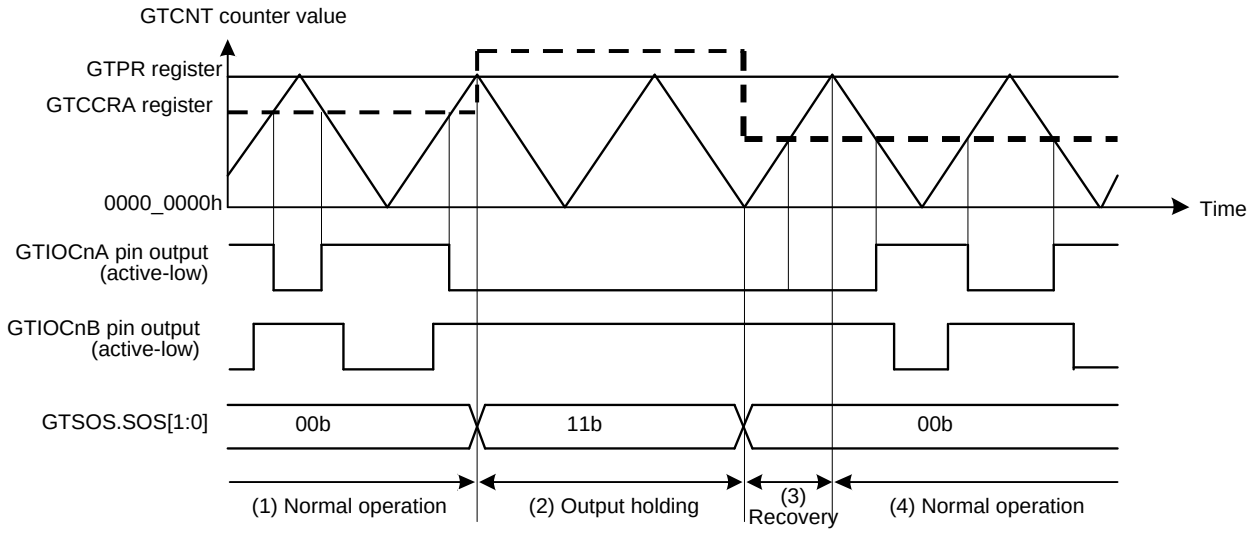
(3) Output Protection Function when GTCCRA Register ≥ GTPR Register is Set during Buffer Transfer at Crests

Figure 5.7-85 and **Figure 5.7-86** show examples of output protection function operation when GTCCRA register ≥ GTPR register is set during buffer transfer at crests.



(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Output holding period: When GTCCRA ≥ GTPR is detected at the crest, the output protection status changes to GTSOS[1:0] = 11b, and the output is held while GTCCRA ≥ GTPR.
(3) Normal operation period: Toggle output due to a compare match is performed normally.

Figure 5.7-85 Example of Output Protection Function Operation when GTCCRA Register ≥ GTPR Register is Set during Buffer Transfer at Crests (Restored to 0 < GTCCRA Register < GTPR Register during Buffer Transfer at Crests, Active Level: Low) (n = 0 to 15)



(1) Normal operation period: Toggle output due to a compare match is performed normally.
(2) Output holding period: When GTCCRA ≥ GTPR Is detected at the crest, the output protection status changes to GTSOS[1:0] = 11b, and the output is held while GTCCRA ≥ GTPR.
(3) Recovery period: When 0 < GTCCRA < GTPR is detected at the trough, the output protection status changes to GTSOS.SOS[1:0] = 00b, and holding output is maintained up to the next crest.
(4) Normal operation period: Toggle output due to a compare match is performed normally.

Figure 5.7-86 Example of Output Protection Function Operation when GTCCRA Register ≥ GTPR Register is Set during Buffer Transfer at Crests (Restored to 0 < GTCCRA Register < GTPR Register during Buffer Transfer at Troughs, Active Level: Low) (n = 0 to 15)

(4) Note on Output Protection Function

The output protection function operates in such a way that either of the positive- and negative-phase outputs becomes non-active even if an incorrect value (0000_0000h or a value greater than or equal to the GTPR register value) is set in the GTCCRA register during counting. However, the output protection function does not operate normally if the following condition is not satisfied.

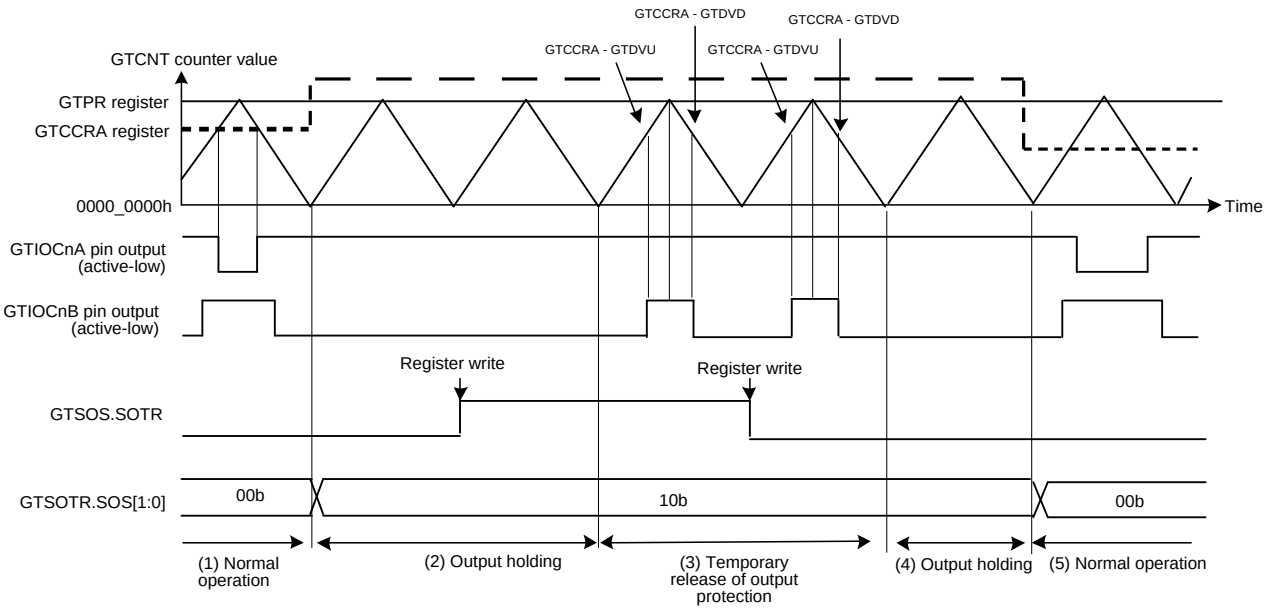
- When the GTCCRA register value at the start of counting is greater than or equal to 0000_0001h and less than the GTPR register value

(5) Temporary Release of Output Protection Function

When the GTSOS.SOS[1:0] bits = 10b (in the protected state indicating GTCCRA register ≥ GTPR register during transfer at troughs), the protected state of the GTIOCnB pin output can be temporarily released by setting the GTSOTR.SOTR bit to 1b. The GTSOS.SOS[1:0] bits retain 10b even if the output protection function has been released.

When the SOTR bit is set to 0b, the GTIOCnB pin output protection can be resumed.

Figure 5.7-87 shows an operation example of temporary release of the output protection function when the setting of the GTCCRA register ≥ GTPR register during buffer transfer at troughs.



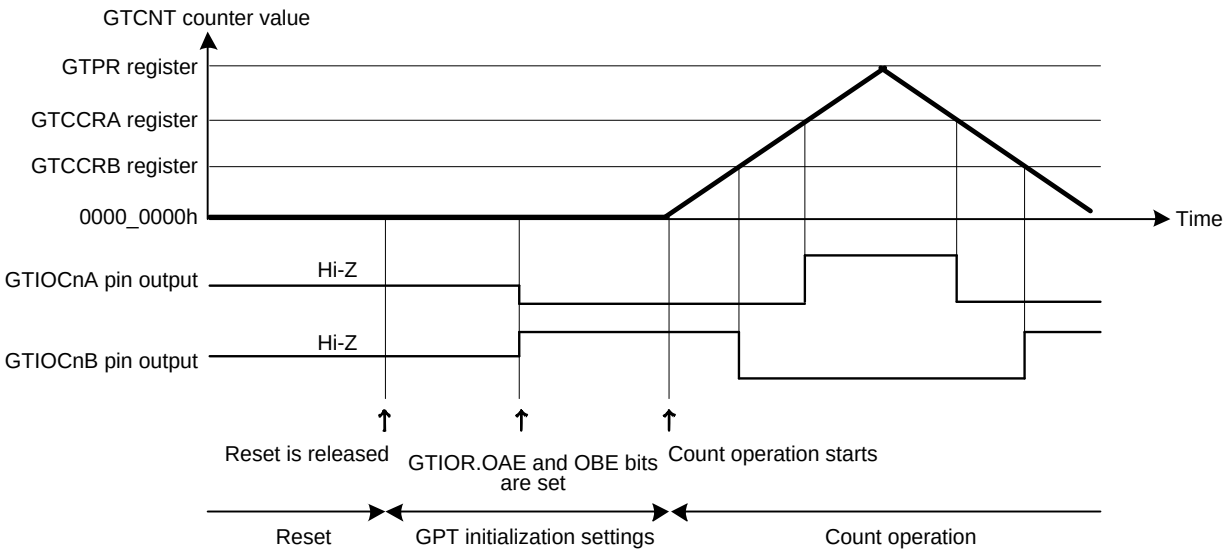
(1) Normal operation period:	Toggle output due to compare match is performed normally.
(2) Output holding period:	When GTCCRA register ≥ GTPR register is detected in the trough, the output protection status changes to GTSOS[1:0] = 10b, and the output is held while GTCCRA register ≥ GTPR register. Even when the GTSOTR.SOTR bit is set to 1 during the output holding period, the GTIOCnB pin output is held until the first trough after the setting.
(3) Temporary release of output protection period	After GTSOTR.SOTR bit is set to 1, the GTIOCnB pin is performed normally from the first trough after the setting. Even when the GTSOTR.SOTR bit is set to 0 during the output protection releasing period, the GTIOCnB pin output is held until the first trough after the setting.
(4) Output holding period:	The output is held until 0 < GTCCRA register < GTPR register is detected in the trough.
(5) Normal operation period:	When 0 < GTCCRA register < GTPR register is detected in the trough, the output protection status changes to GTSOS[1:0] = 00b, and toggle output due to a compare match is performed normally.

Figure 5.7-87 Operation Example of Temporary Release of Output Protection Function when GTCCRA Register ≥ GTPR Register is Set during Buffer Transfer at Troughs (Restored to 0 < GTCCRA Register < GTPR Register during Buffer Transfer at Troughs, Active Level: Low) (n = 0 to 15)

5.7.9 Initialization Method of Output Pins

5.7.9.1 Pin Settings after Reset

The GPT registers are initialized at a reset. Start counting after selecting the port pin function with the PmnPFS register, setting GTIOR.OAE and GTIOR.OBE bits, and output of the GPT function to external pins.



[Setting examples]
GTIOR.GTIOA[4:0] bits: Initial low output, output retained at cycle end, output toggled at compare match
GTIOR.GTIOB[4:0] bits: Initial high output, output retained at cycle end, output toggled at compare match

Note: The base addresses of the registers are the same.

Figure 5.7-88 Example of Pin Settings after Reset

5.7.9.2 Pin Initialization Due to Error during Operation

If an error occurs during GPT operation, the following four types of pin control can be performed before pin initialization:

- Set the OAHLD and OBHLD bits in GTIOR to 1b and retain the outputs at count stop
- Set the OAHLD and OBHLD bits in GTIOR to 0b, specify arbitrary output values at OADFLT and OBDFLT in GTIOR, and output the arbitrary values at count stop
- Set the pin to output an arbitrary value as a general output port by setting the PDR, PODR registers and PmnPFS.PMR bit of the I/O port in advance. Set the OAE and OBE bits in GTIOR to 0b, and the control bit corresponding to the pin in the PMR to 0 to allow arbitrary values to be output from the pin set as a general output port when an error occurs.
- Drive the output to a high impedance state using the POEG function.

If the automatic dead time setting is made, clear the GTDTCR.TDE bit to 0b after counting stops. When counting stops, only the values of registers that are changed by a GPT external source change. If counting is resumed, operation continues from where it stopped. If counting is stopped, the registers must be initialized before counting starts.

5.7.10 Usage Notes

5.7.10.1 Settings of the GTCCRn Register during Compare Match Operation (n = A to F)

(1) When Automatic Dead Time Setting is Made in Triangle-Wave PWM Mode

The GTCCRA register must satisfy the following conditions:

- GTCCRA register > GTDVU register
- GTCCRA register > GTDVD register
- GTCCRA register < GTPR register

When the GTCCRA register is set to 0000_0000h or a value greater than or equal to the GTPR register value during counting, the output protection function is activated.

However, if the following conditions are not satisfied, the output protection does not operate normally.

- When the GTCCRA register value at the start of counting operation is 0000_0001h or greater, and less than the setting value of the GTPR register

For details, see **5.7.8.4 GTIOCnm Protection Function for GTIOCnm Pin Output (n = 0 to 15; m = A, B)**.

(2) When Automatic Dead Time Setting is not Made in Triangle-Wave PWM Mode

Set a value greater than 0000_0000h and less than the setting value of the GTPR register in the GTCCRA register. When 0000_0000h or the same value as that of the GTPR register is set in the GTCCRA register, compare match is generated in one cycle only when [GTCCRA register = 0000_0000h] or [GTCCRA register = GTPR register] is met. Furthermore, a value exceeding the setting value of the GTPR register is set in the GTCCRA register, compare match does not occur.

Similarly, set a value greater than 0000_0000h, and less than the setting value of the GTPR register in the GTCCRB register. When 0000_0000h or the same value as that of the GTPR register is set in the GTCCRB register, compare match is generated in one cycle only when [GTCCRB register = 0000_0000h] or [GTCCRB register = GTPR register] is met.

Furthermore, a value exceeding the setting value of the GTPR register is set in the GTCCRB register, compare match does not occur.

(3) When Automatic Dead Time Setting is Made in Saw-Wave One-Shot Pulse Mode

The GTCCRC and GTCCRD registers must be set to satisfy the following restrictions. If the restrictions are not satisfied, correct output waveforms with secured dead time may not be obtained.

- In up-counting:
  GTCCRC register < GTCCRD register
  GTCCRC register > GTDVU register
  GTCCRD register < GTPR register − GTDVD register
- In down-counting:
  GTCCRC register > GTCCRD register
  GTCCRC register < GTPR register − GTDVU register
  GTCCRD register > GTDVD register

(4) When Automatic Dead Time Setting is not Made in Saw-Wave One-Shot Pulse Mode

The GTCCRC and GTCCRD registers must be set to satisfy the following restrictions. If the restrictions are not satisfied, two compare matches do not occur and pulse output cannot be performed.

- In up-counting: 0 < GTCCRC register < GTCCRD register < GTPR register
- In down-counting: GTPR register > GTCCRC register > GTCCRD register > 0

Similarly, GTCCRE and GTCCRF registers must be set to satisfy the following restrictions. If the restrictions are not satisfied, two compare matches do not occur and pulse output cannot be performed.

- In up-counting: 0 < GTCCRE register < GTCCRF register < GTPR register
- In down-counting: GTPR register > GTCCRE register > GTCCRF register > 0

(5) In Saw-Wave PWM Mode

The GTCCRA register must be set with the range of 0000_0000h < GTCCRA register < GTPR register. If GTCCRA register = 0000_0000h or GTCCRA register = GTPR register is set, a compare match occurs within the cycle only when GTCCRA register = 0000_0000h or GTCCRA register = GTPR register is satisfied. If GTCCRA register > GTPR register is set, no compare match occurs.

5.7.10.2 Setting Range for GTCNT Counter

The GTCNT counter register must be set with the range of $0 \leq GTCNT \leq GTPR$.

5.7.10.3 Starting and Stopping the GTCNT Counter

The control timing of starting and stopping the GTCNT counter by the GTCR.CST bit synchronizes the count clock that is selected in GTCR.TPCS[3:0]. When GTCR.CST is updated, the GTCNT counter starts/stops after a count clock that is selected in GTCR.TPCS[3:0]. Therefore, an event generated before the GTCNT counter actually starts is ignored, resulting in situations in which an event is accepted or an interrupt occurs after GTCR.CST is set to 0b.

5.7.10.4 Priority Order of Each Event

(1) GTCNT Register

Table 5.7-51 shows a priority order of events updating the GTCNT register.

Table 5.7-51 Priority Order of Sources Updating GTCNT

Source Updating GTCNT	Priority Order
Writing by CPU (writing to GTCNT/GTCLR)	High
Clear by hardware sources set in GTCSR	↑
Count up or down by hardware sources set in GTUPSR/GTDNSR	↑
Count operation	Low

If up-counting and down-counting by hardware sources occur at the same time, the GTCNT counter value does not change. When there is a conflict between updating the GTCNT register and reading by the CPU, pre-update data is read.

(2) GTCR.CST Bit

When there is a conflict between starting/stopping by hardware sources set in the GTSSR/GTPSR registers and writing by the CPU (writing to GTCR/GTSTR/GTSTP registers), the writing by CPU has priority over the starting/stopping by hardware sources.

When there is a conflict between starting by hardware sources set in the GTSSR register and stopping by hardware sources set in GTPSR register, the GTCR.CST bit value does not change. When there is a conflict between updating the GTCR.CST bit and reading by the CPU (reading from GTCR/GTSTR/GTSTP registers), pre-update data is read.

(3) GTCCRm Registers (m = A to F)

When there is a conflict between input capture/buffer transfer operation and writing to the GTCCRm registers, the writing to GTCCRm registers has priority over input capture/buffer transfer operation. When there is a conflict between input capture and writing to the counter register by the CPU or updating the counter register by hardware sources, the pre-update counter value is captured. When there is a conflict between updating the GTCCRm registers and reading by the CPU, pre-update data is read.

(4) GTPR Register

When there is a conflict between buffer transfer operation and writing to the GTPR register, writing to GTPR register has priority over buffer transfer operation. When there is a conflict between updating GTPR register and reading by the CPU, pre-update data is read.

(5) GTADTRm Register (m = A, B)

In case of contention between a buffer transfer and the GTADTRm register writing, the GTADTRm register writing takes priority over a buffer transfer.

In case of contention between the GTADTRm register updating and the CPU reading, data before updating is read.

(6) GTDVm Register (m = U, D)

In case of contention between a buffer transfer and the GTDVm register writing, the GTDVm register writing takes priority over a buffer transfer.

In case of contention between the GTDVm register updating and the CPU reading, data before updating is read.

SECTION 5 TIMER

5.8 Port Output Enable for GPT (POEG)

5.8.1 Overview

The Port Output Enable (POEG) function can place the General Purpose Timer (GPT) output pins in the output disable state in one of the following ways:

- Input level detection of the GTETRGn (n = A to H) pins
- Output-disable request from the GPT
- Register settings

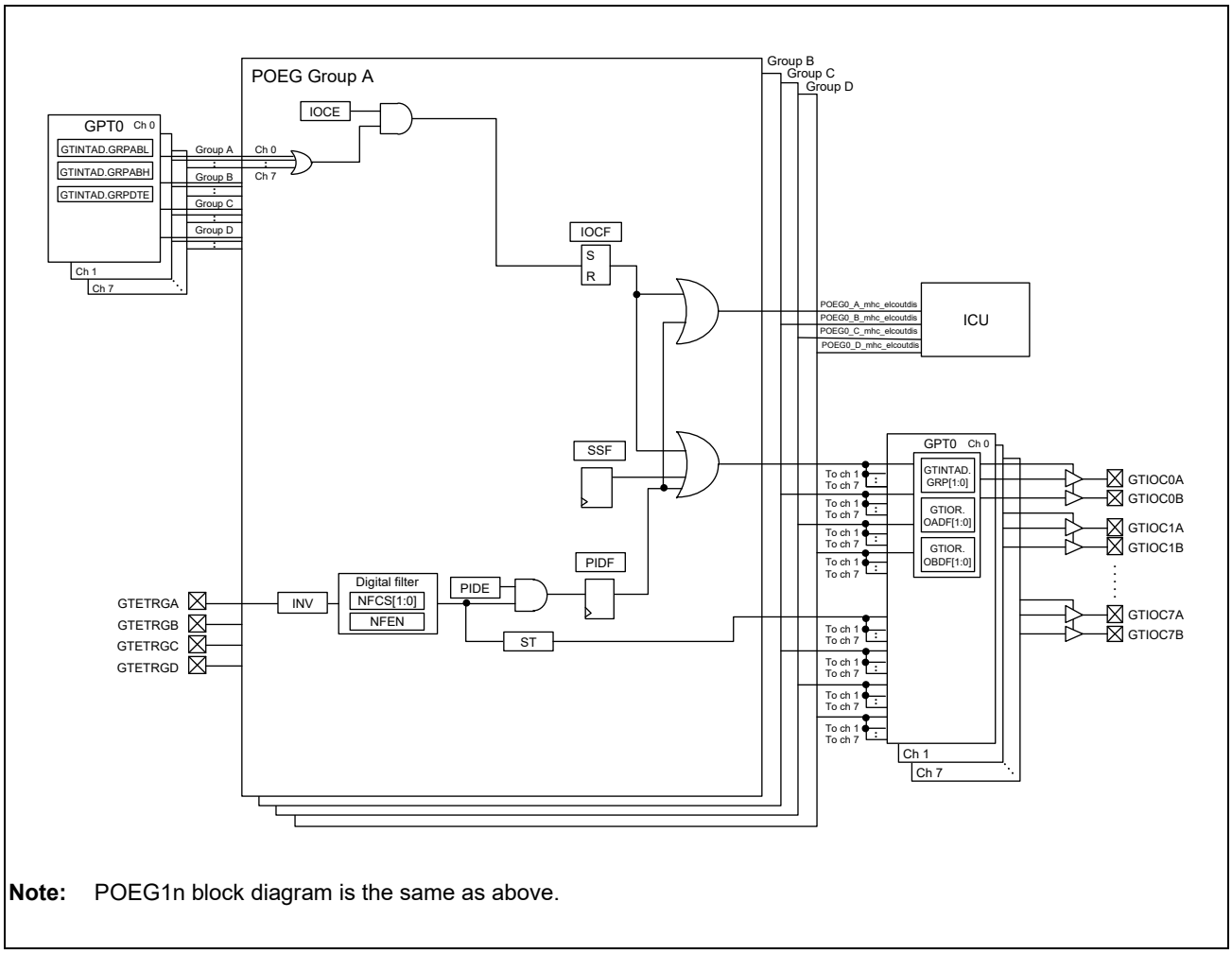
The GTETRGn (n = A to H) pins can be used as GPT external trigger input pins.

Table 5.8-1 lists the POEG specifications, **Figure 5.8-1** shows a block diagram, and **Table 5.8-2** lists the input pins.

Table 5.8-1 POEG Specifications

Parameter	Specifications
Output-disable control through input level detection	• The GPT output pins can be disabled when a GTETRGn rising edge or high level is sampled after polarity and filter selection.
Output-disable request from the GPT	• When the GTIOCmA and GTIOCmB pins are driven to an active level simultaneously, the GPT generates an output-disable request to the POEG. Through reception of these requests, the POEG can control whether the GTIOCmA and GTIOCmB pins are output-disabled.
Output-disable control by software (registers)	• The GPT output pins can be disabled by modifying the register settings.
Interrupt	• Interrupts can be generated by detecting the input level of external trigger input pins (GTETRGn pins). • Interrupts can be generated when all GPT output pins are driven to an active level simultaneously.
External trigger output to the GPT	• The GTETRGn signals can be output to the GPT after polarity and filter selection. (count start, count stop, count clear, up-count, down-count, or input capture function)
Noise filtering	• For input from the GTETRGn pins, PCLKB/1, PCLKB/8, PCLKB/32, or PCLKB/128 can be selected as the noise filtering clock. (Filtering is performed by sampling the input signals three times using the selected clock.)*1 • Positive or negative polarity can be selected for any of the GTETRGn input pins. • Signal state after polarity and filter selection can be monitored.

Note 1. n = A to H
m = 0 to 15
PCLKB is equivalent to POEGk_0_PCLK or POEGk_1_PCLK. (k = A to D)



Note: POEG1n block diagram is the same as above.

Figure 5.8-1 POEG0n Block Diagram (n = A to D)

Table 5.8-2 POEG Input Pins

Pin Name	Input/Output	Description
GTETRGA	Input	GPT0 output pin output-disable request signal or GPT0 external trigger input pin A
GTETRGB	Input	GPT0 output pin output-disable request signal or GPT0 external trigger input pin B
GTETRGC	Input	GPT0 output pin output-disable request signal or GPT0 external trigger input pin C
GTETRGD	Input	GPT0 output pin output-disable request signal or GPT0 external trigger input pin D
GTETRGE	Input	GPT1 output pin output-disable request signal or GPT1 external trigger input pin A
GTETRGF	Input	GPT1 output pin output-disable request signal or GPT1 external trigger input pin B
GTETRGG	Input	GPT1 output pin output-disable request signal or GPT1 external trigger input pin C
GTETRGH	Input	GPT1 output pin output-disable request signal or GPT1 external trigger input pin D

5.8.2 Registers

Table 5.8-3 Address Map

Base Address Name	POEGGn	Base Address
<POEG0A_base>	n = A	0_1300_1C00 (5300_1C00*1, 4300_1C00*2)
<POEG0B_base>	n = B	0_1300_2000 (5300_2000*1, 4300_2000*2)
<POEG0C_base>	n = C	0_1300_2400 (5300_2400*1, 4300_2400*2)
<POEG0D_base>	n = D	0_1300_2800 (5300_2800*1, 4300_2800*2)
<POEG1A_base>	n = E	0_1300_2C00 (5300_2C00*1, 4300_2C00*2)
<POEG1B_base>	n = F	0_1300_3000 (5300_3000*1, 4300_3000*2)
<POEG1C_base>	n = G	0_1300_3400 (5300_3400*1, 4300_3400*2)
<POEG1D_base>	n = H	0_1300_3800 (5300_3800*1, 4300_3800*2)

Note 1. Cortex-M33 address space (non-secure)

Note 2. Cortex-M33 address space (secure)

5.8.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
POEG Group n Setting Register	POEG_POEGGn	0000_0000h	0000h	32

5.8.2.2 Register Description

The prefix (POEG_) of the register names is omitted in this and subsequent sections.

5.8.2.2.1 POEG Group n Setting Register (POEG_POEGGn) (n = A to H)

The POEGGn (n = A to H) registers control the output-disabled state of the GPT pins, interrupts, and the external trigger input to the GPT.

Access Size : 32 bits

Address : <POEGk_base> + 0000h (k = 0A to 0D, 1A to 1D)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NFCS[1:0]		NFEN	INV	-	-	-	-	-	-	-	-	-	-	-	ST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	IOCE	PIDE	SSF	-	IOCF	PIDF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	NFCS[1:0]	0h	RW	Noise Filter Clock Select 00b: Sample GTETRGn pin input level three times every PCLKB 01b: Sample GTETRGn pin input level three times every PCLKB/8 10b: Sample GTETRGn pin input level three times every PCLKB/32 11b: Sample GTETRGn pin input level three times every PCLKB/128
29	NFEN	0h	RW	Noise Filter Enable 0b: Disable noise filtering 1b: Enable noise filtering
28	INV	0h	RW	GTETRGn Input Reverse 0b: Input GTETRGn as-is 1b: Input GTETRGn in reverse
27 to 17	-	All 0	R	Reserved These bits are always read as 0b. The written value will be ignored.
16	ST	0h	R	GTETRGn Input Status Flag 0b: GTETRGn input after filtering was 0 1b: GTETRGn input after filtering was 1
15 to 7	-	All 0	R	Reserved These bits are always read as 0b. The written value will be ignored.
6	-	0h	RW*2	Reserved These bits are read as 0b. The write value should be 0b.
5	IOCE	0h	RW*2	Enable for GPT Output-Disable Request 0b: Disable output-disable requests from GPT 1b: Enable output-disable requests from GPT
4	PIDE	0h	RW*2	Port Input Detection Enable 0b: Disable output-disable requests from the GTETRGn pins 1b: Enable output-disable requests from the GTETRGn pins
3	SSF	0h	RW	Software Stop Flag 0b: No output-disable request from software occurred 1b: Output-disable request from software occurred
2	-	0h	RW*1	Reserved These bits are read as 0b. The write value should be 0b.

Bit	Bit Name	Initial Value	R/W	Description
1	IOCF	0h	RW*1	Detection Flag for GPT Output-Disable Request 0b: No output-disable request from GPT occurred 1b: Output-disable request from GPT occurred
0	PIDF	0h	RW*1	Port Input Detection Flag 0b: No output-disable request from the GTETRGn pin occurred 1b: Output-disable request from the GTETRGn pin occurred

Note 1. Only 0 can be written to clear the flag.

Note 2. This bit can be written only once after a reset.

5.8.3 Output-Disable Control Operation

If any of the following conditions is satisfied, the GTIOCmA and GTIOCmB (m = 0 to 15) pins can be set to output-disabled:

- Input level or edge detection of the GTETRGn pins
  When POEGGn.PIDE is 1, the POEGGn.PIDF flag is set to 1b. (n = A to H)
- Output-disable request from the GPT
  When POEGGn.IOCE is 1, the POEGGn.IOCF flag is set to 1 if the disabled request is enabled by the General Purpose Timer Interrupt Output Setting Register (GTINTAD). The GTINTAD.GRPABH and GTINTAD.GRPABL settings apply to the group selected by the GPT register GTINTAD.GRP[1:0].
- SSF bit setting
  When POEGGn.SSF is set to 1, the PWM output is disabled.

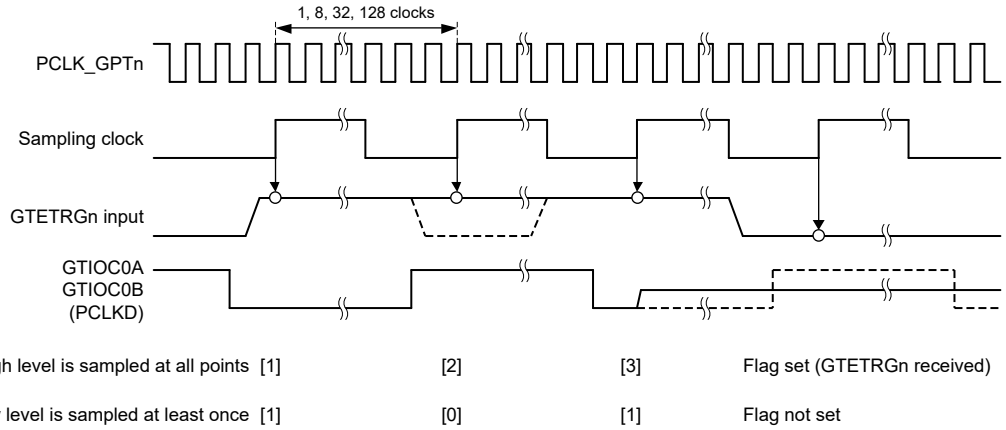
The output-disabled state is controlled in the GPT module. The output-disable for the GTIOCmA and GTIOCmB (m = 0 to 15) pins is set in the GTINTAD.GRP[1:0], General Purpose Timer I/O Control Register (GTIOR).OADF[1:0], and GTIOR.OBDF[1:0] bits in GPTm (m = 0, 1).

5.8.3.1 Pin Input Level Detection Operation

If the input conditions set in POEGGn.PIDE, POEGGn.NFCS[1:0], POEGGn.NFEN, and POEGGn.INV occur on GTETRGn, the GPT output pins are output-disabled.

5.8.3.1.1 Digital Filter

Figure 5.8-2 shows high-level detection by the digital filter. When a high level associated with the POEGGn.INV polarity setting is detected three times consecutively with the sampling clock selected in POEGGn.NFCS[1:0] and POEGGn.NFEN, the detected level is recognized as high, and the GPT output pins are output-disabled. If even one low level is detected during this interval, the detected level is not recognized as high. In addition, in an interval where the sampling clock is not output, changes of the levels on the GTETRGn pins are ignored.



Note: PCLKB is equivalent to POEGk_0_PCLK or POEGk_1_PCLK. (k = A toD)

Note: Each channel output can be set in the GPT setting. Low level sampling can be set in the POEGGn.INV setting.

Figure 5.8-2 Example of Digital Filter Operation

5.8.3.2 Output-Disable Requests from the GPT

For details on the operation, see the description for GTIOCmA and GTIOCmB Pin Output Negate Control in **5.7 General Purpose Timer (GPT)**.

5.8.3.3 Output-Disable Control Using Registers

The GPT output pins can be directly controlled by writing 1 to the Software Stop flag, POEGGn.SSF.

5.8.3.4 Release from Output-Disable

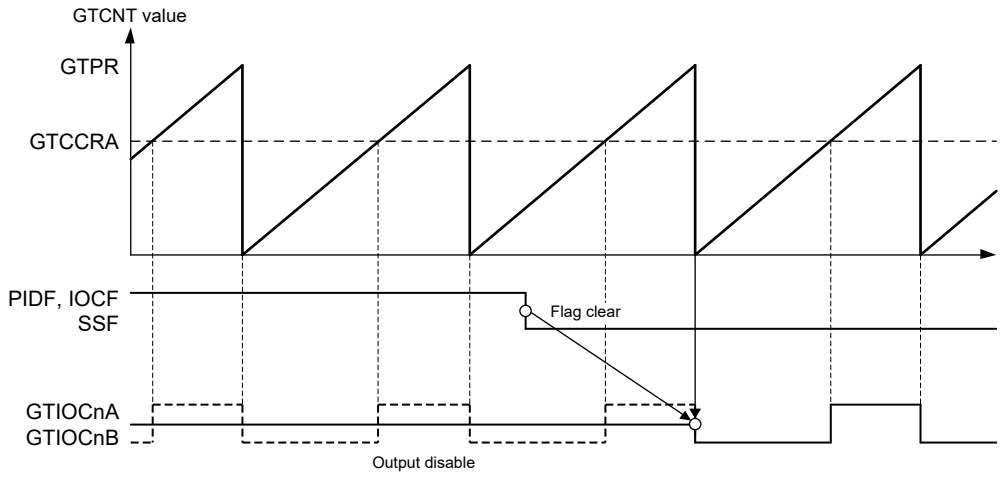
To release the GPT output pins placed in the output-disabled state, either return them to their initial state with a reset or clear all of the following flags:

- POEGGn.PIDF
- POEGGn.IOCF
- POEGGn.SSF

Writing 0 to the POEGGn.PIDF flag is ignored (the flag is not cleared) if the external input pins, GTETRGn are not disabled and the POEGGn.ST bit is not set to 0.

Writing 0 to the POEGGn.IOCF flag is valid (the flag is cleared) only if all of the General Purpose Timer Status Register (GTST).OABHF, GTST.OABLF, and GTST.DTEF flags in the GPT are set to 0.

Figure 5.8-3 shows the release timing for output-disable. The output-disable is released at the beginning of the next count cycle of the GPT after the flag is cleared.



Note: The base addresses of the registers are the same.

Note: n = 0 to 15

Figure 5.8-3 Output-Disable Release Timing for GPT Pin Outputs

5.8.4 Interrupt Sources

The POEG generates an interrupt request when triggered by these sources:

- Output-disable control by the input level detection
- Output-disable request from the GPT

Table 5.8-4 lists the conditions for interrupt requests.

Table 5.8-4 Interrupt Sources and Conditions

Name	Interrupt Sources	CA55 GIC Request	CM33 GIC Request	DMAC Activation
POEG_CH0_mhc_elcoutdis	POEG0 group A interrupt for channels in GPT0	Possible	Possible	Not possible
POEG_CH1_mhc_elcoutdis	POEG0 group B interrupt for channels in GPT0	Possible	Possible	Not possible
POEG_CH2_mhc_elcoutdis	POEG0 group C interrupt for channels in GPT0	Possible	Possible	Not possible
POEG_CH3_mhc_elcoutdis	POEG0 group D interrupt for channels in GPT0	Possible	Possible	Not possible
POEG_CH4_mhc_elcoutdis	POEG1 group A interrupt for channels in GPT1	Possible	Possible	Not possible
POEG_CH5_mhc_elcoutdis	POEG1 group B interrupt for channels in GPT1	Possible	Possible	Not possible
POEG_CH6_mhc_elcoutdis	POEG1 group C interrupt for channels in GPT1	Possible	Possible	Not possible
POEG_CH7_mhc_elcoutdis	POEG1 group D interrupt for channels in GPT1	Possible	Possible	Not possible

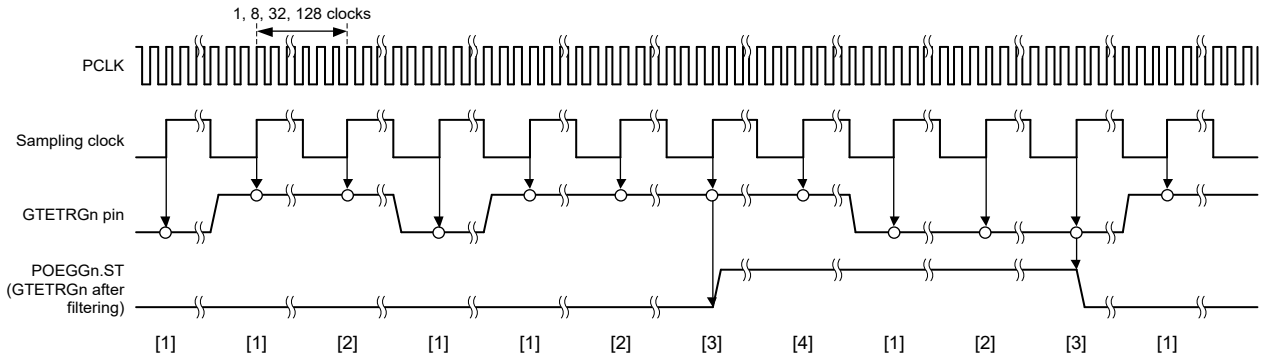
5.8.5 External Trigger Output to the GPT

The POEG outputs signals generated by filtering and level detection of GTETRGn pins input signals as the GPT operation trigger signal for the following:

- Count start
- Count stop
- Count clear
- Up-count
- Down-count
- Input capture

For the POEGGn.INV polarity setting signal, when the same level is input three times continuously with the sampling clock selected in POEGGn.NFCS[1:0] and POEGGn.NFEN, that value is output. Set the control registers the same as for the input level detection operation described in **5.8.3.1 Pin Input Level Detection Operation**. The state after filtering can be monitored in POEGGn.ST.

Figure 5.8-4 shows the output timing of an external trigger to the GPT.



Note: PCLKB is equivalent to POEGk_0_PCLK or POEGk_1_PCLK. (k = A to D)

Note: Each channel output can be set in the GPT settings. Polarity can be reversed in POEGGn.INV.

Figure 5.8-4 Output Timing of External Trigger to the GPT

5.8.6 Usage Notes

5.8.6.1 Transition to Software Standby Mode

When using the POEG, do not invoke Software Standby mode. In this mode, the POEG stops and therefore output disable of the pins cannot be controlled.

5.8.6.2 Specifying Pins Associated with the GPT

The POEG controls output-disable only when a pin is associated with the GPT in the PMC_mn and PFC_mn register settings of the PFC. When the pin is specified as a general I/O pin, the POEG does not perform output-disable control.

SECTION 6 HIGH-SPEED INTERFACE

6.1 High-Speed Interface Overview

This sections describes the High-Speed Interface unit of this LSI. For details, refer to the sections of each unit.

■ SD/MMC Host Interface (SD) (See 6.2)

This LSI equips 3ch of SD.

SD0 supports 8-bit bus and SD1/SD2 supports 4-bit bus MMC Interface.

■ Gigabit Ethernet Interface (GBETH) (See 6.3)

This LSI equips 2ch of GBETH. Conforms to the following standards.

- IEEE 802.3-2008 for Ethernet MAC, Gigabit Media Independent Interface (GMII), and Media Independent Interface (MII)

■ USB3.2 Gen2x1 Interface (USB3) (See 6.4)

This LSI equips 1ch of USB3. Conforms to the following standards (Host mode only).

- Universal Serial Bus 3.2 Specification Revision 1.0 and ECNs approved in Jan 25, 2021.

■ USB2.0 Interface (USB2) (See 6.5)

This LSI equips 2ch of USB2. Conforms to the following standards.

- Universal Serial Bus Specification Revision 2.0

■ PCI Express 3.0 Interface (PCIe) (See 6.6)

This LSI equips 1ch of PCIe. 1 unit × 2 lanes usage is available.

Conforms to the following standards.

- PCI Express Base Specification 4.0 for Gen1 / Gen2 / Gen3

SECTION 6 HIGH-SPEED INTERFACE

6.2 SD/MMC Host Interface (SD)

This section describes the functions of the SD/MMC host interface (SD).

CAUTION

Development of the SD host-related products needs the conclusion of the following agreement.

- "SD Host/Ancillary Product License Agreement (SD HALA)"

6.2.1 Overview

6.2.1.1 Features

- 3 channels
- Channel 0 supports SD and e-MMC.
- Channel 1 supports SD and e-MMC.
- Channel 2 supports SD.
- SD memory/IO card interface (1-bit/4-bit SD bus)
- SD, SDHC, and SDXC SD memory card access supported
- Default, high-speed, UHS-I/SDR50, SDR104, and DDR50 transfer modes supported
- SD clock (SD_CLK) frequency = SDHI_x_IMCLK frequency/2^n (n = 0 to 9) (x = 0 to 2)
- Error check function: CRC7 (for command/response), CRC16 (for data)
- Interrupt request: 2
- Card detect function
- Write protect supported
- MMC interface (1-/4-/8-bit MMC bus)
- e-MMC device access supported
- Backward-compatible, high-speed, HS-DDR, and HS200 transfer modes supported
- High-priority interrupt (HPI) supported

6.2.1.2 Block Diagram

Figure 6.2-1 shows a block diagram of the SD/MMC host interface.

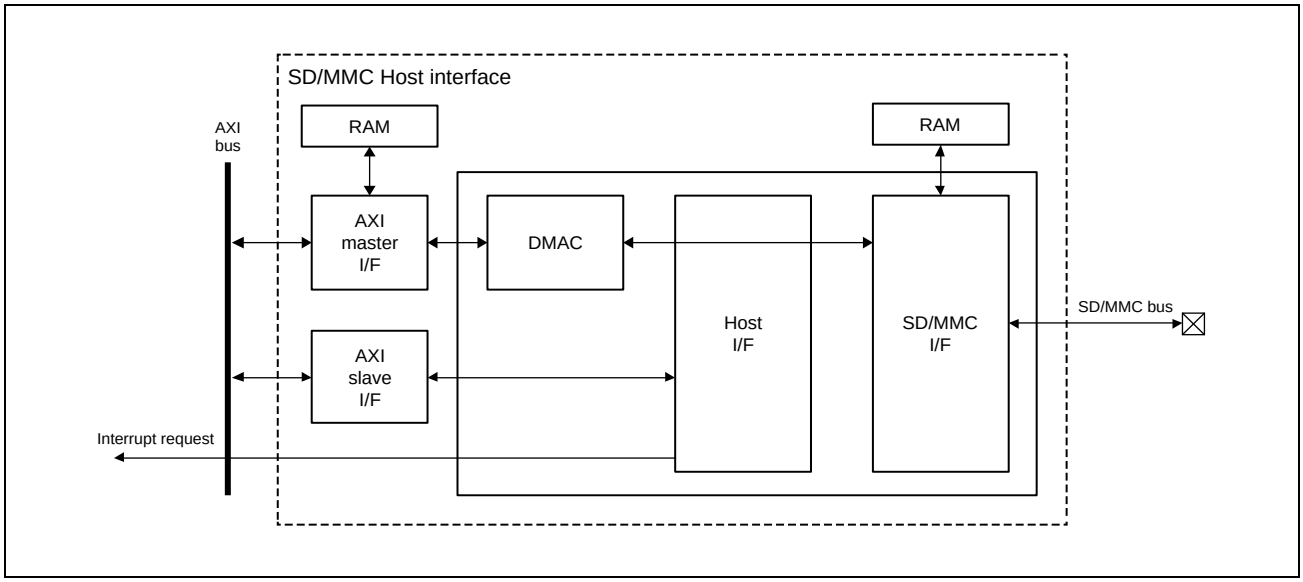


Figure 6.2-1 Block Diagram of SD/MMC Host Interface

6.2.1.3 External Pins

Figure 6.2-1 lists the input and output pins used by the interface. The operating voltage on the pins of the SD/MMC host interface is 3.3 V or 1.8 V.

Table 6.2-1 Pin Configuration

Pin Name	Input/Output	Function
SDxCLK*1	Output	SD/MMC clock output
SDxCMD*1	I/O	SD/MMC command output, response input
SDxDAT0*1	I/O	SD/MMC Data 0 [bit 0]
SDxDAT1*1	I/O	SD/MMC Data 1 [bit 1], SDIO interrupt
SDxDAT2*1	I/O	SD/MMC Data 2 [bit 2], read wait
SDxDAT3*1	I/O	SD/MMC Data 3 [bit 3], card detection
SD0DAT4*1	I/O	SD/MMC Data 4 [bit 4] (SD0 only)
SD0DAT5*1	I/O	SD/MMC Data 5 [bit 5] (SD0 only)
SD0DAT6*1	I/O	SD/MMC Data 6 [bit 6] (SD0 only)
SD0DAT7*1	I/O	SD/MMC Data 7 [bit 7] (SD0 only)
SDxCD*1	Input	SD/MMC card detection*2
SDxWP*1	Input	SD/MMC write protection*2
SDyRSTN*2	Output	SD/MMC reset (SD0 and SD1 only)
SDxPWEN	Output	SD power-enable signals for the power supply IC
SDxIOVS	Output	SD IO voltage level signals for the SD device

Note 1. x (= 0, 1, 2) is the channel number of the SD/MMC host interface. In this manual, those pins are referred to as SDDAT0, SDDAT1, …, SDDAT7.

Note 2. y (= 0, 1) is the channel number of SD/MMC host interface.

6.2.2 SD Registers

The base addresses for each channel are as follows. The SD interface and MMC interface are switched by setting the command type register (SDm_SD_CMD).

Table 6.2-2 Register Base Address

Base Register Name	Unit Name	Base Address
<SD0_base>	SD0	0_15C0_0000h (55C0_0000h*1, 45C0_0000h*2)
<SD1_base>	SD1	0_15C1_0000h (55C1_0000h*1, 45C1_0000h*2)
<SD2_base>	SD2	0_15C2_0000h (55C2_0000h*1, 45C2_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

6.2.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Command Type Register	SDm_SD_CMD	0000_0000_0000_0000h	0000h	16, 32, 64
Reserve	-	-	0008h to 000Fh	-
Command Argument Register	SDm_SD_ARG	0000_0000_0000_0000h	0010h	16, 32, 64
Command Argument Register 1	SDm_SD_ARG1	0000_0000_0000_0000h	0018h	16, 32, 64
Data Stop Register	SDm_SD_STOP	0000_0000_0000_0000h	0020h	16, 32, 64
Block Count Register	SDm_SD_SECCNT	0000_0000_0000_0000h	0028h	16, 32, 64
SD Card Response Registers 10	SDm_SD_RSP10	0000_0000_0000_0000h	0030h	16, 32, 64
SD Card Response Registers 1	SDm_SD_RSP1	0000_0000_0000_0000h	0038h	16, 32, 64
SD Card Response Registers 32	SDm_SD_RSP32	0000_0000_0000_0000h	0040h	16, 32, 64
SD Card Response Registers 3	SDm_SD_RSP3	0000_0000_0000_0000h	0048h	16, 32, 64
SD Card Response Registers 54	SDm_SD_RSP54	0000_0000_0000_0000h	0050h	16, 32, 64
SD Card Response Registers 5	SDm_SD_RSP5	0000_0000_0000_0000h	0058h	16, 32, 64
SD Card Response Registers 76	SDm_SD_RSP76	0000_0000_0000_0000h	0060h	16, 32, 64
SD Card Response Registers 7	SDm_SD_RSP7	0000_0000_0000_0000h	0068h	16, 32, 64
SD Card Interrupt Flag Register 1	SDm_SD_INFO1	0000_0000_0000_0xx0h	0070h	16, 32, 64
SD Card Interrupt Flag Register 2	SDm_SD_INFO2	0000_0000_0000_20x0h	0078h	16, 32, 64
SD_INFO1 Interrupt Mask Register	SDm_SD_INFO1_MASK	0000_0000_0001_031Dh	0080h	16, 32, 64
SD_INFO2 Interrupt Mask Register	SDm_SD_INFO2_MASK	0000_0000_0000_8B7Fh	0088h	16, 32, 64
SD Clock Control Register	SDm_SD_CLK_CTRL	0000_0000_0000_0020h	0090h	16, 32, 64
Transfer Data Length Register	SDm_SD_SIZE	0000_0000_0000_0200h	0098h	16, 32, 64
SD Card Access Control Option Register	SDm_SD_OPTION	0000_0000_0000_40EEh	00A0h	16, 32, 64
Reserve	-	-	00A8h to 00AFh	-
Error Status Register 1	SDm_SD_ERR_STS1	0000_0000_xxxx_x000h	00B0h	16, 32, 64
SD Error Status Register 2	SDm_SD_ERR_STS2	0000_0000_0000_0000h	00B8h	16, 32, 64
SD Buffer Read/Write Register	SDm_SD_BUF0	xxxx_xxxx_xxxx_xxxxh	00C0h	16, 32, 64
Reserve	-	-	00C8h to 00CFh	-
SDIO Mode Control Register	SDm_SDIO_MODE	0000_0000_0000_0000h	00D0h	16, 32, 64
SDIO Interrupt Flag Register	SDm_SDIO_INFO1	0000_0000_0000_0000h	00D8h	16, 32, 64
SDIO_INFO1 Interrupt Mask Register	SDm_SDIO_INFO1_MASK	0000_0000_0000_C007h	00E0h	16, 32, 64
Reserve	-	-	00E8h to 035Fh	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
DMA Mode Enable Register	SDm_CC_EXT_MODE	0000_0000_0000_1010h	0360h	16, 32, 64
Reserve	-	-	0368h to 037Fh	-
Software Reset Register	SDm_SOFT_RST	0000_0000_0000_0007h	0380h	16, 32, 64
Version Register	SDm_VERSION	0000_0000_0000_CC10h	0388h	16, 32, 64
Host Interface Mode Setting Register	SDm_HOST_MODE	0000_0000_0000_0000h	0390h	16, 32, 64
SD Interface Mode Setting Register	SDm_SDIF_MODE	0000_0000_0000_0000h	0398h	16, 32, 64
Reserve	-	-	03A0h to 03C7h	-
SD Status Register	SDm_SD_STATUS	0000_0000_0000_0001h	03C8h	16, 32, 64
Reserve	-	-	03D0h to 081Fh	-
DMAC Transfer Mode Register	SDm_DM_CM_DTRAN_MODE	0000_0000_0000_0030h	0820h	16, 32, 64
DMAC Transfer Control Register	SDm_DM_CM_DTRAN_CTRL	0000_0000_0000_0000h	0828h	16, 32, 64
DMAC Reset Register	SDm_DM_CM_RST	0000_0000_FFFF_FFFFh	0830h	16, 32, 64
Reserve	-	-	0838h to 083Fh	-
DMAC Interrupt Register 1	SDm_DM_CM_INFO1	0000_0000_0000_0000h	0840h	16, 32, 64
DM_CM_INFO1 Interrupt Mask Register	SDm_DM_CM_INFO1_MASK	0000_0000_FFFF_FFFFh	0848h	16, 32, 64
DMAC Interrupt Register 2	SDm_DM_CM_INFO2	0000_0000_0000_0000h	0850h	16, 32, 64
DM_CM_INFO2 Interrupt Mask Register	SDm_DM_CM_INFO2_MASK	0000_0000_FFFF_FFFFh	0858h	16, 32, 64
Reserve	-	-	0860h to 087Fh	-
DMAC Transfer Address Register	SDm_DM_DTRAN_ADDR	0000_0000_0000_0000h	0880h	16, 32, 64
Reserve	-	-	0888h to 0FFFh	-

6.2.2.2 SD Register Description

The prefix (SDm_) of the register names is omitted in this and subsequent sections.

6.2.2.2.1 Command Type Register (SDm_SD_CMD)

The command type register (SD_CMD) is used to select the command type and response type. The command sequence is started by writing to SD_CMD.

For details on the SD_CMD setting, refer to **6.2.4.14 Example of SD_CMD Register Setting**.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0000h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MD7, MD6[1:0]		MD5	MD4	MD3	MD2 to MD0[2:0]			C1, C0[1:0]		CF45 to CF40[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15, 14	MD7, MD6[1:0]	0h	RW	Multiple Block Transfer Mode (enabled at multiple block transfer) 00b: CMD12 is automatically issued at multiple block transfer. 01b: CMD12 is not automatically issued at multiple block transfer. 10b: Setting prohibited 11b: Setting prohibited
13	MD5	0h	RW	Single/Multiple Block Transfer (enabled when the command with data is handled) 0b: Single block transfer 1b: multi-block transfer
12	MD4	0h	RW	Write/Read Mode (enabled when the command with data is handled) 0b: Write (SD/MMC host interface -> SD card) 1b: Read (SD/MMC host interface <- SD card)
11	MD3	0h	RW	Data Mode (Command Type) 0b: Command without data transfer (bc, bcr, ac) 1b: Command with data transfer (adtc)

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	MD2 to MD0[2:0]	0h	RW	Mode/Response Type 000: Normal mode The response type and the transfer mode are selected by SD_CMD[7:0], and the SD_CMD[15:11] setting is disabled. 001b: Setting prohibited 010b: Setting prohibited 011b: Extended mode/No response 100b: Extended mode/R1, R5, R6, or R7 response from the SD card 101b: Extended mode/R1b response from the SD card 110b: Extended mode/R2 response from the SD card 111b: Extended mode/R3 or R4 response from the SD card Some commands cannot be used in normal mode. For details, see **6.2.4.14 Example of SD_CMD Register Setting** to select mode/response type.
7, 6	C1, C0[1:0]	0h	RW	00b: CMD 01b: ACMD 10b: Setting prohibited 11b: Setting prohibited
5 to 0	CF45 to CF40[5:0]	0h	RW	Command Index These bits specify command format[45:40] (command index). [Examples] CMD6: SD_CMD[7:0] = 0000_0110b CMD18: SD_CMD[7:0] = 0001_0010b ACMD13: SD_CMD[7:0] = 0100_1101b

6.2.2.2.2 Command Argument Registers

Command arguments for SD cards are set in the SD command argument registers (SD_ARG). Set the command arguments before writing to SD_CMD.

Note that the argument of CMD12 within command sequences is 0000 0000h regardless of the setting of SD_ARG.

(1) Command Argument Register (SDm_SD_ARG)

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0010h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CF39 to CF8[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF39 to CF8[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
31 to 0	CF39 to CF8[31:0]	0h	RW	Set command format[39:8] (argument).

(2) Command Argument Register 1 (SDm_SD_ARG1) (Mirror of SDm_SD_ARG[31:16])

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0018h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF39 to CF24[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 0	CF39 to CF24[15:0]	0h	RW	Set command format[39:24] (argument).

6.2.2.2.3 Data Stop Register (SDm_SD_STOP)

The data stop register (SD_STOP) is used to enable or disable block counting at multiple block transfer, and to control the issuing of CMD12 within command sequences.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0020h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	HPIMODE	HPICMD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	SEC	-	-	-	-	-	-	-	STP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18	-	0h*1	RW	Reserved The written value should always be 0b.
17	HPIMODE	0h*1	RW	HPI Mode Enable 0b: Disables HPI mode. 1b: Enables HPI mode.
16	HPICMD	0h*1	RW*3	HPI Command Issue When HPICMD is set to 1b while HPIMODE is 1b, the HPI command (CMD12) is issued. This bit is cleared to 0 when reception of the response to CMD12 is completed. The timing with which this bit is set to 1b is as follows. - After reception of the response to CMD12 that was issued by setting the STP bit to 1b has been completed during the CMD6/CMD38 or CMD25 sequence. - After reception of the response to CMD24/CMD25 has been completed After HPICMD is set to 1b, do not write 0b to this bit while the CBSY bit in SD_INFO2 is 1b. Do not set this bit to 1b when the CBSY bit in SD_INFO2 is 0b.
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
8	SEC	0h*1	RW	Block Count Enable*2 0: Disables SD_SECCNT setting value. 1: Enables SD_SECCNT setting value. Set SEC to 1 at multiple block transfer. When SD_CMD is set as follows to start the command sequence while SEC is set to 1, CMD12 is automatically issued to stop multiple block transfer with the number of blocks which is set to SD_SECCNT. 1. CMD18 or CMD25 in normal mode (SD_CMD[10:8] = 000b) 2. SD_CMD[15:13] = 001b in extended mode (CMD12 is automatically issued, multiple block transfer) When the command sequence is halted because of a communications error or timeout, CMD12 is not automatically issued.
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	STP	0h*1	RW	Stop - When STP is set to 1 during multiple block transfer, CMD12 is issued to halt the transfer through the SD/MMC host interface. However, if a command sequence is halted because of a communications error or timeout, CMD12 is not issued. Although continued buffer access is possible even after STP has been set to 1, the buffer access error bit (ERR5 or ERR4) in SD_INFO2 will be set accordingly. - When STP has been set to 1 during transfer for single block write, the access end flag is set when SD_BUF becomes empty, and CMD12 is not issued. If SD_BUF does contain data, the access end flag is set on completion of reception of the busy state without CMD12 having been issued. - When STP has been set to 1 during transfer for single block read, the access end flag is set immediately after setting of the STP bit and CMD12 is not issued. - When STP is set to 1 during reception of the busy state after an R1b response, the access end flag is set on completion of reception of the busy state without CMD12 having been issued. - When STP is set to 1 after a command sequence has been completed, CMD12 is not issued and the access end flag is not set. - Set STP to 1 after the response end flag has been set. - Set STP to 0 after the response end flag has been set.

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 2. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1.

Note 3. Only effective when 1 is written.

6.2.2.2.4 Block Count Register (SDm_SD_SECCNT)

The block count register (SD_SECCNT) is used to specify the number of transfer blocks at multiple block transfer.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0028h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNT31 to CNT0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNT31 to CNT0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
31 to 0	CNT31 to CNT0[31:0]	0h	RW	Number of Transfer Blocks*1 When 0000 0001h is set, the number of transfer blocks is 1. : When 0000 FFFFh is set, the number of transfer blocks is 65535. : When FFFF FFFFh is set, the number of transfer blocks is 4294967295. Do not set this register to 0000 0000h if multiple blocks are to be transferred.

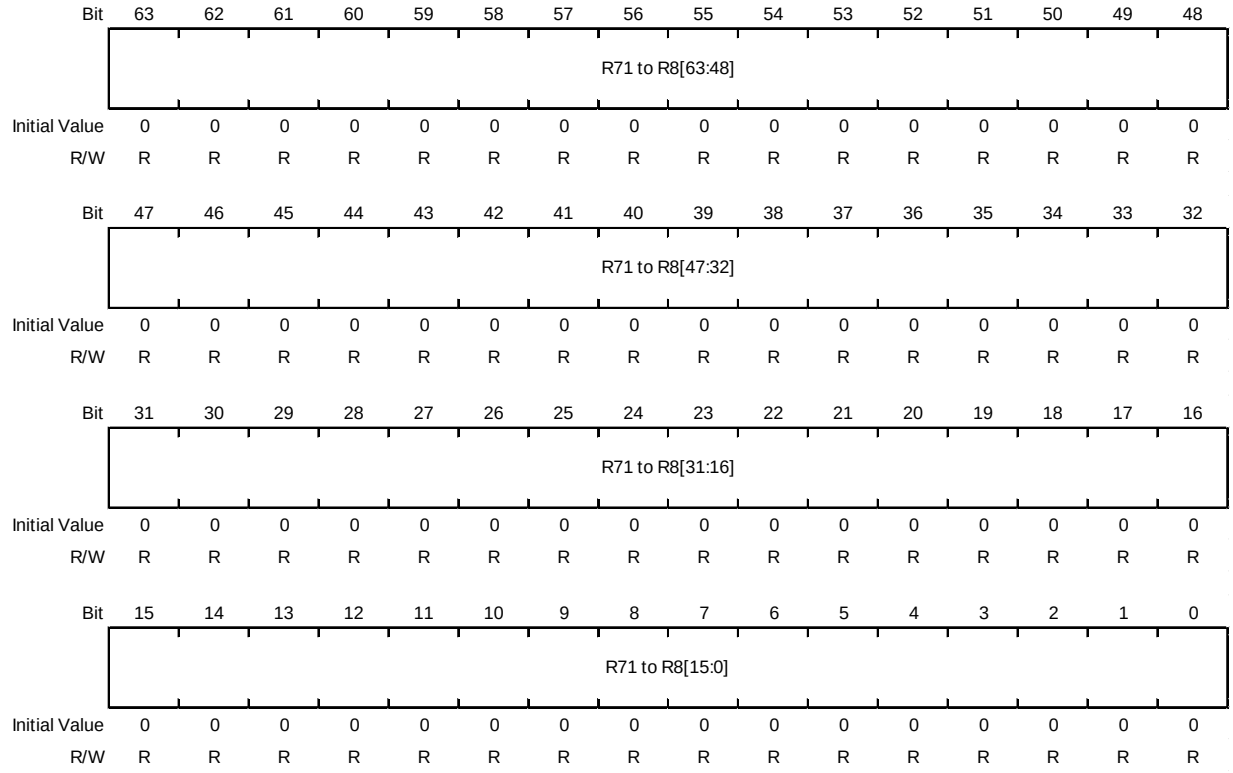
Note 1. Do not change the value of these bits when the CBSY bit in SD_INFO2 is set to 1.

6.2.2.2.5 SD Card Response Registers

The SD card response registers (SD_RSP) hold the response from the SD card.

(1) SD Card Response Registers 10 (SDm_SD_RSP10)

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0030h
Initial Value : 0000_0000_0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
63 to 0	R71 to R8[63:0]	0h	R	Hold the response from the SD card

(2) SD Card Response Registers 1 (SDm_SD_RSP1) (Mirror of SD_RSP10[31:16])

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0038h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R39 to R24[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 0	R39 to R24[15:0]	0h	R	Hold the response from the SD card

(3) SD Card Response Registers 32 (SDm_SD_RSP32) (Mirror of SD_RSP10[63:32])

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0040h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	R71 to R40[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R71 to R40[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
31 to 0	R71 to R40[31:0]	0h	R	Hold the response from the SD card

(4) SD Card Response Registers 3 (SDm_SD_RSP3) (Mirror of SD_RSP32[31:16])

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0048h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R71 to R56[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 0	R71 to R56[15:0]	0h	R	Hold the response from the SD card

(5) SD Card Response Registers 54 (SDm_SD_RSP54)

Access Size : 16, 32, 64 bits

Offset Address : <SDm_base> + 0050h

Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	R127 to R72[55:48]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	R127 to R72[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	R127 to R72[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R127 to R72[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 56	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
55 to 0	R127 to R72[55:0]	0h	R	Hold the response from the SD card

(6) SD Card Response Registers 5 (SDm_SD_RSP5) (Mirror of SD_RSP54[31:16])

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0058h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R103 to R88[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 0	R103 to R88[15:0]	0h	R	Hold the response from the SD card

(7) SD Card Response Registers 76 (SDm_SD_RSP76) (Mirror of SD_RSP54[63:32])

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0060h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	R127 to R104[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R127 to R104[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23 to 0	R127 to R104[23:0]	0h	R	Hold the response from the SD card

(8) SD Card Response Registers 7 (SDm_SD_RSP7) (Mirror of SD_RSP76[31:16])

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0068h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	R127 to R120[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 0	R127 to R120[7:0]	0h	R	Hold the response from the SD card

Table 6.2-3 lists the response types and corresponding SD_RSP registers.

Table 6.2-3 Response Types and Corresponding SD_RSP Registers

Response Types	SD_RSP Registers
R1, R1b[39:8]	SD_RSP10 SD_RSP54*1
R2[127:8]	SD_RSP54 and SD_RSP10
R3[39:8]	SD_RSP10
R4[39:8]	SD_RSP10
R5[39:8]	SD_RSP10
R6[39:8]	SD_RSP10
R7[39:8]	SD_RSP10

Note 1. The response to CMD18 and to CMD25 is stored in both R[39:8] and R[103:72].
This makes it possible to confirm the response to CMD18 and CMD25 by reading R[103:72] even if the response to automatic CMD12 is stored in R[39:8].

6.2.2.2.6 SD Card Interrupt Flag Register 1 (SDm_SD_INFO1)

The SD card interrupt flag register 1 (SD_INFO1) indicates the response end and access end in the command sequence. This register also indicates the card detect/write protect state.

For CMD12 and CMD52 (SDIO abort) at multiple block transfer, INFO0 is not set but only INFO2 is set.

Even if the command sequence is halted because of a communications error or timeout, INFO0 or INFO2 is set.

INFO10, INFO9, and INFO8 change depending on the SDDAT3 state after a reset is released and continue to change in 4-bit transfer mode.

To clear a flag, write 0b to the bit to be cleared and 1 to the other bits.

Access Size : 16, 32, 64 bits

Offset Address : <SDm_base> + 0070h

Initial Value : 0000_0000_0000_0xx0h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	HPIRES
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	INFO10	INFO9	INFO8	INFO7	-	INFO5	INFO4	INFO3	INFO2	-	INFO0
Initial Value	0	0	0	0	0	x	0	0	x	0	x	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	RW	RW	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	HPIRES	0h*2	RW*1	Response Reception Completion [Setting condition] When reception of the response to CMD12 that was issued by setting the STP bit to 1b is completed during the CMD6/CMD38 or CMD25 sequence in HPI mode. [Clearing condition] When 0b is written to HPIRES
15 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10	INFO10	x	R	Indicates the SDDAT3 state. 0b: SDDAT3 is set to 0. 1b: SDDAT3 is set to 1.

Bit	Bit Name	Initial Value	R/W	Description
9	INFO9	0h	RW*1	SDDAT3 Card Insertion [Setting condition] After a change in SDDAT3 from 0 to 1, two cycles of 2 P1ϕ have elapsed with SDDAT3 held at 1. [Clearing condition] When 0b is written to INFO9
8	INFO8	0h	RW*1	SDDAT3 Card Removal [Setting condition] After a change in SDDAT3 from 1 to 0, two cycles of 2 P1ϕ have elapsed with SDDAT3 held at 0. [Clearing condition] When 0b is written to INFO8
7	INFO7	x	R	Write Protect Indicates the ISDWP state. 0b: ISDWP is set to 1. 1b: ISDWP is set to 0.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	INFO5	x	R	Indicates the ISDCD state. 0b: Indicates that Mcycle has elapsed with ISDCD held at 1. 1b: Indicates that Mcycle has elapsed with ISDCD held at 0. Mcycle is set by bits 3 to 0 in SD_OPTION.
4	INFO4	0h	RW*1	ISDCD Card Insertion [Setting condition] After a change in ISDCD from 1 to 0, Mcycle has elapsed with ISDCD held at 0. [Clearing condition] When 0b is written to INFO4 Mcycle is set by bits 3 to 0 in SD_OPTION.
3	INFO3	0h	RW*1	ISDCD Card Removal [Setting condition] After a change in ISDCD from 0 to 1, Mcycle has elapsed with ISDCD held at 1. [Clearing condition] When 0 is written to INFO3 Mcycle is set by bits 3 to 0 in SD_OPTION.
2	INFO2	0h*2	RW*1	Access End [Setting conditions] 1. When read access to the buffer is completed in the case of transfer for single block read 2. When read access to the buffer for the last block of data is completed in the case of transfer for multiple block read 3. When read access to the buffer and reception of the response to CMD12 are completed in the case of transfer for multiple block read with automatic issuing of CMD12 4. When reception of the busy state after reception of the CRC status is completed in the case of transfer for single block write 5. When reception of the busy state after reception of the CRC status of the last block of data is completed in the case of transfer for multiple block write 6. When reception of the response busy state for CMD12 is completed in the case of transfer for multiple block write with automatic issuing of CMD12 7. When reception of the response to CMD12 that was issued by setting the STP bit to 1 is completed in the case of transfer for multiple block read 8. When reception of the response busy state for CMD12 that was issued by setting the STP bit to 1 is completed in the case of transfer for multiple block write 9. When reception of the response to CMD52 that was issued by setting the IOABT bit to 1 is completed in the case of transfer for multiple block read 10. When reception of the response to CMD52 that was issued by setting the IOABT bit to 1 is completed in the case of transfer for multiple block write. In addition to the above conditions, this bit is set when a command sequence is halted because of a communications error or timeout. [Clearing condition] When 0 is written to INFO2 When the access end bit is set to 1, the command sequence is terminated.
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
0	INFO0	0h*2	RW*1	Response End [Setting conditions] 1. When the reception of the response is completed 2. When the transmission of the command not requiring a response is completed, 3. When receiving busy reception after R1b response 4. When reception of the response to CMD52 that was issued by setting the C52PUB bit to1 is completed in the case of transfer for multiple block read 5. When reception of the response to CMD52 that was issued by setting the C52PUB bit to1 is completed in the case of transfer for multiple block write, In addition to the above conditions, this bit is set when a command sequence is halted because of a communications error or timeout. [Clearing condition] When 0 is written to INFO0 When issuing a command without data, the command sequence ends when the response end is set to 1.

x: Undefined value

Note 1. Only effective when 0b is written.

Note 2. The value is initialized by a reset and also in the case of a reset by the SDRST bit in SOFT_RST.

6.2.2.2.7 SD Card Interrupt Flag Register 2 (SDm_SD_INFO2)

The SD card interrupt flag register 2 (SD_INFO2) indicates the access status of the SD buffer (SD_BUF) and SD card. To clear a flag, write 0b to the bit to be cleared and 1 to the other bits.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0078h
Initial Value : 0000_0000_0000_20x0h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ILA	CBSY	SCLKDIVEN	-	-	-	BWE	BRE	DAT0	ERR6	ERR5	ERR4	ERR3	ERR2	ERR1	ERR0
Initial Value	0	0	1	0	0	0	0	0	x	0	0	0	0	0	0	0
R/W	RW	R	R	R	RW	R	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15	ILA	0h*2	RW*1	Illegal Access Error [Setting conditions] 1. Writing of data to SD_CMD within a command sequence (CBSY=1) 2. When SD_CMD[11] = 1 (command with data transfer) and SD_CMD[7:0] = 0000_1100b (CMD12) are set in SD_CMD [Clearing condition] When 0 is written to ILA
14	CBSY	0h*2	R	Command Type Register Busy 0: A command sequence has been completed. 1: A command sequence is being executed.
13	SCLKDIVEN	1h*2	R	0: The SD bus (CMD, DAT) is busy. Do not attempt to write to the SD_CLK_CTRL register. 1: The SD bus (CMD, DAT) is not busy. When a command sequence is started by writing to SD_CMD, the CBSY bit is set to 1 and, at the same time, the SCLKDIVEN bit is set to 0. The SCLKDIVEN bit is set to 1 after 8 cycles of SDCLK have elapsed after setting of the CBSY bit to 0 due to completion of the command sequence.
12	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
11	-	0h	RW	Reserved The written value should always be 1b.
10	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
9	BWE	0h*2	RW*1	SD_BUF Write Enable 0b: Data cannot be written in SD_BUF0. 1b: Data can be written in SD_BUF0. [Setting conditions] 1. When SD_BUF is empty at single block transfer 2. When either bank 1 or bank 2 of SD_BUF is empty at multiple block transfer [Clearing conditions] 1. When 0b is written to BWE 2. Writing of a block of data to SD_BUF by DMA transfer When data is written to SD_BUF0 by the CPU, clear BWE and then write the amount of data specified by SD_SIZE*3
8	BRE	0h*2	RW*1	SD_BUF Read Enable 0b: Data cannot be read from SD_BUF0. 1b: Data can be read from SD_BUF0. [Setting conditions] 1. When data set in SD_SIZE is stored in SD_BUF at single block transfer 2. When data set in SD_SIZE is stored in either bank 1 or bank 2 of SD_BUF at multiple block transfer [Clearing conditions] 1. When 0b is written to BRE 2. Reading of a block of data from SD_BUF by DMA transfer When data is read from SD_BUF0 by the CPU, clear BRE and then read the amount of data specified by SD_SIZE*3. Even if a CRC error or an END error occurs while block data is read, data is stored in SD_BUF and BRE is set.
7	DAT0	x	R	SDDAT0 Indicates the SDDAT0 state. 0b: SDDAT0 is set to 0. 1b: SDDAT0 is set to 1. If the data timeout (ERR3) is set but the response timeout (ERR6) is not set after the Erase command has been issued, the end of the Erase sequence (DAT0 = 1) is confirmed by polling DAT0. If a communications error or timeout occurs during a write sequence, the DAT0 bit may retain the value 0. While the SD clock (SDCLK) is stopped, the DAT0 bit retains the value before the clock is stopped.
6	ERR6	0h*2	RW*1	Response Timeout [Setting condition] When a response is not received even after 640 cycles of SDCLK have elapsed (including a response to a command issued within a command sequence*5) [Clearing condition] When 0b is written to ERR6 The command sequence is halted by a response timeout.*4
5	ERR5	0h*2	RW*1	SD_BUF Illegal Read Access [Setting conditions] 1. When SD_BUF is empty while SD_BUF0 is read 2. When data with a CRC error or END error is read from SD_BUF0 [Clearing condition] When 0b is written to ERR5
4	ERR4	0h*2	RW*1	SD_BUF Illegal Write Access [Setting conditions] 1. When data is written to SD_BUF0 while it is not in the data read/write command state 2. When data is written to SD_BUF0 while SD_BUF is full 3. When data is written to SD_BUF0 while an error occurs in the CRC status or CRC status length 4. When data is written to SD_BUF0 while the interface remains in a busy state for at least Ncycle after the CRC status [Clearing condition] When 0b is written to ERR4 Ncycle is set by bits 7 to 4 in SD_OPTION.

Bit	Bit Name	Initial Value	R/W	Description
3	ERR3	0h*2	RW*1	Data Timeout (except response timeout) [Setting conditions] 1. After the reception of the R1b response, the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle. 2. After the reception of the CRC status, the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle. 3. After write data, the CRC status is not received even after Ncycle has elapsed. 4. After the read command, read data is not received even after Ncycle has elapsed. 5. After CMD12 has been issued within a command sequence, the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle. 6. After the reception of read data, read data for the next block are not received even after Ncycle has elapsed. 7. After release of the read wait state, read data for the next block are not received even after Ncycle has elapsed. [Clearing condition] When 0b is written to ERR3 Ncycle is set by bits 7 to 4 in SD_OPTION. The command sequence is halted by the data timeout.
2	ERR2	0h*2	RW*1	END Error [Setting conditions] 1. When an error occurs in the response length (and the end bit has not been detected) 2. When an error occurs in the read data length (and the end bit has not been detected among the valid bits) 3. When an error occurs in the CRC status length (and the end bit has not been detected) 4. An error in the length of a response to a command issued within a command sequence*5 (i.e. the end bit has not been detected) [Clearing condition] When 0b is written to ERR2 The command sequence is halted by the End error.*4
1	ERR1	0h*2	RW*1	CRC Error [Setting conditions] 1. When an error occurs in the CRC status (i.e. the received CRC status was not 010b) 2. When a CRC error occurs in the read data 3. When a CRC error occurs in the response 4. A CRC error in the response to a command issued within a command sequence*5 [Clearing condition] When 0b is written to ERR1 The command sequence is halted by the CRC error.*4
0	ERR0	0h*2	RW*1	CMD Error [Setting conditions] 1. The command index of the transmitted command differing from the command index of the received response 2. The command index of a command issued within a command sequence*5 differing from the command index of the received response [Clearing condition] When 0b is written to ERR0 The command sequence is halted by the CMD error.*4

x: Undefined value

Note 1. Only effective when 0b is written.

Note 2. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0.

Note 3. When the WMODE bit in HOST_MODE is 0, the single byte from the fraction of a full 16-bit unit is regarded as excess data due to an odd value for the number of bytes setting in SD_SIZE. When the WMODE bit in HOST_MODE is 1, the single byte or three bytes from the fraction of a full 64-bit unit are regarded as excess data due to an odd value for the number of bytes setting in SD_SIZE, or the two bytes from the fraction of a full 64-bit unit are regarded as excess data due if the value for the number of bytes setting in SD_SIZE is even but is not on a four-byte boundary.

Note 4. After the C52PUB bit in SDIO_MODE has been set to 1b, if a communications error or timeout for response occurs in response to the CMD52 that is issued, since the command sequence has not been completed, complete the sequence with error processing as in usage examples in **Figure 6.2-18** under **6.2.4.8 IO_RW_EXTENDED (CMD53/Multiple Block Read)** or in **Figure 6.2-21** under **6.2.4.9 IO_RW_EXTENDED (CMD53/Multiple Block Write)**.

Note 5. “Command issued within a command sequence” refers to CMD12 when automatic issuing is enabled for multiple block transfer by the setting in SD_CMD, CMD12 when the STP bit in SD_STOP is set to 1b, or CMD52 when the C52PUB or IOABT bit in SDIO_MODE is set to 1b.

6.2.2.2.8 SD_INFO1 Interrupt Mask Register (SDm_SD_INFO1_MASK)

The SD_INFO1 interrupt mask register (SD_INFO1_MASK) is used to enable or disable the SD_INFO1 interrupt.

When 0b is set in SD_INFO1_MASK while the corresponding flag in SD_INFO1 is set, an interrupt occurs.

Access Size : 16, 32, 64 bits

Offset Address : <SDm_base> + 0080h

Initial Value : 0000_0000_0001_031Dh

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	IMASK16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	IMASK9	IMASK8	-	-	-	IMASK4	IMASK3	IMASK2	-	IMASK0
Initial Value	0	0	0	0	0	0	1	1	0	0	0	1	1	1	0	1
R/W	R	R	R	R	R	R	RW	RW	R	R	R	RW	RW	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	IMASK16	1h	RW	HPIRES interrupt masked
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	IMASK9	1h	RW	INFO9 interrupt masked
8	IMASK8	1h	RW	INFO8 interrupt masked
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	IMASK4	1h	RW	INFO4 interrupt masked
3	IMASK3	1h	RW	INFO3 interrupt masked
2	IMASK2	1h	RW	INFO2 interrupt masked
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	IMASK0	1h	RW	INFO0 interrupt masked

6.2.2.2.9 SD_INFO2 Interrupt Mask Register (SDm_SD_INFO2_MASK)

The SD_INFO2 interrupt mask register (SD_INFO2_MASK) is used to enable or disable the SD_INFO2 interrupt.

When 0b is set in SD_INFO2_MASK while the corresponding flag in SD_INFO2 is set, an interrupt occurs.

Access Size : 16, 32, 64 bits

Offset Address : <SDm_base> + 0088h

Initial Value : 0000_0000_0000_8B7Fh

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IMASK	-	-	-	-	-	BMASK1	BMASK0	-	EMASK6	EMASK5	EMASK4	EMASK3	EMASK2	EMASK1	EMASK0
Initial Value	1	0	0	0	1	0	1	1	0	1	1	1	1	1	1	1
R/W	RW	R	R	R	RW	R	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15	IMASK	1h	RW	ILA interrupt masked
14 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
11	-	1h	RW	Reserved The written value should always be 1b.
10	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	BMASK1	1h	RW	BWE interrupt masked
8	BMASK0	1h	RW	BRE interrupt masked
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6	EMASK6	1h	RW	ERR6 interrupt masked
5	EMASK5	1h	RW	ERR5 interrupt masked
4	EMASK4	1h	RW	ERR4 interrupt masked
3	EMASK3	1h	RW	ERR3 interrupt masked
2	EMASK2	1h	RW	ERR2 interrupt masked
1	EMASK1	1h	RW	ERR1 interrupt masked
0	EMASK0	1h	RW	ERR0 interrupt masked

6.2.2.2.10 SD Clock Control Register (SDm_SD_CLK_CTRL)

The SD clock control register (SD_CLK_CTRL) is used to control the SD clock (SDCLK) output and to set the frequency. Set SCLKEN to 1 before writing to SD_CMD to issue a command. Do not write to SD_CMD while the SCLKDIVEN bit in SD_INFO2 is set to 0.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0090h
Initial Value : 0000_0000_0000_0020h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	SDCLKOFFEN	SCLKEN	DIV7 to DIV0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	-	0h	RW	Reserved The written value should always be 0b.
15 to 11	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10	-	0h*1	RW	Reserved The written value should always be 0b.
9	SDCLKOFFEN	0h	RW	SD Clock (SDCLK) Output Automatic Control Enable 0b: Automatic control for SD clock (SDCLK) output is disabled. 1b: Automatic control for SD clock (SDCLK) output is enabled. This function of automatic control for SD clock (SDCLK) output causes SDCLK output only within a command sequence. The timing with which SDCLK output starts and stops is as follows. SDCLK output starts after writing to SD_CMD. SDCLK output stops when 8 cycles of SDCLK have elapsed after the end of the command sequence. In addition, SDCLK is fixed to 0 while SCLKEN of SD_CLK_CTRL is 0, regardless of the value of this bit.
8	SCLKEN	0h*1	RW	SD Clock (SDCLK) Output Control Enable 0b: SD clock (SDCLK) output is disabled. The SDCLK signal is fixed 0. 1b: SD clock (SDCLK) output is enabled.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	DIV7 to DIV0[7:0]	20h	RW*2	SD Clock (SDCLK) 1000_0000b: (SDxϕ/4)/512 0100_0000b: (SDxϕ/4)/256 0010_0000b: (SDxϕ/4)/128 0001_0000b: (SDxϕ/4)/64 0000_1000b: (SDxϕ/4)/32 0000_0100b: (SDxϕ/4)/16 0000_0010b: (SDxϕ/4)/8 0000_0001b: (SDxϕ/4)/4 0000_0000b: (SDxϕ/4)/2 1111_1111b: SDxϕ/4 (x = 0, 1) Other settings are prohibited. In addition, in the case of data transfer in DDR mode (DDR bit in SDIF_MODE = 1), do not set DIV[7:0] to 1111_1111b.

Note 1. This initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0b.

Note 2. Writing to SD_CLK_CTRL is impossible when the CBSY bit in SD_INFO2 is 1b.

CAUTION

Notes on when setting the SD clock (SDCLK) to P1ϕ (DIV[7:0] = 1111_1111b)

When changing the setting of bits DIV[7:0] to 1111_1111b, or from 1111_1111b to other setting, perform the following processing before writing to SD_CMD.

1. Set the SCLKEN bit to 0 by writing to SD_CLK_CTRL. (Do not change the setting of bits other than SCLKEN at this time.)
2. Change the setting of bits DIV[7:0] by writing to SD_CLK_CTRL. (Do not change the setting of bits other than DIV[7:0] at this time. The SCLKEN bit should retain the value 0.)
3. Set the SCLKEN bit to 1b by writing to SD_CLK_CTRL. (Do not change the setting of bits other than SCLKEN at this time.)

Also, when changing the setting of bits DIV[7:0] to 1111_1111b after having set the SDRST bit in SOFT_RST to 0 and then changed it to 1b, perform this processing before writing to SD_CMD.

6.2.2.2.11 Transfer Data Length Register (SDm_SD_SIZE)

The transfer data length register (SD_SIZE) is used to specify the transfer data size.

Access Size : 16, 32, 64 bits

Offset Address : <SDm_base> + 0098h

Initial Value : 0000_0000_0000_0200h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	LEN9 to LEN0[9:0]									
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
11 to 10	-	All 0	R	Reserved
9 to 0	LEN9 to LEN0[9:0]	200h	RW	Transfer Data Size*1 These bits specify a size between 1 and 512 bytes for single block transfer. In cases of multiple block transfer with automatic issuing of CMD12 (CMD18 and CMD25), the only specifiable transfer data size is 512 bytes. Furthermore, in cases of multiple block transfer without automatic issuing of CMD12, as well as 512 bytes, 32, 64, 128, and 256 bytes are specifiable. However, in the reading of 32, 64, 128, and 256 bytes for the transfer of multiple blocks, this is restricted to multiple block transfer by CMD53. Additionally, if a command accompanies data transfer, do not set these bits to 0. Do not specify a data size larger than 512 bytes.

Note 1. Do not change the values of these bits when the CBSY bit in SD_INFO2 is 1b.

6.2.2.2.12 SD Card Access Control Option Register (SDm_SD_OPTION)

The SD card access control option register (SD_OPTION) is used to set the bus width and timeout counter.

Access Size : 16, 32, 64 bits

Offset Address : <SDm_base> + 00A0h

Initial Value : 0000_0000_0000_40EEh

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WIDTH	-	WIDTH8	-	-	-	EXTOP	TOUTMASK	TOP27 to TOP24[3:0]				CTOP24 to CTOP21[3:0]			
Initial Value	0	1	0	0	0	0	0	0	1	1	1	0	1	1	1	0
R/W	RW	R	RW	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15	WIDTH	0h*1	RW	Bus width*2 {WIDTH,WIDTH8} = 01b: 8-bit width {WIDTH,WIDTH8} = 00b: 4-bit width {WIDTH,WIDTH8} = 10b or 11b: 1-bit width In the case of data transfer in DDR mode (DDR bit in SDIF_MODE = 1), do not set this bit to 1b. In the case of writing of one-byte block, 8-bit width cannot be specified for the bus width. Change the bus width to 4 bits or 1 bit before writing one-byte block.
14	-	1h	R	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
13	WIDTH8	0h*1	RW	Bus width*2 See the description of the WIDTH bit.
12 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	EXTOP	0h*1	RW	Timeout Mode Select 0b: Bits TOP27 to TOP24 specify the timeout count from SDCLK x 2^{13} to SDCLK x 2^{27}. 1b: Bits TOP27 to TOP24 specify the timeout count from SDCLK x 2^{14} to SDCLK x 2^{28}.
8	TOUTMASK	0h*1	RW	Timeout Mask 0b: Enables timeout. 1b: Disables timeout. (The ERR6 and ERR3 bits in SD_INFO2 and the E6 to E0 bits in SD_ERR_STS2 are not set.) If a timeout occurs while it is disabled, perform a software reset to terminate a command sequence.

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	TOP27 to TOP24[3:0]	Eh*1	RW	Timeout Counter*2 0000b: SDCLK x 2^{13} 0001b: SDCLK x 2^{14} : 1101b: SDCLK x 2^{26} 1110b: SDCLK x 2^{27} 1111b: Setting prohibited
3 to 0	CTOP24 to CTOP21[3:0]	Eh*1	RW	Card Detect Time Counter 0000b: SDCLK x 2^{10} 0001b: SDCLK x 2^{11} : 1101b: SDCLK x 2^{23} 1110b: SDCLK x 2^{24} 1111b: Setting prohibited

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0b.

Note 2. Do not change the values of these bits when the CBSY bit in SD_INFO2 is 1b.

6.2.2.2.13 SD Error Status Register 1 (SDm_SD_ERR_STS1)

The SD error status register 1 (SD_ERR_STS1) indicates the CRC status, CRC error, End error, and CMD error.

Access Size : 16, 32, 64 bits

Offset Address : <SDm_base> + 00B0h

Initial Value : 0000_0000_xxxx_x000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	E14 to E12[2:0]			E11	E10	E9	E8	-	-	E5	E4	E3	E2	E1	E0
Initial Value	–	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
31 to 15	-	X	R	Reserved The read value is undefined. The written value should always be 0b.
14 to 12	E14 to E12[2:0]	2h*1	R	These bits hold the CRC status. (normal: 010b)
11	E11	0h*1	R	Set to 1 when an error occurs in the CRC status.
10	E10	0h*1	R	Set to 1 when a CRC error occurs in the read data.
9	E9	0h*1	R	Set to 1 when a CRC error occurs in the response to a command issued within a command sequence*2. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E8.
8	E8	0h*1	R	Set to 1 when a CRC error occurs in a response (other than a response to a command issued within a command sequence*2).
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	E5	0h*1	R	Set to 1 when an error occurs in the CRC status length (and the end bit has not been detected).
4	E4	0h*1	R	Set to 1 when an error occurs in the read data length (and the end bit has not been detected among the valid bits).
3	E3	0h*1	R	Set to 1 when an error occurs in the response length to a command issued within a command sequence*2. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E2.
2	E2	0h*1	R	Set to 1 when an error occurs in the response length (other than a response to a command issued within a command sequence*2).
1	E1	0h*1	R	Set to 1 when an error occurs in the command index of the response to a command issued within a command sequence*2. In cases where CMD12 is issued by setting a command index in SD_CMD, this is Indicated in E0.

Bit	Bit Name	Initial Value	R/W	Description
0	E0	0h*1	R	Set to 1 when an error occurs in the command index of a response (other than a response to a command issued within a command sequence*2).

x: Undefined value

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0b.

Note 2. “Command issued within a command sequence” refers to CMD12 when automatic issuing is enabled for multiple block transfer by the setting in SD_CMD, CMD12 when the STP bit in SD_STOP is set to 1b, or CMD52 when the C52PUB or IOABT bit in SDIO_MODE is set to 1b.

6.2.2.2.14 SD Error Status Register 2 (SDm_SD_ERR_STS2)

The SD error status register 2 (SD_ERR_STS2) indicates the timeout state. Ncycle is set by bits 7 to 4 in SD_OPTION.

Access Size : 16, 32, 64 bits

Offset Address : <SDm_base> + 00B8h

Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	E6	E5	E4	E3	E2	E1	E0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6	E6	0h*1	R	Set to 1b when the interface remains in a busy state for at least Ncycle after the CRC status.
5	E5	0h*1	R	Set to 1b when the CRC status is not received even after Ncycle has elapsed after data writing.
4	E4	0h*1	R	Set to 1b when read data is not received even after Ncycle has elapsed after the command has been read. Set to 1b when read data for the next block are not received even after Ncycle has elapsed after the reception of read data. Set to 1b when read data for the next block are not received even after Ncycle has elapsed after release of the read wait state.
3	E3	0h*1	R	Set to 1b when the interface remains in a busy state for at least Ncycle after CMD12 has been issued within a command sequence. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E2.
2	E2	0h*1	R	Set to 1b when the interface remains in a busy state for at least Ncycle after R1b response.
1	E1	0h*1	R	Set to 1b when the response to a command issued within a command sequence*2 is not received even after 640 cycles of SDCLK have elapsed. In cases where CMD12 is issued by setting a command index in SD_CMD, this is indicated in E0.
0	E0	0h*1	R	Set to 1b when the response (other than a response to a command issued within a command sequence*2) is not received even after 640 cycles of SDCLK have elapsed.

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0b.

Note 2. “Command issued within a command sequence” refers to CMD12 when automatic issuing is enabled for multiple block transfer by the setting in SD_CMD, CMD12 when the STP bit in SD_STOP is set to 1b, or CMD52 when the C52PUB or IOABT bit in SDIO_MODE is set to 1b.

6.2.2.2.15 SD Buffer Read/Write Register (SDm_SD_BUF0)

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 00C0h
Initial Value : xxxx_xxxx_xxxx_xxxxh

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	BUF63 to BUF0[63:48]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	BUF63 to BUF0[47:32]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BUF63 to BUF0[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BUF63 to BUF0[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 0	BUF63 to BUF0[63:0]	x	RW	When writing to the SD card, the write data is written to this register. When reading from the SD card, the read data is read from this register. This register is internally connected to two 512-byte buffers (SD_BUF). When both buffers are not empty at multiple block read, suspend data reception by stopping the SD clock. When either buffer becomes empty, restart data reception by starting supply of the SD clock.

x: Undefined value

Note: When using the DMAC, the bus width should be fixed at 64 bits.

6.2.2.2.16 SDIO Mode Control Register (SDm_SDIO_MODE)

The SDIO mode control register (SDIO_MODE) controls the CMD52 issuance and the read wait state at multiple block transfer, and the reception of SDIO interrupt. C52PUB and IOABT should not be set to 1b simultaneously.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 00D0h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	C52PUB	IOABT	-	-	-	-	-	RWREQ	-	IOMOD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	C52PUB	0h	RW	SDIO None Abort - When C52PUB is set to 1b in the CMD53 (multiple block) write sequence, CMD52 is automatically issued between blocks if SD_BUF becomes empty. C52PUB is automatically cleared to 0 after reception of the response to CMD52 is completed. Additionally, if C52PUB is set to 1b while the last block is being transferred, CMD52 is not issued. In this case, C52PUB is automatically cleared to 0 after the access end flag has been set to 1b. - When C52PUB and RWREQ are set to 1b in the CMD53 (multiple block) read sequence, the block transfer enters the read wait state between blocks and CMD52 is automatically issued. C52PUB is automatically cleared to 0 after reception of the response to CMD52 is completed. Additionally, if C52PUB is set to 1b while the last block is being transferred, CMD52 is not issued. In this case, C52PUB is automatically cleared to 0 after the access end flag has been set to 1b. - If C52PUB is set to 1b in the CMD53 (multiple block) read sequence, be sure to set RWREQ to 1 as well as C52PUB. - Set SD_ARG before setting C52PUB to 1. - Set C52PUB to 1 after the response end flag has been set.

Bit	Bit Name	Initial Value	R/W	Description
8	IOABT	0h	RW	SDIO Abort - When IOABT is set to 1b in the CMD53 (multiple block) sequence, the CMD53 sequence is halted and CMD52 is issued. However, if a command sequence is halted because of a communications error or timeout, CMD52 is not issued. Although continued buffer access is possible even after IOABT has been set to 1b, the buffer access error bit (ERR5 or ERR4) in SD_INFO2 will be set accordingly. Set SD_ARG before setting IOABT to 1. - When IOABT has been set to 1b during transfer for single block write, the access end flag is set when SD_BUF becomes empty, and CMD52 is not issued. If SD_BUF does contain data, the access end flag is set on completion of reception of the busy state without CMD52 having been issued. - When IOABT has been set to 1b during transfer for single block read, the access end flag is set immediately after setting of IOABT and CMD52 is not issued. - When IOABT is set to 1b during reception of the busy state after an R1b response, the access end flag is set on completion of reception of the busy state without CMD52 having been issued. - When IOABT is set to 1b after a command sequence has been completed, CMD52 is not issued and the access end flag is not set. - Set IOABT to 1 after the response end flag has been set. Set IOABT to 0 after the access end flag has been set.
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	RWREQ	0h	RW	Read Wait Request When RWREQ is set to 1 in the CMD53 (multiple block) read sequence, the block transfer enters the read wait state between blocks. [Releasing the read wait state] (1) The read wait state is released when RWREQ is cleared to 0 in the read wait state. (2) When IOABT is set to 1 in the read wait state, RWREQ is automatically cleared to 0 after CMD52 has been issued, and then the read wait state is released. (3) When C52PUB and RWREQ are set to 1 simultaneously in the CMD53 (multiple block) read sequence, the read wait state is not automatically released. Therefore, after the CMD52 response is received, clear RWREQ. (Be sure to set RWREQ and C52PUB simultaneously.) When RWREQ is set to 1 while the last block in the CMD53 (multiple block) read sequence is transferred, the read wait state is not entered and RWREQ is automatically cleared to 0 by setting access end. Set RWREQ to 1 after the response end flag has been set.
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	IOMOD	0h	RW	SDIO Mode*[1] 0b: Disables the SD/MMC host interface to receive SDIO interrupts from the SDIO card 1b: Enables the SD/MMC host interface to receive SDIO interrupts from the SDIO card

Note 1. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1b.

6.2.2.2.17 SDIO Interrupt Flag Register (SDm_SDIO_INFO1)

The SDIO interrupt flag register (SDIO_INFO1) indicates the status regarding to the SDIO card access. To clear a flag, write 0b to the bit to be cleared and 1b to the other bits.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 00D8h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EXWT	EXPUB52	-	-	-	-	-	-	-	-	-	-	-	-	-	IOIRQ
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15	EXWT	0h*2	RW*1	[Setting condition] While the last block in the CMD53 (multiple block) read sequence is transferred, RWREQ in SDIO_MODE is set to 1. [Clearing condition] When 0 is written to EXWT
14	EXPUB52	0h*2	RW*1	[Setting conditions] 1. While the last block in the CMD53 (multiple block) sequence is transferred, C52PUB in SDIO_MODE is set to 1b. 2. While C52PUB is set to 1b in the CMD53 (multiple block) write sequence, the last block is transferred. [Clearing condition] When 0b is written to EXPUB52
13 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2, 1	-	All 0*2	RW	Reserved The read value is undefined. The written value should always be 1b.
0	IOIRQ	0h*2	RW*1	[Setting condition] When an SDIO interrupt from the SDIO card is received while IOMOD in SDIO_MODE is set to 1b. [Clearing condition] When 0b is written to IOIRQ*3

Note 1. Only effective when 0b is written.

Note 2. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0b.

Note 3. Before clearing this bit, access the SDIO card to negate the SDIO interrupt signal from the SDIO card. If the interrupt signal is not negated, this bit may be set again.

6.2.2.2.18 SDIO_INFO1 Interrupt Mask Register (SDm_SDIO_INFO1_MASK)

The SDIO_INFO1 interrupt mask register (SDIO_INFO1_MASK) enables or disables the SD_INFO1 interrupt. When 0b is set in SDIO_INFO1_MASK while the corresponding flag in SD_INFO1 is set, an interrupt occurs.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 00E0h
Initial Value : 0000_0000_0000_C007h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MEXWT	MEXPUB52	-	-	-	-	-	-	-	-	-	-	-	-	-	IOMSK
Initial Value	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	RW	RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15	MEXWT	1h	RW	EXWT interrupt masked
14	MEXPUB52	1h	RW	EXPUB52 interrupt masked
13 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2, 1	-	All 1	RW	Reserved The written value should always be 1b.
0	IOMSK	1h	RW	IOIRQ interrupt masked

6.2.2.2.19 DMA Mode Enable Register (SDm_CC_EXT_MODE)

The DMA mode enable register (CC_EXT_MODE) enables the DMA transfer.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0360h
Initial Value : 0000_0000_0000_1010h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DMASDRW	-
Initial Value	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	RW	R	R	R	RW	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12	-	1h	R	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
11, 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9, 8	-	All 0	RW	Reserved The written value should always be 0b.
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	-	0h	RW	Reserved The written value should always be 0b.
4	-	1h	R	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	DMASDRW	0h	RW	SD_BUF Read/Write DMA Transfer*1 0b: The SD_BUF read/write DMA transfer is disabled. 1b: The SD_BUF read/write DMA transfer is enabled.
0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Note 1. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1b.

6.2.2.2.20 Software Reset Register (SDm_SOFT_RST)

The software reset register (SOFT_RST) sets a software reset. Also use this register to check that release from the reset state has been completed before attempting to use the SD/MMC host interface and before attempting access to the other registers.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0380h
Initial Value : 0000_0000_0000_0007h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SDRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2, 1	-	All 1	R	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
0	SDRST	1h	RW	Software Reset of SD Interface Unit 0b: Reset 1b: Reset released

6.2.2.2.21 Version Register (SDm_VERSION)

The version register (VERSION) indicates the version of the SD/MMC host interface.

Access Size : 16, 32, 64 bits

Offset Address : <SDm_base> + 0388h

Initial Value : 0000_0000_0000_CC10h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UR7, UR6[1:0]		UR5, UR4[1:0]		UR3 to UR0[3:0]				IP7 to IP0[7:0]							
Initial Value	1	1	0	0	1	1	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15, 14	UR7, UR6[1:0]	3h	R	Reserved These bits are always read as 1b. The write value should always be 1. Operation is not guaranteed if a value other than 1 is written to these bits.
13, 12	UR5, UR4[1:0]	0h	R	Reserved These bits are always read as 0b. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.
11 to 8	UR3 to UR0[3:0]	Ch	R	Version of Renesas' IP
7 to 0	IP7 to IP0[7:0]	10h	R	Version of introductory IP

6.2.2.2.22 Host Interface Mode Setting Register (SDm_HOST_MODE)

The host interface mode setting register (HOST_MODE) selects the width for access to the data bus.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0390h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	BUSWIDTH	-	-	-	-	-	-	ENDIAN	WMODE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8	BUSWIDTH	0h	RW	Width for Access to SD_BUF*1*2 Read or write access to SD_BUF0 can be performed with the specified width for access. 0b: 16-bit access 1b: 32-bit access This bit is enabled while the WMODE bit is set to 1b.
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	ENDIAN	0h	RW	SD_BUF0 data swap
0	WMODE	0h	RW	Width for Access to SD_BUF*1*2 Read or write access to SD_BUF0 can be performed with the specified width for access. 0b: 64-bit access 1b: 16-bit or 32-bit access

Note 1. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1b.

Note 2. When using the built-in DMAC of this module, fix the bus width to 64 bits.

6.2.2.2.23 SD Interface Mode Setting Register (SDm_SDIF_MODE)

The SD interface mode setting register specifies DDR mode.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0398h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	NOCHKCR	-	-	-	-	-	-	-	DDR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	-	0h*1	RW	Reserved The written value should always be 0b.
8	NOCHKCR	0h*1	RW	CRC Check Mask (test command for MMC supported) Enables or disables checking of the CRC16 and CRC status. 0b: Enables the CRC check. 1b: Disables the CRC check (the CRC16 value is ignored at read, and the CRC status is not detected at write)
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	DDR	0h*1	RW	DDR Mode Select*2 0b: Normal mode (default, high speed, or SDR) 1b: DDR mode Set this bit to 0b when the SD clock division ratio is specified as 1:1 (bits DIV[7:0] in SD_CLK_CTRL are set to 1111_1111b).

Note 1. The initial value is applied at a reset and when the SDRST bit in SOFT_RST is 0b.

Note 2. Do not change the value of this bit when the CBSY bit in SD_INFO2 is set to 1b.

6.2.2.2.24 SD Status Register (SDm_SD_STATUS)

The effective bit of the SD status register controls the output value on the SDmPWEN, SDmIOVS and SD0RSTN pins.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 03C8h
Initial Value : 0000_0000_0000_0001h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SD_IOVS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SD_RST	SD_PWEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	SD_IOVS	0h	RW	Controls the output value on SDmIOVS pin. 0b: The output value on the SDmIOVS pin is 0 (3.3 V). 1b: The output value on the SDmIOVS pin is 1 (1.8 V).
15 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	SD_RST	0h	RW	Controls the output value on the SD0RSTN pin. (SD0 only) 0b: The output value on the SD0RSTN pin is 0. 1b: The output value on the SD0RSTN pin is 1.
0	SD_PWEN	1h	RW	Controls the output value on the SDmPWEN pin. 0b: The ouptut value on the SDmPWEN pin is 0. 1b: The ouput value on the SDmPWEN pin is 1.

6.2.2.2.25 DMAC Transfer Mode Register (SDm_DM_CM_DTRAN_MODE)

The DMAC transfer mode register (DM_CM_DTRAN_MODE) sets the operation mode of the DMAC of this module.

Higher-order 32 bits (bit 63 to 32) are read-only and read as 0.

Access Size : 16, 32, 64 bits

Offset Address : <SDm_base> + 0820h

Initial Value : 0000_0000_0000_0030h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CH_NUM[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	BUS_WIDTH [1:0]		-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
17, 16	CH_NUM[1:0]	0h	RW	DMAC channel selector 00b : SD downstream 01b : SD upstream Other settings are prohibited.
15 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5, 4	BUS_WIDTH [1:0]	3h	RW	Bus width selector 11b: 64 bits Other settings are prohibited.
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

6.2.2.2.26 DMAC Transfer Control Register (SDm_DM_CM_DTRAN_CTRL)

The DMAC transfer control register (DM_CM_DTRAN_CTRL) controls DMAC operation of the module. Higher-order 32 bits (bit 63 to 32) are read-only and read as 0.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0828h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DM_START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8	-	0h	RW	Reserved The written value should always be 0b.
7 to 1	-	All 0	R	Reserved
0	DM_START	0h	RW	DMAC Start Writing 1b to this bit starts DMAC operation. This bit is automatically cleared when DMA transfer is started.

6.2.2.2.27 DMAC Reset Register (SDm_DM_CM_RST)

Higher-order 32 bits (bit 63 to 32) are read-only and read as 0b.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0830h
Initial Value : 0000_0000_FFFF_FFFFh

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	DTRANRST1	DTRANRST0	-	-	-	-	-	-	-	SEQRST
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
31 to 10	-	All 1	R	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
9	DTRANRST1	1h	RW	Software reset of DMAC channel 1 of the module 0b: Reset 1b: Reset released
8	DTRANRST0	1h	RW	Software reset of DMAC channel 0 of the module 0b: Reset 1b: Reset released
7 to 1	-	All 1	R	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
0	SEQRST	1h	RW	Software reset of the sequencer 0b: Reset 1b: Reset released

NOTE

Make sure there is no communication to the SD/MMC device before applying a software reset using this register.

6.2.2.2.28 DMAC Interrupt Register 1 (SDm_DM_CM_INFO1)

The DMAC interrupt register 1 (DM_CM_INFO1) indicates the status of the DMAC of the module and the sequencer. To clear a flag, write 0b to the bit to be cleared and 1b to the other bits.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0840h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	DTRANEND1	-	-	-	DTRANEND0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SEQEND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20	DTRANEND1	0h*1	RW	DMAC Channel 1 Transfer End [Setting conditions] 1. When transfer of DMAC channel 1 is completed 2. When an error occurs on DMAC channel 1 [Clearing condition] When 0b is written to DTRANEND1
19 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	DTRANEND0	0h*2	RW	DMAC Channel 0 Transfer End [Setting conditions] 1. When transfer of DMAC channel 0 is completed 2. When an error occurs on DMAC channel 0 [Clearing condition] When 0b is written to DTRANEND0
15 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	SEQEND	0h*3	RW	Sequencer Operation End [Setting conditions] 1. When operation of the sequencer is completed 2. When a sequencer error occurs [Clearing condition] When 0b is written to SEQEND

Note 1. The initial value is applied at a reset and when the DTRANRST1 bit in DM_CM_RST is 0b.

Note 2. The initial value is applied at a reset and when the DTRANRST0 bit in DM_CM_RST is 0b.

Note 3. The initial value is applied at a reset and when the SEQRST bit in DM_CM_RST is 0b.

6.2.2.2.29 DM_CM_INFO1 Interrupt Mask Register (SDm_DM_CM_INFO1_MASK)

The DM_CM_INFO1 interrupt mask register (DM_CM_INFO1_MASK) enables or disables the DM_CM_INFO1 interrupt. When 0 is set in DM_CM_INFO1_MASK while the corresponding flag in DM_CM_INFO1 is set, an interrupt occurs.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0848h
Initial Value : 0000_0000_FFFF_FFFFh

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	DTRANEND1_MASK	-	-	-	DTRANEND0_MASK
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SEQEND_MASK
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
31 to 21	-	All 1	R	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
20	DTRANEND1_MASK	1h	RW	DTRANEND1 interrupt masked
19 to 17	-	All 1	RW	Reserved The written value should always be 1b.
16	DTRANEND0_MASK	1h	RW	DTRANEND0 interrupt masked
15 to 1	-	All 1	R	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
0	SEQEND_MASK	1h	RW	SEQEND interrupt masked

6.2.2.2.30 DMAC Interrupt Register 2 (SDm_DM_CM_INFO2)

The DMAC interrupt register 2 (DM_CM_INFO2) indicates the status of the DMAC of the module and the sequencer. To clear a flag, write 0b to the bit to be cleared and 1b to the other bits.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0850h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DTRANERR1	DTRANERR0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SEQERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
19, 18	-	All 0	R	Reserved The written value should always be 0b.
17	DTRANERR1	0h*1	RW	DMAC Channel 1 Error [Setting condition] When an error occurs on DMAC channel 1 [Clearing condition] When 0b is written to DTRANERR1
16	DTRANERR0	0h*2	RW	DMAC Channel 0 Error [Setting condition] When an error occurs on DMAC channel 0 [Clearing condition] When 0b is written to DTRANERR0
15 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	SEQERR	0h*3	RW	Sequencer error [Setting condition] When a sequencer error occurs [Clearing condition] When 0b is written to SEQERR

Note 1. The initial value is applied at a reset and when the DTRANRST1 bit in DM_CM_RST is 0b.

Note 2. The initial value is applied at a reset and when the DTRANRST0 bit in DM_CM_RST is 0b.

Note 3. The initial value is applied at a reset and when the SEQRST bit in DM_CM_RST is 0b.

6.2.2.2.31 DM_CM_INFO2 Interrupt Mask Register (SDm_DM_CM_INFO2_MASK)

The DM_CM_INFO2 interrupt mask register (DM_CM_INFO2_MASK) enables or disables the DM_CM_INFO2 interrupt. When 0b is set in DM_CM_INFO2_MASK while the corresponding flag in DM_CM_INFO2 is set, an interrupt occurs.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0858h
Initial Value : 0000_0000_FFFF_FFFFh

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DTRANERR1_MASK	DTRANERR0_MASK
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SEQERR_MASK
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
31 to 20	-	All 1	R	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
19, 18	-	All 1	R	Reserved The written value should always be 1b.
17	DTRANERR1_MASK	1h	RW	DTRANERR1 interrupt masked
16	DTRANERR0_MASK	1h	RW	DTRANERR0 interrupt masked
15 to 1	-	All 1	R	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
0	SEQERR_MASK	1h	RW	SEQERR interrupt masked

6.2.2.2.32 DMAC Transfer Address Register (SDm_DM_DTRAN_ADDR)

The DMAC transfer address register (DM_DTRAN_ADDR) sets the transfer destination and source addresses for the DMAC of the module.

Higher-order 32 bits (bit 63 to 32) are read-only and read as 0b.

Access Size : 16, 32, 64 bits
Offset Address : <SDm_base> + 0880h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DADDR[28:13]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DADDR[12:0]													-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 32	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
31 to 3	DADDR[28:0]	0h	RW	Destination address/Source address (8 byte unit) Note that the value of DM_DTRAN_ADDR + transfer data length is less than or equal to 2^{32}.
2 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

6.2.3 Operation

6.2.3.1 SD Interface

6.2.3.1.1 SD data format

When data is read from the SD card, the procedure is as follows.

1. The SD/MMC host interface receives data from the SD card via the SDDAT signal. (SDDAT signal: see **Figure 6.2-2**, **Figure 6.2-3** and **Figure 6.2-5**.)
2. The receive data is stored in SD_BUF of the SD/MMC host interface. (SD_BUF store data: see **Figure 6.2-7**)
3. The data stored in SD_BUF is read from SD_BUF0. (Reading from SD_BUF0: see **Table 6.2-4**)

When data is written to the SD card, the above procedure will be reversed.

When accessing SD_BUF0, caution should be taken for the transfer order in SDDAT and the store order in SD_BUF. In addition, data stored in SD_BUF0 can be replaced in bytes with the EXT_SWAP. (See **Figure 6.2-7**)

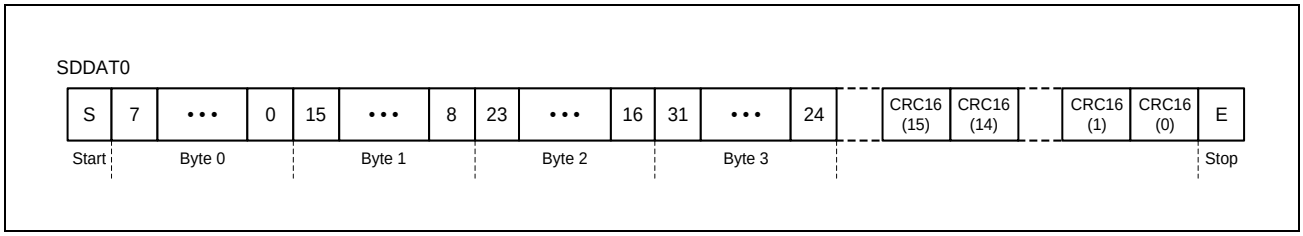


Figure 6.2-2 SDDAT in 1-Bit Width Mode

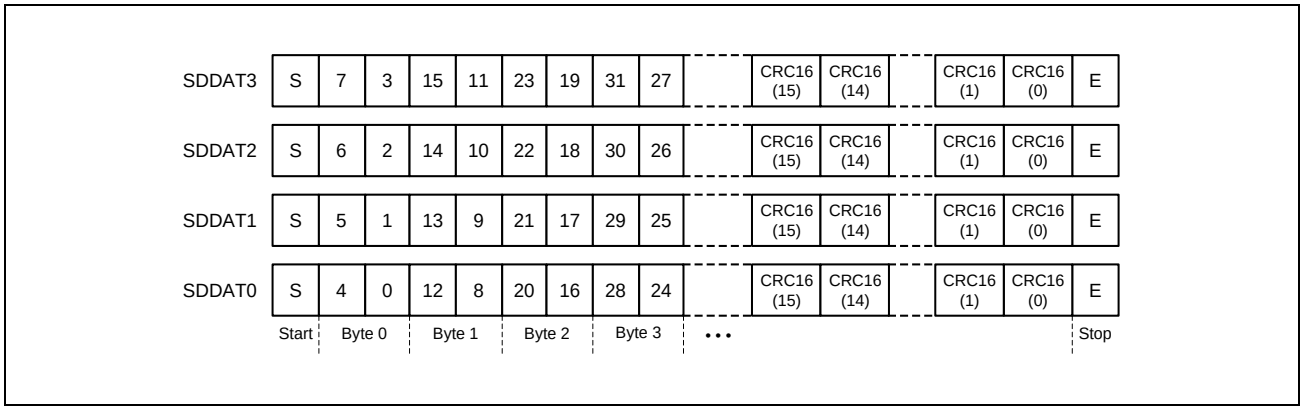


Figure 6.2-3 SDDAT in 4-Bit Width Mode

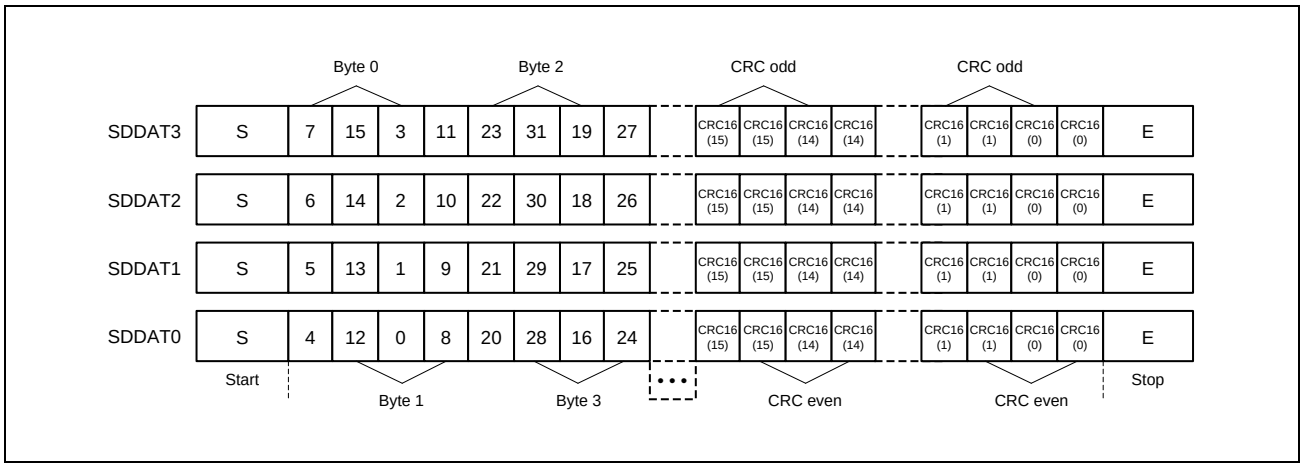


Figure 6.2-4 SDDAT in 4-Bit Width DDR Mode

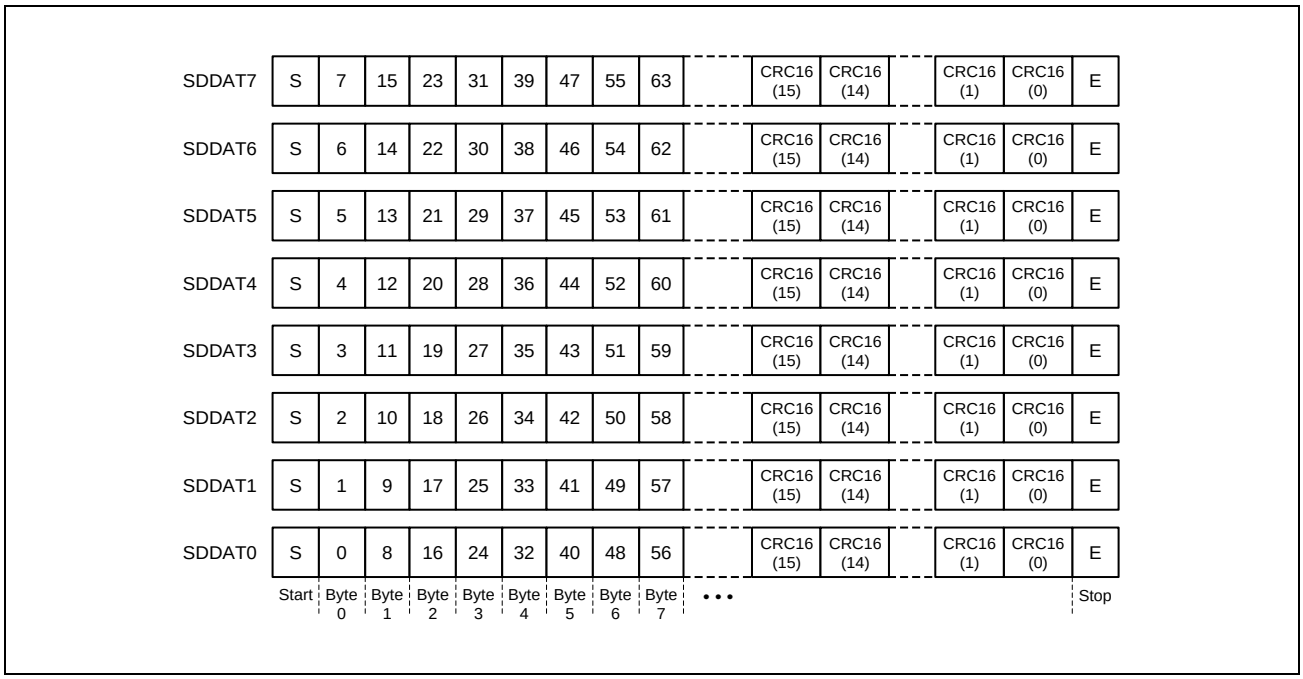


Figure 6.2-5 SDDAT in 8-Bit Width Mode

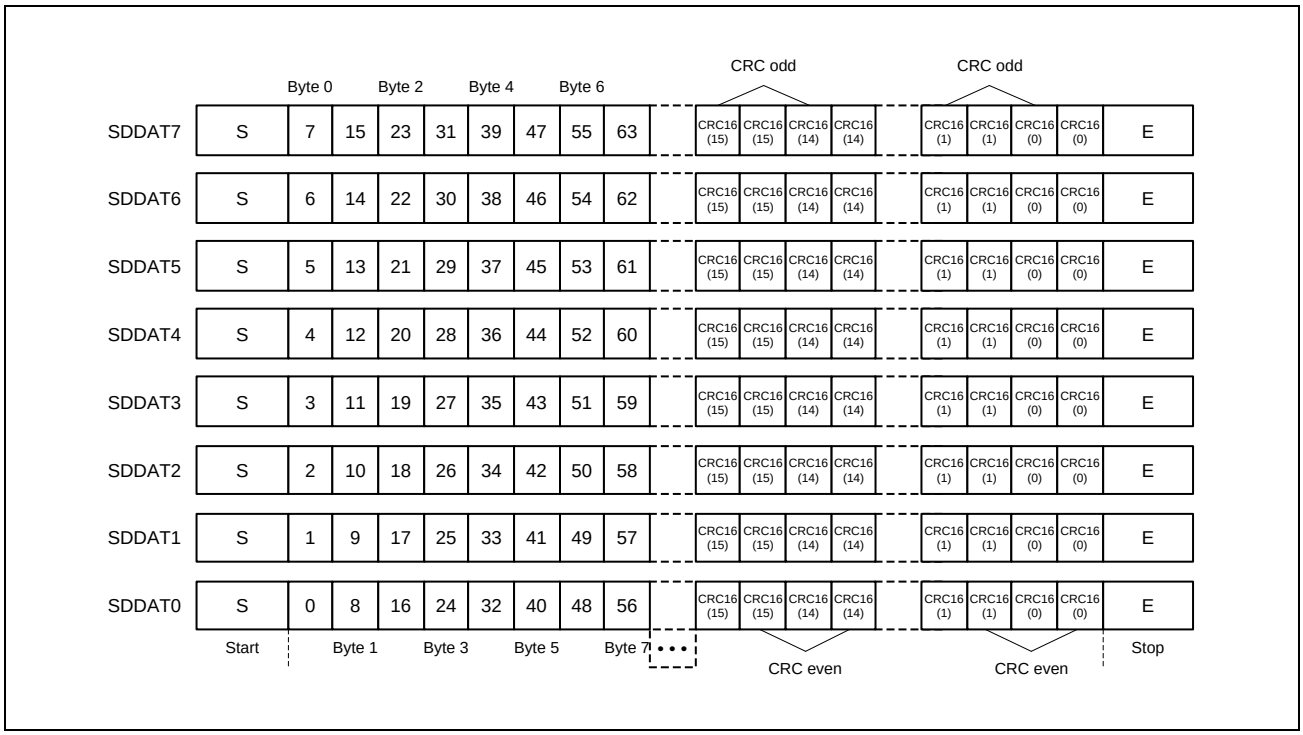


Figure 6.2-6 SDDAT in 8-Bit Width DDR Mode

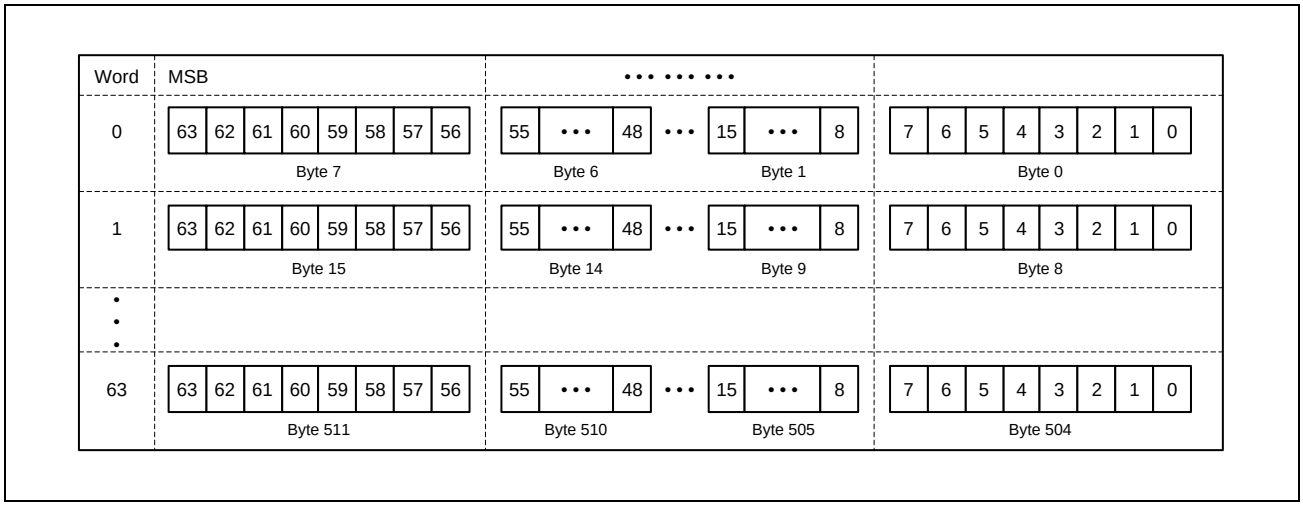


Figure 6.2-7 SD_BUF Store Data

Table 6.2-4 Reading from SD_BUF0

WMODE*1	BUSWIDTH*1	ENDIAN*1	Read Data*2
0	0	0	0123_4567_89AB_CDEFh
0	0	1	EFC_DAB8_9674_52301h
1	1	0	89AB_CDEFh (1st) 0123_4567h (2nd)
1	1	1	EFCD_AB89h (1st) 6745_2301h (2nd)
1	0	0	CDEFh (1st) 89ABh (2nd) 4567h (3rd) 0123h (4th)
1	0	1	EFCDh (1st) AB89h (2nd) 6745h (3rd) 2301h (4th)

Note 1. The name of a bit in HOST_MODE.

Note 2. When the data stored in SD_BUF is 0123_4567_89AB_CDEFh

6.2.3.1.2 Bus signal voltage switch

Change the electric potential of the bus signal in the following procedure after checking that the SD card supports 1.8 V.

1. Issuing CMD11
   Perform command sequence processing of CMD11.

2. Stopping the SD clock (a)
   Set the SCLKEN bit in the SD_CLK_CTRL to 0 to stop*1 the output of the SD clock. When the SDCLKOFFEN bit in the SD_CLK_CTRL register is 1, the SDCLKOFFEN bit is also set to 0b.

Note 1. When the SDCLKOFFEN bit in the SD_CLK_CTRL register is 1, the SD clock has automatically been stopped.

3. Checking the value of SDDAT
   Check that the DAT0 bit in the SD_INFO2 register is 0.

4. Changing the supply voltage of the host device
   Change the voltage which is supplied through the power supply pin of the given channel (VDD1833_SD0 for channel 0, VDD1833_SD1 for channel 1 or VDD1833_SD2 for channel 2) from 3.3 V to 1.8 V.

   For details, see the following sections:

   4.2.2.5 Port Function Control Registers (PFC_mn)
   6.2.2.2.24 SD Status Register (SDm_SD_STATUS)

5. Starting supply of the SD clock (b)
   After the SD clock has been stopped ((a) above) and 5 ms or more has elapsed, set the SCLKEN bit in the SD_CLK_CTRL register to 1 and allow the output of the SD clock. The SDCLKOFFEN bit must be 0.

6. Checking the value of SDDAT
   After supplying the SD clock has been started ((b) above) and 1 ms or more has elapsed, check that the DAT0 bit in the SD_INFO2 register is 1. It is possible to set the SDCLKOFFEN bit in the SD_CLK_CTRL register to 1 and allow SD Clock (SDCLK) Output Automatic Control Enable.

6.2.3.2 Card Detect/Write Protect

6.2.3.2.1 Card detect

The SD/MMC host interface has two types of card detect functions as described in the following.

(1) Card detect with ISDCD

Figure 6.2-8 shows the timing chart of card detect using ISDCD. ISDCD is connected to the card socket and pulled up on the host device. The resistance of the pull-up resistor is decided by the specification of the SD host device.

[Card insertion]

ISDCD is pulled down when a card is inserted. At this time, if ISDCD has been pulled down for the Mcycle period (set in SD_OPTION), INFO4 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

[Card removal]

ISDCD is pulled up when a card is removed. At this time, if ISDCD has been pulled up for the Mcycle period (set in SD_OPTION), INFO3 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

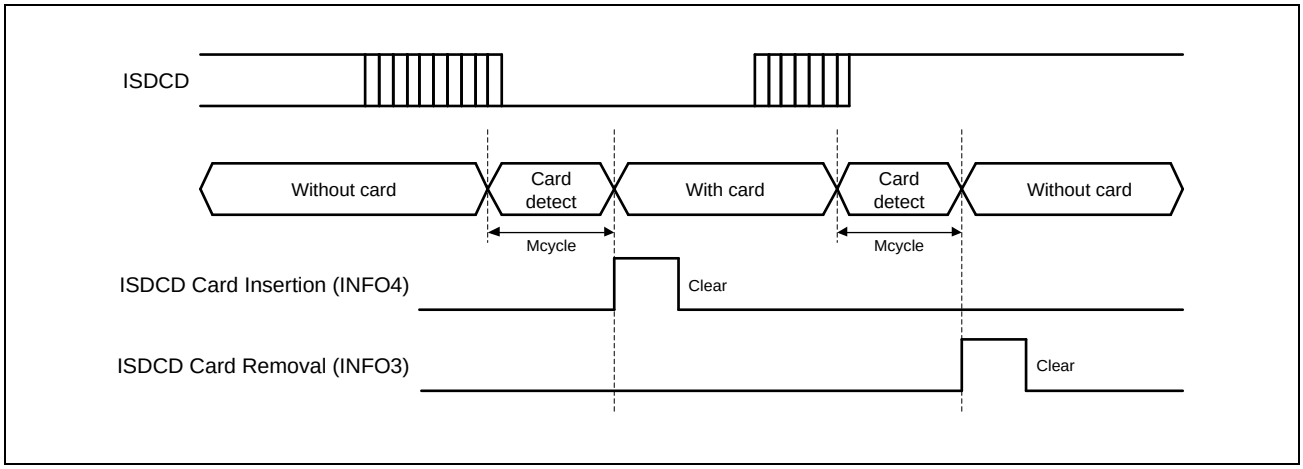


Figure 6.2-8 Example of Card Detect with ISDCD

(2) SD card detect with SDDAT3

Figure 6.2-9 shows the timing chart when the SD card is detected using SDDAT3. In addition, SDDAT3 is pulled down on the host device. The resistance of the pull-down resistor is decided by the specification of the SD host device.

[Card insertion]

When an SD card is inserted, SDDAT3 is pulled up. Accordingly, INFO9 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

[Card removal]

When an SD card is removed, SDDAT3 is pulled down. Accordingly, INFO8 in SD_INFO1 is set to 1. (It is cleared to by writing 0.)

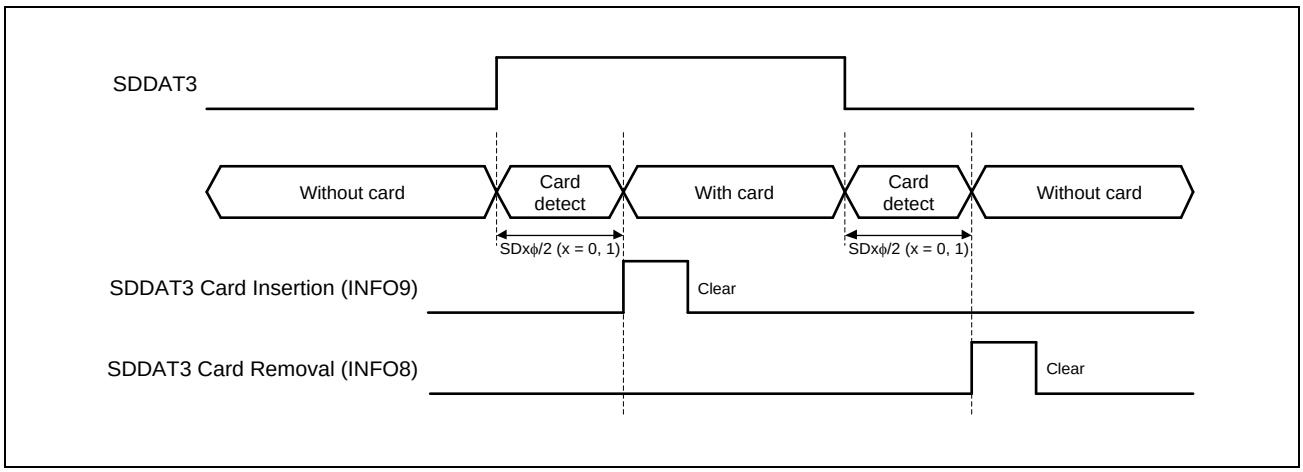


Figure 6.2-9 SD Card Detect with SDDAT3

6.2.3.2.2 Write protect

The SD/MMC host interface has two types of write protect functions.

(1) Write protect with ISDWP

ISDWP is connected to the card socket, and pulled up or pulled down by the card insertion. The selection of pulling up or pulling down and the resistance value is decided by the specifications of the SD host device. As the ISDWP state is reflected to INFO7 in SD_INFO1, the write protect is decided after the SD card is inserted.

(2) Write protect with command

The card’s internal write protection and the card lock/unlock operation are realized by the command.

6.2.3.3 Interrupt Request

6.2.3.3.1 Interrupt request

The SD/MMC host interface has the interrupt requests shown in **Table 6.2-5** shows the relationship between the interrupt flag registers and the interrupt mask registers. When a bit in an interrupt mask register is set to 0, an interrupt occurs by setting the corresponding bit in the interrupt flag register to 1.

To clear a flag, write 0b to the bit to be cleared and 1b to the other bits.

Table 6.2-5 Interrupt Request

	Interrupt Flag Register		Interrupt Mask Register	
Interrupt Request	Register Name	Bit Name	Register Name	Bit Name
Card access interrupt*1	SD_INFO1	INFO2	SD_INFO1_MASK	IMASK2
		INFO0		IMASK0
	SD_INFO2	ILA	SD_INFO2_MASK	IMASK
		BWE		BMASK1
		BRE		BMASK0
		ERR6		EMASK6
		ERR5		EMASK5
		ERR4		EMASK4
		ERR3		EMASK3
		ERR2		EMASK2
		ERR1		EMASK1
		ERR0		EMASK0
SDIO access interrupt*2	SDIO_INFO1	EXWT	SDIO_INFO1_MASK	MEXWT
		EXPUB52		MEXPUB52
		IOIRQ		IOMSK
Card detect interrupt*1	SD_INFO1	INFO9	SD_INFO1_MASK	IMASK9
		INFO8		IMASK8
		INFO4		IMASK4
		INFO3		IMASK3
DMAC interrupt*1	DM_CM_INFO1	DTRANEND1	DM_CM_INFO1_MASK	DTRANEND1_MASK
		DTRANEND0		DTRANEND0_MASK
		SEQEND		SEQEND_MASK
	DM_CM_INFO2	DTRANERR1	DM_CM_INFO2_MASK	DTARERR1_MASK
		DTRANERR0		DTARERR0_MASK
		SEQERR		SEQERR_MASK

Note 1. Interrupt signal “OXMNIRQ” is asserted.

Note 2. Interrupt signal both “OXMNIRQ” and “OXASIOIRQ” are asserted.

6.2.3.4 Communications Errors and Timeouts

1) Communications errors and timeouts

Table 6.2-6 and **Table 6.2-7** show the relationships between the SD card interrupt flag register and SD error status register for communications errors and timeouts, respectively. When a bit in the SD card interrupt flag register is set to 1, the corresponding bit in the SD error status register is set to 1. The values of the SD error status register are cleared by writing to SD_CMD or writing 0 to the SDRST bit in SOFT_RST.

Table 6.2-6 Communications Errors

Communication Error	SD Interrupt Flag Register		SD Error Status Register		Description
	Register Name	Bit Name	Register Name	Bit Name	
END error	SD_INFO2	ERR2	SD_ERR_STS1	E5	When an error occurs in the CRC status length
				E4	When an error occurs in read data length
				E3	When an error occurs in the response length to a command issued within a command sequence
				E2	When an error occurs in the response length (other than a response to a command issued within a command sequence)
CRC error		ERR1		E11	When an error occurs in the CRC status
				E10	When a CRC error occurs in the read data
				E9	When a CRC error occurs in the response to a command issued within a command sequence
				E8	When a CRC error occurs in the response (other than a response to a command issued within a command sequence)
CMD error		ERR0		E1	The command index of the transmitted command differed from the command index of the received response (for a command issued within a command sequence)
				E0	The command index of the transmitted command differed from the command index of the received response (for a command issued other than within a command sequence)

Table 6.2-7 Timeouts

Timeout	SD Interrupt Flag Register		SD Error Status Register		Description
	Register Name	Bit Name	Register Name	Bit Name	
Response timeout	SD_INFO2	ERR6	SD_ERR_STS2	E1	When the response to a command issued within a command sequence is not received even after 640 cycles of SDCLK have elapsed
				E0	When the response (other than a response to a command issued within a command sequence) is not received even after 640 cycles of SDCLK have elapsed
Data timeout (other than response timeout)		ERR3		E6	When the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle*1 after the CRC status
				E5	When the CRC status is not received even after Ncycle*1 has elapsed after data writing
				E4	When read data are not received even after Ncycle*1 has elapsed after read command
					When read data for the next block are not received even after Ncycle*1 has elapsed after the reception of read data
					When read data for the next block are not received even after Ncycle*1 has elapsed after release of the read wait state
				E3	When the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle*1 after CMD12 has been issued within a command sequence
				E2	When the interface remains in a busy state (SDDAT0 = 0) for at least Ncycle*1 after R1b response

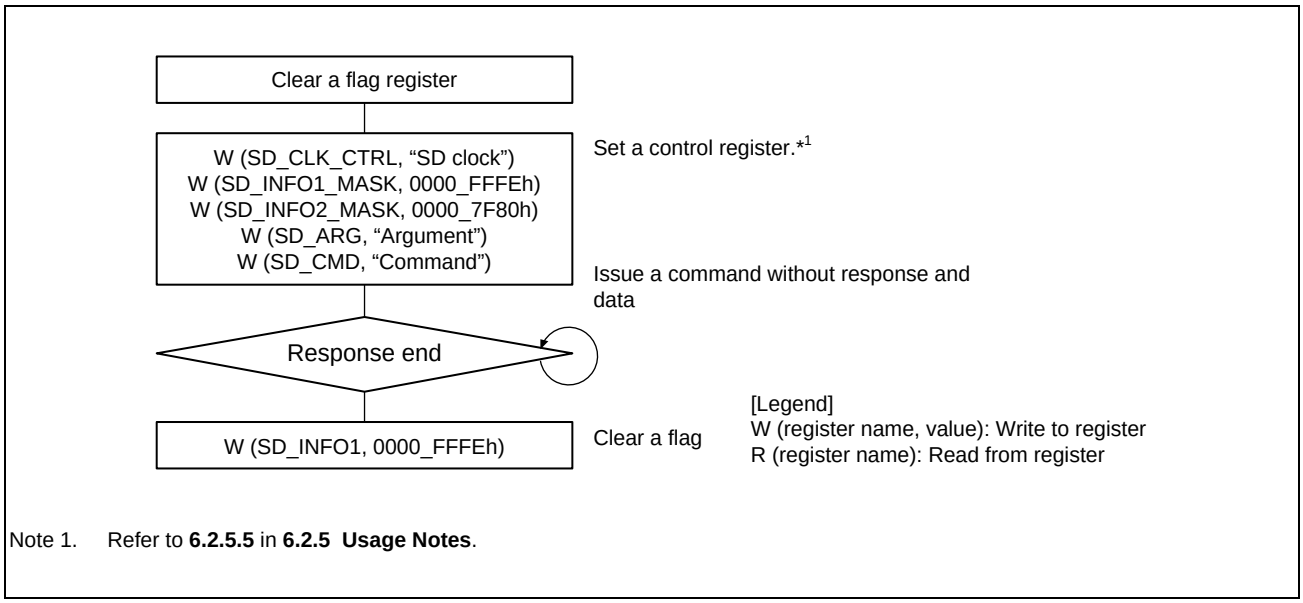
Note 1. Ncycle is set by bit 7 to bit 4 in SD_OPTION.

6.2.4 Usage Example

6.2.4.1 Command without Data Transfer

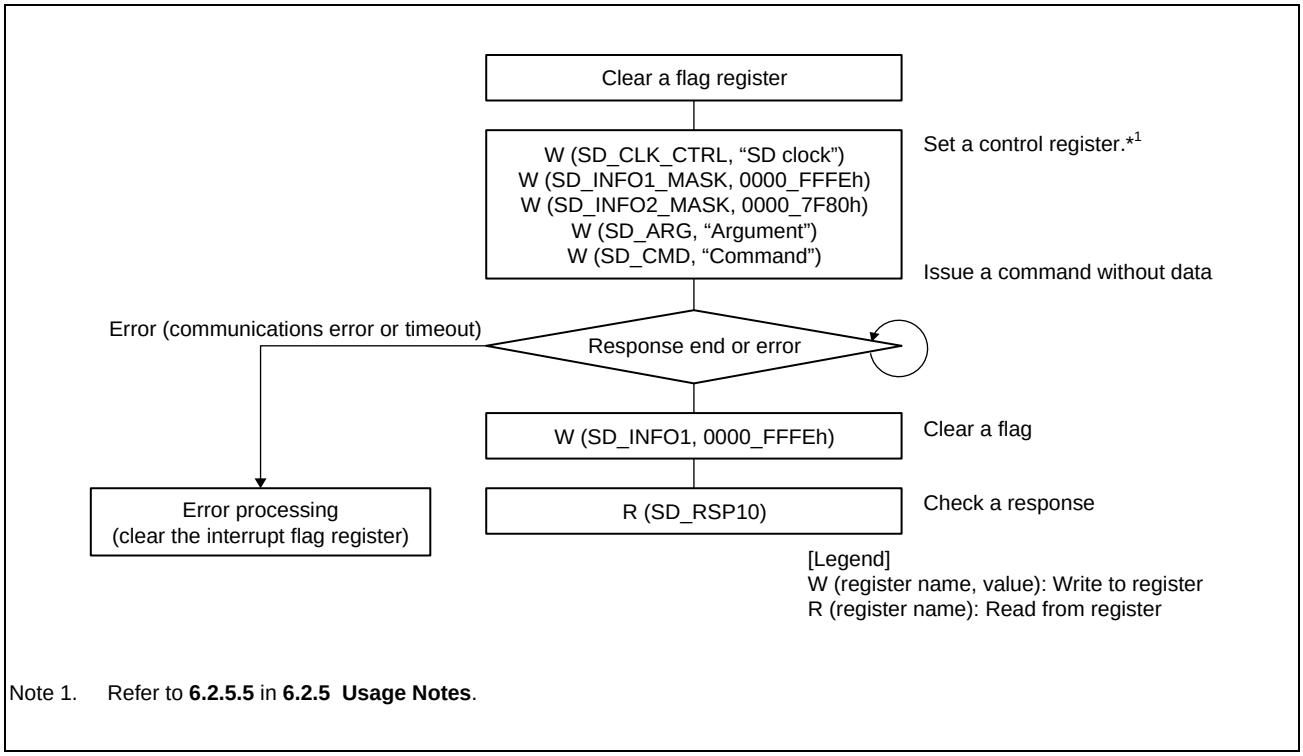
6.2.4.1.1 Flowchart

Figure 6.2-10 and **Figure 6.2-11** show flowchart examples.



Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-10 Flow Example of Command without Response and Data



Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-11 Flow Example of Command without Data

6.2.4.1.2 Operation for command without data transfer

The legend below is used for description of register read/write.

— W (register name, value): Write to register

— R (register name): Read from register

The operation is described as below.

(1) Command without response and data

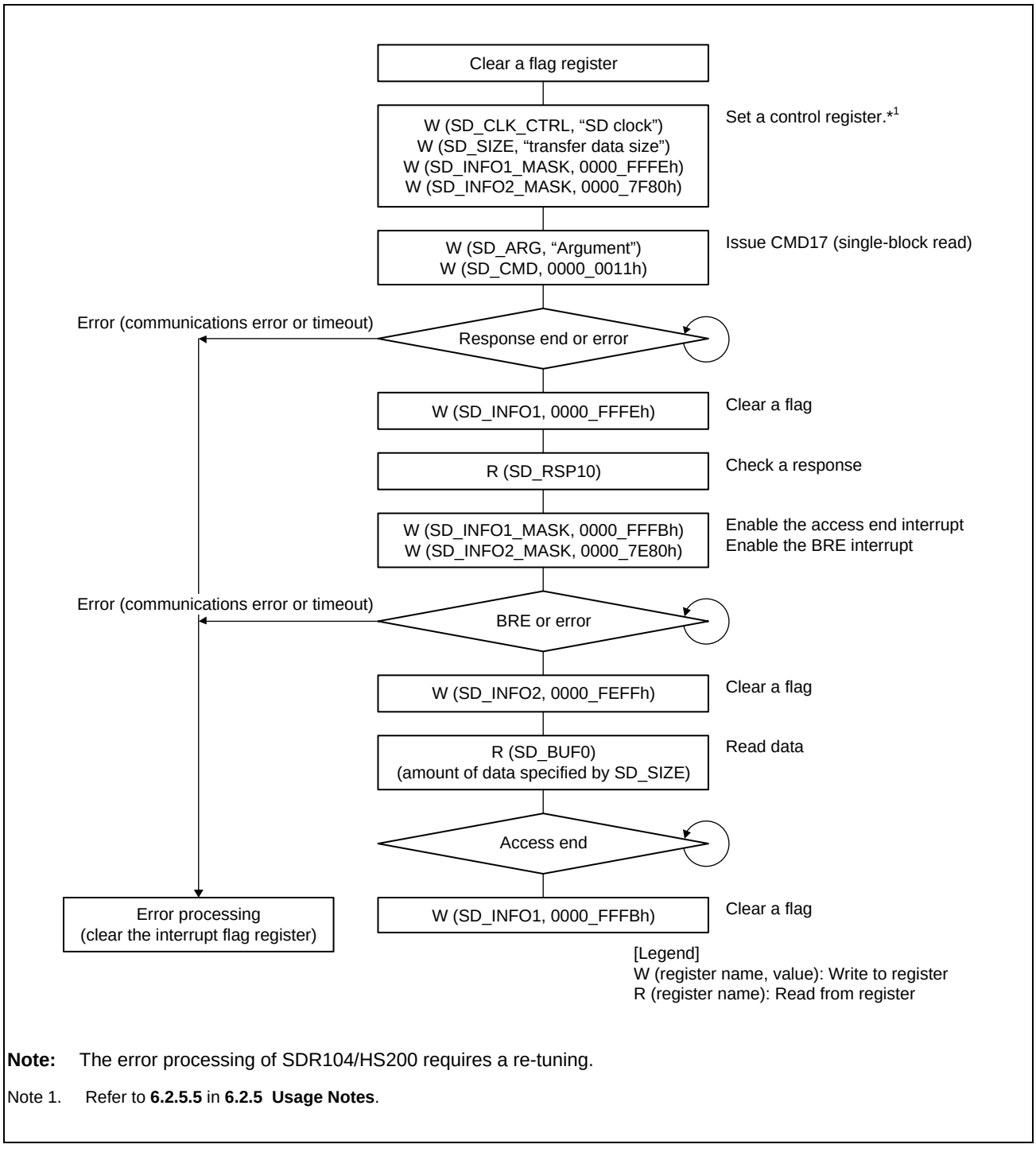
1. Flag register clear
   First, clear the bits in the flag registers. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), interrupt mask, and so on. (SD_CLK_CTRL, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue
   Set CMD Argument in SD_ARG and write to SD_CMD.
   Accordingly, CMD is issued, and the operation is started.
4. Flag clear
   When transmission of a command is completed, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0.

(2) Command without data

1. Flag register clear
   First, clear the bits in the flag registers. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), interrupt mask, and so on. (SD_CLK_CTRL, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue
   Set CMD Argument in SD_ARG and write to SD_CMD.
   Accordingly, CMD is issued, and the operation is started.
4. Flag clear
   When a response is received, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0.
5. Read a response from SD_RSP10.
   Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

6.2.4.2 Single Block Read

6.2.4.2.1 Flowchart



Note: The error processing of SDR104/HS200 requires a re-tuning.

Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-12 Single Block Read Flowchart Example

6.2.4.2.2 Operation for single block read

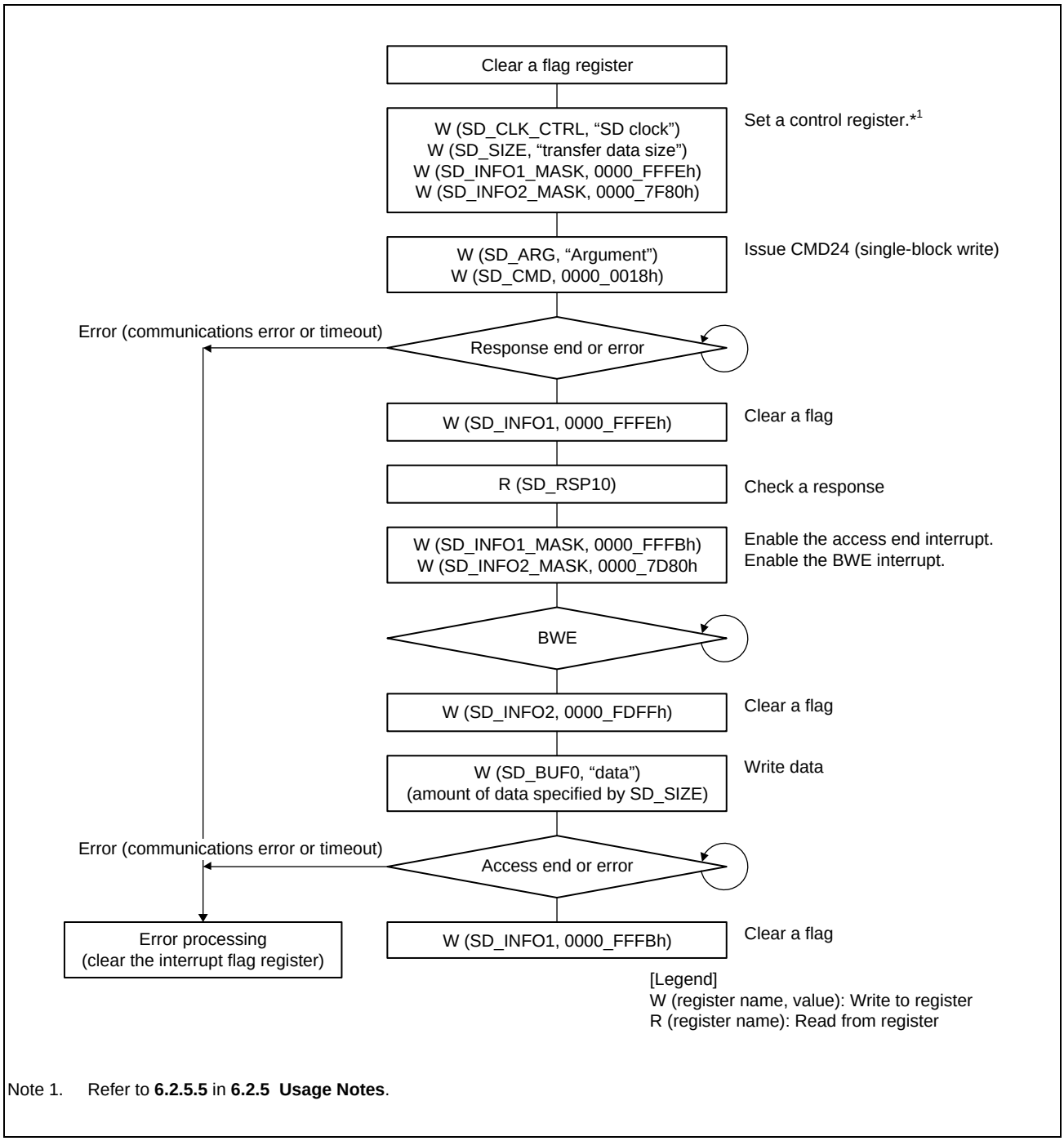
The operation of single block read is described below.

1. Flag register clear
   First, clear the bits in the flag registers. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue (CMD17)
   Set CMD17 Argument in SD_ARG and write 0000_0011h to SD_CMD.
   Accordingly, CMD17 is issued, and the single block read operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP10.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP or the IOABT bit in SDIO_MODE to 1. Furthermore, this causes CMD12 and CMD52 to not be issued.
   If the command sequence is halted, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt.
5. Data receive from SD card and data read
   Write 0000_FFFBh to SD_INFO1_MASK to enable the access end interrupt. In addition, write 0000_7E80h to SD_INFO2_MASK to enable the BRE interrupt. When the data receive from the SD card is completed, the BRE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BRE bit to 0 and read the amount of data specified by SD_SIZE from SD_BUF0.
   However, a communications error or timeout may be generated if data are being received while reading of SD_BUF0 is in progress.
6. Operation complete
   When the data read from SD_BUF0 is completed, INFO2 (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO2 to 0 to end the single block read operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

6.2.4.3 Single Block Write

6.2.4.3.1 Flowchart



Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-13 Single Block Write Flowchart Example

6.2.4.3.2 Operation for single block write

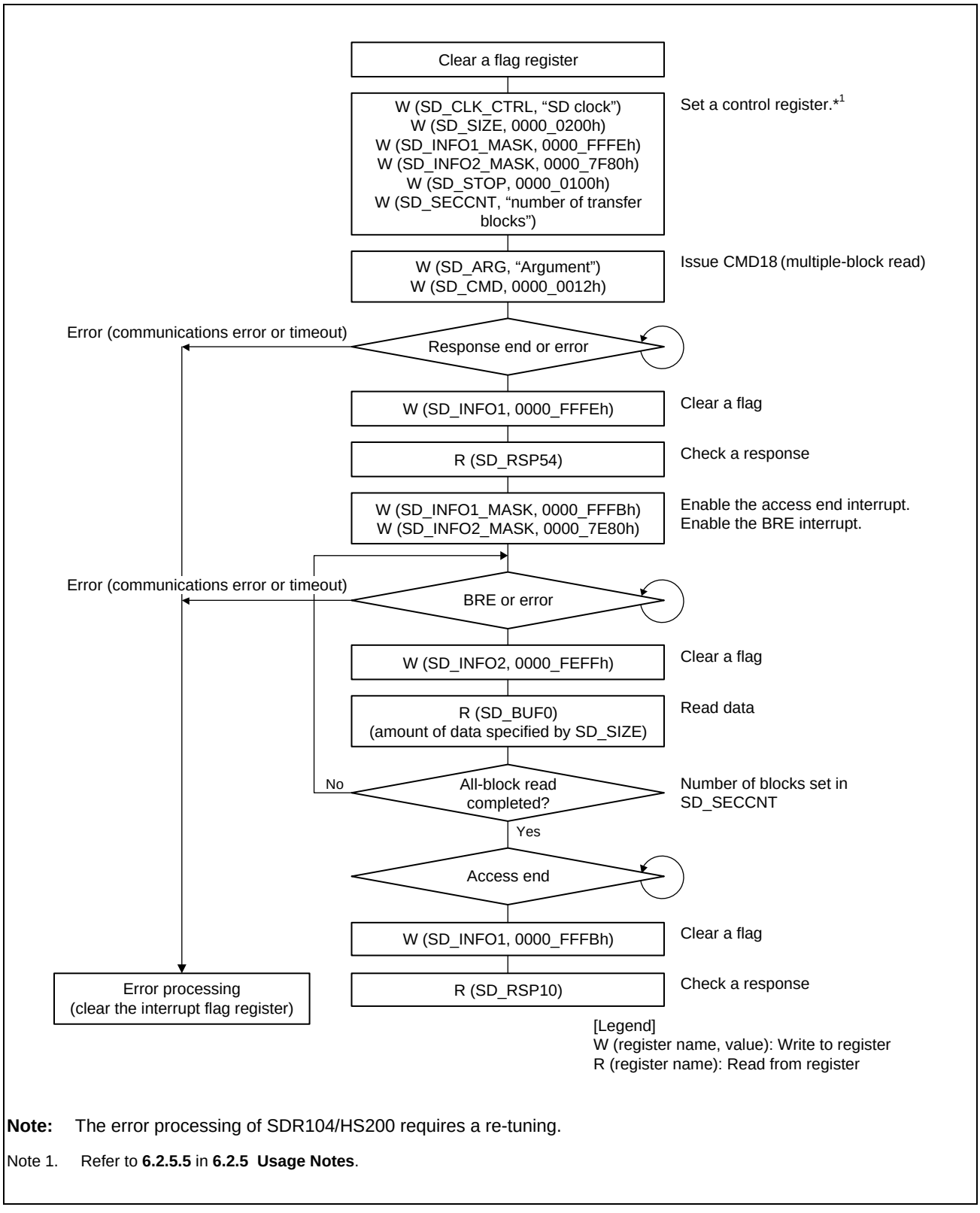
The operation of single block write is described below.

1. Flag register clear
   First, clear the bits in the flag registers. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue (CMD24)
   Set CMD24 Argument in SD_ARG and write 0000_0018h to SD_CMD.
   Accordingly, CMD24 is issued, and the single block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP10.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP or the IOABT bit in SDIO_MODE to 1. Furthermore, this causes CMD12 and CMD52 to not be issued.
   If the command sequence is halted, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt.
5. Data write and data transmit to SD card
   Write 0000_FFFBh to SD_INFO1_MASK to enable the access end interrupt. In addition, write 0000_7D80h to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card.
   However, a communications error or timeout may be generated if data are being transmitted after writing to SD_BUF0.
6. Operation complete
   When the CRC status and busy state are received from the SD card, INFO2 (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO2 bit to 0 to end the single block write operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

6.2.4.4 Multiple Block Read

6.2.4.4.1 Flowchart



Note: The error processing of SDR104/HS200 requires a re-tuning.

Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-14 Multiple Block Read Flowchart Example

6.2.4.4.2 Operation for multiple block read

The operation of multiple block read is described below.

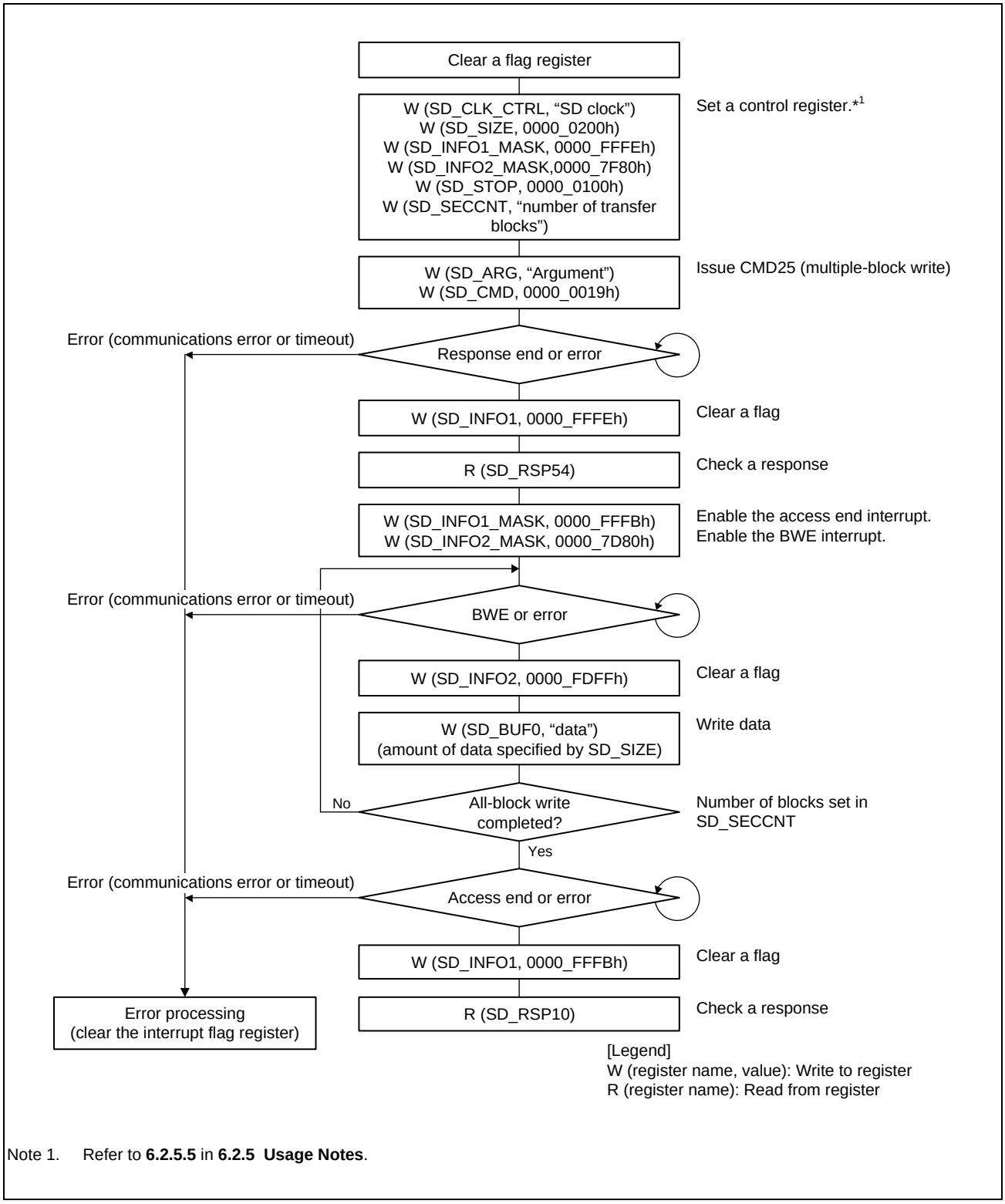
1. Flag register clear
   First, clear the bits in the flag registers. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD18)
   Set CMD18 Argument in SD_ARG and write 0000_0012h to SD_CMD.
   Accordingly, CMD18 is issued, and the multiple block read operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting the INFO2 bit (access end) in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data receive from SD card and data read
   Write 0000_FFFBh to SD_INFO1_MASK to enable the access end interrupt. In addition, write 0000_7E80h to SD_INFO2_MASK to enable the BRE interrupt. When one-block data receive from the SD card is completed, the BRE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BRE bit to 0 and read the amount of data specified by SD_SIZE from SD_BUF0. Doing this repeats transfer of the number of blocks set in SD_SECCNT. However, a communications error or timeout may be generated if data are being received while reading of SD_BUF0 is in progress. CMD12 is automatically issued to stop multi-block transfer with the number of blocks which is set to SD_SECCNT and the response is received. At this point, CMD12 Argument is automatically set to 000_0000h.
6. Operation complete
   When all-block data read and the CMD12 response receive are completed, INFO2 (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO2 to 0 to read the response. This is the end of multiple block read operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

6.2.4.5 Multiple Block Write (when Using Internal Timer)

6.2.4.5.1 Flowchart

Figure 6.2-15 shows the flowchart when using an internal timer.



Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-15 Multiple Block Write Flowchart Example (when Using Internal Timer)

6.2.4.5.2 Operation for multiple block write

The operation of multiple block write is described below.

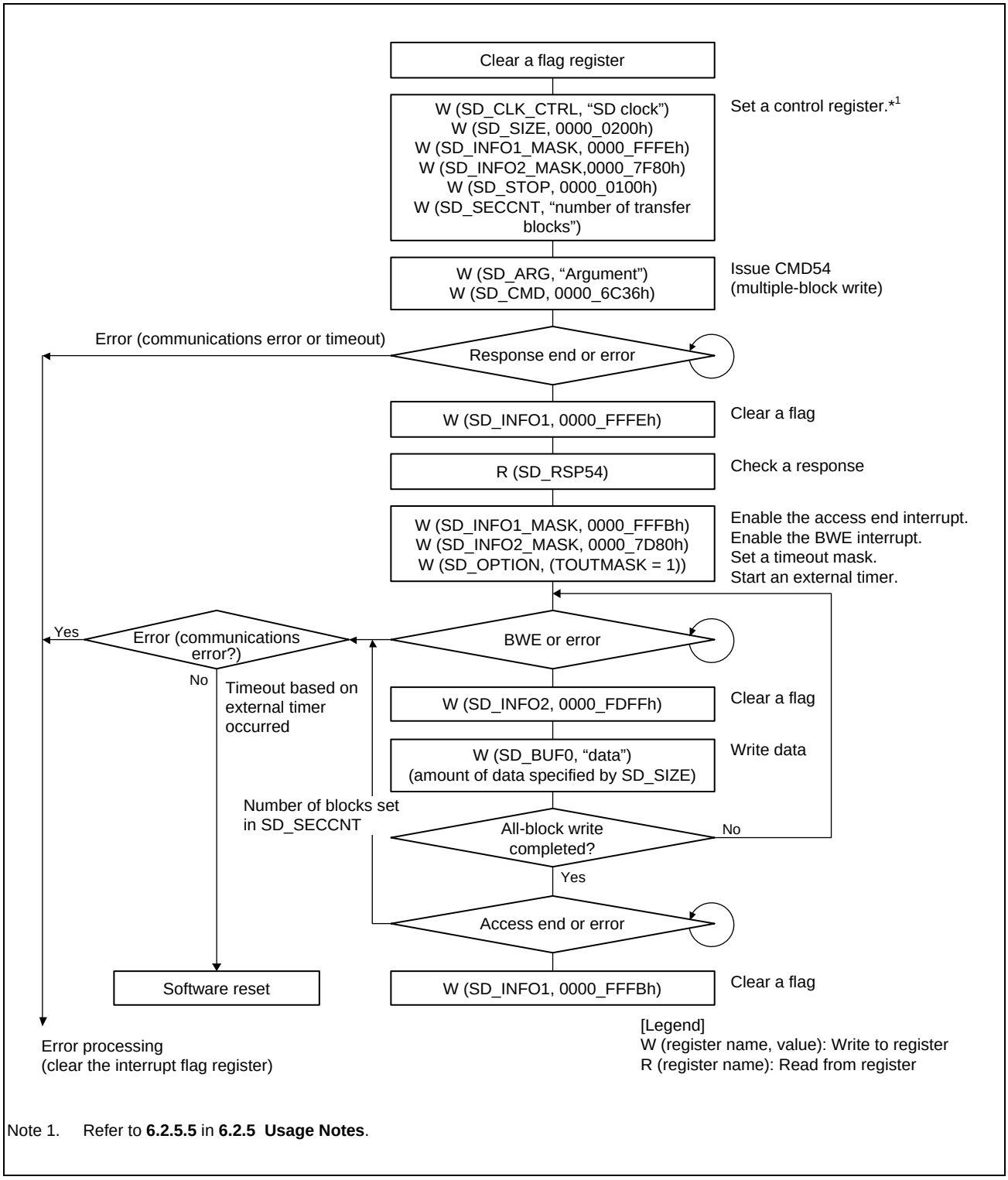
1. Flag register clear
   First, clear the bits in the flag registers. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD25)
   Set CMD25 Argument in SD_ARG and write 0000_0019h to SD_CMD.
   Accordingly, CMD25 is issued, and the multiple block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting the INFO2 bit (access end) in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card
   Write 0000_FFFBh to SD_INFO1_MASK to enable the access end interrupt. In addition, write 0000_7D80h to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card. Doing this repeats transfer of the number of blocks set in SD_SECCNT.
   However, a communications error or timeout may be generated if data are being transmitted while writing to SD_BUF0 is in progress. CMD12 is automatically issued to stop multi-block transfer with the number of blocks which is set to SD_SECCNT and the response is received. At this point, CMD12 Argument is automatically set to 0000_0000h.
6. Operation complete
   When all-block data transmit and the CRC status receive are completed, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO2 bit to 0 to read the response. This is the end of multiple block write operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

6.2.4.6 Multiple Block Write (when Using External Timer)

6.2.4.6.1 Flowchart

The flowchart when using an external timer instead of an internal timer of this module is shown below.



Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-16 Multiple Block Write Flowchart Example (when Using External Timer)

6.2.4.6.2 Operation for multiple block write

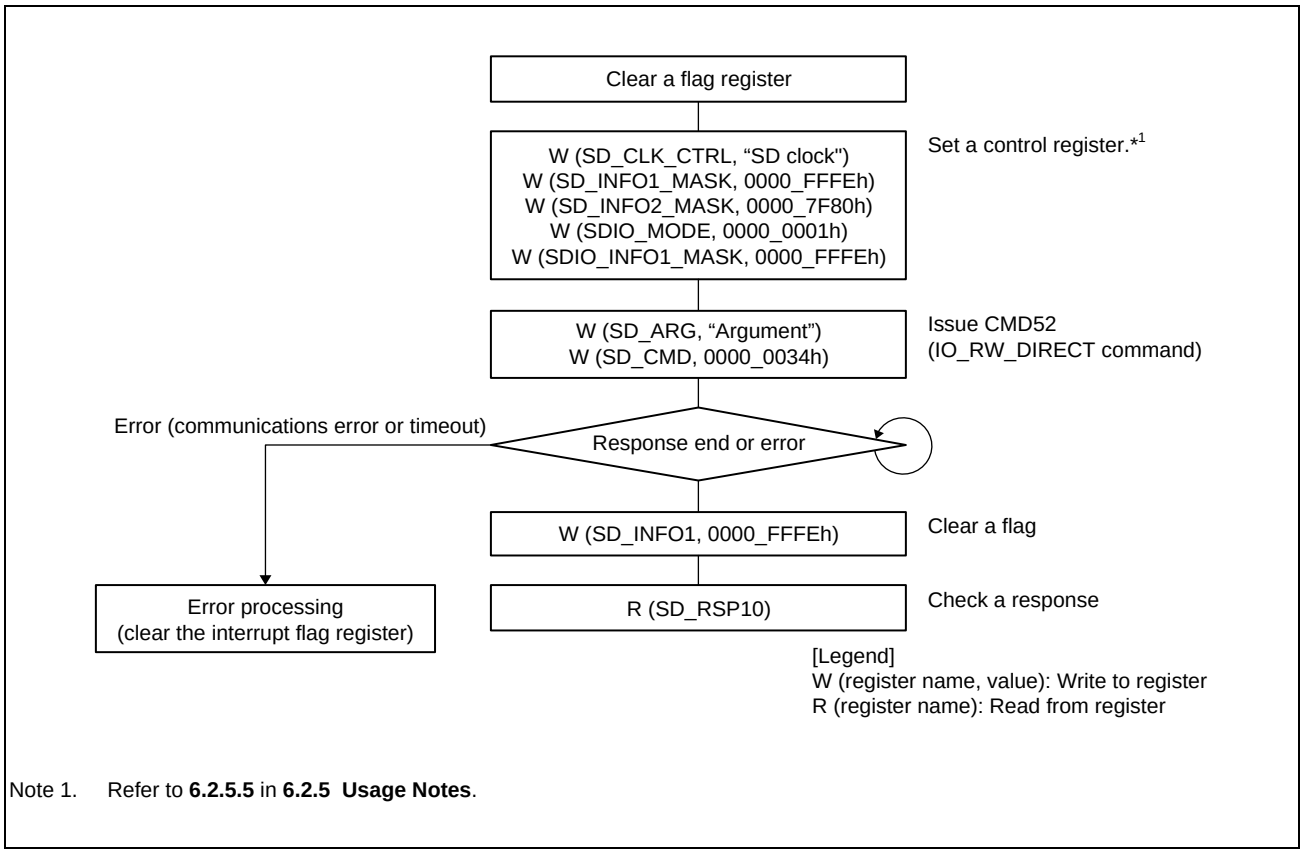
The operation of multiple block write is described below.

1. Flag register clear
   First, clear the bits in the flag registers. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD54)
   Set CMD54 Argument in SD_ARG and write 0000_6C36h to SD_CMD.
   Accordingly, CMD54 is issued, and the multiple block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting the INFO2 bit (access end) in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card
   Write 0000_FFFBh to SD_INFO1_MASK to enable the access end interrupt. In addition, write 0000_7D80h to SD_INFO2_MASK to enable the BWE interrupt. Set the TOUTMASK bit in SD_OPTION to disable timeout and start an external timer.
   When SD_BUF0 is ready for the data to be written, the BWE bit in the SD_INFO2 register is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card. Doing this repeats transfer of the number of blocks set in SD_SECCNT. However, a communications error may be generated if data are being transmitted while writing to SD_BUF0 is in progress.
6. Operation complete
   When all-block data transmit and the CRC status receive are completed, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO2 bit to 0 to read the response. This is the end of multiple block write operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs at response reception (a communications error or timeout) or at data transmission. Perform a software reset if a timeout occurs at data transmission based on an external timer.

6.2.4.7 IO_RW_DIRECT Command (CMD52)

6.2.4.7.1 Flowchart



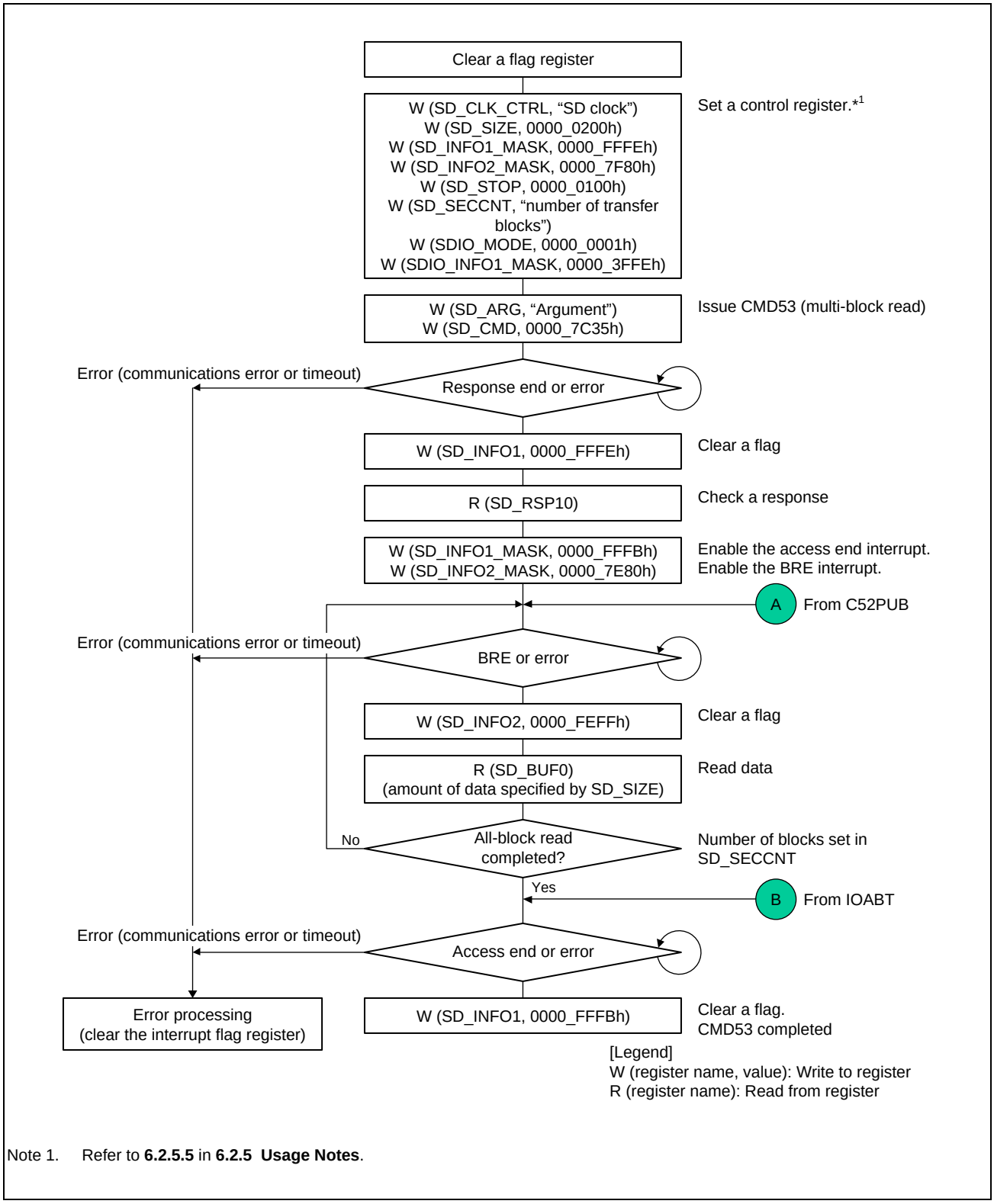
Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-17 IO_RW_DIRECT Command (CMD52) Flowchart Example

6.2.4.8 IO_RW_EXTENDED (CMD53/Multiple Block Read)

6.2.4.8.1 Flowchart

Figure 6.2-18 shows a flowchart example for CMD53 (multiple block read).



Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-18 CMD53 (Multiple Block Read) Flowchart Example

Figure 6.2-19 shows a flowchart example when CMD52 (SDIO abort) is issued at CMD53 (multiple block read).

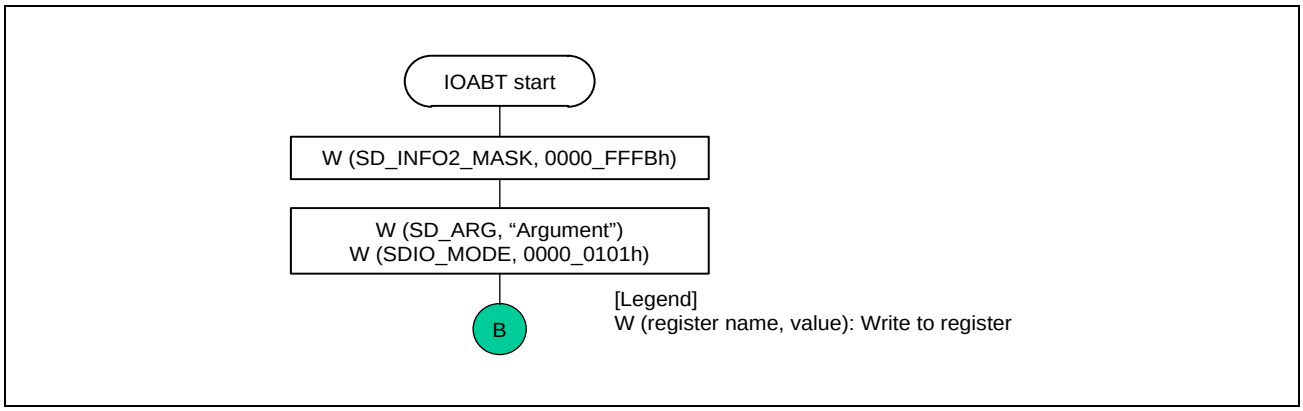


Figure 6.2-19 Flowchart Example when CMD52 (SDIO Abort) is Issued at CMD53 (Multiple Block Read)

Figure 6.2-20 shows a flowchart example when CMD52 (SDIO none abort) is issued at CMD53 (multiple block read) while the SD/MMC host interface is in the read wait state.

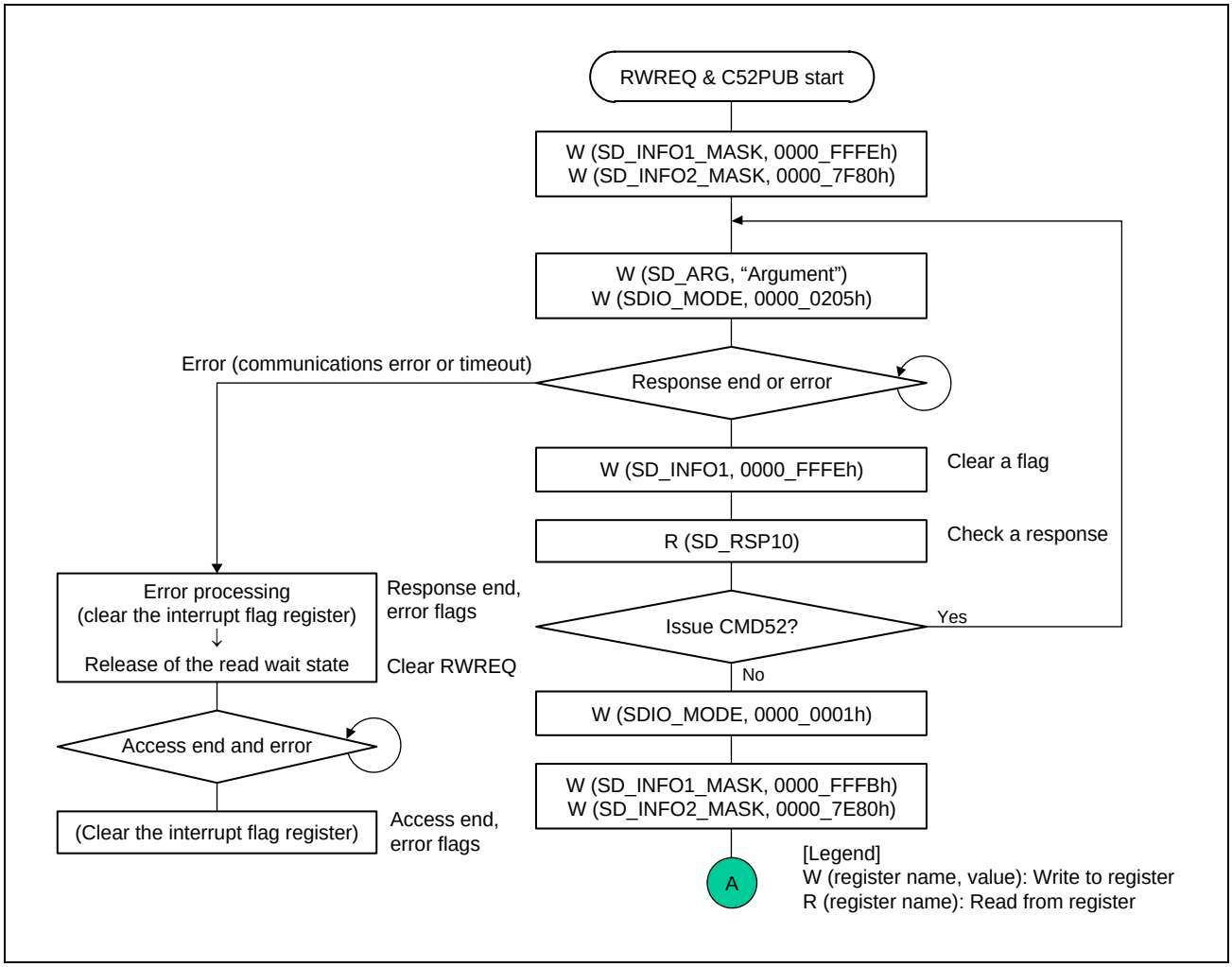
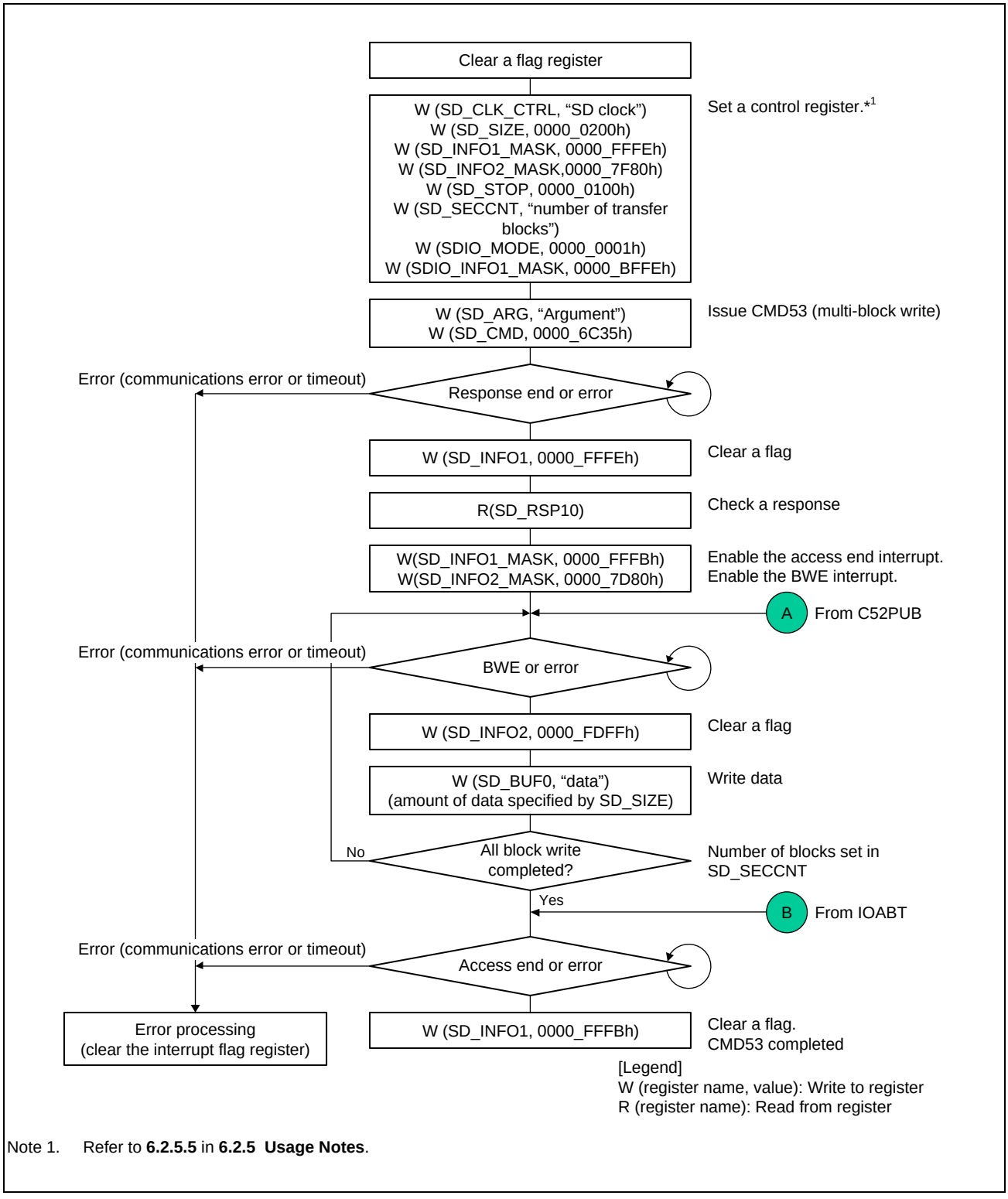


Figure 6.2-20 Flowchart Example when CMD52 (SDIO None Abort) is Issued after Read Wait State is Entered at CMD53 (Multi Block Read)

6.2.4.9 IO_RW_EXTENDED (CMD53/Multiple Block Write)

6.2.4.9.1 Flowchart

Figure 6.2-21 shows a flowchart example for CMD53 (multiple block write).



Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-21 CMD53 (Multiple Block Write) Flowchart Example

Figure 6.2-22 shows a flowchart example when CMD52 (SDIO abort) is issued at CMD53 (multiple block write).

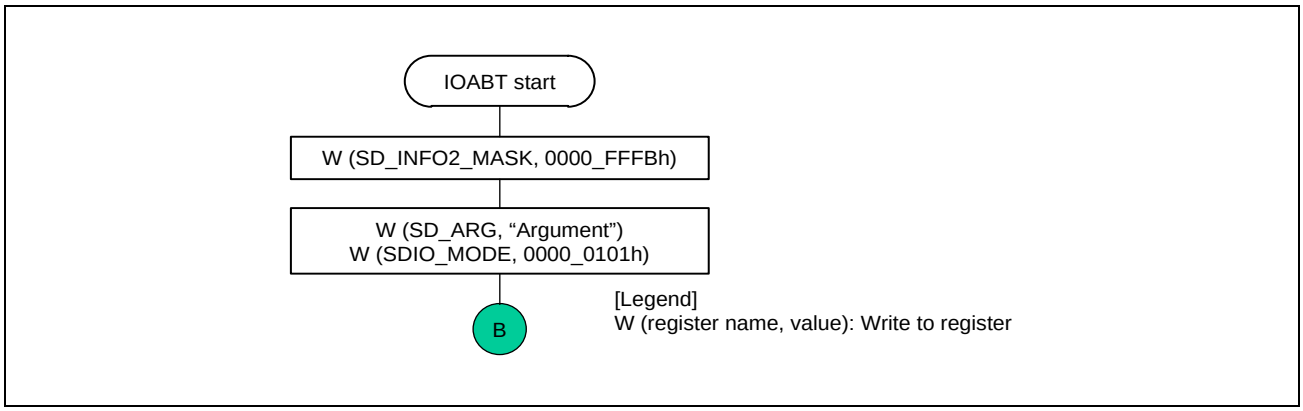


Figure 6.2-22 Flowchart Example when CMD52 (SDIO Abort) is Issued at CMD53 (Multiple Block Write)

Figure 6.2-23 shows a flowchart example when CMD52 (SDIO none abort) is issued at CMD53 (multiple block write).

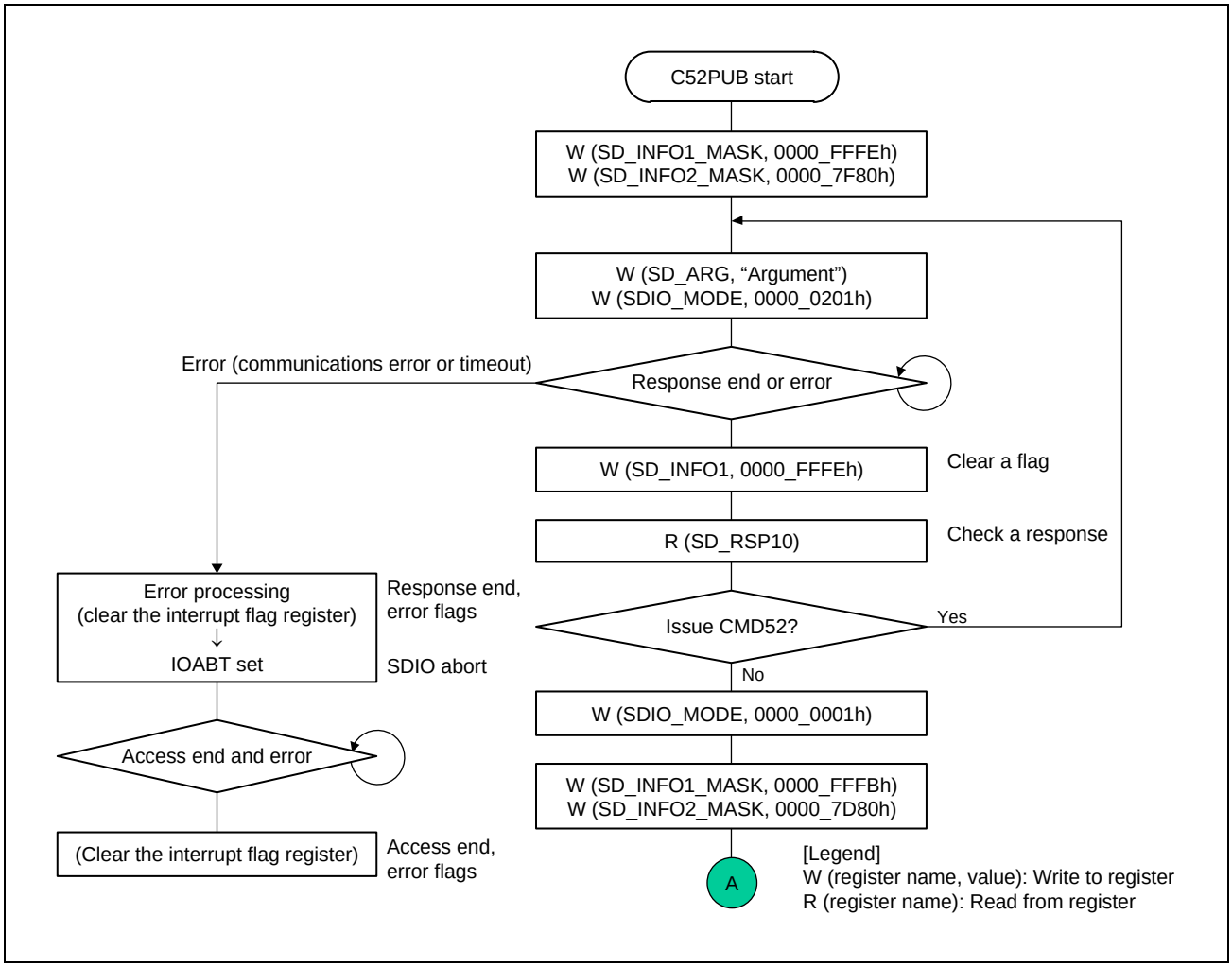
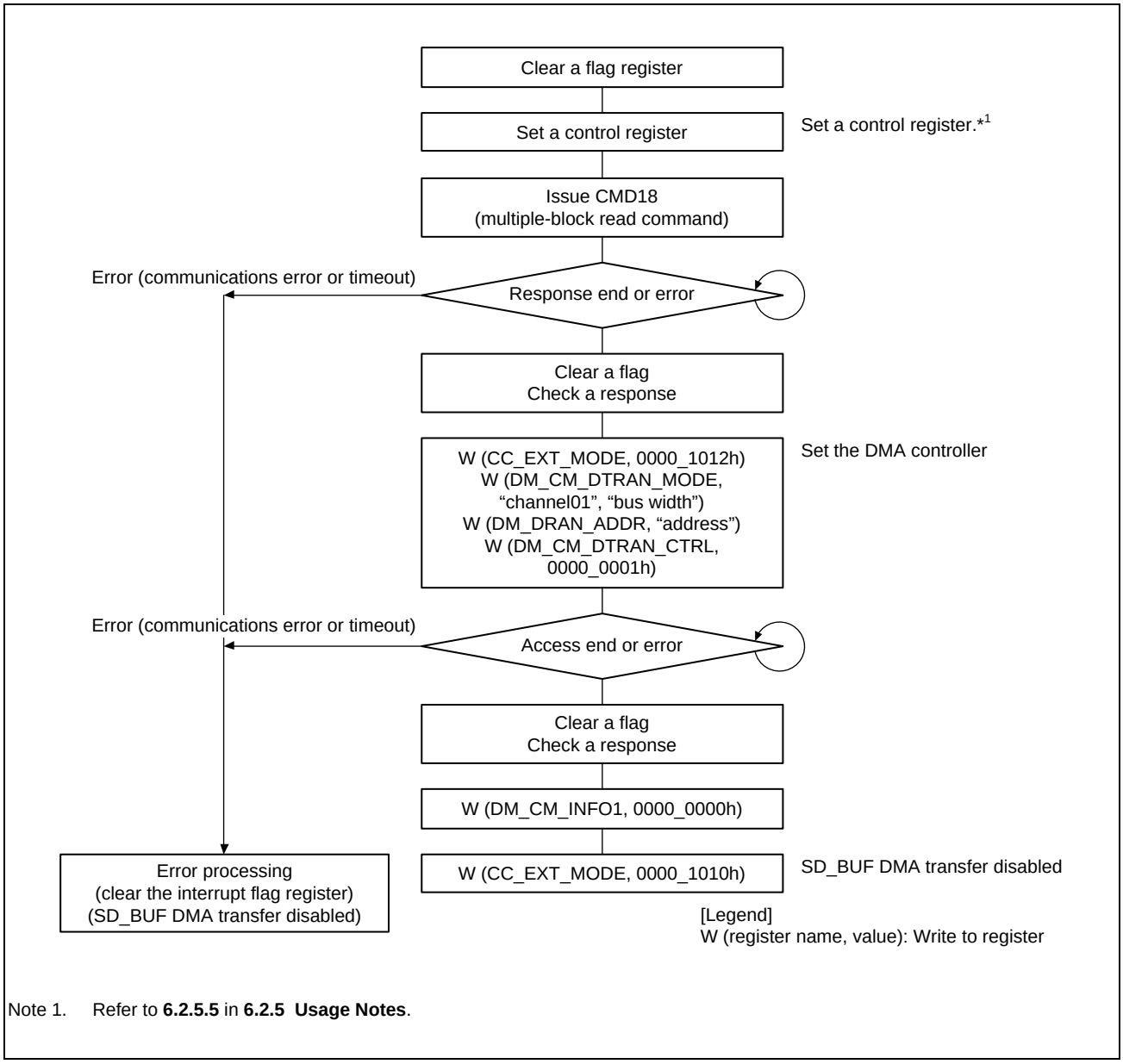


Figure 6.2-23 Flowchart Example when CMD52 (SDIO None Abort) is Issued at CMD53 (Multiple Block Write)

6.2.4.10 DMA Transfer

6.2.4.10.1 SD_BUF DMA transfer

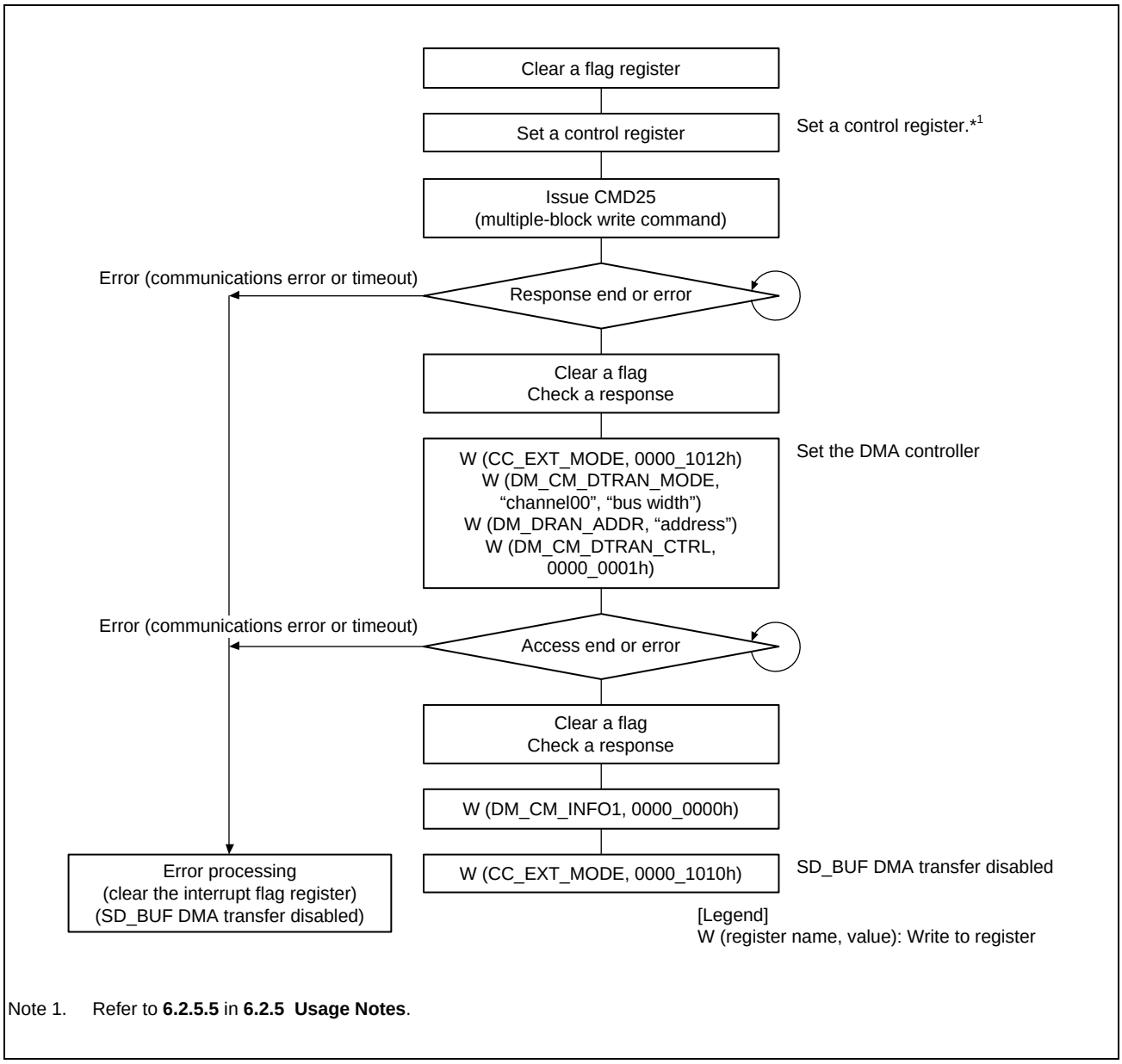
Figure 6.2-24 shows a flowchart example for SD_BUF DMA read when CMD18 (multiple block read) is issued.



Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-24 SD_BUF DMA Read Flowchart Example

Figure 6.2-25 shows a flowchart example for SD_BUF DMA write when CMD25 (multiple block write) is issued.

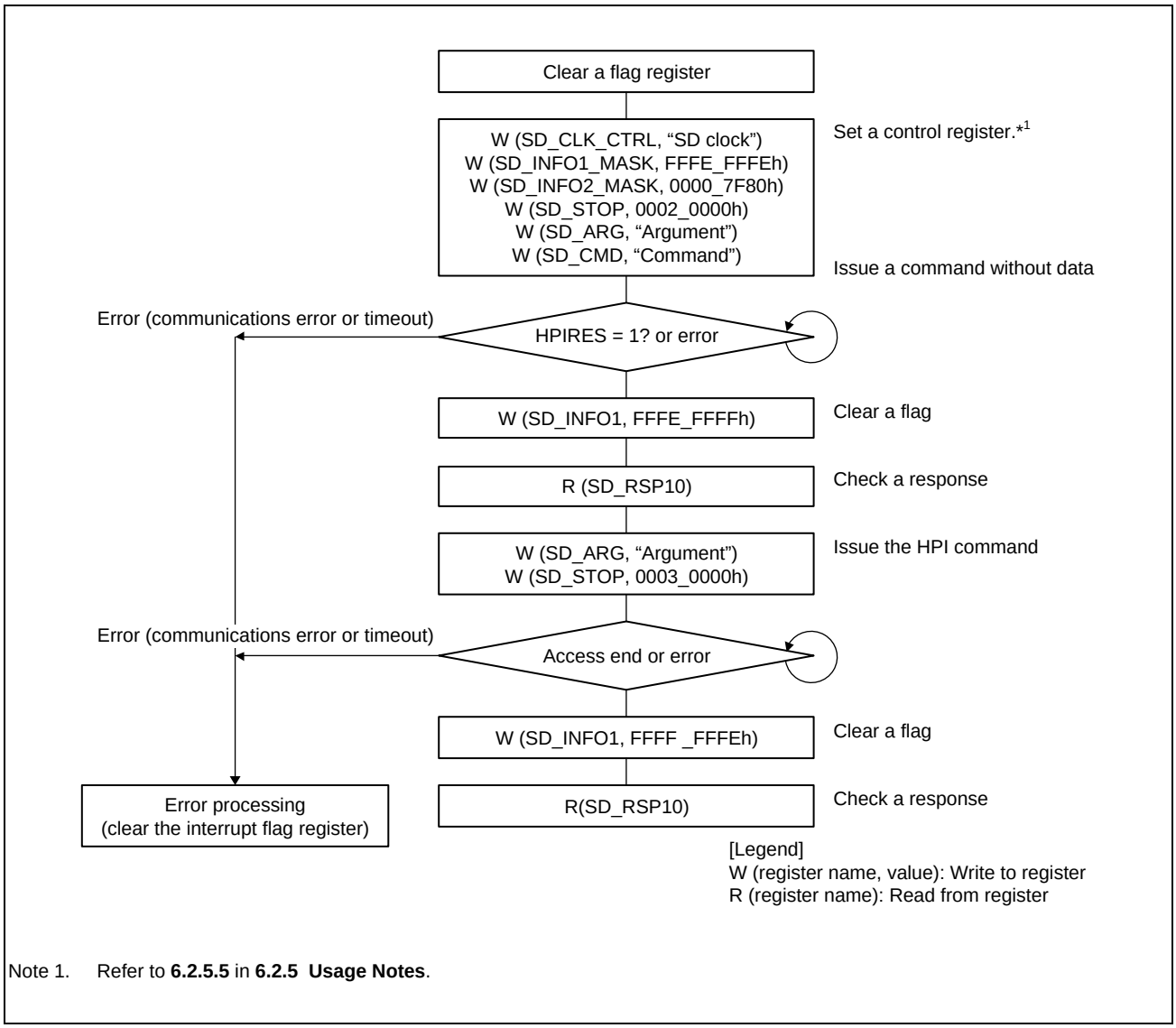


Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-25 SD_BUF DMA Write Flowchart Example

6.2.4.11 High-Priority Interrupt (without Data Transfer)

6.2.4.11.1 Flowchart



Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-26 Example of the High-Priority Interrupt (without Data Transfer) Flowchart

6.2.4.11.2 Operation for high-priority interrupt without data transfer

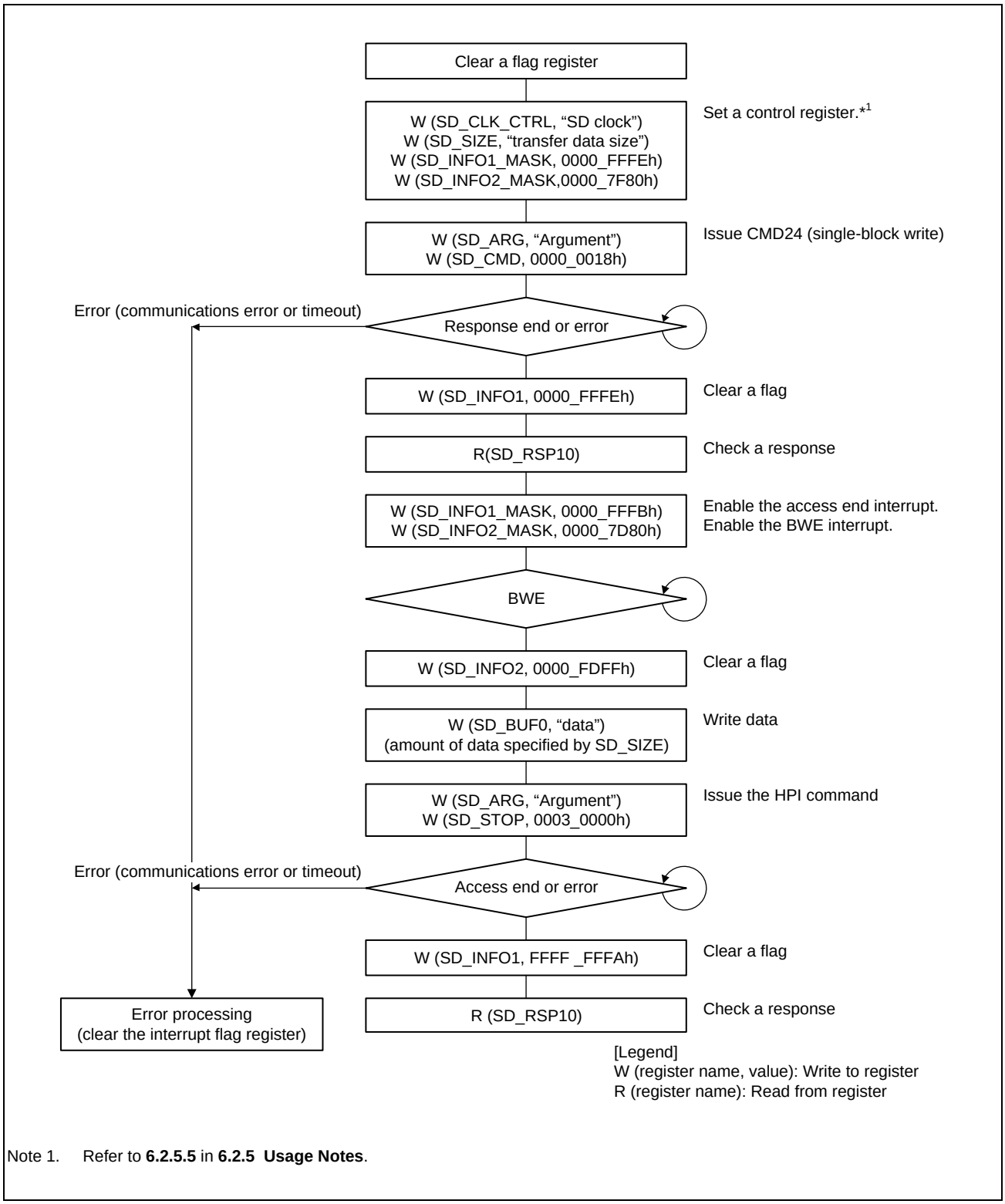
The operation of the high-priority interrupt (HPI) without data transfer is described below.

1. Flag register clear
   First, clear the bits in the flag registers. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), HPI enable, interrupt mask, and so on. (SD_CLK_CTRL, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue
   Set the CMD Argument in SD_ARG and write to SD_CMD.
   Accordingly, CMD is issued, and the operation is started.
4. Flag clear
   On receiving the response, the HPIRES bit in SD_INFO1 is set to 1 to generate an interrupt. Clear the HPIRES bit to 0.
5. Read the response from SD_RSP10.
6. HPI command issue
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
7. Operation complete
   When reception of the response to the HPI command is completed and the busy state is released, the INFO0 bit in SD_INFO1 is set to 1 to generate an interrupt. Clear the INFO0 bit to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

6.2.4.12 High-Priority Interrupt (at Single Block Write)

6.2.4.12.1 Flowchart



Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-27 Example of the High-Priority Interrupt (at Single Block Write) Flowchart

6.2.4.12.2 Operation for HPI at single block write

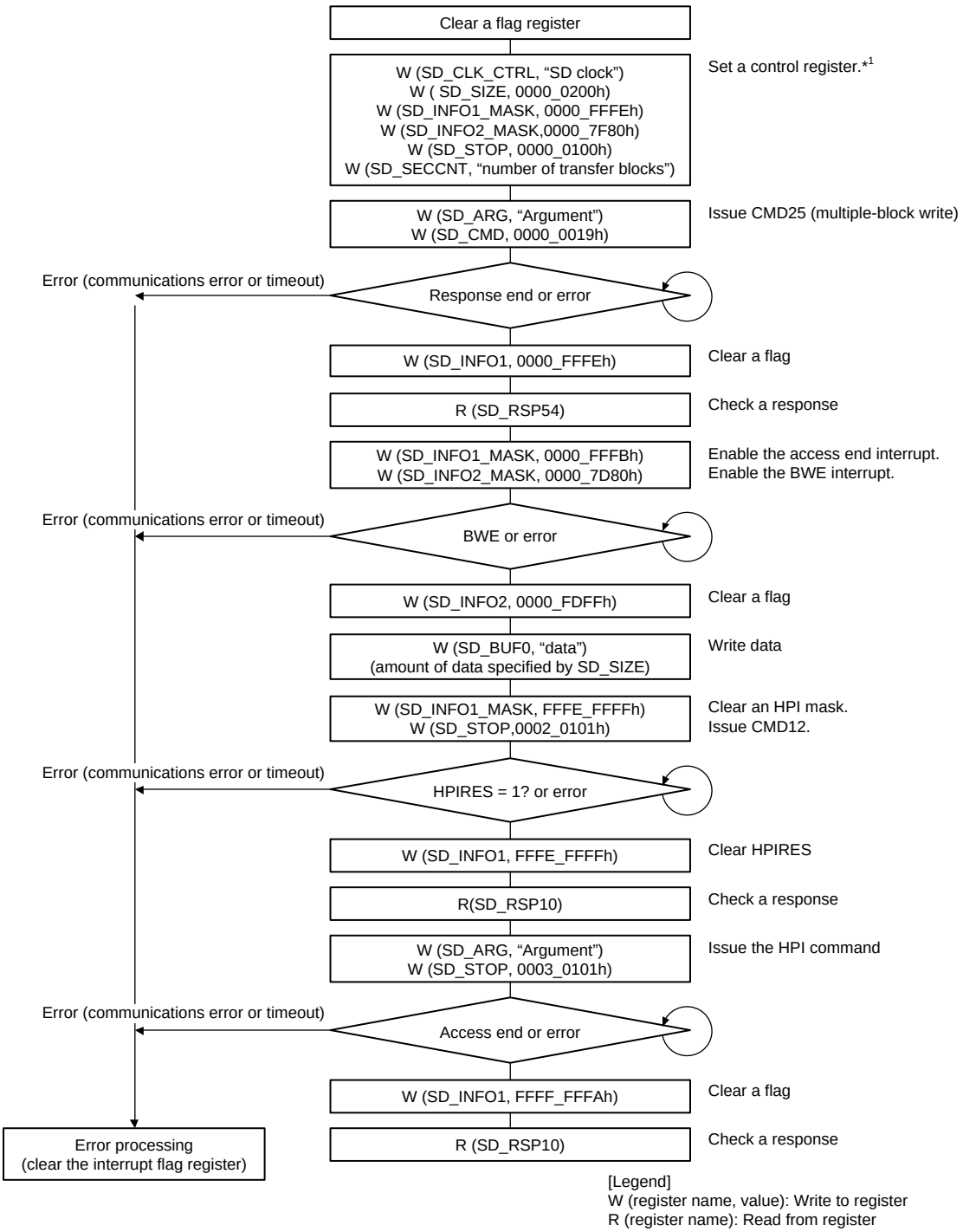
The operation of HPI at single block write is described below.

1. Flag register clear
   First, clear the bits in the flag registers. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
3. Command issue (CMD24)
   Set CMD24 Argument in SD_ARG and write 0000_0018h to SD_CMD.
   Accordingly, CMD24 is issued, and the single block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP10.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP or the IOABT bit in SDIO_MODE to 1. Furthermore, this causes CMD12 and CMD52 to not be issued.
   If the command sequence is halted, the INFO2 bit (access end) in SD_INFO1 is set to 1 to generate an interrupt.
5. Data write and data transmit to SD card
   Write 0000_FFFBh to SD_INFO1_MASK to enable the access end interrupt. In addition, write 0000_7D80h to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card.
6. HPI command issue
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
7. Operation complete
   When reception of the response to the HPI command is completed and the busy state is released, the INFO2 and INFO0 bits in SD_INFO1 are set to 1 to generate an interrupt. Clear the INFO2 and INFO0 bits to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

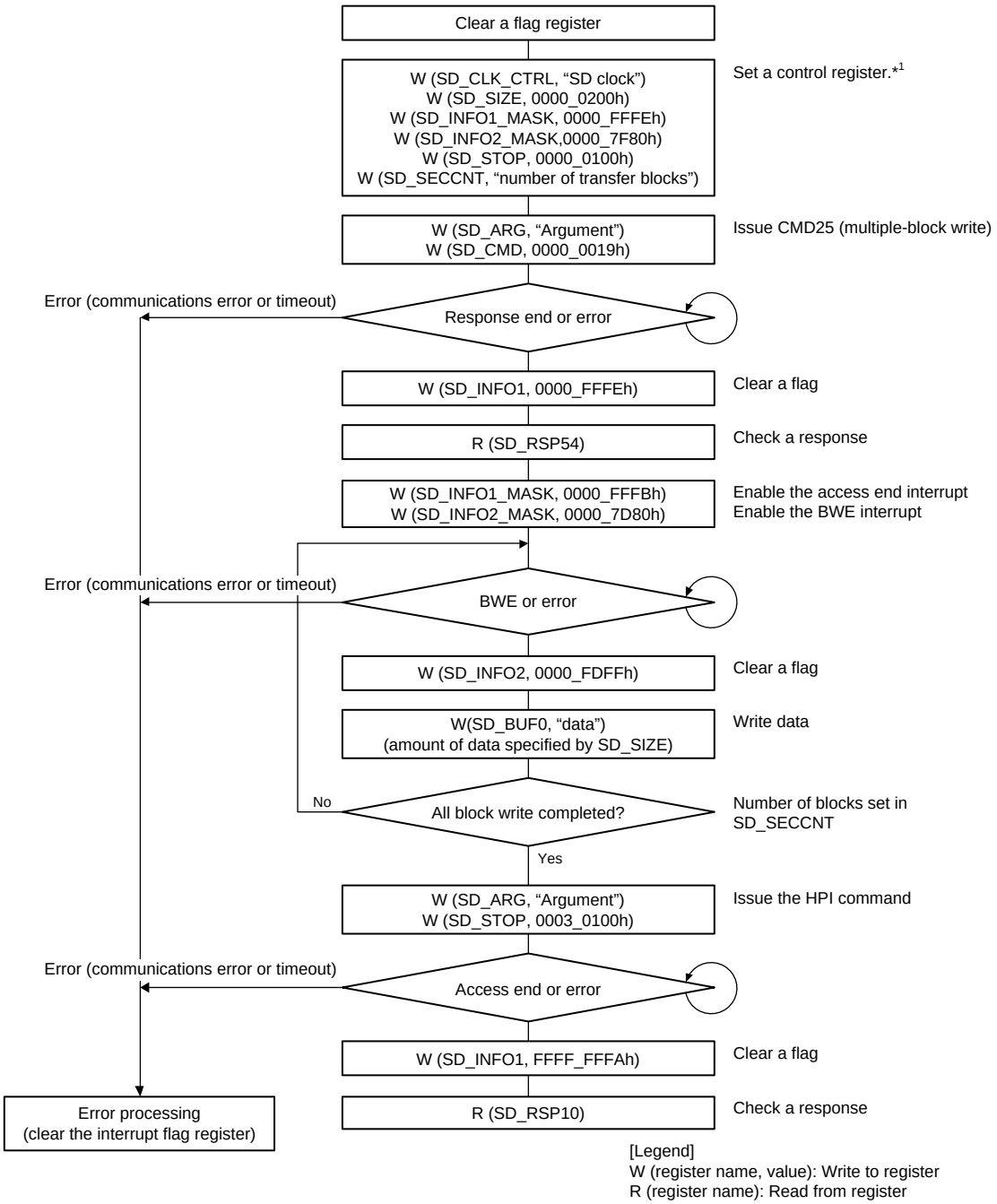
6.2.4.13 High-Priority Interrupt (at Multiple Block Write)

6.2.4.13.1 Flowchart



Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-28 Example of the High-Priority Interrupt (at Multiple Block Write) Flowchart (a)



Note 1. Refer to **6.2.5.5** in **6.2.5 Usage Notes**.

Figure 6.2-29 Example of the High-Priority Interrupt (at Multiple Block Write) Flowchart (b)

6.2.4.13.2 Operation for HPI at multiple block transfer

The operation of HPI at multiple block write is described below.

(1) When not all the data has been written to SD_BUF

1. Flag register clear
   First, clear the bits in the flag registers. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD25)
   Set CMD25 Argument in SD_ARG and write 0000_0019h to SD_CMD.
   Accordingly, CMD25 is issued, and the multiple block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting the INFO2 bit (access end) in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card
   Write 0000_FFFBh to SD_INFO1_MASK to enable the access end interrupt. In addition, write 0000_7D80h to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card. Then, the CRC status and busy state are received from the SD card.
   However, a communications error or timeout may be generated if data are being transmitted while writing to SD_BUF0 is in progress.
6. Command issue (CMD12)
   Write FFFE_FFFFh to SD_INFO1_MASK to enable the HPIRES interrupt. Write 0002_0101h to SD_STOP, which causes CMD12 to be issued.
7. Response check
   When the response is received, the HPIRES bit in SD_INFO1 is set to 1 to generate an interrupt. Clear the HPIRES bit to 0 to read the response from SD_RSP10.
8. HPI command issue (CMD25)
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
9. Operation complete
   When reception of the response to the HPI command is completed and the busy state is released, the INFO2 and INFO0 bits in SD_INFO1 are set to 1 to generate an interrupt. Clear the INFO2 and INFO0 bits to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

(2) When all the data has been written to SD_BUF

1. Flag register clear
   First, clear the bits in the flag registers. (SD_INFO1 and SD_INFO2)
2. Control register set
   Set the SD clock (SDCLK), transfer data size, interrupt mask, and so on. (SD_CLK_CTRL, SD_SIZE, SD_INFO1_MASK, and SD_INFO2_MASK)
   Set the SEC bit in SD_STOP to 1, and set the number of transfer blocks in SD_SECCNT.
3. Command issue (CMD25)
   Set CMD25 Argument in SD_ARG and write 0000_0019h to SD_CMD.
   Accordingly, CMD25 is issued, and the multiple block write operation is started.
4. Response check
   On receiving the response, INFO0 (response end) in SD_INFO1 is set to 1 to generate an interrupt. Clear INFO0 to 0 and read the response from SD_RSP54.
   If the result of response decoding is an error, the command sequence can be halted by setting the STP bit in SD_STP to 1. Setting the STP bit to 1 also causes CMD12 to be issued and the response received. If the command sequence is halted because the access end interrupt has been enabled, an interrupt will be generated by setting the INFO2 bit (access end) in SD_INFO1 to 1 when reception of the response has been completed. Clear the INFO2 bit to 0 and read the response.
5. Data write and data transmit to SD card
   Write 0000_FFFBh to SD_INFO1_MASK to enable the access end interrupt. In addition, write 0000_7D80h to SD_INFO2_MASK to enable the BWE interrupt. When SD_BUF0 is ready for the data to be written, the BWE bit in SD_INFO2 is set to 1 to generate an interrupt. Clear the BWE bit to 0 and write the amount of data specified by SD_SIZE to SD_BUF0. When the data write to SD_BUF0 is completed, data is transmitted to the SD card.
   Then, the CRC status and busy state are received from the SD card. Doing this repeats transfer of the number of blocks set in SD_SECCNT.
   However, a communications error or timeout may be generated if data are being transmitted while writing to SD_BUF0 is in progress.
6. HPI command issue
   Set the HPI command argument in SD_ARG and set the HPIMODE and HPICMD bits in SD_STOP to 1.
7. Operation complete
   When reception of the response to the HPI command is completed and the busy state is released, the INFO2 and INFO0 bits in SD_INFO1 are set to 1 to generate an interrupt. Clear the INFO2 and INFO0 bits to 0 to read the response from SD_RSP10. This is the end of HPI operation.

Furthermore, perform error processing (clear the interrupt flag register) if an error occurs (a communications error or timeout).

6.2.4.14 Example of SD_CMD Register Setting

Table 6.2-8 lists the example of SD_CMD (SD interface) register setting.

Table 6.2-8 Example of SD_CMD Register Setting (SD) (1/2)

Type	Command	Example of SD_CMD Register Setting	Remark
CMD	CMD0	0000_0000h	
	CMD2	0000_0002h	
	CMD3	0000_0003h	
	CMD4	0000_0004h	
	CMD5	0000_0705h or 0000_0005h	
	CMD6	0000_1C06h or 0000_0006h	
	CMD7	0000_0007h	When the card is placed in the deselected state, the response timeout flag will be set since there is no response.
	CMD8	0000_0408h or 0000_0008h	
	CMD9	0000_0009h	
	CMD10	0000_000Ah	
	CMD11	0000_040Bh or 0000_000Bh	
	CMD12	0000_000Ch	
	CMD13	0000_000Dh	
	CMD15	0000_000Fh	
	CMD16	0000_0010h	
	CMD17	0000_0011h	
	CMD18	0000_0012h	With auto CMD12 enabled (other than in SDR104 mode)
		0000_7C12h	With auto CMD12 disabled (only in SDR104 mode)
	CMD19	0000_1C13h or 0000_0013h	
	CMD20	0000_0514h or 0000_0014h	
	CMD23	0000_0417h or 0000_0017h	
	CMD24	0000_0018h	
	CMD25	0000_0019h	With auto CMD12 enabled (other than in SDR104 mode)
		0000_6C19h	With auto CMD12 disabled (only in SDR104 mode)
	CMD27	0000_001Bh	
	CMD28	0000_001Ch	
	CMD29	0000_001Dh	
	CMD30	0000_001Eh	
	CMD32	0000_0020h	
	CMD33	0000_0021h	
	CMD38	0000_0026h	
	CMD42	0000_002Ah	
	CMD48	0000_1C30h	
	CMD49	0000_0C31h	
	CMD52	0000_0434h or 0000_0034h	

Table 6.2-8 Example of SD_CMD Register Setting (SD) (2/2)

Type	Command	Example of SD_CMD Register Setting	Remark
CMD	CMD53	0000_1C35h	Single read
		0000_0C35h	Single write
		0000_7C35h	Multiple read
		0000_6C35h	Multiple write
		0000_0035h	The value on the left can be set irrespective of whether it is single or multiple. However, the CF39 bit in SD_ARG must be set as follows. Read: 0 Write: 1
	CMD55	0000_0037h	
	CMD56	0000_0038h	
	CMD58	0000_7C3Ah	
	CMD59	0000_6C3Bh	
ACMD	ACMD6	0000_0046h	
	ACMD13	0000_004Dh	
	ACMD22	0000_0056h	
	ACMD23	0000_0057h	
	ACMD41	0000_0069h	
	ACMD42	0000_006Ah	
	ACMD51	0000_0073h	

Table 6.2-9 lists the example of SD_CMD (MMC interface) register setting.

Table 6.2-9 Example of SD_CMD Register Setting (MMC)

Type	Command	Example of SD_CMD Register Setting	Remark
CMD	CMD0	0000_0000h	
	CMD1	0000_0701h	
	CMD2	0000_0002h	
	CMD3	0000_0003h	
	CMD4	0000_0004h	
	CMD5	0000_0505h	
	CMD6	0000_0506h	In the response busy state
		0000_0406h	Not in the response busy state
	CMD7	0000_0007h	When the card is placed in the deselected state, the response timeout flag will be set since there is no response.
	CMD8	0000_1C08h	
	CMD9	0000_0009h	
	CMD10	0000_000Ah	
	CMD12	0000_000Ch	
	CMD13	0000_000Dh	
	CMD14	0000_1C0Eh	SDIF_MODE must be set to 0100h (with CRC16 disabled).
	CMD15	0000_000Fh	
	CMD16	0000_0010h	
	CMD17	0000_0011h	
	CMD18	0000_7C12h	Pre-defined
	CMD19	0000_0C13h	SDIF_MODE must be set to 0100h (with CRC16 disabled).
	CMD21	0000_1C15h	Setting prohibited in DDR mode
	CMD23	0000_0017h	
	CMD24	0000_0018h	
	CMD25	0000_6C19h	Pre-defined
	CMD26	0000_0C1Ah	
	CMD27	0000_001Bh	
	CMD28	0000_001Ch	
	CMD29	0000_001Dh	
	CMD30	0000_001Eh	
	CMD31	0000_1C1Fh	
	CMD35	0000_0423h	
	CMD36	0000_0424h	
	CMD38	0000_0026h	
	CMD39	0000_0427h	
	CMD40	0000_0428h	
	CMD42	0000_002Ah	
	CMD49	0000_0C31h	
	CMD53	0000_7C35h	
	CMD54	0000_6C36h	
	CMD55	0000_0037h	
	CMD56	0000_0038h	

6.2.5 Usage Notes

6.2.5.1 SD_BUF Illegal Write Access

When writing data to SD_BUF0 after the single block write or multi block write command is issued, the data of the size specified by SD_SIZE must be written to.

If the data of the size which exceeds the size specified by SD_SIZE is written to, the ERR4 bit in SD_INFO2 is set to 1. In addition, the data written to SD_BUF0 may not be transmitted and it causes the SCLKDIVEN bit in SD_INFO2 to hold the value of 0. In such cases, clearing the SDRST bit in SOFT_RST to 0 and then restoring its value to 1 clears the SCLKDIVEN bit to 1.

However, for the single byte (in the case of 16- or 32-bit access) or three bytes (in the case of 32-bit access) when the number of bytes setting in SD_SIZE is odd, or the fraction of bytes when the number of bytes setting in SD_SIZE is even (in the case of 32-bit access), since the portion of dummy data writing is regarded as excess data and ignored, it is not within the scope of the above description (the fraction of bytes: the two bytes that are not in a four-byte unit).

6.2.5.2 Block Number Limitation for Multiple Block Read

When performing a multiple block read of one or two blocks, depending on the timing with which the response register is read, the response value may not be read properly. This must be avoided by either of the following countermeasures.

1) When receiving one or two blocks of data, use single block reading.
2) Read the response to CMD18 from SD_RSP54.

[Mechanism of incorrect reading]

Figure 6.2-30 shows the processing flows of SD/MMC host interface (hardware) operation and software operation when a multiple block read is performed on two blocks. As shown in the incorrect operation of **Figure 6.2-30**, when an interrupt is generated on reception of the CMD18 response and the timing with which the SD card response register (SD_RSP10) is read by the interrupt is delayed, the data during the CMD12 response reception or the CMD12 response may be read. In the case of a multiple block read of three or more blocks, CMD12 is not issued until the block of data has been read, so this problem does not arise. Furthermore, in the case of a multiple block write, since the CMD25 response is read before the block of data is sent, the problem does not arise.

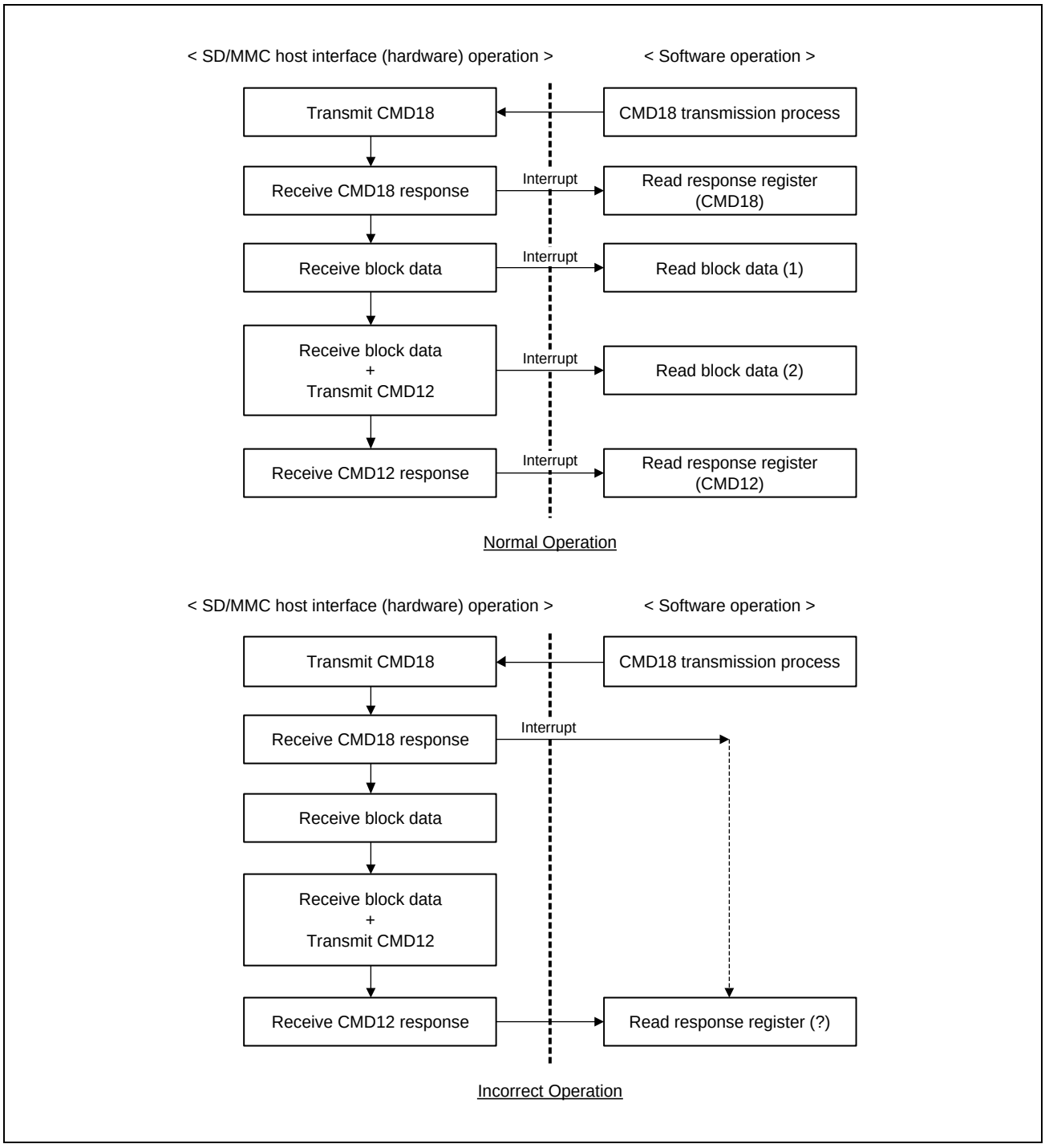


Figure 6.2-30 Flowcharts for Multiple Block Read Operation (Two Blocks)

6.2.5.3 Automatic Control of SDCLK Output

In the SD Card standard, 74 cycles of SDCLK must be output before initialization of the card. For this reason, use automatic control of SDCLK output after 74 cycles of SDCLK have been output. Furthermore, if automatic control of SDCLK output was in use, SDCLK output is stopped on completion of the sequence for a communications error or timeout. Thus, in cases where state transitions within the SD card are necessary and so on after completion of the sequence, release automatic control of SDCLK output and restart supply of SDCLK to the SD card.

6.2.5.4 Control of the C52PUB Setting for Multiple Block Write

If the C52PUB bit in SDIO_MODE is set to 1 during a sequence of multiple block write due to CMD53, CMD52 is not issued until SD_BUF becomes empty. For this reason, set the C52PUB bit after suspending writing to SD_BUF by following the appropriate procedure below.

- When DMA transfer is not in use

1. Before setting the C52PUB bit, suspend writing to SD_BUF by making the setting in SD_INFO2 to disable BWE interrupts.
2. Set the C52PUB bit in SDIO_MODE to 1 (so that CMD52 is issued when SD_BUF becomes empty).
3. After the INFO0 interrupt processing in SD_INFO1 due to the issuing of CMD52 has been completed, restart writing to SD_BUF by making the setting in SD_INFO2 to enable BWE interrupts.

- When DMA transfer is in use

1. Every time DMA transfer of the value set in SD_SIZE × n blocks (where n = 1, 2, ...) proceeds, suspend writing to SD_BUF by DMA transfer before the C52PUB bit is set.
2. Set the C52PUB bit in SDIO_MODE to 1 (so that CMD52 is issued when SD_BUF becomes empty).
3. After the INFO0 interrupt processing in SD_INFO1 due to the issuing of CMD52 has been completed, restart writing to SD_BUF by DMA transfer.

6.2.5.5 Notes on SD_CLK_CTRL Register Settings

When the SCLKDIVEN bit in SD_INFO2 is 0, SD_CLK_CTRL cannot be written to. Before writing to SD_CLK_CTRL, be sure to check that the SCLKDIVEN bit in SD_INFO2 is 1.

6.2.5.6 Restrictions on specifications

1. The SDIO suspend/resume is not supported.
2. The SPI bus is not supported.
3. The shared bus and 8-bit SD bus for embedded SDIO are not supported.
4. The stream transfer for MMC cards is not supported.
5. MMC open ended multiple block read is not supported.

6.2.5.7 STP bit setting during multiple block read

During execution of multiple block read with automatic CMD12 execution by setting the SEC bit in SD_STOP to 1, even if the STP bit in SD_STOP is set to 1 to forcibly stop the execution, the command sequence may not stop depending on the timing of setting the STP bit.

To avoid this, when setting the STP bit in SD_STOP to 1 during multiple block transfer, clear the SEC bit in SD_STOP to 0 at the same time. (Even when the SCLKDIVEN bit in SD_INFO2 is 0, change the SEC bit from 1 to 0.)

When the command sequence has not stopped because the SEC bit was not cleared to 0, the command sequence can be stopped by clearing the SDRST bit in SOFT_RST to 0.

When forcibly terminating the CMD53 multiple block transfer through the IOABT bit in SDIO_MODE, be sure to leave the SEC bit in SD_STOP as 1.

6.2.6 Sampling Clock Controller (SCC)

6.2.6.1 Features

This module controls a sampling clock (hereafter referred to as the SCC sampling clock) that is used for SD UHS-I/ SDR104 and MMC HS200. When this module is used with the LSI, SD UHS-I/SDR104 and MMC HS200 can be supported.

6.2.6.2 SCC Block Diagram

Figure 6.2-31 shows a block diagram of the sampling clock controller.

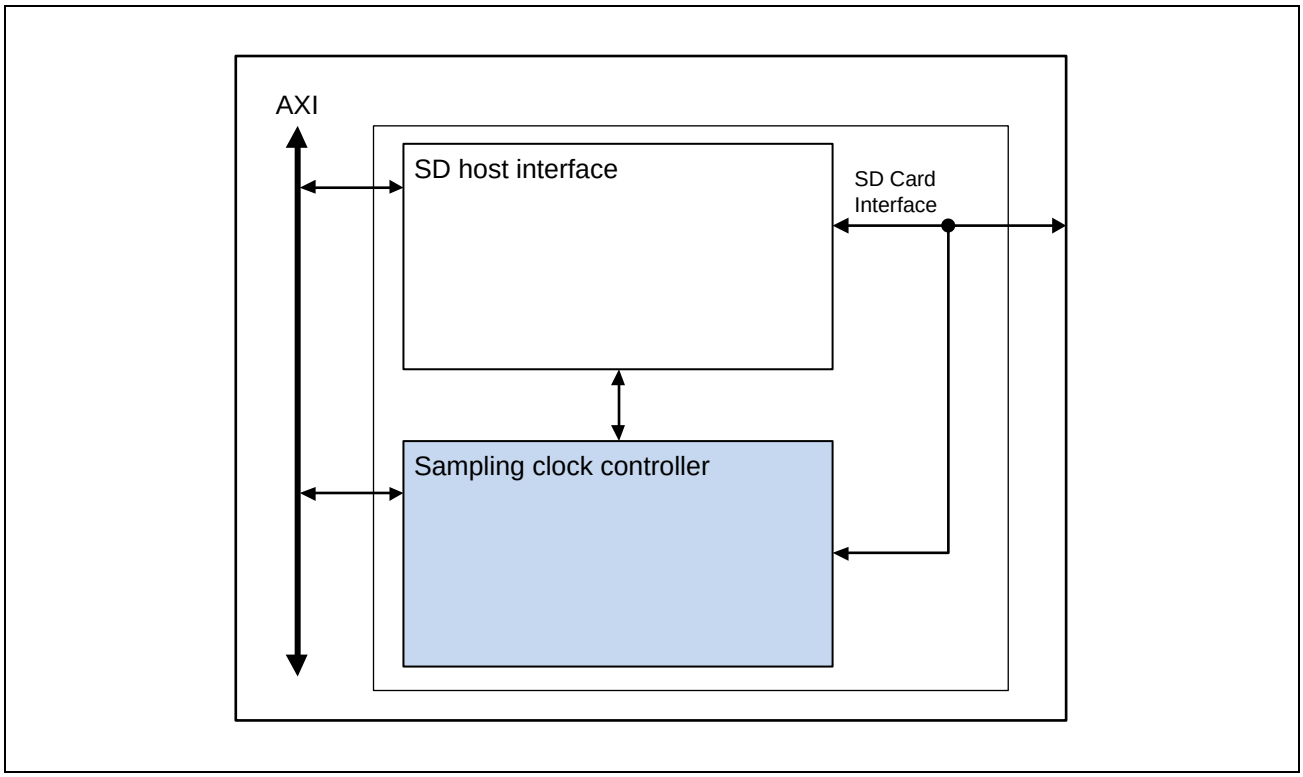


Figure 6.2-31 Block Diagram of the Sampling Clock Controller

6.2.7 SCC Registers

6.2.7.1 List of Registers

The registers are listed below.

The SCC registers are listed below. Regarding the base addresses of the following registers, refer to **6.2.2 SD Registers**.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Initial Setting Register	SDm_SCC_DTCNTL	0000_0000_0008_0000h	1000h	32, 64
Sampling Clock Position Setting Register	SDm_SCC_TAPSET	0000_0000_0000_0000h	1008h	32, 64
Hardware Adjustment Register 1	SDm_SCC_DT2FF	0000_0000_0000_0300h	1010h	32, 64
Sampling Clock Selection Register	SDm_SCC_CKSEL	0000_0000_0000_0000h	1018h	32, 64
Sampling Clock Position Correction Register	SDm_SCC_RVSCNTL	0000_0000_0000_0000h	1020h	32, 64
Sampling Clock Position Correction Request Register	SDm_SCC_RVSREQ	0000_0000_0000_0000h	1028h	32, 64
Sampling data comparison register	SDm_SCC_SMPCMP	0000_0000_0000_0000h	1030h	32, 64
Hardware adjustment register 2	SDm_SCC_TMPPORT	0000_0000_0000_0000h	1038h	32, 64

6.2.7.2 SCC Register Descriptions

6.2.7.2.1 Initial Setting Register (SDm_SCC_DTCNTL)

Access Size : 32, 64 bits

Offset Address : <SDm_base> + 1000h

Initial Value : 0000_0000_0008_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	TAPNUM7 to TAPNUM0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TAPEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23 to 16	TAPNUM7 to TAPNUM0[7:0]	8h	RW	When bits DIV7 to DIV0 in the SD_CLK_CTRL register are FFh (1: 1 mode), set these bits to 08h.
15 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	TAPEN	0h	RW	SCC Sampling Clock Operation Enable 0b: SCC sampling clock operation is disabled. 1b: SCC sampling clock operation is enabled.

6.2.7.2.2 Sampling Clock Position Setting Register (SDm_SCC_TAPSET)

Access Size : 32, 64 bits

Offset Address : <SDm_base> + 1008h

Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TAPSET7 to TAPSET0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 0	TAPSET7 to TAPSET0[7:0]	0h	RW	SCC Sampling Clock Position Set the tuning result in the range from 0 to TAPNUM-1.

6.2.7.2.3 Sampling Clock Selection Register (SDm_SCC_CKSEL)

Access Size : 32, 64 bits

Offset Address : <SDm_base> + 1018h

Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DTSEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	DTSEL	0h	RW	Sampling Clock Selection 0b: An SCC sampling clock is not used (for other than SDR104 and HS200). 1b: An SCC sampling clock is used (for SDR104 or HS200). - For SDR104 or HS200, set DTSEL to 1. DIV[7:0] in the SD_CLK_CTRL register to FFh (1:1 mode). - When this bit is switched, stop the SD clock output from the SD/MMC host interface (set SCLKEN in SD_CLK_CTRL to 0).

6.2.7.2.4 Sampling Clock Position Correction Register (SDm_SCC_RVSCNTL)

Access Size : 32, 64 bits

Offset Address : <SDm_base> + 1020h

Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAPSEL7 to TAPSEL0[7:0]								-	-	-	-	-	-	RVSW	RVSEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 8	TAPSEL7 to TAPSEL0[7:0]	0h	R	SCC Sampling Clock Position Display Displays the SCC sampling clock position selected by hardware. After RVSEN has been set to 1b, the value may differ from that of TAPSET.
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	RVSW	0h	RW	This bit is read as 0b. The write value should be 1b.
0	RVSEN	0h	RW	SCC Sampling Clock Position Correction Enable 0b: SCC sampling clock position correction is disabled. 1b: SCC sampling clock position correction is enabled. When RVSEN is set to 1b after tuning has been performed, this module corrects the SCC sampling clock position each time of a command sequence of the SD/MMC host interface. However, when RVSERR is 1b, this module does not correct the SCC sampling clock position. While tuning is being performed, set RVSEN to 0. When this controller is used with an SDIO device which asserts SDIO interrupt, set RVSEN to 0.

6.2.7.2.5 Sampling Clock Position Correction Request Register (SDm_SCC_RVSREQ)

Access Size : 32, 64 bits

Offset Address : <SDm_base> + 1028h

Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	RVSERR	REQTAPUP	REQTAPDWN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	RVSERR	0h	RW	SCC Sampling Clock Position Correction Error 0b: There is no correction error. 1b: There is a correction error. - If this bit is set to 1b after a command sequence, write 0b to this bit and perform tuning again. - Ignore this bit while tuning is being performed. - Writing 1 to this bit is disabled and writing 0b to this bit is only enabled.
1	REQTAPUP	0h	RW	SCC Sampling Clock Position Positive Direction Correction Request 0b: There is no correction request. 1b: There is a correction request. - If this bit is set to 1b after a command sequence, write 0b to this bit and rewrite TAPSET in the positive direction (when TAPSEL = TAPNUM-1, set 0 to TAPSET). - When RVSEN is 1b, this bit is disabled (this bit is not set to 1). - Writing 1b to this bit is disabled and writing 0b to this bit is only enabled.
0	REQTAPDWN	0h	RW	SCC Sampling Clock Position Negative Direction Correction Request 0b: There is no correction request. 1b: There is a correction request. - If this bit is set to 1b after a command sequence, write 0b to this bit and rewrite TAPSET in the negative direction (when TAPSEL = 0, set TAPNUM-1 to TAPSET). - When RVSEN is 1b, this bit is disabled (this bit is not set to 1b). - Writing 1b to this bit is disabled and writing 0b to this bit is only enabled.

6.2.7.2.6 Hardware Adjustment Register 1 (SDm_SCC_DT2FF)

This register makes a setting that SD_DATA, which has been fetched by the sampling clock at each TAP position, is used in the appropriate timing.

Access Size : 32, 64 bits
Offset Address : <SDm_base> + 1010h
Initial Value : 0000_0000_0000_0300h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DT2NE[7:0]								DT2NS[7:0]							
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 8	DT2NE[7:0]	3h	RW	Hardware Adjustment 2 This is a setting register for adjusting the timing inside the IP when using Tuning. When using Tuning, set 8'h02 to this bit, make the setting before setting 1 in SCC_CKSEL [0] .DTSEL.
7 to 0	DT2NS[7:0]	0h	RW	Hardware Adjustment 1 This is a setting register for adjusting the timing inside the IP when using Tuning. When using Tuning, set 8'h07 to this bit., make the setting before setting 1 in SCC_CKSEL [0] .DTSEL.

6.2.7.2.7 Sampling Data Comparison Register (SDm_SCC_SMPCMP)

Data comparison register indicates the result of the comparison of the sampling data. The subject of the comparison is the previous TAP and the subsequent TAP.

Access Size : 32, 64 bits
Offset Address : <SDm_base> + 1030h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	CMPNGU[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CMPNGD[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
24 to 16	CMPNGU[8:0]	0h	RW	Comparison of sampling data with the previous TAP Clock. Bits 16-23 are the comparison result of data 0-7. Bit 24 is the comparison result of CMD. 0b: Match 1b: Mismatch < Clear conditions.> The start of the command sequence Write to SCC_TAPSET register
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8 to 0	CMPNGD[8:0]	0h	RW	Comparison of sampling data with the subsequent TAP Clock. Bits 0-7 are the comparison result of data 0-7. Bit 8 is the comparison result of CMD. 0b: Match 1b: Mismatch < Clear conditions.> The start of the command sequence Write to SCC_TAPSET register

6.2.7.2.8 Hardware Adjustment Register 2 (SDm_SCC_TMPPORT)

This register makes a setting that SD_DATA, which has been fetched by the sampling clock at each TAP position, is used in the appropriate timing.

Access Size : 32, 64 bits
Offset Address : <SDm_base> + 1038h
Initial Value : 0000_0000_0000_0000h

Bit	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMPOUT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
63 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 0	TMPOUT[15:0]	0h	RW	Hardware adjustment 3 When using the delay tuning mechanism, set the following values. When TAPNUM = 8, set 0000h to these bits. When operating with other than SDR104 / HS200, this register adjusts the clock delay of the Flip Flop that latches the received data from the outside of the chip. For this LSI, set 0000h when transferring at 3.3 V, and set 0001h when transferring at 1.8 V.

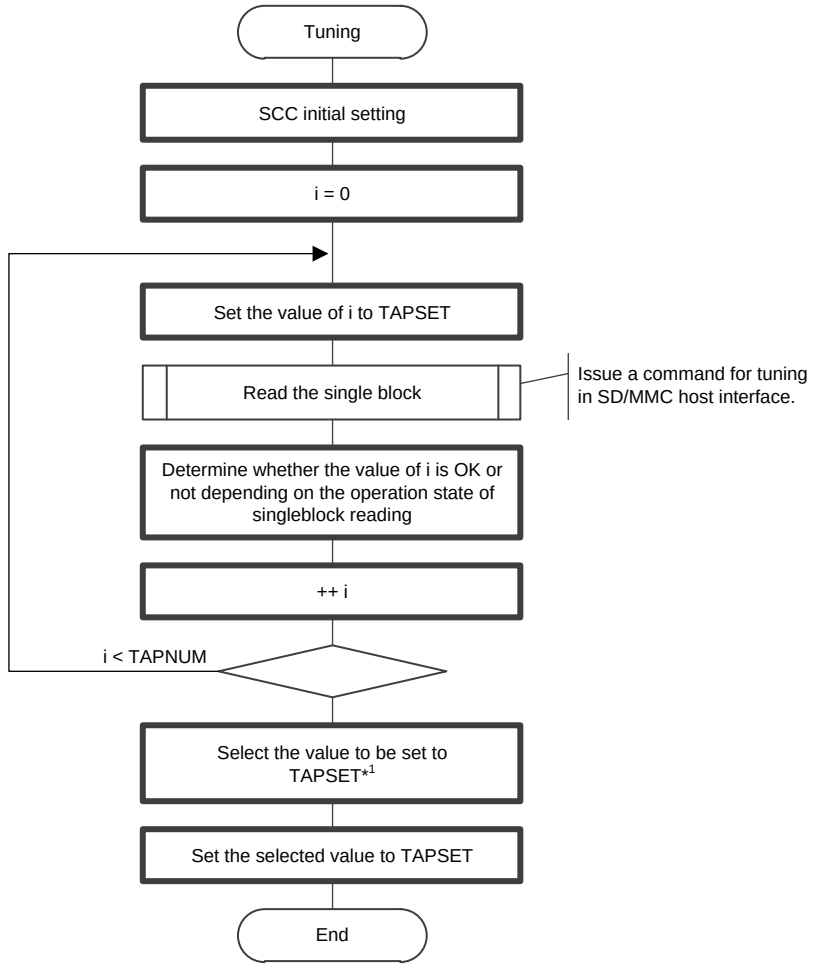
6.2.8 Usage Example of SCC

6.2.8.1 Tuning

SCC is tuned by using operation of single-block reading.

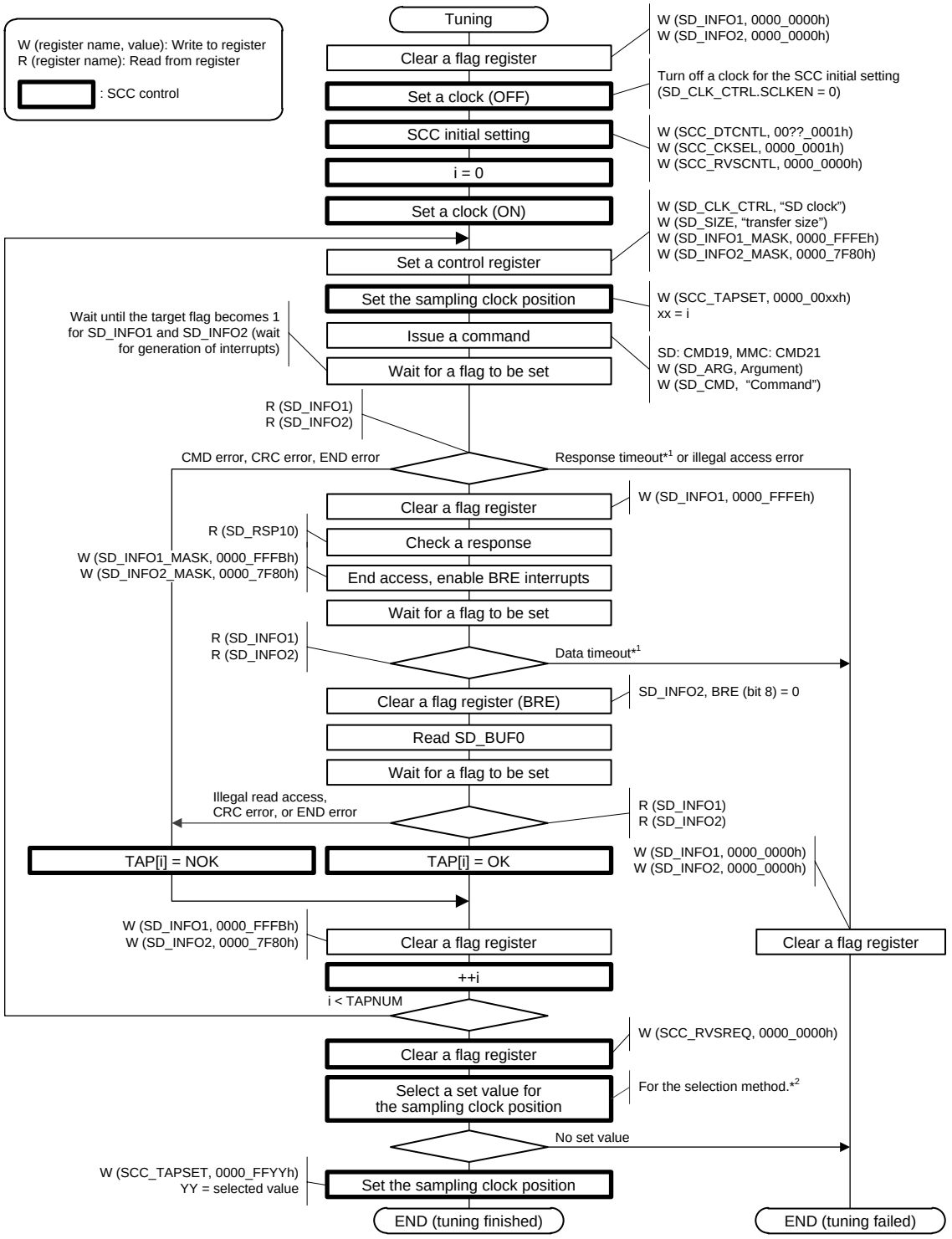
As shown in **Figure 6.2-32**, check whether the single-block read command normally ends when the sampling clock position is changed from 0 to TAPNUM-1 and save the result. After checking, confirm that there exists the range which has three or more continuous normal ends (OK). Then, the median value within the continuous range is determined as the final adjustment value.

Figure 6.2-33 and **Table 6.2-10** show the detailed tuning flow and how to select the sampling clock position (example when TAPNUM = 8).



Note 1. See the example shown in **Table 6.2-10**.

Figure 6.2-32 Example of Tuning Flow (Outline)



Note 1. When all of TAP[i] are "NOK".

Note 2. See **Table 6.2-10**.

Figure 6.2-33 Example of Tuning Flow (Detailed)

Table 6.2-10 Example of How to Select the Sampling Clock Position (when TAPNUM = 8)

Item	i	Case 1	Case 2	Case 3	Case 4	Case 5*1
TAP[i]	0	NOK	OK	NOK	OK	OK
	1	OK	OK	NOK	NOK	OK
	2	OK	NOK	OK	NOK	OK
	3	OK(←)	NOK	OK	NOK	OK
	4	OK	NOK	NOK	NOK	OK
	5	OK	OK	NOK	OK	OK
	6	NOK	OK	NOK	OK(←)	OK
Max. value→	7	NOK	OK(←)	NOK	OK	OK
	(0)	NOK	OK	NOK	OK	OK
	(1)	OK	OK	NOK	NOK	OK
	(2)	OK	NOK	OK	NOK	OK
	(3)	OK	NOK	OK	NOK	OK
	(4)	OK	NOK	NOK	NOK	OK
	(5)	OK	OK	NOK	OK	OK
	(6)	NOK	OK	NOK	OK	OK
	(7)	NOK	OK	NOK	OK	OK
Selected value		i = 3	i = 7	Fail	i = 6 or 7	i = 0 to 7

Remarks: (←): Example of selection, (x): repeated display of index x of TAP[x]

(a) The sampling clock position is selected by considering a margin in the range which has three or more continuous 'TAP[i] = OK'.

(b) The sampling clock position is repeated from 0 after the maximum value (TAPNUM-1). In case 2 above, that position is continued in the order of 5 → 6 → 7 → 0 → 1.

Note 1. If all of the TAP [i] is OK, the sampling clock position is selected by identifying the change point of data. Change point of the data can be found in the value of SCC_SMPCMP register. Usage example is **6.2.8.3 Change point of the input data**.

6.2.8.2 Sampling Clock Position Correction after Tuning

After tuning, correction of the sampling clock position may be required when a command is issued.

There are manual and automatic correction methods. After a command sequence, if the CMD, CRC, END error or time out occurs or the correction error occurs, tuning will be performed again. The following shows examples of manual and automatic correction methods.

6.2.8.2.1 Manual correction of the sampling clock position

Figure 6.2-34 shows the flow of manual correction of the sampling clock position. **Table 6.2-11** shows set values determined when correction is required (when TAPNUM = 8).

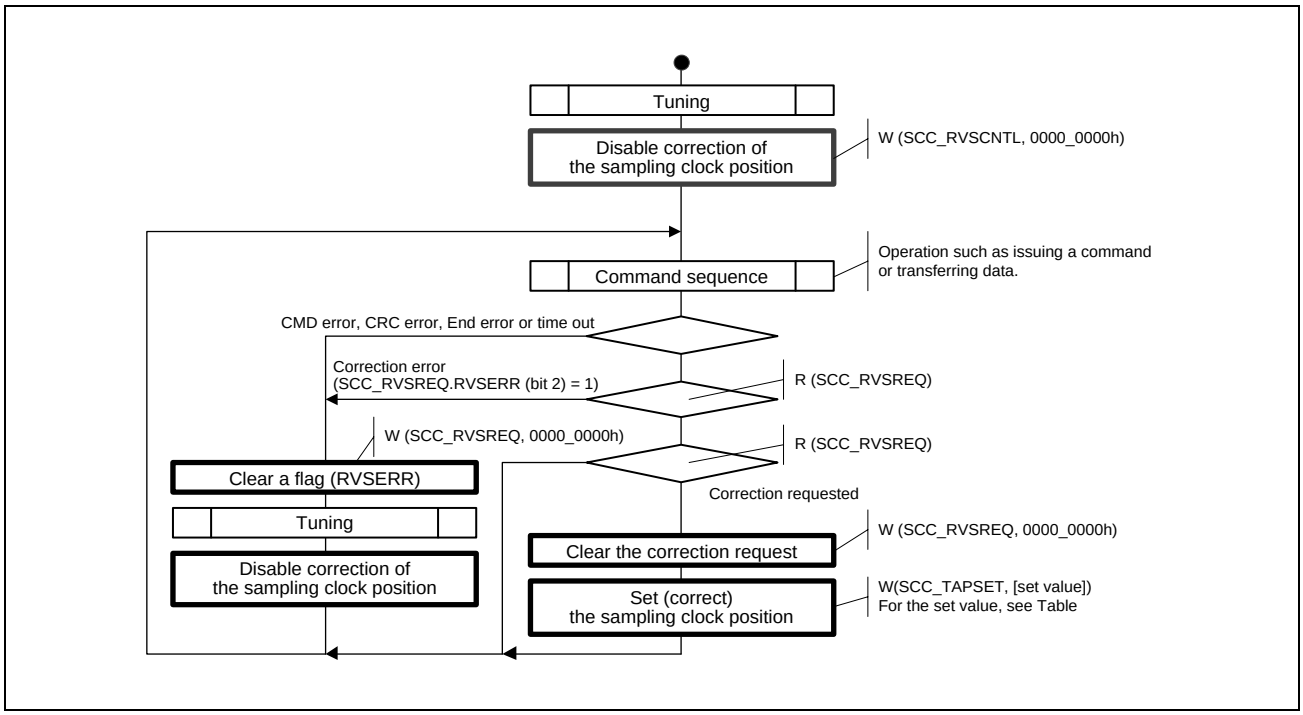


Figure 6.2-34 Flow of Manual Correction of the Sampling Clock Position (Example)

Table 6.2-11 Set Values for TAPSET when Correction is Required (when TAPNUM = 8)

No.	Current value of TAPSET	Value set to TAPSET when REQTAPUP = 1	Value set to TAPSET when REQTAPDWN = 1
1	0	1	7
2	1	2	0
3	2	3	1
4	3	4	2
5	4	5	3
6	5	6	4
7	6	7	5
8	7	0	6

Note: As is the case in the tuning selection method, the sampling clock position is 0 after the maximum value (TAPNUM-1).

6.2.8.2.2 Automatic correction of the sampling clock position

Figure 6.2-35 shows the flow of automatic correction of the sampling clock position.

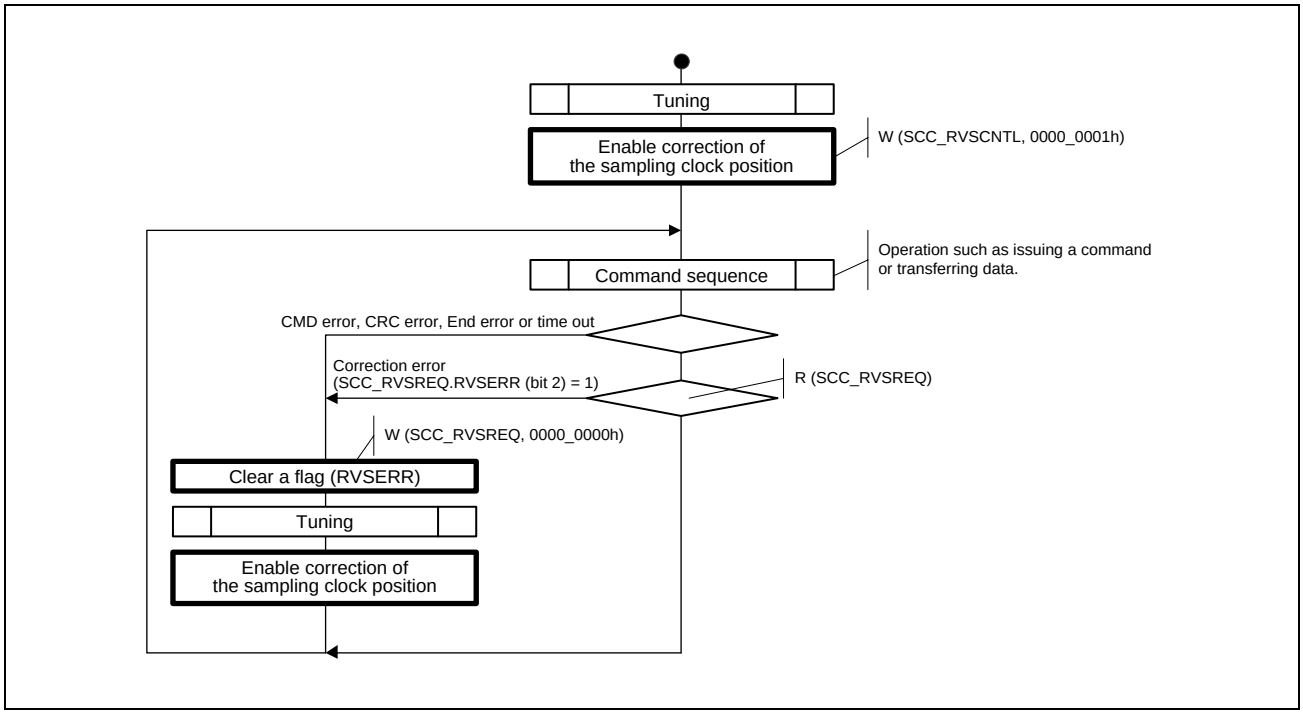
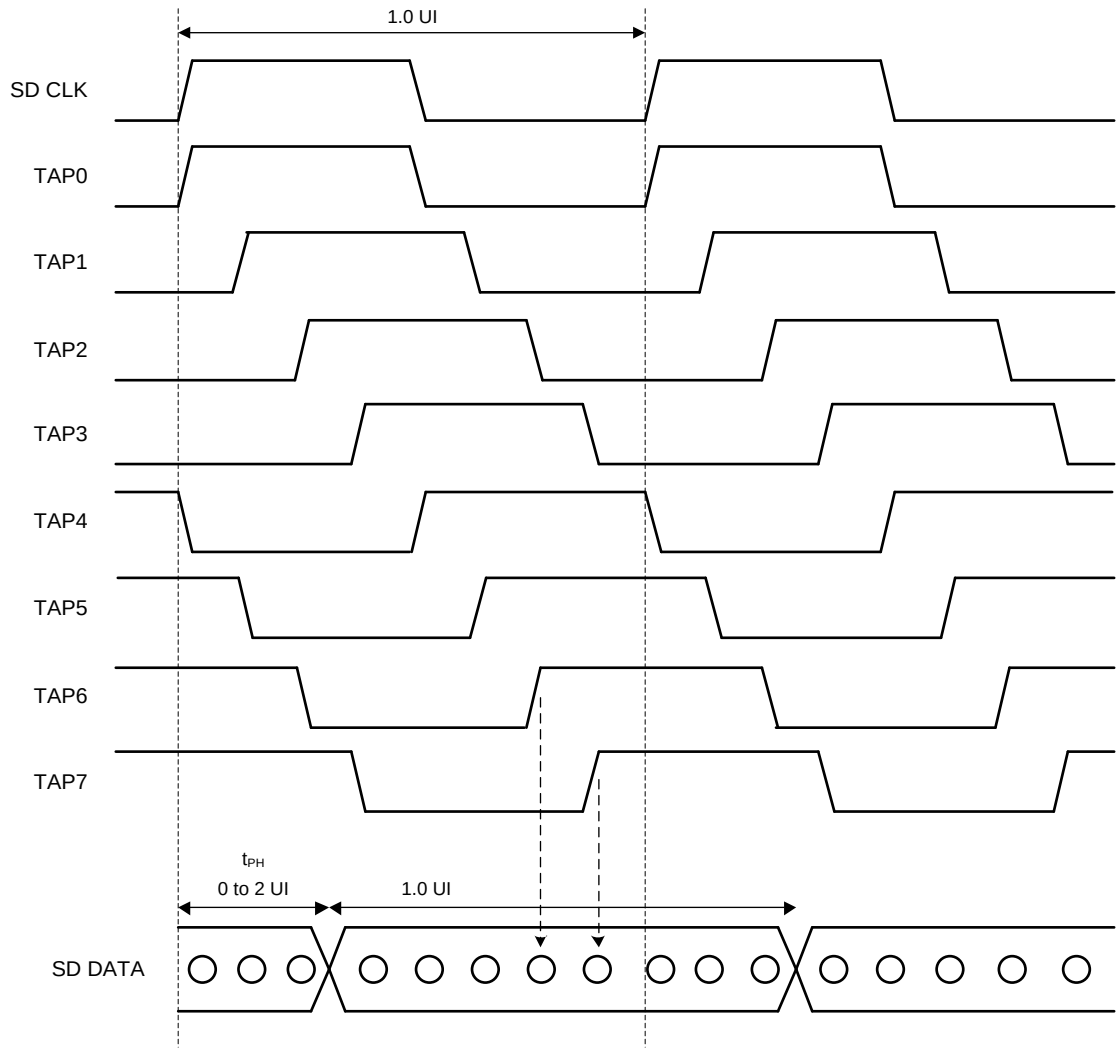


Figure 6.2-35 Flow of Automatic Correction of the Sampling Clock Position (Example)

6.2.8.3 Change point of the input data

Tuning is capture the data by the TAP clock selected. However, also captures the data by the previous TAP clock and the subsequent TAP clock at the same time. This result is reflected in the sampling data comparison register (SCC_SMPCMP). Point of mismatch before and after the selected TAP clock is the changing point of the data. In this example, it is desirable to set the TAP clock as TAP6 or TAP7.



Note: After recognizing the changing point of the input data, need to set the TAP clock.

Figure 6.2-36 Example: All Taps are OK

Figure 6.2-36 shows the change point of data between TAP2 and TAP3. Change point of the data can be confirmed by sampling data comparison register (SCC_SMPCMP). For the tuning of TAP2 or TAP3, the CMPNGU bit of sampling data comparison register indicates a mismatch. As the width of the input data is 1 (UI), select TAP6 or TAP7 which is the median of next TAP3 from TAP3.

SECTION 6 HIGH-SPEED INTERFACE

6.3 Gigabit Ethernet Interface (GBETH)

This section describes the GBETH functions of this LSI.

This manual is a simplified version. For more information, refer to the User's Manual Additional Document.

6.3.1 Overview

6.3.1.1 Features

The LSI includes an Ethernet MAC that:

- Is compliant with the following standards:
  - IEEE 802.3-2008 for Ethernet MAC, Gigabit Media Independent Interface (RGMII), and Media Independent Interface (MII)
  - IEEE 1588-2008 v2 standard for precision networked clock synchronization, IEEE 1588-2008 v2 is compliant with Power IEEE-C37.238 profile
  - IEEE 802.3-az, version D2.0 for Energy Efficient Ethernet (EEE).
  - IEEE 802.1Qav-2009, IEEE 802.1Qat-2009, IEEE 802.1AS-2011
- Supports 10/100/1000 Mbps data transfer rates
- Supports both half-duplex and full-duplex operation
- Has programmable frame length to support both standard and jumbo ethernet frames of size up to 16 KB (16 KB - 1):
  - Jumbo mode support in cut-through mode only (not implemented in store and forward due to TX and RX FIFO size).
- Contains 31 MAC address registers for the address filter block
- Supports variety of flexible addresses filtering modes:
  - Perfect (DA) address filters with masks for each byte
  - SA address comparison check with masks for each byte
  - 256-bit Hash filter for multicast and unicast (DA) addresses
  - Option to pass all multicast addressed frames
  - Promiscuous mode support to pass all frames without any filtering for network monitoring
  - Passes all incoming packets (as per filter) with a status report.
- Supports native DMA with simple independent channels transmit and receive engines:
  - 4 RX channels of 8 KB FIFO size for receive channel
  - 4 TX channels of 8 KB FIFO size for transmit channel
  - DMA implements dual buffer (ring) or linked list (chained) descriptor chaining.
- Supports advanced IEEE 1588-2002 & 2008 ethernet frame time stamping:

- IEEE1588 time base information, with reference clock of 125 MHz
- IEEE1588 external snapshot.

- Supports programmable CRC generation and checking
- Supports RMON statistics (L2 layer only)
- Supports Station Management Block and MDIO interface
- Is Ethernet Energy Efficiency compliant with IEEE 802.3az-2010:
  - Supports wakeup on LAN on magic packet and packet filtering
  - Supports energy efficiency feature (LPI mode)
  - Wakeup capability.
- Has interface for:
  - RGMII and MII (GMII is not supported.)

6.3.1.2 Block Diagram

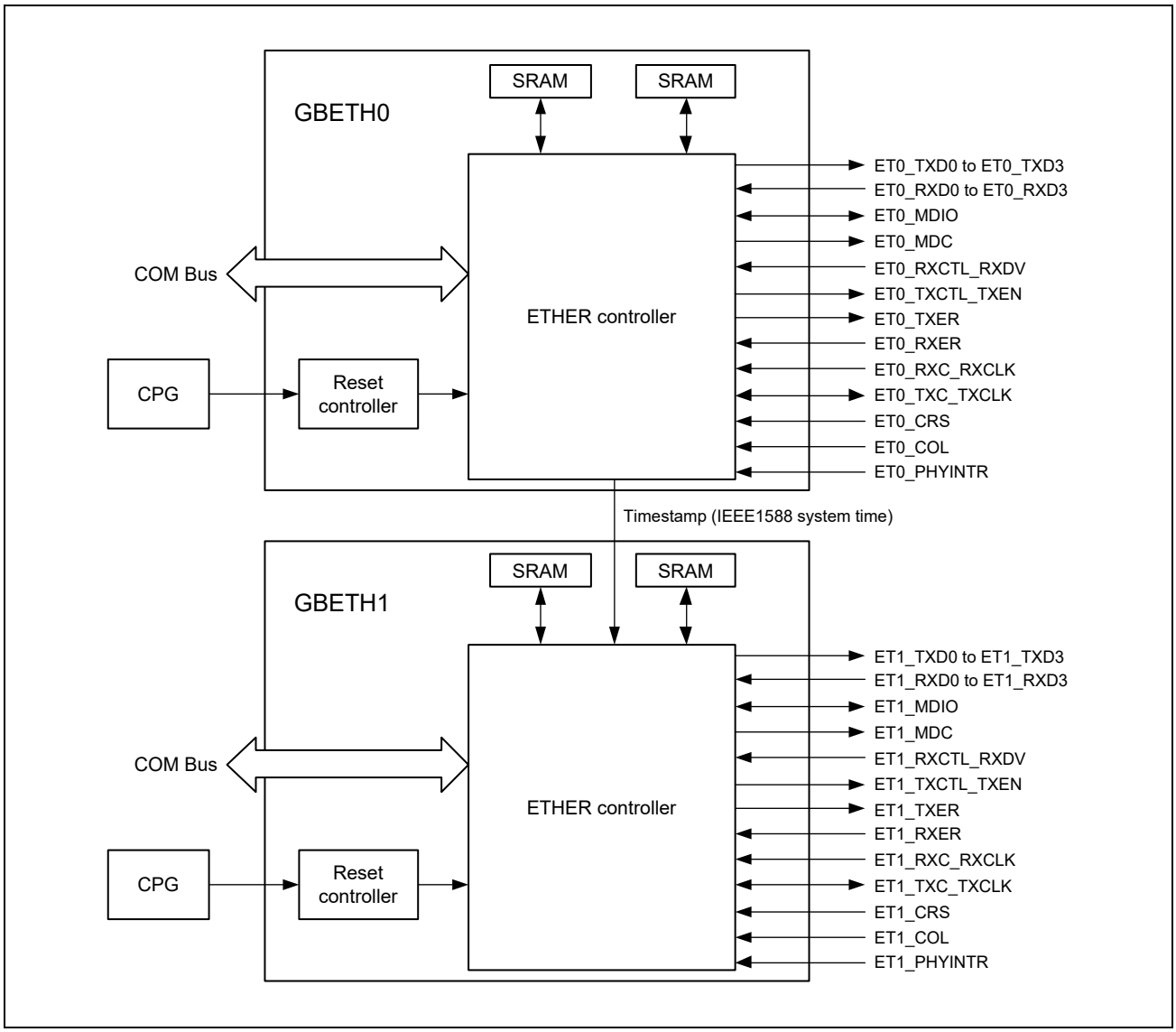


Figure 6.3-1 Block Diagram of the Ethernet Interface

6.3.1.3 External Pins

Table 6.3-1 Input/Output Pins of the GBETHm (m = 0, 1)

Pin Name	Input/Output	Description
ETm_MDC	Output	Management data clock output
ETm_MDIO	I/O	Management data I/O
ETm_RXCTL_RXDV	Input	RX control / data valid
ETm_TXCTL_TXEN	Output	TX control / data enable
ETm_TXER	Output	TX data error (MII mode)
ETm_RXER	Input	RX data error
ETm_RXC_RXCLK	Input	RX clocks
ETm_TXC_TXCLK	I/O	Tx clocks (MII mode)
ETm_CRS	Input	Carrier sense (MII mode)
ETm_COL	Input	Collision detection (MII mode)
ETm_TXD0 to ETm_TXD3	Output	TX data 0 to 3
ETm_RXD0 to ETm_RXD3	Input	RX data 0 to 3
ETm_PHYINTR	Input	PHY interrupt signals

SECTION 6 HIGH-SPEED INTERFACE

6.4 USB3.2 Gen2x1 Interface (USB3)

This section describes the functions of the USB3.2 Gen2x1 interface (USB3).

This manual is a simplified version. For more information, refer to the User's Manual Additional Document.

6.4.1 Functional Overview

In this product, one USB3 module is implemented (ch0). USB3 module is composed of a USB3.2 Gen2 Host controller (USB3HOST), and a USB test module (USB3TEST and USB2TEST). **Table 6.4-1** and **Table 6.4-2** list the functions in outline.

Table 6.4-1 Functions in Outline

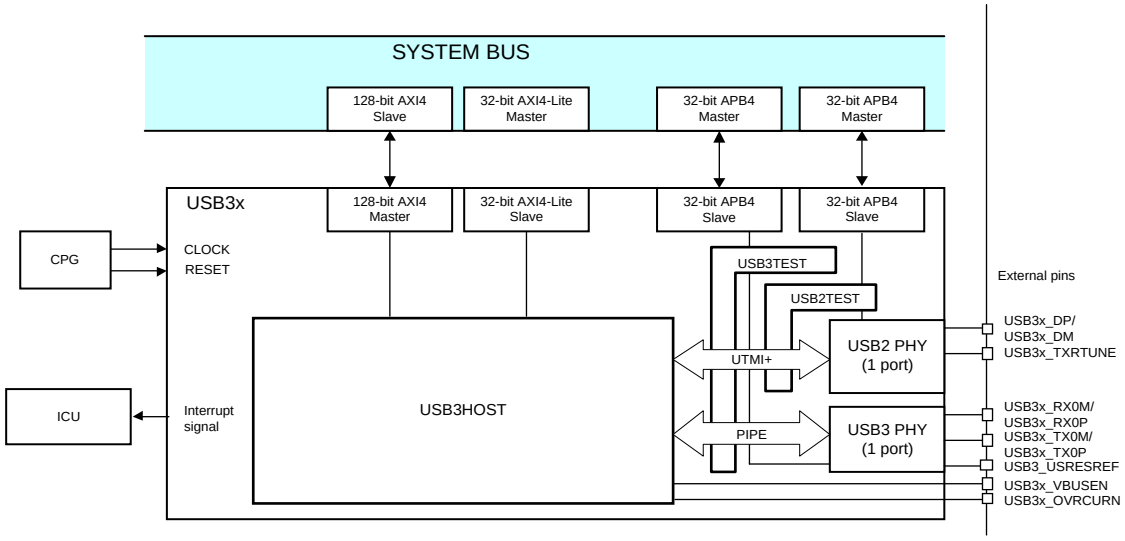
Item	Description
USB3HOST function	• Support for super-speed plus (10 Gbps), super-speed (5 Gbps), high-speed (480 Mbps), full-speed (12 Mbps), and low-speed (1.5 Mbps) transfer • Support for all transfer types: isochronous, interrupt, control, and bulk • Support for isochronous and interrupt high-band transfer • Compliant with the eXtensible Host Controller Interface(xHCI) Specification for USB
USB3TEST function	• Reset control • Control of PHY input pins • Monitor of PHY output pins

Table 6.4-2 Supported Speed Types

	Speed Type				
Function	Super Speed Plus (10 Gbps)	Super Speed (5 Gbps)	High Speed (480 Mbps)	Full Speed (12 Mbps)	Low Speed (1.5 Mbps)
Host Controller	✓	✓	✓	✓	✓

6.4.2 Connection Configuration

Figure 6.4-1 shows the connection configuration of this unit.



Note: PIPE: The PHY Interface for the PCI Express, SATA, and USB3.2
UTMI+: The USB2.0 Transceiver Macrocell Interface Plus
External Pins: USB3x_* means USB30_* for ch0.

Figure 6.4-1 USB Connection Configuration

6.4.3 Pin Functions

Table 6.4-3 lists the external pins of the USB3.

Table 6.4-3 List of External Pins

Pin Name*1	Input/Output	Function
USB3x_DP	I/O	USB2.0 USB D+ signal (positive)
USB3x_DM	I/O	USB2.0 USB D− signal (negative)
USB3x_TXRTUNE	—	Transmitter Resistor Tune Pin (for USB2PHY).
USB3x_RX0M	Input	USB3.2 super-speed plus differential reception pair (negative)
USB3x_RX0P	Input	USB3.2 super-speed plus differential reception pair (positive)
USB3x_TX0M	Output	USB3.2 super-speed plus differential transmission pair (negative)
USB3x_TX0P	Output	USB3.2 super-speed plus differential transmission pair (positive)
USB3_USRESREF	—	Reference resistor connection
USB3x_VBUSEN	Output	VBUS control signal (active high)
USB3x_OVRCURN	Input	Overcurrent detection (active low)

Note 1. For the details, see **4.2 Pin Function Controller (PFC)**.

6.4.4 USB3.2 Gen1 Host Controller (USB3HOST)

6.4.4.1 Introduction

The USB3.2 Gen1 Host Controller (USB3HOST) complies with the Universal Serial Bus (USB) 3.2 Specification.

6.4.4.1.1 Features

The functions of the USB3HOST are outlined below.

■ **USB functions**

The USB3HOST is compliant with the Universal Serial Bus 3.2 Specification Revision 1.0 and ECNs approved in Jan 25, 2021.

- Supports 1 downstream USB receptacles
  - Number of SSP Gen2 or SS ports: 1
  - Number of HS or FS or LS ports: 1
- Supports Super Speed Plus Gen2x1 (10 Gbps), Super Speed (5 Gbps), High Speed (480 Mbps), Full Speed (12 Mbps), and Low Speed (1.5 Mbps).
- Supports all transfer-types: Control, Bulk, Interrupt, Isochronous, and these split-transactions.
- Supports Power Control and Over Current Detection.
- Implements the following USB2.0 ECNs USB 2.0 Link Power Management Addendum and LPM functions that are compliant with the xHCI specification.
  - USB 2.0 Link Power Management Addendum Engineering Change Notice to the USB 2.0 specification as of July 16, 2007
  - Errata for USB 2.0 ECN: Link Power Management (LPM) — 7/2007 as of October 11, 2011

■ **Battery charging functions**

The USB3HOST supports the following functions that are compliant with the Battery Charging Specification Revision 1.2.

- The USB3HOST supports the following modes of chargeable port devices.
  - Standard Downstream Port (SDP) Mode
  - Charging Downstream Port (CDP) Mode
  - Dedicated Charging Port (DCP) Mode

■ xHCI capabilities

The USB3HOST has the following capabilities that are compliant with the eXtensible Host Controller Interface for the Universal Serial Bus Specification Revision 1.2.

- Supports 64-bit address capability
- xHCI Host Capability
  - Number of Device Slots (MaxSlots): 32
  - Number of Total Endpoints: 64
  - Number of Event Ring Segment Tables (ERST Max): 2 (1)
  - Supported Page Size: 4 Kbytes
  - Context Size (CSZ): 64 bytes (1)
  - Number of Interrupter (MaxIntrs): 1
  - Number of Stream IDs (MaxPSASize): 65536 (15)
  - Supports Primary Stream Array
  - Supports the USB Legacy Support.
  - Supports the Hardware LPM Capability.
  - Supports Best Effort Service Latency (BESL) LPM Capability.

■ Clock Control

The USB3HOST has the following clock and PLL control functions.

- Clock gating
  - Enables/disables clock gating to applicable functional sub-blocks depending on the USB state.
- Clock dividing:
  - Controls an internal clock frequency of USB3.2 PHY’s clock-domain by USB3.2
  - PHY’s rate-signal.
- PLL control
  - Enables/disables PLLs for USB2.0 and USB3.2 depending on their USB states.

6.4.4.2 Interrupts

6.4.4.2.1 Interrupt diagram

In the USB3HOST, the elements of interrupt are notified by the following layered structure.

- INTSTS register is not implemented in any FFs.
- This register is a read-only register (monitor register).

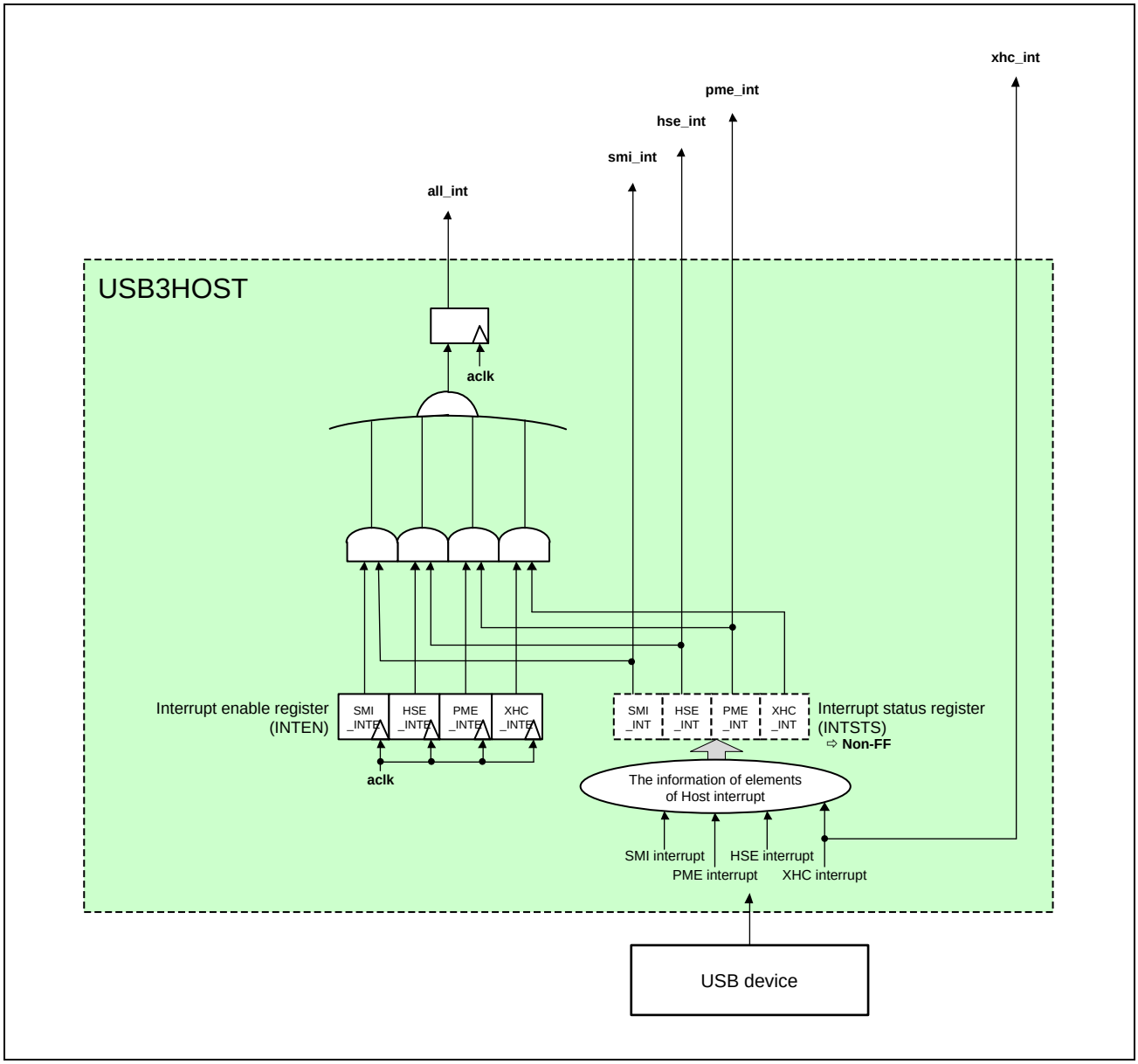


Figure 6.4-2 USB3HOST Interrupts Structure

6.4.4.2.2 Interrupt sources

The USB3HOST has the following interrupt sources.

Table 6.4-4 USB3HOST Interrupt Source Information

Interrupt Name	Type of Notification	Description
xhc_int	Level Pulse	xHC interrupt This signal notifies that Event Data TRB is set in Event Ring of a system memory. In this LSI, “level mode” is selected for the interrupt. Refer to section 4.17 of the xHCI specification for details.
pme_int	Level	Power management event interrupt. When PMCSR.PME Enable is set to 1b and PMCSR.PME Status is set to 1b, this signal is asserted. PMCSR.PME Status is set to 1b by one of the following conditions. • PORTSC.WCE = 1b, and Connected condition • PORTSC.WDE = 1b, and Disconnected condition • PORTSC.WOE = 1b, and Overcurrent condition • PORTSC.PLS = U3, and Remote Wakeup condition
hse_int	Level	Host system error interrupt When the USBCMD.HSEE is set to 1b and the USBSTS.HSE is set to 1b, this signal is asserted. The USBSTS.HSE is set to 1b by one of the following conditions. • Either SLVERR or DECERR is received as the Response in the transaction of AXI master write. • Either SLVERR or DECERR is received as the Response in the transaction of AXI master read.
smi_int	Level	System management interrupt This signal is asserted by one of the following conditions. • USBLEGCTLSTS.SEI = 1b and USBLEGCTLSTS.USE = 1b • USBLEGCTLSTS.SHSE = 1b and USBLEGCTLSTS.SHSEE = 1b • USBLEGCTLSTS.SOOC = 1b and USBLEGCTLSTS.SOOE = 1b • USBLEGCTLSTS.SPC = 1b and USBLEGCTLSTS.SPCE = 1b • USBLEGCTLSTS.SBA = 1b and USBLEGCTLSTS.SBAE = 1b
all_int	Level	Logical OR of all interrupt signals. This signal is asserted by one of the following conditions. • INTEN.XHC_INTE = 1b and INTSTS.XHC_INT = 1b • INTEN.PME_INTE = 1b and INTSTS.PME_INT = 1b • INTEN.HSE_INTE = 1b and INTSTS.HSE_INT = 1b • INTEN.SMI_INTE = 1b and INTSTS.SMI_INT = 1b

6.4.4.3 Register Specification

Registers in the USB3HOST are composed of the following register groups.

- xHCI compliant registers
  - Host Controller Capability Registers
  - Host Controller Operational Registers
  - Port Register Sets in Host Controller Registers
  - xHCI Extended Capabilities
  - Host Controller Runtime Registers
  - Doorbell Registers

- USB3HOST's specific registers
  - Core Control and Status Registers
  - Battery Charging Register

aclk is supplied to the USB3HOST from CPG, when the user reads or writes a valid value from/to these registers.

When the situations below occurred, the USB3HOST replies "SLVERR" to the AXI interconnect.

[Conditions in which the USB3HOST replies "SLVERR"]

- The case of access to the Renesas private registers in the register group "Renesas Private Registers".
- The case of access to the related CPC's registers (CRCRL, and CRCRH) or the related HEP's registers (IMAN, IMOD, ERSTSZ, ERSTBAL, ERSTBAH, ERDPL, and ERDPH) in the "Suspend" state of the USB Power State.

For the register base address (<USB3m_host_base>), see **Table 6.4-5**.

Table 6.4-5 Register Base Addresses

Unit Name	Base Address Name	Base Address
USB30 (Host)	<USB30_host_base>	0_1585_0000h (5585_0000h*1, 4585_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

6.4.4.3.1 Register Overview

- Operational Base = CAPLENGTH: 0020h
- U3 Port Register Offset: 0420h
- U2 Port Register Offset: 0420h + (CFGSTS1.U3NUM) × 10h
- Extended Capability Base = HCCPARAMS1.xECP << 2: 0500h
- USB Legacy Support Capability: 0500h
- xHCI Supported Protocol Capability for USB3.2: 0510h
- xHCI Supported Protocol Capability for USB2.0: 0530h
- xHCI Extended Power Management Capability: 0550h
- Runtime Base = RTSOFF: 0600h
- Doorbell Base= DBOFF: 0700h

6.4.4.3.2 Register List

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Capability Register Length and Interface Version Number Register	USB3m_HOST_CAPLENGTH_HCIVERSION	0120_0020h	0000h	32
Structural Parameters 1 Register	USB3m_HOST_HCSPARAMS1	0200_0120h	0004h	32
Structural Parameters 2 Register	USB3m_HOST_HCSPARAMS2	0000_0011h	0008h	32
Structural Parameters 3 Register	USB3m_HOST_HCSPARAMS3	0000_0000h	000Ch	32
Capability Parameter 1 Register	USB3m_HOST_HCCPARAMS1	0140_FE8Ch	0010h	32
Doorbell Offset Register	USB3m_HOST_DBOFF	0000_0700h	0014h	32
Runtime Register Space Offset Register	USB3m_HOST_RSTOFF	0000_0600h	0018h	32
Capability Parameter 2 Register	USB3m_HOST_HCCPARAMS2	0000_002Dh	001Ch	32
USB Command Register	USB3m_HOST_USBCMD	0000_0000h	0020h	32
USB Status Register	USB3m_HOST_USBSTS	0000_0001h	0024h	32
Page Size Register	USB3m_HOST_PAGESIZE	0000_0001h	0028h	32
Reserve	-	-	002Ch to 0033h	-
Device Notification Control Register	USB3m_HOST_DNCTRL	0000_0000h	0034h	32
Command Ring Control Low Register	USB3m_HOST_CRCRL	0000_0000h	0038h	32
Command Ring Control High Register	USB3m_HOST_CRCRH	0000_0000h	003Ch	32
Reserve	-	-	0040h to 004Fh	-
Device Context Base Address Array Pointer Low Register	USB3m_HOST_DCBAAPL	0000_0000h	0050h	32
Device Context Base Address Array Pointer High Register	USB3m_HOST_DCBAAPH	0000_0000h	0054h	32
Configure Register	USB3m_HOST_CONFIG	0000_0000h	0058h	32
Reserve	-	-	005Ch to 041Fh	-
U3P1 Port Status and Control Register	USB3m_HOST_U3P1PORTSC	0000_02A0h	0420h	32
U3P1 Port Power Management Status and Control Register	USB3m_HOST_U3P1PORTPMSC	0000_0000h	0424h	32
U3P1 Port Link Info Register	USB3m_HOST_USB_HOST_U3P1PORTLI	0000_0000h	0428h	32
U3P1 Port Hardware LPM Control Register	USB3m_HOST_U3P1PORTHLPMC	0000_0000h	042Ch	32
U2P1 Port Status and Control Register	USB3m_HOST_U2P1PORTSC	0000_02A0h	0430h	32
U2P1 Port Power Management Status and Control Register	USB3m_HOST_U2P1PORTPMSC	0000_0000h	0434h	32
U2P1 Port Link Info Register	USB3m_HOST_U2P1PORTLI	0000_0000h	0438h	32
U2P1 Port Hardware LPM Power Control Register	USB3m_HOST_U2P1PORTHLPMC	0000_0000h	043Ch	32
Reserve	-	-	0440h to 04FFh	-
USB Legacy Support Capability Register	USB3m_HOST_USBLEGSUP	0000_0401h	0500h	32
USB Legacy Support Control and Status Register	USB3m_HOST_USBLEGCTLSTS	0000_0000h	0504h	32
Reserve	-	-	0508h to 050Fh	-
Offset 00h - xHCI Supported Protocol Capability Field Definitions (USB3.2) Register	USB3m_HOST_U3HCSPC1	0320_0802h	0510h	32
Offset 04h - xHCI Supported Protocol Capability Field Definitions (USB3.2) Register	USB3m_HOST_U3HCSPC2	2042_5355h	0514h	32
Offset 08h - xHCI Supported Protocol Capability Field Definitions (USB3.2) Register	USB3m_HOST_U3HCSPC3	0000_0101h	0518h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Offset 0Ch - xHCI Supported Protocol Capability Field Definitions (USB3.2) Register	USB3m_HOST_U3HCSPC4	0000_0000h	051Ch	32
Protocol Speed ID for SS Register	USB3m_HOST_PSISS	0005_0134h	0520h	32
Protocol Speed ID for SSP (PSISSP)	USB3m_HOST_RP	000A_0135h	0524h	32
Reserve	-	-	0528h to 052Fh	-
Offset 00h - xHCI Supported Protocol Capability Field Definitions (USB2.0) Register	USB3m_HOST_U2HCSPC1	0200_0802h	0530h	32
Offset 04h - xHCI Supported Protocol Capability Field Definitions (USB2.0) Register	USB3m_HOST_U2HCSPC2	2042_5355h	0534h	32
Offset 08h - xHCI Supported Protocol Capability Field Definitions (USB2.0) Register	USB3m_HOST_U2HCSPC3	0018_0102h	0538h	32
Offset 0Ch - xHCI Supported Protocol Capability Field Definitions (USB2.0) Register	USB3m_HOST_U2HCSPC4	0000_0000h	053Ch	32
Protocol Speed ID for FS Register	USB3m_HOST_PSIFS	000C_0021h	0540h	32
Protocol Speed ID for LS Register	USB3m_HOST_PSILS	05DC_0012h	0544h	32
Protocol Speed ID for HS Register	USB3m_HOST_PSIHS	01E0_0023h	0548h	32
Reserve	-	-	054Ch to 054Fh	-
Power Management Capabilities Register	USB3m_HOST_PMC	4803_0003h	0550h	32
Power Management Control/Status Register	USB3m_HOST_PMCSR	0000_0008h	0554h	32
Reserve	-	-	0558h to 05FFh	-
Microframe Index Register	USB3m_HOST_MFINDEX	0000_0000h	0600h	32
Reserve	-	-	0604h to 061Fh	-
Interrupter Management Register	USB3m_HOST_IMAN	0000_0000h	0620h	32
Interrupter Moderation Register	USB3m_HOST_IMOD	0000_0FA0h	0624h	32
Event Ring Segment Table Size Register	USB3m_HOST_ERSTSZ	0000_0000h	0628h	32
Reserve	-	-	062Ch to 062Fh	-
Event Ring Segment Table Base Address Low Register	USB3m_HOST_ERSTBAL	0000_0000h	0630h	32
Event Ring Segment Table Base Address High Register	USB3m_HOST_ERSTBAH	0000_0000h	0634h	32
Event Ring Dequeue Pointer Low Register	USB3m_HOST_ERDPL	0000_0000h	0638h	32
Event Ring Dequeue Pointer High Register	USB3m_HOST_ERDPH	0000_0000h	063Ch	32
Reserve	-	-	0640h to 06FFh	-
Host Controller Doorbell Register	USB3m_HOST_HCD	0000_0000h	0700h	32
Device Context Doorbell Register (Slot #n)	USB3m_HOST_DCDn	0000_0000h	0700h + n x 0004h	32
Reserve	-	-	0784h to 0FFFh	-
Revision ID Register	USB3m_HOST_REVID	E302_0100h	1000h	32
Configuration Status1 Register	USB3m_HOST_CFGSTS1	0x29_4111h	1004h	32
Configuration Status2 Register	USB3m_HOST_CFGSTS2	0000_1388h	1008h	32
Configuration Status3 Register	USB3m_HOST_CFGSTS3	0011_0111h	100Ch	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Configuration Status4 Register	USB3m_HOST_CFGSTS4	0000_0021h	1010h	32
Configuration Status5 Register	USB3m_HOST_CFGSTS5	0000_0000h	1014h	32
Configuration Status6 Register	USB3m_HOST_CFGSTS6	0000_0129h	1018h	32
Configuration Status7 Register	USB3m_HOST_CFGSTS7	0000_0000h	101Ch	32
Configuration Status8 Register	USB3m_HOST_CFGSTS8	0000_0000h	1020h	32
Configuration Status9 Register	USB3m_HOST_CFGSTS9	0000_0000h	1024h	32
Configuration Status10 Register	USB3m_HOST_CFGSTS10	0006_0030h	1028h	32
Data Buffer Threshold Control Register	USB3m_HOST_DBUFTH	1111_0000h	102Ch	32
Core Control Register	USB3m_HOST_CORECTRL	0000_2000h	1030h	32
PHY Control Register	USB3m_HOST_PHYCTRL	0000_0000h	1034h	32
PHY Status Register	USB3m_HOST_PHYSTS	0000_0000h	1038h	32
Renesas Private Register2	USB3m_HOST_RP2	000x_xxxxh	103Ch	32
Interrupt Status Register	USB3m_HOST_INTSTS	0000_0000h	1040h	32
Interrupt Enable Register	USB3m_HOST_INTEN	0000_0000h	1044h	32
Reserve	-	-	1048h to 104Fh	-
Frame Length Adjustment Register (FLADJ) & Serial Bus Release Number Register	USB3m_HOST_FLADJ_SBRN	00xx_2032h	1050h	32
Reserve	-	-	1054h to 107Fh	-
Battery Charging Control Register	USB3m_HOST_BCCTRL	0000_0000h	1080h	32
Reserve	-	-	1084h to 10BFh	-
PIPE Status and Control Register 0	USB3m_HOST_U3P0PIPESC0	3013_2000h	10C0h	32
PIPE Status and Control Register 1	USB3m_HOST_U3P0PIPESC1	0016_2000h	10C4h	32
PIPE Status and Control Register 2	USB3m_HOST_U3P0PIPESC2	3015_0000h	10C8h	32
PIPE Status and Control Register 3	USB3m_HOST_U3P0PIPESC3	3013_2000h	10CCh	32
PIPE Status and Control Register 4	USB3m_HOST_U3P0PIPESC4	0018_0000h	10D0h	32

6.4.4.4 Register Descriptions

The function description of each register is given below.

The prefix (USB3m_HOST_) of the register names is omitted in the register descriptions and the field descriptions in this section.

The prefix (USB3m_HOST_) means below (m = 0):

- USB30_HOST_*: USB30 registers

6.4.4.4.1 Host Controller Capability Registers

(1) Capability Register Length and Interface Version Number Register (USB3m_HOST_CAPLENGTH_HCIVERSION)

This register indicates the capability register length and the interface version number.

Access Size : 32 bits
Address : <USB3m_host_base> + 0000h
Initial Value : 0120_0020h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HCIVERSION[15:0]															
Initial Value	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CAPLENGTH[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	HCIVERSION [15:0]	120h	R	This is a two-byte register containing a BCD encoding of the xHCI specification revision number supported by this host controller. The most significant byte of this register represents a major revision and the least significant byte is the minor revision. (Ex) 0120h corresponds to xHCI version 1.2.
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	CAPLENGTH [7:0]	20h	R	This register is used as an offset to add to register base to find the beginning of the Operational Register Space.

(2) Structural Parameters 1 Register (USB3m_HOST_HCSPARAMS1)

This register is the structural parameter register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0004h
Initial Value : 0200_0120h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Number of Ports[7:0]								-	-	-	-	-	Number of Interrupters[10:8]		
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Number of Interrupters[7:0]								Number of Device Slots[7:0]							
Initial Value	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Number of Ports[7:0]	2h	R	This field specifies the maximum Port Number value, i.e. the highest numbered Port Register Sets that are addressable in the Operational Register Space. Valid values are in the range of 1h to FFh. The value in this field shall reflect the maximum Port Number value assigned by an xHCI Supported Protocol Capability, described in section 7.2 of xHCI Specification. Software shall refer to these capabilities to identify whether a specific Port Number is valid, and the protocol supported by the associated Port Register Set.
23 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 8	Number of Interrupters [10:0]	1h	R	This field specifies the number of Interrupters implemented on this host controller. Each Interrupter may be allocated to MSI or MSI-X vector and controls its generation and moderation. The value of this field determines how many Interrupter Register Sets are addressable in the Runtime Register Space. Valid values are in the range of 1h to 400h. A 0h in this field is undefined
7 to 0	Number of Device Slots[7:0]	20h	R	This field specifies the maximum number of Device Context Structures and Doorbell Array entries this host controller can support. Valid values are in the range of 1 to 255. The value of 0b is reserved.

(3) Structural Parameters 2 Register (USB3m_HOST_HCSPARAMS2)

This register is the structural parameter register 2.

Access Size : 32 bits
Address : <USB3m_host_base> + 0008h
Initial Value : 0000_0011h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Max Scratchpad Buffers (Max Scratchpad Bufs Lo)[4:0]					Scratchpad Restore(SPR)	Max Scratchpad Buffers (Max Scratchpad Bufs Hi)[4:0]					-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	Event Ring Segment Table Max (ERST Max)[3:0]				Isochronous Scheduling Threshold (IST)[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	Max Scratchpad Buffers (Max Scratchpad Bufs Lo)[4:0]	0h	R	Default = implementation dependent. Valid values for Max Scratchpad Buffers (Hi and Lo) are 0 to 1023. This field indicates the lower-order 5 bits of the number of Scratchpad Buffers system software reserves for the xHC.
26	Scratchpad Restore(SPR)	0h	R	Default = implementation dependent. If Max Scratchpad Buffers is greater than 0 then this flag indicates whether the xHC uses the Scratchpad Buffers for saving state when executing Save and Restore State operations. If Max Scratchpad Buffers is = 0b, then this flag is 0b. The value 1 indicates that the xHC requires the integrity of the space for the Scratchpad Buffer to be maintained across power events. The value 0 indicates that the space for the Scratchpad Buffer can freed and reallocated between power events.
25 to 21	Max Scratchpad Buffers (Max Scratchpad Bufs Hi)[4:0]	0h	R	Default = implementation dependent. This field indicates the higher-order 5 bits of the number of Scratchpad Buffers system software reserves for the xHC.
20 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 4	Event Ring Segment Table Max (ERST Max)[3:0]	1h	R	Default = implementation dependent. Valid values are 0 to 15. This field determines the maximum value supported the Event Ring Segment Table Base Size registers, where: The maximum number of Event Ring Segment Table entries = 2 ERST Max. e.g. if the ERST Max = 7, then the xHC Event Ring Segment Table(s) supports up to 128 entries, 15 then 32K entries, etc.
3 to 0	Isochronous Scheduling Threshold (IST)[3:0]	1h	R	Default = implementation dependent. The value in this field indicates to system software the minimum distance (in time) that it is required to stay ahead of the host controller while adding TRBs, in order to have the host controller process them at the correct time. The value shall be specified in terms of number of frames/microframes. If bit [3] of the IST is cleared to 0, software can add a TRB no later than IST[2:0] Microframes before that TRB is scheduled to be executed. If bit [3] of the IST is set to 1, a TRB is added by software no later than IST[2:0] Frames before that TRB is scheduled to be executed.

(4) Structural Parameters 3 Register (USB3m_HOST_HCSPARAMS3)

This register is structural parameter register 3.

Access Size : 32 bits
Address : <USB3m_host_base> + 000Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	U2 Device Exit Latency[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	U1 Device Exit Latency[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	U2 Device Exit Latency[15:0]	0h	R	Worst case latency to transition from U2 to U0. Applies to all root hub ports. The following are permissible values: 0000h: Zero 0001h: Less than 1 µs 0002h: Less than 2 µs ⋮ 07FFh: Less than 2047 µs 0800h to FFFFh: Reserved
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	U1 Device Exit Latency[7:0]	0h	R	Worst case latency to transition a root hub Port Link State (PLS) from U1 to U0. Applies to all root hub ports. The following are permissible values: 00h: Zero 01h: Less than 1 µs 02h: Less than 2 µs ⋮ 0Ah: Less than 10 µs 0Bh to FFh: Reserved

(5) Capability Parameter 1 Register (USB3m_HOST_HCCPARAMS1)

This register is the capability parameter register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0010h
Initial Value : 0140_FE8Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	xHCI Extended Capabilities Pointer (xECP)[15:0]															
Initial Value	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Maximum Primary Stream Array Size (MaxPSASize)[3:0]				Contiguous Frame ID Capability (CFC)	Stopped EDTLA Capability (SEC)	Stopped Short Packet Capability (SPC)	Parse All Event Data (PAE)	No Secondary SID Support (NSS)	Latency Tolerance Messaging Capability (LTC)	Light HC Reset Capability (LHRC)	Port Indicators (PIND)	Port Power Control (PPC)	Context Size (CSZ)	BW Negotiation Capability (BNC)	64-bit Addressing Capability (AC64)
Initial Value	1	1	1	1	1	1	1	0	1	0	0	0	1	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	xHCI Extended Capabilities Pointer (xECP)[15:0]	140h	R	This field indicates the existence of a capabilities list. The value of this field indicates a relative offset, in 32-bit words, from Base to the beginning of the first extended capability. For example, using the offset of Base is 1000h and the xECP value of 0068h, we can calculate the following effective address of the first extended capability: 1000h + (0068h << 2) -> 1000h + 01A0h -> 11A0h
15 to 12	Maximum Primary Stream Array Size (MaxPSASize) [3:0]	Fh	R	This field identifies the maximum size Primary Stream Array that the xHC supports. The Primary Stream Array size = 2MaxPSASize+1. Valid MaxPSASize values are 0 to 15, where 0 indicates that streams are not supported
11	Contiguous Frame ID Capability (CFC)	1h	R	This flag indicates that the host controller implementation is capable of matching the Frame ID of consecutive Isoch TDs.
10	Stopped EDTLA Capability (SEC)	1h	R	This flag indicates that the host controller implementation Stream Context supports a Stopped EDTLA field. Stopped EDTLA Capability support (i.e. SEC = '1') shall be mandatory for all xHCI 1.2 compliant xHCs.
9	Stopped Short Packet Capability (SPC)	1h	R	This flag indicates that the host controller implementation is capable of generating a Stopped Short Packet Completion Code.
8	Parse All Event Data (PAE)	0h	R	This flag indicates whether the host controller implementation Parses all Event Data TRBs while advancing to the next TD after a Short Packet, or it skips all but the first Event Data TRB. A 1 in this bit indicates that all Event Data TRBs are parsed. A 0 in this bit indicates that only the first Event Data TRB is parsed.
7	No Secondary SID Support (NSS)	1h	R	This flag indicates whether the host controller implementation supports Secondary Stream IDs. A 1 in this bit indicates that Secondary Stream ID decoding is not supported. A 0 in this bit indicates that Secondary Stream ID decoding is supported.
6	Latency Tolerance Messaging Capability (LTC)	0h	R	This flag indicates whether the host controller implementation supports Latency Tolerance Messaging (LTM). A 1 in this bit indicates that LTM is supported. A 0 in this bit indicates that LTM is not supported.
5	Light HC Reset Capability (LHRC)	0h	R	This flag indicates whether the host controller implementation supports a Light Host Controller Reset. A 1 in this bit indicates that Light Host Controller Reset is supported. A 0 in this bit indicates that Light Host Controller Reset is not supported. The value of this flag affects the functionality of the Light Host Controller Reset (LHCRST) flag in the USBCMD register.

Bit	Bit Name	Initial Value	R/W	Description
4	Port Indicators (PIND)	0h	R	This bit indicates whether the xHC root hub ports support port indicator control. When this bit is 1, the port status and control registers include a readable/writeable field for controlling the state of the port indicator.
3	Port Power Control (PPC)	1h	R	This flag indicates whether the host controller implementation includes port power control. A 1 in this bit indicates the ports have port power switches. A 0 in this bit indicates the port do not have port power switches. The value of this flag affects the functionality of the PP flag in each port status and control register.
2	Context Size (CSZ)	1h	R	If this bit is set to 1b, then the xHC uses 64 byte Context data structures. If this bit is cleared to 0b, then the xHC uses 32 byte Context data structures. Note: This flag does not apply to Stream Contexts
1	BW Negotiation Capability (BNC)	0h	R	This flag identifies whether the xHC has implemented the Bandwidth Negotiation. Values for this flag have the following interpretation: 0b: BW Negotiation is not implemented. 1b: BW Negotiation is implemented.
0	64-bit Addressing Capability (AC64)	0h	R	This flag documents the addressing range capability of this implementation. The value of this flag determines whether the xHC has implemented the higher-order 32 bits of a 64-bit register and data structure pointer fields. Values for this flag have the following interpretation: 0b: 32-bit address memory pointers are implemented 1b: 64-bit address memory pointers are implemented If 32-bit address memory pointers are implemented, the xHC ignores the higher-order 32 bits of a 64-bit data structure pointer fields, and system software ignores the higher-order 32 bits of a 64-bit xHC registers.

(6) Doorbell Offset Register (USB3m_HOST_DBOFF)

This register indicates the doorbell array offset.

Access Size : 32 bits
Address : <USB3m_host_base> + 0014h
Initial Value : 0000_0700h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Doorbell Array Offset[29:14]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Doorbell Array Offset[13:0]														-	-
Initial Value	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	Doorbell Array Offset[29:0]	1C0h	R	Default = implementation dependent. This field defines the offset in Dwords of the base address of the Doorbell Array from main base address (i.e. the base address of the xHCI Capability register address space).
1, 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(7) Runtime Register Space Offset Register (USB3m_HOST_RSTOFF)

This register indicates the 32-byte offset of the xHCI Runtime Registers from the Base.

Access Size : 32 bits

Address : <USB3m_host_base> + 0018h

Initial Value : 0000_0600h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Runtime Register Space Offset[26:11]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Runtime Register Space Offset[10:0]											-	-	-	-	-
Initial Value	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	Runtime Register Space Offset[26:0]	30h	R	Default = implementation dependent. This field defines the 32-byte offset of the xHCI Runtime Registers from the Base. i.e. Runtime Register Base Address = Base + Runtime Register Set Offset.
4 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(8) Capability Parameter 2 Register (USB3m_HOST_HCCPARAMS2)

This register is capability parameter register 2.

Access Size : 32 bits

Address : <USB3m_host_base> + 001Ch

Initial Value : 0000_002Dh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	Configuration Information Capability (CIC)	Large ESIT Payload Capability (LEC)	Compliance Transition Capability (CTC)	Force Save Context Capability (FSC)	Configure Endpoint Command Max Exit Latency Too Large Capability (CMC)	U3 Entry Capability (U3C)
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	Configuration Information Capability (CIC)	1h	R	This bit indicates if the xHC supports extended Configuration Information. When this bit is 1b, the Configuration Value, Interface Number, and Alternate Setting fields in the Input Control Context are supported. When this bit is 0b, the extended Input Control Context fields are not supported.
4	Large ESIT Payload Capability (LEC)	0h	R	This bit indicates whether the xHC supports ESIT Payloads greater than 48K bytes. When this bit is 1b, ESIT Payloads greater than 48K bytes are supported. When this bit is 0b, ESIT Payloads greater than 48K bytes are not supported.
3	Compliance Transition Capability (CTC)	1h	R	This bit indicates whether the xHC USB3 Root Hub ports support the Compliance Transition Enabled (CTE) flag. When this bit is 1b, transition of the port state machine of a USB3 Root Hub to the Compliance substate shall be explicitly enabled by software. When this bit is 0b, transition of the port state machine of a USB3 Root Hub to the Compliance substate are automatically enabled.
2	Force Save Context Capability (FSC)	1h	R	This bit indicates whether the xHC supports the Force Save Context Capability. When this bit is 1b, the Save State operation shall save any cached Slot, Endpoint, Stream or other Context information to memory.
1	Configure Endpoint Command Max Exit Latency Too Large Capability (CMC)	0h	R	This bit indicates whether a Configure Endpoint Command is capable of generating a Max Exit Latency Too Large Capability Error. When this bit is 1b, a Max Exit Latency Too Large Capability Error can be returned by a Configure Endpoint Command. When this bit is 0b, a Max Exit Latency Too Large Capability Error shall not be returned by a Configure Endpoint Command. This capability is enabled by the CME flag in the USBCMD register.
0	U3 Entry Capability (U3C)	1h	R	This bit indicates whether the xHC Root Hub ports support port Suspend Complete notification. When this bit is 1b, PLC shall be asserted on any transition of PLS to the U3 State.

6.4.4.4.2 Host Controller Operational Registers

(1) USB Command Register (USB3m_HOST_USBCMD)

This register issues USB commands.

Access Size : 32 bits
Address : <USB3m_host_base> + 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	CEM Enable (CME)	Stopped – Short Packet Enable (SPE)	Enable U3 MFINDEX Stop (EU3S)	Enable Wrap Event (EWE)	Controller Restore State (CRS)	Controller Save State (CSS)	Light Host Controller Reset (LHCRST)	-	-	-	Host System Error Enable (HSEE)	Interrupter Enable (INTE)	Host Controller Reset (HCRST)	Run/Stop (R/S)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	CEM Enable (CME)	0h	RW	When set to 1b, a Max Exit Latency Too Large Capability Error can be returned by a Configure Endpoint Command. When cleared to 0b, a Max Exit Latency Too Large Capability Error shall not be returned by a Configure Endpoint Command. This bit is Reserved if CMC = '0b'.
12	Stopped – Short Packet Enable (SPE)	0h	RW	When set to 1b, the xHC can generate a Stopped Short Packet Completion Code. This bit is Reserved if SPC = '0b'.
11	Enable U3 MFINDEX Stop (EU3S)	0h	RW	Default = 0b. When set to 1b, the xHC can stop the MFINDEX counting action if all Root Hub ports are in the U3, Disconnected, Disabled, or Powered-off state. When cleared to 0b, the xHC can stop the MFINDEX counting action if all Root Hub ports are in the Disconnected, Disabled, Training, or Powered-off state. The USB3HOST does not implement this function.
10	Enable Wrap Event (EWE)	0h	RW	Default = 0b. When set to 1b, the xHC generates a MFINDEX Wrap Event every time the MFINDEX register transitions from 0_3FFFh to 0_0000h. When cleared to 0b, no MFINDEX Wrap Events are generated
9	Controller Restore State (CRS)	0h	RW	Default = 0b. When set to 1b, and HCHalted (HCH) = 1b, then the xHC performs a Restore State operation and restores its internal state. When set to 1b and Run/Stop (R/S) = 1b or HCHalted (HCH) = 0b, or when cleared to 0b, no Restore State operation is performed. This flag always returns 0b when read. Refer to the Restore State Status (RSS) flag in the USBSTS register for information on Restore State completion. Note that undefined behavior can occur if a Restore State operation is initiated while Save State Status (SSS) = 1b.
8	Controller Save State (CSS)	0h	RW	Default = 0b. When 1b is written by software and HCHalted (HCH) = 1b, then the xHC saves any internal state that can be restored by a subsequent Restore State operation. When 1b is written by software and HCHalted (HCH) = 0b, or 0b is written, no Save State operation is performed. This flag always returns 0b when read. Refer to the Save State Status (SSS) flag in the USBSTS register for information on Save State completion. Note that undefined behavior can occur if a Save State operation is initiated while Restore State Status (RSS) = 1b.
7	Light Host Controller Reset (LHCRST)	0h	R	The Default = Controller Light HC Reset Capability (LHRC) bit in the HCCPARAMS1 register is formed on completion of the save state. Note that the xHC does not affect the state of the ports. When system software reads this bit, the value 0 indicates that the Light Host Controller Reset has been completed and it is safe for software to re-initialize the xHC. The value 1 indicates that the Light Host Controller Reset has not yet been completed. If not implemented, reading of this flag always returns a "0b".

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	Host System Error Enable (HSEE)	0h	RW	Default = 0b. When this bit is 1b, and the HSE bit in the USBSTS register is 1b, the xHC asserts the out-of-band error signal for the host. The signal is acknowledged by software clearing the HSE bit.
2	Interrupter Enable (INTE)	0h	RW	Default = 0b. This bit provides system software with a means of enabling or disabling the host system interrupts generated by Interrupters. When this bit is 1b, then Interrupter host system interrupt generation is allowed, e.g. the xHC issues an interrupt at the next interrupt threshold if the host system interrupt mechanism (e.g. MSI, MSI-X, etc.) is enabled. The interrupt is acknowledged by a host system interrupt specific mechanism.
1	Host Controller Reset (HCRST)	0h	RW	Default = 0b. This control bit is used by software to reset the host controller. The effects of this bit on the xHC and the Root Hub registers are similar to those of a Chip Hardware Reset. When 1b is written to this bit by software, the Host Controller resets its internal pipelines, timers, counters, state machines, etc. to their initial values. Any transaction currently in progress on the USB is immediately terminated. A USB reset is not driven on USB2 downstream ports, however a Hot or Warm Reset is initiated on USB3 Root Hub downstream ports. PCI Configuration registers are not affected by this reset. All operational registers, including port registers and port state machines are set to their initial values. Software reinitializes the host controller in order to return the host controller to an operational state. This bit is cleared to 0b by the Host Controller when the reset process is complete. Software cannot terminate the reset process early by writing 0b to this bit and does not write any xHC Operational or Runtime registers until while HCRST is 1b. Note that the completion of the xHC reset process is not gated by the Root Hub port reset process. This bit is not set to 1b by software when the HCHalted (HCH) bit in the USBSTS register is 0b. Attempting to reset an actively running host controller can result in undefined behavior.
0	Run/Stop (R/S)	0h	RW	Default = 0b. 1b = Run. 0b = Stop. When set to 1b, the xHC proceeds with execution of the schedule. The xHC continues execution as long as this bit is set to 1b. When this bit is cleared to 0b, the xHC completes any current or queued commands or TDs, and any USB transactions associated with them, then halt. The xHC halts within 16 ms. After the Run/Stop bit is cleared by software if the above conditions have been met. The HCHalted (HCH) bit in the USBSTS register indicates when the xHC has finished its pending pipelined transactions and has entered the stopped state. 1b is not written by software to this flag unless the xHC is in the Halted state (i.e. HCH in the USBSTS register is 1b). Doing so can yield undefined results. Writing 0b to this flag when the xHC is in the Running state (i.e. HCH = 0b) and any Event Rings are in the Event Ring Full state can result in lost events.

(2) USB Status Register (USB3m_HOST_USBSTS)

This register is the USB status register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0024h
Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	Host Controller Error (HCE)	Controller Not Ready (CNR)	Save/Restore Error (SRE)	Restore State Status (RSS)	Save State Status (SSS)	-	-	-	Port Change Detect (PCD)	Event Interrupt (EINT)	Host System Error (HSE)	-	HCHalted (HCH)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	RW1	R	R	R	R	R	RW1	RW1	RW1	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12	Host Controller Error (HCE)	0h	R	Default = 0b. 0b = No internal xHC error conditions exist and 1b = Internal xHC error condition. This flag shall be set to indicate detection of an internal error condition, which requires a reset and reinitialization of the xHC by software.
11	Controller Not Ready (CNR)	0h	R	Default = 0b. 0b = Ready and 1b = Not Ready. Doorbell or Operational register of the xHC, other than the USBSTS register, are not written by software while CNR = 0b. This flag is set by the xHC after a Chip Hardware Reset and cleared when the xHC is ready to begin accepting register writes. This flag remains cleared (0) until the next Chip Hardware Reset.
10	Save/Restore Error (SRE)	0h	RW1	Default = 0b. If an error occurs during a Save or Restore operation, this bit is set to 1b. This bit is cleared to 0b when a Save or Restore operation is initiated or when written with 1b.
9	Restore State Status (RSS)	0h	R	Default = 0b. When the Controller Restore State (CRS) flag in the USBCMD register is written with 1b, this bit is set to 1b and remains 1b while the xHC restores its internal state. When the Restore State operation is complete, this bit is cleared to 0b.
8	Save State Status (SSS)	0h	R	Default = 0b. When the Controller Save State (CSS) flag in the USBCMD register is written with 1b, this bit is set to 1b and remains 1b while the xHC saves its internal state. When the Save State operation is complete, this bit is cleared to 0b.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	Port Change Detect (PCD)	0h	RW1	Default = 0b. The xHC sets this bit to 1b when a bit of any port has a state transition from 0b to 1b. This bit can be maintained in the Aux Power as well. Alternatively, loading this bit with the OR of all of the PORTSC change bits on a D3 to D0 transition of the xHC is also acceptable. This bit provides system software an efficient means of determining if there has been Root Hub port activity. It is cleared by writing 1b. Writing 0b has no effect.
3	Event Interrupt (EINT)	0h	RW1	Default = 0b. The xHC sets this bit to 1b in response to a transition of the Interrupt Pending (IP) bit from any interrupt source from 0b to 1b. Software that uses EINT clears it prior to clearing any IP flags. A race condition can occur if software clears the IP flags then clears the EINT flag, and another IP 0 to 1 transition occurs between these operations. In this case the new transition of the IP bit shall be lost. It is cleared by writing 1b. Writing 0b has no effect.
2	Host System Error (HSE)	0h	RW1	Default = 0b. The xHC sets this bit to 1b when a serious error is detected, either internal to the xHC or during a host system access involving the xHC module. In a PCI system, conditions that lead to this bit being set to 1b include a PCI Parity error, PCI Master Abort, and PCI Target Abort. When this error occurs, the xHC clears the Run/Stop (R/S) bit in the USBCMD register to prevent further execution of the scheduled TDs. If the HSEE bit in the USBCMD register is 1, the xHC also asserts the out-of-band error signal to the host. It is cleared by writing 1b. Writing 0b has no effect.

Bit	Bit Name	Initial Value	R/W	Description
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	HCHalted (HCH)	1h	R	Default = 1b. This bit is 0b whenever the Run/Stop (R/S) bit is set to 1b. The xHC sets this bit to 1b after it has stopped executing as a result of the Run/Stop (R/S) bit being cleared to 0b, either by software or by the xHC hardware (e.g. internal error). If this bit is 1b, then SOFs, microSOFs, or Isochronous Timestamp Packets (ITP) is not generated by the xHC, and any received Transaction Packet is dropped.

(3) Page Size Register (USB3m_HOST_PAGESIZE)

This register indicates the page size supported by the xHC implementation.

Access Size : 32 bits
Address : <USB3m_host_base> + 0028h
Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Page Size[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	Page Size[15:0]	1h	R	Default = Implementation defined. This field defines the page size supported by the xHC implementation. This xHC supports a page size of 2^(n+12) if bit n is set. For example, if bit 0b is set, the xHC supports 4-Kbyte page sizes. For a Virtual Function, this register reflects the page size selected in the System Page Size field of the SR-IOV Extended Capability structure. For the Physical Function 0, this register reflects the implementation dependent default xHC page size. Various xHC resources reference PAGESIZE to describe their minimum alignment requirements. The maximum possible page size is 128M.

(4) Device Notification Control Register (USB3m_HOST_DNCTRL)

This register controls the device notification.

Access Size : 32 bits
Address : <USB3m_host_base> + 0034h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Notification Enable[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	Notification Enable[15:0]	0h	RW	When a Notification Enable bit is set, a Device Notification Event is generated when a Device Notification Transaction Packet is received with the matching value in the Notification Type field. For example, setting N1 to 1 enables Device Notification Event generation if a Device Notification TP is received with its Notification Type field set to 1 (FUNCTION_WAKE), etc.

(5) Command Ring Control Low Register (USB3m_HOST_CRCRL)

This register contains the lower-order bits of the initial value of the 64-bit command ring dequeue pointer and the command ring control bits.

Access Size : 32 bits
Address : <USB3m_host_base> + 0038h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Command Ring Pointer Low[25:10]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Command Ring Pointer Low[9:0]										-	-	Command Ring Running (CRR)	Command Abort (CA)	Command Stop (CS)	Ring Cycle State (RCS)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	RW1	RW1	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	Command Ring Pointer Low[25:0]	0h	RW	Default = 0b. This field defines lower-order bits of the initial value of the 64-bit Command Ring Dequeue Pointer. Writing to this field is ignored when Command Ring Running (CRR) = 1b. In writing to the CRCR (combination of CRCRL and CRCRH) while the Command Ring is stopped (CRR = 0b), the value of this field is used to fetch the first Command TRB the next time writing to the Host Controller Doorbell register proceeds with the DB Reason field set to Host Controller Command. If the CRCR is not written while the Command Ring is stopped (CRR = 0b), then the Command Ring begins fetching Command TRBs at the current value of the internal xHC Command Ring Dequeue Pointer. Reading this field always returns 0b.
5, 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	Command Ring Running (CRR)	0h	R	Default = 0b. This flag is set to 1b if the Run/Stop (R/S) bit is 1 and the Host Controller Doorbell register is written with the DB Reason field set to Host Controller Command. It is cleared to 0b when the Command Ring is “stopped” after writing 1b to the Command Stop (CS) or Command Abort (CA) flags, or if the R/S bit is cleared to 0b.
2	Command Abort (CA)	0h	RW1	Default = 0b. Writing a 1b to this bit immediately terminates the command currently being executed, stops the Command Ring, and generates a Command Completion Event with the Completion Code set to Command Ring Stopped. The next write to the Host Controller Doorbell with DB Reason field set to Host Controller Command shall restart the Command Ring operation. Writing to this flag is ignored by the xHC if Command Ring Running (CRR) = 0b. Reading this bit always returns 0b.
1	Command Stop (CS)	0h	RW1	Default = 0b. Writing a 1b to this bit stops the operation of the Command Ring after completion of the command currently being executed, and generates a Command Completion Event with the Completion Code set to Command Ring Stopped and the Command TRB Pointer set to the current value of the Command Ring Dequeue Pointer. The next write to the Host Controller Doorbell with DB Reason field set to Host Controller Command shall restart the Command Ring operation. Writing to this flag is ignored by the xHC if Command Ring Running (CRR) = 0b. Reading this bit always returns 0b.
0	Ring Cycle State (RCS)	0h	RW	This bit identifies the value of the xHC Consumer Cycle State (CCS) flag for the TRB referenced by the Command Ring Pointer. Writing to this flag is ignored if Command Ring Running (CRR) is 1b. In case of writing to the CRCR while the Command Ring is stopped (CRR = 0b), the value of this flag is used to fetch the first Command TRB the next time writing to the Host Controller Doorbell register proceeds with the DB Reason field set to Host Controller Command. If the CRCR is not written while the Command Ring is stopped (CRR = 0b), then the Command Ring begins fetching Command TRBs using the current value of the internal Command Ring CCS flag. Reading this flag always returns 0b.

(6) Command Ring Control High Register (USB3m_HOST_CRCRH)

This register defines higher-order bits of the initial value of the 64-bit Command Ring Dequeue Pointer.

Access Size : 32 bits
Address : <USB3m_host_base> + 003Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Command Ring Pointer High[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Command Ring Pointer High[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Command Ring Pointer High[31:0]	0h	RW	Default = 0b. This field defines higher-order bits of the initial value of the 64-bit Command Ring Dequeue Pointer. Writing to this field is ignored when Command Ring Running (CRR) = 1b. In case of writing to the CRCR while the Command Ring is stopped (CRR = 0b), the value of this field is used to fetch the first Command TRB the next time writing to the Host Controller Doorbell register proceeds with the DB Reason field set to Host Controller Command. If the CRCR is not written while the Command Ring is stopped (CRR = 0b) then the Command Ring begins fetching Command TRBs at the current value of the internal xHC Command Ring Dequeue Pointer. Reading this field always returns 0b.

(7) Device Context Base Address Array Pointer Low Register (USB3m_HOST_DCBAAPL)

This register defines lower-order bits of the 64-bit base address of the Device Context Pointer Array.

Access Size : 32 bits
Address : <USB3m_host_base> + 0050h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Device Context Base Address Array Pointer Low[25:10]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Device Context Base Address Array Pointer Low[9:0]										-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	Device Context Base Address Array Pointer Low[25:0]	0h	RW	This field defines lower-order bits of the 64-bit base address of the Device Context Pointer Array. A table of address pointers that reference Device Context structures for the devices is attached to the host.
5 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(8) Device Context Base Address Array Pointer High Register (USB3m_HOST_DCBAAPH)

This register defines higher-order bits of the 64-bit base address of the Device Context Pointer Array.

Access Size : 32 bits
Address : <USB3m_host_base> + 0054h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Device Context Base Address Array Pointer High[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Device Context Base Address Array Pointer High[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Device Context Base Address Array Pointer High[31:0]	0h	RW	This field defines higher-order bits of the 64-bit base address of the Device Context Pointer Array. A table of address pointers that reference Device Context structures for the devices is attached to the host.

(9) Configure Register (USB3m_HOST_CONFIG)

This register configures USB3HOST.

Access Size : 32 bits
Address : <USB3m_host_base> + 0058h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Configuration Information Enable (CIE)	U3 Entry Enable (U3E)	Max Device Slots Enabled (MaxSlotsEn)[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	Configuration Information Enable (CIE)	0h	RW	When set to 1b, the software shall initialize the Configuration Value, Interface Number, and Alternate Setting fields in the Input Control Context when it is associated with a Configure Endpoint Command. When this bit is 0b, the extended Input Control Context fields are not supported.
8	U3 Entry Enable (U3E)	0h	RW	When set to 1b, the xHC shall assert the PLC flag (1b) when a Root Hub port transitions to the U3 State.
7 to 0	Max Device Slots Enabled (MaxSlotsEn) [7:0]	0h	RW	Default = 0b. This field specifies the maximum number of enabled Device Slots. Valid values are in the range of 0 to MaxSlots. Enabled Devices Slots are allocated contiguously. E.g.) A value of 16 specifies that Device Slots 1 to 16 are active. A value of 0 disables all Device Slots. A disabled Device Slot shall not respond to Doorbell Register references. This field shall not be modified by software if the xHC is running (Run/Stop (R/S) = '1b').

6.4.4.4.3 Port Register Sets in Host Controller Operational Registers

(1) U3P1 Port Status and Control Register (USB3m_HOST_U3P1PORTSC)

This register contains the USB 3.2 port status and control bits.

Access Size : 32 bits

Address : <USB3m_HOST_base> + 0420h

Initial Value : 0000_02A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Warm Port Reset (WPR)	Device Removable (DR)	-	-	Wake on Overcurrent Enable (WOE)	Wake on Disconnect Enable (WDE)	Wake on Connect Enable (WCE)	Cold Attach Status (CAS)	Port Config Error Change (CEC)	Port Link State Change (PLC)	Port Reset Change (PRC)	Over-current Change (OCC)	Warm Port Reset Change (WRC)	Port Enable/Disable Change (PEC)	Connect Status Change (CSC)	Port Link State Write Strobe (LWS)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	R	R	R	RW	RW	RW	R	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Port Indicator Control (PIC)[1:0]		Port Speed[3:0]				Port Power (PP)	Port Link State (PLS)[3:0]				Port Reset (PR)	Over-current Active (OCA)	-	Port Enable/ Disable (PED)	Current Connect Status (CCS)
Initial Value	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0
R/W	RW	RW	R	R	R	R	RW	RW	RW	RW	RW	RW1	R	R	RW1	R

Bit	Bit Name	Initial Value	R/W	Description
31	Warm Port Reset (WPR)	0h	RW1	1b: Starts Warm Reset sequence, and sets the PR bit to 1b.
30	Device Removable (DR)	0h	R	1b: A device connected to this Port is non-removable. 0b: A device connected to this Port is removable.
29, 28	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
27	Wake on Overcurrent Enable (WOE)	0h	RW	1b: Enables over-current condition as System Wake-up Event The USB3HOST can write this bit in the case of both “PED = 1b” and “PED = 0b”
26	Wake on Disconnect Enable (WDE)	0h	RW	1b: Enables device disconnection as System Wake-up Event The USB3HOST can write this bit in the case of both “PED = 1b” and “PED = 0b”.
25	Wake on Connect Enable (WCE)	0h	RW	1b: Enables device connection as System Wake-up Event The USB3HOST can write this bit in the case of both “PED = 1b” and “PED = 0b”.
24	Cold Attach Status (CAS)	0h	R	1b: Far-end Receiver Termination is detected but Port Status cannot transition to Enabled State. This bit is fixed to “0b” in the USB3HOST.
23	Port Config Error Change (CEC)	0h	RW1	This bit is set when configuration with Link Partner fails.
22	Port Link State Change (PLC)	0h	RW1	0b: No Change 1b: Link state is changed. This bit is set in the case of below Link transitions. U3 -> Resume Resume -> Recovery -> U0 U3 -> Recovery ->U0 Any state -> Inactive (In the case of U3E = 1b) Any state -> U3 When 1b is written by software, the bit is cleared.

Bit	Bit Name	Initial Value	R/W	Description
21	Port Reset Change (PRC)	0h	RW1	This bit is set when Port reset process completes. (Port Reset changes from 1b to 0b) But if Port reset process is finished in below cases, this bit is not set to "1b". When SW sets PP bit to "0b" When SW sets PED bit to "1b" When the Over Current is occurred 0b: No Change 1b: Reset is complete. Writing 1b by software clears this bit. When 1b is written by software, this bit is cleared.
20	Over-current Change (OCC)	0h	RW1	When Over-current Active changes from "0b" to "1b", this bit is set to 1b. When 1b is written by software, this bit is cleared.
19	Warm Port Reset Change (WRC)	0h	RW1	This bit is set to 1b when a Warm Reset is completed
18	Port Enable/Disable Change (PEC)	0h	RW1	0b: No Change 1b: Port enabled/disabled status has changed. This bit for a port is set when the port is a USB2 protocol port and is disabled due to exit under the appropriate conditions at the EOF2 point. When 1b is written by software, this bit is cleared. If Port Power is 0b, this bit is set to 0b. U3P[m]PORTSC.PEC is fixed to "0b".
17	Connect Status Change (CSC)	0h	RW1	0b: No Change 1b: Change in Current Connect Status The xHC sets this bit to 1b at change of port device connection status. When 1b is written by software, this bit is cleared. If Port Power is 0b, this bit is set to 0b.
16	Port Link State Write Strobe (LWS)	0h	RW	1b: Writing to PLS field is enabled. 0b: Writing to PLS field is ignored. The read value is 0
15, 14	Port Indicator Control (PIC)[1:0]	0h	RW	If Port Indicator (PIND) bit in HCCPRAMS1 Register is 0, writing to this field has no effect. If the Port Indicator (PIND) bit is 1b: 00b: Port indicators are off 01b: Amber 10b: Green 11b: Undefined If Port Power is 0b, this field is set to 0b.
13 to 10	Port Speed[3:0]	0h	R	Speed type of USB device connected to the port. This is enabled only when a device is connected to the port. (Current Connect Status is 1b) After a port reset, this field indicates the speed of the device connected to the port. If the reset or speed detection fails, this field indicates an unknown speed. When the Current Connect Status changes from 0b to 1b, this field indicates an undefined speed. 0000b: Undefined speed (before speed detection) 0001b to 0011b: Reserved 0100b: SuperSpeed Gen1x1 device is attached. 0101b: SuperSpeed Plus Gen2x1 device is attached. 0110b: SuperSpeed Plus Gen1x2 device is attached (not supported). 0111b: SuperSpeed Plus Gen2x2 device is attached (not supported). 1000b to 1111b: Reserved
9	Port Power (PP)	1h	RW	This field reflects a port's logical power state. 0b: This port is in the Powered-off state. (USB3m_VBUSEN is low.) 1b: This port is not in the Powered-off state. (USB3m_VBUSEN is high.) When the Port Power Control (PPC) bit in the HCCPARAMS1 Register is 1b, the xHC has a port power control switch and this bit shows the current state of the switch. (0b: off, 1b: on) When the Port Power Control (PPC) bit in the HCCPARAMS1 Register is set to "0b", the xHC has no port power control switch and each port is hard wired to power and not affected by this bit. When the USB3m_OVRCURN pin detects low within a fixed interval, the PP bit is changed from 1 to 0. After modifying PP, software shall read PP and confirm that it is reached its target state before modifying it again (page 312 of the xHCI specification). PP flag can be delayed in reflecting this change, (Ex) due to waiting for a port related state machine to complete reset signaling or other operation. (page 315 of the xHCI specification)

Bit	Bit Name	Initial Value	R/W	Description
8 to 5	Port Link State (PLS)[3:0]	5h	RW	This field is used for port power management and shows the Current Link state. When the PED is 1b, System software sets this bit then sets Link U state. Port Link State is not set to Disabled, RxDetect, or Inactive State. 0000b: The link makes a transition to the U0 state from any U state. 0001b: Ignored. 0010b: Ignored. 0011b: The link makes a transition to the U3 state from any U state. 0100b: Ignored. 0101b: When PLS is set to the Disabled and PP is set to "1", the link makes a transition to the Disconnected. 0110b to 1001b: Ignored. 1010b: The link makes a transition to the Compliance mode. 1011b to 1111b: Ignored. To write to this field, the Port Link State Write strobe should be 1. Software reads this field to determine the success or failure of a transition to the U2 state. Write 0: Deasserts L1 signal. Write 1: No influence 0000b: Link is in the U0 State 0001b: Link is in the U1 State 0010b: Link is in the U2 State 0011b: Link is in the U3 State (Device Suspended) 0100b: Link is in the Disabled State 0101b: Link is in the RxDetect State 0110b: Link is in the Inactive State 0111b: Link is in the Polling State 1000b: Link is in the Recovery State 1001b: Link is in the Hot Reset State 1010b: Link is in the Compliance Mode State 1011b: Link is in the Test State 1100 to 1110b: Reserved 1111b: Link is in the Resume State If Port Power is 0, this field is undefined. Transitions between different states are not reflected until the transition is complete. (page 311 of the xHCI specification)
4	Port Reset (PR)	0h	RW1	0b: Port is not being reset. 1b: Port reset signal is being asserted. When software sets this bit from 0b to 1b, the bus reset sequence is started. This bit remains set to 1b until a reset by root hub is completed. When an error is detected during resetting, Port Speed field indicates an undefined speed. If Port Power is 0b, this field is set to 0b.
3	Over-current Active (OCA)	0h	R	0b: Not over-current condition 1b: Over-current condition When an over-current condition is removed, this bit changes from 1b to 0b automatically
2	-	0h	R	Reserved These bits are read as 0b. The write value should always be 0b.
1	Port Enable/Disable (PED)	0h	RW1	0b: Disabled 1b: Enabled After detecting a connection at a port, if SS port initialization or reset by the system software is successful, this bit is automatically enabled by the xHC. The Port is disabled by fault condition (disconnection event or other fault condition), or USB System Software. The bit status does not change until port status changes. If Port Power is 0b, this field is set to 0b.
0	Current Connect Status (CCS)	0h	R	0b: No device is present 1b: A device is present on port. Reflects current port status. If Port Power is 0b, this field is set to 0b.

(2) U3P1 Port Power Management Status and Control Register (USB3m_HOST_U3P1PORTPMSC)

This register contains the USB 3.2 port power management status and control bits.

Access Size : 32 bits
Address : <USB3m_host_base> + 0424h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Force Link PM Accept (FLA)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	U2 Timeout[7:0]								U1 Timeout[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	Force Link PM Accept (FLA)	0h	RW	0b -> 1b: St Link Function LMP with FLA = 1b is generated. 1b -> 0b: Set Link Function LMP with FLA = 0b is generated. This bit is set to 0 when PR changes from “0b” to “1b” or CCS changes from “0b” to “1b”. While PP is set to “0”, writing action to this bit is ignored.
15 to 8	U2 Timeout[7:0]	0h	RW	U2 inactivity timer Timeout value If FFh, a port is in the U2 entry disabled state. When PR bit is changed from “0b” to “1b”, this bit is initialized. 00h: Zero (default) 01h: 256 µs 02h: 512 µs ⋮ FEh: 65.024 ms FFh: Infinite
7 to 0	U1 Timeout[7:0]	0h	RW	U1 inactivity timer Timeout value If FFh, port is in the U1 entry disabled state. When PR bit is changed from “0b” to “1b”, this bit is initialized. 00h: Zero (default) 01h: 1 µs 02h: 2 µs ⋮ 7Fh: 127 µs 80h to FEh: Reserved FFh: Infinite

(3) U3P1 Port Link Info Register (USB3m_HOST_U3P1PORTLI)

This register indicates the USB 3.2 port link information.

Access Size : 32 bits
Address : <USB3m_host_base> + 0428h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	Tx Lane Count (TLC)[3:0]				Rx Lane Count (RLC)[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Link Error Count[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 20	Tx Lane Count (TLC)[3:0]	0h	R	This field that identifies the number of Transmit Lanes negotiated by the port. This is a “zerobased” value, where 0 to 15 represents Lane Counts of 1 to 16, respectively. This value is valid only when CCS = ‘1b’. TLC shall equal ‘0b’ for a simplex Sublink. This core’s value is fixed to “0h”.
19 to 16	Rx Lane Count (RLC)[3:0]	0h	R	This field that identifies the number of Receive Lanes negotiated by the port. This is a “zerobased” value, where 0 to 15 represents Lane Counts of 1 to 16, respectively. This value is valid only when CCS = ‘1b’. RLC shall equal ‘0b’ for a simplex Sublink. This core’s value is fixed to “0h”.
15 to 0	Link Error Count[15:0]	0h	RW	This field returns the number of link errors detected by the port. This value shall be reset to ‘0b’ by the assertion of a Chip Hardware Reset, HCRST, when PR transitions from ‘1b’ to ‘0b’, or when reset by software by writing 0 to it. This register will saturate at max and will increment by one each time a port transitions from U0 to Recovery to recover an error event.

(4) U3P1 Port Hardware LPM Control Register (USB3m_HOST_U3P1PORTHLPMC)

This register is reserved for USB 3.2. The Hardware LPM Control Register is used for USB 2.0 (**(8) U2P1 Port Hardware LPM Power Control Register (USB3m_HOST_U2P1PORTHLPMC)**).

Access Size : 32 bits
Address : <USB3m_host_base> + 042Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	—[31:0]	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(5) U2P1 Port Status and Control Register (USB3m_HOST_U2P1PORTSC)

This register contains the USB 2.0 port status and control bits.

Access Size : 32 bits
Address : <USB3m_host_base> + 0430h
Initial Value : 0000_02A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	Device Removable (DR)	-	-	Wake on Overcurrent Enable (WOE)	Wake on Disconnect Enable (WDE)	Wake on Connect Enable (WCE)	Cold Attach Status (CAS)	-	Port Link State Change (PLC)	Port Reset Change (PRC)	Over-current Change (OCC)	-	Port Enable/Disable Change (PEC)	Connect Status Change (CSC)	Port Link State Write Strobe (LWS)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	R	R	RW1	RW1	RW1	R	RW1	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Port Indicator Control (PIC)[1:0]		Port Speed[3:0]				Port Power (PP)	Port Link State (PLS)[3:0]				Port Reset (PR)	Over-current Active (OCA)	-	Port Enable/ Disable (PED)	Current Connect Status (CCS)
Initial Value	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0
R/W	RW	RW	R	R	R	R	RW	RW	RW	RW	RW	RW1	R	R	RW1	R

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30	Device Removable (DR)	0h	R	1b: A device connected to this Port is non-removable. 0b: A device connected to this Port is removable. In the USB3HOST, this bit is fixed to 0b to indicate that all USB2.0 ports are removable.
29, 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27	Wake on Overcurrent Enable (WOE)	0h	RW	Write 1b: Enables over-current condition as System Wake-up Event The USB3HOST can write this bit in the case of both “PED = 1b” and “PED = 0b”.
26	Wake on Disconnect Enable (WDE)	0h	RW	Write 1b: Enables device disconnection as System Wake-up Event The USB3HOST can write this bit in the case of both “PED = 1b” and “PED = 0b”.
25	Wake on Connect Enable (WCE)	0h	RW	Write 1b: Enables device connection as System Wake-up Event The USB3HOST can write this bit in the case of both “PED = 1b” and “PED = 0b”.
24	Cold Attach Status (CAS)	0h	R	This bit is fixed to 0b for USB2 protocol ports.
23	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	Port Link State Change (PLC)	0h	RW1	0b: No Change 1b: Port Link State has changed. This flag is set to 1b due to the following PLS transitions: Refer to **Table 6.4-6**. When 1b is written by software, the bit is cleared
21	Port Reset Change (PRC)	0h	RW1	This bit is set when Port reset process completes. (Port Reset changes from 1 to 0) 0b: No Change 1b: Reset is complete. When 1b is written by software, the bit is cleared
20	Over-current Change (OCC)	0h	RW1	When Over-current Active changes, this bit is set to 1b. When 1b is written by software, the bit is cleared.
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
18	Port Enable/Disable Change (PEC)	0h	RW1	0b: No Change 1b: Port enabled/disabled has changed. This bit for a port is set when the port is a USB2 protocol port and is disabled due to exit under the appropriate conditions at the EOF2 point. When 1b is written by software, the bit is cleared. Note: This flag is not set if the PED transition was due to software setting PP to “0b”.
17	Connect Status Change (CSC)	0h	RW1	0b: No Change 1b: Change in Current Connect Status The xHC sets this bit to 1b at change of CCS-bit’s status. When 1b is written by software, the bit is cleared. Note: This flag shall not be set if the CCS transition was due to software setting PP to ‘0b’.
16	Port Link State Write Strobe (LWS)	0h	RW1	1b: Writing to PLS field is enabled. 0b: Writing to PLS field is ignored. The read value is 0
15, 14	Port Indicator Control (PIC)[1:0]	0h	RW	If Port Indicator (PIND) bit in HCCPRAMS1 Register is 0b, writing to this field has no effect. If the Port Indicator (PIND) bit is 1b: 00b: Port indicators are off 01b: Amber 10b: Green 11b: Undefined In the USB3HOST, this bit is fixed to 0b.
13 to 10	Port Speed[3:0]	0h	R	Speed type of USB device connected to the port. This is enabled only when a device is connected to the port (Current Connect Status is 1b). After a port reset, this field indicates the speed of the device connected to the port. If the reset or speed detection fails, this field indicates an unknown speed. When the Current Connect Status changes from 0b to 1b, this field indicates an undefined speed. 0000b: Undefined speed (before speed detection) 0001b: Full-speed device attached 0010b: Low-speed device attached 0011b: High-speed device attached 0100b to 1111b: Reserved
9	Port Power (PP)	1h	RW	This field reflects a port’s logical power state. 0b : This port is in the Powered-off state. (USB3m_VBUSEN is low.) 1b : This port is not in the Powered-off state. (USB3m_VBUSEN is high.) When the Port Power Control (PPC) bit in the HCCPARAMS1 Register is 1b, the xHC has a port power control switch and this bit shows the current state of the switch. (0b: off, 1b: on) When the Port Power Control (PPC) bit in the HCCPARAMS1 Register is set to “0b”, the xHC has no port power control switch and each port is hard wired to power and not affected by this bit. When the USB3m_OVRCURN pin detects low within a fixed interval, the PP bit of each port is changed from 1b to 0b. After the HC reset releases, this bit is set from 0b to 1b automatically. This bit is set to 1b when in the Compliance Test Mode.

Bit	Bit Name	Initial Value	R/W	Description
8 to 5	Port Link State (PLS)[3:0]	5h	RW	This field is used for port power management and shows the Current Link state. When the PED is 1b, System software sets this bit then sets Link U state. Port Link State is not set to Disabled, RxDetect, or Inactive State. Write 0000b: The link makes a transition to the U0 state from any U state. Write 0001b: Ignored. Write 0010b: Requests LPM and asserting L1 signal on the USB2 bus. The link should enter U2 state if device response ACK. Write 0011b: The link makes a transition to the U3 state from any U state. Write 0100b to 1110b: Ignored. Write 1111b: xHC initial resume when port in U3 state. Writing to this field sets the Port Link State Write strobe 1. For USB2 Protocol port, writing 2 requests LPM. Software reads this field to determine success/failure of transition to U2 state. Write 0b: Deasserts L1 signal Write 1b: No influence Read 0000b: Link is in the Enabled / WLPM State Read 0001b: Reserved Read 0010b: Link is in the L1Suspend State Read 0011b: Link is in the Suspended State Read 0100b: Link is in the Powered-off State Read 0101b: Link is in the Disconnected State Read 0110b: Reserved Read 0111b: Link is in the Disabled State Read 1000b: Reserved Read 1001b: Link is in the Hot Reset State Read 1010b: Reserved Read 1011b: Link is in the Test Mode State Read 1100b to 1110b: Reserved Read 1111b: Link is in the Resuming / L1Resuming / SendEOR State If Port Power is set to “0b”, this field is undefined.
4	Port Reset (PR)	0h	RW1	0b: port is not in reset 1b: port reset assertion When this bit is set from 0b to 1b by software, the bus reset sequence is started. This bit remains to be set to 1b until a reset by root hub is completed. When an error is detected during resetting, Port Speed field indicates an undefined speed. If Port Power is 0b, this field is set to 0b.
3	Over-current Active (OCA)	0h	R	0b: Over-current condition 1b: Not over-current condition When an over-current condition is removed, this bit changes from 1b to 0b automatically.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	Port Enable/ Disable (PED)	0h	RW1	0b: Disable 1b: Enable This bit is set to 1b when the USB2.0 port enters U0 state. If Port Power is 0b, this field is set to 0b.
0	Current Connect Status (CCS)	0h	R	0b: No device is present. 1b: A device is present on port. Reflects current port status. If Port Power is 0b, this field is set to 0b.

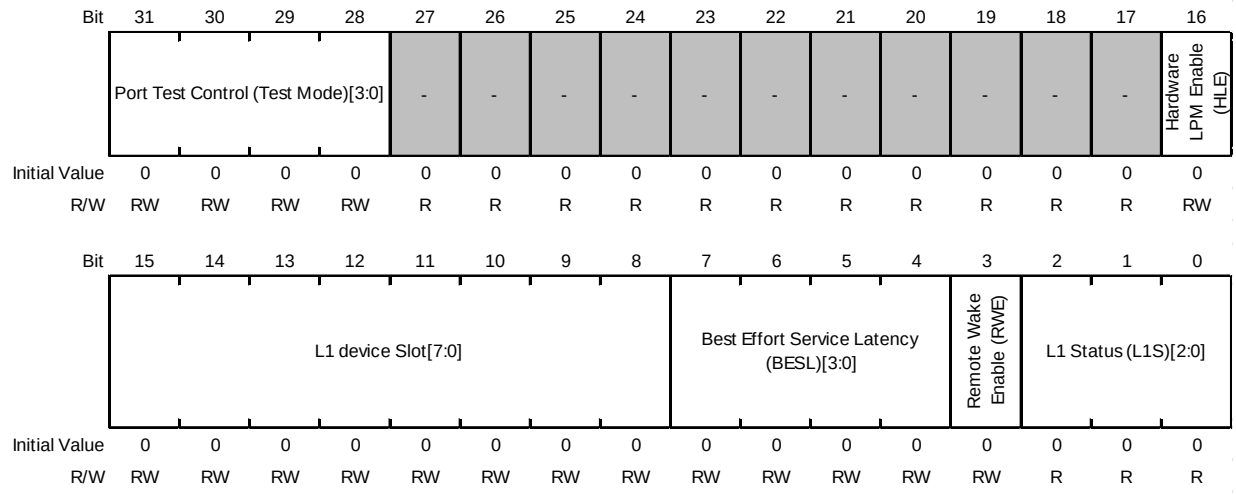
Table 6.4-6 PLS Transition

Transition	Condition
U3 -> Resume	Wakeup signal from a device
Resume -> U0	Host/Device Remote Resume complete
U2 -> U0	L1 Resume complete
U0 -> U0	L1 Entry Reject

(6) U2P1 Port Power Management Status and Control Register (USB3m_HOST_U2P1PORTPMSC)

This register contains the USB 2.0 port power management status and control bits.

Access Size : 32 bits
Address : <USB3m_host_base> + 0434h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 28	Port Test Control (Test Mode)[3:0]	0h	RW	When this field is '0b', the port is NOT operating in a test mode. A non-zero value indicates that it is operating in test mode and the specific test mode is indicated by the specific value. A nonzero Port Test Control value is only valid to a port that is in the Powered-Off state (PLS = Disabled). If the port is not in this state, the xHC shall respond with the Port Test Control field set to Port Test Control Error. Refer to section 4.19.6 of xHCI 1.2 specification for the operational model for using these test modes. The encoding of the Test Mode bits for a USB2 protocol port are : 0000b: Test mode not enabled 0001b: Test J_STATE 0010b: Test K_STATE 0011b: Test SE0_NAK 0100b: Test Packet 0101b: Test FORCE_ENABLE 0110b to 1110b: Reserved 1111b: Port Test Control Error Refer to sections 7.1.20 and 11.24.2.13 of the USB2 spec for more information on Test Modes.
27 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	Hardware LPM Enable (HLE)	0h	RW	If this bit is set to '1b', then hardware controlled LPM shall be enabled for this port. Refer to section 4.23.5.1.1.1 of xHCI 1.2 specification. If the USB2 Hardware LPM Capability is not supported (HLC = '0b') this field shall be RsvdZ. Note the BESL LMP Capability support (i.e. HLE = '1b' and BLC = '1b') shall be mandatory for all xHCI 1.2 compliant xHCs.
15 to 8	L1 device Slot[7:0]	0h	RW	System software sets this field to indicate the ID of the Device Slot associated with the device directly attached to the Root Hub port. A value of '0' indicates no device is present. The xHC uses this field to lookup information necessary to generate the LPM Token packet.
7 to 4	Best Effort Service Latency (BESL)[3:0]	0h	RW	System software sets this field to indicate to the recipient device how long the xHC will drive resume if it (the xHC) initiates an exit from L1. The BESL value encoding is defined in Table 13 of xHCI 1.2 specification. Note: The BESL field is used by both software and hardware controlled LPM. Refer to section 4.23.5.1.1 of xHCI 1.2 for more information on BESL use. Refer to section 5.2.5 of xHCI 1.2 for information on how DBESL can be used to establish an initial value for BESL.
3	Remote Wake Enable (RWE)	0h	RW	System software sets this flag to enable or disable the device for remote wake from L1. The value of this flag shall temporarily (while in L1) override the current setting of the Remote Wake feature set by the standard Set/ClearFeature() commands defined in *Universal Serial Bus Specification, revision 2.0, Chapter 9*.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	L1 Status (L1S)[2:0]	0h	R	This field is used by software to determine whether an L1-based suspend request (LPM transaction) was successful, specifically: 000b: Invalid – This field shall be ignored by software. 001b: Success – Port successfully transitioned to L1 (ACK) 010b: Not Yet – Device is unable to enter L1 at this time (NYET) 011b: Not Supported – Device does not support L1 transitions (STALL) 100b: Timeout/Error – Device failed to respond to the LPM Transaction or an error occurred 101b to 111b: Reserved The value of this field is only valid when the port resides in the L0 or L1 state (PLS = '0' or '2'). Refer to section 4.23.5.1.1 of xHCI 1.2 for more information.

(7) U2P1 Port Link Info Register (USB3m_HOST_U2P1PORTLI)

This register is reserved for USB 2.0. the Port Link Info Register is used for USB 3.2 (**(3) U3P1 Port Link Info Register (USB3m_HOST_U3P1PORTLI)**).

Access Size : 32 bits
Address : <USB3m_host_base> + 0438h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	—[31:0]	0h	R	Reserved These bits are read as 0b. The write value should always be 0b.

(8) U2P1 Port Hardware LPM Power Control Register (USB3m_HOST_U2P1PORTHLPMC)

This register control USB 2.0 port Hardware LPM Power.

Access Size : 32 bits
Address : <USB3m_host_base> + 043Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Best Effort Service Latency Deep (BESLD)[3:0]				L1 Timeout[7:0]								Host Initiated Resume Duration Mode (HIRDM)[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 10	Best Effort Service Latency Deep (BESLD)[3:0]	0h	RW	System software sets this field to indicate to the recipient device how long the xHC will drive resume on an exit from U2. Refer to section 4.23.5.1.1.1 of xHCI 1.2 for more information on BESLD use. The BESLD value encoding is defined in Table 13 of xHCI 1.2. Refer to section 5.2.6 of xHCI 1.2 for information on how DBESLD can be used to establish an initial value for BESLD.
9 to 2	L1 Timeout[7:0]	0h	RW	Timeout value for the L1 inactivity timer (LPM Timer). This field shall be set to 00h by the assertion of PR to '1'. Refer to section 4.23.5.1.1.1 of xHCI 1.2 for more information on L1 Timeout operation. The following are permissible values: 00h: 128 µs (default) 01h: 256 µs 02h: 512 µs 03h: 768 µs ⋮ FFh: 65.280 ms
1, 0	Host Initiated Resume Duration Mode (HIRDM)[1:0]	0h	RW	Indicates which HIRD value should be used. The following are permissible values: 00b: Initiate L1 using BESL only on timeout. (default) 01b: Initiate L1 using BESLD on timeout. If rejected by device, initiate L1 using BESL. 10b, 11b: Reserved

6.4.4.4.4 xHCI Extended Capabilities

(1) USB Legacy Supported Protocol Capability

(a) USB Legacy Support Capability Register (USB3m_HOST_USBLEGSUP)

USB legacy support capability register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0500h
Initial Value : 0000_0401h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	HC OS Owned Semaphore (HOOS)	-	-	-	-	-	-	-	HC BIOS Owned Semaphore (HBOS)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Next Capability Pointer[7:0]								Capability ID[7:0]							
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	HC OS Owned Semaphore (HOOS)	0h	RW	System software sets this bit to request ownership of the xHC. Ownership is obtained when this bit is read as 1b and the HC BIOS Owned Semaphore bit is read as 0b.
23 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	HC BIOS Owned Semaphore (HBOS)	0h	RW	The BIOS sets this bit to establish ownership of the xHC. System BIOS will set this bit to a 0b in response to a request for ownership of the xHC by system software.
15 to 8	Next Capability Pointer[7:0]	4h	R	This field indicates the location of the next capability relative to the effective address of the current capability.
7 to 0	Capability ID[7:0]	1h	R	Legacy Support

(b) USB Legacy Support Control and Status Register (USB3m_HOST_USBLEGCTLSTS)

USB legacy support control and status register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0504h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SMI on BAR (SBA)	SMI on PCI Comm and (SPC)	SMI on OS Ownership Change (SOOC)	-	-	-	-	-	-	-	-	SMI on Host System Error (SHSE)	-	-	-	SMI on Event Interrupt (SEI)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	RW1	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SMI on BAR Enable (SBAE)	SMI on PCI Command Enable (SPCE)	SMI on OS Ownership Enable (SOOE)	-	-	-	-	-	-	-	-	SMI on Host System Error Enable (SHSEE)	-	-	-	USB SMI Enable (USE)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	R	R	R	R	R	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	SMI on BAR (SBA)	0h	RW1	This bit is set to 1b whenever the Base Address Register (BAR) is written. This bit is fixed to 0b, because the USB3HOST does not have the Base Address Register.
30	SMI on PCI Command (SPC)	0h	RW1	This bit is set to 1b whenever the PCI Command Register is written. This bit is fixed to 0b, because the USB3HOST does not have the PCI Command Register.
29	SMI on OS Ownership Change (SOOC)	0h	RW1	This bit is set to 1b whenever the HC OS Owned Semaphore bit in the USBLEGSUP register transitions from 1b to 0b or 0b to 1b.
28 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	SMI on Host System Error (SHSE)	0h	R	Shadow bit of the Host System Error (HSE) bit in the USBSTS register.
19 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	SMI on Event Interrupt (SEI)	0h	R	Shadow bit of Event Interrupt (EINT) bit in the USBSTS register.
15	SMI on BAR Enable (SBAE)	0h	RW	When this bit is 1b and SMI on BAR is 1b, then the host controller will issue an SMI.
14	SMI on PCI Command Enable (SPCE)	0h	RW	When this bit is 1b and SMI on PCI Command is 1b, then the host controller will issue an SMI.
13	SMI on OS Ownership Enable (SOOE)	0h	RW	When this bit is 1b and the OS Ownership Change bit is 1b, the host controller will issue an SMI
12 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	SMI on Host System Error Enable (SHSEE)	0h	RW	When this bit is 1b and the SMI on Host System Error bit (below) in this register is 1b, the host controller will issue an SMI immediately.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	USB SMI Enable (USE)	0h	RW	When this bit is 1b and the SMI on Event Interrupt bit in this register is 1b, the host controller will issue an SMI immediately.

(2) xHCI Supported Protocol Capability for USB3.2

(a) Offset 00h - xHCI Supported Protocol Capability Field Definitions (USB3.2) Register (USB3m_HOST_U3HCSPC1)

xHCI supported protocol capability field definitions (USB3.2) register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0510h
Initial Value : 0320_0802h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Major Revision[7:0]								Minor Revision[7:0]							
Initial Value	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Next Capability Pointer[7:0]								Capability ID[7:0]							
Initial Value	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Major Revision[7:0]	3h	R	Major Specification Release Number in Binary-Coded Decimal (i.e., version 3.x is 03h). This field identifies the major release number component of the specification with which the xHC is compliant.
23 to 16	Minor Revision[7:0]	20h	R	Minor Specification Release Number in Binary-Coded Decimal (i.e., version x.10 is 10h). This field identifies the minor release number component of the specification with which the xHC is compliant.
15 to 8	Next Capability Pointer[7:0]	8h	R	This field indicates the location of the next capability with relative to the effective address of the current capability.
7 to 0	Capability ID[7:0]	2h	R	xHCI Supported Protocol

(b) Offset 04h - xHCI Supported Protocol Capability Field Definitions (USB3.2) Register (USB3m_HOST_U3HCSPC2)

xHCI supported protocol capability field definitions (USB3.2) register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0514h
Initial Value : 2042_5355h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Name String[31:16]															
Initial Value	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Name String[15:0]															
Initial Value	0	1	0	1	0	0	1	1	0	1	0	1	0	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Name String[31:0]	20425355h	R	This field is a mnemonic name string that references the specification with which the xHC is compliant. Four ASCII characters can be defined. Allowed characters are: alphanumeric, space, and underscore. Alpha characters are case sensitive

(c) Offset 08h - xHCI Supported Protocol Capability Field Definitions (USB3.2) Register (USB3m_HOST_U3HCSPC3)

xHCI supported protocol capability field definitions (USB3.2) register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0518h
Initial Value : 0000_0101h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Protocol Speed ID Count (PSIC)[3:0]				Hub Depth (MHD)[2:0]			-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Compatible Port Count[7:0]								Compatible Port Offset[7:0]							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	Protocol Speed ID Count (PSIC)[3:0]	0h	R	This field indicates the number of Protocol Speed ID (PSI) Dwords that the xHCI Supported Protocol Capability data structure contains. If this field is non-zero, then all speeds supported by the protocol shall be defined using PSI Dwords, i.e. no implied Speed ID mappings apply. PSIC value of 0 implies that only the default SuperSpeed Gen1x1 and SuperSpeed Plus Gen2x1 bit rate are supported
27 to 25	Hub Depth (MHD)[2:0]	0h	R	If this field is '0h', then the standard USB3 hub depth constraints apply, if this field is > '0h', then it indicates the maximum hub depth supported by the USB3 ports. This bit is fixed to 0h in the USB3HOST.
24 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 8	Compatible Port Count[7:0]	1h	R	This field identifies the number of consecutive Root Hub Ports (starting at the Compatible Port Offset) that support this protocol. Valid values are 1 to MaxPorts
7 to 0	Compatible Port Offset[7:0]	1h	R	This field specifies the starting Port Number of Root Hub Ports that support this protocol. Valid values are 1 to MaxPorts.

(d) Offset 0Ch - xHCI Supported Protocol Capability Field Definitions (USB3.2) Register (USB3m_HOST_U3HCSPC4)

xHCI supported protocol capability field definitions (USB3.2) register.

Access Size : 32 bits
Address : <USB3m_host_base> + 051Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	Protocol Slot Type[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4 to 0	Protocol Slot Type[4:0]	0h	R	This field specifies the Slot Type value which can specified when allocating Device Slots that support this protocol. Valid values are 0 to 31.

(e) Protocol Speed ID for SS Register (USB3m_HOST_PSISS)

Protocol speed ID for ss register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0520h
Initial Value : 0005_0134h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Protocol Speed ID Mantissa (PSIM)[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	PSI Full-duplex (PFD)	PSI Type (PLT)[1:0]		Protocol Speed ID Exponent (PSIE)[1:0]		Protocol Speed ID Value (PSIV)[3:0]			
Initial Value	0	0	0	0	0	0	0	1	0	0	1	1	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Protocol Speed ID Mantissa (PSIM)[15:0]	5h	R	This field defines the mantissa that is applied to the PSIE when calculating the maximum bit rate represented by this PSI Dword.
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	PSI Full-duplex (PFD)	1h	R	If this bit is 1b, the link is full-duplex, and if 0b, the link is half-duplex.
7, 6	PSI Type (PLT)[1:0]	0h	R	This field identifies whether the PSI Dword defines a symmetric or asymmetric bit rate, and if asymmetric, then this field also indicates if this Dword defines the receive or transmit bit rate. Note that the Asymmetric PSI Dwords shall be paired, i.e. an Rx immediately followed by a Tx, and both Dwords shall define the same value for the PSIV. Refer to **Table 6.4-7**.
5, 4	Protocol Speed ID Exponent (PSIE)[1:0]	3h	R	This field defines the base 10 exponent to be applied with the Protocol Speed ID Mantissa is used in calculating the maximum bit rate represented by this PSI Dword. The power is three times the value of the field. Refer to **Table 6.4-8**.
3 to 0	Protocol Speed ID Value (PSIV)[3:0]	4h	R	If a device is attached that operates at the bit rate defined by this PSI Dword, then the value of this field is reported in the Port Speed field of PORTSC register of a compatible port. Note that the PSIV value of 0 is reserved and shall not be defined by a PSI.

Table 6.4-7 PLT

RLT Value	Bit Rate	Note
0	Symmetric	Single PSI Dword
1	Reserved	
2	Asymmetric Rx	Paired with Asymmetric Tx PSI Dword
3	Asymmetric Tx	Immediately follows Rx Asymmetric PSI Dword

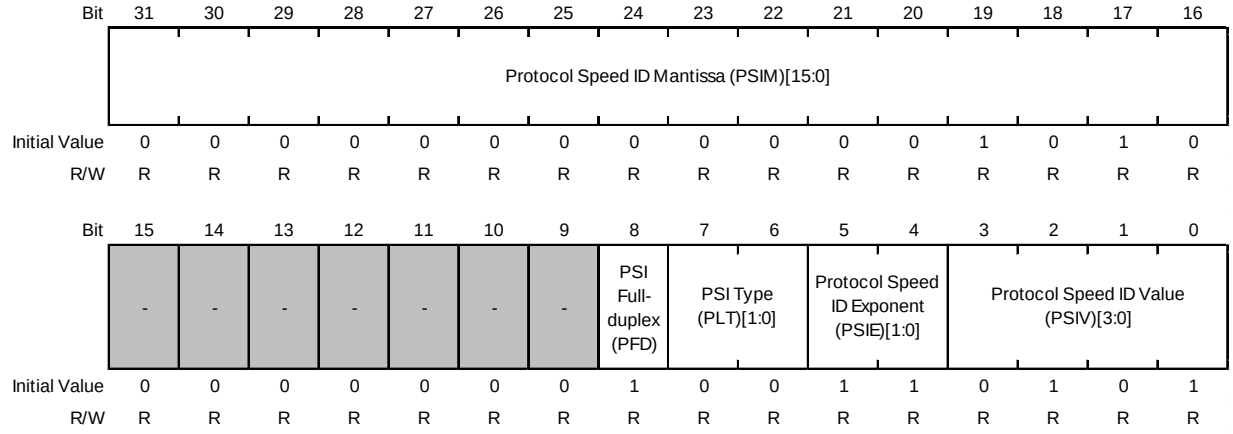
Table 6.4-8 PSIE

PSIE Value	Bit Rate
0	Bits per second
1	Kb/s
2	Mb/s
3	Gb/s

(f) Protocol Speed ID for SSP (USB3m_HOST_RP)

Protocol speed ID for SSP.

Access Size : 32 bits
Address : <USB3m_host_base> + 0524h
Initial Value : 000A_0135h



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Protocol Speed ID Mantissa (PSIM)[15:0]	Ah	R	This field defines the mantissa that is applied to the PSIE when calculating the maximum bit rate represented by this PSI Dword.
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	PSI Full-duplex (PFD)	1h	R	If this bit is 1b, the link is full-duplex, and if 0b, the link is halfduplex.
7, 6	PSI Type (PLT)[1:0]	0h	R	This field identifies whether the PSI Dword defines a symmetric or asymmetric bit rate, and if asymmetric, then this field also indicates if this Dword defines the receive or transmit bit rate. Note that the Asymmetric PSI Dwords shall be paired, i.e. an Rx immediately followed by a Tx, and both Dwords shall define the same value for the PSIV. PLT Value: Bit Rate: Note 0h: Symmetric Single: PSI Dword 1h: Reserved 2h: Asymmetric Rx: Paired with Asymmetric Tx PSI Dword 3h: Asymmetric Tx : Immediately follows Rx Asymmetric PSI Dw
5, 4	Protocol Speed ID Exponent (PSIE)[1:0]	3h	R	This field defines the base 10 exponent to be applied with the Protocol Speed ID Mantissa is used in calculating the maximum bit rate represented by this PSI Dword. The power is three times the value of the field. PSIE Value: Bit Rate 0h: Bits per second 1h: Kb/s 2h: Mb/s 3h: Gb/s
3 to 0	Protocol Speed ID Value (PSIV)[3:0]	5h	R	If a device is attached that operates at the bit rate defined by this PSI Dword, then the value of this field is reported in the Port Speed field of PORTSC register of a compatible port. Note that the PSIV value of 0 is reserved and shall not be defined by a PSI.

(3) xHCI Supported Protocol Capability for USB2.0

(a) Offset 00h - xHCI Supported Protocol Capability Field Definitions (USB2.0) Register (USB3m_HOST_U2HCSPC1)

xHCI supported protocol capability field definitions (USB2.0) register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0530h
Initial Value : 0200_0802h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Major Revision[7:0]								Minor Revision[7:0]							
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Next Capability Pointer[7:0]								Capability ID[7:0]							
Initial Value	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Major Revision[7:0]	2h	R	Major Specification Release Number in Binary-Coded Decimal (i.e., version 3.x is 03d). This field identifies the major release number component of the specification with which the xHC is compliant.
23 to 16	Minor Revision[7:0]	0h	R	Minor Specification Release Number in Binary-Coded Decimal (i.e., version x.10 is 10d). This field identifies the minor release number component of the specification with which the xHC is compliant.
15 to 8	Next Capability Pointer[7:0]	8h	R	This field indicates the location of the next capability with relative to the effective address of the current capability.
7 to 0	Capability ID[7:0]	2h	R	xHCI Supported Protocol

(b) Offset 04h - xHCI Supported Protocol Capability Field Definitions (USB2.0) Register (USB3m_HOST_U2HCSPC2)

xHCI supported protocol capability field definitions (USB2.0) register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0534h
Initial Value : 2042_5355h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Name String[31:16]															
Initial Value	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Name String[15:0]															
Initial Value	0	1	0	1	0	0	1	1	0	1	0	1	0	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Name String[31:0]	20425355h	R	This field is a mnemonic name string that references the specification with which the xHC is compliant. Four ASCII characters can be defined. Allowed characters are: alphanumeric, space, and underscore. Alpha characters are case sensitive.

(c) Offset 08h - xHCI Supported Protocol Capability Field Definitions (USB2.0) Register (USB3m_HOST_U2HCSPC3)

xHCI supported protocol capability field definitions (USB2.0) register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0538h
Initial Value : 0018_0102h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Protocol Speed ID Count (PSIC)[3:0]				Hub Depth (MHD)[2:0]			-	-	-	-	BESL LPM Capability (BLC)	Hardware LPM Capability (HLC)	Integrated Hub Implemented (IHI)	High-speed Only (HSO)	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Compatible Port Count[7:0]								Compatible Port Offset[7:0]							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	Protocol Speed ID Count (PSIC)[3:0]	0h	R	This field indicates the number of Protocol Speed ID (PSI) Dwords that the xHCI Supported Protocol Capability data structure contains. If this field is non-zero, then all speeds supported by the protocol shall be defined using PSI Dwords, i.e. no implied Speed ID mappings apply. PSIC value of 0 implies that the default Full-speed, Low-speed, and High-speed bit rates are supported.
27 to 25	Hub Depth (MHD)[2:0]	0h	R	If this field is '0h', then the standard USB2 hub depth constraints apply, if this field is > '0h', then it indicates the maximum hub depth supported by the USB2 ports. This bit is fixed to 0h in the USB3HOST.
24 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	BESL LPM Capability (BLC)	1h	R	If this bit is set to 1b, the ports described by this xHCI Supported Protocol Capability shall apply BESL timing to BESL and BESLD fields of the PORTPMSC and PORTHLPMC registers. If this bit is cleared to 0b, the ports described by this xHCI Supported Protocol Capability shall apply HIRD timing to BESL and BESLD fields of the PORTPMSC and PORTHLPMC registers. Note the BESL LMP Capability support (i.e. HLE = '1b' and BLC = '1b') shall be mandatory for all xHCI 1.2 compliant xHCs. This bit is fixed to 1b in the USB3HOST.
19	Hardware LPM Capability (HLC)	1h	R	If this bit is set to 1b, the ports described by this xHCI Supported Protocol Capability support hardware controlled USB2 Link Power Management. This bit is fixed to 1b in the USB3HOST.
18	Integrated Hub Implemented (IHI)	0h	R	If this bit is cleared to 0b, the Root hub to External xHC port mapping adheres to the default mapping. If this bit is set to 1b, the Root Hub to External xHC port mapping does not adhere to the default mapping, and ACPI or other mechanism is required to define the mapping. This bit is fixed to 0b in the USB3HOST.
17	High-speed Only (HSO)	0h	R	If this bit is cleared to 0b, the USB2 ports described by this capability are High-speed only, e.g. the ports do not support Low- or Full-speed operation. High-speed only implementations can introduce a "Tier mismatch".
16	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 8	Compatible Port Count[7:0]	1h	R	This field identifies the number of consecutive Root Hub Ports (starting at the Compatible Port Offset) that support this protocol. Valid values are 1 to MaxPorts.
7 to 0	Compatible Port Offset[7:0]	2h	R	This field specifies the starting Port Number of Root Hub Ports that support this protocol. Valid values are 1 to MaxPorts.

(d) Offset 0Ch - xHCI Supported Protocol Capability Field Definitions (USB2.0) Register (USB3m_HOST_U2HCSPC4)

xHCI supported protocol capability field definitions (USB2.0) register.

Access Size : 32 bits
Address : <USB3m_host_base> + 053Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	Protocol Slot Type[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4 to 0	Protocol Slot Type[4:0]	0h	R	This field specifies the Slot Type value which can specified when allocating Device Slots that support this protocol. Valid values are 0 to 31.

(e) Protocol Speed ID for FS Register (USB3m_HOST_PSIFS)

Protocol speed ID for FS register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0540h
Initial Value : 000C_0021h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Protocol Speed ID Mantissa (PSIM)[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	PSI Full-duplex (PFD)	PSI Type (PLT)[1:0]		Protocol Speed ID Exponent (PSIE)[1:0]		Protocol Speed ID Value (PSIV)[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Protocol Speed ID Mantissa (PSIM)[15:0]	Ch	R	This field defines the mantissa that is applied to the PSIE when calculating the maximum bit rate represented by this PSI Dword.
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	PSI Full-duplex (PFD)	0h	R	If this bit is 1b, the link is full-duplex, and if 0b, the link is half-duplex.
7, 6	PSI Type (PLT)[1:0]	0h	R	This field identifies whether the PSI Dword defines a symmetric or asymmetric bit rate, and if asymmetric, then this field also indicates if this Dword defines the receive or transmit bit rate. Note that the Asymmetric PSI Dwords shall be paired, i.e. an Rx immediately followed by a Tx, and both Dwords shall define the same value for the PSIV. Refer to **Table 6.4-9**.
5, 4	Protocol Speed ID Exponent (PSIE)[1:0]	2h	R	This field defines the base 10 exponent to be applied with the Protocol Speed ID Mantissa is used in calculating the maximum bit rate represented by this PSI Dword. The power is three times the value of the field. Refer to **Table 6.4-10**.
3 to 0	Protocol Speed ID Value (PSIV)[3:0]	1h	R	If a device is attached that operates at the bit rate defined by this PSI Dword, then the value of this field is reported in the Port Speed field of PORTSC register of a compatible port. Note that the PSIV value of 0 is reserved and shall not be defined by a PSI.

Table 6.4-9 PLT

RLT Value	Bit Rate	Note
0	Symmetric	Single PSI Dword
1	Reserved	
2	Asymmetric Rx	Paired with Asymmetric Tx PSI Dword
3	Asymmetric Tx	Immediately follows Rx Asymmetric PSI Dword

Table 6.4-10 PSIE

PSIE Value	Bit Rate
0	Bits per second
1	Kb/s
2	Mb/s
3	Gb/s

(f) Protocol Speed ID for LS Register (USB3m_HOST_PSILS)

Protocol speed ID for LS register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0544h
Initial Value : 05DC_0012h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Protocol Speed ID Mantissa (PSIM)[15:0]															
Initial Value	0	0	0	0	0	1	0	1	1	1	0	1	1	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	PSI Full-duplex (PFD)	PSI Type (PLT)[1:0]		Protocol Speed ID Exponent (PSIE)[1:0]		Protocol Speed ID Value (PSIV)[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Protocol Speed ID Mantissa (PSIM)[15:0]	5DCh	R	This field defines the mantissa that is applied to the PSIE when calculating the maximum bit rate represented by this PSI Dword.
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	PSI Full-duplex (PFD)	0h	R	If this bit is 1b, the link is full-duplex, and if 0b, the link is half-duplex.
7, 6	PSI Type (PLT)[1:0]	0h	R	This field identifies whether the PSI Dword defines a symmetric or asymmetric bit rate, and if asymmetric, then this field also indicates if this Dword defines the receive or transmit bit rate. Note that the Asymmetric PSI Dwords shall be paired, i.e. an Rx immediately followed by a Tx, and both Dwords shall define the same value for the PSIV. Refer to **Table 6.4-11**.
5, 4	Protocol Speed ID Exponent (PSIE)[1:0]	1h	R	This field defines the base 10 exponent to be applied with the Protocol Speed ID Mantissa is used in calculating the maximum bit rate represented by this PSI Dword. The power is three times the value of the field. Refer to **Table 6.4-12**.
3 to 0	Protocol Speed ID Value (PSIV)[3:0]	2h	R	If a device is attached that operates at the bit rate defined by this PSI Dword, then the value of this field is reported in the Port Speed field of PORTSC register of a compatible port. Note that the PSIV value of 0 is reserved and shall not be defined by a PSI.

Table 6.4-11 PLT

RLT Value	Bit Rate	Note
0	Symmetric	Single PSI Dword
1	Reserved	
2	Asymmetric Rx	Paired with Asymmetric Tx PSI Dword
3	Asymmetric Tx	Immediately follows Rx Asymmetric PSI Dword

Table 6.4-12 PSIE

PSIE Value	Bit Rate
0	Bits per second
1	Kb/s
2	Mb/s
3	Gb/s

(g) Protocol Speed ID for HS Register (USB3m_HOST_PSIHS)

Protocol speed ID for HS register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0548h
Initial Value : 01E0_0023h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Protocol Speed ID Mantissa (PSIM)[15:0]															
Initial Value	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	PSI Full-duplex (PFD)	PSI Type (PLT)[1:0]		Protocol Speed ID Exponent (PSIE)[1:0]		Protocol Speed ID Value (PSIV)[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Protocol Speed ID Mantissa (PSIM)[15:0]	1E0h	R	This field defines the mantissa that is applied to the PSIE when calculating the maximum bit rate represented by this PSI Dword.
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	PSI Full-duplex (PFD)	0h	R	If this bit is 1b, the link is full-duplex, and if 0b, the link is half-duplex.
7, 6	PSI Type (PLT)[1:0]	0h	R	This field identifies whether the PSI Dword defines a symmetric or asymmetric bit rate, and if asymmetric, then this field also indicates if this Dword defines the receive or transmit bit rate. Note that the Asymmetric PSI Dwords shall be paired, i.e. an Rx immediately followed by a Tx, and both Dwords shall define the same value for the PSIV. Refer to **Table 6.4-13**.
5, 4	Protocol Speed ID Exponent (PSIE)[1:0]	2h	R	This field defines the base 10 exponent to be applied with the Protocol Speed ID Mantissa is used in calculating the maximum bit rate represented by this PSI Dword. The power is three times the value of the field. Refer to **Table 6.4-14**.
3 to 0	Protocol Speed ID Value (PSIV)[3:0]	3h	R	If a device is attached that operates at the bit rate defined by this PSI Dword, then the value of this field is reported in the Port Speed field of PORTSC register of a compatible port. Note that the PSIV value of 0 is reserved and shall not be defined by a PSI.

Table 6.4-13 PLT

RLT Value	Bit Rate	Note
0	Symmetric	Single PSI Dword
1	Reserved	—
2	Asymmetric Rx	Paired with Asymmetric Tx PSI Dword
3	Asymmetric Tx	Immediately follows Rx Asymmetric PSI Dword

Table 6.4-14 PSIE

PSIE Value	Bit Rate
0	Bits per second
1	Kb/s
2	Mb/s
3	Gb/s

(4) xHCI Extended Power Management Capability

(a) Power Management Capabilities Register (USB3m_HOST_PMC)

Power management capabilities register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0550h
Initial Value : 4803_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PME Support[4:0]					D2 Support	D1 Support	Aux Current[2:0]			DSI	-	PME Clock	Version[2:0]		
Initial Value	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Next Capability Pointer[7:0]								Capability ID[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	PME Support[4:0]	9h	R	This 5-bit field indicates the power states in which the function can assert PME#. A value of 0b for any bit indicates that the function is not capable of asserting the PME# signal in that power state. Bit[27] XXXX1b – PME# can be asserted from D0 Bit[28] XXX1Xb – PME# can be asserted from D1 Bit[29] XX1XXb – PME# can be asserted from D2 Bit[30] X1XXXb – PME# can be asserted from D3hot Bit[31] 1XXXXb – PME# can be asserted from D3cold
26	D2 Support	0h	R	If this bit is 1b, this function supports the D2 Power Management State. Functions that do not support D2 shall always return a value of 0 for this bit.
25	D1 Support	0h	R	If this bit is 1b, this function supports the D1 Power Management State. Functions that do not support D1 shall always return a value of 0 for this bit.
24 to 22	Aux Current[2:0]	0h	R	This 3 bit field reports the 3.3 Vaux auxiliary current requirements for the PCI function. If the Data Register has been implemented by this function: Reading this field returns a value of 000b. The Data Register takes precedence over this field for 3.3 Vaux current requirement reporting. If PME# generation from D3cold is not supported by the function (PMC (15) = 0), this field returns a value of 000b, when read.
21	DSI	0h	R	The Device Specific Initialization bit indicates whether special initialization of this function is required (beyond the standard PCI configuration header) before the generic class device driver is able to use it. Note that this bit is not used by some operating systems. Microsoft Windows and Windows NT, for instance, do not use this bit to determine whether to use D3. Instead, they use the driver's capabilities to determine this. A 1 indicates that the function requires a device specific initialization sequence following transition to the D0 uninitialized state.
20	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	PME Clock	0h	R	When this bit is 1b, it indicates that the function relies on the presence of the PCI clock for PME# operation. When this bit is 0b, it indicates that no PCI clock is required for the function to generate PME#. Functions that do not support PME# generation in any state returns 0 for this field.
18 to 16	Version[2:0]	3h	R	A value of 011b indicates that this function complies with revision 1.2 of the PCI Power Management Interface Specification.
15 to 8	Next Capability Pointer[7:0]	0h	R	This field provides an offset into the function's PCI Configuration Space pointing to the location of next item in the function's capability list. If there are no additional items in the Capabilities List, this register is set to 0h.
7 to 0	Capability ID[7:0]	3h	R	Extended Power Management

(b) Power Management Control/Status Register (USB3m_HOST_PMCSR)

Power management control/status register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0554h
Initial Value : 0000_0008h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PME Status	Data Scale[1:0]		Data Select[3:0]				PME En	-	-	-	-	No Soft Reset	-	PowerState [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	RW1	R	R	R	R	R	R	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15	PME Status	0h	RW1	This bit is set when the function would normally assert the PME# signal independent of the state of the PME_En bit. Writing 1b clears this bit and causes the function to stop asserting a PME# (if enabled). Writing 0b has no effect. This bit defaults to 0b if the function does not support PME# generation from D3cold. If the function supports PME# from D3cold, then this bit is sticky and shall be explicitly cleared by the operating system each time the operating system is initially loaded.
14, 13	Data Scale[1:0]	0h	R	This 2-bit read-only field indicates the scaling factor to be used when interpreting the value of the Data register. The value and meaning of this field will vary depending on which data value has been selected by the Data_Select field. This field is a required component of the Data register (offset 7) and shall be implemented if the Data register is implemented. If the Data register has not been implemented, this field shall return 00b when the PMCSR is read.
12 to 9	Data Select[3:0]	0h	R	This 4-bit field is used to select which data is to be reported through the Data register and Data_Scale field. This field is a required component of the Data register (offset 7) and shall be implemented if the Data register is implemented. If the Data register is not implemented, this field should be read only and return 0000b when the PMCSR is read.
8	PME En	0h	RW	Writing 1b to this bit enables the function to assert PME#. When 0b, PME# assertion is disabled. This bit defaults to 0b if the function does not support PME# generation from D3cold. If the function supports PME# from D3cold, then this bit is sticky and shall be explicitly cleared by the operating system each time it is initially loaded. Functions that do not support PME# generation from any D-state (i.e., PMC (15:11) = 0_0000b), can hardwire this bit to be read-only always returning 0 when read by system software.
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	No Soft Reset	1h	R	When set (1), this bit indicates that devices are transitioning from D3hot to D0 because of PowerState commands do not perform an internal reset. Configuration Context is preserved. Upon transition from the D3hot to the D0 Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the PowerState bits. When clear (0), devices do perform an internal reset upon transitioning from D3hot to D0 via software control of the PowerState bits. Configuration Context is lost when performing the soft reset. Upon transition from the D3hot to the D0 state, full reinitialization sequence is needed to return the device to D0 Initialized. Regardless of this bit, devices that transition from D3hot to D0 by a system or bus segment reset will return to the device state D0 Uninitialized with only PME context preserved if PME is supported and enabled.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PowerState [1:0]	0h	RW	This 2-bit field is used both to determine the current power state of a function and to set the function into a new power state. The definition of the field values is given below. 00b: D0 01b: D1 10b: D2 11b: D3hot If an unsupported or optional state is written to this field by software, the write operation shall complete normally on the bus. However, the data is discarded and no state change occurs.

6.4.4.4.5 Host Controller Runtime Registers

(1) Microframe Index Register (USB3m_HOST_MFINDEX)

Microframe index register.

Access Size : 32 bits

Address : <USB3m_host_base> + 0600h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Microframe Index[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 0	Microframe Index[13:0]	0h	R	The value in this register increments at the end of each microframe (e.g. 125 µs.). Bits [13:3] can be used to determine the current 1ms Frame Index.

(2) Interrupter Register Set 0

(a) Interrupter Management Register (USB3m_HOST_IMAN)

Interrupter management register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0620h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Interrupt Enable (IE)	Interrupt Pending (IP)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	Interrupt Enable (IE)	0h	RW	Default = 0b. This flag specifies whether the Interrupter is capable of generating interrupts. When this bit and the IP bit are set to (1b), the Interrupter generates an interrupt when the Interrupter Moderation Counter reaches 0. If this bit is 0b, then the Interrupter is prohibited from generating interrupts.
0	Interrupt Pending (IP)	0h	RW1	Default = 0b. This flag represents the current state of the Interrupter. If IP = 1b, an interrupt is pending for this Interrupter. A 0 value indicates that no interrupt is pending for the Interrupter.

(b) Interrupter Moderation Register (USB3m_HOST_IMOD)

Interrupter moderation register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0624h
Initial Value : 0000_0FA0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Interrupt Moderation Counter (IMODC)[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Interrupt Moderation Interval (IMODI)[15:0]															
Initial Value	0	0	0	0	1	1	1	1	1	0	1	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Interrupt Moderation Counter (IMODC)[15:0]	0h	RW	Default = undefined. Down counter. Loaded with the IMODI value whenever IP is cleared to 0, counts down to 0, and stops. The associated interrupt is signaled whenever this counter is 0, the Event Ring is not empty, the IE and IP flags = 1b, and EHB = 0b. This counter can be directly written by software at any time to alter the interrupt rate.
15 to 0	Interrupt Moderation Interval (IMODI)[15:0]	FA0h	RW	Default = '4000' (up to 1 ms). Minimum inter-interrupt interval. The interval is specified in 250ns increments. A value of 0 disables interrupt throttling logic and interrupts are generated immediately if IP = 0b, EHB = 0b, and the Event Ring is not empty.

(c) Event Ring Registers

(c-1) Event Ring Segment Table Size Register (USB3m_HOST_ERSTSZ)

Event ring segment table size register.

Access Size : 32 bits

Address : <USB3m_host_base> + 0628h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Event Ring Segment Table Size[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	Event Ring Segment Table Size[15:0]	0h	RW	Default = 0b. This field identifies the number of valid Event Ring Segment Table entries in the Event Ring Segment Table pointed to by the Event Ring Segment Table Base Address register. The maximum value supported by an xHC implementation for this register is defined by the ERST Max field in the HCSPARAMS2 register. For the Primary Interrupter: Writing a value of 0 to this field results in undefined behavior of the Event Ring. The Primary Event Ring cannot be disabled.

(c-2) Event Ring Segment Table Base Address Low Register (USB3m_HOST_ERSTBAL)

Event ring segment table base address low register.

Access Size : 32 bits

Address : <USB3m_host_base> + 0630h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Event Ring Segment Table Base Address Low Register[25:10]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Event Ring Segment Table Base Address Low Register[9:0]										-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	Event Ring Segment Table Base Address Low Register[25:0]	0h	RW	Default = 0b. This field defines the lower-order bits of the address of where the Event Ring Segment Table starts. Writing to this register sets advancement of the event ring state machine (EREP) to the start state. This field shall not be modified if HCHalted (HCH) = 0b.
5 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(c-3) Event Ring Segment Table Base Address High Register (USB3m_HOST_ERSTBAH)

Event ring segment table base address high register.

Access Size : 32 bits

Address : <USB3m_host_base> + 0634h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Event Ring Segment Table Base Address High Register[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

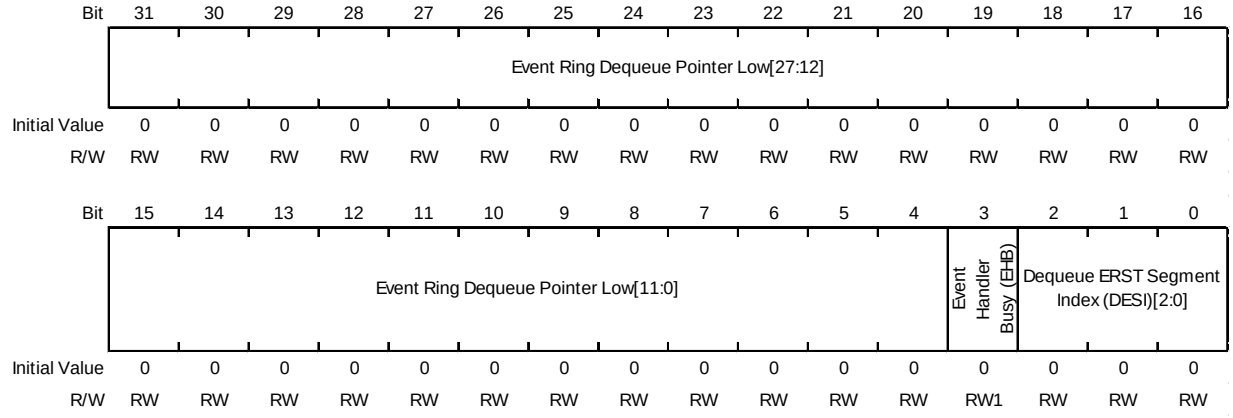
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Event Ring Segment Table Base Address High Register[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Event Ring Segment Table Base Address High Register[31:0]	0h	RW	Default = 0b. This field defines the higher-order bits of the start address where the Event Ring Segment Table starts. Writing to this register sets advancement of the event ring state machine (EREP) to the start state. This field shall not be modified if HCHalted (HCH) = 0b.

(c-4) Event Ring Dequeue Pointer Low Register (USB3m_HOST_ERDPL)

Event ring dequeue pointer low register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0638h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 4	Event Ring Dequeue Pointer Low[27:0]	0h	RW	Default = 0b. This field defines the lower-order bits of the 64-bit address of the current Event Ring Dequeue Pointer.
3	Event Handler Busy (EHB)	0h	RW1	Default = 0b. This flag is set to 1b when the IP bit is set to 1b and cleared to 0b by software when the Dequeue Pointer register is written.
2 to 0	Dequeue ERST Segment Index (DESI)[2:0]	0h	RW	Default = 0b. This field can be used by the xHC to accelerate checking the Event Ring full condition. This field is written with the lower-order 3 bits of the offset of the ERST entry which defines the Event Ring segment that the Event Ring Dequeue Pointer resides in.

(c-5) Event Ring Dequeue Pointer High Register (USB3m_HOST_ERDPH)

Event ring dequeue pointer high register.

Access Size : 32 bits

Address : <USB3m_host_base> + 063Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Event Ring Dequeue Pointer High[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Event Ring Dequeue Pointer High[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Event Ring Dequeue Pointer High[31:0]	0h	RW	Default = 0b. This field defines the higher-order bits of the 64-bit address of the current Event Ring Dequeue Pointer.

6.4.4.4.6 Doorbell Registers

(1) Host Controller Doorbell Register (USB3m_HOST_HCD)

Host controller doorbell register.

Access Size : 32 bits

Address : <USB3m_host_base> + 0700h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

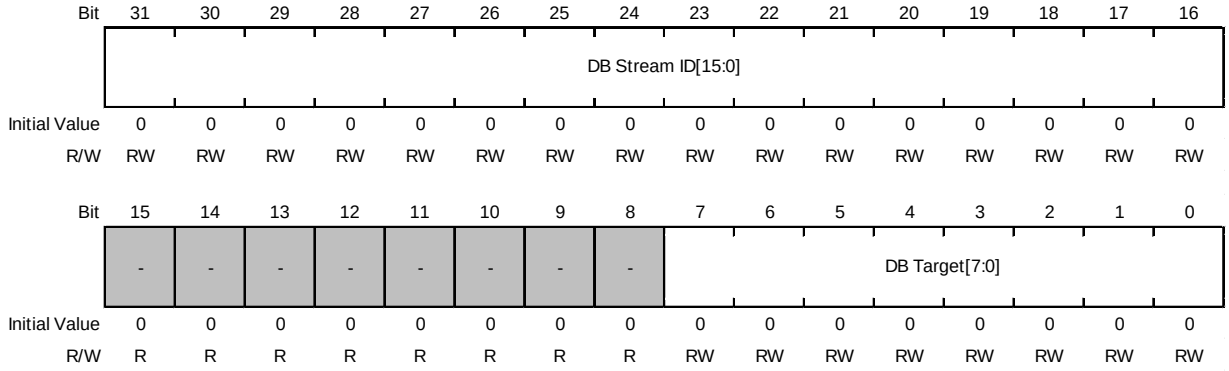
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	DB Target[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	DB Target[7:0]	0h	RW	Doorbell Target. This field defines the target of a doorbell reference. The table below defines the xHC notifications that are generated by ringing a doorbell. Note that Doorbell Register 0 is dedicated to the Command Ring and this field of that register is decoded differently from the field in the Doorbell Registers. Host Controller Doorbell (0) 00h: Command Doorbell 01h to F7h: Reserved F8h to FFh: Vendor Defined This field returns 0b when read and should be treated as “undefined” by software.

(2) Device Context Doorbell Register (Slot #n) (USB3m_HOST_DCDn) (n = 1 to 32)

Device context doorbell register.

Access Size : 32 bits
Address : <USB3m_host_base> + 0700h + n x 0004h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	DB Stream ID[15:0]	0h	RW	Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines streams, then this field can be used to identify the stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint defines streams (MaxPStreams > 0), then 0, 65535 (no stream) and 65534 (prime) are reserved stream ID values and shall not be written to this field. If the endpoint does not define streams (MaxPStreams = 0) and a non-0 value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0b' for Host Controller Command Doorbells. This field returns '0b' when read.
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	DB Target[7:0]	0h	RW	Doorbell Target. This field defines the target of a doorbell reference. The table below defines the xHC notifications that are generated by ringing a doorbell. Note that Doorbell Register 0 is dedicated to the Command Ring and this field of that register is decoded differently from the field in the Doorbell Registers. Device Context Doorbells (1-255) 00h: Reserved 01h: Control EP 0 Enqueue Pointer Update 02h: EP 1 OUT Enqueue Pointer Update 03h: EP 1 IN Enqueue Pointer Update 04h: EP 2 OUT Enqueue Pointer Update 05h: EP 2 IN Enqueue Pointer Update ⋮ 1Eh: EP 15 OUT Enqueue Pointer Update 1Fh: EP 15 IN Enqueue Pointer Update 20h to F7h: Reserved F8h to FFh: Vendor Defined This field returns 0b when read and should be treated as "undefined" by software.

6.4.4.4.7 Core Defined Registers

(1) Core Control and Status Registers

(a) Revision ID Register (USB3m_HOST_REVID)

Revision ID register.

Access Size : 32 bits
Address : <USB3m_host_base> + 1000h
Initial Value : E302_0100h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Core ID1[7:0]								Core ID2[7:0]							
Initial Value	1	1	1	0	0	0	1	1	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Major Version[7:0]								Minor Version[7:0]							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Core ID1[7:0]	E3h	R	Identified upper number of the core
23 to 16	Core ID2[7:0]	2h	R	Identified lower number of the core
15 to 8	Major Version[7:0]	1h	R	Major version of the core
7 to 0	Minor Version[7:0]	0h	R	Minor version of the core

(b) Configuration Status1 Register (USB3m_HOST_CFGSTS1)

Register for reading core configuration values.

Access Size : 32 bits
Address : <USB3m_host_base> + 1004h
Initial Value : 0x29_4111h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Renesas Private18	Renesas Private17	Renesas Private16	Renesas Private15	Renesas Private14	Renesas Private13	Renesas Private12	Renesas Private11	Renesas Private10	Renesas Private9	Renesas Private8	Renesas Private7	Support SSP (SUPSSP)	Renesas Private5[1:0]		Renesas Private4
Initial Value	0	0	0	0	1	1	x	x	0	0	1	0	1	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Renesas Private3	Renesas Private2	Renesas Private1	Renesas Private0	-	Number of USB Receptacle (REPNUM)[2:0]			-	Number of U3 Port (U3NUM)[2:0]			-	Number of U2 Port (U2NUM)[2:0]		
Initial Value	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	Renesas Private18	0h	R	Renesas Private bit
30	Renesas Private17	0h	R	Renesas Private bit
29	Renesas Private16	0h	R	Renesas Private bit
28	Renesas Private15	0h	R	Renesas Private bit
27	Renesas Private14	1h	R	Renesas Private bit
26	Renesas Private13	1h	R	Renesas Private bit
25	Renesas Private12	x	R	Renesas Private bit
24	Renesas Private11	x	R	Renesas Private bit
23	Renesas Private10	0h	R	Renesas Private bit
22	Renesas Private9	0h	R	Renesas Private bit
21	Renesas Private8	1h	R	Renesas Private bit
20	Renesas Private7	0h	R	Renesas Private bit
19	Support SSP (SUPSSP)	1h	R	Indicates whether the "SSP" is supported or not in this core. 0b : Not supported 1b : Supported
18, 17	Renesas Private5[1:0]	0h	R	Renesas Private bit
16	Renesas Private4	1h	R	Renesas Private bit
15	Renesas Private3	0h	R	Renesas Private bit
14	Renesas Private2	1h	R	Renesas Private bit
13	Renesas Private1	0h	R	Renesas Private bit

Bit	Bit Name	Initial Value	R/W	Description
12	Renesas Private0	0h	R	Renesas Private bit
11	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10 to 8	Number of USB Receptacle (REPNUM)[2:0]	1h	R	Indicate the number of USB receptacle
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 4	Number of U3 Port (U3NUM)[2:0]	1h	R	Indicate the number of U3 ports
3	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	Number of U2 Port (U2NUM)[2:0]	1h	R	Indicate the number of USB2.0 ports.

(c) Configuration Status2 Register (USB3m_HOST_CFGSTS2)

Register for reading core configuration values.

Access Size : 32 bits
Address : <USB3m_host_base> + 1008h
Initial Value : 0000_1388h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ACLK's period (ACLKFREQ)[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ACLK's period (ACLKFREQ)[15:0]															
Initial Value	0	0	0	1	0	0	1	1	1	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ACLK's period (ACLKFREQ) [31:0]	1388h	R	Indicates "the aclk's Period". This value 1388h means aclk's period is 5000ps (5ns), that is 200MHz.

(d) Configuration Status3 Register (USB3m_HOST_CFGSTS3)

Register for reading core configuration values.

Access Size : 32 bits

Address : <USB3m_host_base> + 100Ch

Initial Value : 0011_0111h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	Renesas Private16	Renesas Private15	Renesas Private14	Renesas Private13	Renesas Private12	-	Renesas Private11[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Renesas Private10	Renesas Private9	Renesas Private8	Renesas Private7	Renesas Private6	Renesas Private5	Renesas Private4	Renesas Private3	-	-	Renesas Private2	Renesas Private1	Renesas Private0[3:0]			
Initial Value	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	Renesas Private16	0h	R	Renesas Private bit
23	Renesas Private15	0h	R	Renesas Private bit
22	Renesas Private14	0h	R	Renesas Private bit
21	Renesas Private13	0h	R	Renesas Private bit
20	Renesas Private12	1h	R	Renesas Private bit
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	Renesas Private11[2:0]	1h	R	Renesas Private bit
15	Renesas Private10	0h	R	Renesas Private bit
14	Renesas Private9	0h	R	Renesas Private bit
13	Renesas Private8	0h	R	Renesas Private bit
12	Renesas Private7	0h	R	Renesas Private bit
11	Renesas Private6	0h	R	Renesas Private bit
10	Renesas Private5	0h	R	Renesas Private bit
9	Renesas Private4	0h	R	Renesas Private bit
8	Renesas Private3	1h	R	Renesas Private bit
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	Renesas Private2	0h	R	Renesas Private bit

Bit	Bit Name	Initial Value	R/W	Description
4	Renesas Private1	1h	R	Renesas Private bit
3 to 0	Renesas Private0[3:0]	1h	R	Renesas Private bit

(e) Configuration Status4 Register (USB3m_HOST_CFGSTS4)

Register for reading core configuration values.

Access Size : 32 bits
Address : <USB3m_host_base> + 1010h
Initial Value : 0000_0021h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	Renesas Private1[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Renesas Private0[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 16	Renesas Private1[9:0]	0h	R	Renesas Private field.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	Renesas Private0[9:0]	21h	R	Renesas Private field.

(f) Configuration Status5 Register (USB3m_HOST_CFGSTS5)

Register for reading core configuration values.

Access Size : 32 bits
Address : <USB3m_host_base> + 1014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	Renesas Private1[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Renesas Private0[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 16	Renesas Private1[9:0]	0h	R	Renesas Private field.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	Renesas Private0[9:0]	0h	R	Renesas Private field.

(g) Configuration Status6 Register (USB3m_HOST_CFGSTS6)

Register for reading core configuration values.

Access Size : 32 bits
Address : <USB3m_host_base> + 1018h
Initial Value : 0000_0129h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	Renesas Private1[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Renesas Private0[9:0]									
Initial Value	0	0	0	0	0	0	0	1	0	0	1	0	1	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 16	Renesas Private1[9:0]	0h	R	Renesas Private field.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	Renesas Private0[9:0]	129h	R	Renesas Private field.

(h) Configuration Status7 Register (USB3m_HOST_CFGSTS7)

Register for reading core configuration values.

Access Size : 32 bits
Address : <USB3m_host_base> + 101Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	Renesas Private1[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Renesas Private0[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 16	Renesas Private1[9:0]	0h	R	Renesas Private field.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	Renesas Private0[9:0]	0h	R	Renesas Private field.

(i) Configuration Status8 Register (USB3m_HOST_CFGSTS8)

Register for reading core configuration values.

Access Size : 32 bits
Address : <USB3m_host_base> + 1020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	Renesas Private1[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Renesas Private0[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 16	Renesas Private1[9:0]	0h	R	Renesas Private field.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	Renesas Private0[9:0]	0h	R	Renesas Private field.

(j) Configuration Status9 Register (USB3m_HOST_CFGSTS9)

Register for reading core configuration values.

Access Size : 32 bits
Address : <USB3m_host_base> + 1024h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	Renesas Private1[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Renesas Private0[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 16	Renesas Private1[9:0]	0h	R	Renesas Private field.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	Renesas Private0[9:0]	0h	R	Renesas Private field.

(k) Configuration Status10 Register (USB3m_HOST_CFGSTS10)

Register for reading core configuration values.

Access Size : 32 bits
Address : <USB3m_host_base> + 1028h
Initial Value : 0006_0030h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	SS Data Buffer Size[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	Renesas Private1[4:0]					-	U2 Data Buffer Size[2:0]			-	-	Renesas Private0[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20 to 16	SS Data Buffer Size[4:0]	6h	R	Indicate the Data buffer size for U3. 6: 6 Kbytes
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 8	Renesas Private1[4:0]	0h	R	Renesas Private field.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 4	U2 Data Buffer Size[2:0]	3h	R	Indicate the Data buffer size for U2. 3: 1 Kbyte
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	Renesas Private0[1:0]	0h	R	Renesas Private field.

(I) Data Buffer Threshold Control Register (USB3m_HOST_DBUFTH)

Before USBCMD register's Run/Stop bit is set to 1, this register shall be written.

Access Size : 32 bits

Address : <USB3m_host_base> + 102Ch

Initial Value : 1111_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	Renesas Private7[1:0]		-	-	Renesas Private6[1:0]		-	-	Renesas Private5[1:0]		-	-	Renesas Private4[1:0]	
Initial Value	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1
R/W	R	R	RW	RW	R	R	RW	RW	R	R	RW	RW	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Renesas Private3[1:0]		-	-	Renesas Private2[1:0]		-	-	Renesas Private1[1:0]		-	-	Renesas Private0[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	RW	RW	R	R	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29, 28	Renesas Private7[1:0]	1h	RW	Renesas Private bit This field shall be set to 1h.
27, 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25, 24	Renesas Private6[1:0]	1h	RW	Renesas Private bit This field shall be set to 1h.
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21, 20	Renesas Private5[1:0]	1h	RW	Renesas Private bit This field shall be set to 1h.
19, 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17, 16	Renesas Private4[1:0]	1h	RW	Renesas Private bit This field shall be set to 1h.
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13, 12	Renesas Private3[1:0]	0h	RW	Renesas Private bit This field shall be set to 0h.
11, 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9, 8	Renesas Private2[1:0]	0h	RW	Renesas Private bit This field shall be set to 0h.
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5, 4	Renesas Private1[1:0]	0h	RW	Renesas Private bit This field shall be set to 0h.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	Renesas Private0[1:0]	0h	RW	Renesas Private bit This field shall be set to 0h.

(m) Core Control Register (USB3m_HOST_CORECTRL)

Core control register.

Access Size : 32 bits

Address : <USB3m_host_base> + 1030h

Initial Value : 0000_2000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Renesas Private5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Renesas Private4
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Renesas Private4[3:0]				-	Renesas Private3	Renesas Private2	Renesas Private1	-	-	-	-	-	-	-	Renesas Private0
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	RW	RW	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	Renesas Private5	0h	RW	Renesas Private bit This field shall be set to 0h.
30 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16 to 12	Renesas Private4[4:0]	2h	RW	Renesas Private bit This field shall be set to 2h.
11	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	Renesas Private3	0h	RW	Renesas Private bit This field shall be set to 0h.
9	Renesas Private2	0h	RW	Renesas Private bit This field shall be set to 0h.
8	Renesas Private1	0h	RW	Renesas Private bit This field shall be set to 0h.
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	Renesas Private0	0h	RW	Renesas Private bit This field shall be set to 0h.

(n) PHY Control Register (USB3m_HOST_PHYCTRL)

Phy control register.

Access Size : 32 bits

Address : <USB3m_host_base> + 1034h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Renesas Private6	Renesas Private5	Renesas Private4	Renesas Private3	-	-	-	-	-	-	-	-	Renesas Private2	Renesas Private1	Renesas Private0	USB3.2 PHY for Port1's Reset (U3P1PHY_RST)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	Force Active USB2.0 PHY PLL (FA_U2PLL)	-	-	-	-	-	-	-	USB2.0 PHY Reset (U2PHY_RST)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	Renesas Private6	0h	RW	Renesas Private bit This field shall be set to 0h.
30	Renesas Private5	0h	RW	Renesas Private bit This field shall be set to 0h.
29	Renesas Private4	0h	RW	Renesas Private bit This field shall be set to 0h.
28	Renesas Private3	0h	RW	Renesas Private bit This field shall be set to 0h.
27 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	Renesas Private2	0h	RW	Renesas Private bit This field shall be set to 0h.
18	Renesas Private1	0h	RW	Renesas Private bit This field shall be set to 0h.
17	Renesas Private0	0h	RW	Renesas Private bit This field shall be set to 0h.
16	USB3.2 PHY for Port1's Reset (U3P1PHY_RST)	0h	RW	This is a bit to output a reset signal for USB3.2 PHY for Port1. 0b: Non-reset assertion 1b: Reset assertion
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	Force Active USB2.0 PHY PLL (FA_U2PLL)	0h	RW	Setting this bit forces the PLL for USB2.0 to the active condition. 0b: PLL for USB2.0 does not force to the active condition 1b: PLL for USB2.0 forces to the active condition
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	USB2.0 PHY Reset (U2PHY_RST)	0h	RW	This is a bit to output a reset signal for USB2.0 PHY. 0b: Non-reset assertion 1b: Reset assertion

(o) PHY Status Register (USB3m_HOST_PHYSTS)

Phy status register.

Access Size : 32 bits

Address : <USB3m_host_base> + 1038h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	Renesas Private2	Renesas Private1	Renesas Private0	USB3.2 for Port1 PLL Lock (U3P1PLL_LOCK)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	USB2.0 PLL Lock (U2PLL_LOCK)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	Renesas Private2	0h	R	Renesas Private bit
18	Renesas Private1	0h	R	Renesas Private bit
17	Renesas Private0	0h	R	Renesas Private bit
16	USB3.2 for Port1 PLL Lock (U3P1PLL_LOCK)	0h	R	Indicates that USB3.2 for Port 1 PLL is locked. 0b: USB3.2 PLL is not locked. 1b: USB3.2 PLL is locked.
15 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	USB2.0 PLL Lock (U2PLL_LOCK)	0h	R	Indicates that USB2.0 PLL is locked. 0b: USB2.0 PLL is not locked. 1b: USB2.0 PLL is locked

(p) Renesas Private Register (USB3m_HOST_RP2)

Renesas private register.

Access Size : 32 bits
Address : <USB3m_host_base> + 103Ch
Initial Value : 000x_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Renesas Private[17:16]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Renesas Private[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17 to 0	Renesas Private[17:0]	x	RW	Renesas Private field. This field shall be set to this field's read value. Note: The default value depends on User-settable Keywords.

Note: X: Undefined

(q) Interrupt Status Register (USB3m_HOST_INTSTS)

This register indicates the interrupt status of the USB3HOST.

Access Size : 32 bits
Address : <USB3m_host_base> + 1040h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	SMI Interrupt (SMI_INT)	-	HSE Interrupt (HSE_INT)	PME Interrupt (PME_INT)	XHC Interrupt (XHC_INT)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	SMI Interrupt (SMI_INT)	0h	R	SMI interrupt status When an SMI interrupt occurs, this bit is set to 1b. Smi_int is this interrupt output signal.
3	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	HSE Interrupt (HSE_INT)	0h	R	HSE interrupt status When an HSE interrupt occurs, this bit is set to 1b. Hse_int is this interrupt output signal
1	PME Interrupt (PME_INT)	0h	R	PME interrupt status When a PME interrupt occurs, this bit is set to 1b. Pme_int is this interrupt output signal.
0	XHC Interrupt (XHC_INT)	0h	R	XHC interrupt status When an XHC interrupt occurs, this bit is set to 1b. Xhc_int is this interrupt output signal.

(r) Interrupt Enable Register (USB3m_HOST_INTEN)

Interrupt enable register.

Access Size : 32 bits

Address : <USB3m_host_base> + 1044h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	SMI Interrupt Enable (SMI_INTE)	-	HSE Interrupt Enable (HSE_INTE)	PME Interrupt Enable (PME_INTE)	XHC Interrupt Enable (XHC_INTE)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	SMI Interrupt Enable (SMI_INTE)	0h	RW	This bit selects whether the interrupt state of INTSTS.SMI_INT is joined all_int interrupt output or not. 0b: The interrupt state of INTSTS.SMI_INT is not joined all_int. 1b: The interrupt state of INTSTS.SMI_INT is joined all_int.
3	-	0h	RW	Reserved When writing to this bit, Read modify write should be done.
2	HSE Interrupt Enable (HSE_INTE)	0h	RW	This bit selects whether the interrupt state of INTSTS.HSE_INT is joined all_int interrupt output. 0b: The interrupt state of INTSTS.HSE_INT is not joined all_int. 1b: The interrupt state of INTSTS.HSE_INT is joined all_int.
1	PME Interrupt Enable (PME_INTE)	0h	RW	This bit selects whether the interrupt state of INTSTS.PME_INT is joined all_int interrupt output. 0b: The interrupt state of INTSTS.PME_INT is not joined all_int. 1b: The interrupt state of INTSTS.PME_INT is joined all_int.
0	XHC Interrupt Enable (XHC_INTE)	0h	RW	This bit selects whether the interrupt state of INTSTS.XHC_INT is joined all_int interrupt output. 0b: The interrupt state of INTSTS.XHC_INT is not joined all_int. 1b: The interrupt state of INTSTS.XHC_INT is joined all_int.

(s) Frame Length Adjustment Register (FLADJ) & Serial Bus Release Number Register (SBRN) (USB3m_HOST_FLADJ_SBRN)

This is Serial Bus Release Number (SBRN) & Frame Length Adjustment (FLADJ) & Default Best Effort Service Latency (DBESL) & Default Best Effort Service Latency Deep (DBESLD) Register.

Access Size : 32 bits

Address : <USB3m_host_base> + 1050h

Initial Value : 00xx_2032h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	Default Best Effort Service Latency Deep (DBESLD)[3:0]				Default Best Effort Service Latency (DBESL)[3:0]			
Initial Value	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	No Frame Length Timing Capability (NFC)	Frame Length Adjustment (FLADJ)[5:0]						Serial Bus Release Number (SBRN)[7:0]							
Initial Value	0	0	1	0	0	0	0	0	0	0	1	1	0	0	1	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 20	Default Best Effort Service Latency Deep (DBESLD)[3:0]	x	R	If the value of this field is non-zero, it defines the recommended value for programming the PORTPMSC register BESLD field.
19 to 16	Default Best Effort Service Latency (DBESL)[3:0]	x	R	If the value of this field is non-zero, it defines the recommended value for programming the PORTPMSC register BESL field.
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14	No Frame Length Timing Capability (NFC)	0h	R	This flag indicates whether the host controller implementation supports a Frame Length Timing Value. A 1b in this bit indicates that the Frame Length Timing Value is not supported. A 0b in this bit indicates that the Frame Length Timing Value is supported.
13 to 8	Frame Length Adjustment (FLADJ)[5:0]	20h	RW	Each decimal value change to this register corresponds to 16 high-speed bit times. The SOF cycle time (number of SOF counter clock periods to generate an SOF microframe length) is equal to 59488 + value in this field. The default value is decimal 32 (20h), which gives an SOF cycle time of 60000. Refer to **Table 6.4-15**.
7 to 0	Serial Bus Release Number (SBRN)[7:0]	32h	R	The version of the Universal Serial Bus Specification that is compliant with Host Controller. 32h: Release 3.2

Note: X: Undefined

Table 6.4-15 FLADJ

Frame Length (# HS bit times) (decimal)	FLADJ Value (decimal)
59488	0 (00h)
59504	1 (01h)
59520	2 (02h)
…	…
59984	31 (1Fh)
60000	32 (20h)
…	…
60480	62 (3Eh)
60496	63 (3Fh)

(2) Battery Charging Register

(a) Battery Charging Control Register (USB3m_HOST_BCCTRL)

Battery charging control register.

Access Size : 32 bits

Address : <USB3m_host_base> + 1080h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	Renesas Private6	Renesas Private5	Renesas Private4	Renesas Private3	Renesas Private2[1:0]		Renesas Private1[1:0]		Renesas Private0[1:0]		BC_MODE_P1 [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11	Renesas Private6	0h	RW	Renesas Private bit This bit shall be set to 0b.
10	Renesas Private5	0h	RW	Renesas Private bit This bit shall be set to 0b.
9	Renesas Private4	0h	RW	Renesas Private bit This bit shall be set to 0b.
8	Renesas Private3	0h	RW	Renesas Private bit This bit shall be set to 0b.
7, 6	Renesas Private2[1:0]	0h	RW	Renesas Private bit This field shall be set to 0h.
5, 4	Renesas Private1[1:0]	0h	RW	Renesas Private bit This field shall be set to 0h.
3, 2	Renesas Private0[1:0]	0h	RW	Renesas Private bit This field shall be set to 0h.
1, 0	BC_MODE_P1 [1:0]	0h	RW	Battery Charging Mode for Port1 00b: SDP 01b: CDP 10b: DCP 11b: Reserved

6.4.4.4.8 PIPE Status and Control Registers

(1) PIPE Status and Control Register 0 (USB3m_HOST_U3P0PIPESC0)

PIPE Status and Control Register 0.

Access Size : 32 bits

Address : <USB3m_host_base> + 10C0h

Initial Value : 3013_2000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	Tx Coefficient C+1[5:0]						-	-	Tx Coefficient C0[5:0]					
Initial Value	0	0	1	1	0	0	0	0	0	0	0	1	0	0	1	1
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Tx Coefficient C-1[5:0]						-	-	-	Tx Swing	-	Tx Margin[2:0]		
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	RW	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
29 to 24	Tx Coefficient C+1[5:0]	30h	RW	Renesas Private bit Specify d'3.
23, 22	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
21 to 16	Tx Coefficient C0[5:0]	13h	RW	Renesas Private bit Specify d'19.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
13 to 8	Tx Coefficient C-1[5:0]	20h	RW	Renesas Private bit Specify d'2.
7 to 5	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
4	Tx Swing	0h	RW	Renesas Private bit Keep the initial value.
3	-	0h	R	Reserved This bit is read as 0b. The write value should always be 0b.
2 to 0	Tx Margin[2:0]	0h	RW	Renesas Private bit Keep the initial value.

(2) PIPE Status and Control Register 1 (USB3m_HOST_U3P0PIPESC1)

PIPE Status and Control Register 1.

Access Size : 32 bits

Address : <USB3m_host_base> + 10C4h

Initial Value : 0016_2000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	Tx Coefficient C+1[5:0]						-	-	Tx Coefficient C0[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Tx Coefficient C-1[5:0]						-	-	-	-	-	-	-	-
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
29 to 24	Tx Coefficient C+1[5:0]	0h	RW	Renesas Private bit Specify d'0.
23, 22	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
21 to 16	Tx Coefficient C0[5:0]	16h	RW	Renesas Private bit Specify d'22.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
13 to 8	Tx Coefficient C-1[5:0]	20h	RW	Renesas Private bit Specify d'2.
7 to 0	-	0h	R	Reserved These bits are read as 0b. The write value should always be 0b.

(3) PIPE Status and Control Register 2 (USB3m_HOST_U3P0PIPESC2)

PIPE Status and Control Register 2.

Access Size : 32 bits

Address : <USB3m_host_base> + 10C8h

Initial Value : 3015_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	Tx Coefficient C+1[5:0]						-	-	Tx Coefficient C0[5:0]					
Initial Value	0	0	1	1	0	0	0	0	0	0	0	1	0	1	0	1
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Tx Coefficient C-1[5:0]						-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
29 to 24	Tx Coefficient C+1[5:0]	30h	RW	Renesas Private bit Specify d'3.
23, 22	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
21 to 16	Tx Coefficient C0[5:0]	15h	RW	Renesas Private bit Specify d'21.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
13 to 8	Tx Coefficient C-1[5:0]	0h	RW	Renesas Private bit Specify d'0.
7 to 0	-	0h	R	Reserved These bits are read as 0b. The write value should always be 0b.

(4) PIPE Status and Control Register 3 (USB3m_HOST_U3P0PIPESC3)

PIPE Status and Control Register 3.

Access Size : 32 bits

Address : <USB3m_host_base> + 10CCh

Initial Value : 3013_2000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	Tx Coefficient C+1[5:0]						-	-	Tx Coefficient C0[5:0]					
Initial Value	0	0	1	1	0	0	0	0	0	0	0	1	0	0	1	1
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Tx Coefficient C-1[5:0]						-	-	-	-	-	-	-	-
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
29 to 24	Tx Coefficient C+1[5:0]	30h	RW	Renesas Private bit Specify d'3.
23, 22	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
21 to 16	Tx Coefficient C0[5:0]	13h	RW	Renesas Private bit Specify d'19.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
13 to 8	Tx Coefficient C-1[5:0]	20h	RW	Renesas Private bit Specify d'2.
7 to 0	-	0h	R	Reserved These bits are read as 0b. The write value should always be 0b.

(5) PIPE Status and Control Register 4 (USB3m_HOST_U3P0PIPESC4)

PIPE Status and Control Register 4.

Access Size : 32 bits

Address : <USB3m_host_base> + 10D0h

Initial Value : 0018_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	Tx Coefficient C+1[5:0]						-	-	Tx Coefficient C0[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Tx Coefficient C-1[5:0]						-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
29 to 24	Tx Coefficient C+1[5:0]	0h	RW	Renesas Private bit Specify d'0.
23, 22	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
21 to 16	Tx Coefficient C0[5:0]	18h	RW	Renesas Private bit Specify d'24.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0b.
13 to 8	Tx Coefficient C-1[5:0]	0h	RW	Renesas Private bit Specify d'0.
7 to 0	-	0h	R	Reserved These bits are read as 0b. The write value should always be 0b.

6.4.4.5 Description of Functions

The prefix (USB3m_HOST_) of the register names is omitted in this and subsequent sections.

6.4.4.5.1 Battery Charging

The USB3HOST supports the Battery Charging function of the Charging Port Device which is on the supply side.

The Battery Charging function is used to pull more current from VBUS than the current specified in the USB specification.

The following Port Modes are supported in the USB3HOST;

- Standard Downstream Port (SDP)
- Charging Downstream Port (CDP)
- Dedicated Charging Port (DCP)

Remark For detail, refer to the Battery Charging Specification Revision1.2.

(1) Setting Battery Charging Mode

The user can select one of BC modes by setting BCCTRL.BC_MODE_P[m] (m: 1)'s field in Battery Charging Control Register (BCCTRL) (refer to **(a) Battery Charging Control Register (USB3m_HOST_BCCTRL)**), when the user uses the battery charging function.

Table 6.4-16 BC_MODE_P[m]'s information of Battery Charging Control Register (BCCTRL)

Bit	Field	SW	Default Value	Function
1:0	BC_MODE_P1	RW	0h	Battery Charging Mode for Port1 0h: SDP 1h: CDP 2h: DCP 3h: Reserved

(2) How to Set Battery Charging Mode

The host driver can set the battery charging mode by executing the flow in **Figure 6.4-3**.

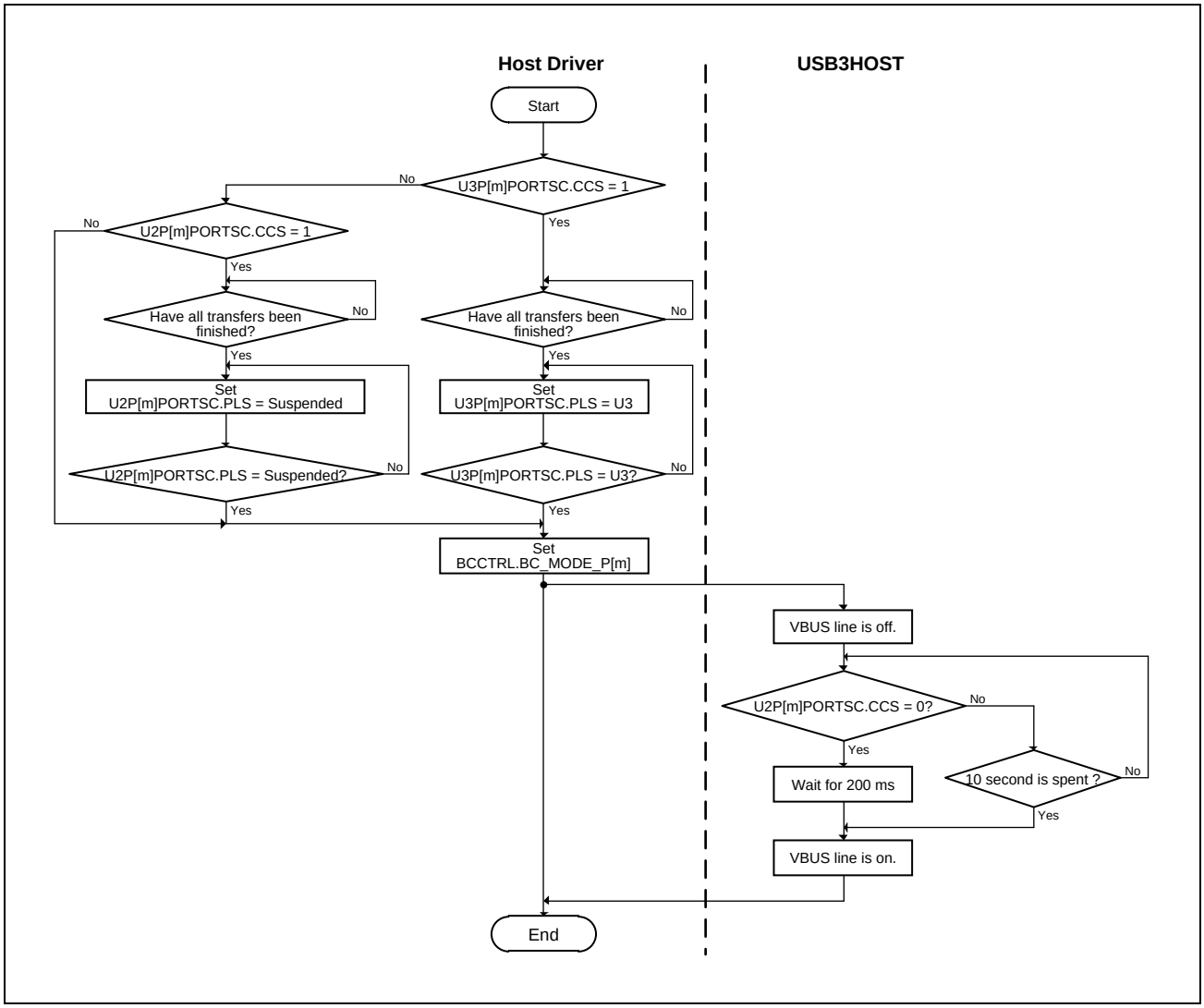


Figure 6.4-3 Host Driver's Flow to Set BC Mode (m = 1)

6.4.4.5.2 Initialization Flow

Figure 6.4-4 shows the “initialization flow” to the USB3HOST in USB system.

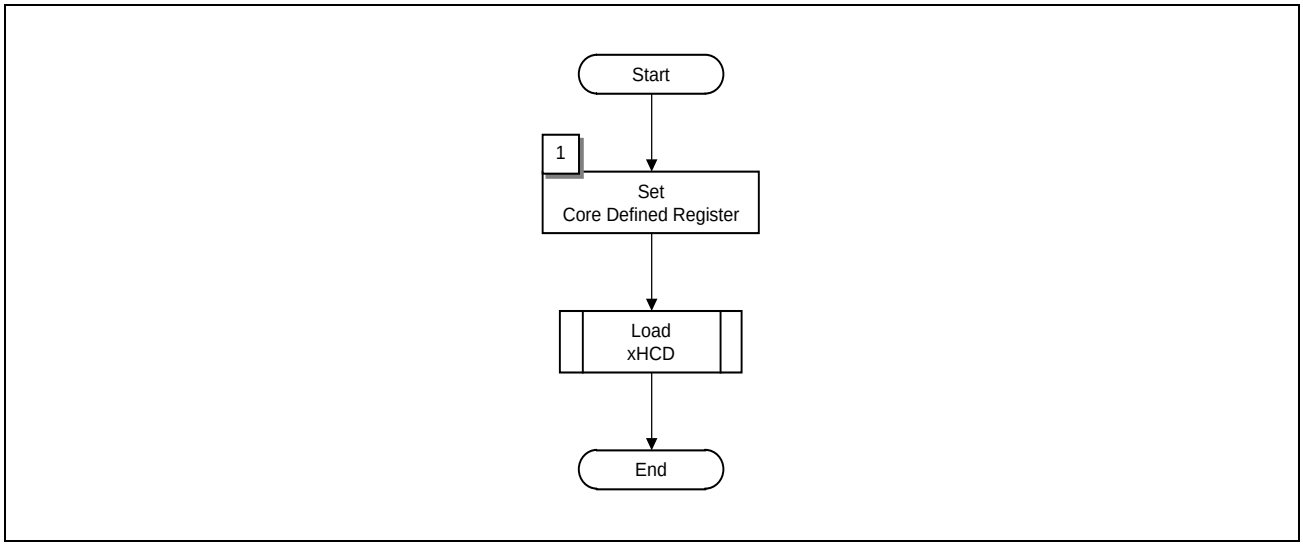


Figure 6.4-4 Initialization Flowchart

1. Set Core Defined Register
   Information of related interrupts is set in this phase.

6.4.4.5.3 xHCD Initialization Flow

Figure 6.4-5 shows the “initialization flow by xHCD (xHCI Driver)” to the USB3HOST in USB system.

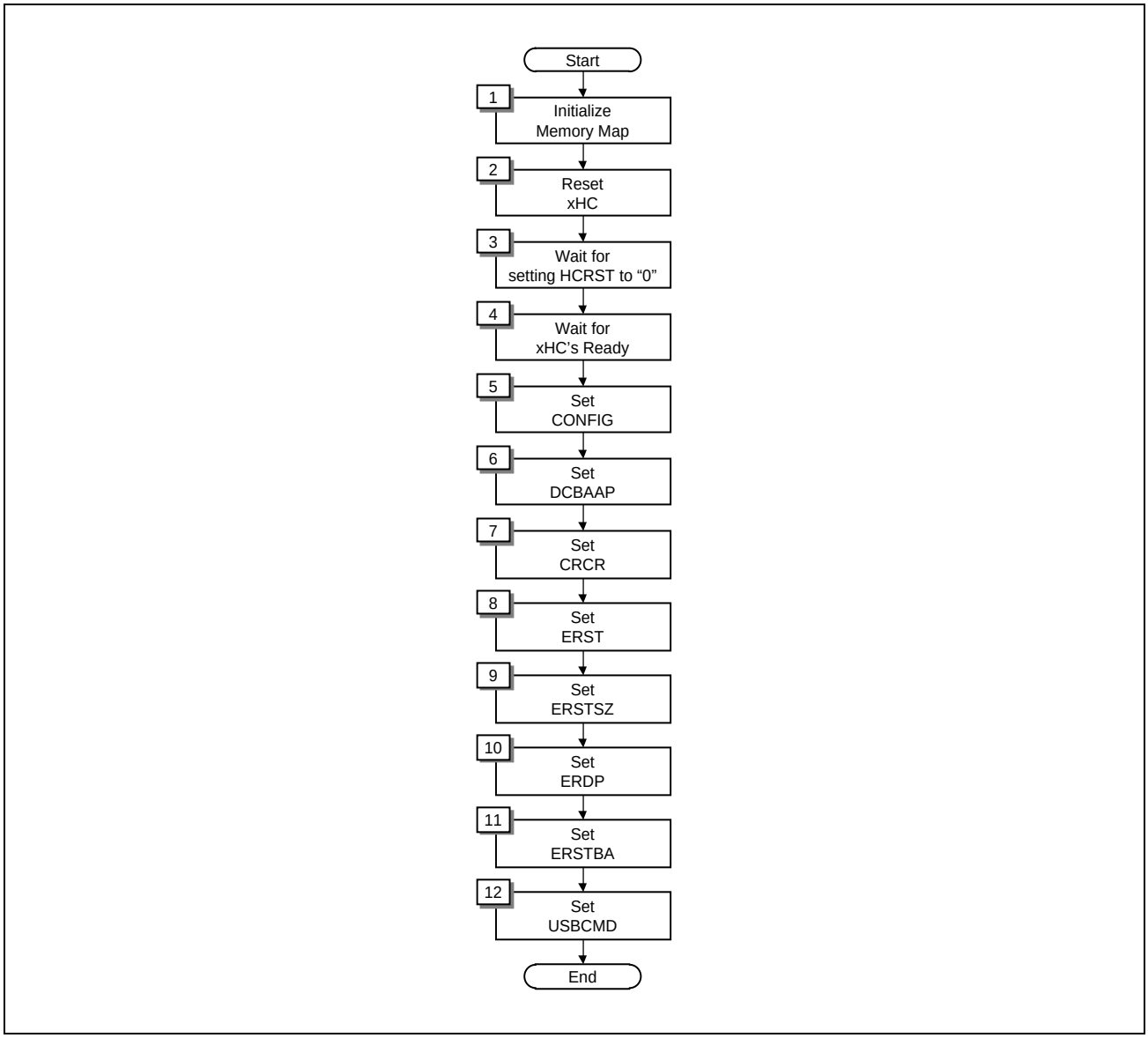


Figure 6.4-5 Initialization Flowchart by xHCD (xHCI Driver)

1. Initialize Memory Map
   xHCD initializes the memory area in the system.
2. Reset xHC
   xHCD confirms that USBSTS.HCH is set to “1b”.
   Then, xHCD resets xHC (the USB3HOST) by setting USBCMD.HCRST to “1b”.
3. Wait for setting HCRST to “0b”
   xHCD waits for setting USBCMD.HCRST to “0b”.
4. Wait for xHC’s Ready
   xHCD waits for setting USBSTS.CNR to “0b”.

5. Set CONFIG
   xHCD sets CONFIG.MaxSlotsEn to a valid value.
6. Set DCBAAP
   xHCD sets the “Device Context Base Address Array (DCBAAP)” to a valid value.
7. Set CRCR
   xHCD sets the “Command Ring Control Register (CRCR)” to the Command Ring’s address.
8. Set ERST
   xHCD sets the “Event Ring Segment Table (ERST)” to the Event Ring’s address and size in the system memory.
9. Set ERSTSZ
   xHCD sets the “Event Ring Segment Table Size (ERSTSZ)” to the Event Ring Segment Table’s size.
10. Set ERDP
    xHCD sets the “Event Ring Dequeue Pointer (ERDP) Register” to the Event Ring’s read address.
11. Set ERSTBA
    xHCD sets the “Event Ring Segment Table Base Address (ERSTBA) Register” to the Event Ring Segment Table’s address.
12. Set USBCMD
    xHCD sets USBCMD.INTE to”1b” and also sets USBCMD.R/S to “1b”.

6.4.4.5.4 Interrupt Flow

(1) “all_int” Interrupt Flow

Figure 6.4-6 shows “all_int” interrupt flow in USB system.

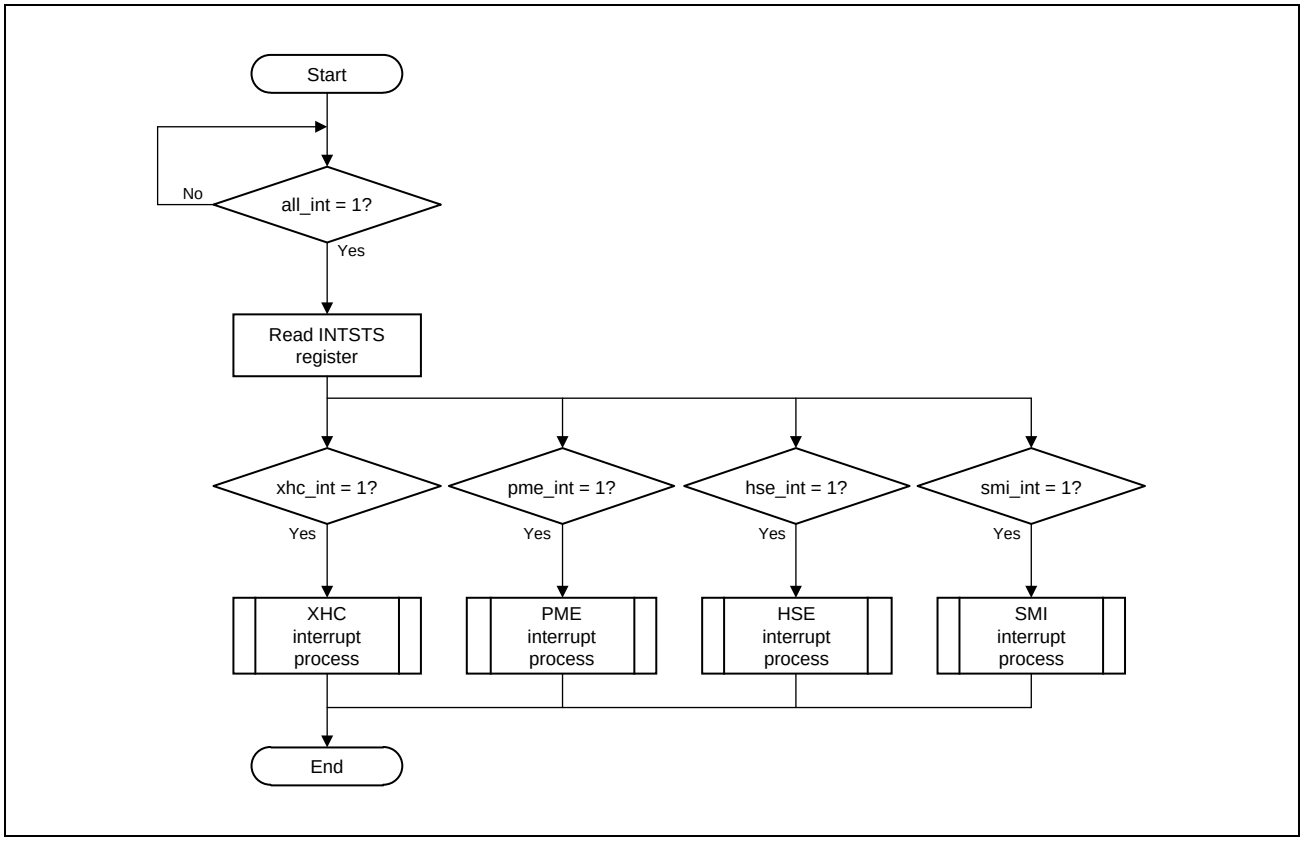


Figure 6.4-6 “all_int” Interrupt Flowchart

(2) XHC Interrupt Flow

Figure 6.4-7 shows “xhc_int” interrupt flow in USB system.

In the case of “CFGSTS1.INTEDGEN = 0b”, the notification mode of “xhc_int” is the Level mode.

(This mode is the same as INTA# in PCI System.)

In the case of “CFGSTS1.INTEDGEN = 1b”, the notification mode of “xhc_int” is the Pulse mode.

(This is the same as MSI / MSI-X in PCI System.)

In this LSI, “level mode” is selected as default interrupt mode.

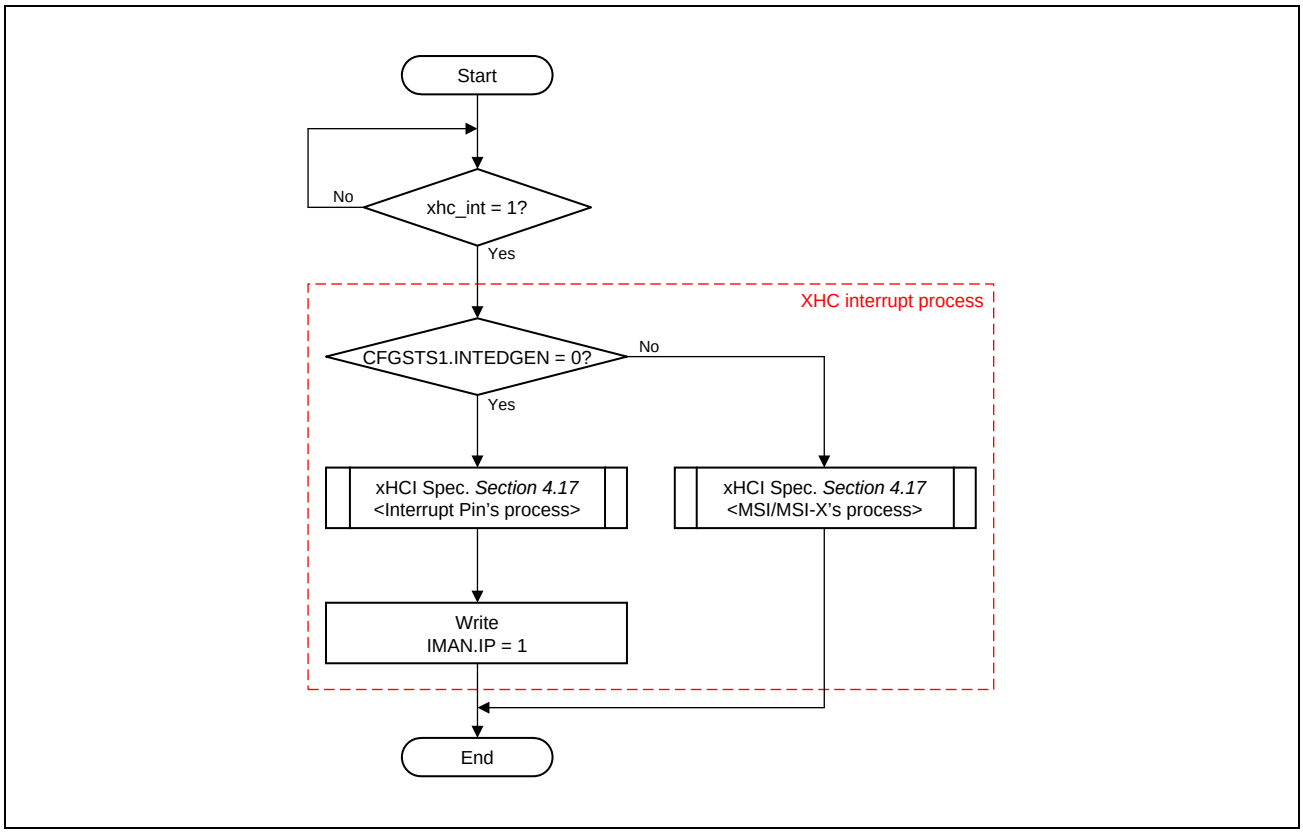


Figure 6.4-7 XHC Interrupt Flowchart

(3) PME Interrupt Flow

Figure 6.4-8 shows the “pme_int” interrupt flow in USB system.

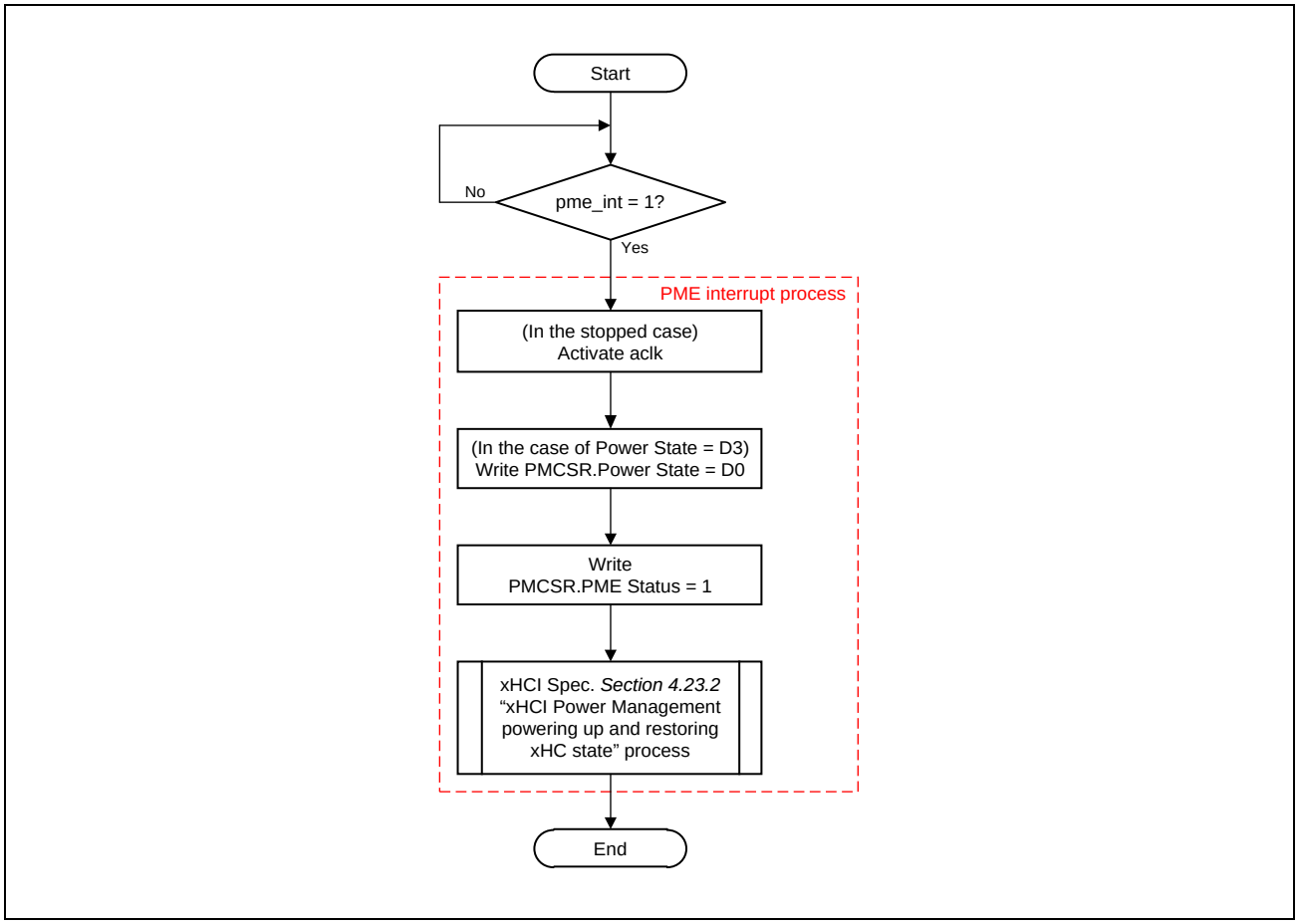


Figure 6.4-8 PME Interrupt Flowchart

(4) HSE Interrupt Flow

Figure 6.4-9 shows the “hse_int” interrupt flow in USB system.

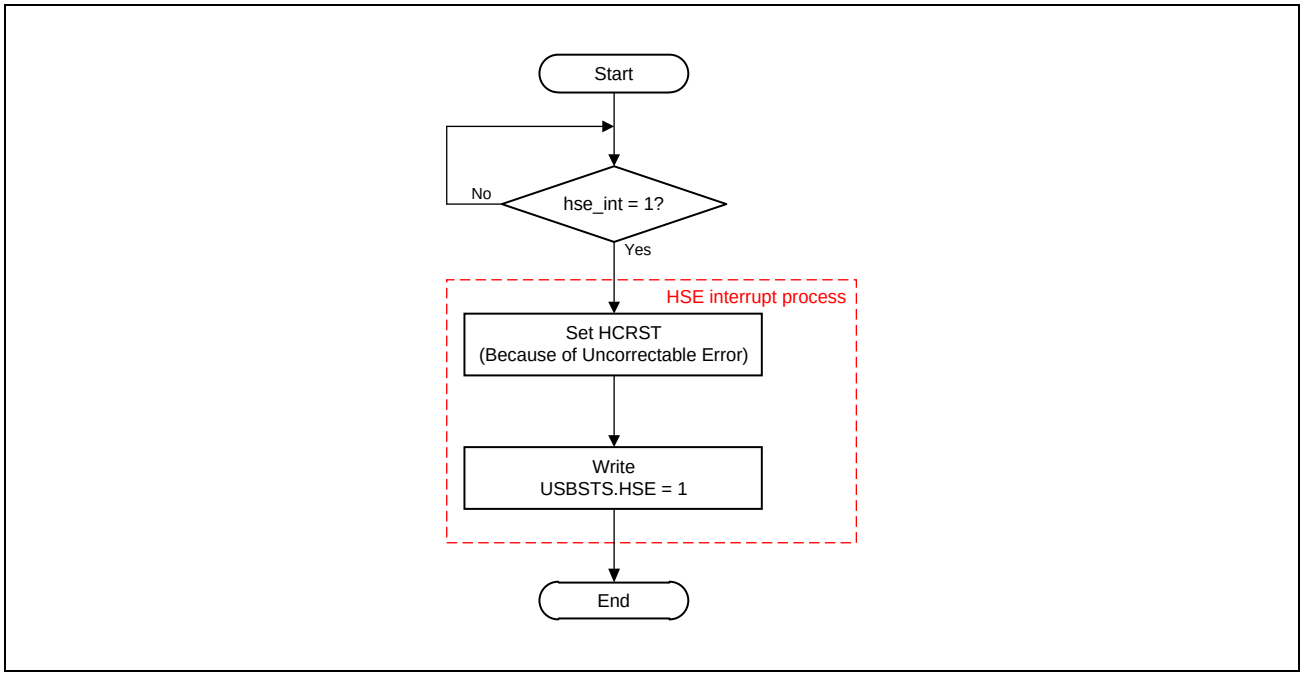


Figure 6.4-9 HSE Interrupt Flowchart

(5) SMI Interrupt Flow

Figure 6.4-10 shows the “smi_int” interrupt flow in USB system.

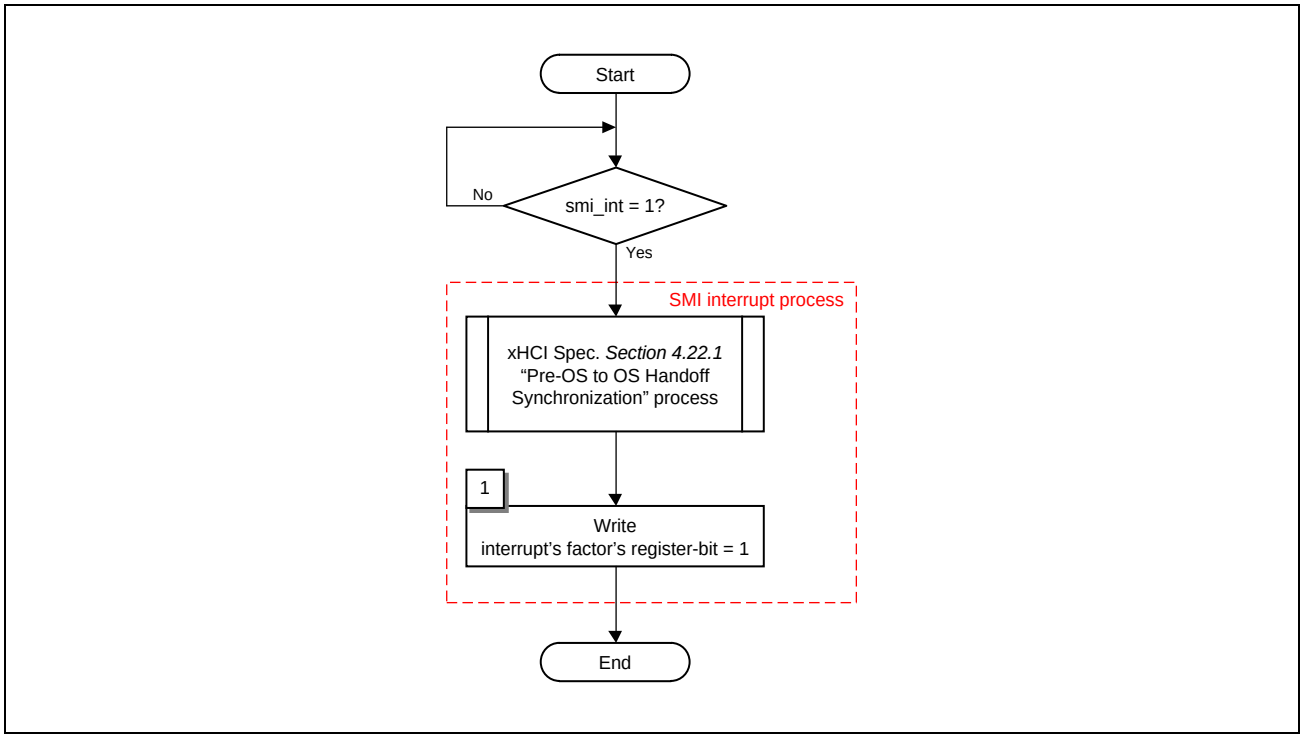


Figure 6.4-10 SMI Interrupt Flowchart

1. Write interrupt’s factor’s register-bit = 1b
   If this interrupt’s factor is USBLEGCTLSTS.SBA, write “1b” to USBLEGCTLSTS.SBA.
   If this interrupt’s factor is USBLEGCTLSTS.SPC, write “1b” to USBLEGCTLSTS.SPC.
   If this interrupt’s factor is USBLEGCTLSTS.SOOC, write “1b” to USBLEGCTLSTS.SOOC.
   If this interrupt’s factor is USBLEGCTLSTS.SHSE, write “1b” to USBSTS.HSE.
   If this interrupt’s factor is USBLEGCTLSTS.SEI, write “1b” to USBSTS.EINT.

6.4.4.5.5 xHCI Command

Figure 6.4-11 shows an abstract of xHCI Command flow by using the USB3HOST.

Refer to "*Section 4.11.3 Event TRBs*" and "*Section 4.11.4 Command TRBs*" of the *xHCI specification* for detail information.

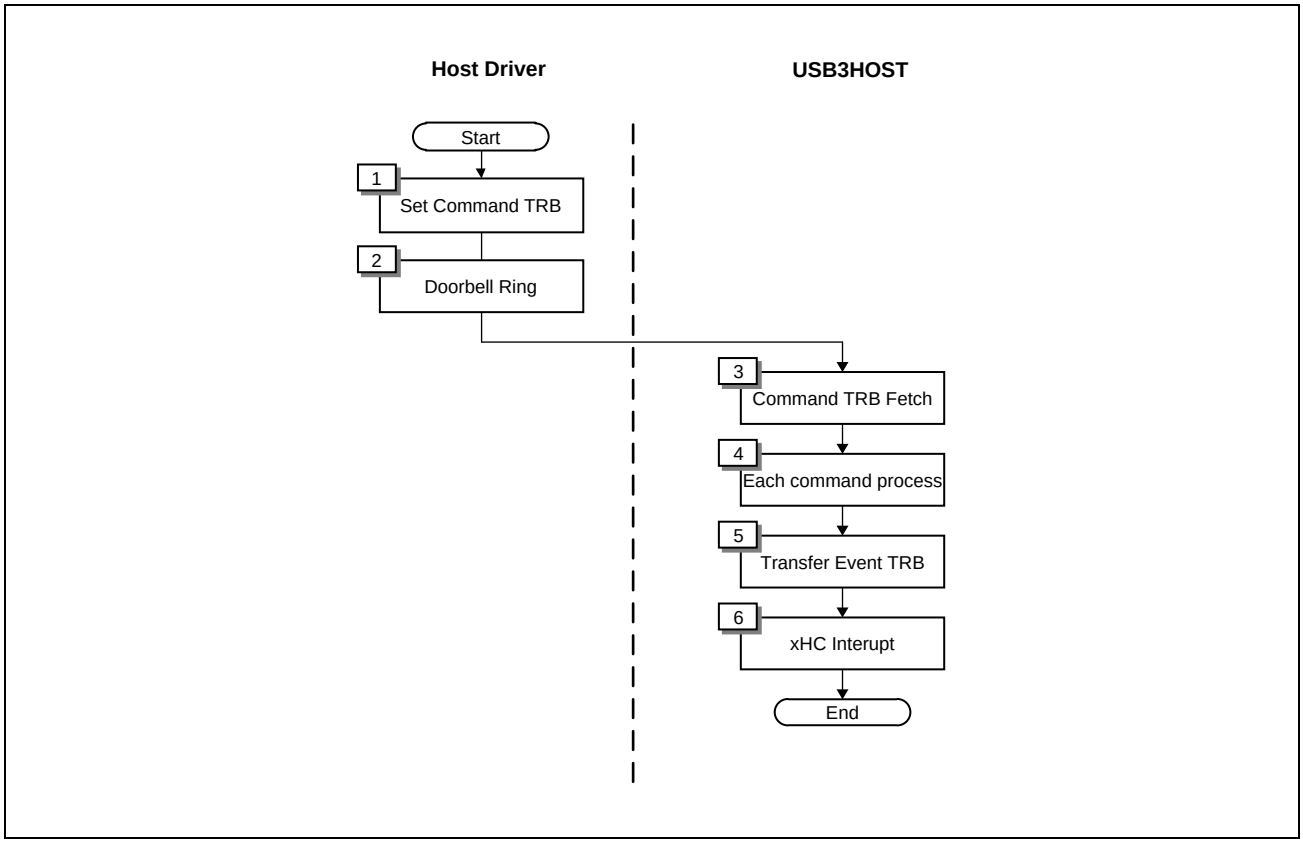


Figure 6.4-11 xHCI Command Flowchart

1. Set Command TRB
   Host driver sets the Command TRB in the Command Ring.
2. Doorbell Ring
   Host driver rings the Doorbell to prepare data for each xHCI command's process.
3. Command TRB Fetch
   The USB3HOST fetches Command TRB from Command Ring,
4. Each command process
   The USB3HOST decodes that command's content and executes each command process.
5. Transfer Event TRB
   The USB3HOST writes Event TRB to Event Ring.
6. xHC Interrupt
   The USB3HOST sends the interrupt to the system.

6.4.4.5.6 Examples of Transaction Flows

This section describes transactions when the USB3HOST is used.

As examples, Bulk OUT and Bulk IN transactions are described.

The USB3HOST complies with the xHCI Specification.

Refer to "*Section 4.11.2 Transfer TRBs*" and "*Section 4.11.3 Event TRBs*" of the *xHCI Specification* for detail information.

Bulk OUT Transaction Flow

Bulk OUT transaction flow is shown below.

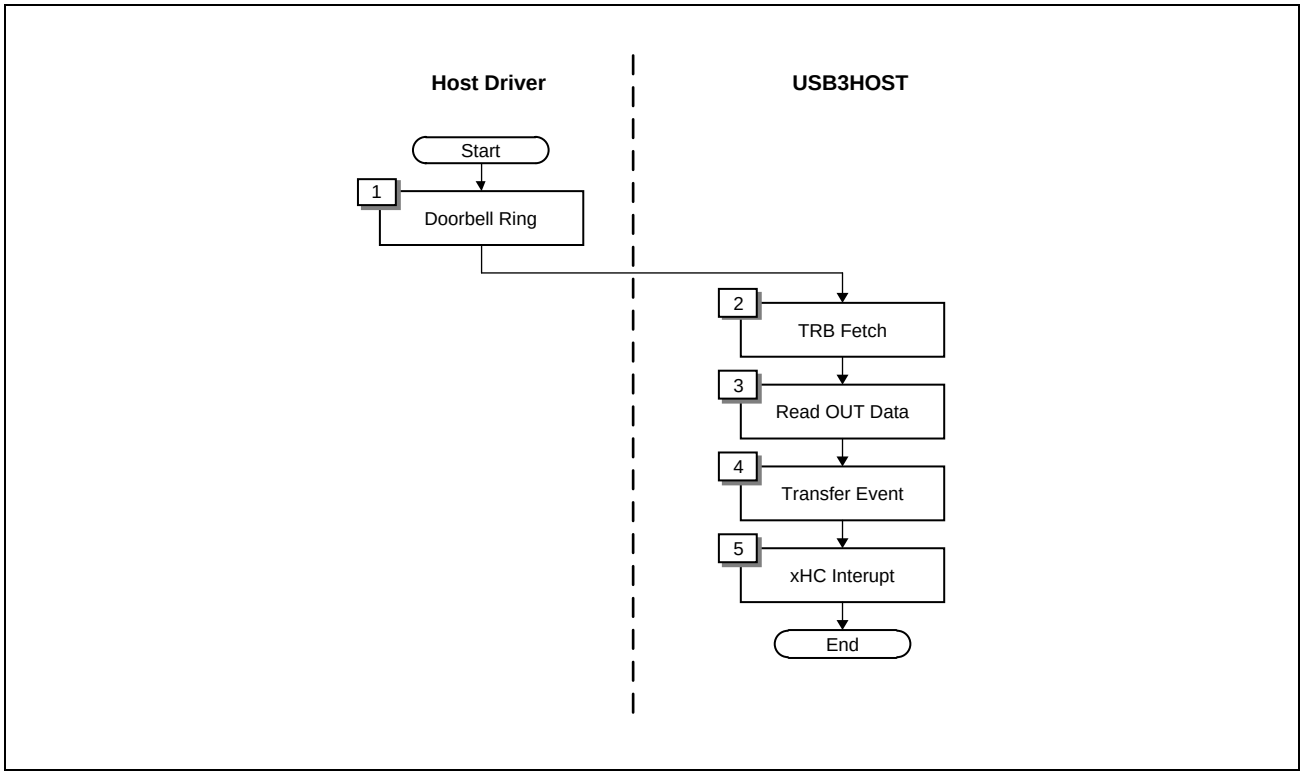


Figure 6.4-12 Example of Bulk OUT Transaction Flowchart

1. Doorbell Ring
   Host Driver rings Doorbell to prepare Transfer Data.
2. TRB Fetch
   The USB3HOST fetches TRB from OUT Transfer Ring,
3. Read OUT Data
   The USB3HOST reads OUT Data from Data Buffer Pointer.
4. Transfer Event
   The USB3HOST writes Transfer Event TRB to Event Ring.
5. xHC Interrupt
   The USB3HOST sends the interrupt to the system.

Bulk IN transaction

Bulk IN transaction flow is shown below.

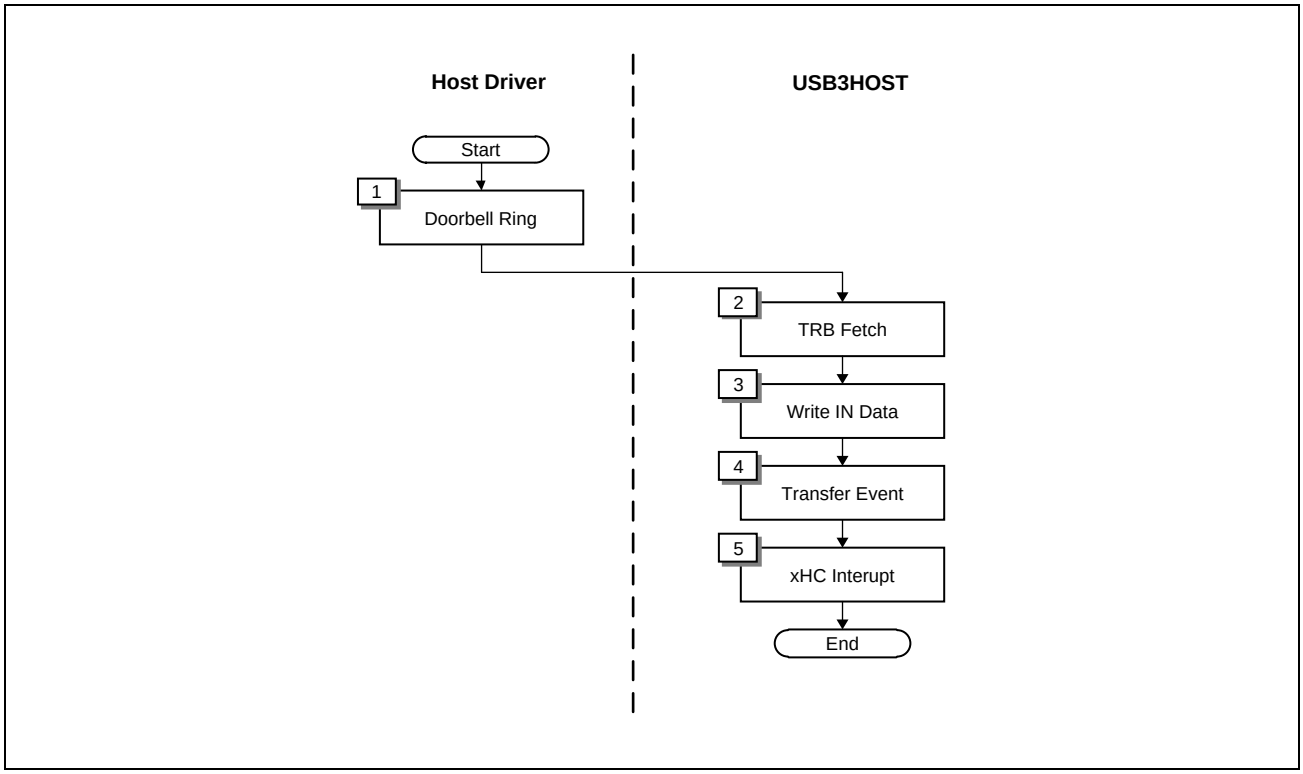


Figure 6.4-13 Example of Bulk IN Transaction Flowchart

1. Doorbell Ring
   Host driver rings Doorbell to prepare Transfer Data.
2. TRB Fetch
   The USB3HOST fetches TRB from OUT Transfer Ring.
3. Write IN Data
   The USB3HOST writes IN Data to Data Buffer Pointer.
4. Transfer Event
   The USB3HOST writes Transfer Event TRB to Event Ring.
5. xHC Interrupt
   The USB3HOST sends the interrupt to the system.

6.4.4.5.7 Functional Implementation Constraint

The USB3HOST core has functional implementation constraints.

One packet shall be composed of 8 TRBs or less.

If one packet is composed of nine or more TRBs, the USB3HOST generates a TRB Error for the higher-level system.

SW shall not set to U3 state before an event of the Force Header Command is completed.

If SW sets to U3 state before this event is completed, the USB3HOST results in undefined xHC behavior.

6.4.5 USB Test Module (USB3TEST/USB2TEST)

6.4.5.1 Features

This section mainly describes the USB test modules (USB3TEST and USB2TEST) which are connected between the USB3HOST and the PHY module. The main functions of USB3TEST/USB2TEST are as follows.

- Reset control
- Control of PHY input pins
- Monitoring of PHY output pins

For more information, refer to the User's Manual Additional Document.

SECTION 6 HIGH-SPEED INTERFACE

6.5 USB2.0 Interface

This section describes the USB2 functions of this LSI.

This LSI includes 1 channel USB2.0 OTG/DRD (Host/Function) interface and 1 channel USB2.0 Host interface.

This manual is a simplified version. For more information, refer to the User's Manual Additional Document.

6.5.1 USB2.0 PHY Controller

This section describes the overview of USB2.0 and USB2.0 PHY control.

The detail function of USB2.0 Host controller and the common function both Host module and Function module are described in **6.5.2 USB2.0 Host Controller**.

The detail function of USB2.0 Function controller is described in **6.5.3 USB2.0 Function Controller**.

6.5.1.1 Features

This interface complies the following specifications.

- Universal Serial Bus Specification Revision 2.0
- On-The-Go and Embedded Host Supplement to the USB Revision 2.0 Specification Revision 2.0 plus errata and ecn*[1]
- Enhanced Host Controller Interface (EHCI) Specification for Universal Serial Bus Revision 1.0
- EHCI v1.1 Addendum*[2]
- Open Host Controller Interface (OHCI) Specification for USB Release 1.0a
- Battery Charging Specification Revision 1.2*[3]

Note 1. Session Request Protocol(SRP) and Host Negotiation Protocol(HNP) are not supported

Note 2. Some EHCI v1.1 feature are not supported. (Refer to **6.5.2 USB2.0 Host Controller** for detail)

Note 3. DCP (Dedicated Charging Port) is not supported as downstream port.
For this LSI, IDP_SRC will flow to a maximum of 32 µA.

6.5.1.1.1 Ch0: USB2.0 OTG/DRD (host/function) interface

The following table shows features of this interface.

Table 6.5-1 Ch0: USB2.0 OTG/DRD (Host/Function) Interface Features

Function	Description
Host mode	• Support mode: – High-Speed (480 Mbps) – Full-Speed (12 Mbps) – Low-Speed (1.5 Mbps) • Support Isochronous / Interrupt/Control / Bulk transfer modes • Support Isochroous / Interrupt high bandwidth transfer
Function mode	• Support mode: – High-Speed (480 Mbps) – Full-Speed (12 Mbps) • Isochronous / Interrupt / Control / Bulk transfer • Up to 10 ch PIPE (includes default control PIPE)
Other functions	• OTG function (Rev. 2.0) • Battery Charging function • DRD (Dual-Role-Device) function (Static switch between Host and Function)

6.5.1.1.2 Ch1: USB2.0 host interface

The following table shows features of this interface.

Table 6.5-2 Ch1: USB2.0 Host Interface Function Features

Features	Description
Host mode	• Support mode: – High-Speed (480 Mbps) – Full-Speed (12 Mbps) – Low-Speed (1.5 Mbps) • Support Isochronous / Interrupt / Control / Bulk transfer modes • Support Isochroous / Interrupt high bandwidth transfer
Other function	• Battery Charging function

6.5.1.1.3 Block diagram

The block diagram of USB interface is as follows.

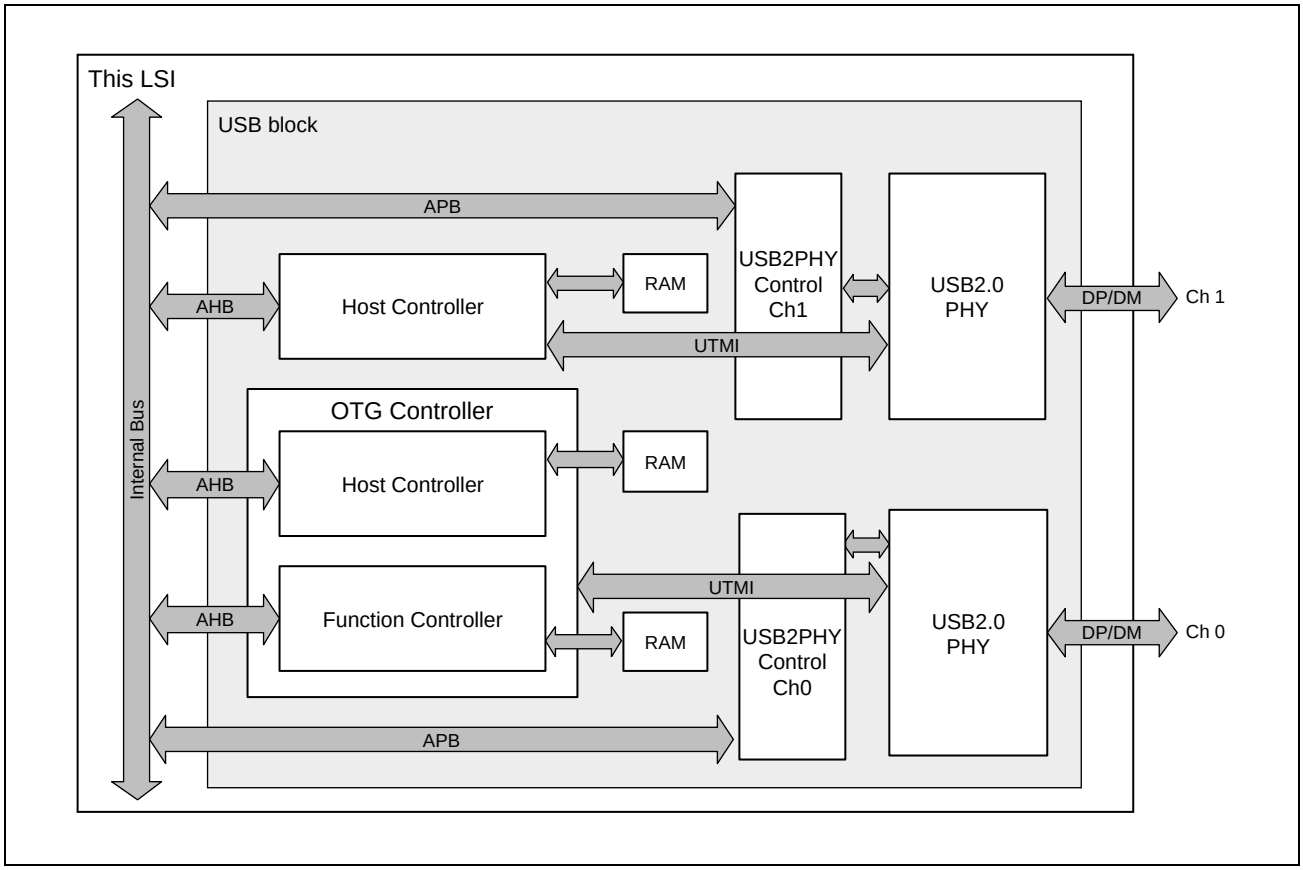


Figure 6.5-1 USB Interface Block Diagram

6.5.1.1.4 External signal pins

The following table shows external signal pins for USB block.

Table 6.5-3 External Signal Pins

Name	Input/Output	Description	Initial Value
USB20_DP	I/O	USB data pin Data+ (Ch0)	—
USB20_DM	I/O	USB data pin Data− (Ch0)	—
USB20_OVRCURN	Input	Over current detection input (Ch0)	—
USB20_VBUSEN	Output	VBUS output enable (Ch0)	0b
USB20_VUBUSIN	Input	Peripheral VBUS input (Ch0) *Note 1.* To connect a resistance between USB20_VUBUSIN and USB VBUS. See **Figure 6.5-36** for how to connect the USB VBUS. *Note 2.* Need to countermeasure against reverse current via the USB20_VUBUSIN from USB2_VDD33 and USB20_VDD18. For the details, refer to “3.2.1 Power supply [RZ/G3E]” of the *PCB Design Guidelines for High Speed Signal Interfaces.*	—
USB20_OTGEXICEN	Output	External power IC control (Ch0)	0b
USB20_OTGID	Input	OTG ID input (Ch0)	—
USB21_DP	I/O	USB data pin Data+ (Ch1)	—
USB21_DM	I/O	USB data pin Data− (Ch1)	—
USB21_OVRCURN	Input	Over current detection input (Ch1)	—
USB21_VBUSEN	Output	VBUS output enable (Ch1)	0b

6.5.1.1.5 Power and GND pins

The following table shows Power and GND pins for USB block.

Table 6.5-4 Power and GND Pins

Name	Description
USB2_USDVDD	Common core power supply (0.8 V) for USB2PHY implemented in USB2 ch0/ch1 and USB3. *Note*: Need to countermeasure against reverse current from USB2_USDVDD. For the details, refer to “3.2.1 Power supply [RZ/G3E]” of the *PCB Design Guidelines for High Speed Signal Interfaces*.
USB20_USVDD18	USB2PHY ch0 Internal Regulator Power (1.8 V), HS receiver power (1.8 V)
USB2_USVDD33	Common core power supply (3.3 V) for USB2PHY implemented in USB2 ch0/ch1 and USB3.
USB20_TXRTUNE	USB2PHY ch0 Reference voltage *Note*: To connect a pull-down resistor. Resistance: 200Ω ±1%.
USB21_USVDD18	USB2PHY ch1 Internal Regulator Power (1.8 V), HS receiver power (1.8 V)
USB21_TXRTUNE	USB2PHY ch1 Reference voltage *Note*: To connect a pull-down resistor. Resistance: 200Ω ±1%.

6.5.1.1.6 Interrupt

The following table shows interrupt list for USB block.

Table 6.5-5 USB Interrupt

Name	Description	Active Level	Type
U2H0_INT	USB2.0 Host AHB Interrupt (Ch0)	H	Level-Sensitive
U2H0_OHCI_INT	USB2.0 Host OHCI Interrupt (Ch0)	H	Level-Sensitive
U2H0_EHCI_INT	USB2.0 Host EHCI Interrupt (Ch0)	H	Level-Sensitive
U2H0_WAKEON_INT	USB2.0 Host EHCI Wakeup Interrupt (Ch0)	H	Level-Sensitive
U2H0_OBINT	USB2.0 Host OTG and Battery Charging Interrupt (Ch0)	H	Level-Sensitive
U2H1_INT	USB2.0 Host AHB Interrupt (Ch1)	H	Level-Sensitive
U2H1_OHCI_INT	USB2.0 Host OHCI Interrupt (Ch1)	H	Level-Sensitive
U2H1_EHCI_INT	USB2.0 Host EHCI Interrupt (Ch1)	H	Level-Sensitive
U2H1_WAKEON_INT	USB2.0 Host EHCI Wakeup Interrupt (Ch1)	H	Level-Sensitive
U2H1_OBINT	USB2.0 Host OTG and Battery Charging Interrupt (Ch1)	H	Level-Sensitive
U2P_IXL_INT	USB2.0 Function controller Interrupt (Ch0)	H	Edge-Triggered
U2P_INT_DMA[1:0]	USB2.0 Function controller DMA transaction complete interrupt (Ch0)	H	Level-Sensitive
U2P_INT_DMAERR	USB2.0 Function controller DMA error response interrupt (Ch0)	H	Level-Sensitive

6.5.1.2 Interface Registers

6.5.1.2.1 USB2PHY Control Register

USB2PHY Control Register mainly controls reset and power down of the USB2PHY.

Table 6.5-6 Register Base Addresses

Unit Name	Base Address Name	Base Address
USB20	<USB20_phy_base>	0_1583_0000h (5583_0000h*1, 4583_0000h*2)
USB21	<USB21_phy_base>	0_1584_0000h (5584_0000h*1, 4584_0000h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

The prefix (USB2m_PHY_) of the register names is omitted in the register descriptions and the field descriptions in this section.

The prefix (USB2m_PHY_) means below (m = 0, 1):

- USB20_PHY_*: USB20 registers
- USB21_PHY_*: USB21 registers

For more information, refer to the User's Manual Additional Document.

6.5.1.3 Operation

For more information, refer to the User's Manual Additional Document.

6.5.2 USB2.0 Host Controller

6.5.2.1 Overview

This LSI has two USB 2.0 host/function modules. For each module, you can switch between the host mode and the peripheral mode by specifying the UCOM register setting. This section describes the circuits that are common to both modes, and the host controller itself.

6.5.2.1.1 Features

This module has the following features:

Function	Description
Host function	• Supporting high-speed (480 Mbps), full-speed (12 Mbps), and low-speed (1.5 Mbps) transfers • Compliant with Open Host Controller Interface (OHCI) Specification for USB Revision 1.0a • Compliant with Enhanced Host Controller Interface (EHCI) Specification for USB Revision 1.1*1
Other functions	• Battery charging (compliant with Battery Charging Specification Revision 1.2)*2 • Dual-role-device function (static switching between the USB host and USB peripheral functions)

Note 1. Some functions (specifications) are not supported.

Note 2. The setting for battery charging is handled by the host controller even if the function controller is selected.
This LSI does not support DCP (Dedicated Charging Port) as a Downstream Port.

(1) EHCI v1.1 functions

Conventional host controller modules comply with EHCI Specification Revision.1.0. Meanwhile, this module supports the additional functions listed below to comply with EHCI v1.1 Addendum.

Note, however, that this module supports only the functions marked with "✓" in the table below.

Function Name	Support
Per-Port Change Events	✓
Shorter Periodic Frame List	✓
Hardware Prefetching	—
Link Power Management (LPM)	✓

Of the three features to support, the registers associated with "Per-Port Change Events" and "Shorter Periodic Frame List" are listed below. For more information on each register, refer to the appropriate register specification in **6.5.2.2 Host Registers**.

"Link Power Management (LPM)" is described in **(2) Link power management (LPM) function**.

Table 6.5-7 Per-Port Change Events

Relevant Register	Relevant Bit		Attribute	Function
HCCPARAMS (offset: 108h)	bit 18	Per-Port Change Event Capability	R	This bit indicates support for this function. This bit is fixed to 1b.
USBCMD (offset: 120h)	bit 15	Per-Port Change Events Enable	RW	Per-Port event notification setting. Writing 1b to this bit enables the Per-Port event notification function.
USBSTS (offset: 124h)	bit 16	Port-1 Change Detect	RW(1)	If a port-1 change event is detected, 1b is written to this bit.
USBINTR (offset: 128h)	bit 16	Port-1 Change Event Enable	RW	Enable/Disable setting of the above "port-1 Change Detect" field. To reflect the port event on the above field for the port, set the corresponding bit to 1b.

Table 6.5-8 Shorter Periodic Frame List

Relevant Register	Relevant Bit		Attribute	Function
HCCPARAMS (offset: 108h)	bit 19	32-Frame Periodic List Capability	R	This bit indicates support for this function. This bit is fixed to 1b.
USBCMD (offset: 120h)	bit[3:2]	Frame List Size	RW	This bit determines the Frame List Size. To determine 32-frames, set this field to 11b.

(2) Link power management (LPM) function

This module supports the power management function conforming to “USB 2.0 Link Power Management Addendum to the Universal Serial Bus v2.0 Specification” (abbreviated as LPM).

When the peripheral device which supports the LPM function is connected to this module, the device can be moved to the suspend state faster than conventional by using the LPM function.

Remark When using the LPM function: 8 to 10 µs

The relevant registers are listed in the following. Refer to the appropriate register specification of each register detail in **6.5.2.2 Host Registers**.

Relevant Register	Relevant Bit		Attribute	Function
HCCPARAMS (offset: 108h)	bit 17	Link Power Management Capability	R	This bit indicates support for this function. This bit is fixed to 1b.
USBCMD (offset: 120h)	bit[27:24]	Host-Initiated Resume Duration	RW	The minimum duration time of K-state drive for resumption from LPM state.
PORTSC1 (offset: 164h)	bit[31:25]	Device Address [7:0]	RW	The device address of the destination of LPM Token. When using LPM function, this setting is needed. Set the address of the device connected to the corresponding port before sending LPM Token.
	bit[24:23]	Suspend Status [1:0]	R	The device response to LPM Token.
	bit 9	Suspend using L1	RW	When using LPM function, set this bit to 1b.
	bit 7	Suspend	RW	When this bit is set to 1b this module starts LPM Token transaction, in condition of “Suspend using L1 bit = 1b” and “Device address field = 000h”.
PORT_LPM_CTR1 (offset: 320h)	bit[7:4]	NYET_RETRY_CNT_P1 [3:0]	RW	The number of retries for NYET response of device in LPM transaction.
	bit 3	REMOTEWAKE_EN_P1	RW	Setting of LPM RemoteWakeup permission. 0b: permitted (default) 1b: not permitted
	bit 2	SLEEP_INT_EN_P1	RW	In LPM transaction, setting to generate an interrupt or not when a response other than ACK is received. 0b: not to generate an interrupt (default) 1b: to generate an interrupt
	bit 1	RETRY_ENABLE_NYET_P1	RW	In LPM transaction, setting to the host behavior when NYET response from the device is received. 0b: not to Retry (default) 1b: to Retry
	bit 0	HIRD_SEL_P1	RW	Setting the duration time of the K-state drive for resumption from LPM state.

(3) OTG function

Detects a change of the USB0_OTG_ID pin by U2H0_OBINT interrupt and switches the Host/Function role. Confirm the IDCHG_STA bit of **(2) OTG-BC Interrupt Status Register (USB2m_HOST_OBINTSTA)** and the IDMON bit of **(5) Line Control Port 1 Register (USB2m_HOST_LINECTRL1)**.

Host/Function role switching is performed by the OTG_PERI bit of **(1) Common Control Register (USB2m_HOST_COMMCTRL)**.

OTG_PERI bit
0: Host mode
1: Function mode (Only CH0 can be set)

(4) Battery-charging function

The Control and monitoring of the Battery Charging I/F of the USBPHY is set in the UCOM register of this module. For the detail of the UCOM register, refer to **6.5.2.2.6 UCOM Register Descriptions**.

(5) Suspend extension function

This module implements the following two suspend extension functions to reduce power consumption by stopping PLL of USBPHY.

For the detail of the register, refer to **(6) Suspend Control Register (USB2m_HOST_SPD_CTRL)**.

[Relevant register]

Suspend Control Register (offset: 308h)

[Functional specification]

Function		Relevant Bit	
[1]	Function to assert USBPHY SUSPENDM by asserting the Suspend bit in the OHCI/EHCI Operational Register	bit 31	SUSPENDM_ENABLE
[2]	Function to forcibly assert USBPHY SUSPENDM	bit 0	GLOBAL_SUSPENDM_P1

6.5.2.1.2 Support of USB-related specifications

USB-Related Specification or Function			Support
Host function	High Speed	Bulk IN/OUT transfer	✓
		Control IN/OUT transfer	✓
		Isochronous IN/OUT transfer	✓
		Isochronous high-bandwidth transfer	✓
		Interrupt IN/OUT transfer	✓
	Full Speed	Bulk IN/OUT transfer	✓
		Control IN/OUT transfer	✓
		Isochronous IN/OUT transfer	✓
		Interrupt IN/OUT transfer	✓
	Low Speed	Control IN/OUT transfer	✓
		Interrupt IN/OUT transfer	✓
	No. of hub connection stages	HS: 5 stages	✓
		FS: 5 stages	✓
	Support of EHCI V1.1	Hardware Prefetching	×
		Link Power Management	✓
		Per-Port Change Events	✓
		Shorter Periodic Frame List	✓
Battery-charging function (hereafter called the “BC function”)			✓
Dual role device * Function to statically switch between the host and peripheral modes			✓

Remarks: ✓: Support
×: No support

6.5.2.2 Host Registers

Table 6.5-9 Register Base Addresses

Unit Name	Base Address Name	Base Address
USB20	<USB20_host_base>	0_1580_0000h (5580_0000h*1, 4580_0000h*2)
USB21	<USB21_host_base>	0_1581_0000h (5581_0000h*1, 4581_0000h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

The prefix (USB2m_HOST_) of the register names is omitted in the register descriptions and the field descriptions in this section.

The prefix (USB2m_HOST_) means below (m = 0, 1):

- USB20_HOST_*: USB20 registers
- USB21_HOST_*: USB21 registers

6.5.2.2.1 List of Registers

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
HcRevision	USB2m_HOST_HcRevision	0000_0000h	0000h	32
HcControl	USB2m_HOST_HcControl	0000_0000h	0004h	32
HcCommandStatus	USB2m_HOST_HcCommandStatus	0000_0000h	0008h	32
HcInterruptStatus	USB2m_HOST_HcInterruptStatus	0000_0000h	000Ch	32
HcInterruptEnable	USB2m_HOST_HcInterruptEnable	0000_0000h	0010h	32
HcInterruptDisable	USB2m_HOST_HcInterruptDisable	0000_0000h	0014h	32
HcHCCA	USB2m_HOST_HcHCCA	0000_0000h	0018h	32
HcPeriodCurrentED	USB2m_HOST_HcPeriodCurrentED	0000_0000h	001Ch	32
HcControlHeadED	USB2m_HOST_HcControlHeadED	0000_0000h	0020h	32
HcControlCurrentE	USB2m_HOST_HcControlCurrentED	0000_0000h	0024h	32
HcBulkHeadED	USB2m_HOST_HcBulkHeadED	0000_0000h	0028h	32
HcBulkCurrentED	USB2m_HOST_HcBulkCurrentED	0000_0000h	002Ch	32
HcDoneHead	USB2m_HOST_HcDoneHead	0000_0000h	0030h	32
HcFmInterval	USB2m_HOST_HcFmInterval	0000_2EDFh	0034h	32
HcFmRemaining	USB2m_HOST_HcFmRemaining	0000_2EDFh	0038h	32
HcFmNumber	USB2m_HOST_HcFmNumber	0000_0000h	003Ch	32
HcPeriodicStart	USB2m_HOST_HcPeriodicStart	0000_0000h	0040h	32
HcLSThreshold	USB2m_HOST_HcLSThreshold	0000_0628h	0044h	32
HcRhDescriptorA	USB2m_HOST_HcRhDescriptorA	0F00_0901h	0048h	32
HcRhDescriptorB	USB2m_HOST_HcRhDescriptorB	0002_0000h	004Ch	32
HcRhStatus	USB2m_HOST_HcRhStatus	0000_0000h	0050h	32
HcRhPortStatus1	USB2m_HOST_HcRhPortStatus1	0000_0000h	0054h	32
Reserve	-	-	0058h to 00FFh	-
HCIVERSION / CAPLENGTH	USB2m_HOST_CAPL_VERSION	0110_0020h	0100h	32
HCSPARAMS	USB2m_HOST_HCSPARAMS	0000_1191h	0104h	32
HCCPARAMS	USB2m_HOST_HCCPARAMS	000E_0006h	0108h	32
HCSP_PORTROUTE	USB2m_HOST_HCSP_PORTROUTE	0000_0000h	010Ch	32
Reserve	-	-	0110h to 011Fh	-
USBCMD	USB2m_HOST_USBCMD	0008_0B00h	0120h	32
USBSTS	USB2m_HOST_USBSTS	0000_1000h	0124h	32
USBINTR	USB2m_HOST_USBINTR	0000_0000h	0128h	32
FRINDEX	USB2m_HOST_FRINDEX	0000_0000h	012Ch	32
CTRLDSSEGMENT	USB2m_HOST_CTRLDSSEGMENT	0000_0000h	0130h	32
PERIODICLISTBASE	USB2m_HOST_PERIODICLISTBASE	0000_0000h	0134h	32
ASYNCLISTADDR	USB2m_HOST_ASYNCLISTADDR	0000_0000h	0138h	32
Reserve	-	-	013Ch to 015Fh	-
CONFIGFLAG	USB2m_HOST_CONFIGFLAG	0000_0000h	0160h	32
PORTSC1	USB2m_HOST_PORTSC1	0000_2000h	0164h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	0168h to 01FFh	-
INT_ENABLE	USB2m_HOST_INT_ENABLE	0000_0000h	0200h	32
INT_STATUS	USB2m_HOST_INT_STATUS	0000_0000h	0204h	32
AHB_BUS_CTR	USB2m_HOST_AHB_BUS_CTR	0000_0000h	0208h	32
USBCTR	USB2m_HOST_USBCTR	0000_0002h	020Ch	32
Reserve	-	-	0210h to 0303h	-
Register Enable/Clock Gating Control	USB2m_HOST_REGEN_CG_CTRL	0000_0000h	0304h	32
Suspend Control	USB2m_HOST_SPD_CTRL	0000_0000h	0308h	32
Suspend/Resume Timer Setting	USB2m_HOST_SPD_RSM_TIMSET	01F4_03E8h	030Ch	32
Overcurrent Detection/Sleep Timer Setting	USB2m_HOST_OC_SLP_TIMSET	0C83_0D40h	0310h	32
SBRN/FLADJ/PORTWAKECAP	USB2m_HOST_SBRN_FLADJ_PW	0003_2020h	0314h	32
UTMI+PHY Control Register	USB2m_HOST_UTMI_CTRL	8000_0004h	0318h	32
Reserve	-	-	031Ch to 031Fh	-
PORT_LPM_CTRL1	USB2m_HOST_PORT_LPM_CTRL1	0000_0000h	0320h	32
Reserve	-	-	0324h to 07FFh	-
Common Control	USB2m_HOST_COMMCTRL	8000_0000h	0800h	32
OTG-BC Interrupt Status	USB2m_HOST_OBINTSTA	0000_0001h	0804h	32
OTG-BC Interrupt Enable	USB2m_HOST_OBINTEN	0000_0000h	0808h	32
VBUS Control	USB2m_HOST_VBCTRL	0000_0000h	080Ch	32
Line Control Port 1	USB2m_HOST_LINECTRL1	0000_0000h	0810h	32
Reserve	-	-	0814h to 081Fh	-
BC Control Port 1	USB2m_HOST_BCCTRL1	0000_0000h	0820h	32

6.5.2.2.2 OHCI Operational Register Descriptions

(1) HcRevision Register (USB2m_HOST_HcRevision)

Access Size : 32 bits

Address : <USB2m_host_base> + 0000h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	Revision[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
7 to 0	Revision[7:0]	0h	R	This field indicates the version of HCI specifications implemented in this host controller module. Because this module conforms to OHCI standard 1.0a, 10h is indicated.

(2) HcControl Register (USB2m_HOST_HcControl)

Access Size : 32 bits
Address : <USB2m_host_base> + 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	RWC	-	HCFS[1:0]		BLE	CLE	IE	PLE	CBSR[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9	RWC	0h	RW	Remote Wakeup Connected This bit indicates whether the host controller supports remote wakeup signaling. Software should set this bit to 1b in the initialization sequence if it is required to support remote wakeup. Note that this bit can be initialized at hardware reset only. 0b: The remote wakeup is not supported. 1b: The remote wakeup is supported
8	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
7, 6	HCFS[1:0]	0h	RW	Host Controller Functional State This field indicates the operating state of the host controller. 00b: USB Reset 01b: USB Resume 10b: USB Operational 11b: USB Suspend When the state is changed to USB Operational, the host controller module starts SOF transmission at 1 ms boundary. This field is basically controlled by software, but it can be controlled by the host controller in the state of USB Suspend only. If the host controller (in the state of USB Suspend) detects a Remote Wakeup signal from the connecting device, the state in this field is changed to USB Resume. Note that the reset value of this field differs between hardware reset and software reset. Hardware reset: 00b (USB Reset) Software reset: 11b (USB Suspend)
5	BLE	0h	RW	BulkList Enable This bit sets whether the bulk list processing is performed for the next frame. The setting value of this bit is enabled from the next frame. When you correct the bulk list, this bit must be 0b. 0b: The processing of the Bulk list is disabled. 1b: The processing of the Bulk list is enabled
4	CLE	0h	RW	Control List Enable This bit sets whether the control list processing is performed for the next frame. The setting value of this bit is enabled from the next frame. When you correct the control list, this bit must be 0b. 0b: The processing of the Control list is disabled. 1b: The processing of the Control list is enabled.

Bit	Bit Name	Initial Value	R/W	Description
3	IE	0h	RW	Isochronous Enable This bit sets whether the isochronous ED processing is performed. The setting value of this bit is enabled from the next frame. If the host controller module detects isochronous ED (F = 1) during the periodic list processing, it checks the bit and determines whether to perform isochronous ED processing. 1b: The processing of the isochronous ED is continued. 0b: The periodic list processing is stopped, and the bulk/control list processing is started. 0b: The processing of the isochronous ED is disabled. 1b: The processing of the isochronous ED is enabled.
2	PLE	0h	RW	Periodic List Enable This bit indicates whether the periodic list processing is performed for the next frame. The setting value of this bit is enabled from the next frame. The host controller module checks this bit before starting the periodic list processing. 0b: The processing of the periodic list is disabled. 1b: The processing of the periodic list is enabled.
1, 0	CBSR[1:0]	0h	RW	Control Bulk Service Ratio This field defines the service ratio of the control transfer and bulk transfer. When the periodic list is processed, the service ratio defined in this field is used for transfer. CBSR No. of Control EDs Over Bulk EDs Served 00b 1 : 1 01b 2 : 1 10b 3 : 1 11b 4 : 1

(3) HcCommandStatus Register (USB2m_HOST_HcCommandStatus)

Access Size : 32 bits
Address : <USB2m_host_base> + 0008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SOC[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	BLF	CLF	HCR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
17, 16	SOC[1:0]	0h	R	Scheduling Overrun Count This field counts the number of scheduling overrun. This field is initialized to 00b, and is counted up every time scheduling overrun is detected. After the field is incremented to 11b, it returns to 00b. Even if the SO (Scheduling Overrun) bit in the HcInterrupt Status register is set, this field is counted up when scheduling overrun is detected.
15 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
2	BLF	0h	RW	BulkList Filled This bit indicates whether TD exists in the bulk list. To add TD to ED in the bulk list, set this bit to 1b. The host controller checks this bit when it starts processing of the head ED in the bulk list. If this bit is set to 0b, the host controller does not start the list processing. If this bit is set to 1b, the host controller starts processing of the bulk list, and sets this bit to 0b. When the host controller finds TD in the bulk list, it sets this bit to 1b again, and continues processing of the bulk list. When the host controller finishes the list processing, this bit is set to 0b. However, if TD is not found in the bulk list, or if this bit is not set to 1b, this bit remains to be 0b, and the list processing stops. To rebuild the list and start the list processing, before you set the BLE bit of the HcControl register and start the list processing, you need to set this bit. 0b: TD does not exist in the bulk list. 1b: TD exists in the bulk list.
1	CLF	0h	RW	Control List Filled This bit indicates whether TD exists in the control list. To add TD to ED in the control list, set this bit to 1b. The host controller checks this bit when it starts processing of the head ED in the control list. If this bit is set to 0b, the host controller does not start the processing of the control list. If this bit is set to 1b, the host controller starts processing of the control list, and set this bit to 0b. When the host controller finds TD in the control list, it sets this be to 1b again, and continues the list processing. When the host controller finishes the list processing, this bit is set to 0b. However, if TD is not found in the control list, or if this bit is not set to 1b, this bit remains to be 0b, and the list processing stops. To rebuild the list and start the list processing, before you set the CLE bit of the HcControl register and start the list processing, you need to set this bit. 0b: TD does not exist in the control list. 1b: TD exists in the control list.
0	HCR	0h	W	Host Controller Reset This bit is used to start OHCI software reset for the host controller. When this bit is set to 1b, the operating status of the host controller is changed to USB Suspend regardless of the functional state of the host controller. Also, the most OHCI Operational registers and OHCI control circuits are initialized. When the software reset finishes, the host controller clears this bit to 0b.

(4) HcInterruptStatus Register (USB2m_HOST_HcInterruptStatus)

Access Size : 32 bits
Address : <USB2m_host_base> + 000Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	RHSC	FNO	UE	RD	SF	WDH	SO
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
6	RHSC	0h	RW	Root Hub Status Change Interrupt bit that indicates that the state of the HcRhStatus register or HcRhPortStatus1 register has changed. When the Root Hub status is changed, the host controller sets this bit to 1b. Writing 1b to this bit clears the interrupt. 0b: The status of root hub has not changed. 1b: The status of root hub has changed.
5	FNO	0h	RW	Frame Number Overflow Interrupt bit that indicates that the MSB in the Frame Number (FN) field of theHcFmNumber register has changed. When the MSB in the Frame Number field is changed from 0 to 1, or from 1 to 0, this bit is set after HccaFrameNumber is updated. Writing this bit to 1b clears the interrupt. 0b: The overflow of frame number has not occurred. 1b: The overflow of frame number has occurred.
4	UE	0h	RW	Unrecoverable Error Interrupt bit that indicates that a system error that is not related to the USB (for example, an error on the system bus) has been detected. Writing this bit to 1b clears the interrupt. 0b: The unrecoverable error has not occurred. 1b: The unrecoverable error has occurred.
3	RD	0h	RW	Resume Detected Interrupt bit that indicates that Resume has been detected. When the host controller detects the Resume signal (RemoteWakeup) from an USB device, it sets this bit to 1b. This bit is not set when the Resume signal is sent by setting the HCFS field to USB Resume. Writing this bit to 1b clears the interrupt. 0b: The host controller has not detected resume signaling (RemoteWakeup). 1b: The host controller has detected resume signaling (RemoteWakeup).
2	SF	0h	RW	Start of Frame Interrupt bit that indicates that Hcca Frame Number was updated when each frame started. The host controller sends an SOF packet and updates HccaFrameNumber at the same time, and sets this bit to 1b. Writing this bit to 1b clears the interrupt. 0b: The host controller has not started new frame. 1b: The host controller has started new frame.

Bit	Bit Name	Initial Value	R/W	Description
1	WDH	0h	RW	Writeback DoneHead Interrupt bit that indicates that the host controller has written the contents of HcDoneHead to HccDoneHead. The host controller sets this bit to 1b immediately after it updates HccaDoneHead. Note that HccaDoneHead is not updated until this bit is cleared. Writing this bit to 1b clears the interrupt. This bit must be cleared only after the contents of HccaDoneHead are saved. 0b: The write back to HccaDoneHead has not occurred. 1b: The write back to HccaDoneHead has occurred.
0	SO	0h	RW	Scheduling Overrun Interrupt bit that indicates that overrun of the USB schedule occurred. When USB scheduling overrun occurs, the host controller updates HccaFrameNumber, and sets this bit to 1b. When this bit is set, the SchedulingOverrunCount field of the HcCommandStatus register is also incremented. Writing this bit to 1b clears the interrupt. 0b: The scheduling overrun has not occurred. 1b: The scheduling overrun has occurred.

(5) HcInterruptEnable Register (USB2m_HOST_HcInterruptEnable)

Access Size : 32 bits
Address : <USB2m_host_base> + 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MIE	OCE	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	RHSCE	FNOE	UEE	RDE	SFE	WDHE	SOE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	MIE	0h	RW	Master Interrupt Enable This bit sets whether each Interrupt Enable setting (which is set in HcInterruptEnable [30:0]) is enabled. If this bit is set to 0b, all OHCI interrupts are masked. To clear this bit, set the MID bit (bit 31) of the HcInterruptDisable register to 1b. 0b: All interrupts are disabled. 1b: Interrupts that are set to 1b are enabled.
30	OCE	0h	RW	OC (Ownership Change) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes OC (OwnershipChange). To clear this bit, set the OCD bit (bit 30) of the HcInterruptDisable register to 1b. 0b: OC (Ownership Change) interrupt is disabled. 1b: OC (Ownership Change) interrupt is enabled.
29 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
6	RHSCE	0h	RW	RHSC (Root Hub Status Change) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes RHSC (Root Hub Status Change). To clear this bit, set the RHSCD bit (bit 6) of the HcInterruptDisable register to 1b. 0b: RHSC (Root Hub Status Change) interrupt is disabled. 1b: RHSC (Root Hub Status Change) interrupt is enabled.
5	FNOE	0h	RW	FNO (Frame Number Overflow) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes FNO (Frame Number Overflow). To clear this bit, set the FNOD bit (bit 5) of the HcInterruptDisable register to 1b. 0b: FNO (Frame Number Overflow) interrupt is disabled. 1b: FNO (Frame Number Overflow) interrupt is enabled.
4	UEE	0h	RW	UE (Unrecoverable Error) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes UE (Unrecoverable Error). To clear this bit, set the UED bit (bit 4) of the HcInterruptDisable register to 1b. 0b: UE (Unrecoverable Error) interrupt is disabled. 1b: UE (Unrecoverable Error) interrupt is enabled.
3	RDE	0h	RW	RD (Resume Detect) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes RD (Resume Detect). To clear this bit, set the RDD bit (bit 3) of the HcInterruptDisable register to 1b. 0b: RD (Resume Detect) interrupt is disabled. 1b: RD (Resume Detect) interrupt is enabled.
2	SFE	0h	RW	SF (Start of Frame) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes SF (Start of Frame). To clear this bit, set the SFD bit (bit 2) of the HcInterruptDisable register to 1b. 0b: SF (Start of Frame) interrupt is disabled. 1b: SF (Start of Frame) interrupt is enabled.
1	WDHE	0h	RW	WDH (Writeback DoneHead) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes WDH (Writeback DoneHead). To clear this bit, set the WDHDD bit (bit 1) of the HcInterruptDisable register to 1b. 0b: WDH (Writeback DoneHead) interrupt is disabled. 1b: WDH (Writeback DoneHead) interrupt is enabled.

Bit	Bit Name	Initial Value	R/W	Description
0	SOE	0h	RW	SO (Scheduling Overrun) Interrupt Enable bit. If this bit is set to 1b, the interrupt source becomes SO (Scheduling Overrun). To clear this bit, set the SOD bit (bit 0) of the HcInterruptDisable register to 1b. 0b: SO (Scheduling Overrun) interrupt is disabled. 1b: SO (Scheduling Overrun) interrupt is enabled.

(6) HcInterruptDisable Register (USB2m_HOST_HcInterruptDisable)

Access Size : 32 bits
Address : <USB2m_host_base> + 0014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MID	OCD	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	RHSCD	FNOD	UED	RDD	SFD	WDHD	SOD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW1	RW1	RW1	RW1	RW1	RW1	RW1

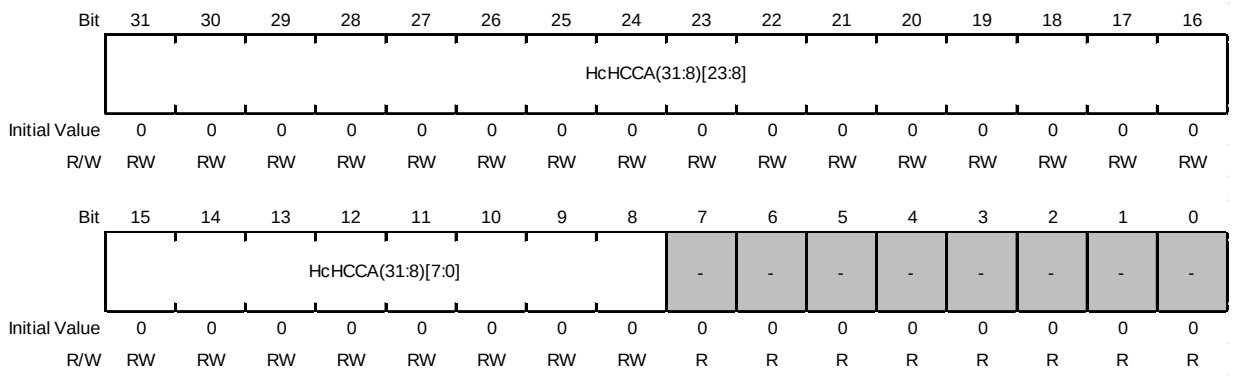
Bit	Bit Name	Initial Value	R/W	Description
31	MID	0h	RW1	Master Interrupt Disable This bit sets whether the Enable setting (for each interrupt) that is set by HcInterruptEnable[30:0] is disabled. If this bit is set to 1b, the MIE bit (bit 31) of the HcInterruptEnable register is cleared to 0b, and all OHCI interrupts are masked. Writing 0b to this bit is ignored.
30	OCD	0h	RW1	OC (Ownership Change) Interrupt Disable bit. If this bit is set to 1b, the OCE bit (bit 30) of the HcInterruptEnable register is cleared to 0b, and OC (Ownership Change) is excluded from the interrupt source. Writing 0b to this bit is ignored.
29 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
6	RHSCD	0h	RW1	RHSC (Root Hub Status Change) Interrupt Disable bit. If this bit is set to 1b, the RHSCE bit (bit 6) of the HcInterruptEnable register is cleared to 0b, and RHSC (Root Hub Status Change) is excluded from the interrupt source. Writing 0b to this bit is ignored.
5	FNOD	0h	RW1	FNO (Frame Number Overflow) Interrupt Disable bit. If this bit is set to 1b, the FNOE bit (bit 5) of the HcInterruptEnable register is cleared to 0b, and FNO (Frame Number Overflow) is excluded from the interrupt source. Writing 0b to this bit is ignored.
4	UED	0h	RW1	UE (Unrecoverable Error) Interrupt Disable bit. If this bit is set to 1b, the UEE bit (bit 4) of the HcInterruptEnable register is cleared to 0b, and UE (Unrecoverable Error) is excluded from the interrupt source. Writing 0b to this bit is ignored.
3	RDD	0h	RW1	RD (Resume Detected) Interrupt Disable bit. If this bit is set to 1b, the RDE bit (bit 3) of the HcInterruptEnable register is cleared to 0b, and RD (Resume Detected) is excluded from the interrupt source. Writing 0b to this bit is ignored.
2	SFD	0h	RW1	SF (Start of Frame) Interrupt Disable bit. If this bit is set to 1b, the SFE bit (bit 2) of the HcInterruptEnable register is cleared to 0b, and SF (Start of Frame) is excluded from the interrupt source. Writing 0b to this bit is ignored.
1	WDHD	0h	RW1	WDH (Writeback DoneHead) Interrupt Disable bit. If this bit is set to 1b, the WDHE bit (bit 1) of the HcInterruptEnable register is cleared to 0b, and WDH (Writeback DoneHead) is excluded form the interrupt source. Writing 0b to this bit is ignored.
0	SOD	0h	RW1	SO (Scheduling Overrun) Interrupt Disable bit. If this bit is set to 1b, the SOE bit (bit 1) of the HcInterruptEnable register is cleared to 0b, and SO (Scheduling Overrun) is excluded from the interrupt source. Writing 0b to this bit is ignored.

(7) HcHCCA Register (USB2m_HOST_HcHCCA)

Access Size : 32 bits

Address : <USB2m_host_base> + 0018h

Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 8	HcHCCA(31:8)[23:0]	0h	RW	This field sets the base address (of the RAM) that is assigned as the Host Controller Communication Area. This field must be set at initialization. The host controller requests (as HCCA) 256-byte area from the base address specified in this field.
7 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(8) HcPeriodicCurrentED Register (USB2m_HOST_HcPeriodCurrentED)

Access Size : 32 bits

Address : <USB2m_host_base> + 001Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PECD[27:12]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PECD[11:0]												-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	PECD[27:0]	0h	R	Period Current ED This pointer indicates the physical address of ED in the periodic list that is currently processed. The host controller updates this pointer when the list processing of the periodic ED finishes.
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(9) HcControlHeadED Register (USB2m_HOST_HcControlHeadED)

Access Size : 32 bits

Address : <USB2m_host_base> + 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CHED[27:12]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CHED[11:0]												-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	CHED[27:0]	0h	RW	Control Head ED This field specifies the physical address of the head ED of the control list. These bits must be set for control transfer before the CLE bit of the HcControl register is set. The host controller starts processing of the control list from the HcBulkHeadED pointer.
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(10) HcControlCurrentED Register (USB2m_HOST_HcControlCurrentED)

Access Size : 32 bits

Address : <USB2m_host_base> + 0024h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CCED[27:12]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCED[11:0]												-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	CCED[27:0]	0h	RW	Control Current ED This pointer indicates the physical address of the ED that is currently processed in the control list. After the current ED processing finishes, this pointer proceeds to the next ED. The host controller continues the list processing until the end of the frame. When the end of the control list is reached, the host controller checks the Control List Filled bit of the HcCommandStatus register. If the corresponding bit is set to 1b, the contents of the HcControlHeadED field are copied to the HcControlCurrentED field, and the Control List Filled bit is cleared. If the corresponding bit is set to 0b, nothing is performed. Update of this register is allowed only when the Control List Enable bit of the HcControl register is cleared. If the Control List Enable bit is set to 1b, the value of this register is only read. This register is initially set to 0h to indicate the end of the control list.
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(11) HcBulkHeadED Register (USB2m_HOST_HcBulkHeadED)

Access Size : 32 bits
Address : <USB2m_host_base> + 0028h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BHED[27:12]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BHED[11:0]												-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	BHED[27:0]	0h	RW	Bulk Head ED This field specifies the physical address of the head ED of the bulk list. This field must be set for bulk transfer before the BLE bit of the HcControl register is set. The host controller starts the processing of the control list from the HcBulkHeadED pointer.
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(12) HcBulkCurrentED Register (USB2m_HOST_HcBulkCurrentED)

Access Size : 32 bits

Address : <USB2m_host_base> + 002Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BCED[27:12]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BCED[11:0]												-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	BCED[27:0]	0h	RW	Bulk Current ED This pointer indicates the physical address of the ED that is currently processed in the bulk list. After the current ED processing finishes, this pointer proceeds to the next ED. The host controller continues the list processing until the end of the frame. When the end of the bulk list is reached, the host controller checks the Control List Filled bit of the HcCommandStatus register. If the corresponding bit is set to 1b, the contents of the HcBulkHeadED field are copied to the HcBulklCurrentED field, and the ControlListFilled bit is cleared. If the corresponding bit is set to 0b, nothing is performed. Update of this register is allowed only when the ControlListEnable bit of the HcControl register is cleared. If the ControlListEnable bit is set to 1b, the value of this register is only read. This register is initially set to 0h to indicate the end of the bulk list.
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(13) HcDoneHead Register (USB2m_HOST_HcDoneHead)

Access Size : 32 bits
Address : <USB2m_host_base> + 0030h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DH[27:12]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DH[11:0]												-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	DH[27:0]	0h	R	Done Head This field indicates the physical address of the HcDoneHead of the host controller. The physical address of the TD that lately finished and is added to the Done queue. After the TD processing finishes, the host controller writes the contents of the HcDoneHead to the NextTD field of the TD. At the same time, the host controller overwrites the contents of HcDoneHead with the TD address. After the host controller writes the contents of this register into HCCA, it sets 0b to this register. Then, the Writeback DoneHead bit of the HcInterruptStatus register is set to 1b.
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(14) HcFmInterval Register (USB2m_HOST_HcFmInterval)

Access Size : 32 bits

Address : <USB2m_host_base> + 0034h

Initial Value : 0000_2EDFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIT	FSMPS[14:0]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	FI[13:0]													
Initial Value	0	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	FIT	0h	RW	Frame Interval Toggle This bit is used to synchronize the frame setting value with the hardware (host controller). When the FI field is updated, the toggle value is written to this bit. When the FI field value is applied to the FR field of the HcFmRemaining register, the value of this bit is also applied to the FRT bit of the HcFmRemaining register. By checking the toggle value in the FRT bit of the cFmRemaining register, the software can check whether the value in the FI field has been applied to the FR field of the HcFmRemaining register.
30 to 16	FSMPS[14:0]	0h	RW	FS Largest Data Packet This field sets the maximum amount of data (bits) that the host controller can send and receive without the fear of schedule overrun. The host controller compares the current frame position and the setting value in this field to determine the length (of the frame) that is ready to be transferred. This value differs depending on the capacity of the system bus and other reasons, estimate the value and set it to this field. Note: The maximum setting value for this field is 2778h. Do not set any value that is larger than 2778h.
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 0	FI[13:0]	2EDFh	RW	Frame Interval This field is used to set the length of the frame (bit time) used for Full Speed. Set the value of this field to “2EDFh” so that 1 frame (= 1 ms) of USB standard is satisfied.

(15) HcFmRemaining Register (USB2m_HOST_HcFmRemaining)

Access Size : 32 bits
Address : <USB2m_host_base> + 0038h
Initial Value : 0000_2EDFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FRT	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	FR[13:0]													
Initial Value	0	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	FRT	0h	R	Frame Remaining Toggle This bit is used to synchronize the frame setting value with the hardware (host controller). When the FR (Frame Remaining) field is set to 0000h, the host controller copies the FI (Frame Interval) field value to the FR field, and copies the FIT (Frame Interval Toggle) bit value to this bit. This bit can be used to check that the FI field of the HcFmInterval register has been correctly copied to the FR field.
30 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
13 to 0	FR[13:0]	2EDFh	R	Frame Remaining This field indicates the current frame value for 14-bit down counter. The value in this field counts down as time passes. When the value becomes 0000h, the value of FI (Frame Interval) of the HcFmInterval register is loaded. When the state of the host controller is changed to the USB Operational state, the host controller reloads the value in the FI (Frame Interval) field of the HcFmInterval register, and the new value is used from the next SOF.

(16) HcFmNumber Register (USB2m_HOST_HcFmNumber)

Access Size : 32 bits

Address : <USB2m_host_base> + 003Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FN[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	FN[15:0]	0h	R	Frame Number This field indicates the number of passed frames. When the HcFmRemaining register is reloaded, this field is incremented. If this field is reached to FFFFh, the value is rolled over to 0000h. When the state of the host controller is changed to the USB Operational state, this field is automatically incremented. After the host controller increments the frame number at the frame boundary and sends SOF, the contents of this field are written to HCCA. This is performed before the host controller reads the first ED of the frame. After writing to HCCA, the host controller sets the SF bit of the HcInterruptStatus register.

(17) HcPeriodicStart Register (USB2m_HOST_HcPeriodicStart)

Access Size : 32 bits

Address : <USB2m_host_base> + 0040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PS[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
13 to 0	PS[13:0]	0h	RW	Periodic Start This field indicates the time the host controller starts the periodic list processing in the frame. Estimate an appropriate value, and set the value to this field at the initial setting of the host controller. OHCI standard recommends that you set this setting value to about 90% of the FI field value of the HcFmInterval register. The recommended value is 2A2Fh. When the value in the FR field of the HcFmRemaining register reaches the value set to this field, the periodic list processing is given priority over the control/bulk list processing. Therefore, after the currently running control or bulk transfer finishes, the host controller starts the Interrupt list processing.

(18) HcLSThreshold Register (USB2m_HOST_HcLSThreshold)

Access Size : 32 bits

Address : <USB2m_host_base> + 0044h

Initial Value : 0000_0628h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	LST[11:0]											
Initial Value	0	0	0	0	0	1	1	0	0	0	1	0	1	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
11 to 0	LST[11:0]	628h	RW	LS Threshold This field indicates the threshold value of whether transfer is available for the remaining time of the LS transfer frame. If the value of the FR field of the FmRemaining register is larger than the value set to this field, the host controller can start LS transfer.

(19) HcRhDescriptorA Register (USB2m_HOST_HcRhDescriptorA)

Access Size : 32 bits
Address : <USB2m_host_base> + 0048h
Initial Value : 0F00_0901h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	POTPGT[7:0]								-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	NOCP	OCPM	DT	NPS	PSM	NDP[7:0]							
Initial Value	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1
R/W	R	R	R	RW	RW	R	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	POTPGT[7:0]	0Fh	RW	PowerOn To PowerGood Time This field indicates the time for waiting until software can access the root hub port after the power is supplied to the port. The unit of time is 2 ms. Therefore, the wait time is POTPGT x 2 ms.
23 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
12	NOCP	0h	RW	No OverCurrent Protection This bit sets whether the overcurrent function of the root hub is supported. If this bit is set to 0b, the overcurrent state is reported depending on the setting of the OCPM bit. 0b: The overcurrent state is supported. 1b: The overcurrent state is not supported. If you do not require the overcurrent function, set this bit to 1b once the module is released from module standby. For details, see **6.5.2.9.1 Host/Peripheral Common Setting Sequence**.
11	OCPM	1h	RW	OverCurrent Protection Mode This bit sets how to report the overcurrent state of the root hub. If this bit is reset, this bit must indicate the same mode as the PSM (Power Switching Mode) bit. This bit is valid only when the NOCP (No OverCurrent Protection) bit is cleared (0b). 0b: The overcurrent state is collectively reported for all ports. 1b: The overcurrent state is reported for each port.
10	DT	0h	R	Device Type This bit indicates that the root hub is not a composite device. This bit is always 0b because the root hub is not allowed to be a composite device.
9	NPS	0h	RW	No Power Switching This bit sets how to control the port power. If this bit is set to 0b, the PSM bit is used to set whether the power control is collectively performed for all ports or is performed for each port. 0b: The port power can be switched between on and off. 1b: The power is always on while the host controller is running.
8	PSM	1h	RW	Power Switching Mode This bit sets how to control the power switch for each port of the root hub. This bit is valid only when the NPS bit is 0b. 0b: The power of all ports is collectively controlled. 1b: The power of ports is controlled for each port. If the PPCM (Port Power Control Mask) bit of the HcRhDescriptorB register is set, each port responds only to the Set/ClearPortPower command. If the PPCM bit is cleared, each port is controlled by the Set/ClearGlobalPower command.
7 to 0	NDP[7:0]	1h	R	Number Down stream Ports This field indicates the number of downstream ports supported by the root hub.

(20) HcRhDescriptorB Register (USB2m_HOST_HcRhDescriptorB)

Access Size : 32 bits

Address : <USB2m_host_base> + 004Ch

Initial Value : 0002_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PPCM	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DR	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
17	PPCM	1h	RW	Port Power Control Mask This bit sets the port power control command when the PSM (Power Switching Mode) bit of the HcRhDescriptorA register is set. If this bit is 0b, the Global Power Control command (Set/ClearGlobalPower) is used for control. If this bit is 1b, the Port Power Control command (Set/ClearPortPower) is used for control. If the PSM bit is 0b, this bit is ignored.
16 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
1	DR	0h	RW	Device Removable This bit indicates whether each port of the root hub is removable. If this bit is 0b, the connected device is removable. If this bit is 1b, the connected device is not removable.
0	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(21) HcRhStatus Register (USB2m_HOST_HcRhStatus)

Access Size : 32 bits

Address : <USB2m_host_base> + 0050h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRWE	-	-	-	-	-	-	-	-	-	-	-	-	-	OCIC	LPSC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW1	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRWE	-	-	-	-	-	-	-	-	-	-	-	-	-	OCI	LPS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CRWE	0h	W	Clear RemoteWakeup Enable This bit is used to clear the DRWE bit. If this bit is set to 1b, DRWE (Device RemoteWakeup Enable) bit can be cleared. Writing 0b to this bit has no effect.
30 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
17	OCIC	0h	RW1	OverCurrent Indicator Change This bit is used to report the change in the OCI bit. If there is any change in the OCI bit, the host controller sets this bit to 1b. If 1b is written to this bit while this bit is set to 1b, this bit can be cleared. Writing 0b to this bit has no effect. 0b: There is no change in the Overcurrent state. 1b: There is a change in the Overcurrent state.
16	LPSC	0h	RW	The meaning of this bit differs depending on whether the operation is read or write. [Read] Local Power Status Change This bit is always read as 0b because the Local Power Status is not supported. [Write] Set Global Power If this bit is set to 1b, the power to the ports is turned on. The ports whose power is turned on are determined by the settings of the PSM (Power Switching Mode) bit and the PPCM (Port Power Control Mask) bit of the HcRhDescriptorA register. Writing 0b to this bit has no effect.
15	DRWE	0h	RW	[Read] Device RemoteWakeup Enable This bit sets whether the RemoteWakeup event includes the CSC (Connect Status Change) bit. If this bit is set to 1b, the CSC bit of the HcRhPortStatus register becomes valid as the Resume event. If the CSC bit is changed to 1b, the state is changed from USB Suspend to USB Resume, and the Resume detection interrupt occurs. 0b: Connect Status Change is not the source of RemoteWakeup. 1b: Connect Status Change is the source of RemoteWakeup. [Write] Set RemoteWakeup Enable This bit is used to set the DRWE bit. Writing 0b to this bit has no effect.
14 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
1	OCI	0h	R	Over Current Indicator This bit is used to report the overcurrent state in the global overcurrent detection mode (OCPM bit = 0b). This bit always indicates 0b when overcurrent for each port is reported (when OPCM bit = 1b). 0b: The port state is normal. 1b: The port is in the overcurrent state.

Bit	Bit Name	Initial Value	R/W	Description
0	LPS	0h	RW	The meaning of this bit differs depending on whether the operation is read or write. [Read] Local Power Status This bit is always read as 0b because the Local Power Status is not supported. [Write] Clear Global Power If this bit is set to 1b, the power to the ports is turned off. The ports whose power is turned off are determined by the settings of the PSM (Power Switching Mode) bit and PPCM (Port Power Control Mask) bit of the HcRhDescriptorA register. Writing 0b to this bit has no effect.

Note: ● Supplementary note on LPSC bit

Value written to this bit	PSM	PPCM[N]	Description
0	—	—	Setting ignored
1	0	—	1b is set to the PPS bit.
	1	0	1b is set to the PPS bit.
		1	Setting ignored

● Supplementary note on LPS bit

Value written to this bit	PSM	PPCM[N]	Description
0	—	—	Setting ignored
1	0	—	The PPS bit is cleared to 0b.
	1	0	The PPS bit is cleared to 0b.
		1	Setting ignored

(22) HcRhPortStatus[1:NDP] Register (USB2m_HOST_HcRhPortStatus1)

Access Size : 32 bits

Address : <USB2m_host_base> + 0054h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	PRSC	OCIC	PSSC	PESC	CSC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW1	RW1	RW1	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	LSDA	PPS	-	-	-	PRS	POCI	PSS	PES	CCS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
20	PRSC	0h	RW1	Port Reset Status Change This bit indicates that port reset (bus reset) has finished. The host controller sets this bit when 10 ms port reset (bus reset) finishes. 0b: Port reset has not finished, or the PRS (Port Reset Status) bit is not changed. 1b: Port reset has finished.
19	OCIC	0h	RW1	Port OverCurrent Indicator Change This bit is set when the overcurrent state of the port is detected. This bit is set when the host controller changed the POCI bit value. This bit is valid only under the setting that the overcurrent state is reported for each port (OCPM bit = 1b). If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. 0b: The overcurrent state has not changed. 1b: The overcurrent state has changed (POCI bit changed).
18	PSSC	0h	RW1	Port Suspend Status Change This bit indicates that the Resume sequence finished. This sequence includes 20 ms of the Resume signal, LS EOP, and 3 ms of resynchronization delay. If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. If the PRSC (Port Reset Status Change) bit is set, this bit is cleared. 0b: The Resume sequence has not finished. 1b: The Resume sequence has finished.
17	PESC	0h	RW1	Port Enable Status Change This bit indicates that the PES (Port Enable Status) bit was changed. If a hardware event clears the PES bit, this bit is set to 1b. If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. 0b: The PES (Port Enable Status) bit has not changed. 1b: The PES (Port Enable Status) bit has changed.
16	CSC	0h	RW1	Connect Status Change This bit indicates that the CCS (Current Connect Status) bit was changed. If the Connect/Disconnect event occurs, this bit is set to 1b. If this bit is set to 1b, this bit is cleared. Writing 0b to this bit has no effect. If a request (Port Reset/Port Enable/Port Suspend) is received during the Disconnect status, this bit is set for reevaluation of device connection confirmation. 0b: The CCS (Current Connect Status) bit has not changed. 1b: The CCS (Current Connect Status) bit has changed.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
9	LSDA	0h	RW	The meaning of this bit differs depending on whether the operation is read or write. [Read] Low Speed Device Attached This bit indicates the speed of the device connected to the port. This bit is valid only when the CCS (Current Connect Status) bit is set. 0b: Full Speed device is connected. 1b: Low Speed device is connected. [Write] Clear Port Power This bit is used to turn off the power of the port when the port is power controlled. Writing 1b to this bit turns off the port power. Writing 0b to this bit has no effect.
8	PPS	0h	RW	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Power Status This bit indicates the power status of the port. This bit is cleared when the overcurrent is detected. 0b: Port power is off. 1b: Port power is on. [Write] Set Port Power This bit is used to turn on the power of the port when the port is power controlled. Writing 1b to this bit turns on the port power. Writing 0b has no effect.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
4	PRS	0h	RW	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Reset Status This bit indicates the port reset (bus reset) status. When 10 ms of port reset finishes, the PRSC (Port Reset Status Change) bit is set and this bit is cleared. When the CCS bit is cleared (when no device is connected), this bit cannot be set. 0b: Not during port reset 1b: During port reset [Write] Set Port Reset This bit is used to issue a port reset (bus reset) to the downstream port. Writing 1b to this bit starts 10 ms of port reset. Writing 0b to this bit has no effect. If CCS is cleared, this bit cannot be set. Instead, CSC is set to 1b. This is performed to report to ports to which no device is connected, that port reset was performed.
3	POCI	0h	RW	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port OverCurrent Indicator This bit indicates that the downstream port is in the overcurrent status. This bit is valid only when the setting is specified that the overcurrent status is reported for each port (OCPM bit = 1b). On the other hand, if the setting is specified that the overcurrent status is reported for all ports, this bit is always read as 0b. 0b: The port is in normal status. 1b: The port is in overcurrent status. [Write] Clear Suspend Status This bit is used to finish the Suspend status and start the Resume sequence. Writing 1b to this bit starts the Resume sequence. Writing 0b has no effect. The Resume sequence starts only when PSS (Port Suspend Status) is set.
2	PSS	0h	RW	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Suspend Status This bit indicates that the port status is in Suspend or Resume sequence. 0b: The port is not in Suspend status. 1b: The port is in Suspend status. When the CCS bit is cleared (when no device is connected), this bit cannot be set. This bit is cleared under the following conditions: When the Resume sequence finished, and the PSSC (Port Suspend Status Change) bit is set When the port reset finished, and the PRSC (Port Reset Status Change) bit is set When the host controller is in the USB Resume state [Write] Set Port Suspend This bit is used to change the port status to Suspend. Writing 1b to this bit changes the status to Suspend. Writing 0b to this bit has no effect. If CCS is cleared, this bit cannot be set. Instead, CSC is set to 1b. This is performed to report to the ports to which no device is connected that the Suspend command was issued.

Bit	Bit Name	Initial Value	R/W	Description
1	PES	0h	RW	The meaning of this bit differs depending on whether the operation is read or write. [Read] Port Enable Status This bit indicates whether the port status is Enable or Disable. If the host controller detects a bus error (for example, overcurrent state, disconnect, port power off, and babble error), it clears this bit. Then, the PESC (Port Enable Status Change) bit is set. When the CCS bit is cleared (when no device is connected), this bit cannot be set. This bit is set “when port reset finished and the PRSC bit is set” or “when the port status becomes Suspend and the PRSC bit is set”. 0b: The port status is Disable. 1b: The port status is Enable. [Write] Set Port Enable This bit is used to set the PES bit. Writing 1b to this bit changes the port status to Enable. Writing 0b to this bit has no effect. If CCS is cleared, this bit cannot be set. Instead, CSC is set to 1b. This is performed to report that the status of the ports to which no device is connected was tried to be changed to Enable.
0	CCS	0h	RW	The meaning of this bit differs depending on whether the operation is read or write. [Read] Current Connect Status The current connection status of the downstream port is applied to this bit. 0b: No device is connected. 1b: A device is connected. [Write] Clear Port Enable This bit is used to clear the PES (Port Enable Status) bit. Writing 1b to this bit changes the port status to Disable. Writing 0b to this bit has no effect.

6.5.2.2.3 EHCI Controller Capability Register Descriptions

(1) HCIVERSION/CAPLENGTH Register (USB2m_HOST_CAPL_VERSION)

Access Size : 32 bits

Address : <USB2m_host_base> + 0100h

Initial Value : 0110_0020h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Interface Version Number[15:0]															
Initial Value	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	Capability Registers Length[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Interface Version Number[15:0]	110h	R	This field indicates the EHCI version supported by the host controller. 0110h is indicated because this host controller supports EHCI Rev1.1.
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
7 to 0	Capability Registers Length[7:0]	20h	R	This field is used as an offset that is added to the base address to find the start address of the EHCI Operational register. 20h is indicated because the EHCI Operation register of this module starts from 20h.

(2) HCSPARAMS Register (USB2m_HOST_HCSPARAMS)

Access Size : 32 bits

Address : <USB2m_host_base> + 0104h

Initial Value : 0000_1191h

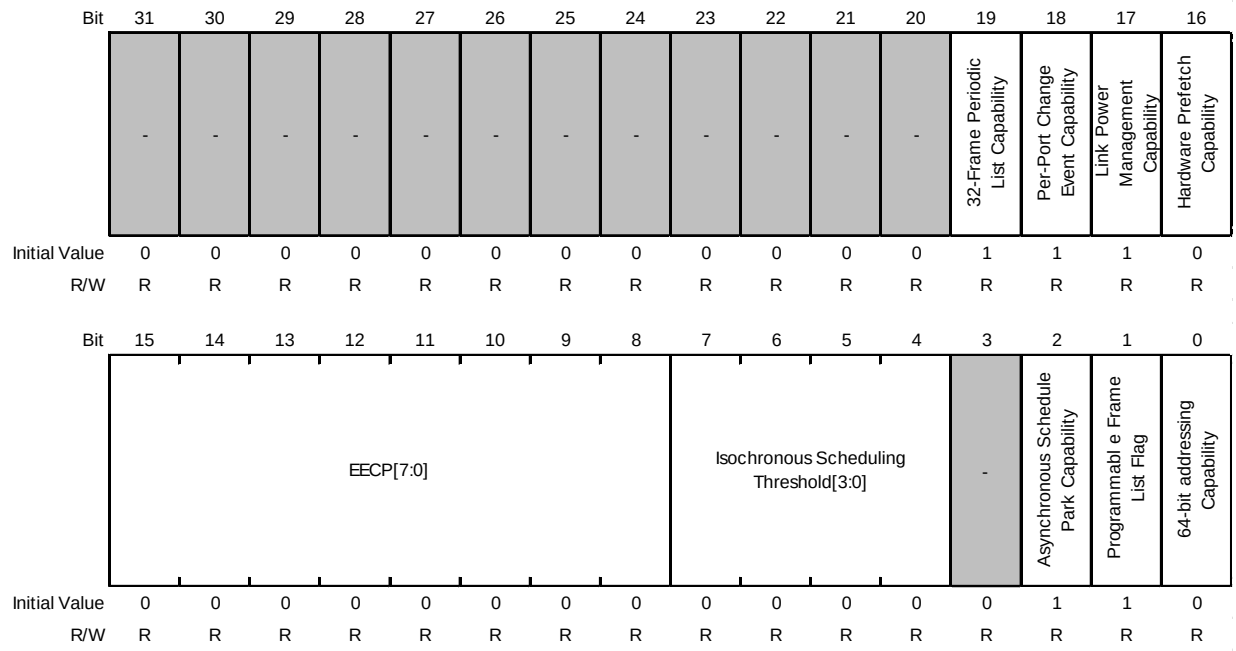
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	Debug Port Number[3:0]				-	-	-	P_INDICATOR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	N_CC[3:0]				N_PCC[3:0]				Port Routing Rules	-	-	PPC	N_PORTS[3:0]			
Initial Value	0	0	0	1	0	0	0	1	1	0	0	1	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
23 to 20	Debug Port Number[3:0]	0h	R	This field indicates that the host controller port is a debug port. 0000b is indicated because this module does not have a debug port.
19 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
16	P_INDICATOR	0h	R	Port Indicators This bit indicates whether the host controller supports port indicator control. 0b is indicated because this module does not support port indicator control.
15 to 12	N_CC[3:0]	1h	R	Number of Companion Controller This field indicates the number of OHCI host controllers implemented in this module. 1h is indicated because this module has one OHCI host controller.
11 to 8	N_PCC[3:0]	1h	R	Number of Ports per Companion Controller This field indicates the number of ports supported by an OHCI host controller. The setting value of the Port_no field of the PCI Configuration EXT1 register is applied to this field.
7	Port Routing Rules	1h	R	This bit indicates how individual ports are mapped to the OHCI host controller. This bit indicates 1b because, in this module, the contents of the HCSP_PORTROUTE register show the mapping method.
6, 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
4	PPC	1h	R	Port Power Control This bit indicates how the port power of this module is controlled. 0b: The port power is always on. 1b: The PP bit of the PORTSC register controls the port power.
3 to 0	N_PORTS[3:0]	1h	R	This field indicates the number of physical downstream ports used by this module. 0001b: 1 Port

(3) HCCPARAMS Register (USB2m_HOST_HCCPARAMS)

Access Size : 32 bits
Address : <USB2m_host_base> + 0108h
Initial Value : 000E_0006h



Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
19	32-Frame Periodic List Capability	1h	R	This bit indicates 1b because this module supports 32-Frame Periodic List defined in EHCI V1.1. This means that the Frame List Size field of the USBCMD register is set to 11b, so this module supports 32-Frame Periodic List.
18	Per-Port Change Event Capability	1h	R	This bit indicates 1b because this module supports event detection function for ports that are defined in EHCI V1.1. This means that this module supports event detection function for ports, and is associated with the Pre-Port Change Event Enable field of the USBCMD register, Port-1 Change Detect field of the USBSTS register, and Port-1 Change Interrupt Enable field of the USBINT register. If this bit is 0b, the corresponding fields of the above registers are treated as Reserved.
17	Link Power Management Capability	1h	R	This bit indicates 1b because this module supports LPM (Link Power Management) defined in EHCI V1.1. This means that this module supports LPM L1 state, and is controlled by the Suspend using L1 bit, Suspend Status bit, and Device Address field of the PORTSC register. If this bit is 0b, the corresponding bits and field of the above PORTC register are treated as Reserved.
16	Hardware Prefetch Capability	0h	R	This bit indicates 0b because this module does not support the hardware prefetch function defined in EHCI V1.1.
15 to 8	EECP[7:0]	0h	R	This field indicates the offset address of the EHCI Extend Capabilities Registers. This field indicates 00h because this module does not use EHCI Extend Capabilities Registers.
7 to 4	Isochronous Scheduling Threshold[3:0]	0h	R	This field indicates 0h because this module does not support caches with isochronous data structure for the entire frame.
3	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
2	Asynchronous Schedule Park Capability	1h	R	This bit indicates whether the Park mode is supported for High Speed QH (Queue Head) in an asynchronous schedule. This bit indicates 1b because this module supports the above function.
1	Programmabl e Frame List Flag	1h	R	This bit indicates the setting for the available frame list size. This bit indicates 1b for this module. If this bit is set to 1b, bit [3:2] (Frame List Size) of the USBCMD register can be used to set the available frame list size, and the frame list size smaller than 4 Kbyte is configurable.

Bit	Bit Name	Initial Value	R/W	Description
0	64-bit addressing Capability	0h	R	This bit indicates which type of memory pointer the data structure uses (32 bit address memory pointer or 64 bit address memory pointer). This bit indicates 0b for this module because this module has the data structure that uses 32 bit address memory pointer. 64 bit address is not supported.

(4) HCSP-PORTROUTE Register (USB2m_HOST_HCSP_PORTROUTE)

Access Size : 32 bits

Address : <USB2m_host_base> + 010Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Companion Port Route[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Companion Port Route[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Companion Port Route[31:0]	0h	R	This field indicates the ports controlled by the OHCI host controller. This field indicates 0b for this module because this module has one OHCI host controller.

6.5.2.2.4 HCI Operational Register Descriptions

(1) USBCMD Register (USB2m_HOST_USBCMD)

Access Size : 32 bits

Address : <USB2m_host_base> + 0120h

Initial Value : 0008_0B00h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	Host-Initiated Resume Duration[3:0]				Interrupt Threshold Control[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Per-Port Change Events Enable	-	-	-	Asynchronous Schedule Park Mode Enable	-	Asynchronous Schedule Park Mode Count[1:0]		-	Interrupt on Async Advance Doorbell	Asynchronous Schedule Enable	Periodic Schedule Enable	Frame List Size[1:0]		HCRESET	RS
Initial Value	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	RW	R	RW	RW	R	RW	RW	RW	RW	RW	W	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
27 to 24	Host-Initiated Resume Duration[3:0]	0h	RW	This field indicates the minimum time of the K-state drive while the host controller resumes from the LPM (L1) state. The value in this field is sent to the connected device that has the LPM function via the HIDR field in the bmAttributes field of the LPM token. The encoded value of this field is defined as the name of the HIRD field in the LPM Token. Specifically, 0h means 50 µs, and if the value is incremented by 1, 75 µs is incremented. For example, 1h means 125 µs, and 4h means 1175 µs.
23 to 16	Interrupt Threshold Control[7:0]	8h	RW(R)	This field indicates the maximum rate until the host controller issues an interrupt. Note that the setting values other than the following values are not guaranteed. Value Maximum Interrupt Interval 00h Reserved 01h 1 micro-frame 02h 2 micro-frames 04h 4 micro-frames 08h 8 micro-frames (default, equals to 1 ms) 10h 16 micro-frames (2 ms) 20h 32 micro-frames (4 ms) 40h 64 micro-frames (8 ms) Do not set 00h to this field while the Halted bit is 0b.
15	Per-Port Change Events Enable	0h	RW	This field is used to enable the event report function of the ports defined by the Port-1 Change Detect field of the USBSTS register and the Port-1 Change Detect Enable field of the USBINTR register. 0b: The event report function of the ports is disabled. 1b: The event report function of the ports is enabled.
14 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
11	Asynchronous Schedule Park Mode Enable	1h	RW	This bit enables or disables the Asynchronous Schedule Park mode. 0b: The Park mode is disabled. 1b: The Park mode is enabled.
10	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
9, 8	Asynchronous Schedule Park Mode Count[1:0]	3h	RW	This field sets the number of transactions that the host controller can serially execute for one QH (Queue Head) fetch in an asynchronous schedule. The valid value range is 1h to 3h. This field is valid when bit 11 (Asynchronous Schedule Park Mode Enable) is 1b. Do not set 0h to this field.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
6	Interrupt on Async Advance Doorbell	0h	RW	This bit is used as the doorbell. In an asynchronous schedule processing, if you want an interrupt to occur before proceeding to the next QH (Queue), set this bit to 1b. After a QH processing normally finishes, the host controller clears this bit to 0b, and sets bit 5 (Interrupt on Async Advance bit) of the USBSTS register to 1b. If bit 5 (Interrupt on Async Advance Enable bit) of the UBINTR register is set to 1b, an interrupt occurs at the next interrupt timing. If the asynchronous schedule is disabled, do not write 1b to this bit.
5	Asynchronous Schedule Enable	0h	RW	This bit sets whether the host controller proceeds to the asynchronous list processing or skip the processing. 0b: The asynchronous list processing is skipped. 1b: Use the ASYNCLISTADDR register to proceed the asynchronous list processing.
4	Periodic Schedule Enable	0h	RW	This bit sets whether the host controller proceeds or skips the periodic list processing. 0b: The periodic list processing is skipped. 1b: Use the PERIODICLISTBASE register to proceed the periodic list processing.
3, 2	Frame List Size[1:0]	0h	RW	This field specifies the frame list size. The setting value of this field determines the size of the Frame List Current index of the FRINDEX register. Value the number of frames in Frame List 00b 1024 frames (default) 01b 512 frames 10b 256 frames 11b 32 frames
1	HCRESET	0h	W	Host Controller Reset This bit is used to initialize the EHCI circuit of the host controller. If this bit is set to 1b, the host controller initializes the internal pipelines, counters, and state machines, and communications on the USB immediately stop. At this time, port reset is not issued to the downstream ports. Reset by this bit has no effect on the registers other than the EHCI Operational register. The EHCI Operational register is initialized, and the port owner returns to OHCI. Software must be reset to return the host controller to the operational state. When this reset processing finishes, the host controller sets this bit to 0b. Writing 0b to this bit cannot stop the reset processing. If the HCHalted bit of the USBSTS register is set to 0b, do not set 1b to this bit.
0	RS	0h	RW	Run/Stop This bit is used to run or stop the EHCI host controller. If this bit is set to 1b, the host controller starts operation. As long as this bit is set to 1b, the host controller continues running. If this bit is set to 0b, the host controller finishes the currently executing transaction and some other transactions, and then is changed to the Halt status. The HCHalted bit of the USBSTS register indicates that the host controller finished the transaction processing and entered into the stop status. If the host controller is in a status other than Halt (the HCHalted bit of the USBSTS register is 1b), do not write 1b to this bit.

(2) USBSTS Register (USB2m_HOST_USBSTS)

Access Size : 32 bits

Address : <USB2m_host_base> + 0124h

Initial Value : 0000_1000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Port-1Change Detect
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Asynchronous Schedule Status	Periodic Schedule Status	Reclamation	HCHalted	-	-	-	-	-	-	Interrupt on Async Advance	Host System Error	Frame List Rollover	Port Change Detect	USBERRINT	USBINT
Initial Value	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW1	RW1	RW1	RW1	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
16	Port-1Change Detect	0h	RW1	If this bit is set to 1b, it indicates that a change in the port status was detected. This bit is used only when the Per-Port Change Events Enable bit of the USBCMD register is set to 1b.
15	Asynchronous Schedule Status	0h	R	This bit indicates the current status of the asynchronous schedule. 0b: The asynchronous schedule is disabled. 1b: The asynchronous schedule is enabled. If this bit and bit 5 (Asynchronous Schedule Enable) of the USBCMD register have the same value, the asynchronous schedule is enabled (1b) or disabled (0b).
14	Periodic Schedule Status	0h	R	This bit indicates the current status of the periodic schedule. 0b: The periodic schedule is disabled. 1b: The periodic schedule is enabled. If this bit and bit 4 (Periodic Schedule Enable) of the USBCMD register have the same value, the periodic schedule is enabled (1b) or disabled (0b).
13	Reclamation	0h	R	This bit is used to detect an empty asynchronous schedule. If this bit is 1b, the asynchronous schedule is empty. After reset or when a QH (H = 1) is fetched, the host controller clears this bit to 0b. Also, when the host controller executes an asynchronous transaction or detects a start event, it sets this bit to 1b. If this bit is 0b and a QH (H = 1) is fetched, the host controller is entered into the Async Sched Sleeping mode.
12	HCHalted	1h	R	If the Run/Stop bit of the USBCMD register is 1b, this bit indicates 0b. If the software or host controller sets the Run/Stop bit to 0b, the host controller stops operation, and sets 1b to this bit. 0b: The EHCI host controller is running. 1b: The EHCI host controller is stopped.
11 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
5	Interrupt on Async Advance	0h	RW1	This bit indicates the Async Advance Interrupt status. After the host controller fetches the QH, it checks bit 6 (Interrupt on Async Advance Doorbell [IAAD] bit) of the USBCMD Register. If the IAAD bit is set to 1b, the host controller clears the IAAD bit after the QH processing normally finishes, and sets this bit. If bit 5 (Interrupt on Async Advance Enable bit) of the UBINTR register is set to 1b, an interrupt due to this source will occur at the next interrupt timing after 1b is set to this bit. If HCD writes 1b to this bit, this bit can be cleared. Writing 0b to this bit has no effect. 0b: Async Advance Interrupt not occurred. 1b: Async Advance Interrupt status is detected.

Bit	Bit Name	Initial Value	R/W	Description
4	Host System Error	0h	RW1	This bit is set to 1b if a serious error occurs in the host controller. If this error occurs, the host controller clears the Run/Stop bit of the USBCMD register to 0b so that the subsequent schedules are not executed. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: No system error occurred. 1b: A system error occurred.
3	Frame List Rollover	0h	RW1	If a frame list rollover occurs, the host controller sets this bit to 1b. Specifically, when the Frame Index field of the FRINDEX register is returned to 000h from the maximum value (rollover), the host controller sets this bit to 1b. The maximum value (the value at which rollover occurs) depends on the Frame List Size field of the USBCMD register. For example, if the Frame List Size is 1024 frame, rollover occurs every time FRIMDEX [13] toggles. If the Frame List Size is 512 frame, rollover occurs every time FRINDEX [12] toggles. If the Frame List Size is 256 frame, rollover occurs every time FRINDEX [11] toggles. Writing 1b to this bit can clears this bit. Writing 0b to this bit has no effect. 0b: The frame list is not returned to 000h. 1b: The frame list is returned to 000h (rollover occurred).
2	Port Change Detect	0h	RW1	This bit indicates that the port status has changed. Among the ports for which the Port Owner bit of the PORTSC1 register is set to 0b, if any port satisfies one of the following conditions, the host controller sets this bit to 1b: Connect or Disconnect status of a device is detected, and the Connect Status Change bit of the PORTSC1 register is changed from 0b to 1b. A change of the Enable status of the port is detected, and the Port Enable/Disable Change bit of the PORTSC1 register is changed from 0b to 1b. The overcurrent state is detected, and the Over-current Change bit of the PORTSC1 register is changed from 0b to 1b. J-K transition is detected on a port in the Suspend status, and the Force Port Resume bit of the PORTSC1 register is changed from 0b to 1b. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect.
1	USBERRINT	0h	RW1	USB Error Interrupt This bit indicates that a USB transaction finished with an error. When a USB transaction finishes with an error, the host controller sets this bit to 1b. If 1b is set to the IOC bit of qTD at which an error interrupt occurred, 1b is set to both of this bit and USBINT bit. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: USB transaction is normal. 1b: USB transaction finished with an error.
0	USBINT	0h	RW1	USB Interrupt This bit indicates that a USB transfer has finished. The host controller sets this bit to 1b if one of the following conditions is satisfied: A USB transfer has finished. A short packet is received. Even if a USB transfer finished with an error, if IOC (Interrupt On Complete) of the TD is set to 1b, this bit is set to 1b. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: A USB transfer has not finished. 1b: A USB transfer has finished.

(3) SBINTR Register (USB2m_HOST_USBINTR)

Access Size : 32 bits
Address : <USB2m_host_base> + 0128h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Port-1 Change EventEnable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	Interrupt on Async Advance Enable	Host System Error Enable	Frame List Rollover Enable	Port Change Detect Enable	USB Error Interrupt Enable	USB Interrupt Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
16	Port-1 Change EventEnable	0h	RW	If this bit is 1b and the Port Change Detect bit of the USBSTS register is set to 1b, the host controller issues an interrupt.
15 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
5	Interrupt on Async Advance Enable	0h	RW	This bit sets whether bit 5 (Interrupt on Async Advance [IAA] bit) of the USBSTS register is enabled or disabled. If the IAA bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the IAA bit).
4	Host System Error Enable	0h	RW	This bit sets whether bit 4 (Host System Error [HSE] bit) of the USBSTS register is enabled or disabled. If the HSE bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the HSE bit).
3	Frame List Rollover Enable	0h	RW	This bit sets whether the bit 3 (Frame List Rollover [FLR] bit) of the USBSTS register is enabled or disabled. If the FLR bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the FLR bit).
2	Port Change Detect Enable	0h	RW	This bit sets whether bit 2 (Port Change Detect [PCD] bit) of the USBSTS register is enabled or disabled. If the USBERRINT bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the PCD bit).
1	USB Error Interrupt Enable	0h	RW	This bit sets whether bit 1 (USBERRINT bit) of the USBSTS register is enabled or disabled. If the USBERRINT bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the USBERRINT bit).
0	USB Interrupt Enable	0h	RW	This bit sets whether bit 0 (IUSBINT bit) of the USBSTS register is enabled or disabled. If the USBINT bit is set to 1b while this bit is set to 1b, the host controller issues an interrupt at the next interrupt timing. 0b: Disabled. 1b: Enabled (an interrupt occurs via the USBINT bit).

(4) FRINDEX Register (USB2m_HOST_FRINDEX)

Access Size : 32 bits
Address : <USB2m_host_base> + 012Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Frame Index[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
13 to 0	Frame Index[13:0]	0h	RW	This field is used by the host controller to add an index to the periodic frame list. The value in this field is incremented at the end of a micro frame. Bit [N:3] of this field is used as the Frame List Current index. This means that, before the next index arrives, the current frame list is accessed 8 times. The value for N is determined, as follows, by the setting value of bit [3:2] (Frame List Size field) of the USBCMD register. Access this register only when the host controller is in stop status (bit 12 [HCHalted] = 1b). The setting value of this field is applied to the SOF frame number of the SOF token.

Note: ■ Supplementary Note on Frame Index Bit

Frame List Size	Number of Frames	
00b	1024	(N = 12)
01b	512	(N = 11)
10b	256	(N = 10)
11b	32	(N = 12)

(5) CTRLDSSEGMENT Register (USB2m_HOST_CTRLDSSEGMENT)

Access Size : 32 bits

Address : <USB2m_host_base> + 0130h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CTRLDSSEG MENT[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CTRLDSSEG MENT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CTRLDSSEG MENT[31:0]	0h	R	This register is not used because this module does not support 64-bit address method. Therefore, HCD must not access this register.

(6) PERIODICLISTBASE Register (USB2m_HOST_PERIODICLISTBASE)

Access Size : 32 bits

Address : <USB2m_host_base> + 0134h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Base Address[19:4]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Base Address[3:0]				-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

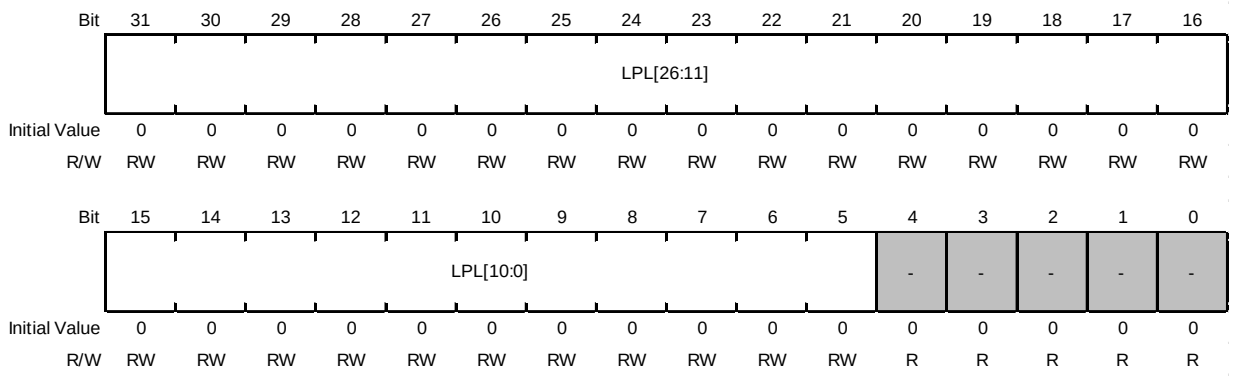
Bit	Bit Name	Initial Value	R/W	Description
31 to 12	Base Address[19:0]	0h	RW	This field indicates the head address of the periodic frame list on the system memory. The host controller loads the contents of this register before starting the list processing. The host controller determines the frame list to be processed by using this field and Frame Index of the FRINDEX register. Align the address of the periodic frame list by 4 Kbyte. Normal operation is not guaranteed if any of these bits are changed during operation.
11 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(7) ASYNCLISTADDR Register (USB2m_HOST_ASYNCLISTADDR)

Access Size : 32 bits

Address : <USB2m_host_base> + 0138h

Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 5	LPL[26:0]	0h	RW	Link Pointer Low This field indicates the address (on the system memory) of the Asynchronous Queue Head to be processed next time. Align the address of Asynchronous Queue Head by 32 byte.
4 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(8) CONFIGFLAG Register (USB2m_HOST_CONFIGFLAG)

Access Size : 32 bits

Address : <USB2m_host_base> + 0160h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
0	CF	0h	RW	Configuration Flag This bit controls which of OHCI or EHCI is routed by the port routing control circuit by default. At the end of the host controller configuration, this bit is set to 1b. 0b: The port routing control circuit routes each port to the OHCI host controller by default. 1b: The port routing control circuit routes each port to the EHCI host controller by default.

(9) PORTSC1Register (USB2m_HOST_PORTSC1)

Access Size : 32 bits
Address : <USB2m_host_base> + 0164h
Initial Value : 0000_2000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Device Address[6:0]							Suspend Status[1:0]		WKOC_E	WKDSCNNT_E	WKCNNT_E	Port Test Control[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Port Owner	PP	Line Status[1:0]		Suspend using L1	Port Reset	Suspend	Force Port Resume	Over-current Change	Over-current Active	Port Enable/ Disable Change	Port Enabled/ Disabled	Connect Status Change	Current Connect Status
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	RW	RW	RW	RW	RW1	R	RW1	RW	RW1	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	Device Address[6:0]	0h	RW	This field indicates the USB device address (7 bits) of the device connected to the down port. This address is used when the LPM Token is sent. If the value of this filed is 00h, it means that no device that needs to use this field has been connected.
24, 23	Suspend Status[1:0]	0h	R	This field indicates the response from the connected device to the LPM Token (L1 transition request). 00b: The device succeeded transition to the L1 state (ACK received from the device). 01b: The device has not changed to the L1 state (NYET received from the device). 10b: The device does not support the L1 state transition (STALL received from the device). 11b: Other response (for example, Timeout error) Change this field only when the Suspend bit is 0b.
22	WKOC_E	0h	RW	Wake on Overcurrent Enable By writing 1b to this bit, the overcurrent state can be detected as an EHCI Wakeup event. If bit 12 (PP [Port Power] bit) is 0b, this bit becomes 0b.
21	WKDSCNNT_E	0h	RW	Wake on Disconnect Enable By writing 1b to this bit, device disconnection can be detected as an EHCI Wakeup event. If bit 12 (PP [Port Power] bit) is 0b, this bit becomes 0b.
20	WKCNNT_E	0h	RW	Wake on Connect Enable By writing 1b to this bit, device connection can be detected as an EHCI Wakeup event. If bit 12 (PP [Port Power] bit) is 0b, this bit becomes 0b.
19 to 16	Port Test Control[3:0]	0h	RW	This field is controlled by the test mode. If the value of this field is other than 0000b, it indicates that this module is running in the test mode.
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
13	Port Owner	1h	RW	This bit indicates which of OHCI or EHCI has the port ownership. 0b: EHCI has the port ownership. 1b: OHCI has the port ownership. When bit 0 (Configure Flag bit) of the CONFIGFLA register is changed from 0b to 1b, this bit becomes 0b. If bit 0 (Configure Flag bit) of the CONFIGFLA register is 0b, this bit becomes 1b. If the connected device is not a High Speed device, this bit is set to 1b to transfer the port ownership to OHCI.

Bit	Bit Name	Initial Value	R/W	Description
12	PP	0h	RW	Port Power This bit controls power supply to the port. If this bit is 0b, power is not supplied to the port. Therefore, the port does not function, and does not recognize connection and disconnection. If overcurrent is detected while this bit is set to 1b, the host controller clears this bit to 0b, and the power supplied to the port is stopped. Note: As described later, if the PPC bit is 0b, this bit is fixed to 1b, so the power supplied to the port is not stopped. The function of this bit differs depending on the value of bit 4 (PPC [Port Power Control] bit) of the HCSPARAMS register.
11, 10	Line Status[1:0]	0h	R	This field indicates the logical level of D+/D- lines of the current USB bus. (bit 11: DP / bit 10: DM) This field is used to detect an LS device before starting a sequence for port reset and port enable. Therefore, this bit is valid only when bit 3 (Port Enable/Disable bit) is 0b and bit 0 (Current Connect Status bit) is 1b.
9	Suspend using L1	0h	RW	Suspend using L1 (LPM) control bit. If 1b is written to the Suspend bit (bit 7) while this bit is 1b, the state of this module is changed to the LPM state. Writing to this bit is possible only when the Suspend bit (bit 7) is 0b. 0b: Suspend using L2 1b: Suspend using L1 (LPM) If this bit is 1b and the Device Address field is other than 0000h, when the Suspend bit is set to 1b, the host controller generates an LPM Token to change the state to the L1 state. If this bit is 0b, the host controller operates as L2 Suspend.
8	Port Reset	0h	RW	This bit indicates the reset status of the port. 0b: The port is not being reset. 1b: The port is being reset. If 1b is written to this bit while this bit is 0b, the bus reset sequence defined in USB 2.0 standard starts. To finish the bus reset sequence, 0b must be written to this bit. Note that this bit must remain 1b long time enough to guarantee that the bus reset sequence defined in USB 2.0 standard will be complete. If bit 12 (HCHalted) of the USBSTS register is 1b, do not set this bit to 1b. If any of the PP (Port Power) bit, Port Owner bit, and Current Connect Status bit is in the following status, this bit becomes 0b. Note: Even if 1b is written to this bit, the bus reset sequence does not start. PP (Port Power) bit = 0b Port Owner bit = 1b Current Connect Status bit = 0b
7	Suspend	0h	RW	This bit indicates the Suspend control and status of the port. This bit and bit 2 (Port Enabled/Disabled bit) indicate the port status. (see below) To change the port state to L1 or L2 Suspend, set this bit to 1b. Whether the host controller supports L1 Suspend state or L2 Suspend state depends on the value in the Suspend Using L1 bit. In the Suspend state, data transfer to the downstream port is blocked by this port (except for port reset). If this bit is set to 1b during data transfer, blocking of transfer data does not occur until the current data transfer finishes. Writing 0b to this bit has no effect. This bit can be set to 1b only when all the following conditions are satisfied: “PP (Port Power) bit = 1b”, “Port Owner bit = 0b”, and “Current Connect Status bit = 1b”. If any of the following conditions is satisfied, the host controller clears this bit: “Resume is complete” is detected. The PR bit is set to 1b when PR (Port Reset) bit is 0b. The port owner is 1b (OHCI). The PP (Port Power) bit is set to 0b. The Port Enabled/Disabled bit is set to 0b.

Bit	Bit Name	Initial Value	R/W	Description
6	Force Port Resume	0h	RW	This bit indicates that the Resume state of the port is detected. 0b: Resume signal is not detected or output. 1b: Resume (K-state) is detected or output. When the port is in Suspend state, if the host controller detects transition of the state from J to K (if RemoteWakeup is detected from the connected device), it sets this bit to 1b. The host controller also sets the Port Change Detect bit or Port-1 Changes Detect bit of the USBSTS register to 1b. If this bit is set to 1b, the host controller does not set 1b to the Port Change Detect bit and Port-1 Changes Detect bit of the USBSTS register. While this bit is 1b, the Resume signal (FS K) is driven onto the USB bus. For L2 transition, this bit must be cleared to 0b after an appropriate time has passed. By writing 0b to this bit while this bit is 1b, the port status is recovered to be the HS Idle status. This bit remains 1b until the port is recovered. The host controller must finish transition to the HS Idle state within 2msec since this bit is cleared to 0b. On the other hand, for L1 transition, the host controller sends a Resume signal at necessary timing, and this bit is cleared to 0b at Resume recovery. Note that the software sets the length of the Resume signal driven by the host controller, by using the Host-Initiated Resume Duration field of the USBCMD register. If the Port Power (PP) bit is 0b, this bit becomes 0b.
5	Over-current Change	0h	RW1	This bit indicates that bit 4 (Over-current Active bit) has changed. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. 0b: Over-current Active bit has not changed. 1b: Over-current Active bit has changed.
4	Over-current Active	0h	R	This bit indicates the overcurrent status of the port. If the host controller detects overcurrent, it disables the port, and set this bit to 1b. After the overcurrent state is released, the host controller automatically clears this bit from 1b to 0b. 0b: The port is not in overcurrent state. 1b: The port is in overcurrent state.
3	Port Enable/ Disable Change	0h	RW1	This bit indicates that the host controller detected frame babble. If the Host Controller detects frame babble, it disables the port and sets this bit to 1b. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. If the Port Power (PP) bit is 0b, this bit becomes 0b. 0b: Frame babble has not occurred. 1b: Frame babble is detected.
2	Port Enabled/ Disabled	0h	RW1	This bit indicates the Enable/Disable status of the port. The host controller resets the port, and enables the port if the connected device is recognized as an HS device, and sets this bit to 1b. The software cannot set this bit to 1b. If the host controller detects disconnection of a device or other errors, it disables the port, and clears this bit to 0b. The port also becomes disabled when 0b is written to this bit. If the port is disabled, data transfer to the downstream port is blocked except for port reset. If Port Test Control [3:0] = 0101b (Test FORCE_ENABLE), the port becomes enabled, and this bit is set to 1b. If the Port Power (PP) bit is 0b, this bit becomes 0b. 0b: The port is disabled. 1b: The port is enabled.
1	Connect Status Change	0h	RW1	This bit indicates that bit 0 (Current Connect Status bit) has changed. Writing 1b to this bit can clear this bit. Writing 0b to this bit has no effect. If the Port Power (PP) bit is 0b, this bit becomes 0b. 0b: The Current Connect Status bit has no change. 1b: The Current Connect Status bit has changed.
0	Current Connect Status	0h	R	This bit indicates the connection status of the port. If the host controller detects connection of a device, it sets this bit to 1b. Also, if Port Test Control [3:0] = 0101b (Test FORCE_ENABLE), the Host Controller sets this bit to 1b even if no device is connected. On the other hand, if the host controller detects disconnection of a device, it sets this bit to 0b. If the Port Power (PP) bit is 0b, or the Port Owner (PO) bit is 0b, this bit becomes 0b. 0b: No device is connected to the port. 1b: A device is connected to the port.

Note: ■ Supplementary Note on Port Test Control (1/1)

Value	Test Mode
0000b	Test mode not enabled
0001b	Test J_STATE
0010b	Test K_STATE
0011b	Test SE0_NAK

■ Supplementary Note on Port Test Control (2/2)

Value	Test Mode
0100b	Test Packet
0101b	Test FORCE_ENABLE
Others	Reserved

■ Supplementary Note on PP

PPC	PP	Operation		
0b	1b	RO	This bit is fixed to 1b, and power is always supplied to the port.	
1b	0b/1b	R/W	Whether or not power is supplied to the port depends on the setting of this bit.	
			0b	Power is not supplied to the port.
			1b	Power is supplied to the port.

■ Supplementary Note on Line Status Bit

Bit 11	Bit 10	USB Bus Status	Description
0b	0b	SE0	The device is not an LS device.
			EHCI port reset is being executed.
1b	0b	J-state	The device is not an LS device.
			EHCI port reset is being executed.
0b	1b	K-state	An LS device was connected.
			The port ownership is transferred from EHCI to OHCI.
1b	1b	Undefined	The device is not an LS device.
			EHCI port reset is being executed.

■ Supplementary Note on Suspend Bit

Port Enabled	Suspend	Port Status
0b	—	Disable
1b	0b	Enable
	1b	Suspend

6.5.2.2.5 AHB Bridge Register Descriptions

(1) INT_ENABLE Register (USB2m_HOST_INT_ENABLE)

Access Size : 32 bits

Address : <USB2m_host_base> + 0200h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	WAKEON_INTEN	UCOM_INTEN	USBH_INTBEN	USBH_INTAEN	AHB_INTEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
4	WAKEON_INTEN	0h	RW	This bit enables or disables bit 4 (WAKEON_INT) of the INT_STATUS register. 0b: disable 1b: enable
3	UCOM_INTEN	0h	RW	This bit enables or disables bit 3 (UCOM_INT) of the INT_STATUS register. 0b: disable 1b: enable
2	USBH_INTBEN	0h	RW	This bit enables or disables bit 2 (USBH_INTB) of the INT_STATUS register. 0b: disable 1b: enable
1	USBH_INTAEN	0h	RW	This bit enables or disables bit 1 (USBH_INTA) of the INT_STATUS register. 0b: disable 1b: enable
0	AHB_INTEN	0h	RW	This bit enables or disables bit 0 (AHB_INT) of the INT_STATUS register. 0b: disable 1b: enable

(2) INT_STATUS Register (USB2m_HOST_INT_STATUS)

Access Size : 32 bits

Address : <USB2m_host_base> + 0204h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	WAKEON_INT	UCOM_INT	USBH_INTB	USBH_INTA	AHB_INT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW1	RW	RW	RW	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
4	WAKEON_INT	0h	RW1	This bit indicates the state of WAKEON interrupt from the HOST module. Writing 1b to this bit can clear this bit. 0b: No WAKEON interrupt 1b: WAKEON interrupt
3	UCOM_INT	0h	RW	This bit indicates the state of interrupt from the UCOM register. To clear the interrupt, use the UCOM2 Register. 0b: No UCOM register interrupt 1b: UCOM register interrupt
2	USBH_INTB	0h	RW	This bit indicates the state of EHCI interrupt. To clear the interrupt, use the USBSTS Register (of the EHCI Operational Register). 0b: No INTB interrupt 1b: INTB interrupt
1	USBH_INTA	0h	RW	This bit indicates the state of OHCI interrupt. To clear the interrupt, use the HcInterruptStatus Register (of the OHCI Operational Register). 0b: No INTA interrupt 1b: INTA interrupt
0	AHB_INT	0h	RW1	This bit indicates that a BUS Master error occurred. Writing 1b to this bit can clear this bit. 0b: No bus error occurred. 1b: A bus error occurred.

(3) AHB_BUS_CTR Register (USB2m_HOST_AHB_BUS_CTR)

Access Size : 32 bits
Address : <USB2m_host_base> + 0208h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT_TYPE[3:0]				-	-	-	PROT_MODE	-	-	ALIGN_ADDRESS[1:0]		-	-	MAX_BURST_LEN[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	R	RW	R	R	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
15 to 12	PROT_TYPE [3:0]	0h	RW	This field sets MHPROT [3:0] used when the BUS Master interface issues a transfer request. bit 15 0b: Cache disabled. 1b: Cache enabled. bit 14 0b: Buffer disabled. 1b: Buffer enabled. bit 13 0b: User access 1b: Privileged access bit 12 0b: Operation code 1b: Data
11 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
8	PROT_MODE	0h	RW	This bit selects the mode of MHPROT [3:0] used when the Master interface issues a transfer request. 0b: The value of PROT_TYPE is output as MHPROT [3:0]. 1b: When a DMA transfer is performed, MHPROT [3:0] is set to 0000b if the final burst is performed, or MHPROT [3:0] is set to the PROT_TYPE value if another burst transfer is performed.
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
5, 4	ALIGN_ADDRESS[1:0]	0h	RW	This field sets the address boundary used when the BUS Master interface issues a burst transfer. 00b: A burst transfer is issued so that not to exceed 1-Kbyte boundary. 01b: A burst transfer is issued so that not to exceed 64-byte boundary. 10b: A burst transfer is issued so that not to exceed 32-byte boundary. (The maximum burst length is INCR8. This is because, if INCR16 is used, 32-byte boundary is exceeded.) 11b: A burst transfer is issued so that not to exceed16 byte boundary. (The maximum burst length is INCR4. This is because, if the length is at least INCR8, 16-byte boundary is exceeded.)
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
1, 0	MAX_BURST_LEN[1:0]	0h	RW	This field selects the maximum burst length used when the BUS Master interface issues a transfer request. 00b: INCR16 01b: INCR8 10b: INCR4 11b: SINGLE

(4) USBCTR Register (USB2m_HOST_USBCTR)

Access Size : 32 bits

Address : <USB2m_host_base> + 020Ch

Initial Value : 0000_0002h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	DIRPD	PLL_RST	USBH_RST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
2	DIRPD	0h	RW	USBPHY Standby Mode Control 0b: USBPHY normal operating mode 1b: USBPHY standby mode Note: When the USB module is not in use, setting this bit to 1 reduces power consumption by the USBPHY module. Only set it to 1b when the USB module is not in use. In transitions from USBPHY standby mode to USBPHY normal operating mode, assert the reset signal for the USBPHY module for at least 1 µs before the transition.
1	PLL_RST	1h	RW	This bit controls resetting of the USBPHY module. 0b: The USBPHY reset is released. 1b: The USBPHY module is reset.
0	USBH_RST	0h	W	The read value is undefined Software reset to this module. Setting this bit to 1b resets this module entirely. This bit is always read as 0b. Note: Set this bit only when the BUS Master interface of this module is not running. Access to this module becomes valid 10 CLK (internal bus clock [P1φ]) after this bit is written. 0b: Nothing occurs. 1b: Reset is issued to this module.

(5) Register Enable/Clock Gating Control Register (USB2m_HOST_REGEN_CG_CTRL)

Access Size : 32 bits

Address : <USB2m_host_base> + 0304h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NONUSE_CLK_MSK	-	HOST_CLK_MSK	PERI_CLK_MSK	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	UPHY_WEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	NONUSE_CLK_MSK	0h	RW	This bit is used to mask the clock for the unused host or peripheral controller, depending on the setting value of the OTG_PERI bit of the COMMCTRL register. 0b: Do not mask the clock. 1b: Mask the clock. For details, see **(2) Specifications of NONUSE_CLK_MSK operation**.
30	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
29	HOST_CLK_MSK	0h	RW	This bit is used to forcibly mask clock supply to the host controller. 0b: Do not mask the clock supply. 1b: Mask the clock supply. For details, see **(3) Specifications of PERI_CLK_MSK and HOST_CLK_MSK operations**.
28	PERI_CLK_MSK	0h	RW	This bit is used to forcibly mask the clock supply to the peripheral controller. 0b: Do not mask the clock supply. 1b: Mask the clock supply. For details, see **(3) Specifications of PERI_CLK_MSK and HOST_CLK_MSK operations**.
27 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
0	UPHY_WEN	0h	RW	This bit controls write access to the UTMI_CTRL register. 0b: Write access to the UTMI_CTRL register is disabled. 1b: Write access to the UTMI_CTRL register is enabled.

(6) Suspend Control Register (USB2m_HOST_SPD_CTRL)

Access Size : 32 bits

Address : <USB2m_host_base> + 0308h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SUSPENDM_ENABLE	SLEEPM_ENABLE	-	-	-	-	-	-	WKCNNT_ENABLE	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	R	R	R	R	R	R	RW	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	GLOBAL_SUSPENDM_P1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	SUSPENDM_ENABLE	0h	RW	The setting of this bit is valid only in the Host mode (when the OTG_PERI bit of the COMMCTRL register is 0b). This bit is used to place USBPHY into the suspend status (the state in which PHY built-in PLL is stopped) when this module is changed to the USB Suspend mode. If this bit is set to 1b, the Suspend related bits of the OHCI/EHCI Operational registers below are set. If this module is changed to the USB Suspend mode, USBPHY is placed into the suspend status. [This function is valid for the following OHCI/EHCI Operational registers] EHCI: bit [7] (Suspend bit) of the PORTSC1 register OHCI: bit [2] (PSS bit) of the HcRhPortStatus register OHCI: bit [7:6] (HCFS field) of the HcControl register
30	SLEEPM_ENABLE	0h	RW	The setting of this bit is valid only in the Host mode (when the OTG_PERI bit of the COMMCTRL register is 0b). This bit is used to place USBPHY into the sleep status when the LPM function is used to change to the L1 Suspend mode. (In the sleep status, the PHY built-in PLL is running, but the 60MHz clock from USBPHY is gated.) This bit is valid when "Suspend using L1" of the POTSC register is 1b. If this bit is set to 1b, an L1 transition request is issued to the device. If the request is accepted, USBPHY is placed into the sleep status after the following EHCI register bit is set to 1b. [This function is valid for the following OHCI/EHCI Operational register] EHCI: bit [7] (Suspend bit) of the PORTSC1 register
29 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
23	WKCNNT_ENABLE	0h	RW	The setting of this bit is valid only in the Host mode (the OTG_PERI bit of the COMMCTRL register is 0b). If this bit is set to 1b, when a device disconnect occurs while USBPHY is in the suspend or sleep status, the suspend or sleep status is released. This bit is valid only when the SUSPENDM_ENABLE bit (bit 31) or SLEEP_ENABLE bit (bit 30) is 1b. Note: To set 1b to the SUSPENDM_ENABLE bit (bit 31) or SLEEPM_ENABLE bit (bit 30), as the general rule, set this bit to 1b. If, in the above case, this bit is not set to 1b, the suspend or sleep status cannot be released even if a device disconnect occurs.
22 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
0	GLOBAL_SUSPENDM_P1	0h	RW	The setting of this bit is valid regardless of the value of the OTG_PERI bit of the COMMCTRL register. This bit is used to forcibly place USBPHY into the suspend status (in which the PHY built-in PLL is stopped). If this bit is set to 1b, USBPHY is placed into the suspend status, regardless of the operating status and port status of the host controller. Note: – Do not set this bit to 1b during data transfer. We recommend that you set this bit to 1b after "stopping the EHCI/OHCI list processing" and "placing the port in the Disable status". – If the SUSPENDM_ENABLE bit (bit 31) is 1b, do not set this bit to 1b.

(7) Suspend/Resume Timer Setting Register (USB2m_HOST_SPD_RSM_TIMSET)

Access Size : 32 bits
Address : <USB2m_host_base> + 030Ch
Initial Value : 01F4_03E8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TIMER_CONNECT[15:0]															
Initial Value	0	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIMER_RESUME[15:0]															
Initial Value	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TIMER_CONNECT[15:0]	1F4h	RW	This field indicates the timer value used by USBPHY to detect Device Connect/Disconnect of the device in the suspend status (in which the PHY built-in PLL is stopped) when the SUSPENDM_ENABLE bit of the SPD_CTRL register is set to 1b. When USBPHY is in suspend status, whether Connect/Disconnect occurs is judged by the internal bus clock (P1φ). According to the internal bus clock (P1φ) frequency, specify the setting so that this timer value becomes at least 2.5 µs. 1 bit = 1 cycle (µs) (Setting guideline) For 100 MHz: At least FAh
15 to 0	TIMER_RESUME[15:0]	3E8h	RW	This field indicates the timer value used by USBPHY to detect RemoteWakeup signal from the device in suspend status (in which the PHY built-in PLL is stopped) when the SUSPENDM_ENABLE bit of the SPD_CTRL register is set to 1b. When USBPHY is in suspend status, whether the RemoteWakeup signal or not is judged by the internal bus clock (P1φ). According to the internal bus clock (P1φ) frequency, specify the setting so that this timer value becomes at least 5 µs. 1 bit = 1 cycle (µs) (Setting guideline) For 100 MHz: At least 1F4h

(8) Overcurrent Detection/Sleep Timer Setting Register (USB2m_HOST_OC_SLP_TIMSET)

Access Size : 32 bits

Address : <USB2m_host_base> + 0310h

Initial Value : 0C83_0D40h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	TIMER_SLEEP[8:0]									TIMER_OC[19:16]			
Initial Value	0	0	0	0	1	1	0	0	1	0	0	0	0	0	1	1
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIMER_OC[15:0]															
Initial Value	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
28 to 20	TIMER_SLEEP [8:0]	C8h	RW	This field indicates the timer value used by USBPHY when the SLEEPM_ENABLE bit of the SPD_CTRL register is set to 1b. This timer value is used to measure the time for detecting RemoteWakeup reception and Resume-K drive time during sleep status (in which the PHY built-in PLL is running, but 60MHz clock from the USBPHY is gated). When USBPHY is in sleep status, whether the RemoteWakeup signal or not is judged by the internal bus clock (P1φ). According to the internal bus clock (P1φ) frequency, specify the setting so that this timer value becomes 1 µs. 1 bit = 1 cycle (µs) (Setting guideline) For 100 MHz: 064h
19 to 0	TIMER_OC [19:0]	30D40h	RW	This field indicates the timer value used for overcurrent detection. If the overcurrent input (OVRCUR) set in this field is continuously asserted (0b) for the duration set in this register, this module determines that overcurrent occurred. According to the internal bus clock (P1φ) frequency, specify the setting so that this timer value becomes at least 1 ms. 1 bit is 1 cycle (µs). (Setting guideline) For 100 MHz and 1 ms: At least 1_86A0h

(9) SBRN_FLADJ_PORTWAKECAP Register (USB2m_HOST_SBRN_FLADJ_PW)

Access Size : 32 bits

Address : <USB2m_host_base> + 0314h

Initial Value : 0003_2020h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PORTWAKECAP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FLADJ[7:0]								SBRN[7:0]							
Initial Value	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	PORTWAKECAP[15:0]	3h	RW	This field is used to mask the ports (of the connected device) that are used for Wakeup event. Operation on this field has no effect on the operation of the HOST module.
15 to 8	FLADJ[7:0]	20h	RW	This field adjusts the length of one micro frame by 16HS bit time unit. The initial value indicates 20h (60000d HS bit time).
7 to 0	SBRN[7:0]	20h	R	This field indicates the Serial Bus Release Number. The fixed value “20h” is indicated.

(10) UTMI+PHY Control Register (USB2m_HOST_UTMI_CTRL)

Abbreviated name of register: HOST_UTMI_CTRL

Access Size : 32 bits

Address : <USB2m_host_base> + 0318h

Initial Value : 8000_0004h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Renesas Private	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	Renesas Private								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	Renesas Private	1h	RW	Reserved bit. Set the value specified in **6.5.2.9.1 Host/Peripheral Common Setting Sequence**.
30 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
8 to 0	Renesas Private	4h	RW	Reserved bit. Set the value specified in **6.5.2.9.1 Host/Peripheral Common Setting Sequence**.

(11) PORT_LPM_CTR1 Register (USB2m_HOST_PORT_LPM_CTRL1)

Access Size : 32 bits
Address : <USB2m_host_base> + 0320h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	NYET_RETRY_CNT_P1[3:0]				REMOTEWAKE_EN_P1	SLEEP_INT_EN_P1	RETRY_ENABLE_NYET_P1	HIRD_SEL_P1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
7 to 4	NYET_RETRY_CNT_P1[3:0]	0h	RW	This field sets the number of retries that are allowed when the response from the device in the LPM transaction was NYET. The setting value of this bit is valid if the RETRY_ENABLE_NYET_P1 bit (bit 1) is 1b. 0000b: No retry. 0001b to 1111b: Retries the set number of times. (MAX: 15 retries)
3	REMOTEWAKE_EN_P1	0h	RW	This bit is used to indicate the value of the RemoteWakeup bit of the LPM Token. 0b: RemoteWakeup is supported. 1b: RemoteWakeup is not supported.
2	SLEEP_INT_EN_P1	0h	RW	This bit is used to enable an interrupt to occur when a status other than ACK is received in the LPM transaction. The Per-Port Change interrupt can occur. 0b: No interrupt occurs. 1b: An interrupt occurs.
1	RETRY_ENABLE_NYET_P1	0h	RW	This bit is used to set the behavior of the host controller when a response from the device is NYET in an LPM transaction. 0b: No retry. 1b: Retries are made.
0	HIRD_SEL_P1	0h	RW	This bit sets the time for K drive for when recovered from the Sleep state. Based on the setting values of this bit and EHCI USBCMD Register bit [27:24], the K drive time is determined as **Note**.

Note: ■ Supplementary Note on HIRD_SEL_P1 Bit

USBCMD Register	HIRD_SEL_P1 (Setting value of this bit)	
Bits [27:24]	0b	1b
0000b	75 µs	50 µs
0001b	100 µs	125 µs
0010b	150 µs	200 µs
0011b	250 µs	275 µs
0100b	350 µs	350 µs
0101b	450 µs	425 µs
0110b	950 µs	500 µs
0111b	1950 µs	575 µs

USBCMD Register	HIRD_SEL_P1 (Setting value of this bit)	
Bits [27:24]	0b	1b
1000b	2950 µs	650 µs
1001b	3950 µs	725 µs
1010b	4950 µs	800 µs
1011b	5950 µs	875 µs
1100b	6950 µs	950 µs
1101b	7950 µs	1025 µs
1110b	8950 µs	1100 µs
1111b	9950 µs	1175 µs

6.5.2.2.6 UCOM Register Descriptions

(1) Common Control Register (USB2m_HOST_COMMCTRL)

Access Size : 32 bits

Address : <USB2m_host_base> + 0800h

Initial Value : 8000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OTG_PERI	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	OTG_PERI	1h	RW	This bit specifies whether this module is set to the host mode or function mode. 0b: Host mode 1b: Function mode
30 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(2) OTG-BC Interrupt Status Register (USB2m_HOST_OBINTSTA)

Access Size : 32 bits

Address : <USB2m_host_base> + 0804h

Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DPMONCHG_STA	DMMONCHG_STA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	CHGDETCHG1_STA	-	PDDETCHG1_STA	-	-	OCINT_STA	IDCHG_STA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	RW1	R	RW1	R	R	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
17	DPMONCHG_STA	0h	RW1	This bit is set if the DPMON bit of the LINECTRL1 register has changed. 0b: The DPMON bit of the LINECTRL1 register has not changed. 1b: The DPMON bit of the LINECTRL1 register has changed.
16	DMMONCHG_STA	0h	RW1	This bit is set if the DMMON bit of the LINECTRL1 register has changed. 0b: The DMMON bit of the LINECTRL1 register has not changed. 1b: The DMMON bit of the LINECTRL1 register has changed.
15 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
6	CHGDETCHG1_STA	0h	RW1	This bit is set if the CHGDETSTS bit of the BCCTRL1 register has changed from 0b to 1b. 0b: The CHGDETSTS bit of the BCCTRL1 register has not changed from 0b to 1b 1b: The CHGDETSTS bit of the BCCTRL1 register has changed from 0b to 1b.
5	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
4	PDDETCHG1_STA	0h	RW1	This bit is set if the PDDETSTS bit of the BCCTRL1 register has changed from 0b to 1b. 0b: The PDDETSTS bit of the BCCTRL1 register has not changed from 0b to 1b. 1b: The PDDETSTS bit of the BCCTRL1 register has changed from 0b to 1b.
3, 2	-	0h	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	OCINT_STA	0h	RW1	This bit is set if the OVRCUR pin is asserted. 0b: The OVRCUR pin is not asserted (and remains 1b). 1b: The OVRCUR pin is asserted (and became 0b).
0	IDCHG_STA	1h	RW1	This bit is set if the input value from the OTG_ID pin has changed. 0b: There is no change in the OTG_ID pin. 1b: There is a change in the OTG_ID pin. Note: The initial value is 1. Before using this bit, clear the status.

(3) OTG-BC Interrupt Enable Register (USB2m_HOST_OBINTEN)

Access Size : 32 bits

Address : <USB2m_host_base> + 0808h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DPMONCHG_EN	DMMONCHG_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	CHGDETCHG1_EN	-	PDDETCHG1_EN	-	-	OCINT_EN	IDCHG_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	R	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
17	DPMONCHG_EN	0h	RW	DPMONCHG_STA bit interrupt enable 0b: Interrupt via the DPMONCHG_STA bit is disabled. 1b: Interrupt via the DPMONCHG_STA bit is enabled.
16	DMMONCHG_EN	0h	RW	DMMONCHG_STA bit interrupt enable 0b: Interrupt via the DMMONCHG_STA bit is disabled. 1b: Interrupt via the DMMONCHG_STA bit is enabled.
15 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
6	CHGDETCHG1_EN	0h	RW	CHGDETCHG1_STA bit interrupt enable 0b: Interrupt via the CHGDETCHG1_STA bit is disabled. 1b: Interrupt via the CHGDETCHG1_STA bit is enabled.
5	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
4	PDDETCHG1_EN	0h	RW	PDDETCHG1_STA bit interrupt enable 0b: Interrupt via the PDDETCHG1_STA bit is disabled. 1b: Interrupt via the PDDETCHG1_STA bit is enabled.
3, 2	-	0h	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	OCINT_EN	0h	RW	OCINT_STA bit interrupt enable 0b: Interrupt via the OCINT_STA bit is disabled. 1b: Interrupt via the OCINT_STA bit is enabled.
0	IDCHG_EN	0h	RW	IDCHG_STA bit interrupt enable 0b: Interrupt via the IDCHG_STA bit is disabled. 1b: Interrupt via the IDCHG_STA bit is enabled.

(4) VBUS Control Register (USB2m_HOST_VBCTRL)

Access Size : 32 bits

Address : <USB2m_host_base> + 080Ch

Initial Value : 0001_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	VGPUO	-	-	-	VBOUT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
16	-	1h	R	Reserved Whenever it is read, 1b is read. The write value should always be 1b.
15 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
4	VGPUO	0h	RW	The level corresponding to the inverse of this bit (in terms of positive logic) is output from the OTG_EXICEN pin. This bit is used, for example, for control of the external power IC.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
0	VBOUT	0h	RW	This bit is one of the VBUSEN control bits. This bit is used to assert VBUS by controlling the external power IC. 0b: VBUS output disable 1b: VBUS output enable If overcurrent occurs, this bit is automatically cleared to 0b.

(5) Line Control Port 1 Register (USB2m_HOST_LINECTRL1)

Access Size : 32 bits
Address : <USB2m_host_base> + 0810h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	DPRPU_EN	DP_RPU	DPRPD_EN	DP_RPD	DMRPD_EN	DM_RPD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	DPMON	DMMON	-	IDMON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
25, 24	-	0h	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
21	DPRPU_EN	0h	RW	This bit enables the DP_RPU (bit 20) setting for pullup control of D+ in port 1. 0: DP_RPU (bit 20) setting for pullup control of D+ in port 1 is disabled. 1: DP_RPU (bit 20) setting for pullup control of D+ in port 1 is enabled.
20	DP_RPU	0h	RW	This bit controls pullup of D+ in port 1b. This control is enabled only when DPRPU_EN (bit 21) = 1b. Note that if this bit is cleared to 0b when DPRPU_EN (bit 21) = 1b, D+ is in the floating state while no device drives the USB bus. Note also that setting DPRPD_EN (bit 19) and DP_RPD (bit 18) to enabled is prohibited while this bit is set to enabled - that is, enabling both pullup and pulldown of D+ together is prohibited. 0: Pullup of D+ in port 1 is disabled. 1: Pullup of D+ in port 1 is enabled.
19	DPRPD_EN	0h	RW	This bit enables DP_RPD (bit 18) to control USB bus (DP) 15 kΩ Pulldown resistor. 0b: Control of DP-side 15 kΩ Pulldown resistor by DP_RPD (bit 18) is disabled. 1b: Control of DP-side 15 kΩ Pulldown resistor by DP_RPD (bit 18) is enabled.
18	DP_RPD	0h	RW	This bit controls USB bus (DP) 15 kΩ Pulldown resistor when DPRPD_EN (bit 19) = 1b. 0b: DP-side 15 kΩ Pulldown resistor is OFF. 1b: DP-side 15 kΩ Pulldown resistor is ON.
17	DMRPD_EN	0h	RW	This bit enables DM_RPD (bit 16) to control USB bus (DM) 15 kΩ Pulldown resistor. 0b: Control of DM-side 15 kΩ Pulldown resistor by DM_RPD (bit 16) is disabled. 1b: Control of DM-side 15 kΩ Pulldown resistor by DM_RPD (bit 16) is enabled.
16	DM_RPD	0h	RW	This bit controls USB bus (DM) 15 kΩ Pulldown resistor when DMRPD_EN (bit 17) = 1b. 0b: DM-side 15 kΩ Pulldown resistor is OFF. 1b: DM-side 15 kΩ Pulldown registor is ON.
15 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
3	DPMON	0h	R	This bit indicates the value of USB bus DP.
2	DMMON	0h	R	This bit indicates the value of USB bus DM.
1	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
0	IDMON	0h	R	This bit indicates the value of the OTG_ID pin.

(6) BC Control Port 1 Register (USB2m_HOST_BCCTRL1)

Access Size : 32 bits
Address : <USB2m_host_base> + 0820h
Initial Value : 0300_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PDDETSTS	CHGDETSTS	-	-	DCPMODE	VDMSRCE	IDPSINKE	VDPSRCE	IDMSINKE	IDPSRCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
25, 24	-	3h	R	Reserved Whenever it is read, 1b is read. The write value should always be 1b.
23 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9	PDDETSTS	0h	R	This bit indicates the USBPHY Portable Device Detect signal state.
8	CHGDETSTS	0h	R	This bit indicates the USBPHY Charging Downstream Port Detect signal state.
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
5	DCPMODE	0h	RW	If USBPHY is used as DCP (Dedicated Charging Port), this bit is set to 1b.
4	VDMSRCE	0h	RW	This bit controls the USBPHY built-in VDM_SRC circuit. If this bit is set to 1b, VDM_SRC goes ON, and the DM pin is driven.
3	IDPSINKE	0h	RW	This bit controls the USBPHY built-in Portable Device Detect circuit. If this bit is set to 1b, Portable Device detection is enabled.
2	VDPSRCE	0h	RW	This bit controls the USBPHY built-in VDP_SRC circuit. If this bit is set to 1b, VDP_SRC goes ON, and the DP pin is driven.
1	IDMSINKE	0h	RW	This bit controls the USBPHY built-in Charging Downstream Port Detect circuit. If this bit is set to 1b, Charging Downstream Port detection is enabled.
0	IDPSRCE	0h	RW	This bit controls the USBPHY built-in IDP_SRC circuit. If this bit is set to 1b, IDP_SRC goes ON, and the DP pin is driven.

6.5.2.3 Clock Signals

6.5.2.3.1 Clock Gating Specifications

(1) Overview of clock gating

Because this module has a feature of switching the host controller and peripheral controller, clock supply to the unused controller might not be necessary.

Therefore, three clock gating control bits are allocated in the Register Enable/Clock Gating Control Register for the purpose of reducing power consumption by implementing clock gating for the “circuits that do not require clock supply temporarily.”

The clock gating control bits can be controlled to gate the clock that is supplied to the host controller or the peripheral controller.

[Target Register]

Register Enable/Clock Gating Control Register (offset: 304h)

[Functional specification]

Bit	Symbol	Functional Specification
31	NONUSE_CLK_MSK	Gating clocks for unused host controller or peripheral controller.
29	HOST_CLK_MSK	Gating clocks for host controller.
28	PERI_CLK_MSK	Gating clocks for peripheral controller.

(2) Specifications of NONUSE_CLK_MSK operation

This function automatically gates the clock to the unused host controller or the unused peripheral controller.

This function is enabled if the NONUSE_CLK_MSK bit (bit 31) is set to 1b.

If it is no problem whether the clock supply to the unselected function is stopped, use this function.

The following table shows the operating specifications of this function, based on the setting of the OTG_PERI bit (bit 31) of the Common Control Register (offset: 800h).

Clock Gating Register Setting			Host/Peripheral Switching Setting	Gating Target	
			Register		
NONUSE_CLK_MSK	HOST_CLK_MSK	PERI_CLK_MSK	OTG_PERI	Host Controller	Function Controller
1	0	0	0	—	✓
			1	✓	—

[Note on using the NONUSE_CLK_MSK bit]

If the NONUSE_CLK_MSK bit is used, as the general rule, set the HOST_CLK_MSK/PERI_CLK_MSK bit to 0b.

If the HOST_CLK_MSK/PERI_CLK_MSK bit is 1b, the effect of clock gating becomes logical OR of each bit.

(3) Specifications of PERI_CLK_MSK and HOST_CLK_MSK operations

This function forcibly gates the clock to the host controller and function controller.

The following table shows the operating specifications of this function.

Clock Gating Register Setting			Host/Peripheral Switching Setting	Gating Target	
			Register		
NONUSE_CLK_MSK	HOST_CLK_MSK	PERI_CLK_MSK	OTG_PERI	Host Controller	Function Controller
0	1	0	—	✓	
	0	1	—		✓
	1	1	0	✓	✓

Note: If the HOST_CLK_MSK/PERI_CLK_MSK bit is used, as the general rule, set the NONUSE_CLK_MSK bit to 0b.
If the NONUSE_CLK_MSK bit is set t0 1b, the effect of clock gating becomes logical OR of each bit.

6.5.2.4 Interrupt Sources

6.5.2.4.1 Interrupt Signals

This module has the five interrupt signals listed below.

Interrupt Source Name	Interrupt Type	Pulse/ Level	Active Level
U2H_INT	BUS Master interrupt signal. This signal is asserted when a bus error occurs in the BUS Master. Interrupt control is performed by the AHB Bridge Register.	Level	H
U2H_OHCI_INT	OHCI interrupt signal This signal is asserted during FS/LS transfer, when data transfer finishes or when the change of the USB bus state is detected. Interrupt control is performed by the OHCI Operational Register.	Level	H
U2H_EHCI_INT	EHCI interrupt signal. This signal is asserted during HS transfer, when data transfer finishes or when the change of the USB bus state is detected. Interrupt control is performed by the EHCI Operational Register.	Level	H
U2H_WAKEON_INT	EHCI Wakeup interrupt signal. This signal is asserted by an EHCI Wakeup event. Interrupt control is performed by the EHCI Operational Register.	Level	H
U2H_OBINT	OTG/Battery Charging interrupt signal. This signal is asserted by OTG or Battery Charging related event. Interrupt control is performed by the UCOM2 Register.	Level	H

6.5.2.4.2 Interrupt Sources and Control

(1) U2H_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the register below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset: 200h) bit[0] (AHB_INTEN)

[Interrupt source]

A bus error (MHRESP = 1b) occurs in the AHB master.

[Clearing the interrupt]

To clear the interrupt, write 1b to the relevant bit in the register below.

AHB Bridge Register INT_STATUS Register (offset: 204h) bit[0] (AHB_INT)

(2) U2H_OHCI_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset: 200h) bit[1] (USBH_INTAEN)

OHCI Operational Register HcInterruptEnable Register (offset: 010h) bit[31], bit[6:0]*1

Note 1. Enable the bit required as the assertion source.

[Interrupt source]

Interrupt Source		Registers That Require Interrupt Enable Setting							
			HcInterruptEnable						
		USBH_INTAEN	bit [31] MIE	bit [6] RHSCE	bit [5] FNOE	bit [3] ROE	bit [2] SFE	bit [1] WDHE	bit [0] SOE
1	Device connection is detected.	✓	✓	✓	—	—	—	—	—
2	Device disconnection is detected.	✓	✓	✓	—	—	—	—	—
3	Port power is OFF (excluding overcurrent detection).	✓	✓	✓	—	—	—	—	—
4	Babble error is detected during a USB transfer.	✓	✓	✓	—	—	—	—	—
5	Resume is complete.	✓	✓	✓	—	—	—	—	—
6	Overcurrent is detected.	✓	✓	✓	—	—	—	—	—
7	Bus reset is complete.	✓	✓	✓	—	—	—	—	—
8	When the HcRhDescritorB Register DR bit is 1b, OHCI becomes “USB Operational” (HCFS[1:0] bit = 10b) or “USB Suspend” (HCFS[1:0] bit = 11b).	✓	✓	✓	—	—	—	—	—
9	When no device is connected (CCS bit = 0b), 1 is written to bit [0] (Clear Port Enable) of the OHCI HcRhPort Status register.	✓	✓	✓	—	—	—	—	—
10	When no device is connected (CCS bit = 0b), 1 is written to bit [1] (Set Port Enable) of the OHCI HcRhPort Status register.	✓	✓	✓	—	—	—	—	—
11	When no device is connected (CCS bit = 0b), 1 is written to bit [2] (Set Port Suspend) of the OHCI HcRhPort Status register.	✓	✓	✓	—	—	—	—	—
12	When no device is connected (CCS bit = 0b), 1 is written to bit [3] (Clear Suspend Status) of the OHCI HcRhPort Status register.	✓	✓	✓	—	—	—	—	—
13	When no device is connected (CCS bit = 0b), 1 is written to bit [4] (Set Port Reset) of the OHCI HcRhPort Status register.	✓	✓	✓	—	—	—	—	—
14	When no device is connected (CCS bit = 1b), the port power is turned off.	✓	✓	✓	—	—	—	—	—
15	The MSB of bit [15:0] (Frame Number) of the HcFmNumber register has changed.	✓	✓	—	✓	—	—	—	—
16	RemoteWakeup signal (Resume signal) is detected from a device.	✓	✓	—	—	✓	—	—	—
17	HccaFrameNumber is updated. (Almost the same meaning as SOF is sent.)	✓	✓	—	—	—	✓	—	—
18	A transfer finishes (including an error), and the host module updated HccaDoneHead.	✓	✓	—	—	—	—	✓	—
19	USB schedule overrun occurred for the frame.	✓	✓	—	—	—	—		✓

[Clearing the interrupt]

To clear the interrupt, write 1b to the bit corresponding to the interrupt source in the register below to clear the interrupt.

OHCI Operational Register HcInterruptStatus Register (offset: 00Ch) bit[6:0]

(3) U2H_EHCI_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.
AHB Bridge Register INT_ENABLE Register (offset: 200h) bit[2] (USBH_INTBEN)

EHCI Operational Register USBINTR Register (offset: 128h) bit[16], bit[5:0]*[1]

Note 1. Enable the bit required as the assertion source.

Also, control the relevant bit(s) in the register below as required.

EHCI Operational Register USBCMD Register (offset: 120h) bit[15], bit[6]

[Interrupt source]

Interrupt Source		Registers That Require Interrupt Enable Setting								
			USBCMD		USBINTR					
			bit [15]	bit [6]	bit [16]	bit [5]	bit [3]	bit [2]	bit [1]	bit [0]
		USBH_INTB EN	Per-Port Change Event	Doorbell	Port-1 Change Event	Async Advance	Frame List Rollover	Port Change Detect	USB ERRINT	USB INT
[1]	Device connection is detected.	✓	—	—	—	—	—	✓	—	—
[2]	Device disconnection is detected.	✓	—	—	—	—	—	✓	—	—
[3]	Overcurrent is detected.	✓	—	—	—	—	—	✓	—	—
[4]	RemoteWakeup signal (Resume Signal) is detected from a device.	✓	—	—	—	—	—	—	—	—
[5]	Babble status of the USB bus is detected.	✓	—	—	—	—	—	✓	—	—
[6]	USB transfer with “qTD IOC = 1b” normally finishes.	✓	—	—	—	—	—	—	—	✓
[7]	Short packet is received.	✓	—	—	—	—	—	—	—	✓
[8]	USB transfer finished with an error. (Retry transfer failed three times. A bubble error was detected. STALL was received.)	✓	—	—	—	—	—	—	✓	—
[9]	The QH processing normally finished while the USBCMD Register bit [6] (Interrupt on Async Advance Doorbell) is 1b.	✓	—	✓	—	✓	—	—	—	—
[10]	The FRINDEX Register Frame Index bit returned from the maximum value to 000h (rollover detected).	✓	—	—	—	—	✓	—	—	—
[11]	The Port Change Detect event (interrupt source 1 to 5) was detected.	✓	✓	—	✓	—	—	—	—	—

[Clearing the interrupt]

To clear the interrupt, write 1b to the bit corresponding to the interrupt source in the register below to clear the interrupt.

EHCI Operational Register USBSTS Register (offset: 124h) bit[17:16], bit[5:0]

(4) U2H_WAKEON_INT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset: 200h) bit[4] (WAKEON_INTEN)

EHCI Operational Register PORTSC[1:2] Register (offset: 164h/168h) bit[22:20]*[1]

Note 1. Enable the bit required as the assertion source.

[Interrupt source]

Interrupt Source		Registers That Require Interrupt Enable Setting			
			PORTSC		
		WAKEON_INTEN	bit [22]	bit [21]	bit [20]
			WKOC_E	WKDSCNNT_E	WKCNNT_E
1	Device connection is detected.	✓	—	—	✓
2	Device disconnection is detected.	✓	—	✓	—
3	Overcurrent is detected.	✓	✓	—	—
4	RemoteWakeup signal (Resume Signal) is detected from a device.	✓	—	—	—

[Clearing the interrupt]

To clear the interrupt, write 1b to the relevant bit in the register below.

AHB Bridge Register INT_STATUS Register (offset: 204h) bit[4] (WAKEON_INT)

(5) U2H_OBINT assertion source and control

[Interrupt enable control by register]

Assert the interrupt enable bit in the registers below. The interrupt signal is asserted when an interrupt source occurs.

AHB Bridge Register INT_ENABLE Register (offset: 200h) bit[3] (UCOM_INTEN)

UCOM Register OTG-BC Interrupt Enable Register (offset: 808h) bit[17:16], bit[6], bit[4], bit[1:0]*[1]

Note 1. Enable the bit required as the assertion source.

[Interrupt source]

Interrupt Source		Registers That Require Interrupt Enable Setting		
			OTG-BC Interrupt Enable	
			bit [17]	bit [16]
		UCOM_INTEN	DPMONCHG_EN	DMMONCHG_EN
1	The DP pin has changed.	✓	✓	—
2	The DM pin has changed.	✓	—	✓

Interrupt Source		Registers That Require Interrupt Enable Setting		
			OTG-BC Interrupt Enable	
			bit [6]	bit [4]
		UCOM_INTEN	CHGDETCHG1_EN	PDDETCHG1_EN
1	The portable device detection signal has changed.	✓	—	✓
2	The charging port detection signal has changed.	✓	✓	—

Interrupt Source		Registers That Require Interrupt Enable Setting		
			OTG-BC Interrupt Enable	
			bit [1]	bit [0]
		UCOM_INTEN	OCINT_EN	IDCHG_EN
1	Overcurrent is detected. (The change of the OVRCURN pin from 1 to 0b is detected.)	✓	✓	—
2	The OTG_ID pin has changed.	✓	—	✓

[Clearing the interrupt]

To clear the interrupt, write 1b to the bit corresponding to the interrupt source in the register below to clear the interrupt.

UCOM Register OTG-BC Interrupt Status Register (offset: 804h) bit[17:16], bit[6], bit[4], bit[1:0]

6.5.2.4.3 Timing of De-asserting Interrupt Signals

After the register access to clear an interrupt source, it may take time to begin clearing the interrupt triggered by the interrupt source. Therefore, take a measure to prevent the false recognition of interrupts during the period from the end of the register clear access until the next interrupt is recognized.

6.5.2.5 Power-Saving Function

This module controls power consumption by using the following two methods:

(1) Controlling the SUSPENDM/SLEEPM pin of USB2PHY

(2) Using clock gating to stop the clock to the host controller or peripheral controller.

6.5.2.5.1 Controlling the SUSPENDM and SLEEPM Pins of the USB2PHY

You can expect the following power-saving effects by asserting the SUSPENDM and SLEEPM pins of the USB2PHY:

- Reducing the power consumption by the USB2PHY
- Reducing the power consumption by the host core by stopping the clocks from the USB2PHY

As described at **(5) Suspend extension function**, by default, the SUSPENDM and SLEEPM pins of the USB2PHY are not asserted even when EHCI and OHCI are put into the Suspended state.

See **(6) Suspend Control Register (USB2m_HOST_SPD_CTRL)** and control the relevant registers appropriately.

6.5.2.5.2 Controlling the Clock-Gating Function

See **(5) Register Enable/Clock Gating Control Register (USB2m_HOST_REGEN_CG_CTRL)** and **6.5.2.3.1 Clock Gating Specifications**, then control the relevant registers appropriately.

6.5.2.6 Battery Charging

6.5.2.6.1 Support of charging port

The charging port refers to a port that supplies power in compliance with the Battery Charging Specification. The charging port is usually installed on the host controller.

Charging ports can be generally classified by function as described below.

Type	Function
CDP (Charging Downstream Port)	Downstream port that can supply power in compliance with the Battery Charging Specification. This type of charging port detects a portable device, and after the handshake (for Battery Charging) finishes, it proceeds to the usual device connect sequence (and operates as the usual host).
DCP (Dedicated Charging Port)	Port that provides only power supply function in compliance with the Battery Charging Specification. This type of charging port does not operate as the usual host.
SDP (Standard Downstream Port)	Standard Downstream Port that is not compliant with the Battery Charging Specification. This type of charging port supplies power in the range conforming to the conventional USB 2.0 standard, and operates as the usual host.

6.5.2.6.2 Support of portable device

A portable device refers to a device that is supplied power (or that requests power supply) in compliance with the Battery Charging Specification. The portable port is usually installed in the peripheral controller.

6.5.2.7 Bus Master

6.5.2.7.1 Functional specifications of the bus master

(1) Supported bus master functions

Function as the BUS MASTER	Status
Residual burst after an error response is received	The transfer is not stopped.
1 Kbyte boundary processing	The fixed length burst (INCRx) across 1 Kbyte boundary is not performed.

(2) Issuing requests for different types of bus transfer

MHTRANS[1:0]	MHSIZE[2:0]	MHWRITE	MHBURST[2:0]	Reply	Remarks
IDLE (00b)	—	—	—	—	—
BUSY (01b)	—	—	—	—	Not issued.
NONSEQ (10b)	32-bit (010b)	WRITE	SINGLE INCR4 INCR8 INCR16	OKAY/ERROR	32-bit transfer is issued. A response error is reported by an interrupt, and the transfer is not stopped.
		READ			
	8-bit (000b) 16-bit (001b)	WRITE	SINGLE	OKAY/ERROR	8-bit/16-bit transfer is issued (SINGLE only). A response error is reported by an interrupt.
	other than the above	—	—	—	Not issued.
SEQ (11b)	32-bit (010b)	WRITE	INCR4 INCR8 INCR16	OKAY/ERROR	32-bit transfer is issued. A response error is reported by an interrupt, and the transfer is not stopped.
		READ		OKAY/ERROR	
	other than the above	—	—	—	Not issued.

(3) Supported responses

Response type	Response	Remarks
OKAY	Enable	Supported.
ERROR	Enable	A response error is reported by an interrupt, and termination of transfer (Early Burst Termination) is not performed.

(4) Protection control information

The value of MHPROT[3:0] can be set in the PROT_TYPE bits (bits [15:12]) in the AHB_BUS_CTR Register (offset: 208h).

Also, when the PROT_MODE bit (bit 8) in the AHB_BUS_CTR Register is set, only the last data transfer in an EHCI/OCHI DMA transfer can be handled as a non-buffered transfer, and other transfers can be handled as buffered transfers.

(5) Maximum burst length

The maximum burst length can be selected from SINGLE, INCR4, INCR8, and INCR16 by using the MAX_BURST_LEN bits (bit[1:0]) in the AHB_BUS_CTR Register (offset: 208h). The maximum burst length is common to reading and writing.

(6) Boundary of transfer data

The value of MHADDR[31:0] bits does not exceed 1 KB boundary during a burst transfer. Also, you can change the address boundary for burst transfer to 16, 32, or 64 bytes by writing a value to the ALIGN_ADDRESS bits (bit[5:4]) in the AHB_BUS_CTR Register (offset: 208h).

(7) Start address of fixed-length INCR burst transfer

The following table lists the values of the lower bits of MHADDR to be applied when a fixed-length INCR burst transfer starts.

ALIGN_ADDRESS Setting	Fixed length Start Address of INCR Burst		
	INCR4	INCR8	INCR16
00b (Aligned at the 1-Kbyte boundary)	MHADDR[9:0] = 000h 004h 008h 00Ch 010h ⋮ 3D0h 3D4h 3D8h 3DCh 3E0h 3E4h 3E8h 3ECh 3F0h	MHADDR[9:0] = 000h 004h 008h 00Ch 010h ⋮ 3C8h 3CCh 3D0h 3D4h 3D8h 3DCh 3E0h	MHADDR[9:0] = 000h 004h 008h 00Ch 010h ⋮ 3B8h 3BCh 3C0h
01b (Aligned at the 16-byte boundary)	MHADDR[3:0] = 0h	— (Not issued)	— (Not issued)
10b (Aligned at the 32-byte boundary)	MHADDR[4:0] = 00h 04h 08h 0Ch 10h	MHADDR[4:0] = 00h	— (Not issued)
11b (Aligned at the 64-byte boundary)	MHADDR[5:0] = 00h 04h 08h 0Ch 10h 14h 18h 1Ch 20h 24h 28h 2Ch 30h	MHADDR[5:0] = 00h 04h 08h 0Ch 10h 14h 18h 1Ch 20h	MHADDR[5:0] = 00h

6.5.2.8 Overcurrent Control and VBUS Control

6.5.2.8.1 OVRCURN/VBUSEN pin

Overcurrent detection on the USB port and port power (VBUS) control are performed by the external power IC connected to this module.

This module pin	Input/Output	Level	Description
OVRCURN	Input	L	Overcurrent status was detected.
		H	Overcurrent status was not detected.
VBUSEN	Output	L	Port Power (VBUS) OFF
		H	Port Power (VBUS) ON

6.5.2.8.2 Overcurrent detection timer setting

This module detects overcurrent when the OVRCURN pin remains asserted (0b) for a set period.

The period over which assertion of the OVRCURN pin is required (“Overcurrent detection time”) can be set in the following register.

Register	Overcurrent detection/sleep timer setting register (offset: 310h)
Bits	TIMER_OC[19:0]
Initial value	3_0D40h

The overcurrent detection time can be converted from the setting value of the above register, taking that one bit equal to the internal bus clock (P1ϕ) cycle.

Therefore, set the value of the above register at initial configuration, considering “the internal bus clock (P1ϕ) frequency to be used” and “the overcurrent detection time you want to specify”.

6.5.2.8.3 Port Power (VBUS) control specifications

VBUSEN can be controlled by the Port Power bit of the EHCI/OHCI Operational register or by the VBOUT bit of the VBUS Control Register. Which of the above bit controls VBUSEN is determined by the VBUSEN control (VBENCTL) register in USBPHY Control.

When using as Function, VBUSEN should be 0.

The following table describes control examples by the Port Power bit of the EHCI/OHCI Operational register.

Situation		Register		bit
EHCI control		PORTSC1 (offset: 164h)		bit 12 (PP)
OHCI control	Global control*1	ON setting	HcRhStatus Register (offset: 050h)	bit 16 (Set Global Power)
		OFF setting		bit 0 (Clear Port Status)
	Selective control*2	ON setting	HcRhPortStatus1 Register (offset: 054h)	bit 8 (Set Port Power)
		OFF setting		bit 9 (Clear Port Power)

Note 1. Global control refers to the status in which the register settings are as follows:
HcRhDescriptorA Register (offset: 048h) bit 8 (PSM) = 0b
or
HcRhDescriptorA Register (offset: 048h) bit 8 (PSM) = 1b
and
HcRhDescriptorB Register (offset: 04Ch) bit 17 (PPCM[1]) = 0b

Note 2. Selective control refers to the status in which the register settings are as follows:
HcRhDescriptorA Register (offset: 048h) bit 8 (PSM) = 1b
and
HcRhDescriptorB Register (offset: 04Ch) bit 17 (PPCM[1]) = 1b

However, if the register settings are as follows, the VBUSEN pin is always asserted (1b), and the Port Power (VBUS) becomes ON, regardless of the OVRCURN pin's status.

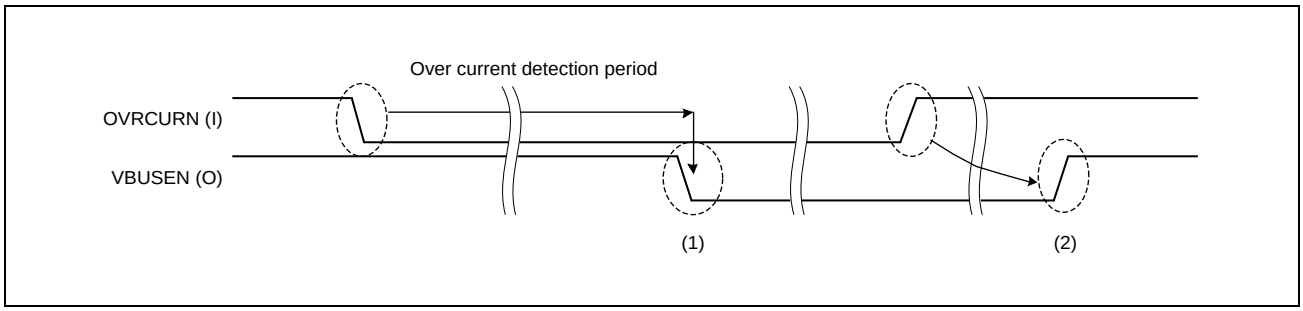
Specify the settings as necessary (for example, your system needs that VBUS is always ON).

EHCI Operational	OHCI Operational				
HCSPARAMS (offset: 104h)	HcRhDescriptorA (offset: 048h)			HcRhDescriptorB (offset: 04Ch)	Pin Operation
PPC (bit 4)	NOCP (bit 12)	NPS (bit 9)	PSM (bit 8)	PPCM[1] (bit 17)	When the OVRCURN pin is asserted (0b)
0	—	—	—	—	Fixed to 1b
—	1	—	—	—	Fixed to 1b
—	—	1	—	—	Fixed to 1b
1	0	0	0	—	0b
			1	0	
				1	

6.5.2.8.4 Timing Chart for Overcurrent Detection and Recovery

The figure below shows the assertion/de-assertion timings of the OVRCURN and VBUSEN pins signals at overcurrent detection and recovery.

Note that this timing chart is on the assumption that the changes of register settings to fix the Port Power bit to asserted state (see **6.5.2.8.3 Port Power (VBUS) control specifications**) have not been made.



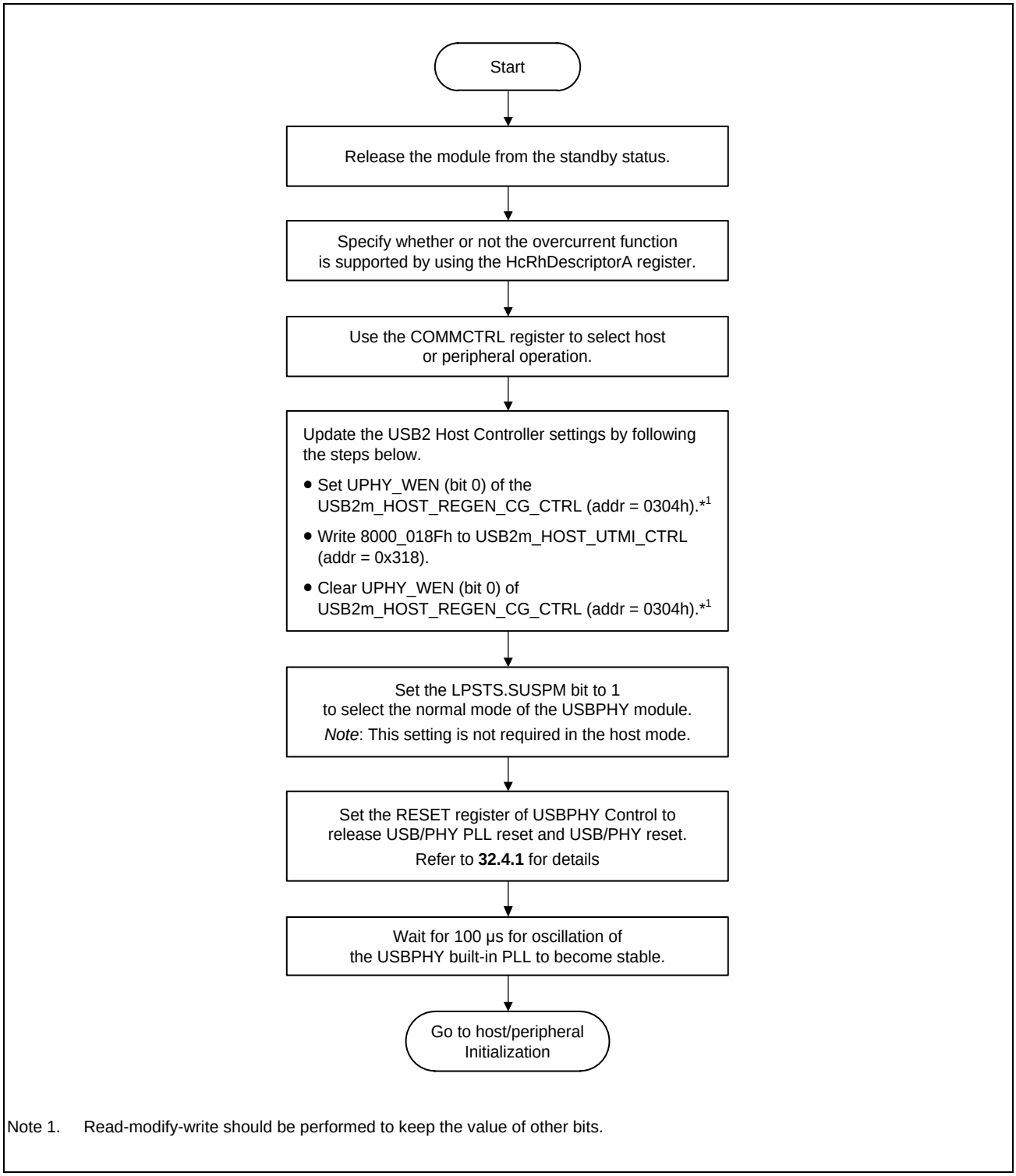
1. When the OVRCURN pin is kept asserted (0b) for the overcurrent detection time, this module determines the occurrence of overcurrent, and then de-asserts the VBUSEN pin (0b).
2. After the overcurrent status has been resolved, and de-assertion of the OVRCURN pin (1b) is confirmed, 1b is written to the Port Power bit described in **6.5.2.8.3 Port Power (VBUS) control specifications** to turn on the Port Power (Vbus).

Remark Before the Port Power bit is set by firmware, be sure to check that the OVRCURN pin has been deasserted.

6.5.2.9 Procedure for Setting this Module

6.5.2.9.1 Host/Peripheral Common Setting Sequence

The following shows the necessary sequence common to both host and peripheral modes.



Note 1. Read-modify-write should be performed to keep the value of other bits.

Figure 6.5-2 Sequence Common to Both Host and Peripheral Modes

6.5.2.9.2 Initialization Sequence

The following shows the initialization sequence in the Host mode

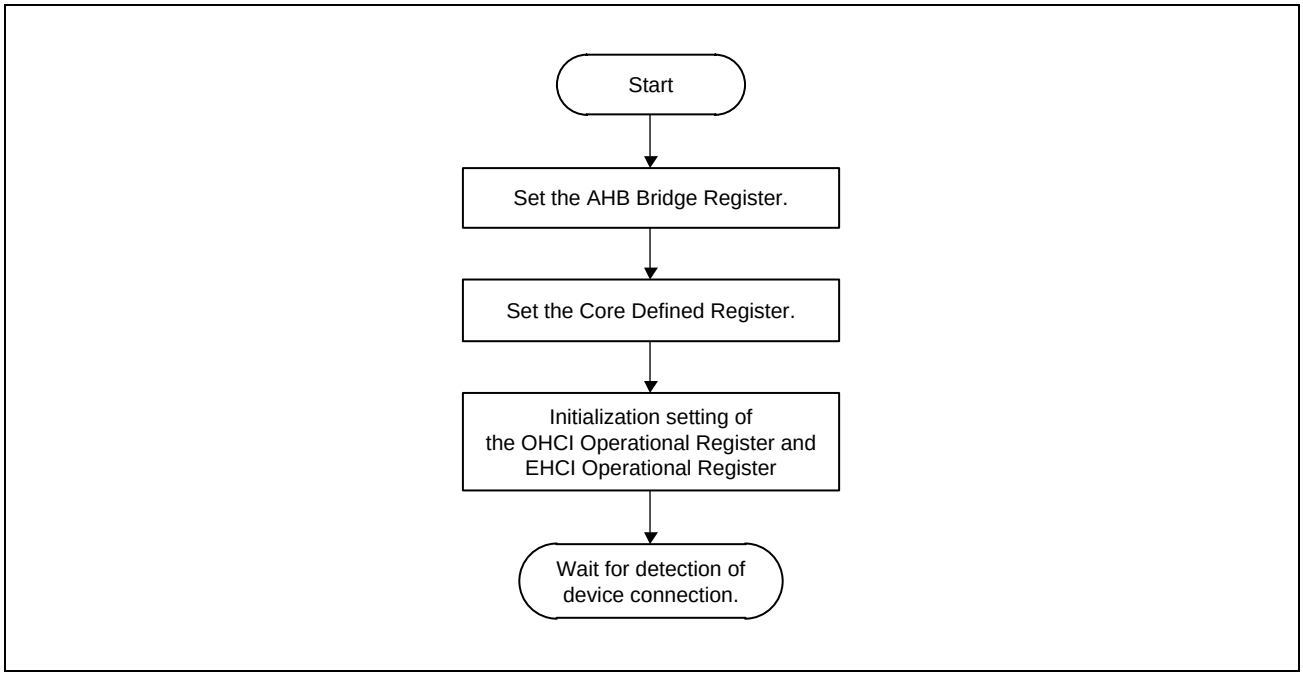


Figure 6.5-3 Initialization Sequence

6.5.2.9.3 Flow of Error Handling

While operating this module, if the operation falls into an abnormal state and recovery to the normal flow seems difficult, perform the following steps for reset.

[When an abnormality occurs while EHCI is running]

1. Write 1b to the HCRESET bit (bit 1) of the EHCI Operational Register USBCMD Register (offset: 120h) to execute EHCI software reset.
2. Re-initialize the EHCI Operational Register.

[When an abnormality occurs while OHCI is running]

1. Write 1b to the HCR bit (bit 0) of the OHCI Operational Register HcCommandStatus Register (offset: 008h) to execute OHCI software reset.
2. Re-initialize the OHCI Operational Register.

6.5.3 USB2.0 Function Controller

6.5.3.1 Overview

This LSI has two channels of USB 2.0 host/function module. Switching between the host and peripheral functions for each channel is possible by setting the UCOM register.

The setting for battery charging is handled by the host controller even if the peripheral controller is selected.

This section describes the peripheral controller. For details on the host/peripheral common circuit and battery charging, see **6.5.2 USB2.0 Host Controller**.

This module is a universal serial bus (USB) controller that has peripheral functions.

This module supports high-speed and full-speed transfer defined by the Universal Serial Bus Specification Revision 2.0.

This module supports all transfer types defined in the USB Specification. This module incorporates 8 Kbytes of buffer memory for data transfer, and can use a maximum of 10 pipes. You can assign any endpoint number to any pipe other than pipe 0 in conformity to the peripheral equipment or system to communicate with.

6.5.3.1.1 Features

Peripheral controller supporting high-speed USB

- On-chip peripheral USB controller

Support of all types of USB transfer

- Supporting all types of USB transfer, including isochronous transfer
- Control transfer
- Bulk transfer
- Interrupt transfer (high-bandwidth transfers not supported)
- Isochronous transfer (high-bandwidth transfers not supported)

Bus interface

- Includes a two-channel DMA interface

Pipe configuration

- 8 Kbytes of buffer memory for USB communications for each channel
- Up to 10 pipes (including the default control pipe) selectable for each channel
- Programmable pipe configuration
- Any endpoint number assignable to pipes other than pipe 0

Table 6.5-10 Pipe Settings

PIPE	Transfer Type	Double Buffer	Continuous Transfer Mode	Buffer Size
PIPE0	Control	—	—	Fixed to 64 bytes/ 256 bytes (CNTMD = 1)
PIPE1	Iso/Bulk	✓	✓ (Bulk only)	Up to 2 Kbytes
PIPE2	Iso/Bulk	✓	✓ (Bulk only)	Up to 2 Kbytes
PIPE3	Bulk	✓	✓	Up to 2 Kbytes
PIPE4	Bulk	✓	✓	Up to 2 Kbytes
PIPE5	Bulk	✓	✓	Up to 2 Kbytes
PIPE6	Int	—	—	Fixed to 64 bytes
PIPE7	Int	—	—	Fixed to 64 bytes
PIPE8	Int	—	—	Fixed to 64 bytes
PIPE9	Int	—	—	Fixed to 64 bytes

Features of peripheral functions

- Support of high-speed transfer (at 480 Mbps) and full-speed transfer (at 12 Mbps)
- Automatic recognition of high-speed or full-speed operation based on automatic response to the reset handshake
- Control transfer stage monitoring function
- Device state monitoring function
- Automatic response to SET_ADDRESS request
- NAK response interrupt (NRDY)
- SOF interpolation

Features of DMA transfer

- DMA transaction mode:
  Fetching in both register and link modes are supported.
- Interrupt:
  Level is supported.
- Transfer size:
  A transfer size from 1 to 128 bytes can be selected separately for the transfer source and transfer destination.
- Skip (scatter/gather) function:
  The access size and skip size can be specified separately for the transfer source and destination.
- Suspend function:
  A running DMA transaction can be suspended temporarily.
- Interval function:
  The interval of DMA transfers can be specified to control the bus occupancy.

Other functions

- Byte endian swap function to support both big endian and little endian as data formats (when using only CFIFO)
- Transfer ending function using a transaction counter
- SOF pulse output function
- BRDY interrupt event notification timing change function (BFRE)
- Function (SHTNAK) to set NAK in the response PID when transfer ends
- Support of the Link Power Management (LPM) ECN, making available a new low-power-consumption state (L1 state)

6.5.3.1.2 Overview of functions

(1) Automatic recognition of USB transfer speed

This module automatically recognizes USB transfer speed.

(a) Methods of FIFO buffer memory access

This module supports the two types of access described below to the FIFO buffer memory for USB data transfer.

a-1) Access from the CPU

Specify a FIFO port address, and then write data to or read data from the FIFO buffer memory.

a-2) Direct memory access (DMA)

Selecting a pipe window and setting the DMA control registers enables writing data to or reading data from the FIFO buffer memory.

(2) USB event

This module notifies the event in USB operation by issuing an interrupt.

You can specify whether to enable notification by interrupt for individual interrupt types and sources through software settings.

(3) USB data transfer

This module performs all types of USB data transfer: control transfer, bulk transfer, interrupt transfer, and isochronous transfer. The following pipe resources are available for individual transfer types:

a) One pipe dedicated to control transfer

b) Four pipes dedicated to interrupt transfer

c) Three pipes dedicated to bulk transfer

d) Two pipes selectively used for bulk or isochronous transfer

For each pipe, specify the settings, including transfer type, endpoint number, and maximum packet size, required for USB transfer according to the system.

This module can incorporate up to 8 Kbytes of buffer memory. For the pipes dedicated to bulk transfer and those selectively used for bulk transfer or isochronous transfer, allocate buffer memory and specify a buffer operating mode and other necessary settings according to the system. Setting the buffer operating mode enables high-speed data transfers with fewer interrupts to be performed by using double-buffering and continuous transfer of data packets.

(4) SOF pulse output function

This module has a function to output an SOF pulse to indicate the timing of SOF packet transmission. This module asserts a SOF pulse output signal when an SOF packet is received. This module outputs pulses at regular intervals based on an SOF interpolation timer even when an SOF packet is damaged.

6.5.3.1.3 Restrictions and notes

(1) Restrictions

(a) Restrictions on the USB specifications

There is no support for the following USB 2.0 specification.

- High-bandwidth transfer is not supported.

(b) Restrictions on DMA master

- DAD = 1 (destination address fixed) and skip transfer cannot be used on the destination side. When forwarding by such setting, movement is unsettled. Such transmission should not be done.
- SAD = 1 (source address fixed), and skip transfer cannot be used on the source side. When forwarding by such setting, movement is unsettled. Such transmission should not be done.
- DAD = 1 (destination address fixed), and beat-unaligned transfer cannot be used on the destination side. When forwarding by such setting, movement is unsettled. Such transmission should not be done.
- SAD = 1 (source address fixed), and beat-unaligned transfer cannot be used on the source side. When forwarding by such setting, movement is unsettled. Such transmission should not be done.
- When REQD =1, SBE = 1 (sweep mode) and compulsion discharge function cannot be used.

(2) Notes

(a) DMA transfer and avoiding problem with interrupt signal in DMA Master configuration

a-1) Overview

A USBFDMAmn interrupt (m, n = 0, 1) might occur before the last data of DMA transaction is written to the write-target device.

a-2) Workaround

A workaround for the above problem is described below.

HPROT is set as non bufferable.

According to the DMA mode, set 0 in the DPR[2] bit in the CHEXT_n register or the LDPR[2] bit in the DCTRL register, set the HPROT signal as non bufferable, and then perform the DMA transaction.

If all transfers are set to non bufferable, transfer efficiency might fall.

In such cases, set the most part of transaction as bufferable and perform a transfer, and then perform the last transfer by setting a register set or descriptor as non bufferable.

(b) Setting for battery charging when the peripheral controller is selected

The setting for battery charging is handled by the host controller even if the peripheral controller is selected.

6.5.3.2 Function Controller Registers

Table 6.5-11 Register Base Addresses

Unit Name	Base Address Name	Base Address
USB20	<USB20_func_base>	0_1582_0000h (5582_0000h*1, 4582_0000h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

The prefix (USB20_FUNC_) of the register names is omitted in the register descriptions and the field descriptions in this section.

Remark USB21 does not have the function controller.

6.5.3.2.1 How to read the register table

(1) Bit number:

(2) State after reset: Initial state of the register that occurs immediately after a reset “Power on Reset” indicates the initial state at power-on reset.
The state after USB reset is the initial state of the register that occurs when this module detects a USB bus reset.
Significant points regarding reset operation are indicated in notes.
“—” indicates that a user’s setting is retained without this module operation having been performed. “X” indicates that the value is undefined.

(3) Access condition: The condition to be met when this module accesses the register for an operation.

R: Reading only
W: Writing only
RW: Reading or writing
R(0): 0-reading only
W(1): 1-writing only

(4) Name: Bit symbol and bit name

(5) Function: Description of functions.

<Example of description>

(1) Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	A bit	B bit	C bit	—	—	—	—	—	—	—	—	—	—	—	—
(2) Initial Value	X	0	0	0												
R/W	X	0	—	—												

Bit	Bit Name	Initial Value	R/W	Description
15	—			Nothing is assigned. Fix this bit to 0b.
14	A bit		RW	AAA enable 0: Disables operation 1: Enables operation
13	B bit		R	BBB operation 0: Outputs low-level signal 1: Outputs high-level signal
12	C bit		R(0)/ W(1)	CCC control 0: ..... 1: .....
	(4)		(3)	(5)

6.5.3.2.2 List of Registers

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
System Configuration Control Register 0	USB20_FUNC_SYSCFG0	0020h	0000h	16
System Configuration Control Register 1	USB20_FUNC_SYSCFG1	000Fh	0002h	16
System Configuration Status Register	USB20_FUNC_SYSSTS0	0000h	0004h	16
Reserve	-	-	0006h to 0007h	-
Device State Control Register 0	USB20_FUNC_DVSTCTR0	0000h	0008h	16
Reserve	-	-	000Ah to 000Bh	-
USB Test Mode Register	USB20_FUNC_TESTMODE	0000h	000Ch	16
Reserve	-	-	000Eh to 0013h	-
CFIFO Port Register	USB20_FUNC_CFIFO	0000_0000h	0014h	32/16/8
Reserve	-	-	0018h to 001Fh	-
CFIFO Port Select Register	USB20_FUNC_CFIFOSEL	0000h	0020h	16
CFIFO Port Control Register	USB20_FUNC_CFIFOCTR	0000h	0022h	16
Reserve	-	-	0024h to 0027h	-
D0FIFO Port Select Register	USB20_FUNC_D0FIFOSEL	0000h	0028h	16
D0FIFO Port Control Register	USB20_FUNC_D0FIFOCTR	0000h	002Ah	16
D1FIFO Port Select Register	USB20_FUNC_D1FIFOSEL	0000h	002Ch	16
D1FIFO Port Control Register	USB20_FUNC_D1FIFOCTR	0000h	002Eh	16
Interrupt Enable Register 0	USB20_FUNC_INTENB0	0000h	0030h	16
Reserve	-	-	0032h to 0035h	-
BRDY Interrupt Enable Register	USB20_FUNC_BRDYENB	0000h	0036h	16
NRDY Interrupt Enable Register	USB20_FUNC_NRDYENB	0000h	0038h	16
BEMP Interrupt Enable Register	USB20_FUNC_BEMPENB	0000h	003Ah	16
SOF Pin Configuration Register	USB20_FUNC_SOFCFG	0000h	003Ch	16
Reserve	-	-	003Eh to 003Fh	-
Interrupt Status Register 0	USB20_FUNC_INTSTS0	0000h	0040h	16
Reserve	-	-	0042h to 0045h	-
BRDY Interrupt Status Register	USB20_FUNC_BRDYSTS	0000h	0046h	16
NRDY Interrupt Status Register	USB20_FUNC_NRDYSTS	0000h	0048h	16
BEMP Interrupt Status Register	USB20_FUNC_BEMPSTS	0000h	004Ah	16
Frame Number Register	USB20_FUNC_FRMNUM	0000h	004Ch	16
Micro Frame Number Register	USB20_FUNC_UFRMNUM	0000h	004Eh	16
USB Address Register	USB20_FUNC_USBADDR	0000h	0050h	16
Reserve	-	-	0052h to 0053h	-
USB Request Type Register	USB20_FUNC_USBREQ	0000h	0054h	16
USB Request Value Register	USB20_FUNC_USBVAL	0000h	0056h	16
USB Request Index Register	USB20_FUNC_USBINDX	0000h	0058h	16
USB Request Length Register	USB20_FUNC_USBLENG	0000h	005Ah	16
DCP Configuration Register	USB20_FUNC_DCPCFG	0000h	005Ch	16
DCP Max. Packet Size Register	USB20_FUNC_DCPMAXP	0040h	005Eh	16
DCP Control Register	USB20_FUNC_DCPCTR	0000h	0060h	16

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	0062h to 0063h	-
Pipe Window Select Register	USB20_FUNC_PIPESEL	0000h	0064h	16
Reserve	-	-	0066h to 0067h	-
Pipe Configuration Register	USB20_FUNC_PIPECFG	0000h	0068h	16
Pipe Buffer Setting Register	USB20_FUNC_PIPEBUF	0000h	006Ah	16
Pipe Maximum Packet Size Register	USB20_FUNC_PIPEMAXP	0000h	006Ch	16
Pipe Cycle Control Register	USB20_FUNC_PIPEPERI	0000h	006Eh	16
PIPE1 Control Register	USB20_FUNC_PIPE1CTR	0000h	0070h	16
PIPE2 Control Register	USB20_FUNC_PIPE2CTR	0000h	0072h	16
PIPE3 Control Register	USB20_FUNC_PIPE3CTR	0000h	0074h	16
PIPE4 Control Register	USB20_FUNC_PIPE4CTR	0000h	0076h	16
PIPE5 Control Register	USB20_FUNC_PIPE5CTR	0000h	0078h	16
PIPE6 Control Register	USB20_FUNC_PIPE6CTR	0000h	007Ah	16
PIPE7 Control Register	USB20_FUNC_PIPE7CTR	0000h	007Ch	16
PIPE8 Control Register	USB20_FUNC_PIPE8CTR	0000h	007Eh	16
PIPE9 Control Register	USB20_FUNC_PIPE9CTR	0000h	0080h	16
Reserve	-	-	0082h to 008Fh	-
PIPE1 Transaction Counter Enable Register	USB20_FUNC_PIPE1TRE	0000h	0090h	16
PIPE1 Transaction Counter Register	USB20_FUNC_PIPE1TRN	0000h	0092h	16
PIPE2 Transaction Counter Enable Register	USB20_FUNC_PIPE2TRE	0000h	0094h	16
PIPE2 Transaction Counter Register	USB20_FUNC_PIPE2TRN	0000h	0096h	16
PIPE3 Transaction Counter Enable Register	USB20_FUNC_PIPE3TRE	0000h	0098h	16
PIPE3 Transaction Counter Register	USB20_FUNC_PIPE3TRN	0000h	009Ah	16
PIPE4 Transaction Counter Enable Register	USB20_FUNC_PIPE4TRE	0000h	009Ch	16
PIPE4 Transaction Counter Register	USB20_FUNC_PIPE4TRN	0000h	009Eh	16
PIPE5 Transaction Counter Enable Register	USB20_FUNC_PIPE5TRE	0000h	00A0h	16
PIPE5 Transaction Counter Register	USB20_FUNC_PIPE5TRN	0000h	00A2h	16
Reserve	-	-	00A4h to 00FFh	-
Low Power Control Register	USB20_FUNC_LPCTRL	0000h	0100h	16
Low Power Status Register	USB20_FUNC_LPSTS	0000h	0102h	16
PHY Function Control Register	USB20_FUNC_PHYFUNCTR	0000h	0104h	16
Reserve	-	-	0106h to 0109h	-
PHY_OTG Control Register	USB20_FUNC_PHYOTGCTR	0600h	010Ah	16
Reserve	-	-	010Ch to 0143h	-
Peripheral L1 Control Register 1	USB20_FUNC_PL1CTRL1	0000h	0144h	16
Peripheral L1 Control Register 2	USB20_FUNC_PL1CTRL2	0000h	0146h	16
Reserve	-	-	0148h to 03FFh	-
Next0 Source Address Register ch0	USB20_FUNC_N0SA_0	0000_0000h	0400h	32
Next0 Destination Address Register ch0	USB20_FUNC_N0DA_0	0000_0000h	0404h	32
Next0 Transaction Byte Register ch0	USB20_FUNC_N0TB_0	0000_0000h	0408h	32
Next1 Source Address Register ch0	USB20_FUNC_N1SA_0	0000_0000h	040Ch	32
Next1 Destination Address Register ch0	USB20_FUNC_N1DA_0	0000_0000h	0410h	32
Next1 Transaction Byte Register ch0	USB20_FUNC_N1TB_0	0000_0000h	0414h	32
Current Source Address Register ch0	USB20_FUNC_CRSA_0	0000_0000h	0418h	32
Current Destination Address Register ch0	USB20_FUNC_CRDA_0	0000_0000h	041Ch	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Current Transaction Byte Register ch0	USB20_FUNC_CRTB_0	0000_0000h	0420h	32
Channel Status Register ch0	USB20_FUNC_CHSTAT_0	0000_0000h	0424h	32
Channel Control Register ch0	USB20_FUNC_CHCTRL_0	0000_0000h	0428h	32
Channel Configuration Register ch0	USB20_FUNC_CHCFG_0	0000_0000h	042Ch	32
Channel Interval Register ch0	USB20_FUNC_CHITVL_0	0000_0000h	0430h	32
Channel Extension Register ch0	USB20_FUNC_CHEXT_0	0000_0000h	0434h	32
Next Link Address Register ch0	USB20_FUNC_NXLA_0	0000_0000h	0438h	32
Current Link Address Register ch0	USB20_FUNC_CRLA_0	0000_0000h	043Ch	32
Next0 Source Address Register ch1	USB20_FUNC_N0SA_1	0000_0000h	0440h	32
Next0 Destination Address Register ch1	USB20_FUNC_N0DA_1	0000_0000h	0444h	32
Next0 Transaction Byte Register ch1	USB20_FUNC_N0TB_1	0000_0000h	0448h	32
Next1 Source Address Register ch1	USB20_FUNC_N1SA_1	0000_0000h	044Ch	32
Next1 Destination Address Register ch1	USB20_FUNC_N1DA_1	0000_0000h	0450h	32
Next1 Transaction Byte Register ch1	USB20_FUNC_N1TB_1	0000_0000h	0454h	32
Current Source Address Register ch1	USB20_FUNC_CRSA_1	0000_0000h	0458h	32
Current Destination Address Register ch1	USB20_FUNC_CRDA_1	0000_0000h	045Ch	32
Current Transaction Byte Register ch1	USB20_FUNC_CRTB_1	0000_0000h	0460h	32
Channel Status Register ch1	USB20_FUNC_CHSTAT_1	0000_0000h	0464h	32
Channel Control Register ch1	USB20_FUNC_CHCTRL_1	0000_0000h	0468h	32
Channel Configuration Register ch1	USB20_FUNC_CHCFG_1	0000_0000h	046Ch	32
Channel Interval Register ch1	USB20_FUNC_CHITVL_1	0000_0000h	0470h	32
Channel Extension Register ch1	USB20_FUNC_CHEXT_1	0000_0000h	0474h	32
Next Link Address Register ch1	USB20_FUNC_NXLA_1	0000_0000h	0478h	32
Current Link Address Register ch1	USB20_FUNC_CRLA_1	0000_0000h	047Ch	32
Reserve	-	-	0458h to 05FFh	-
Source Continuous Register ch0	USB20_FUNC_SCNT_0	0000_0000h	0600h	32
Source Skip Register ch0	USB20_FUNC_SSKP_0	0000_0000h	0604h	32
Destination Continuous Register ch0	USB20_FUNC_DCNT_0	0000_0000h	0608h	32
Destination Skip Register ch0	USB20_FUNC_DSKP_0	0000_0000h	060Ch	32
Reserve	-	-	0610h to 061Fh	-
Source Continuous Register ch1	USB20_FUNC_SCNT_1	0000_0000h	0620h	32
Source Skip Register ch1	USB20_FUNC_SSKP_1	0000_0000h	0624h	32
Destination Continuous Register ch1	USB20_FUNC_DCNT_1	0000_0000h	0628h	32
Destination Skip Register ch1	USB20_FUNC_DSKP_1	0000_0000h	062Ch	32
Reserve	-	-	0630h to 06FFh	-
DMA Control Register	USB20_FUNC_DCTRL	0000_0000h	0700h	32
Descriptor Interval Register	USB20_FUNC_DSCITVL	0000_0000h	0704h	32
Reserve	-	-	0708h to 070Fh	-
DMA Status EN Register	USB20_FUNC_DSTAT_EN	0000_0000h	0710h	32
DMA Status ER Register	USB20_FUNC_DSTAT_ER	0000_0000h	0714h	32
DMA Status END Register	USB20_FUNC_DSTAT_END	0000_0000h	0718h	32
DMA Status TC Register	USB20_FUNC_DSTAT_TC	0000_0000h	071Ch	32
DMA Status SUS Register	USB20_FUNC_DSTAT_SUS	0000_0000h	0720h	32

6.5.3.2.3 System Configuration Control Registers

(1) System Configuration Control Register 0 (USB20_FUNC_SYSCFG0)

Access Size : 16 bits
Address : <USB20_func_base> + 0000h
Initial Value : 0020h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CNEN	HSE	-	DRPD	DPRPU	-	-	-	USBE
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	RW	RW	R	RW	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
8	CNEN	0h	RW	This bit prohibits or enables single-ended receiver operation. 0b: Single-ended receiver operation prohibited 1b: Single-ended receiver operation enabled
7	HSE	0h	RW	This bit prohibits or enables High-Speed operation. 0b: High-Speed operation prohibited (Full-Speed) 1b: High-Speed operation enable (The controller detects the communication speed.)
6	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
5	DRPD	1h	RW	D+/D- line resistor control Set this bit to 0 to use this module. For details, see **Table 6.5-12 Control of USB Data Bus Resistors**.
4	DPRPU	0h	RW	D+ line resistor control This bit prohibits or enables D+ line pull-up for the peripheral controller function. For details, see **Table 6.5-12 Control of USB Data Bus Resistors**. 0b: Pull Up prohibited 1b: Pull Up enabled
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
0	USBE	0h	RW	USB block operation prohibited This bit prohibits or enables USB block operation. 0b: USB block operation prohibited 1b: USB block operation enabled

Note 1. Data can be written to and read from this register even while the UTMI+PHY clock is stopped. However, if a value is set while the UTMI+PHY clock is stopped, the corresponding function is enabled after the UTMI+PHY clock starts oscillating.

(a) Single-ended receiver operation enable (CNEN) bit

Setting this bit enables the single-ended receiver to operate. This bit is intended to prevent damage by inrush current that can be caused when the single-ended receiver in unattached status goes floating. This bit also allows the LNST bit to be referenced.

Set this bit when VBUS is detected as the result of a VBUS interrupt. Clear this bit when VBUS is removed.

(b) High-speed operation enable (HSE) bit

Setting this bit enables the high-speed operation. When this bit is 1, this module performs a high-speed or full-speed operation according to the result of reset handshake.

- When the HSE bit is 0, this module performs a full-speed operation.
- When the HSE bit is 1, this module executes the reset handshake protocol, and then automatically performs a high-speed or full-speed operation according to the result of reset handshake.

Rewriting the value of this bit must be done when the DPRPU bit is 0.

(c) D+/D- line resistor control (DRPD or DPRPU) bit

Table 6.5-12 shows available settings of the resistors for the USB data bus. Use the DPRPU bit to select the USB data bus resistors.

Table 6.5-12 Control of USB Data Bus Resistors

Setting		Control of USB Data Bus Resistors		
DRPD	DPRPU	D- Line	D+ Line	Remarks
0	0	Open	Open	
0	1	Open	Pull-Up	Specify the settings as shown in the left.
1	0	Pull-Down	Pull-Down	Initial state (When power on reset is canceled)
1	1	Pull-Down	Pull-Up	Setting prohibited

(d) D+ pull-up resistor control (DPRPU) bit

Setting this bit enables this module to notify the USB host of attaching by pulling up the D+ line voltage to 3.3 V. Clearing this bit enables this module to let the USB host know that the device has been detached by stopping pulling up the D+ line voltage.

(e) USB block operation enable (USBE) bit

This bit enables or disables the operation of the USB block of this module.

When this bit is changed from 1 to 0, this module initializes the bits shown in **Table 6.5-13**.

Table 6.5-13 Register Bits That Are Initialized by Writing 0 to the USBE Bit

Register Name	Bit Name
SYSSTS0	LNST
DVSTCTR0	RHST
INTSTS0	DVSQ
USBADDR	USBADDR
USBREQ	bRequest bmRequestType
USBVAL	wValue
USBINDX	wIndex
USBLENG	wLength

Note: Changing the value of this bit must be done when the SUSPENDM bit is 1 and after the oscillation of UTMI+PHY clock starts.

(2) System Configuration Control Register 1 (USB20_FUNC_SYSCFG1)

Access Size : 16 bits
Address : <USB20_func_base> + 0002h
Initial Value : 0F0Fh

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	BWAIT[5:0]					
Initial Value	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
11 to 8	-	Fh	R	Reserved Whenever it is read, 1b is read. The write value should always be 1b.
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
5 to 0	BWAIT[5:0]	Fh	RW	CPU bus access wait specification These bits specify the number of wait cycles for the access to this module. 00_0000b: 0 wait cycles (2 access cycles) : 00_0010b: 2 wait cycles (4 access cycles) : 00_0100b: 4 wait cycles (6 access cycles) : 00_1111b: 15 wait cycles (17 access cycles) (default) : 11_1111b: 63 wait cycles (65 access cycles)

(a) CPU bus access wait specification (BWAIT) bits

These bits specify the wait cycles for the access to the HPB.

The following restriction is placed on the cycle of the access to the registers at address 04h or after of this module:
Restriction on wait cycle: The cycle of continuous accesses to registers of this module must be at least 67 ns.

To comply with this restriction, you must control the number of wait cycles with the internal bus clock (P1ϕ) frequency.

The default of wait cycles is 17 clock cycles (maximum limit). Select an optimum setting.

This setting is also applied to accesses to FIFO port registers. The maximum speeds of accesses to FIFO ports are as follows:

MBW = 10 (32-bit access width): Max 60 Mbytes/sec
MBW = 01 (16-bit access width): Max 30 Mbytes/sec
MBW = 00 (8-bit access width): Max 15 Mbytes/sec

6.5.3.2.4 System Configuration Status

(1) System Configuration Status Register (USB20_FUNC_SYSSTS0)

Access Size : 16 bits
Address : <USB20_func_base> + 0004h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LNST[1:0]	
Initial Value	-	-	0	0	0	0	0	0	0	0	0	0	0	–	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 2	-	See above	R	Reserved Whenever it is read, 0b or "-" is read. The written value will be ignored.
1, 0	LNST[1:0]	0h	R	USB Line status monitor The USB line status is displayed. Note: See the detailed description.

(a) Line status monitor (LNST) bits

Table 6.5-14 shows the line status of the USB data bus of this module. This module monitors the line status (status of the D+ and D- lines) of the USB data bus in the LNST bits of the SYSSTS0 register.

Referencing the LNST bits must be done only after the USBE bit is set and attaching is performed (the DPRPU bit is set).

Table 6.5-14 Line Status of USB Data Bus

LNST [1]	LNST [0]	Full-Speed operation	High-Speed operation	Chirp operation
0	0	SE0	Squelch	Squelch
0	1	J State	Unsquelch	Chirp J
1	0	K State	Invalid	Chirp K
1	1	SE1	Invalid	Invalid

Note: Chirp: State in which high-speed operation is enabled (HSE = 1) and the reset handshake protocol is being executed
Squelch: SE0 or idle state
Unsquelch: High-speed J or high-speed K state
Chirp J: Chirp J State
Chirp K: Chirp K State

6.5.3.2.5 USB Signal Control Registers

(1) Device State Control Register 0 (USB20_FUNC_DVSTCTR0)

Access Size : 16 bits
Address : <USB20_func_base> + 0008h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	WKUP	-	-	-	-	-	RHST[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	0	-	-	-	-	-	-	-	-
R/W	RW	R	R	R	R	R	R	RW1	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	-	0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
14 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
8	WKUP	0h	RW1	Remote wakeup output This bit prohibits or enables Remote wakeup (resume signal output). 0b: Remote wakeup signal is not output. 1b: Remote wakeup signal is output.
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
2 to 0	RHST[2:0]	0h	R	Reset handshake This bit indicates the reset handshake status. Note: See the detailed description.

(a) Remote wakeup (resume signal output) enable (WKUP) bit

When this bit is set, this module outputs the remote wakeup signal to the USB.

This module manages the time of remote wakeup signal output. When the WKUP bit is set, this module outputs the K state for 10 ms, and then clears the WKUP bit.

The USB Specification requires the USB idle state to be retained for at least 5 ms before the remote wakeup signal is sent. Therefore, even if the WKUP bit is set immediately after the suspended state is detected, this module waits for 2 ms, and then outputs the K state.

Writing 1 to the WKUP bit must be done only when the device is in the suspended state (DVSQ = 1xx) and remote wakeup is allowed by the USB host.

When setting the WKUP bit, do not stop the internal clock even if the device is in the suspended state. (Write 1 to the WKUP bit when the SUSPM bit is 1.)

When the WKUP bit is set at a transition to the L1 state, this module outputs the K state for 50 µs, and then clears the WKUP bit. In the L1 state, setting the WKUP bit must be done only when the DVSQ[4] bit is 1.

(b) Reset handshake status (RHST) bits

This module outputs the result of reset handshake to this bit. **Table 6.5-15** lists the results of reset handshake.

Table 6.5-15 Reset Handshake Status

Bus State	Value of RHST Bit
Powered or disconnected state	000
Reset handshake in process	100
Full-speed connection	010
High-speed connection	011

If the HSE bit is 1, the RHST bits indicate 100 when this module detects a USB bus reset. Then, after this module has output Chirp K, these bits indicate 011 when this module detects Chirp JK from the USB host three times. If the status is not fixed to High-Speed within 2.5 ms after Chirp K is output, these bits indicate 010.

If the HSE bit is 0, the RHST bits indicate 010 when this module detects a bus reset.

After this module has detected a USB reset, a DVST interrupt occurs when the value of the RHST bits is fixed to 010 or 011.

6.5.3.2.6 Test Mode Register

(1) USB Test Mode Register (USB20_FUNC_TESTMODE)

Access Size : 16 bits
Address : <USB20_func_base> + 000Ch
Initial Value : 0100h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	UTST[3:0]			
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	-	10h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
3 to 0	UTST[3:0]	0h	RW	Test mode See the detailed description.

(a) Test mode (UTST) bits

When a value is written to these bits, this module outputs a USB test signal during high-speed operation.

Table 6.5-16 lists the test modes of this module.

Table 6.5-16 List of test mode operation

Test mode	Value of UTST Bits
Normal operation	0000
Test_J	0001
Test_K	0010
Test_SE0_NAK	0011
Test_Packet	0100
Test_Force_Enable	—
Reserved	0101 to 0111

Write a value to these bits according to the SetFeature request sent from the USB host during high-speed communication. When these bits contain a value from 0001 to 0100, this module does not enter the suspended state.

To perform a normal USB communication after setting a test mode, perform power on reset.

6.5.3.2.7 FIFO Port Registers

(1) CFIFO Port Register (USB20_FUNC_CFIFO)

Access Size : 32/16/8 bits
Address : <USB20_func_base> + 0014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIFOPORT[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FIFOPORT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	FIFOPORT [31:0]	0h	RW	FIFO port These bits are accessed to read received data from the FIFO buffer or to write send data to the FIFO buffer.

(a) FIFO port control bits (for FIFOPORT)

The send/receive buffer memory of this module has a FIFO structure (FIFO buffer). Use FIFO port registers to access the FIFO buffer. The FIFO port consists of the port register (CFIFO) to read data from and write data to the FIFO buffer, the register (CFIFOSEL) to select the pipe to be allocated to the FIFO port, and the control register (CFIFOCTR).

Individual FIFO ports have the following features:

- The CFIFO port must be used to access the FIFO buffer through the DCP.
- When functions specific to FIFO ports are used, the pipe number (selected pipe) specified in the CURPIPE bits cannot be changed.
- The registers configured for a FIFO port do not affect any other FIFO ports.
- The FIFO buffer memory can be accessed by either the CPU or SIE. Access by the CPU is not possible while the SIE has the right of access to the FIFO buffer memory.

FIFO port bits (CFIFO)

When one of these registers is accessed, this module accesses the FIFO buffer allocated to the pipe number specified in the CURPIPE bits in the corresponding pipe select register (CFIFOSEL).

These registers can be accessed only when the FRDY bit of the respective control registers (CFIFOCTR) is 1 (or when the UCL_Dx_DREQ output is asserted by this module).

The valid bits of these registers vary depending on the values of the NBW and BIGEND bits. The valid bits are shown in **Table 6.5-17** to **Table 6.5-19**.

Table 6.5-17 Endian Operation in 32-Bit Access (When MBW = 10)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	N + 3 address	N + 2 address	N + 1 address	N + 0 address
1	N + 0 address	N + 1 address	N + 2 address	N + 3 address

Table 6.5-18 Endian Operation in 16-Bit Access (When MBW = 01)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	Writing: invalid Reading: prohibited*1		even-numbered address	odd-numbered address
1	even-numbered address	odd-numbered address	Writing: invalid Reading: prohibited*1	

Table 6.5-19 Endian Operation in 8-Bit Access (When MBW = 00)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	Writing: invalid Reading: prohibited*1			Writing: valid Reading: valid
1	Writing: valid Reading: valid	Writing: invalid Reading: prohibited*1		

Note 1. Reading words or bytes from an invalid register is prohibited.

(2) CFIFO Port Select Register (USB20_FUNC_CFIFOSEL)

Access Size : 16 bits
Address : <USB20_func_base> + 0020h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	-	-	MBW[1:0]		-	BIGEND	-	-	ISEL	-	CURPIPE[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	R0W	R	R	RW	RW	R	RW	R	R	RW	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15	RCNT	0h	RW	Read count mode This bit specifies DTLN read mode for the CFIFOCTR register. 0b: Clears the DTLN bits when all received data is read. 1b: Decrements the value of the DTLN bits each time received data is read.
14	REW	0h	R0W	The read value is always 0b. Buffer pointer rewind Set this bit to 1b to rewind the buffer pointer. 0b: Does not rewind the buffer pointer. 1b: Rewinds the buffer pointer.
13, 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
11, 10	MBW[1:0]	0h	RW	CFIFO port access bit width This bit specifies the bit width for access to the CFIFO port. 00b: 8-bit width 01b: 16-bit width 10b: 32-bit width 11b: Setting prohibited
9	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
8	BIGEND	0h	RW	FIFO port byte endian control This bit specifies the byte endian of the CFIFO port. 0b: Little endian 1b: Big endian
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
5	ISEL	0h	RW	FIFO port access direction with DCP selected This bit specifies the FIFO port access direction when DCP is selected for the CURPIPE bits. 0b: Selects reading of buffer memory. 1b: Selects writing of buffer memory.
4	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
3 to 0	CURPIPE[3:0]	0h	RW	FIFO port access pipe specification This bit specifies the pipe number used for accessing the CFIFO port. 0000b: DCP 0001b: PIPE1 0010b: PIPE2 ↓ 1000b: PIPE8 1001b: PIPE9 ↓ 1110b: PIPE14 1111b: PIPE15

(a) Read count mode (RCNT) bit

When this bit is 0, this module clears the DTLN bits in the CFIFOCTR register when all received data has been read from the FIFO buffer allocated to the pipe (selected pipe) specified in the CURPIPE bits (or, in the case of double-buffer configuration, when all relieved data has been read from one buffer).

With this bit is 1, this module decrements the value of the DTLN bits in the CFIFOCTR register each time received data is read from the FIFO buffer allocated to the specified pipe.

(b) Buffer pointer rewind (REW) bit

When this bit is set during data reading from the FIFO buffer while the selected pipe is in the receiving direction, reading can be restarted from the first data in the FIFO buffer (or, in the case of a double-buffer configuration, rereading can be started from the first data in the FIFO buffer being read).

Do not set this bit at the same time as changing the value of the CURPIPE bits. Before setting this bit, always check that the FRDY bit is 1.

If you want to redo writing to the FIFO buffer from the first data in the FIFO buffer when the selected PIPE is in the sending direction, use the BCLR bit.

(c) CFIFO port access bit width (MBW) bits

These bits are used to specify the bit width for the access to the CFIFO port.

If you start reading after setting a value in these bits when the pipe specified in the CURPIPE bits is in the receiving direction, do not change the value of these bits until data is all read.

Also, to set a value in these bits when the pipe specified in the CURPIPE bits is in the receiving direction, temporarily change the original value of the CURPIPE bits to a different value, and then set the values of the CURPIPE and MBW bits at the same time.

For how to change the value of the CURPIPE bits, see the description of the CURPIPE bits.

When the pipe specified in the CURPIPE bits is in the sending direction, you cannot change the bit width from 8 bits to 16 bits or 32 bits or from 16 bits to 32 bits while writing to the buffer memory is in process.

Even with the 16-bit width or 32-bit width setting, you can write data also to odd bytes by using byte access control.

(d) FIFO port byte endian control (BIGEND) bit

This bit is used to specify the byte endian of the CFIFO port.

For details see **(a) FIFO port control bits (for FIFOPORT)**.

(e) FIFO port access direction with DCP selected (ISEL) bit

To change the value of this bit when the specified pipe is DCP, write a value to this bit, read the bit, and then check that the written value is the same as the read value before proceeding to the next processing.

If the value of the bit is changed in the middle of an access to the FIFO buffer, the access is held. Therefore, the same access can be resumed after these bits are restored to the original value.

Set this bit simultaneously with setting of the CURPIPE bits.

(f) FIFO port access pipe specification (CURPIPE) bits

These bits are used to specify the pipe number of the pipe through which to read or write data via the CFIFO port.

If you change the value of these bits, write a desired value to these bits, read these bits, and then check that the written value is the same as the read value before proceeding to the next processing.

If the value of these bits is changed in the middle of an access to the FIFO buffer, the access is held. Therefore, the same access can be resumed after these bits are restored to the original value.

(3) DnFIFO Port Select Register (USB20_FUNC_DnFIFOSEL) (n = 0, 1)

Access Size : 16 bits
Address : <USB20_func_base> + 0028h + n x 0004h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	DCLRM	DREQE	MBW[1:0]		-	-	-	-	-	-	CURPIPE[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	R0W	RW	RW	RW	RW	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15	RCNT	0h	RW	Read count mode This bit specifies Dx_FIFOCTR DTLN read mode. 0b: Clears the DTLN bits when all received data is read. 1b: Decrements the value of the DTLN bits each time received data is read.
14	REW	0h	R0W	The read value is always 0b. Buffer pointer rewind Set this bit to 1b to rewind the buffer pointer. 0b: Does not rewind the buffer pointer. 1b: Rewinds the buffer pointer.
13	DCLRM	0h	RW	Automatic buffer memory clear mode after reading data through the specified pipe This bit prohibits or enables automatic buffer memory clear after data is read through the specified pipe. 0b: Automatic FIFO buffer clear prohibited 1b: Automatic FIFO buffer clear enabled
12	DREQE	0h	RW	UCL_Dx_DREQ output enable This bit prohibits or enables the output of the UCL_Dx_DREQ signal. 0b: Output prohibited 1b: Output enabled
11, 10	MBW[1:0]	0h	RW	DxFIFO port access bit width This bit specifies the bit width for access to the DxFIFO port. 00b: 8-bit width 01b: 16-bit width 10b: 32-bit width 11b: Setting prohibited
9 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
3 to 0	CURPIPE[3:0]	0h	RW	FIFO port access pipe specification 0000b: No specification 0001b: PIPE1 0010b: PIPE2 ↓ 1000b: PIPE8 1001b: PIPE9

(a) Read count mode (RCNT) bit

When this bit is 0, this module clears the DTLN bits in the DxFIFOCTR register when all received data has been read from the FIFO buffer allocated to the pipe (selected pipe) specified in the CURPIPE bits (or, in the case of double-buffer configuration, when all relieved data has been read from one buffer).

With this bit is 1, this module decrements the value of the DTLN bits in the DxFIFOCTR register each time received data is read from the FIFO buffer allocated to the specified pipe.

(b) Buffer pointer rewind (REW) bit

When this bit is set during data reading from the FIFO buffer while the selected pipe is in the receiving direction, reading can be restarted from the first data in the FIFO buffer (or, in the case of a double-buffer configuration, rereading can be started from the first data in the FIFO buffer being read).

Do not set this bit at the same time as changing the value of the CURPIPE bits. Before setting this bit, always check that the FRDY bit is 1.

If you want to redo writing to the FIFO buffer from the first data in the FIFO buffer when the selected PIPE is in the sending direction, use the BCLR bit.

(c) Automatic FIFO buffer clear enable (DCLRM) bit

This bit is used to enable or disable the mode to automatically clear the FIFO buffer memory after reading data from the specified pipe. With this bit set, this module performs the “BCLR = 1” processing on the FIFO buffer, if a zero-length packet is received when the FIFO buffer allocated to the specified pipe is empty or if data reading ends because a short packet is received when the BFRE bit is 1.

Always clears this bit when you use this module with the BRDYM bit set.

(d) UCL_Dx_DREQ output enable (DREQE) bit

This bit is used to enable or disable the output of the UCL_Dx_DREQ signal.

When enabling the UCL_Dx_DREQ signal, set this bit always after setting a value in the CURPIPE bits. When changing the value of the CURPIPE bits, change the value always after clearing this bit.

(e) DxFIFO port access bit width (MBW) bits

These bits are used to specify the bit width for the access to the DxFIFO port. For details, see **(c) CFIFO port access bit width (MBW) bits**.

(f) FIFO port access pipe specification (CURPIPE) bits

These bits are used to specify the pipe number through which to read or write data via the DxFIFO port.

If you change the value of these bits, write a desired value to these bits, read these bits, and then check that the written value is the same as the read value before proceeding to the next processing.

Do not specify the same pipe number for the CURPIPE bits in the CFIFOSEL, D0FIFOSEL, and D1FIFOSEL registers. If the value of these bits is changed in the middle of an access to the FIFO buffer, the access is held. Therefore, the same access can be resumed after these bits are restored to the original value.

(4) CFIFO Port Control Register (USB20_FUNC_CFIFOCTR)
D0FIFO Port Control Register (USB20_FUNC_D0FIFOCTR)
D1FIFO Port Control Register (USB20_FUNC_D1FIFOCTR)

Access Size : 16 bits

Address : <USB20_func_base> + 0022h (CFIFO)
<USB20_func_base> + 002Ah (D0FIFO)
<USB20_func_base> + 002Eh (D1FIFO)

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BVAL	BCLR	FRDY	-	DTLN[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW1	R0W1	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	BVAL	0h	RW1	Buffer memory valid flag Specify 1b for this bit when writing to the FIFO buffer on the CPU side through the pipe specified in CURPIPE (current pipe) ends. 0b: Invalid 1b: Writing complete
14	BCLR	0h	R0W1	The read value is always 0b. CPU buffer clear Specify 1b for this bit to clear the FIFO buffer on the CPU side of the current pipe. 0b: Invalid 1b: CPU buffer memory clear
13	FRDY	0h	R	FIFO port ready This bit indicates whether the FIFO port can be accessed. 0b: The FIFO port cannot be accessed. 1b: The FIFO port can be accessed.
12	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
11 to 0	DTLN[11:0]	0h	R	Receive data length These bits indicate the length of receive data.

(a) Buffer memory valid flag (BVAL)

When the pipe (selected pipe) specified in the CURPIPE bits is in the sending direction, this bit must be set in the cases descried below. This module switches the FIFO buffer from the CPU side to the SIE side to enable data sending.

1. To send short packets, set this bit when data writing ends.
2. To send zero length packets, set this bit before writing data to the FIFO buffer.
3. Set this bit after writing, to the pipe in continuous transfer mode, the data of which the size is a positive integral multiple of the maximum packet size and less than the buffer size.

When the maximum packet size of data is written to the pipe in non-continuous transfer mode, this module sets this bit to switch the FIFO buffer from the CPU side to the SIE side and enable data sending.

When this bit and the BCLR bit are set at the same time when the specified pipe is in the sending direction, this module clears the data that has been written so far and enables zero-length packets to be sent.

Setting this bit must be done only when the FRDY bit in the corresponding port control register is 1. If you want to check the value of the FRDY bit after setting this bit, wait at least 80 ns after setting this bit, and then reference the FRDY bit. Do not set this bit when the specified pipe is in the receiving direction.

(b) CPU buffer clear (BCLR) bit

When this bit is set, this module clears the FIFO buffer on the CPU side among the FIFO buffers allocated to the specified pipe.

Even if the two FIFO buffers in a double-buffer configuration are allocated to the specified pipe and the both buffers can be read, this module clears only one of the two buffers.

If this bit is set when the specified pipe is the DCP, this module clears the FIFO buffer regardless of whether the FIFO buffer is on the CPU side or it is on the SIE side. To clear the buffer on the SIE side, set this bit always after setting the PID bits for DCP to “NAK”.

When the specified pipe is other than the DCP, setting this bit must be done only when the FRDY bit in the corresponding port control register is 1. If you want to check the value of the FRDY bit after setting this bit, wait at least 80 ns after setting this bit, and then reference the FRDY bit.

(c) FIFO port ready (FRDY) bit

This bit indicates whether the FIFO port can be accessed from the CPU. This bit is operated by this module.

In the cases described below, even when this module sets this bit, data cannot be read from the FIFO port because the FIFO buffer does not contain data to be read. In these cases, set the BCLR bit to clear the FIFO buffer to enable the next sending and receiving of data.

1. A zero-length packet has been received while the FIFO buffer allocated to the specified pipe is empty.
2. A short packet has been received and data reading has ended while the BFRE bit is 1.

(d) Receive data length (DTLN) bits

These bits indicate the length of receive data. These bits are operated by this module. During reading of the FIFO buffer, the value of these bits varies depending on the value of the RCNT bit as described below.

- When the RCNT bit is 0:
  This module indicates a receive data length by using these bits until the CPU ends reading all received data from one FIFO buffer.
  When the BFRE bit is 1, this module retains the receive data length until the BCLR bit is set even if reading of received data has ended.
- When the RCNT bit is 1:
  This module decrements the value of these bits each time the CPU reads data.
  (The value is decremented by 1 when the MBW bits are 00, by 2 when the MBW bits are 01 or by 4 when the MBW bits are 10.)

When the CPU ends reading from one FIFO buffer, this module clears these bits. If reading of one of the FIFO buffers in a double-buffer configuration, however, ends before reading of received data from the other FIFO buffer ends, these bits indicate the receive data length for the other FIFO buffer at the end of reading from the FIFO buffer of which reading ends earlier.

When these bits are read during reading of the FIFO buffer when the RCNT bit is 1, this module updates the value of these bits within 150 ns after a cycle of read access to the corresponding FIFO port.

6.5.3.2.8 Interrupt Enable Registers

(1) Interrupt Enable Register 0 (USB20_FUNC_INTENB0)

Access Size : 16 bits

Address : <USB20_func_base> + 0030h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	VBSE	0h	RW	VBUS Interrupt Enable This bit prohibits or enables a USB interrupt when a VBINT interrupt is detected. 0b: Interrupt output prohibited 1b: Interrupt output enabled
14	RSME	0h	RW	Resume interrupt enable This bit prohibits or enables a USB interrupt when a RESM interrupt is detected. 0b: Interrupt output prohibited 1b: Interrupt output enabled
13	SOFE	0h	RW	Frame number update interrupt enable This bit prohibits or enables a USB interrupt when a SOF interrupt is detected. 0b: Interrupt output prohibited 1b: Interrupt output enabled
12	DVSE	0h	RW	Device state transition interrupt enable This bit prohibits or enables a USB interrupt when a DVST interrupt is detected. 0b: Interrupt output prohibited 1b: Interrupt output enabled
11	CTRE	0h	RW	Control transfer stage transition interrupt enable This bit prohibits or enables a USB interrupt when a CTRT interrupt is detected. 0b: Interrupt output prohibited 1b: Interrupt output enabled
10	BEMPE	0h	RW	Buffer empty interrupt enable This bit prohibits or enables a USB interrupt when a BEMP interrupt is detected. 0b: Interrupt output prohibited 1b: Interrupt output enabled
9	NRDYE	0h	RW	Buffer not ready response interrupt enable This bit prohibits or enables a USB interrupt when an NRDY interrupt is detected. 0b: Interrupt output prohibited 1b: Interrupt output enabled
8	BRDYE	0h	RW	Buffer ready interrupt enable This bit prohibits or enables a USB interrupt when a BRDY interrupt is detected. 0b: Interrupt output prohibited 1b: Interrupt output enabled
7 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(2) BRDY Interrupt Enable Register (USB20_FUNC_BRDYENB)

Access Size : 16 bits
Address : <USB20_func_base> + 0036h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PIPEBRDYE[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9 to 0	PIPEBRDYE [9:0]	0h	RW	Pipe BRDY interrupt enable These bits prohibit or enable the BRDY bit to be set when a BRDY interrupt to a pipe is detected. 0b: Interrupt output prohibited 1b: Interrupt output enabled

(a) Pipe BRDY interrupt enable (PIPEBRDYE) bits

When this module detects a BRDY interrupt to the pipe for which is set 1 in this register, this module sets the corresponding PIPEBRDY bit in the BRDYSTS register, sets the BRDY bit in the INTSTS0 register, and asserts the interrupt.

When at least one of the PIPEBRDY bits in the BRDYSTS register is 1 and software changes the corresponding interrupts enable bit in this register changes from 0 to 1, this module asserts the interrupt.

(3) NRDY Interrupt Enable Register (USB20_FUNC_NRDYENB)

Access Size : 16 bits
Address : <USB20_func_base> + 0038h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PIPENRDYE[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9 to 0	PIPENRDYE [9:0]	0h	RW	Pipe NRDY interrupt enable These bits prohibit or enable the NRDY bit to be set when a BRDY interrupt to a pipe is detected. 0b: Interrupt output prohibited 1b: Interrupt output enabled

(a) Pipe NRDY interrupt enable (PIPENRDYE) bits

When this module detects a BRDY interrupt to the pipe for which is set 1 in this register, this module sets the corresponding PIPENRDY bit in the NRDYSTS register, sets the NRDY bit in the INTSTS0 register, and asserts the interrupt.

When at least one of the PIPENRDY bits in the NRDYSTS register is 1 and the corresponding interrupts enable bit in this register changes from 0 to 1, this module asserts the interrupt.

(4) BEMP Interrupt Enable Register (USB20_FUNC_BEMPENB)

Access Size : 16 bits
Address : <USB20_func_base> + 003Ah
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PIPEBEMPE[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9 to 0	PIPEBEMPE [9:0]	0h	RW	Pipe BEMP interrupt enable These bits prohibit or enable the BEMP bit to be set when a BRDY interrupt to a pipe is detected 0b: Interrupt output prohibited 1b: Interrupt output enabled

Note: Bit numbers correspond to pipe numbers.

(a) Pipe BEMP interrupt enable (PIPEBEMPE) bits

When this module detects a BEMP interrupt to the pipe for which is set 1 in this register, this module sets the corresponding PIPEBEMP bit in the BEMPSTS register, sets the BEMP bit in the INTSTS0 register, and asserts the interrupt.

When at least one of the PIPEBEMP bits in the BEMPSTS register is 1 and the corresponding interrupts enable bit in this register changes from 0 to 1, this module asserts the interrupt.

6.5.3.2.9 SOF Control Register

(1) SOF Pin Configuration Register (USB20_FUNC_SOFCFG)

Access Size : 16 bits

Address : <USB20_func_base> + 003Ch

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	BRDYM	-	-	SOFM[1:0]		-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	RW	R	R	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
6	BRDYM	0h	RW	PIPEBRDY interrupt status clear timing This bit specifies the timing at which the PIPEBRDY interrupt is to be cleared. 0b: Software clears the status. 1b: Hardware clears the status by reading from or writing to the FIFO buffer. This bit can be set only during initialization (before communication). The setting cannot be changed after communication.
5, 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
3, 2	SOFM[1:0]	0h	RW	SOF function setting These bits are used to select SOF pulse output mode. 00b: SOF output disabled 01b: SOF output in units of 1 ms 10b: µSOF output in units of 125 µs 11b: Reserved
1, 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

6.5.3.2.10 Interrupt Status

(1) Interrupt Status Register 0 (USB20_FUNC_INTSTS0)

Access Size : 16 bits
Address : <USB20_func_base> + 0040h
Initial Value : 00x0h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY	VBSTS	DVSQ[2:0]			VALID	CTSQ[2:0]		
Initial Value	0	0	0	0	0	0	0	0	x	0	0	0	0	0	0	0
	-	-	-	1	-	-	-	-	-	0	0	1	-	-	-	-
R/W	RW0	RW0	RW0	RW0	RW0	R	R	R	R	R	R	R	RW0	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	VBINT	0h	RW0	Change detection interrupt status This bit indicates the VBUS change detection interrupt status. 0b: No VBUS interrupt is generated. 1b: A VBUS interrupt is generated.
14	RESM	0h	RW0	Resume interrupt status This bit indicates the resume detection interrupt status. 0b: No resume interrupt is generated. 1b: A resume interrupt is generated.
13	SOFR	0h	RW0	Frame number update interrupt status This bit indicates the frame number update interrupt status. 0b: No SOF interrupt is generated. 1b: A SOF interrupt is generated.
12	DVST	0h/1h	RW0	Device state transition interrupt status This bit indicates the device state transition interrupt. 0b: No device state transition interrupt is generated. 1b: A device state transition interrupt is generated.
11	CTRT	0h	RW0	Control transfer stage transition interrupt status This bit indicates the status of a control transfer stage transition interrupt. 0b: No control transfer stage transition interrupt is generated. 1b: A control transfer stage transition interrupt is generated.
10	BEMP	0h	R	BEMP interrupt status This bit indicates the BEMP interrupt status. 0b: No BEMP interrupt is generated. 1b: A BEMP interrupt is generated.
9	NRDY	0h	R	NRDY interrupt status This bit indicates the NRDY interrupt status. 0b: No NRDY interrupt is generated. 1b: An NRDY interrupt is generated.
8	BRDY	0h	R	BRDY interrupt status This bit indicates the BRDY interrupt status. 0b: No BRDY interrupt is generated. 1b: A BRDY interrupt is generated.
7	VBSTS	x	R	VBUS input status This bit indicates the VBUS pin input status. 0b: The VBUS pin is at the low level. 1b: The VBUS pin is at the high level.
6 to 4	DVSQ[2:0]	0h/1h	R	Device state These bits indicate the device state. 000b: Powered state 001b: Default state 010b: Address state 011b: Configured state 1xxb: Suspended state
3	VALID	0h	RW0	USB request reception This bit indicates whether USB request reception is detected. 0b: Not detected 1b: A setup packet is received.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	CTSQ[2:0]	0h	R	Control transfer stage These bits indicate the control transfer stage. 000b: Idle or setup stage 001b: Control reading data stage 010b: Control reading status stage 011b: Control writing data stage 100b: Control writing status stage 101b: Control writing (No Data) status stage 110b: Control transfer sequence error 111b: Reserved

Note: x: Undefined value

Note: When you want to clear the status indicated by the VBINT, RESM, SOFR, DVST, or CTRT bit, write 0b to only the bit to be cleared and 1b to other bits. Do not write 0b to any status bit that is currently 0b.

Note: This module detects a status change indicated by the VBINT or RESM bit in this register even while the clock is stopped (the SUSPM bit is 0b), and reports an interrupt corresponding to the status bit if the interrupt is enabled. Clearing the interrupt status must be done after the clock enabled.

(a) VBUS change interrupt status (VBINT) bit

This module sets this bit when it detects a change of the level of the VBUS pin input (from the high level to low level, or vice versa). This module indicates the level of the VBUS pin input by the VBSTS bit. When a VBINT interrupt occurs, read the VBSTS bit several times to check for consistency and remove chattering.

(b) Resume interrupt status (RESM) bit

This module sets this bit when it is in the suspended state (DVSQ bits are 1XX) and detects a falling edge of the signal at the DP pin.

(c) Frame number update interrupt status (SOFR) bit

This module sets this bit under the following conditions:

This module sets this bit when the frame number is updated. (The frame number update interrupt is monitored at intervals of 1 ms.)

This module detects an SOFR interrupt based on SOF interpolation even when an SOF packet from the USB host is damaged.

(d) Device state transition interrupt status (DVST) bit

When this module detects a change of the device state, this module updates the value of the DVSQ bits, and sets this bit. When a device state transition interrupt occurs, clear the interrupt status before this module detects the next device state transition.

(e) Control transfer stage transition interrupt status (CTRT) bit

When this module detects a transition of control transfer stage, this module updates the value of the CTSQ bits and sets this bit.

When a control transfer stage transition interrupt occurs, clear the interrupt status before this module detects the next transition of control transfer stage.

(f) Buffer empty interrupt status (BEMP) bit

This module sets this bit when at least one of the PIPEBEMP bits in the BEMPSTS register corresponding to the pipes for which the PIPEBEMPE bit in the BEMPENB register is set (that is, when this module detects a BEMP interrupt to at least one of the pipes for which is enabled BEMP interrupt notification).

For the conditions to assert the PIPEBEMP status signal, see the description of the BEMPSTS register.

This module clears this bit when writes 0 to all the PIPEBEMP bits corresponding to the pipes for which a BEMP interrupt has been enabled by setting the PIPEBEMPE bit.

Cannot clear this bit even by writing 0.

(g) Buffer not-ready interrupt status (NRDY) bit

This module sets this bit when at least one of the PIPENRDY bits in the BNRDYSTS register corresponding to the pipes for which the PIPENRDYE bit in the NRDYENB register is set (that is, when this module detects an NRDY interrupt to at least one of the pipes for which is enabled NRDY interrupt notification).

For the conditions to assert the PIPENRDY status signal, see the description of the NRDYSTS register.

This module clears this bit when writes 0 to all the PIPENRDY bits corresponding to the pipes for which a NRDY interrupt has been enabled by setting the PIPENRDYE bit.

Software cannot clear this bit even by writing 0.

(h) Buffer ready interrupt status (BRDY) bit

This module sets this bit when at least one of the PIPEBRDY bits in the BRDYSTS register corresponding to the pipes for which the PIPEBRDYE bit in the BRDYENB register is set (that is, when this module detects an BRDY interrupt to at least one of the pipes for which is enabled BRDY interrupt notification).

For the conditions to assert the PIPEBRDY status signal, see the description of the BRDYSTS register.

This module clears this bit when writes 0 to all the PIPEBRDY bits corresponding to the pipes for which a BRDY interrupt has been enabled by setting the PIPEBRDYE bit.

Software cannot clear this bit even by writing 0.

(2) BRDY Interrupt Status Register (USB20_FUNC_BRDYSTS)

Access Size : 16 bits
Address : <USB20_func_base> + 0046h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PIPEBRDY[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW0	RW0	RW0	RW0	RW0	RW0	RW0	RW0	RW0	RW0

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9 to 0	PIPEBRDY [9:0]	0h	RW0	Pipe BRDY interrupt status These bits indicate the BRDY interrupt status of each pipe. 0b: No interrupt is generated. 1b: An interrupt is generated.

Note: Bit numbers correspond to pipe numbers.

Note: To clear the interrupt status indicated by a bit in this register when the BRDYM bit is 0b, write 0b to only the bit to be cleared and 1b to other bits.

Note: When the BRDYM bit is 0b, clearing the BRDY interrupt status must be done always before the next access to the FIFO.

(a) Pipe BRDY interrupt status (PIPEBRDY) bit

When this module detects a BRDY interrupt to a pipe, this module sets the corresponding PIPEBRDY bit in the BRDYSTS register. At that time, if already set the corresponding bit in the BRDYENB register, this module sets the BRDY bit in the INTSTS0 register, and asserts the interrupt.

Conditions to generate and clear a BRDY interrupt vary depending on the values of the BRDYM bit, and the BFRE bit for each pipe.

a-1) BRDYM = 0 and BFRE = 0

When the BRDYM and BFRE bits are 0, the BRDY interrupt is generated to indicate that the FIFO port has become ready for access.

Under the conditions described below, this module generates an internal BRDY interrupt request trigger, and sets the PIPEBRDY bit corresponding to the pipe for which the request trigger is generated.

1. For the pipe in the sending direction
   - When the DIR bit changes from 0 to 1
   - When this module has ended sending packets through a pipe when data writing by the CPU to the FIFO buffer allocated to that pipe is disabled (when the value read from the BSTS bit is 0)
     In continuous transfer mode, a request trigger is generated when data has all been sent from one FIFO buffer.
   - When, in a double-buffer configuration, one FIFO buffer is empty when writing to the other FIFO buffer has ended
     If sending to one FIFO buffer has ended during writing to the other FIFO buffer, no request trigger is generated until the ongoing writing to the other FIFO buffer ends.
   - When this module flushes the FIFO buffer allocated to the pipe of which the transfer type is isochronous.
   - When the state of the FIFO buffer is changed from the write-disabled state to the write-enabled state by writing 1 to the ACLRM bit

No request trigger is generated when the pipe is the DCP (in other words, when data is sent by a control transfer).

2. For the pipe in the receiving direction

- When this module has ended receiving packets through a pipe and reading from the FIFO buffer is enabled when data reading by the CPU from the FIFO buffer allocated to that pipe is disabled (when the value read from the BSTS bit is 0)
  No request trigger is generated for the transaction that involves a data PID mismatch.
  In continuous transmission/reception mode, no request trigger is generated when the data size is the maximum packet size and the FIFO buffer still has a free space.
  If a short packet is received, a request trigger is generated even when the FIFO buffer has a free space.
  When the transaction counter is used, a request trigger is generated when the specified number of packets have been received.
  In that case, the request trigger is generated even if the FIFO buffer has a free space.
- When, in a double-buffer configuration, one FIFO buffer is in the read-enabled state when reading from the other FIFO buffer has ended
  If receiving from one FIFO buffer has ended during reading from the other FIFO buffer, no request trigger is generated until the ongoing reading from the other FIFO buffer ends.

This interrupt does not occur during the communication at the status stage of a control transfer.

The PIPEBRDY interrupt status of the corresponding pipe can be cleared to "0" by writing "0" to the corresponding bit. When clearing a PIPEBRDY bit by writing 0, write 1 to all the PIPEBRDY bits corresponding to other pipes.

Clearing the pipe BRDT interrupt status must be done always before the next access to the FIFO buffer.

a-2) When BRDYM = 0 and BFRE = 1

When the BRDYM bit is 0 and the BFRE bit is 1, this module determines that a BRDY interrupt occurs when all the data for a transfer has been read through a receiving pipe, and sets the PIPEBRDY bit corresponding to the pipe.

This module determines that the last data in a transfer has been received when one of the following conditions is met:

1. A short packet or a zero-length packet has been received.
2. The transaction counter (TRNCNT bits) is used, and as many packets as the value of the TRNCNT bits have been received.

When one of the above conditions is met and reading of the relevant data has ended, this module determines that all the data in a transfer has been read.

If a zero-length packet is received when the FIFO buffer is empty, this module determines that all the data in a transfer has been read when the FRDY bit is set and the DTLN bits are cleared in the corresponding FIFO Port Control Register. To start the next transfer in that case, write 1 to the BCLR bit in the corresponding FIFOCTR.

When the BRDYM bit is 0 and the BFRE bit is 1, this module does not detect any BRDY interrupt to the pipe in the sending direction.

The PIPEBRDY interrupt status of the corresponding pipe can be cleared to “0” by writing “0” to the corresponding bit. When clearing a PIPEBRDY bit by writing 0, write 1 to the PIPEBRDY bits corresponding to other pipes.

In this mode, do not change the value of the BFRE bit until all the processing for a transfer ends.

If you need to change the value of the BFRE bit during the transfer, set the ACLRM bit to clear all the FIFO buffers for the specified pipe.

a-3) When BRDYM = 1 and BFRE = 0

When the BRDYM bit is 1 and the BFRE bit is 0, the values of individual PIPEBRDY bits interlock with the values of the BSTS bits for individual pipes. In other words, this module sets or clears the BRDY interrupt status of a pipe according to the state of the FIFO buffer allocated to the pipe.

- For the pipe in the sending direction
  This module sets the PIPEBRDY bit for the pipe when data can be written to the FIFO port or clears the PIPEBRDY bit when data cannot be written to the FIFO port.
  The BRDY interrupt signal, however, is not asserted even when the sending pipe is write-enabled if the pipe is the DCP.
- For the pipe in the receiving direction
  This module sets the PIPEBRDY bit for the pipe when data can be read from the FIFO port or clears the PIPEBRDY bit when all data has been read (that is, when data reading from the FIFO port is disabled).
  If a zero-length packet is received when the FIFO buffer is empty, the PIPEBRDY bit corresponding to the specified pipe is kept being set and the BRDY interrupt signal is kept being asserted until sets the BCLR bit.

When the BRDYM bit is 1 and the BFRE bit is 0, this module cannot clear any PIPEBRDY bit.

When the BRDYM bit is 1, all the BFRE bits (for all pipes) must be 0.

(3) NRDY Interrupt Status Register (USB20_FUNC_NRDYSTS)

Access Size : 16 bits
Address : <USB20_func_base> + 0048h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PIPENRDY[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW0	RW0	RW0	RW0	RW0	RW0	RW0	RW0	RW0	RW0

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9 to 0	PIPENRDY [9:0]	0h	RW0	Pipe NRDY interrupt status These bits indicate the NRDY interrupt status of each pipe. 0b: No interrupt is generated. 1b: An interrupt is generated.

Note: Bit numbers correspond to pipe numbers.

Note: To clear the interrupt status indicated by a bit in this register, write 0b only to the bit and 1b to all other bit.

(a) Pipe NRDY interrupt status (PIPENRDY) bit

When this module issues an internal NRDY interrupt request for a pipe of which the PID is set to BUF, this module sets the PIPENRDY bit corresponding to the pipe in the NRDYSTS register. At that time, if the NRDYENB register bit corresponding to the pipe is set, this module sets the NRDY bit in the INTSTS0 register, and asserts the interrupt.

This module issues an internal NRDY interrupt request for individual pipes under the conditions described below. This module does not issue any interrupt request at the status stage of a control transfer.

a-1) For the pipe in the sending direction

- When an IN token is received in a situation where the PID bits corresponding to the specified pipe are 01 (BUF) and the FIFO buffer does not contain send data
  When an IN token is received, this module issues an NRDY interrupt request, and sets the PIPENRDY bit.
  If the transfer type of the pipe for which the interrupt is generated is isochronous transfer, this module sends a zero-length packet, and sets the OVRN bit.

a-2) For the pipe in the receiving direction

- When an OUT token is received in a situation where the PID bits corresponding to the specified pipe are 01 (BUF) and the FIFO buffer is full
  If the transfer type of the pipe for which the interrupt is generated is isochronous transfer, this module issues an NRDY interrupt request, sets the PIPENRDY bit, and sets the OVRN bit.
  If the transfer type of the pipe for which the interrupt is generated is not isochronous transfer, this module issues an NRDY interrupt request when sending an NAK Handshake signal after receiving the data that follows the OUT token, and sets the PIPENRDY bit.
  Note, however, that this module does not issue an NRDY interrupt request when resending data (when a DATA-PID mismatch has occurred).
  This module does not issue an NRDY interrupt request also when an error has occurred in a data packet.
- When a PING token is received in a situation where the PID bits corresponding to the specified pipe are 01 (BUF) and the FIFO buffer is full
  When a PING token is received, this module issues an NRDY interrupt request, and sets the PIPENRDY bit.

- When the transfer type of the specified pipe is isochronous transfer, the PID bits corresponding to the specified pipe are 01 (BUF), and data has not been received normally within an interval frame
  When an SOF token is received, this module issues an NRDY interrupt request, and sets the PIPENRDY bit for the specified pipe.

(4) BEMP Interrupt Status Register (USB20_FUNC_BEMPSTS)

Access Size : 16 bits
Address : <USB20_func_base> + 004Ah
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PIPEBEMP[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW0	RW0	RW0	RW0	RW0	RW0	RW0	RW0	RW0	RW0

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9 to 0	PIPEBEMP[9:0]	0h	RW0	Pipe BEMP interrupt status These bits indicate the BEMP interrupt status of each pipe. 0b: No interrupt is generated. 1b: An interrupt is generated.

Note: Bit numbers correspond to pipe numbers.

Note: To clear the interrupt status indicated by a bit in this register, write 0b only to the bit and 1b to all other bit.

(a) Pipe BEMP interrupt status (PIPEBEMP) bit

When this module detects a BEMP interrupt to a pipe of which the PID is set to BUF, this module sets the PIPEBEMP bit corresponding to the pipe in the BEMPSTS register. At that time, if the BEMPENB register bit corresponding to the pipe is set, this module sets the BEMP bit in the INTSTS0 register, and asserts the interrupt.

This module issues an internal BEMP interrupt request for individual pipes under the conditions described below.

1. When data sending (including sending of zero-length packets) to a pipe in the sending direction has ended and the FIFO buffer allocated to the pipe is empty
   In a single-buffer configuration, this module generates a BRDY interrupt at the same time as issuing an internal BEMP interrupt request for the pipes other than the DCP.
   Note, however, that this module does not issue the internal BEMP interrupt request in the following cases:
   - When software (DMAC) has already started writing to the CPU-side FIFO buffer, in a double-buffer configuration, when sending data for one buffer ends
   - When the buffer is cleared (to empty the buffer) by writing 1 to the ACLRM or BCLR bit
   - During an IN transfer (sending zero-length packets) at the status stage of a control transfer
2. For the pipe in the receiving direction
   When the data larger than the specified maximum packet size has been received normally
   In that case, this module issues a BEMP interrupt request, sets the PIPEBEMP bit for the specified pipe, discards the received data, and changes the value of the PID bits for the specified pipe to 11 (STALL).
   Then, this module returns a STALL packet.
   Note, however, that this module does not issue the internal BEMP interrupt request in the following cases:
   - When a CRC or bit stuff error has been detected in the received data
   - When a SETUP transaction is being executed

Writing 0 to this bit clears the interrupt status.

Writing 1 to this bit has no effect.

6.5.3.2.11 Frame Number Registers

(1) Frame Number Register (USB20_FUNC_FRMNUM)

Access Size : 16 bits
Address : <USB20_func_base> + 004Ch
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVRN	CRCE	-	-	-	FRNM[10:0]										
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW0	RW0	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	OVRN	0h	RW0	Overrun/underrun detection status This bit indicates whether an overrun or underrun is detected in the pipe being used to perform an isochronous transfer. 0b: No error 1b: An error occurred.
14	CRCE	0h	RW0	CRC error detection status This bit indicates the CRC error detection status for the pipe being used to perform an isochronous transfer. 0b: No error 1b: An error occurred.
13 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
10 to 0	FRNM[10:0]	0h	R	Frame number These bits indicate the latest frame number.

Note: The OVRN bit is intended for debugging. When designing a system, design transfer timings appropriately to prevent buffer overruns and underruns.

(a) Overrun/underrun detection status (OVRN) bit

This module sets this bit when this module detects an overrun or underrun in a pipe of which the transfer type is isochronous transfer.

When this module detects an overrun or underrun, this module issues an internal NRDY interrupt request. For details, see **(a) Pipe NRDY interrupt status (PIPENRDY) bit**.

Software can clear this bit by writing 0 to this bit. If the CRCE bit should not be cleared together when this bit is cleared, write 40h.

When the peripheral controller function is selected

This module sets this bit in the following cases:

1. When an IN token is received although writing send data to the FIFO has not ended, in the case of a pipe that is in the sending direction and performs an isochronous transfer
2. When an OUT token is received although the free space of FIFO buffer is less than the size of one FIFO buffer, in the case of a pipe that is in the sending direction and performs an isochronous transfer

(b) CRC error detection status (CRCE) bit

This module sets this bit when this module detects a CRC or bit stuff error in a pipe of which the transfer type is isochronous transfer.

Software can clear this bit by writing 0 to this bit. If the OVRN bit should not be cleared together when this bit is cleared, write 80h.

When this module detects a CRC error, this module issues an internal NRDY interrupt request. For details, see **(a) Pipe NRDY interrupt status (PIPENRDY) bit**.

(c) Frame number (FRNM) bits

This module updates the value of these bits each time an SOF packet is received (once per 1 ms), and indicates the latest frame number in these bits.

When reading these bits, read them twice and check for consistency.

(2) Micro Frame Number Register (USB20_FUNC_UFRMNUM)

Access Size : 16 bits
Address : <USB20_func_base> + 004Eh
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	UFRNM[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
2 to 0	UFRNM[2:0]	0h	R	Micro frame These bits indicate the micro frame number.

(a) Micro frame number (UFRNM) bit

In high-speed transfer mode, this module writes the micro frame number to these bits. In a mode other than high-speed transfer mode, this module writes 00h to these bits.

When reading these bits, read them twice and check for consistency.

6.5.3.2.12 USB Address

(1) USB Address Register (USB20_FUNC_USBADDR)

Access Size : 16 bits

Address : <USB20_func_base> + 0050h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	USBADDR[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
6 to 0	USBADDR[6:0]	0h	R	USB address These bits indicate the USB address allocated by the host.

(a) USB address (USBADDR) bits

When this module has received and normally processed a SetAddress request, this module writes the received USB address to these bits.

When this module detects a USB bus reset, this module writes 00h to these bits.

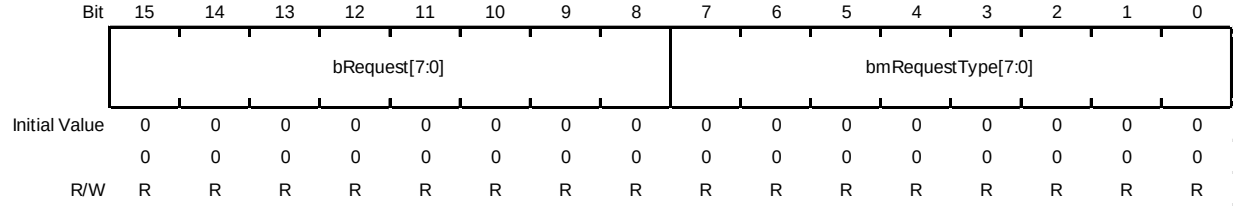
6.5.3.2.13 USB Request Registers

USB request registers are used to store control transfer setup requests.

These registers store the values set in the received USB requests.

(1) USB Request Type Register (USB20_FUNC_USBREQ)

Access Size : 16 bits
Address : <USB20_func_base> + 0054h
Initial Value : 0000h



Bit	Bit Name	Initial Value	R/W	Description
15 to 8	bRequest[7:0]	0h	R	Request Value of USBRequestbRequest
7 to 0	bmRequestType[7:0]	0h	R	Request type Value of USBRequestbmRequestType

(a) USB request (bRequest) bits

These bits indicate the value of the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

(b) USB request type (bRmRequestType) bits

These bits indicate the value of the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

(2) USB Request Value Register (USB20_FUNC_USBVAL)

Access Size : 16 bits
Address : <USB20_func_base> + 0056h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	wValue[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	wValue[15:0]	0h	R	Value Value of the wValue field in a USB request

(a) Value (wValue) bits

These bits indicate the value of the wValue field in a USB request.

These bits indicate the value of wValue field in the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

(3) USB Request Index Register (USB20_FUNC_USBINDX)

Access Size : 16 bits
Address : <USB20_func_base> + 0058h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	wIndex[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	wIndex[15:0]	0h	R	Index Value of the wIndex field in a USB request

(a) Index (wIndex) bits

These bits indicate the value of the wIndex field in a USB request.

These bits indicate the value of wIndex field in the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

(4) USB Request Length Register (USB20_FUNC_USBLENG)

Access Size : 16 bits
Address : <USB20_func_base> + 005Ah
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	wLength[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	wLength[31:0]	0h	R	Length Value of the wLength field in a USB request

(a) Length (wLength) bits

These bits indicate the value of the wLength field in a USB request.

These bits indicate the value of wLength field in the USB request data this module has received in a SETUP transaction. Writing to these bits is ignored.

6.5.3.2.14 DCP Configuration

When performing a data communication by a control transfer, use the default control pipe (DCP).

(1) DCP Configuration Register (USB20_FUNC_DCPCFG)

Access Size : 16 bits
Address : <USB20_func_base> + 005Ch
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CNTMD	SHTNAK	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
8	CNTMD	0h	RW	Continuous transfer mode This bit specifies whether to use the default control pipe to perform communication in continuous transfer mode. 0b: Non-continuous transfer mode 1b: Continuous transfer mode
7	SHTNAK	0h	RW	Pipe disable at transfer end When the default control pipe is in the receiving direction, this bit specifies whether to change the PID setting to NAK when the transfer ends. 0b: Continues the pipe when the transfer ends 1b: Disables the pipe when the transfer ends
6 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(2) DCP Max. Packet Size Register (USB20_FUNC_DCPMAXP)

Access Size : 16 bits
Address : <USB20_func_base> + 005Eh
Initial Value : 0040h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	MXPS[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
6 to 0	MXPS[6:0]	40h	RW	Maximum packet size These bits specify the maximum data payload size (maximum packet size) for the DCP.

(a) Maximum packet size (MXPS) bits

These bits are used to specify the maximum data payload size (maximum packet size) for the DCP.

The default is 40h (64 bytes).

The value of the MXPS bits must conform to the USB Specification.

Writing a value to the MXPS bits must be done when the PID is set to NAK and no value is set in the CURPIPE bits. If you need to change the value of these bits after changing the PID setting for the specified pipe from BUF to NAK, check the PBUSY bit is 0 before changing the value of these bits. If, however, this module has changed the PID setting from BUF to NAK, the value of the PBUSY bit need not be checked.

When the MXPS bits are all 0, data must not be written to the FIFO buffer, and the PID setting must not be changed to BUF.

(3) DCP Control Register (USB20_FUNC_DCPCTR)

Access Size : 16 bits
Address : <USB20_func_base> + 0060h
Initial Value : 0040h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	-	-	-	-	-	-	SQCLR	SQSET	SQMON	PBUSY	-	-	CCPL	PID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0
R/W	R	R	R	R	R	R	R	R0W1	R0W1	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0h	R	Buffer status This bit indicates the accessibility status of the DCP FIFO buffer. 0b: Buffer access is not possible. 1b: Buffer access is possible.
14 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
8	SQCLR	0h	R0W1	The read value is always 0b. Toggle bit clear This bit can set DATA0 as the expected value of the sequence toggle bit for the next transaction in DCP transfer. 0b: Writing disabled 1b: DATA0 specified
7	SQSET	0h	R0W1	The read value is always 0b. Toggle bit set This bit can set DATA1 as the expected value of the sequence toggle bit for the next transaction in DCP transfer. 0b: Writing disabled 1b: DATA1 specified
6	SQMON	1h	R	Sequence toggle bit monitor This bit indicates the expected value of the sequence toggle bit for the next transaction in DCP transfer. 0b: DATA0 1b: DATA1
5	PBUSY	0h	R	Pipe busy This bit indicates whether the specified pipe is being used in the USB bus. 0b: The specified pipe is not used in the USB bus. 1b: The specified pipe is being used in the USB bus.
4, 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
2	CCPL	0h	RW	Control transfer end enable Setting this bit permits the status stage of the control transfer to end. 0b: Does not permit the control transfer to end. 1b: Permits the control transfer to end.
1, 0	PID[1:0]	0h	RW	Response PID These bits control responses from this module in control transfer. 00b: NAK response 01b: BUF response (according to the buffer status) 10b: STALL response 11b: STALL response

(a) Buffer status (BSTS) bit

This bit indicates whether the CPU can access the FIFO buffer allocated to the DCP. This bit is operated by this module. The meaning of this bit varies as follows depending on the value of the ISEL bit:

- When ISEL = 0: This bit indicates whether receive data can be read from the FIFO buffer.
- When ISEL = 1: This bit indicates whether send data can be written to the FIFO buffer.

(b) Sequence toggle bit clear (SQCLR) bit

When software sets this bit, this module specifies DATA0 as the expected value of the sequence toggle bit for the specified pipe. This module always clears this bit.

Do not set the SQCLR and SQSET bits at the same time.

Setting this bit must be done when the PID is set to NAK and no value is set in the CURPIPE bits.

If you need to set this bit after changing the PID setting for the specified pipe from BUF to NAK, check, that the PBUSY bit is 0 before setting this bit. If, however, this module has changed the PID setting from BUF to NAK, the value of the PBUSY bit need not be checked.

(c) Sequence toggle bit set (SQSET) bit

When software sets this bit, this module specifies DATA1 as the expected value of the sequence toggle bit for the specified pipe. This module always clears this bit.

Do not set the SQCLR and SQSET bits at the same time.

Setting this bit must be done when the PID is set to NAK and no value is set in the CURPIPE bits.

If you need to set this bit after changing the PID setting for the specified pipe from BUF to NAK, check, that the PBUSY bit is 0 before setting this bit. If, however, this module has changed the PID setting from BUF to NAK, the value of the PBUSY bit need not be checked.

(d) Sequence toggle bit monitor (SQMON) bit

This bit indicates the expected value of the sequence toggle bit for the specified pipe. This bit is operated by this module. When a transaction ends normally, this module toggles this bit.

This module, however, does not toggle this bit if a DATA-PID mismatch occurs during a transfer in the receiving direction.

When a SETUP packet is received normally, this module sets this bit (to specify DATA1 as the expected value).

This module does not reference this bit in an IN or OUT transaction at the status stage. Also, this module does not toggle this bit even when the transaction ends normally.

(e) Pipe busy (PBUSY) bit

This module changes this bit from 0 to 1 when a USB transaction using the specified pipe starts. This module changes this bit from 1 to 0 when the transaction ends.

Reading this bit after software sets the PID to NAK enables you to check whether you can change pipe settings.

(f) Control transfer end enable (CCPL) bit

When software sets this bit when the PID of the specified pipe is BUF, this module ends the status stage of the ongoing control transfer.

In other words, in a control read transfer, this module sends an ACK handshake in response to an OUT transaction request from the USB host, and, in a control write or no-data control transfer, this module sends a zero-length packet in response to an IN transaction request from the USB host. If, however, a SetAddress request is detected, this module performs automatic response throughout the period from the setup stage to the end of the status stage regardless of the value of this bit.

When a new SETUP packet is received, this module changes this bit from 1 to 0.

When the VALID bit is 1, software cannot set this bit.

(g) Response PID (PID) bits

The setting of these bits must be changed from NAK to BUF when the data stage or status stage of a control transfer is executed.

This module changes the value of these bits in the following cases:

- This module changes the value of these bits to 00 (NAK) when it receives a SETUP packet. At that time this module sets the VALID bit. Software cannot change the value of these bits until it clears the VALID bit.
- When set these bits to 01 (BUF), this module changes the value of these bits to 11 (STALL) when it receives the data exceeding the specified maximum packet size.
- This module changes the value of these bits to 1x (STALL) when it detects a sequence error in a control transfer.
- This module changes the value of these bits to 00 (NAK) when it detects a USB reset.

This module does not reference these bits during SetAddress request processing (automatic processing).

6.5.3.2.15 Pipe Configuration Registers

To configure pipes 1 to 15, use the PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI, PIPExCTR, PIPExTRE, and PIPExTRN registers.

Select the pipes to be used by using the PIPESEL register, and then configure functions of individual pipes by using the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers. Note that you can use the PIPExCTR, PIPExTRE, and PIPExTRN registers for setting regardless of the pipe selection by the PIPESEL register.

(1) Pipe Window Select Register (USB20_FUNC_PIPESEL)

Access Size : 16 bits
Address : <USB20_func_base> + 0064h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	PIPESEL[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
3 to 0	PIPESEL[3:0]	0h	RW	Pipe window select These bits specify a pipe for registers at addresses 68h to 6Eh. 0000b: No selection 0001b: PIPE1 0010b: PIPE2 0011b: PIPE3 0100b: PIPE4 0101b: PIPE5 0110b: PIPE6 0111b: PIPE7 1000b: PIPE8 1001b: PIPE9

Note: When the PIPESEL bits are 0000b, all bits of the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers are cleared. When the PIPESEL bits are 0000b, writing to the registers at addresses 68h to 6Eh is ignored.

(a) Pipe Window select (PIPESEL) bits

When software a value from 0001 to 1111 to these bits, this module indicate the pipe information and settings corresponding to the registers at addresses 68h to 6Eh. After a pipe is selected by these bits, the values set in the areas at addresses 68h to 6Eh are applied, by this module, to the transfer operation using the selected pipe.

When writes 0000 to these bits, this module writes 0 to all bits of the registers at addresses 68h to 6Eh. Then, writing to the areas at addresses 68h to 6Eh is ignored.

(2) Pipe Configuration Register (USB20_FUNC_PIPECFG)

Access Size : 16 bits
Address : <USB20_func_base> + 0068h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TYPE[1:0]		-	-	-	BFRE	DBLB	CNTMD	SHTNAK	-	-	DIR	EPNUM[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	R	R	R	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15, 14	TYPE[1:0]	0h	RW	Transfer type These bits specify the transfer type of the pipe specified in the PIPESEL bit. 00b: The pipe cannot be used. 01b: Bulk transfer 10b: Interrupt transfer 11b: Isochronous transfer
13 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
10	BFRE	0h	RW	BRDY interrupt operation specification This bit specifies the timing at which this module notifies a BRDY interrupt relating to the specified pipe. 0b: A BRDY interrupt is notified when data is sent or received. 1b: A BRDY interrupt is notified when reading of data is completed.
9	DBLB	0h	RW	Double-buffer mode This bit specifies a single or double FIFO buffer to be used by the specified pipe. 0b: Single buffer 1b: Double buffer
8	CNTMD	0h	RW	Continuous transfer mode This bit specifies whether to use the specified pipe to perform communication in continuous transfer mode. 0b: Non-continuous transfer mode 1b: Continuous transfer mode
7	SHTNAK	0h	RW	Pipe disable at transfer end When the specified pipe is in the receiving direction, this bit specifies whether to change the PID setting to NAK when the transfer ends. 0b: Continues the pipe when the transfer ends. 1b: Disables the pipe when the transfer ends.
6, 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
4	DIR	0h	RW	Transfer direction This bit specifies the transfer direction of the specified pipe. 0b: Receiving direction 1b: Sending direction
3 to 0	EPNUM[3:0]	0h	RW	Endpoint number These bits specify the endpoint number of the specified pipe.

(a) Transfer type (TYPE) bits

These bits are used to specify the USB transfer type of the pipe (selected pipe) specified in the PIPESEL bits.

Table 6.5-20 lists pipes and the transfer types specifiable in these bits.

Table 6.5-20 Selected Pipes and the Transfer Types Specifiable in the TYPE Bits

Selected Pipe	TYPE Bits	USB Transfer Type
PIPE1 or PIPE2	01 or 11	Bulk or isochronous transfer
PIPE3 to PIPE5	01	Bulk transfer
PIPE6 to PIPE9	10	Interrupt transfer

Always specify a value other than 00 in these bits for a selected pipe before setting the PID of the selected pipe to BUF (to start USB communication using the selected pipe).

The value of these bits for a selected pipe must be done while the PID of the selected pipe is NAK. If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(b) BRDY interrupt operation specification (BFRE) bit

This bit is valid when the selected PIPE is PIPE1 to PIPE5.

When set this bit and been using the selected pipe in the receiving direction (in other words, the DIR bit is 0), this module detects the end of transfer (when it occurs) and generates a BRDY interrupt when reading of the last packet ends.

If a BRDY interrupt occurs with the above settings, software must write 1 to the BCLR bit. The FIFO buffer allocated to the selected pipe remains unready for reception until 1 is written to the BCLR bit.

When set this bit and been using the selected pipe in the sending direction (in other words, the DIR bit is 1), this module does not generate any BRDY interrupt.

For details, see the description of the PIPEBRDY interrupt status bit.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(c) Double-buffer mode (DBLB) bit

This bit is valid when the selected pipe is PIPE1 to PIPE5.

When set this bit for a selected pipe, this module allocates, to the selected pipe, two FIFO buffers, each of which has the FIFO buffer size specified in the BUFSIZE bits in the PIPEBUF register.

The size of the FIFO buffer this module allocates to the selected pipe is as follows:

$$(BUFSIZE + 1) \times 64 \times (DBLB + 1) \text{ [bytes]}$$

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(d) Continuous transfer mode (CNTMD) bit

This bit is valid when the selected pipe is PIPE1 to PIPE5, and the transfer type of the selected pipe is bulk transfer.

This module determines whether data sending from or receiving in the FIFO buffer allocated to the selected pipe has ended according to the value of this bit in the way described in **Table 6.5-21**.

Table 6.5-21 How to Determine the End of Data Sending from or Receiving in the FIFO Buffer According to the Value of the CNTMD Bit

CNTMD Bit Setting Value	How to Determine Whether Reading or Sending is Enabled
0	Condition for enabling reading from the FIFO buffer when the receiving direction is set (“DIR = 0”): This module receives one packet
	Condition for enabling sending from the FIFO buffer when the sending direction is set (“DIR = 1”): Either of the following conditions 1) and 2) is met: 1) Data for the maximum packet size is written to the FIFO buffer. 2) Data for the short packet (including the case of zero-byte data) is written to the FIFO buffer and then 1 is written in the BVAL bit.
1	Conditions for enabling reading from the FIFO buffer when the receiving direction is set (“DIR = 0”): 1) The number of bytes in data received in the FIFO buffer allocated to the selected pipe becomes equal to the number of allocated bytes ((BUFSIZE + 1) × 64). 2) This module receives a short packet other than a zero-length packet. 3) This controller receives a zero-length packet when the FIFO buffer allocated to the selected pipe already contains data. 4) Packets are received as many times as the value of the transaction counter set for the selected pipe.
	Condition for enabling sending from the FIFO buffer when the sending direction is set (“DIR = 1”): One of the following conditions 1) to 3) is met: 1) The amount of written data becomes equal to the size of one FIFO buffer allocated to the selected pipe. 2) Data for less than the size of one FIFO buffer allocated to the selected pipe (including the case of zero-byte data) is written to the FIFO buffer and then 1 is written in the BVAL bit. 3) Data for less than the size of one FIFO buffer allocated to the selected pipe (including the case of zero-byte data) is written to the FIFO buffer, and a transfer end signal is asserted at the same time of the last writing.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(e) PIPE disable at transfer end (SHTNAK) bit

This bit is valid when the selected pipe is PIPE1 to PIPE5 and is in the receiving direction.

When set this bit for a selected pipe in the receiving direction, this module changes the PID of the selected pipe to NAK when this module determines the end of data transfer to the selected pipe. This module determines the end of transfer when one of the following conditions (1) and (2) is met:

(1) This module has normally received short packet data (including zero-length packets).

(2) When using a transaction counter, this module has normally received as many packets as the value set in the transaction counter.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

For the pipes in the sending direction, this bit must be cleared.

(f) Transfer direction (DIR) bit

When writes 0 to this bit for a selected pipe, this module uses the selected pipe in the receiving direction. When software writes 1 to this bit for the selected pipe, this module uses the selected pipe in the sending direction.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after the USB communication using the selected pipe, not only check the values of above three kinds of register bits but also write 1 and 0 successively to the ACLRM bit to clear the FIFO buffer allocated to the selected pipe.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(g) Endpoint number (EPNUM) bits

These bits are used to specify the endpoint number of the endpoint of a selected pipe. Note that specifying 0000 in these bits for a pipe means that the pipe is not used.

Changing the value of these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

The combination of the values of the DIR bit and EPNUM bits for a pipe must be unique among those for all pipes. (The setting “EPNUM = 000” [the selected pipe is not used] can be duplicated for multiple pipes.)

(3) Pipe Buffer Setting Register (USB20_FUNC_PIPEBUF)

Access Size : 16 bits
Address : <USB20_func_base> + 006Ah
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	BUFSIZE[4:0]					-	-	BUFNMB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
14 to 10	BUFSIZE[4:0]	0h	RW	Buffer size These bits specify the size of the FIFO buffer for the pipe specified in the PIPESEL bit. 00h: 64 bytes 01h: 128 bytes ... (1Fh: 2 Kbytes)
9, 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
7 to 0	BUFNMB[7:0]	0h	RW	Buffer number These bits specify the FIFO buffer number of the specified pipe. (4h to 7Fh)

Note: Changing values of these register bits for a selected pipe must be done when the PID of the selected pipe is set to NAK, and no pipe is specified in the CURPIPE bits.

Note: If you change values of these register bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check that the PBUSY bit is 0b before changing the values of these bits. If this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(a) Buffer size (BUFSIZE) bits

These bits are used to specify the size of the FIFO buffer to be allocated to the selected pipe. Specify the FIFO buffer size in units of blocks. One block has 64 bytes.

When set the DBLB bit for a selected pipe, this module allocates, to the selected pipe, two FIFO buffers, each of which has the FIFO buffer size specified in these bits.

The size of the FIFO buffer this module allocates to the selected pipe is as follows:

(BUFSIZE + 1) × 64 × (DBLB + 1) [bytes]

The following value can be specified in these bits:

- Any value from 00h to 1Fh when the selected pipe is PIPE1 to PIPE5.
- 0h only when the selected pipe is PIPE6 to PIPE9.

In continuous transfer mode (CNTMD = 1), specify an integral multiple of the maximum packet size in the BUFSIZE bits.

(b) Buffer number (BUFNMB) bits

These bits are used to specify the block number of the first block in the FIFO buffer to be allocated to the selected pipe. This module allocates the following blocks of FIFO buffer to the selected pipe:

Block with block number “BUFNMB” to the block with block number “BUFNMB + (BUFSIZE + 1) × (DBLB + 1) − 1”

The value of these bits must be 04h to 7Fh. Note, however, that the following rules must be observed: Value “00h” is exclusively used for DCP.

Value “04h” is exclusively used for PIPE6. When, however, PIPE6 is not used, this value can be used for another pipe. If PIPE6 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets 04h in the BUFNMB bits for PIPE6.

Value “05h” is exclusively used for PIPE7. When, however, PIPE7 is not used, this value can be used for another pipe. If PIPE7 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets 05h in the BUFNMB bits for PIPE7.

Value “06h” is exclusively used for PIPE8. When, however, PIPE8 is not used, this value can be used for another pipe. If PIPE8 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets 06h in the BUFNMB bits for PIPE8.

Value “07h” is exclusively used for PIPE9. When, however, PIPE9 is not used, this value can be used for another pipe. If PIPE9 is selected, writing to the BUFNMB bits is ignored. Then, this module aromatically sets 07h in the BUFNMB bits for PIPE9.

(4) Pipe Maximum Packet Size Register (USB20_FUNC_PIPEMAXP)

Access Size : 16 bits
Address : <USB20_func_base> + 006Ch
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	MXPS[10:0]										
Initial Value	0	0	0	0	0	0	0	0	0	0(1)	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
10 to 0	MXPS[10:0]	0(1)*1	RW	Maximum packet size These bits specify the maximum data payload size (maximum packet size) for the specified pipe. For PIPE6 to PIPE8, a value from 1h to 40h (bytes) can be set.

Note 1. The initial value of the MXPS bits is 00h when no pipe is specified in the PIPESEL bits in the PIPESEL register or 40h when a pipe is specified in the PIPESEL bits.

(a) Maximum packet size (MXPS) bits

These bits are used to specify the maximum data payload size (maximum packet size) for the selected pipe.

The initial value of these bits is 40h (64 bytes).

- For PIPE1 and PIPE2, a value from 1h (1 byte) to 400h (1024 bytes) can be specified.
- For PIPE3 to PIPE5, 8h (8 bytes), 10h (16 bytes), 20h (32 bytes), 40h (64 bytes), or 200h (512 bytes) can be specified. (Bits [2:0] are excluded.)
- For PIPE6 to PIPE9, a value from 1h (1 byte) to 40h (64 bytes) can be specified.

The value of the MXPS bits for individual transfer type must conform to the USB Specification.

Setting a value in these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check the PBUSY bit is 0 before changing the value of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

When the MXPS bits are all 0, data must not be written to the FIFO buffer, and the PID setting must not be changed to BUF.

(5) Pipe Cycle Control Register (USB20_FUNC_PIPEPERI)

Access Size : 16 bits
Address : <USB20_func_base> + 006Eh
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	IFIS	-	-	-	-	-	-	-	-	-	IITV[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	RW	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
12	IFIS	0h	RW	Isochronous IN buffer flush This bit specifies whether to perform a buffer flush when the pipe specified in the PIPESEL bit is used for isochronous IN transfer. 0b: Does not perform a buffer flush. 1b: Performs a buffer flush.
11 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
2 to 0	IITV[2:0]	0h	RW	These bits specify the transfer interval of the specified pipe. The value to be specified is the frame timing multiplied by an n-th power of 2.

(a) Isochronous IN buffer flush (IFIS) bit

When the selected pipe is used for isochronous IN transfer, this bit is used to specify this module automatically clears the FIFO buffer if this module fails to receive the IN token from the USB host in a (micro) frame sent at intervals specified in the IITV bits.

In double-buffer mode (DBLB = 1), this module clears only the data in one buffer used earlier than the other.

This module clears the FIFO buffer when it receives an SOF packet immediately after the (micro) frame in which the IN token hag to be received. Even if the SOF packet is corrupted, this module clears the FIFO buffer in the same timing to receive the SOF packet by the use of the internal interpolation function.

(b) Interval error detection interval (IITV) bits

These bits specify the interval of interval error detection for the selected pipe. The value to be specified is the frame timing multiplied by an n-th power of 2.

Setting a value in these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of these bits for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of these bits. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

If you change the value specified in these bits to another after a USB communication, change the PID to NAK, and then set the ACLRM bit to initialize the interval timer before changing the value.

For PIPE3 to PIPE5 these bits are ignored. Write 0 to all these bits corresponding to PIPE3 to PIPE5.

You can specify a value in these bits when the transfer type of the selected pipe is isochronous.

b-1) When the selected pipe is used for isochronous OUT transfer

If this module does not receive any data packet in the (micro) frame sent at intervals specified in the IITV bits, this module generates an NRDT interrupt.

This module generates an NRDY interrupt also if this module cannot receive data because an error, e.g., CRC error, has occurred in a data packet or because the FIFO buffer is full (such a situation might result if, for example, software [DMAC] delays in reading data from the FIFO buffer).

This module generates the NRDY interrupt when it receives an SOF packet. Even if the SOF packet is corrupted, this module generates the NRDY interrupt in the same timing to receive the SOF packet by the use of the internal interpolation function.

When, however, the value of the IITV bits is not 0, this module generates the NRDY interrupt every time an SOF packet is received at the specified intervals after interval counting starts.

If the PID of the selected pipe is changed to NAK after the interval timer starts, this module does not generate the NRDY interrupt even when it receives an SOF packet.

The condition for starting interval counting varies by the value of the IITV bits.

- When IITV = 0: Interval counting starts when the PID of the selected pipe is changed to BUF.

(Micro) frame	SOF	SOF	SOF	OUT	DATA0	SOF	OUT	DATA0
Setting of PID bits	NAK	NAK	BUF			BUF		
Whether token reception is expected (0: reception expected —: non-reception expected)	—	—	0			0		
Start of interval counting			↑					

Figure 6.5-4 Relationship between (Micro) Frames and Whether Token Reception Is Expected When IITV = 0

- When IITV is not 0: Interval counting starts at the end of the first normal reception of data packet after the PID of the selected pipe is changed to BUF.

(Micro) frame	SOF	SOF	SOF	OUT	DATA0	SOF	SOF	OUT	DATA0	SOF	SOF	OUT	DATA0
Setting of PID bits	NAK	BUF	BUF			BUF	BUF			BUF	BUF		
Whether token reception is expected (0: reception expected —: non-reception expected)	—	—	0			—	0			—	0		
Start of interval counting			↑										

Figure 6.5-5 Relationship between (Micro) Frames and Whether Token Reception Is Expected When IITV = 1

b-2) When the selected pipe is used for isochronous IN transfer

The IITV bits are used in combination with the setting of the IFIS bit to 1. When the IFIS bit is 0, this module sends a data packet in response to a received token regardless of the value of the IITV bits.

When the IFIS bit is 1, if this module does not receive any IN token in the (micro) frame sent at intervals specified in the IITV bits although the FIFO buffer has sendable data, this module clears the FIFO buffer.

This module clears the FIFO buffer also when it cannot receive an IN token normally because of a bus error, e.g., CRC error.

This module clears the FIFO buffer when it receives an SOF packet. Even if the SOF packet is corrupted, this module clears the FIFO buffer in the same timing to receive the SOF packet by the use of the internal interpolation function.

The condition for starting interval counting varies by the value of the IITV bits. (The condition is the same as that for isochronous OUT transfer.)

The interval counter is cleared when one of the following conditions (1) to (3) is met:

(1) This module is reset (then, also the IITV bits are cleared).

(2) The ACLRM bit is set.

(3) This module detects a USB bus reset.

6.5.3.2.16 Pipe Control Registers

(1) PIPEn Control Register (USB20_FUNC_PIPEnCTR) (n = 1 to 5)

Access Size : 16 bits

Address : <USB20_func_base> + 0070h + (n - 1) x 0002h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	INBUFM	-	-	-	ATREPM	ACLRM	SQCLR	SQSET	SQMON	PBUSY	-	-	-	PID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0
R/W	R	R	R	R	R	RW	RW	R0W1	R0W1	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0h	R	Buffer status This bit indicates the FIFO buffer status of the specified pipe. 0b: Buffer access is not possible. 1b: Buffer access is possible.
14	INBUFM	0h	R	Transmit buffer monitor When the specified pipe is in the sending direction, this bit indicates the FIFO buffer status of the specified pipe. 0b: The FIFO buffer does not contain data that can be sent. 1b: The FIFO buffer contains data that can be sent.
13 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
10	ATREPM	0h	RW	Automatic response mode This bit prohibits or enables automatic response of the specified pipe. 0b: Automatic response prohibited 1b: Automatic response enabled. (A zero-length packet response is sent during transmission. For reception, an NAK response is sent and an NRDY interrupt is generated.)
9	ACLRM	0h	RW	Automatic buffer clear mode This bit prohibits or enables automatic buffer clear mode for the specified pipe. 0b: Prohibited 1b: Enabled (all buffers are initialized)
8	SQCLR	0h	R0W1	The read value is always 0b. Toggle bit clear Specify 1b in this bit to clear the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA0. 0b: Writing disabled 1b: DATA0 specified
7	SQSET	0h	R0W1	The read value is always 0b. Toggle bit set Specify 1b in this bit to set the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA1. 0b: Writing disabled 1b: DATA1 specified
6	SQMON	0h	R	Sequence toggle bit monitor This bit indicates the expected value of the sequence toggle bit for the next transaction in the specified pipe. 0b: DATA0 1b: DATA1
5	PBUSY	0h	R	Pipe busy This bit indicates whether the specified pipe is being used in the USB bus. 0b: The specified pipe is not used in the USB bus. 1b: The specified pipe is being used in the USB bus.
4 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PID[1:0]	0h	RW	Response PID These bits specify the response method for the next transaction in the specified pipe. 00b: NAK response 01b: BUF response (according to the buffer status) 10b: STALL response 11b: STALL response

(a) Buffer status (BSTS) bit

This bit indicates whether the CPU can access the FIFO buffer allocated to the selected pipe. This bit is operated by this module.

The meaning of this bit varies as follows depending on the value of the values of the DIR, BFRE, and DCLRM bits:

Table 6.5-22 BSTS Bit Operations

DIR bit Setting Value	BFRE bit Setting Value	DCLRM bit Setting Value	Meaning of the BSTS Bit
0	0	0	This bit indicates 1 when reading of received data from the FIFO buffer becomes possible, and indicates 0 when reading data has finished.
		1	Setting prohibited
	1	0	This bit indicates 1 when reading of received data from the FIFO buffer becomes possible. This bit indicates 0 when 1 is written in the BCLF bit after reading data has finished.
		1	This bit indicates 1 when reading of received data from the FIFO buffer becomes possible, and indicates 0 when reading data has finished.
1	0	0	This bit indicates 1 when writing of send data in the FIFO buffer becomes possible, and indicates 0 when writing data has finished.
		1	Setting prohibited
	1	0	Setting prohibited
		1	Setting prohibited

(b) Transmit buffer monitor (INBUFM) bit

When the selected pipe is in the sending direction (DIR = 1), this module sets this bit when software (or the DMAC) has finished writing data to at least one FIFO buffer.

This module clears this bit when this module finishes sending all data from the FIFO buffer to which the data has been written. In double-buffer mode (DBLB = 1), this module clears this bit when this module has finished sending all data from the two FIFO buffers and software (or the DMAC) has not yet finished writing data to one FIFO buffer.

When the selected pipe is in the receiving direction (DIR = 0), this bit indicates the same value as that of the BSTS bit.

(c) Automatic response mode (ATREPM) bit

This bit can be set when the transfer type of the selected pipe is bulk transfer.

When this bit is 1, this module responds to tokens sent from the USB host as described below.

1. When the selected pipe is used for bulk IN transfer (TYPE = 01 and DIR = 1)
   When the ATREPM bit is 1 and the PID of the selected pipe is BUF, this module responds to an IN token by sending a zero-length packet.
   Each time this module receives ACK from the USB host (the sequence of one transaction is receiving an IN token, sending a zero-length packet, and then receiving ACK), this module updates (toggles) the sequence toggle bit (DATA- PID).
   This module does not generate BRDY and BEMP interrupts.

2. When the selected pipe is used for bulk OUT transfer (TYPE = 01 and DIR = 0)
   When the ATREPM bit is 1 and the PID of the selected pipe is BUF, this module responds to an OUT token (or a PING token) by sending an NAK response and generates an NRDY interrupt.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

To perform a USB communication with this bit set, the FIFO buffer must be empty. No data must be written to the FIFO buffer during the USB communication with this bit set.

When the transfer type of the selected pipe is isochronous transfer, this bit must always be 0.

(d) Automatic buffer clear mode (ACLRM) bit

When you need to clear the whole FIFO buffer allocated to the selected pipe, write 1 and 0 successively to the ACLRM bit.

Table 6.5-23 shows the buffer contents this module clears when 1 and 0 are written successively to the ACLRM bit.

Table 6.5-24 shows the cases that require this processing.

Table 6.5-23 Buffer Contents This Core Clears When the ACLRM Bit is Set

No.	Contents to be Cleared by Setting the ACLRM Bit
(1)	Whole contents of the FIFO buffer allocated to the specified pipe (if the double buffer is set, both FIFO buffers are cleared)
(2)	If the transfer type of the specified pipe is Isochronous transfer, the interval count value is cleared.

Table 6.5-24 Cases Requiring the ACLRM Bit to be Set

No.	Cases when Data Clear is Required
(1)	The whole contents of the FIFO buffer allocated to the specified pipe needs to be cleared.
(2)	The interval count value needs to be reset.
(3)	The value of the BFRE bit is changed.
(4)	The value of the DBLB bit is changed.
(5)	Forced termination of the transaction count function is performed.

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(e) Sequence toggle bit clear (SQCLR) bit

When software sets this bit, this module specifies DATA0 as the expected value of the sequence toggle bit for the selected pipe. This module always clears this bit.

Writing 1 to the SQCLR bit must be done when the PID of the selected pipe is NAK.

If you write 1 to this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(f) Sequence toggle bit set (SQSET) bit

When software sets this bit, this module specifies DATA1 as the expected value of the sequence toggle bit for the selected pipe. This module always clears this bit.

Writing 1 to the SQSET bit must be done when the PID of the selected pipe is NAK.

If you write 1 to this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(g) Sequence toggle bit monitor (SQMON) bit

This bit indicates the expected value of the sequence toggle bit for the selected pipe. This bit is operated by this module.

When the transfer type of the selected pipe is other than isochronous transfer, this module toggles this bit when a transaction ends normally. This module, however, does not toggle this bit if a DATA-PID mismatch occurs during a transfer in the receiving direction.

(h) Pipe busy (PBUSY) bit

This module changes this bit from 0 to 1 when a USB transaction using the selected pipe starts. This module changes this bit from 1 to 0 when the transaction ends normally.

Reading this bit after software sets the PID to NAK enables you to check whether you can change pipe settings.

(i) Response PID (PID) bits

These bits are used to specify, the type of response of this module for individual pipes.

The default of PID is NAK. When the selected pipe is used for USB transfers, the PID setting muse be changed to BUF. For the basic operations (without communication packet errors involved) of this module depending on the value of the PID bits, see **Table 6.5-25**.

If you have changed the PID of a selected pipe from BUF to NAK while the selected pipe is performing a USB communication, check, that the PBUSY bit is 0 to confirm that the USB transfer through the selected pipe has actually changed to the NAK status. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

This module changes the value of the PID bits in the following cases:

- When the selected pipe is in the receiving direction and set the SHTNAK bit for the selected pipe, this module sets the PID to NAK when this module recognizes the end of a transfer.
- When this module has received a data packet of which the payload size is larger than the maximum packet size, this module sets the PID to STALL (PID = 11).
- If this module detects a USB bus reset, this module sets the PID to NAK.

To change the PID from NAK (PID = 00) to STALL, write 10 to the PID bits. To change the PID from BUF (PID = 01) to STALL, write 11 to the PID bits.

To change the PID from STALL (PID = 11) to NAK, write 10 to the PID bits once, and then write 00 to the PID bits. To change the PID from STALL to BUF, change the PID to NAK once, and then change it to BUF.

Table 6.5-25 Core Operations Depending on the PID Setting

PID bit Setting Value	Transfer Type (TYPE Bit Setting Value)	Transfer Direction (DIR Bit Setting Value)	Operation of This Core
00 (NAK)	Bulk (“TYPE = 01”), or Interrupt (“TYPE = 10”)	Independent of the setting value	Sends a NAK response for a token from the USB host.
	Isochronous (“TYPE = 11”)	Receiving direction (“DIR = 0”)	Does not respond to a token from the USB host.
		Sending direction (“DIR = 1”)	Sends a zero-length packet for a token from the USB host.
01 (BUF)	Bulk (“TYPE = 01”)	Receiving direction (“DIR = 0”)	For an OUT token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module receives data and then sends an ACK or NYET response. If receiving data is not possible, this module sends an NAK response. For a PING Token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module sends an ACK response. If receiving data is not possible, this module sends an NAK response.
	Interrupt (“TYPE = 10”)	Receiving direction (“DIR = 0”)	For an OUT token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module receives data and then sends an ACK response. If receiving data is not possible, this module sends an NAK response.
	Bulk (“TYPE = 01”) or Interrupt (“TYPE = 10”)	Sending direction (“DIR = 1”)	If the corresponding FIFO buffer is available for sending data, this module sends data in response to a token from the USB host. If sending data is not possible, this module sends an NAK response.
	Isochronous (“TYPE = 11”)	Receiving direction (“DIR = 0”)	For an OUT token from the USB host, if the FIFO buffer corresponding to the specified PIPE can receive data, this module receives data. If receiving data is not possible, this module discards data.
		Sending direction (“DIR = 1”)	If the corresponding FIFO buffer is available for sending data, this module sends data in response to a token from the USB host. If sending data is not possible, this module sends a zero-length packet.
10 (STALL) or 11 (STALL)	Bulk (“TYPE = 01”) or Interrupt (“TYPE = 10”)	Independent of the setting value	Sends a STALL response for a token from the USB host.
	Isochronous (“TYPE = 11”)	Independent of the setting value	Does not respond to a token from the USB host.

(2) PIPEn Control Register (USB20_FUNC_PIPEnCTR) (n = 6 to 9)

Access Size : 16 bits

Address : <USB20_func_base> + 0070h + (n - 1) x 0002h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	-	-	-	-	-	ACLRM	SQCLR	SQSET	SQMON	PBUSY	-	-	-	PID[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0
R/W	R	R	R	R	R	R	RW	R0W1	R0W1	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0h	R	Buffer status This bit indicates the FIFO buffer status of the specified pipe. 0b: Buffer access is not possible. 1b: Buffer access is possible.
14 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9	ACLRM	0h	RW	Automatic buffer clear mode This bit prohibits or enables automatic buffer clear mode for the specified pipe. 0b: Automatic buffer clear mode prohibited 1b: Automatic buffer clear mode enabled (all buffers are initialized)
8	SQCLR	0h	R0W1	The read value is always 0b. Toggle bit clear Specify 1b in this bit to clear the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA0. 0b: Disabled 1b: DATA0 specified
7	SQSET	0h	R0W1	The read value is always 0b. Toggle bit set Specify 1b in this bit to set the expected value of the sequence toggle bit for the next transaction in the specified pipe to DATA1 0b: Disabled 1b: DATA1 specified
6	SQMON	0h	R	Toggle bit monitor This bit indicates the expected value of the sequence toggle bit for the next transaction in the specified pipe. 0b: DATA0 1b: DATA1
5	PBUSY	0h	R	Pipe busy This bit indicates whether the specified pipe is being used in the USB bus. 0b: The specified pipe is not used in the USB bus. 1b: The specified pipe is being used in the USB bus.
4 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
1, 0	PID[1:0]	0h	RW	Response PID These bits specifies the response method for the next transaction in the specified pipe 00b: NAK response 01b: BUF response (according to the buffer status) 10b: STALL response 11b: STALL response

(a) Buffer status (BSTS) bit

See **(b) Transmit buffer monitor (INBUFM) bit**.

(b) Automatic buffer clear mode (ACLRM) bit

When you need to clear the whole FIFO buffer allocated to the selected pipe, write 1 and 0 successively to the ACLRM bit.

Table 6.5-26 shows the buffer contents this module clears when 1 and 0 are written successively to the ACLRM bit.

Table 6.5-27 shows the cases that require this processing.

Table 6.5-26 Buffer Contents This Core Clears when the ACLRM Bit is Set

No.	Contents Cleared by ACLRM Bit Operation
(1)	All contents of the FIFO buffer allocated to the selected pipe

Table 6.5-27 Cases Requiring the ACLRM Bit to be Set

No.	Contents Cleared by ACLRM Bit Operation
(1)	When clearing contents of the FIFO buffer allocated to the selected pipe
(2)	When resetting the interval counter
(3)	When the value of the BFRE bit is changed
(4)	When the transaction count function is terminated forcibly

Changing the value of this bit for a selected pipe must be done while the PID of the selected pipe is NAK and the selected pipe is not specified in the CURPIPE bits.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

(c) Sequence toggle bit clear (SQCLR) bit

See **(e) Sequence toggle bit clear (SQCLR) bit**.

(d) Sequence toggle bit set (SQSET) bit

See **(f) Sequence toggle bit set (SQSET) bit**.

(e) Sequence toggle bit monitor (SQMON) bit

See **(g) Sequence toggle bit monitor (SQMON) bit**.

(f) Pipe busy (PBUSY) bit

See **(h) Pipe busy (PBUSY) bit**.

(g) Response PID (PID) bits

See **(i) Response PID (PID) bits**.

6.5.3.2.17 Transaction Counters

(1) PIPEn Transaction Counter Enable Register (USB20_FUNC_PIPEnTRE) (n = 1 to 5)

Access Size : 16 bits

Address : <USB20_func_base> + 0090h + (n - 1) x 0004h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	TRENB	TRCLR	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	RW	R0W1	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9	TRENB	0h	RW	Transaction counter enable This bit enables or disables the transaction counter. 0b: Disables the transaction counter. 1b: Enables the transaction counter.
8	TRCLR	0h	R0W1	The read value is always 0b. Transaction counter clear This bit clears the transaction counter to 0b. To clear the counter, write 1b to this bit. 0b: Invalid 1b: Clears the current transaction counter.
7 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(a) Transaction counter enable (TRENB) bit

When software sets this bit for the selected pipe in the receiving direction after specifying a total number of packets in the TRNCNT bits, this module performs the following control when it finishes receiving the same number of packets as the total number specified in the TRNCNT bits:

1. When the continuous transfer mode is used (CNTMD = 1), this module switches the FIFO buffer to the CPU side at the end of reception even if the FIFO buffer is not full.
2. When the SHTNAK bit is 1, this module changes the PID of the selected pipe to NAK when it finishes receiving the same number of packets as the total number specified in the TRNCNT bits.
3. When the DENDE bit is 1 and the PKTMD bit is 0, this module asserts the DEND signal when reading the last data after having received the same number of packets as the total number specified in the TRNCNT bits.
4. When the BFRE bit is 1, this module asserts the BRDY interrupt signal when it finishes reading the last data after having received the same number of packets as the total number specified in the TRNCNT bits.

For the pipe in the sending direction, write 0 to this bit (TRENB bit). When not using the transaction count function, write 0 to this bit.

When using the transaction count function, specify a value in the TRNCNT bits before writing 1 to this bit. Also, write 1 to this bit before receiving the first packet among those to be counted by the transaction count function.

(b) Transaction counter clear (TRCLR) bit

When software sets this bit for a selected pipe, this module clears the current value of the transaction counter corresponding to the selected pipe, and then clears this bit.

(2) PIPEn Transaction Counter Register (USB20_FUNC_PIPEnTRN) (n = 1 to 5)

Access Size : 16 bits
Address : <USB20_func_base> + 0092h + (n - 1) x 0004h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRNCNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TRNCNT[15:0]	0h	RW	When writing: Specifies the total number of packets to be received by the pertinent pipe (number of transactions). When reading: Indicates the specified number of transactions if TRENB is 0b. Indicates the number of the currently counted transaction if TRENB is 1b.

(a) Transaction counter (TRNCNT) bits

When software writes 1 to the TRENB bit after setting the total number of packets to be received for the selected pipe in the receiving direction, this module performs the control described in **(a) Transaction counter enable (TRENB) bit**.

When the TRENB bit is 0, this module indicates, by these bits, the number of transactions set.

When the TRENB bit is 1, this module indicates, by these bits, the current number of transactions counted.

This module increments the value of the TRNCNT bits by 1 when the status of reception meets all the following conditions a) to c):

a) The TRENB bit is 1.

b) When a packet is received, the value of the TRCNT bits is not equal to “current count + 1.”

c) The payload size of received packets has reached the value of the MXPS bits.

This module clears the TRNCNT bits to 0 when any of the following conditions (1) to (3) is met:

(1) All the following conditions a) to c) are met:

a) The TRENB bit is 1.

b) When a packet is received, the value of the TRCNT bits is equal to “current count + 1. ”

c) The payload size of received packets has reached the value of the MXPS bits.

(2) Both of the following conditions a) and b) are met:

a) The TRENB bit is 1.

b) A short packet has been received.

(3) The following condition is met:

a) Written 1 to the TRCLR bit.

For the pipe in the sending direction, write 0 to these bits (TRNCNT bits). When not using the transaction count function, write 0 to these bits.

Changing the value of these bits for a selected pipe must be done while the PID of the selected pipe is NAK and the TRENB bit is 0.

If you change the value of this bit for a selected pipe after having changed the PID of the selected pipe from BUF to NAK, check, that, and the PBUSY bit is 0 before changing the value of this bit. Note, however, that, if this module has changed the PID of the selected pipe to NAK, the value of the PBUSY bit need not be checked.

If you change the value of these bits, write 1 to the TRCLR bit before writing 1 to the TRENB bit.

6.5.3.2.18 Low Power Control Register

(1) Low Power Control Register (USB20_FUNC_LPCTRL)

Access Size : 16 bits
Address : <USB20_func_base> + 0100h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	HWUPM	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
7	HWUPM	0h	RW	0b: Resumes the PHY from the low-power mode while the internal bus clock (P1φ) is operating. 1b: Enables resume from the low-power mode while the internal bus clock (P1φ) is stopped.
6 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(a) HWUPM

This bit is used to specify whether to enable resumption from the low-power mode even while the internal bus clock (P1ϕ) is stopped.

0: Disables resumption while the internal bus clock (P1ϕ) is stopped.
1: Enables resumption while the internal bus clock (P1ϕ) is stopped.

This bit specifies whether to detect resume signaling while the internal bus clock (P1ϕ) is stopped. Whether to resume is controlled by the L1EXTMD bit. To resume from the low-power mode (LPM L1 state) while the internal bus clock (P1ϕ) is stopped, set both this bit and the L1EXTMD bit.

6.5.3.2.19 Low Power Status Register

(1) Low Power Status Register (USB20_FUNC_LPSTS)

Access Size : 16 bits
Address : <USB20_func_base> + 0102h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	SUSPM	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
14	SUSPM	0h	RW	USBPHY Suspend M control This bit controls the Suspend M signal to the USBPHY. 0b: USBPHY suspend mode 1b: USBPHY normal mode
13 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(a) USB2PHY SuspendM control (SUSPM) bit

This bit is used to control the SuspendM signal to the USB2PHY. By default, the value of this bit is 0 and the USB2PHY is in suspend mode. To operate this module, write 1 to this bit.

When the SUSPM bit is 0 (that is, when the UTMI clock is stopped), data cannot be written to this module, but can only be read from this module. Note, however, that data can be written to the registers listed in **Table 6.5-28**.

Table 6.5-28 Registers That Allow Writing when the SUSPM Bit is 0

Address	Register Name
000h	SYSCFG0
002h	BUSWAIT
100h	LPCTRL
102h	SUSPMODE

Note that the values written to the SYSCFG0 register while the USB2PHY clock is stopped (SUSPM = 0) will be applied after the USB2PHY clock starts (SUSPM = 1).

When the L1EXTMD bit is 0, this bit (SUSPM bit) is controlled (set or cleared) by software. When L1EXTMD bit is 1, this bit is controlled by software for the transition to the L1 or L2 state, and controlled by hardware for resumption from the L1 or L2 state, regardless of the level (L1 or L2).

6.5.3.2.20 PHY Function Control Register

(1) PHY Function Control Register (USB20_FUNC_PHYFUNCTR)

Access Size : 16 bits

Address : <USB20_func_base> + 0104h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	SusMon	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
14	SusMon	0h	R	This bit allows reading of the status of the Suspend M signal.
13 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(a) SusMon

SuspendM monitor bit (read only)

The status of the Suspend M signal can be read.

6.5.3.2.21 PHY_OTG Control Register

(1) PHY_OTG Control Register (USB20_FUNC_PHYOTGCTR)

Access Size : 16 bits
Address : <USB20_func_base> + 010Ah
Initial Value : 0600h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	DmPuDwn	DpPuDwn	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
10	DmPuDwn	1h	R	Dm Pulldown monitor bit
9	DpPuDwn	1h	R	Dp Pulldown monitor bit
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(a) DmPuDwn

DmPulldown monitor bit (read only)

0: 15-kΩ Pulldown resistor control on the DM side is disabled.
1: 15-kΩ Pulldown resistor control on the DM side is enabled.

(b) DpPuDwn

DpPulldown monitor bit (read only)

0: 15-kΩ Pulldown resistor control on the DP side is disabled.
1: 15-kΩ Pulldown resistor control on the DP side is enabled.

6.5.3.2.22 Peripheral L1 Control Register 1

(1) Peripheral L1 Control Register 1 (USB20_FUNC_PL1CTRL1)

Access Size : 16 bits
Address : <USB20_func_base> + 0144h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	L1EXTMD	-	-	HIRDTHR[3:0]				DVSQ3	DVSQ(2:0)[2:0]			L1NEGOMD	L1RESPMD[1:0]		L1RESPEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	–	R	R	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
14	L1EXTMD	0h	–	USBPHY control mode on resumption from the L1 state This bit controls the USBPHY resume operation on resumption from the L1 state. 0b: Does not set the Suspend M bit when the Host K signal is received. 1b: Sets the Suspend M bit when the Host K signal is received
13, 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
11 to 8	HIRDTHR[3:0]	0h	RW	L1 response negotiation threshold HIRD threshold to be used for the L1NEGOMD bit The format is the same as the HIRD field of the HL1CTRL register.
7	DVSQ3	0h	R	DVSQ extension bit This bit combined with the device state (DVSQ[2:0]) bit indicates the L1 state. 0000b: Powered state 0001b: Default state 0010b: Address state 0011b: Configured state 01xxb: Suspended state 10xxb: L1 state
6 to 4	DVSQ(2:0)[2:0]	0h	R	These bits mirror the DVSQ[2:0] bits in INTSTS0.
3	L1NEGOMD	0h	RW	L1 response negotiation control This bit sets the negotiation function using for the HIRD value. 0b: Returns an ACK response if the received HIRD value is larger than the value of the HIRDTHR[3:0] bits. In other cases (including same values), an NYET response is returned. 1b: Returns an ACK response if the received HIRD value is smaller than the value of the HIRDTHR[3:0] bits. In other cases (including same values), an NYET response is returned. This bit is valid only when the value of the L1RESPMD[1:0] bit is 11b.
2, 1	L1RESPMD[1:0]	0h	RW	L1 response mode These bits specify how to respond to an LPM token. 00b: NYET 01b: ACK 10b: STALL 11b: Response according to the value of the L1NEGOMD bit
0	L1RESPEN	0h	RW	L1 response enable This bit enables an L1 response. 0b: Does not support LPM. 1b: Supports LPM.

(a) L1 EXT mode (L1EXTMD) bit

This bit specifies how to control the SuspendM bit upon receiving the Host K signal when the USB2PHY is stopped by setting the SuspendM bit in the L1 state.

0: Does not set the SuspendM bit when this module is resumed from the L1 state.
1: Sets the SuspendM bit when this module is resumed from the L1 state.

Remark 1. The Host K period lasts a minimum of 50 µs. Therefore, the USB2PHY might be unable to be resumed within the Host K period if the software settings for resume same as those for the suspend state are applied. Because the initial value of this bit is controlled by software, set this bit at initialization when the L1 state is to be supported.
For the transition to the L1 state, the SuspendM bit is controlled by software regardless of the value of this bit.

Remark 2. When this bit is set, the SuspendM bit will be set also at resumption from the L2 state.

(b) HIRD negotiation threshold (HIRDTHR[3:0]) bits

These bits specify the value of HIRD threshold to be used for the negotiation specified by the L1NEGOMD bit.

The format of the value is the same as that of the HIRD field in the HL1CTRL register.

(c) Device state extension (DVSQ[3]) bit

This bit is used as the fourth bit for the device state (DVSQ) bits.

0000b: Powered state
0001b: Default state
0010b: Address state
0011b: Configured state
01xxb: Suspended state
10xxb: L1 state

(d) Device status (DVSQ[2:0]) bits

These bits mirror the DVSQ[2:0] bits in the Interrupt Status Register (INTSTS0).

(e) L1 negotiation mode (L1NEGOMD) bit

This bit is used to specify the negotiation function using the HIRD value.

0: Returns an ACK response when the received HIRD value is larger than the value in HIRDTHR[3:0] bits, or returns an NYET response in other cases.

1: Returns an ACK response when the received HIRD value is smaller than the value in HIRDTHR[3:0] bits, or returns an NYET response in other cases.

This bit is valid only when the value of L1RESPMD[1:0] bits is 11b.

(f) L1 response mode (L1RSPMD[1:0]) bits

When the L1RSPED bit is set, this module respond to an LPM token according to the value of these bits. These bits specify how to respond to the LPM token.

00b: NYET
01b: ACK
10b: STALL
11b: Response according to the value of L1NEGOMD bit

(g) L1 response enable (L1RSPEN) bit

When this bit is 0, this module does not respond to the LPM token it receives. When this bit is 1, this module responds to the LPM token (it receives) according to the value of the LPMRESPMD[1:0] bits.

6.5.3.2.23 Peripheral L1 Control Register 2

(1) Peripheral L1 Control Register 2 (USB20_FUNC_PL1CTRL2)

Access Size : 16 bits

Address : <USB20_func_base> + 0146h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	RWEMON	HIRDMON[3:0]				-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
12	RWEMON	0h	RW	This bit reflects the value of the RWE bit in the LPM token received last.
11 to 8	HIRDMON[3:0]	0h	RW	These bits reflect the value of the HIRD field in the LPM token received last.
7 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(a) RWE value monitor (RWEMON) bit

This bit is referenced to monitor the value of the RWE bit in a received LPM token.

This bit reflects the value of the RWE bit in the LPM token received last.

(b) HIRD value monitor (HIRDMON) bits

These bits are referenced to monitor the value of the HIRD field in a received LPM token.

These bits reflect the value of the HIRD field in the LPM token received last.

6.5.3.2.24 Next Register Set

(1) Next Source Address Register n

(a) Next k Source Address Register ch n (USB20_FUNC_NkSA_n) (k = 0, 1; n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 0400h + k x 000Ch + n x 00040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SA (in normal mode), WD (in writeonly mode)[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SA (in normal mode), WD (in writeonly mode)[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SA (in normal mode), WD (in writeonly mode)[31:0]	0h	RW	SA (in normal mode): Source Address These bits specify the start address of the DMA transfer source. WD (in write-only mode): Write Data These bits specify the write data in write-only mode.

Note: In a transfer in link mode, the data in the N0SA_n register is overwritten with the descriptor read data.

(2) Next Destination Address Register n

(a) Next k Destination Address Register ch n (USB20_FUNC_NkDA_n) (k = 0, 1; n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 0404h + k x 000Ch + n x 00040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DA[31:0]	0h	RW	Destination Address These bits specify the start address of the DMA transfer destination.

Note: In a transfer in link mode, the data in the N0DA_n register is overwritten with the descriptor read data.

(3) Next Transaction Byte Register n

(a) Next k Transaction Byte Register ch n (USB20_FUNC_NkTB_n) (k = 0, 1; n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 0408h + k x 000Ch + n x 00040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TB[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TB[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TB[31:0]	0h	RW	Transaction Byte These bits specify the total number of transfer bytes. Note: Do not start a DMA transaction with 0b set in these bits.

Note: The N0TB_n register is overwritten by the descriptor read data during link mode transfer.

6.5.3.2.25 Current Register Set

(1) Current Source Address Register

(a) Current Source Address Register ch n (USB20_FUNC_CRSA_n) (n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 0418h + n x 0040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRSA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRSA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRSA[31:0]	0h	R	Current Source Address Register This register indicates the read address for the next DMA transaction. The value of this register is incremented automatically while DMA transactions are in process. (When the SAD bit in the CHCFG_n register is 1b, the value of this register is fixed. When the WONLY bit in the CHCFG_n register is 1b, the value of this register is undefined.) The initial value of this register is loaded from the following register: In register mode: A transfer source address is loaded from the Next0/1 Register Set. In link mode: A transfer source address is loaded from the descriptor. (The descriptor read data is input to the N0SA_n register, and is loaded into the CRSA_n register when a transfer starts.) The value of this register is incremented when a read transfer ends. Read this register after DMA stops (that is, after the TACT bit in the CHSTAT_n register is cleared). (Handle the value read during DMA only as a reference value.)

(2) Current Destination Address Register

(a) Current Destination Address Register ch n (USB20_FUNC_CRDA_n) (n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 041Ch + n x 0040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRDA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRDA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRDA[31:0]	0h	R	Current Destination Address Register This register indicates the write address for the next DMA transaction. The value of this register is incremented automatically while DMA transactions are in process. (When the DAD bit in the CHCFG_n register is 1b, the value of this register is fixed.) The initial value of this register is loaded from the following register: In register mode: A transfer destination address is loaded from the Next0/1 Register Set. In link mode: A transfer destination address is loaded from the descriptor. (The descriptor read data is input to the N0DA_n register, and is loaded into the CRDA_n register when a transfer starts.) The value of this register is incremented when a write transfer ends. Read this register after DMA stops (that is, after the TACT bit in the CHSTAT_n register is cleared). (Handle the value read during DMA only as a reference value.)

(3) Current Transaction Byte Register

(a) Current Transaction Byte Register ch n (USB20_FUNC_CRTB_n) (n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 0420h + n x 0040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRTB[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRTB[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRTB[31:0]	0h	R	Current Transaction Byte Register This register indicates the remaining number of transfer bytes in the ongoing DMA transaction. The value of this register is decremented automatically while the DMA transaction is in process. The initial value of this register is loaded from the following register: In register mode: The number of transfer bytes is loaded from the Next0/1 Register Set. In link mode: The number of transfer bytes is loaded from the descriptor. (The descriptor read data is input to the N0TB_n register, and is loaded into the CRTB_n register when a transfer starts.) The value of this register is decremented when a write transfer ends. Read this register after DMA stops (that is, after the TACT bit in the CHSTAT_n register is cleared). (Handle the value read during DMA only as a reference value.)

6.5.3.2.26 Channel Register Set

(1) Channel Status Register n

(a) Channel Status Register ch n (USB20_FUNC_CHSTAT_n) (n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 0424h + n x 0040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DNUM[7:0]								-	-	-	-	-	SWPRQ	DMARQM	INTM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	MODE	DER	DW	DL	SR	TC	END	ER	SUS	TACT	RQST	EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DNUM[7:0]	0h	R	Data Number These bits indicate the amount of valid data in the buffer. The indicated amount of data is the amount of the data that was read from the source but has not yet been written to the destination. (Unit: byte) Incrementing condition: A DMA read transfer ends. Decrementing condition: A DMA write transfer ends. Clearing conditions: A condition for clearing the EN bit is met. 1b is written to the SWRST bit in the CHCTRL_n register.
23 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
18	SWPRQ	0h	R	Sweep Request This bit indicates the state of sweep request. This bit indicates the state of software sweep request (which has been activated by the SETSSWPRQ bit in the CHCTRL_n register). 1b: The sweep request signal has been asserted. 0b: The sweep request signal has not been asserted. Setting condition: The SETSSWPRQ bit in the CHCTRL_n register is asserted. Clearing conditions: The buffer becomes empty because of sweep. 1b is written to the CLRHSWPRQM bit in the CHCTRL_n register. 1b is written to the SWRST bit in the CHCTRL_n register.
17	DMARQM	0h	R	DMAREQ Mask This bit indicates the state of temporary masking of the DMA transfer request from the USB control. 1b: The request is temporarily masked. 0b: The request is released from temporary masking. Setting condition: The SETDMARQM bit in the CHCTRL_n register is set. Clearing conditions: 1b is written to the CLRDMARQM bit in the CHCTRL_n register. 1b is written to the SWRST bit in the CHCTRL_n register.

Bit	Bit Name	Initial Value	R/W	Description
16	INTM	0h	R	Interrupt Mask This bit indicates the state of temporary masking of the output from the USBFDMAmn interrupt. 1b: The output is temporarily masked. 0b: The output is released from temporary masking. Setting condition: 1b is written to the SETINTM bit in the CHCTRL_n register. Clearing conditions: 1b is written to the CLRINTM bit in the CHCTRL_n register. 1b is written to the SWRST bit in the CHCTRL_n register.
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
11	MODE	0h	R	DMA Mode This bit indicates the DMA mode. The indicated value is the value of the DMS bit in the CHCFG_n register. 0b: Register mode 1b: Link mode
10	DER	0h	R	Descriptor Error This bit indicates whether the data read from the descriptor is invalid (LV = 0) (regardless of the value of the DIM bit in the CHCFG_n register). 0b: No descriptor error has occurred. 1b: A descriptor error has occurred. Setting condition: In link mode, when the DRRP bit in the CHCFG_n register is 0b, the LV bit value read from the descriptor is 0b. Clearing conditions: 1b is written to the CLRDER bit in the CHCTRL_n register. 1b is written to the SWRST bit in the CHCTRL_n register.
9	DW	0h	R	Descriptor Writeback This bit indicates whether data is being written back to the descriptor. If a bus error occurs during write-back to the descriptor, this bit retains 1b. 0b: Status other than write-back to the header in link mode 1b: (When the ER bit in the CHSTAT_n register is 0b) Data is being written back to the header in link mode. (When the ER bit in the CHSTAT_n register is 1b) A bus error has occurred during write-back to the header in link mode. Setting condition: Write-back to the header is started in link mode. Clearing conditions: Write-back to the header in link mode ends with an OK response. 1b is written to the SWRST bit in the CHCTRL_n register. If the bit retains 1b because of an error response, this bit can be cleared only by setting the SWRST bit.
8	DL	0h	R	Descriptor Load This bit indicates whether data is being read from the descriptor. If a bus error occurs during descriptor reading, this bit retains 1b. 0b: Status other than descriptor reading 1b: (When the ER bit is 0b) Descriptor reading is in process in link mode. (When the ER bit is 1b) A bus error has occurred during descriptor reading in link mode. Setting condition: Descriptor reading is started in link mode. Clearing conditions: Descriptor reading in link mode ends with an OK response. 1b is written to the SWRST bit in the CHCTRL_n register. If the bit retains 1b because of an error response, this bit can be cleared only by setting the SWRST bit.
7	SR	0h	R	Selected Register Set In register mode, this bit indicates the register set that is selected. 0b: Next0 Register Set 1b: Next1 Register Set Setting condition: The RSEL bit in the CHCFG_n register is set. Clearing condition: The RSEL bit in the CHCFG_n register is cleared.

Bit	Bit Name	Initial Value	R/W	Description
6	TC	0h	R	Terminal Count This status bit indicates whether the DMA transaction has ended. This bit is set only when the TCM bit in the CHCFG_n register is 0b. 0b: The DMA transfer has not ended. 1b: The DMA transfer has ended. Setting conditions: In register mode, transfer of the total number of transfer bytes specified in the CRTB bits ends. In link mode, when the WBD bit in the header of the descriptor is 1b, transfer of the total number of transfer bytes specified in the CRTB bits ends. In link mode, when the WBD bit in the header of the descriptor is 0b, write-back to the descriptor ends. Clearing conditions: 1b is written to the CLRTC bit in the CHCTRL_n register. 1b is written to the SWRST bit in the CHCTRL_n register.
5	END	0h	R	USBFDMAmn Interrupted This bit indicates whether the DMA transaction has ended and an USBFDMAmn interrupt has occurred. 0b: The DMA transfer has not ended. 1b: The DMA transfer has ended. Setting conditions: The condition for setting the TC bit is met, and the DEM bit in the CHCFG_n register is 0b. In link mode, when the descriptor is read, the LV bit in the header is 0b, and the DRRP and DIM bits in the CHCFG_n register are 0b. Clearing conditions: 1b is written to the CLREND bit in the CHCTRL_n register. 1b is written to the SWRST bit in the CHCTRL_n register.
4	ER	0h	R	Error This bit indicates whether an error response has been received and a DMAERR interrupt has occurred during the DMA transfer. 0b: No error response has been received. 1b: An error response has been received. Setting condition: An error response is received in a bus cycle. Clearing condition: 1b is written to the SWRST bit in the CHCTRL_n register.
3	SUS	0h	R	Suspend This bit indicates whether the channel is suspended. For details, see **6.5.3.9.13(8)(b) Suspension**. 0b: Channel_n is not suspended. 1b: Channel_n is suspended. Setting condition: 1b is written to the SETSUS bit in the CHCTRL_n register during the DMA transfer using channel_n, and, thereby, the inside of the channel is suspended. Clearing conditions: 1b is written to the CLRSUS bit in the CHCTRL_n register. 1b is written to the CLREN bit in the CHCTRL_n register. A condition for clearing the EN bit in the CHSTAT_n register.
2	TACT	0h	R	Transaction Active This bit indicates whether the DMAC is operating. This bit is used to check whether the channel is stopped fully. For details, see **6.5.3.9.13(8) Transfer state**. 0b: The DMA in channel_n is stopped. 1b: The DMA in channel_n is operating. Setting condition: 1b is written to the SETEN bit in the CHCTRL_n register (to start descriptor reading or wait for a DMA request). Clearing condition: The internal state is the idle state (the EN bit in the register has been cleared, and all transfers have ended).

Bit	Bit Name	Initial Value	R/W	Description
1	RQST	0h	R	Request This bit indicates whether a transfer request has been received. 0b: No DMA transfer request has been received. 1b: A DMA transfer request has been received. Setting conditions: 1b is written to the STG bit in the CHCTRL_n register. A DMA transfer request is received from the USB control. Clearing conditions: 1b is written to the SWRST bit in the CHCTRL_n register. 1b is written to the CLRRQ bit in the CHCTRL_n register. In single transfer mode (the TM bit in the CHCFG_n register is 0b), a transfer is executed on the side specified by the REQD bit in the CHCFG_n register In register mode, all DMA transactions are complete (the REN bit in the CHCFG_n register is 0b). In link mode, the DMA transfer of the last descriptor (LE = 1) ends. In link mode, a DMA transfer is stopped during descriptor reading (the LV bit is 0b and the DRRP bit in the CHCFG_n register is 0b). In link mode, when the DEM bit in the CHCFG_n register is 0b, a DMA transaction ends. The master interface receives a bus error signal.
0	EN	0h	R	Enable This bit indicates whether the operation of DMA channel n is enabled or stopped. 0b: Operation is stopped. 1b: Operation is enabled. Setting conditions: 1b is written to the SETEN bit in the CHCTRL_n register. Clearing conditions: 1b is written to the SWRST bit in the CHCTRL_n register. 1b is written to the CLREN bit in the CHCTRL_n register. An error response is received during transfer. In register mode, all DMA transactions are completed (the REN bit in the CHCFG_n register is 0b). In link mode, DMA transfer of the last descriptor (the LE bit is 1b) ends (if the WBD bit is 0b, write- back to the descriptor ends). In link mode, reading of a descriptor is stopped (the LV bit is 0b and the DRRP bit in the CHCFG_n register is 0b).

Note: When the ER bit in the CHSTAT_n register is set, treat the corresponding series of transfers as invalid transactions.

Note: To interrupt a DMA transaction, mask or clear transfer requests or clear the EN bit in the CHSTAT_n register. (For the procedure to interrupt, see **Section 6.5.3.3.13(8) (c), Transfer suspension**.)

Note: If the DMA transfer request from the USB control and the transfer request by software (setting the STG bit in the CHCFG_n register) are used together, the cause of activating the request that is enabled cannot be identified. Therefore, design the system so that only one type of transfer requests is used.

Note: When using the transfer request by software, operate the STG bit for a new transfer request only after the DMA transfer requested last ends (after checking the end of the last DMA transfer by referencing the Current Register Set or another method).

(2) Channel Control Register n

(a) Channel Control Register ch n (USB20_FUNC_CHCTRL_n) (n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 0428h + n x 0040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	CLRDMARQM	SETDMARQM	CLRINTM	SETINTM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	SETSSWPRQ	-	SETREN	-	-	CLRSUS	SETSUS	CLRDER	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	R	RW	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
19	CLRDMARQM	0h	RW	Clear DMAREQ Mask Writing 1b to this bit releases the DMA transfer requests from the USB control from temporary masking. Writing 1b to this bit also clears the DMARQM bit in the CHSTATn register. When this bit is read, 0b is read. 1b: Releases the DNA transfer requests from masking set by using the SETDMARQM bit. 0b: Has no effect on operation.
18	SETDMARQM	0h	RW	SET DMAREQ Mask Writing 1b to this bit temporarily masks the DMA transfer requests from the USB control. Writing 1b to this bit also sets the DMARQM bit in the CHSTATn register. When this bit is read, 0b is read. 1b: Masks the DMA transfer requests from the USB control. 0b: Has no effect on operation.
17	CLRINTM	0h	RW	Clear Interrupt Mask Writing 1b to this bit releases the USBFDMAmn interrupt from masking. Writing 1b to this bit also clears the INTM bit in the CHSTATn register. Releasing the INT_DMA[n] pin output from masking when the LVINT bit in the DCTRL register and the END bit in the CHSTAT_n register are 1b activates the INT_DMA[n] pin output. (The pin output is not activated if the LVINT bit is 0b.) When this bit is read, 0b is read. 1b: Releases the pin output from masking set by using the SETINTM bit. 0b: Has no effect on operation.
16	SETINTM	0h	RW	SETINTMSet Interrupt Mask Writing 1b to this bit temporarily masks the USBFDMAmn interrupt. Writing 1b to this bit also sets the INTM bit in the CHSTATn register. When this bit is read, 0b is read. 1b: Masks the USBFDMAmn interrupt. 0b: Has no effect on operation.
15	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
14	SETSSWPRQ	0h	RW	Set Software Sweep Request Writing 1b to this bit sweeps out the data stored in the buffer to the destination (see **6.5.3.9.13(3)(a) Forced software sweeping request**). When this bit is read, 0b is read. 1b: Writes, to the destination, the data that is stored in the buffer and has not yet been written to the destination. 0b: Has no effect on operation. If the destination asserts a hardware request (REQD = 1), the sweep operation cannot be used.

Bit	Bit Name	Initial Value	R/W	Description
13	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
12	SETREN	0h	RW	Set Register Set Enable Writing 1b to this bit sets the REN bit in the CHCFG_n register. When this bit is read, 0b is read. 1b: Sets the REN bit in the CHCFG_n register. 0b: Has no effect on operation.
11, 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9	CLRSUS	0h	RW	Clear Suspend Writing 1b to this bit when the SUS bit in the CHSTAT_n register is 1b releases the ongoing DMA transfer from the suspended state. When this bit is read, 0b is read. 1b: Releases the ongoing DMA transfer from the suspended state. 0b: Has no effect on operation.
8	SETSUS	0h	RW	Set Suspend Writing 1b to this bit when the EN bit in the CHSTAT_n register is 1b suspends the ongoing DMA transfer. When this bit is read, 0b is read. 1b: Suspends the ongoing DMA transfer. 0b: Has no effect on operation.
7	CLRDER	0h	RW	Clear DER Writing 1b to this bit clears the DER bit in the CHSTAT_n register. Writing 1b to this bit also clear the USBFDMAmn interrupt. When this bit is read, 0b is read. 1b: Clears the DER bit. 0b: Has no effect on operation.
6	CLRTC	0h	RW	Clear TC Writing 1b to this bit clears the TC bit in the CHSTAT_n register. When this bit is read, 0b is read. 1b: Clears the TC bit. 0b: Has no effect on operation.
5	CLREND	0h	RW	Clear End Writing 1b to this bit clears the END bit in the CHSTAT_n register. Writing 1b to this bit also clear the USBFDMAmn interrupt. When this bit is read, 0b is read. 1b: Clears the END bit. 0b: Has no effect on operation.
4	CLRRQ	0h	RW	Clear Request Writing 1b to this bit clears the RQST bit in the CHSTAT_n register. When this bit is read, 0b is read. 1b: Clears the RQST bit in the CHSTAT_n register. 0b: Has no effect on operation.
3	SWRST	0h	RW	Software Reset Writing 1b to this bit clears individual bits in the CHSTAT_n register (for the bits to be cleared, see the description of each bit). Setting this bit must be done when the EN and TACT bits are 0b. When this bit is read, 0b is read. 1b: Clears individual bits in the CHSTAT_n register. 0b: Has no effect on operation.
2	STG	0h	RW	Software Trigger Writing 1b to this bit makes software set an internal transfer request. If this bit and the SWRST bits are set at the same time, clearing by the SWRST bit takes priority. When this bit is read, 0b is read. 1b: Makes software set a transfer request (set the RQST bit in the CHSTAT_n register). 0b: Has no effect on operation.
1	CLREN	0h	RW	Clear Enable Writing 1b to this bit clears the EN bit in the CHSTAT_n register (for details, see **6.5.3.9.13(8)(c) Transfer suspension**). When this bit is read, 0b is read. 1b: Disables DMA transfers (clears the EN bit in the CHSTAT_n register). 0b: Has no effect on operation.

Bit	Bit Name	Initial Value	R/W	Description
0	SETEN	0h	RW	Set Enable Writing 1b to this bit enables DMA transfers in DMA channel n. If this bit and the SWRST bits are set at the same time, clearing by the SWRST bit takes priority, and DMA transfers do not start. When this bit is read, 0b is read. 1b: Enables DMA transfers (sets the EN bit in the CHSTAT_n register). 0b: Has no effect on operation.

Note: Temporary masking (using the CLRDMARQM, and SETDMARQM bits) of the DMA transfer requests from the USB control applies to only the resources for channel n. Setting the SETDMARQM bit for channel n does not affect the operation of channel m.

(3) Channel Configuration Register

(a) Channel Configuration Register ch n (USB20_FUNC_CHCFG_n) (n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 042Ch + n x 0040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMS	REN	RSW	RSEL	SBE	DIM	TCM	DEM	WONLY	-	DAD	SAD	DDS[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDS[3:0]				DRRP	-	-	-	-	-	-	-	REQD	-	-	SEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	RW	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	DMS	0h	RW	DMA Mode Select This bit specifies the DMA mode to be used. 0b: Register mode (default) 1b: Link mode
30	REN	0h	RW	Register Set Enable This bit specifies whether to successively perform, after a DMA transaction ends, another DMA transaction using the Next Register Set selected by the RSEL bit. This bit is valid only in register mode. 0b: Does not perform the DMA transaction successively. 1b: Performs the DMA transaction successively. Setting conditions: 1b is written to this bit. 1b is written to the SETREN bit in the CHCTRL_n register. Clearing conditions: 0b is written to this bit. The REN bit is 1b, and a DMA transaction ends. To re-set the REN bit during a transaction, we recommend you to use the SETREN bit in the CHCTRL_n register.
29	RSW	0h	RW	Register Select Switch This bit specifies whether to automatically invert the RSEL bit after a DMA transaction ends. This bit is valid only in register mode. 0b: Does not invert the RSEL bit after a DMA transaction ends. (Default) 1b: Inverts the RSEL bit after a DMA transaction ends.
28	RSEL	0h	RW	Register Set Select This bit is used to select the Next Register Set to be used for the next DMA transaction. This bit is valid only in register mode. When the RSW bit is 1, this bit is inverted automatically at the end of a DMA transaction. 0b: Uses the Next0 Register Set. (Default) 1b: Uses the Next1 Register Set. Transition condition: A DMA transaction ends with the RSW bit set.
27	SBE	0h	RW	Sweep Buffer Enable This bit specifies whether to sweep (write) the data already read and stored in the buffer and stop transfer when the EN bit in the CHSTAT_n register is cleared during a DMA transaction. The sweep mode can be used only when the REQD bit is 0b. 0b: Stops transfer without sweeping out the buffer. (Default) 1b: Stops transfer after sweeping out the buffer.

Bit	Bit Name	Initial Value	R/W	Description
26	DIM	0h	RW	Descriptor Interrupt Mask This bit specifies whether to mask the USBFDMAmn interrupt when the LV bit value read from descriptor header is 0b. 0b: Does not mask the USBFDMAmn interrupt. (Default) 1b: Mask the USBFDMAmn interrupt.
25	TCM	0h	RW	DMATC Mask This bit is used to mask the DMATC signal, which is sent from the DMAC to USB control. When this bit is 1b at the time the DMATC signal is to be output, the DMATC signal is not asserted. Also, the TC bit in the CHSTAT_n register is not asserted in that case. In register mode, this bit is automatically cleared. In link mode, this bit is not cleared automatically. Use this bit when you control DMA transfers by software. 0b: Does not mask the DMATC signal. (Default) 1b: Masks the DMATC signal. Clearing condition: A DMA transaction ends with the TCM bit set.
24	DEM	0h	RW	USBFDMAmn Mask When this bit is 1b at the time the USBFDMAmn interrupt is not asserted. Also, the END bit in the CHSTAT_n register is not asserted in that case. In register mode, this bit is not cleared automatically. In link mode, this bit is automatically cleared. 0b: Does not mask the USBFDMAmn interrupt. (Default) 1b: Masks the USBFDMAmn interrupt. Clearing condition: A DMA transaction ends with the DEM bit set.
23	WONLY	0h	RW	Write Only Mode This bit is used to switch the transfer operation mode to the write-only mode (see **6.5.3.9.12(2) Write only mode**). 0b: Normal operation (default) 1b: Write-only mode
22	-	0h	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
21	DAD	0h	RW	Destination Address Direction This bit specifies the direction of counting the transfer-destination address in DMA channel n. If the transfer destination is on the USB control side, write 1b (fixed) to this bit. 0b: Incrementing (default) 1b: Fixed When the transfer destination uses the skip mode or is beat-unaligned, do not specify 1b (fixed) in this bit.
20	SAD	0h	RW	Source Address Direction This bit specifies the direction of counting the transfer-source address in DMA channel n. If the transfer source is on the USB control side, write 1b (fixed) to this bit. 0b: Incrementing (default) 1b: Fixed When the transfer source uses the skip mode or is beat-unaligned, do not specify 1b (fixed) in this bit.
19 to 16	DDS[3:0]	0h	RW	Destination Data Size These bits specify the size of DMA transfer data. When the transfer destination is on the USB control side, select the normal mode. Use bit 3 to switch between the normal and skip modes. 0b: Normal mode (default) 1b: Skip mode Use bits 2 to 0 to specify the size of transfer data. (For specifiable values, see **Table 6.5-29**.) 000b: 8 bits (default) 001b: 16 bits 010b: 32 bits 011b: 64 bits 100b: 128 bits 101b: 256 bits 110b: 512 bits 111b: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	SDS[3:0]	0h	RW	Source Data Size These bits specify the size of DMA transfer data. Use bit 3 to switch between the normal and skip modes. 0b: Normal mode (default) 1b: Skip mode Use bits 2 to 0 to specify the size of transfer data. (For specifiable values, see **Table 6.5-29**.) 000b: 8 bits (default) 001b: 16 bits 010b: 32 bits 011b: 64 bits 100b: 128 bits 101b: 256 bits 110b: 512 bits 111b: Setting prohibited
11	DRRP	0h	RW	Descriptor Read Repeat This bit switches the operation to be performed when the LV value in the header read from the descriptor is 0b. (See **6.5.3.9.12(1)(b) b-1) Operation flow of link mode**.) 0b: This module sets the DER bit in the CHSTAT_n register, and then stops descriptor reading. (Default) 1b: This module keeps reading the same descriptor until the LV value changes to 1b, and, when the LV value becomes 1b, starts the DMA transfer using the values in the descriptor. The interval of descriptor reading is controlled by using the DSCITVL register.
10 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
3	REQD	0h	RW	Request Direction This bit specifies whether the USB control is on the transfer source side or it is on the transfer destination side. 0b: The USB control is on the transfer source side. (Default) 1b: The USB control is on the transfer destination side.
2, 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
0	SEL	0h	RW	Terminal Select This bit selects the FIFO channel to be used on the USB control side. 0b: D0FIFO 1b: D1FIFO

The range of values specifiable in the SDS[2:0] and DDS[2:0] bits depends on the data bus width, number of implemented buffer stages, and whether the address to be accessed is beat-aligned or not (beat-unaligned). The table below shows the range of specifiable values.

Table 6.5-29 Range of Sizes That Can Be Specified in SDS and DDS Bits

Transfer Address	REQD	SDS[2:0]	DDS[2:0]
Beat aligned	0	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.	8 to 32 bits (000 to 010) 128 to 512 bits (100 to 110)
	1	8 to 32 bits (000 to 010) 128 to 512 bits (100 to 110)	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.
Beat unaligned	0	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.	8 to 32 bits (000 to 010) 128 to 256 bits (100 to 101)
	1	8 to 32 bits (000 to 010) 128 to 256 bits (100 to 101)	This data size should be equal to that of the MBW bits in the DxFIFOSEL register.

Note: When the destination is beat-unaligned with REQD = 0 or the source is beat-unaligned with REQD = 1, specify values in the range of specifiable values for beat-unaligned transfer addresses in both of the SDS[2:0] and DDS[2:0] bits.
Even if the transfer source and destination are beat-aligned when a DMA transaction starts, they might become beat-unaligned in the middle of the transaction when a skip transfer is used. If this might occur, perform settings in the first place on the assumption that the transfer source and destination are beat-unaligned.
If software cannot determine whether the transfer source and/or destination is beat-aligned, use values in the range of specifiable values for beat-unaligned transfer addresses.

(4) Channel Interval Register n

For details, see **(6) Interval Count Function**.

(a) Channel Interval Register ch n (USB20_FUNC_CHITVL_n) (n = 0, 1)

Access Size : 32 bits
Address : <USB20_func_base> + 0430h + n x 0040h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ITVL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
15 to 0	ITVL[15:0]	0h	RW	Interval These bits specify the DMA transfer interval.

(5) Channel Extension Register n

(a) Channel Extension Register ch n (USB20_FUNC_CHEXT_n) (n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 0434h + n x 0040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	DPR[3:0]				-	-	-	-	SPR[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
19 to 12	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
11 to 8	DPR[3:0]	0h	RW	Destination PROT These bits specify the value to be output to the MHPROT[3:0] pin in a DMA write transfer. The initial value of these bits is 0h.
7 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
3 to 0	SPR[3:0]	0h	RW	Source PROT These bits specify the value to be output to the MHPROT[3:0] pin in a DMA read transfer. The initial value of these bits is 0h.

6.5.3.2.27 Link Register Set

When software sets a descriptor address in the NXLA_n register and starts the DMAC, hardware loads the value set in the NXLA_n register into the CRLA_n register. Then, the descriptor is read, and the DMAC starts a DMA transaction according to the values read from the descriptor. The value in the NXLA_n register is automatically updated to the Next Link Address value read from the descriptor, and the updated value is used as the descriptor address in the next DMA transaction.

(1) Next Link Address Register n

(a) Next Link Address Register ch n (USB20_FUNC_NXLA_n) (n = 0, 1)

Access Size : 32 bits
Address : <USB20_func_base> + 0438h + n x 0040h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NXLA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NXLA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NXLA[31:0]	0h	RW/R	Next Link Address These bits specify the link-destination address. Upper two bits are fixed to 0b, and only the address aligned with a word boundary can be set.

(2) Current Link Address Register n

(a) Current Link Address Register ch n (USB20_FUNC_CRLA_n) (n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 043Ch + n x 0040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRLA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRLA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRLA[31:0]	0h	R	Current Link Address These bits indicate the address of the descriptor being used for current transaction.

6.5.3.2.28 Skip Register Set

This register set is used to specify settings for a skip (scatter/gather) transfer.

(1) Source Continuous Register n

(a) Source Continuous Register ch n (USB20_FUNC_SCNT_n) (n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 0600h + n x 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SCNT[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SCNT[31:0]	0h	RW	Source Continuous These bits specify the size of the space to be accessed continuously by source address access. (Unit: byte)

Note: This register is used in pair with the SSKP_n register (see **Figure 6.5-6**). To use this mode, set the SDS[3] bits in the CHCFG_n register to 1b. To perform a skip transfer on the transfer source side, the SAD bit in the CHCFG_n register must not be set to 1b (fixed). Do not perform a skip transfer with the SCNT bits set to 0b.

(2) Source Skip Register n

(a) Source Skip Register ch n (USB20_FUNC_SSKP_n) (n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 0604h + n x 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SSKP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SSKP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SSKP[31:0]	0h	RW	Source Skip These bits specify the size of area to be skipped in a source address access. (Unit: byte)

Note: This register is used in pair with the SCNT_n register (see **Figure 6.5-6**). To use this mode, set the SDS[3] bits in the CHCFG_n register to 1b. To perform a skip transfer on the transfer source side, the SAD bit in the CHCFG_n register must not set to 1b (fixed).

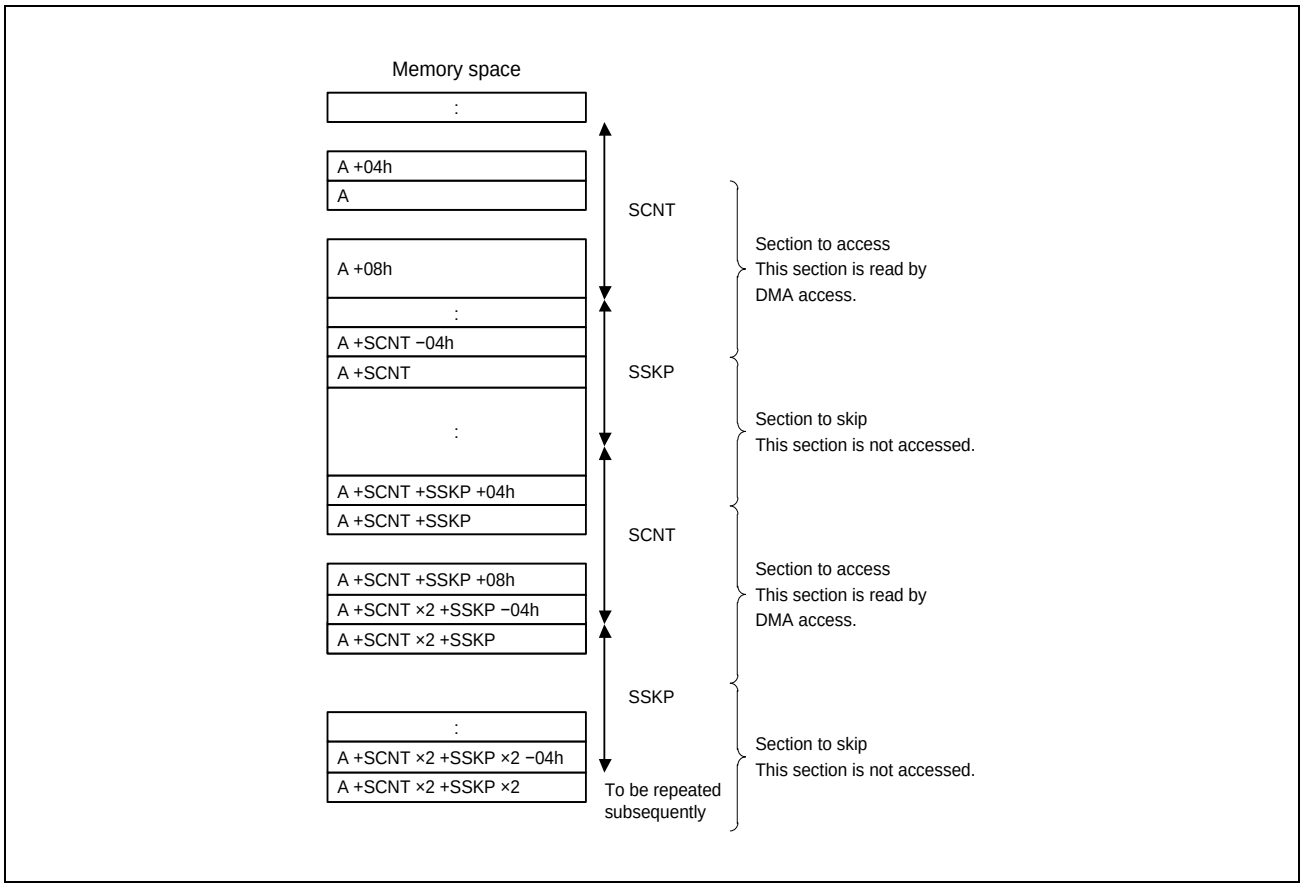


Figure 6.5-6 Relationship between SSKP and SCNT

You can specify values of the SCNT and SSKP bits regardless of the source address and the value of the SDS field in the CHCFG_n register. The DMAC performs access based on the size specified in the SDS field, and fetches only valid data into the buffer (see **(a) Read access**).

(3) Destination Continuous Register n

(a) Destination Continuous Register ch n (USB20_FUNC_DCNT_n) (n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 0608h + n x 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCNT[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DCNT[31:0]	0h	RW	Destination Continuous These bits specify the size of the space to be accessed continuously by destination address access. (Unit: byte)

Note: This register is used in pair with the DSKP_n register (see **Figure 6.5-7**). To use this mode, set the DDS[3] bits in the CHCFG_n register to 1b. To perform a skip transfer on the transfer destination side, the DAD bit in the CHCFG_n register must not be set to 1b (fixed). Do not perform a skip transfer with the DCNT bits set to 0b.

(4) Destination Skip Register n

(a) Destination Skip Register ch n (USB20_FUNC_DSKP_n) (n = 0, 1)

Access Size : 32 bits

Address : <USB20_func_base> + 060Ch + n x 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSKP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSKP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSKP[31:0]	0h	RW	Destination Skip These bits specify the size of area to be skipped in a destination address access. (Unit: byte)

Note: This register is used in pair with the DCNT_n register (see **Figure 6.5-7**). To use this mode, set the DDS[3] bits in the CHCFG_n register to 1b. To perform a skip transfer on the transfer destination side, the DAD bit in the CHCFG_n register must not be set to 1b (fixed).

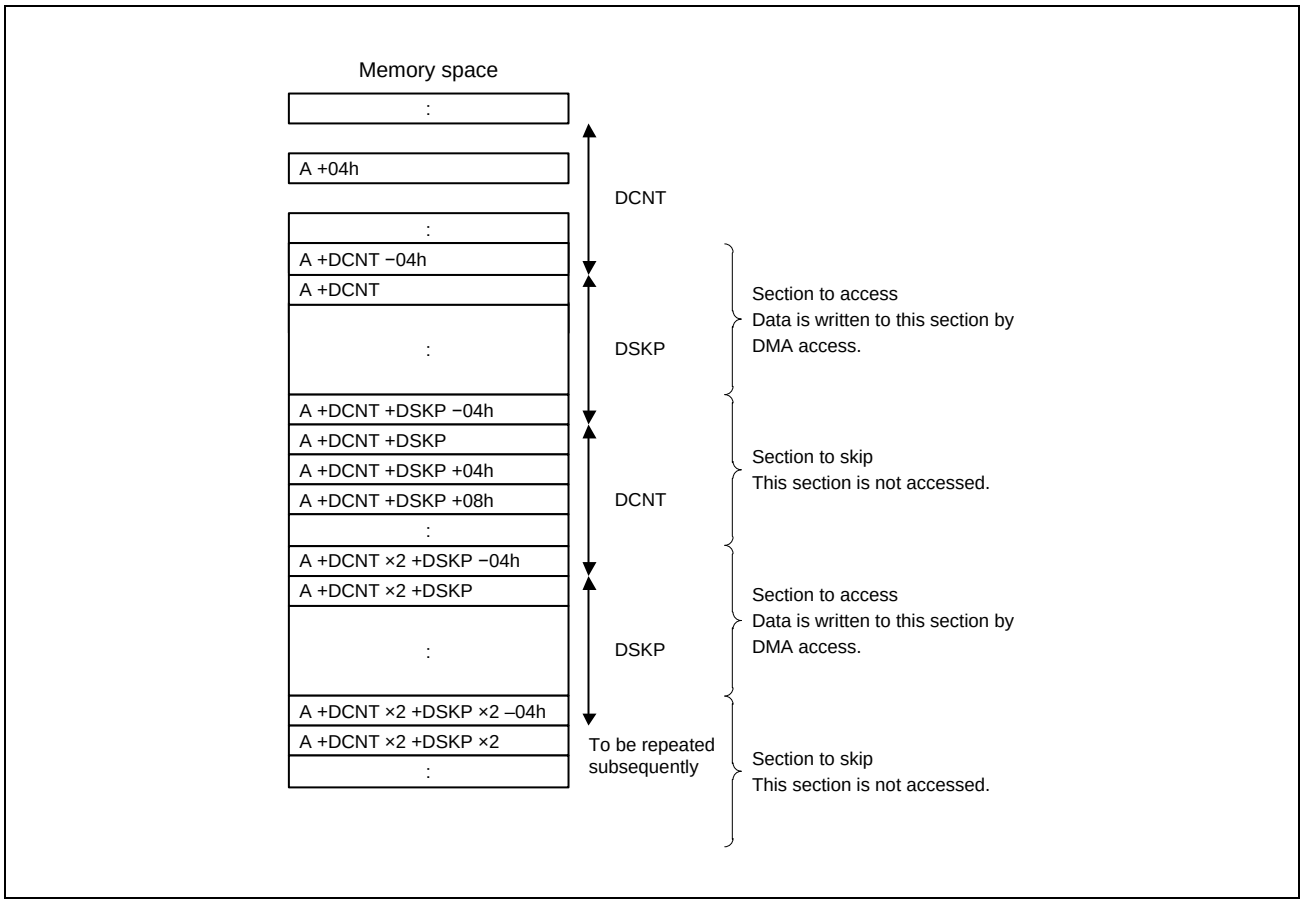


Figure 6.5-7 Relationship between DSKP and DCNT

You can specify values of the DCNT and DSKP bits regardless of the destination address and the value of the DDS field in the CHCFG_n register. The DMAC performs write access to only the specified space that has a combined size not more than the size specified in the DDS field (see **(a) Read access**).

6.5.3.2.29 DMA Register Set

The registers described below are shared by all channels.

(1) DMA Control Register

(a) DMA Control Register (USB20_FUNC_DCTRL)

Access Size : 32 bits

Address : <USB20_func_base> + 0700h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	LWPR[3:0]				-	-	-	-	LDPR[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LVINT	PR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
27 to 24	LWPR[3:0]	0h	RW	Link Writeback PROT These bits specify the value to be output to the MHPROT[3:0] pin at write-back to the descriptor in link mode.
23 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
19 to 16	LDPR[3:0]	0h	RW	Link Descriptor PROT These bits specify the value to be output to the MHPROT[3:0] pin at reading of the descriptor in link mode.
15 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
1	LVINT	0h	RW	Level Interrupt To use this module, be sure to set this bit to 1b.
0	PR	0h	RW	Priority This bit specifies the transfer priority control mode (see **6.5.3.9.13(2) DMA channel priority control**). 0b: Fixed priority mode 1b: Round-robin mode

(2) Descriptor Interval Register

(a) Descriptor Interval Register (USB20_FUNC_DSCITVL)

Access Size : 32 bits

Address : <USB20_func_base> + 0704h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DITVL[7:0]								-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
15 to 8	DITVL[7:0]	0h	RW	Descriptor Interval These bits specify the interval of descriptor read operation. The descriptor will be reread at intervals of “value of DITVL x 256.”
7 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(3) DMA Control Register

(a) DMA Status EN Register (USB20_FUNC_DSTAT_EN)

Access Size : 32 bits

Address : <USB20_func_base> + 0710h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EN1	EN0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
1	EN1	0h	R	This bit indicates the state of the EN bit for DMA channel 1.
0	EN0	0h	R	This bit indicates the state of the EN bit for DMA channel 0.

(4) DMA Status ER Register

(a) DMA Status ER Register (USB20_FUNC_DSTAT_ER)

Access Size : 32 bits

Address : <USB20_func_base> + 0714h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ER1	ER0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
1	ER1	0h	R	This bit indicates the state of the ER bit for DMA channel 1.
0	ER0	0h	R	This bit indicates the state of the ER bit for DMA channel 0.

(5) DMA Status END Register

(a) DMA Status END Register (USB20_FUNC_DSTAT_END)

Access Size : 32 bits

Address : <USB20_func_base> + 0718h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	END1	END0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
1	END1	0h	R	This bit indicates the state of the END bit for DMA channel 1.
0	END0	0h	R	This bit indicates the state of the END bit for DMA channel 0.

(6) DMA Status TC Register

(a) DMA Status TC Register (USB20_FUNC_DSTAT_TC)

Access Size : 32 bits

Address : <USB20_func_base> + 071Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TC1	TC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
1	TC1	0h	R	This bit indicates the state of the TC bit for DMA channel 1.
0	TC0	0h	R	This bit indicates the state of the TC bit for DMA channel 0.

(7) DMA Status SUS Register

(a) DMA Status SUS Register (USB20_FUNC_DSTAT_SUS)

Access Size : 32 bits

Address : <USB20_func_base> + 0720h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SUS1	SUS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
1	SUS1	0h	R	This bit indicates the state of the SUS bit for DMA channel 1.
0	SUS0	0h	R	This bit indicates the state of the SUS bit for DMA channel 0.

6.5.3.3 Functions

6.5.3.3.1 System Control and Oscillation Control

This section describes register manipulations required to perform initial setup of this module. This section also describes the registers required to control power consumption.

For the parts of the sequence that are required in both host and peripheral modes, see **6.5.2.9.1 Host/Peripheral Common Setting Sequence**.

(1) USB data bus resistor control

This module controls switchover between the pull-up resistors for the D+ signal and the pull-down resistors for the D- signal of the USB2PHY. Use the DPRPU bit and DRPD bit in the SYSCFG0 register to set pull-up or pull-down of each signal.

Recognize that a connection to the USB host is established, and then set 1 for the DPRPU bit in the SYSCFG0 register to pull up the D+ signal.

After connected to the host, this module automatically switches the resistor when the state changes to reset handshake, suspend, or resume.

If 0 is set for the DPRPU bit in the SYSCFG0 register during communication with the host, the pull-up resistor (or termination resistor) for the USB data line is disabled. This can notify the host controller of disconnection of a device.

6.5.3.3.2 Interrupt Function

(1) Overview of Interrupt Function (other than DMA Master)

The following shows a list of interrupt functions of this module. An interrupt is notified as U2P_IXL_INT. Check the status register to identify the interrupt factor. The interrupt from this module is not asserted when the supply of the internal bus clock (P1ϕ) is stopped.

Table 6.5-30 List of Interrupt functions

Bit	Interrupt name	Interrupt factor	Related status
VBINT	VBUS interrupt	The status change of the VBUS input pin is detected. (Changes from L to H and from H to L are detected.)	VBSTS
RESM	Resume interrupt	In the suspended state, a change of the USB bus status is detected (from J-State to K-State or from J-State to SE0).	—
SOFR	Frame number update interrupt	If SOFRM is 0: An SOF packet with a different frame number is received. If SOFRM is 1: An SOF with µframe number 0 cannot be received due to a problem such as packet corruption.	—
DVST	Device state transition interrupt	A transition of a device state is detected. USB bus reset detected Suspended state detected Set Address request received Set Configuration request received	DVSQ
CTRT	Control transfer stage transition interrupt	A transition of a control transfer stage is detected. Setup stage completed Control Write transfer status stage transition Control Read transfer status stage transition Control transfer completed Control transfer sequence error	CTSQ
BEMP	Buffer empty interrupt	All data in the buffer memory is sent and the buffer becomes empty. A packet exceeding the maximum packet size is received.	PIPEBEMP
NRDY	Buffer not ready interrupt	A token is received when the PID setting is BUF and the buffer memory is not available for sending and receiving data. A CRC error or bit stuff error occurs when data is received in isochronous transfer. An interval error occurs when data is received in isochronous transfer.	PIPENRDY
BRDY	Buffer ready interrupt	The buffer becomes ready (available for reading or writing data).	PIPEBRDY

The following shows the relationship between interrupts of this module.

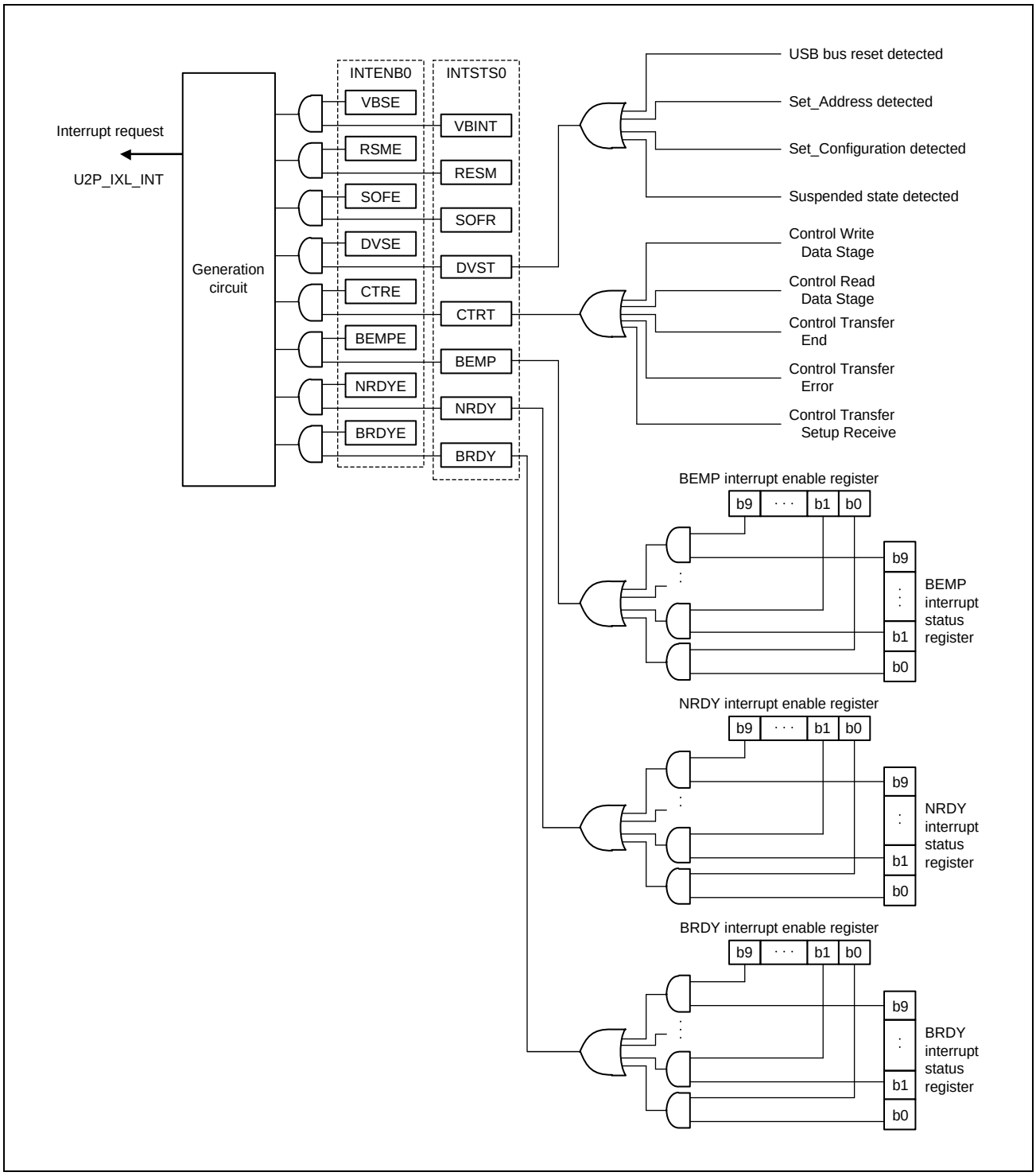


Figure 6.5-8 Interrupt Association Diagram

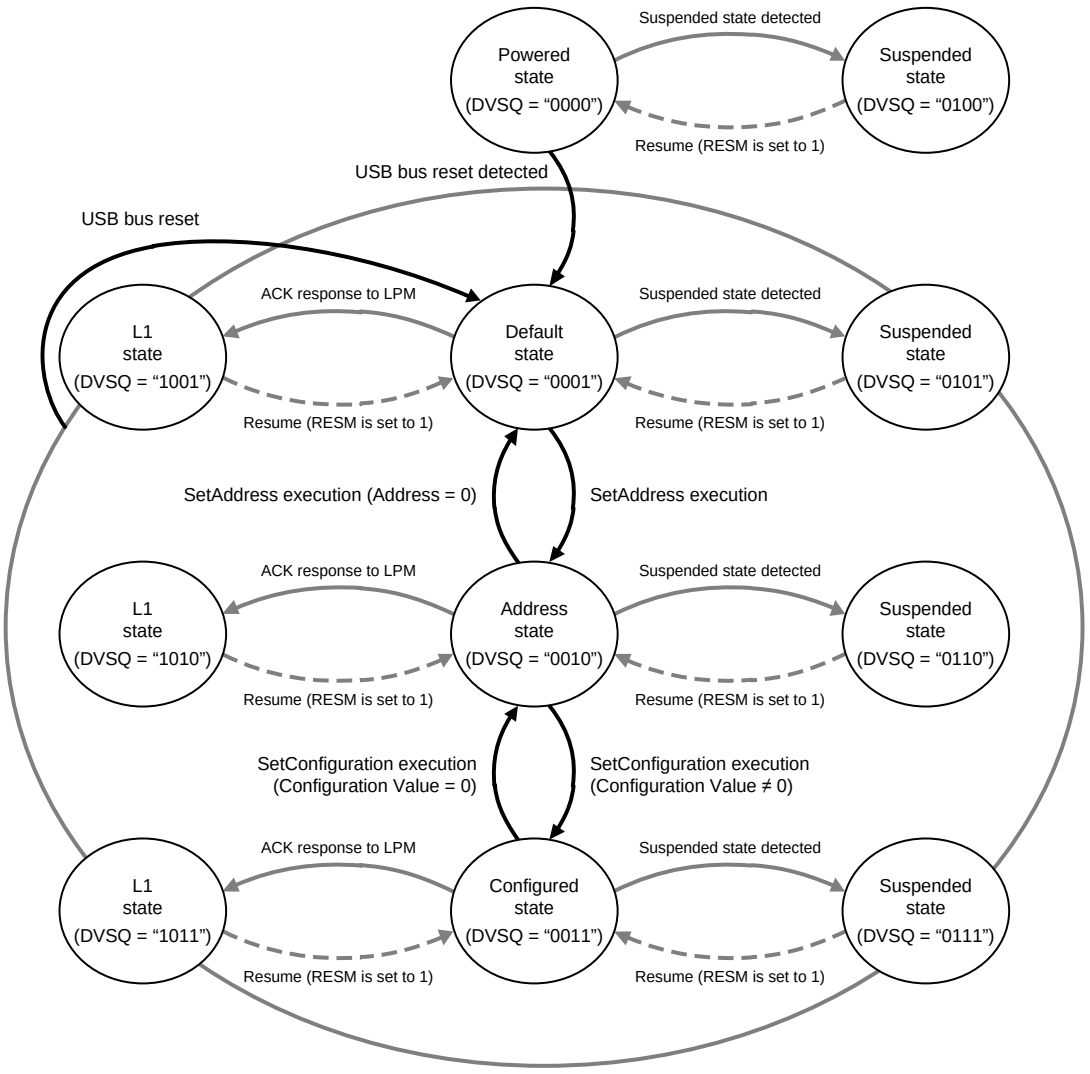
(2) Device State Transition Interrupts

Figure 6.5-9 shows the device state transition diagram of this module.

This module manages device states and generates device state transition interrupts. However, resumption from the suspended state (resume signal detection) is detected by a resume interrupt.

Device state transition interrupts can be enabled and prohibited by using the INTENB0 register. A device state for which a transition has occurred can be checked in the DVSQ bits in the INTSTS0 register.

To trigger a transition to the default state, a device state transition interrupt is generated after a reset hand-shake protocol ends.



Note: When a transition indicated by a solid line occurs, the DVST bits are set to 1.
When a transition indicated by a broken line occurs, the RESM bit is set to 1.

Figure 6.5-9 Device State Transition Diagram

(3) Control Transfer Stage Transition Interrupts

Figure 6.5-10 shows the control transfer stage transition diagram of this module. This module manages a sequence of control transfers, and generates a control transfer stage transition interrupt. Control transfer stage transition interrupts can be enabled and prohibited by using the INTENB0 register. A transfer stage for which a transition has occurred can be checked in the CTSQ bits in the INTSTS0 register.

The following describes sequence errors that can occur during control transfer. If an error occurs, the PID bits in the DCPCTR register are set to 1X (STALL).

(a) For Control Read transfer

a) An OUT or PING token is received in a situation where data has not been transferred yet for an IN token of the data stage.

b) An IN token is received in the status stage.

c) A data packet "DATAPID = DATA0" is received in the status stage.

(b) For Control Write transfer

a) An IN token is received in a situation where an ACK response has not been sent yet for an OUT token of the data stage.

b) The first data packet ”DATAPID = DATA0” is received in the data stage.

c) An OUT or PING token is received in the status stage.

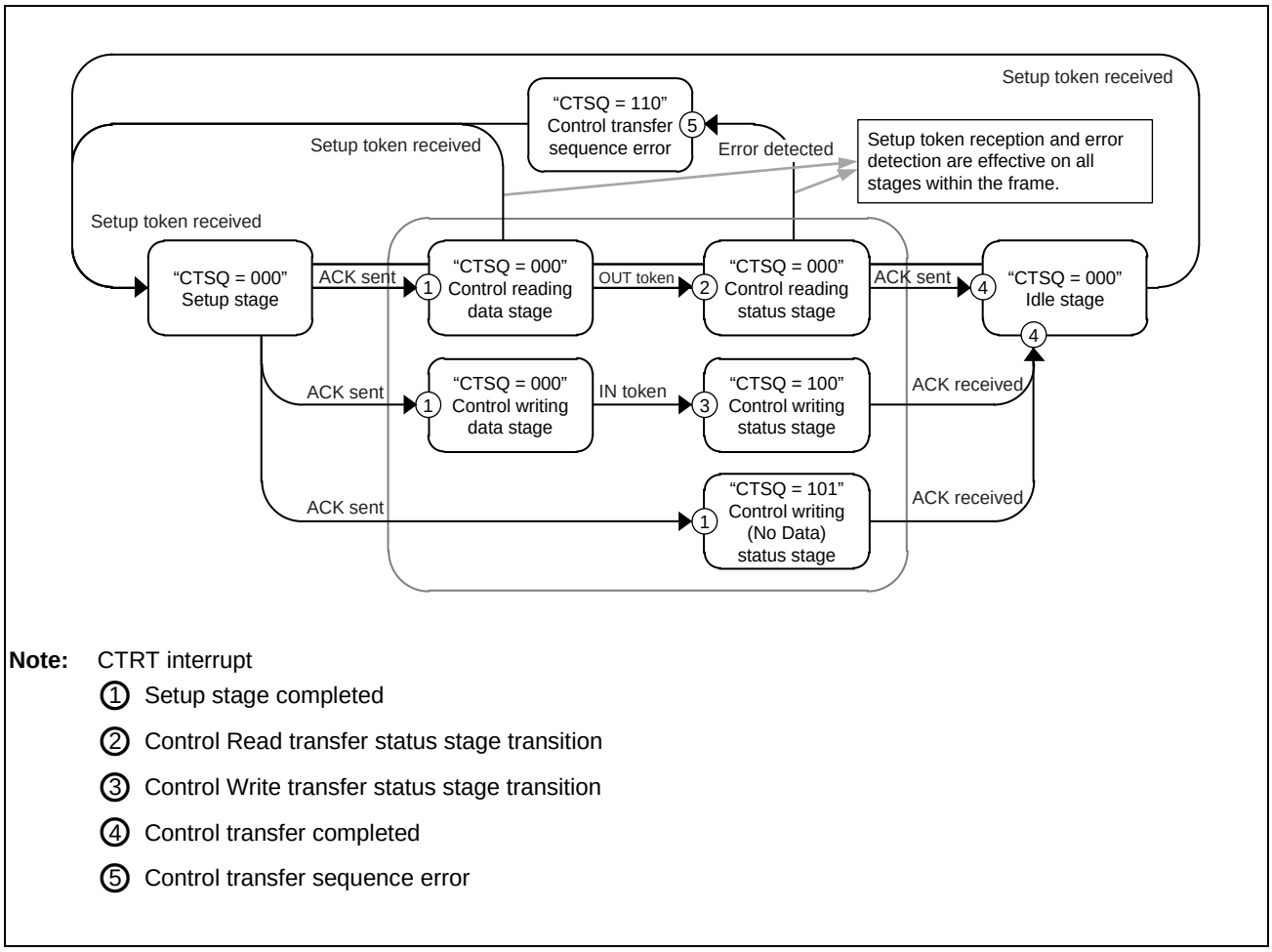
(c) For No-Data Control transfer

a) An OUT or PING token is received in the status stage.

If the amount of received data exceeds the value of the wLength field in a USB request in the Control Write transfer data stage, this module cannot identify this situation as a control transfer sequence error. If a packet other than a zero-length packet is received in the Control Read transfer status stage, this module sends an ACK response and then terminates processing normally.

If a CTRT interrupt is generated (SERR bit is set to 1) due to a sequence error, the CTSQ bits retain 110 until the system writes 0 to the CTRT bits to clear the interrupt status.

Therefore, as long as the CTSQ bits retain 110, even if a new USB request is received, a CTRT interrupt that reports completion of a setup stage is not generated. (Information on completion of the setup stage is retained by this module, and a CTRT interrupt is generated after the interrupt status is cleared.)



Note: CTRT interrupt

① Setup stage completed

② Control Read transfer status stage transition

③ Control Write transfer status stage transition

④ Control transfer completed

⑤ Control transfer sequence error

Figure 6.5-10 Control Transfer Stage Transition Diagram

(4) Interrupts Relating to DMA Master

Table 6.5-31 List of Interrupts

Interrupt Name	Interrupt Type
USBFDMAmn (m, n = 0, 1)	A DMA transaction ends.
	An invalid descriptor is read in link mode.
USBFDMAERRm (m = 0, 1)	An error response is returned for a transfer issued by the master interface.

This module has two types of DMA interrupts USBFDMAmn and USBFDMAERRm.

(1) USBFDMAmn interrupt (m, n = 0, 1)
This interrupt occurs when a DMA transaction ends or when an invalid descriptor is read in link mode.
An interrupt is divided for each DMA channel. USBFDMAm0 corresponds to the interrupt for DMA 0ch, and USBFDMAm1 corresponds to the interrupt for DMA 1ch.

(2) USBFDMAERRm interrupt
This interrupt occurs when an error response is returned for a transfer issued by the master interface. This interrupt applies to all DMA channels.

USBFDMAmn interrupt output can be temporarily masked by setting a register.

Interrupt detection can also be masked by setting the register. If interrupt detection is masked, the status register that indicates generation of an interrupt does not change.

On the other hand, a USBFDMAERRm interrupt signal does not have the masking function.

6.5.3.3.3 Pipe Control

Table 6.5-32 lists the pipe settings for this module. For USB data transfer, logical pipes called endpoints are used to enable data communication. This module provides 16 pipes for data transfer. Set each pipe according to the specification of the system.

Table 6.5-32 PIPE Settings

Register Name	Bit Name	Setting	Comment
DCPCFG	TYPE	Specifies the transfer type.	Pipes 1 to 9: Settable
PIPECFG	BFRE	Selects BRDY interrupt mode.	Pipes 1 to 5: Settable
	DBLB	Selects double-buffer configuration.	Pipes 1 to 5: Settable
	CNTMD	Selects continuous transfer or non-continuous transfer.	DCP: Settable Pipes 1, 2: Settable only in bulk transfers Pipes 3 to 5: Settable
	DIR	Selects transfer direction.	Settable to IN or OUT
	EPNUM	Endpoint number	Pipes 1 to 9: Settable Set to a value other than “0000” when a pipe is in use.
	SHTNAK	Disables pipes when transfer is completed.	DCP: Settable Pipes 1, 2: Settable only in bulk transfers Pipes 3 to 5 Settable
PIPEBUF	BUFSIZE	Buffer memory size	DCP: Unsettable (fixed to 64/256 (CNTMD = 1) bytes) Pipes 1 to 5: Settable (up to 2 Kbytes specifiable) Pipes 6 to 9: Unsettable (fixed to 64 bytes)
	BUFNMB	Buffer memory number	DCP: Unsettable (fixed to area 0 to 3 hex) Pipes 1 to 5 Settable (area 8 to 87 hex specifiable) Pipes 6 to 9: Unsettable (fixed to area 4 to 7 hex)
DCPMAXP PIPEMAXP	MXPS	Maximum packet size	Setting conforming to the USB specification.
PIPEPERI	IFIS	Buffer flush	Pipes 1, 2: Settable only in isochronous transfers Pipes 3 to 5: Unsettable Pipes 6 to 9: Unsettable
	IITV	Interval counter	Pipes 1, 2: Settable only in isochronous transfers Pipes 3 to 5: Unsettable Pipes 6 to 9: Unsettable
DCPCTR PIPExCTR	BSTS	Buffer status	DCP state switched between receive and transmit buffer by ISEL bit
	INBUFM	IN buffer monitor	Available only for pipes 3 to 5
	ACLRM	Auto buffer clear	Pipes 1 to 9: Settable
	SQCLR	Sequence clear	Clears data toggle bit.
	SQSET	Sequence set	Sets data toggle bit.
	SQMON	Sequence monitor	Monitors data toggle bit.
	PBUSY	Pipe busy monitor	
	PID	Response PID	
DCPCTR PIPExCTR	ATREPM	Auto response mode	Pipes 1 to 5: Settable
PIPExTRE	TRENB	Transaction count enable	Pipes 1 to 5: Settable
	TRCLR	Current transaction counter clear	Pipes 1 to 5: Settable
PIPExTRN	TRNCNT	Transaction counter	Pipes 1 to 5: Settable

(1) Maximum packet size setting

Use the MXPS bits in the DCPMAXP and PIPEMAXP registers to specify the maximum packet size for each pipe. The default control pipe (DCP) and pipes 1 to 5 can be set to any of the maximum packet sizes defined by the USB specification. For pipes 6 to 9, 64 bytes are the upper limit of the maximum packet size. Set the maximum packet size before starting transfer (by setting “PID = BUF”).

- DCP: Set 64 for high-speed operation.
- DCP: Set 8, 16, 32, or 64 for full-speed operation. Pipes 1 to 5: Set 512 for high-speed bulk transfer.
- Pipes 1 to 5: Set 8, 16, 32, or 64 for full-speed bulk transfer.
- Pipes 1, 2: Set a value from 1 to 1024 for high-speed isochronous transfer.
- Pipes 1, 2: Set a value from 1 to 1023 for full-speed isochronous transfer.
- Pipes 6 to 9: Set 64.

High-bandwidth transfers used for interrupt transfers and isochronous transfers are not supported.

(2) Response PID

Set the response PID for each pipe with the PID bits in the DCPCTR and PIPExCTR registers.

<Response PID setting>

The response PID specifies the response to a transaction from the host.

- NAK: Always sends a NAK response to a generated transaction.
- BUF: Responds to a transaction in accordance with the buffer memory state.
- STALL: Always sends a STALL response to a generated transaction.

Regardless of the value set in the PID bits, an ACK is always sent as a response to a setup transaction and a USB request is stored in corresponding registers.

This module might write data to the PID bits depending on the transaction result. This module writes data to the PID bits in the following cases:

- NAK:
  - The SETUP token is received normally (for the DCP only).
  - If 1 is set for the SHTNAK bit in the PIPECFG register during bulk transfer, a short packet is received.
  - If 1 is set for the SHTNAK bit during bulk transfer, the transaction counter finishes.
- BUF:
  - This module does not write "BUF".
- STALL:
  - When a maximum packet size over error is detected in a received data packet
  - When a control transfer sequence error is detected

(3) Pipe control register switching procedure

The following bits in the pipe control registers can be modified only when USB transmission is disabled (PID = NAK).

Figure 6.5-11 shows the procedure for changing the pipe control register state from the USB transmission enabled (PID = BUF) state.

The registered that cannot be manipulated in the USB transmission enabled (PID = BUF) state are as follows:

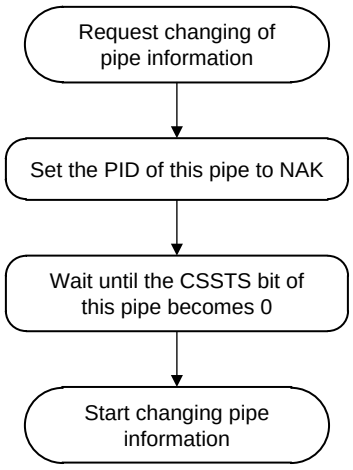
(1) All bits in the DCPCFG and DCPMAXP registers

(2) SQCLR and SQSET bits in the DCPCTR register

(3) All bits in the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers

(4) ATREPM, ACLRM, SQCLR, and SQSET bits in the PIPExCTR register

(5) All bits in the PIPExTRE and PIPExTRN registers



Note: If the pipe is detached during a USB transaction, the PBUSY bit might remain 1.

Figure 6.5-11 Procedure for Changing Pipe Information from USB Transmission Enabled (PID = BUF) State

In addition, for the settings of the following bits in pipe control registers, only the pipe information that is not set for the CURPIPE bit in the CPU, DMA0, or DMA1 FIFO port can be changed.

Registers that cannot be set with pipe information that is set for the CURPIPE bit in a FIFO port:

(1) All bits in the DCPCFG and DCPMAXP registers

(2) All bits in the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers

(3) ACLRM bit in the PIPExCTR register

If you change the pipe information, make sure that the setting of CURPIPE is different from the new pipe number. For the default control pipe (DCP), after modifying the pipe information, use the BCLR bit to clear the buffer.

(4) Data PID sequence bit

When a normal data transfer is performed during bulk transfer or interrupt transfer, or in the data stage of control transfer, this module automatically toggles the sequence bit of a data PID. The sequence bit of the next data PID to be transferred can be confirmed with the SQMON bit in the DCPCTR or PIPExCTR registers. For data transmission, the sequence bit is switched when an ACK handshake is received. For data reception, the sequence bit is switched when an ACK handshake is sent. The data PID sequence bit can be changed by using the SQCLR and SQSET bits in the DCPCTR and PIPExCTR registers.

During control transfer, this module automatically sets the sequence bit when the stage changes. When the setup stage finishes, the data PID is set to DATA1. In the status stage, this module responds with “PID = DATA1” without referencing the sequence bit.

Note that, the data PID sequence bit must set, when a clear feature request is sent or received.

Also note that for isochronous transfer setting pipes, you cannot use the SQSET bit to manipulate the sequence bit.

6.5.3.3.4 FIFO Buffer

This section describes the processing related to the FIFO buffer of this module.

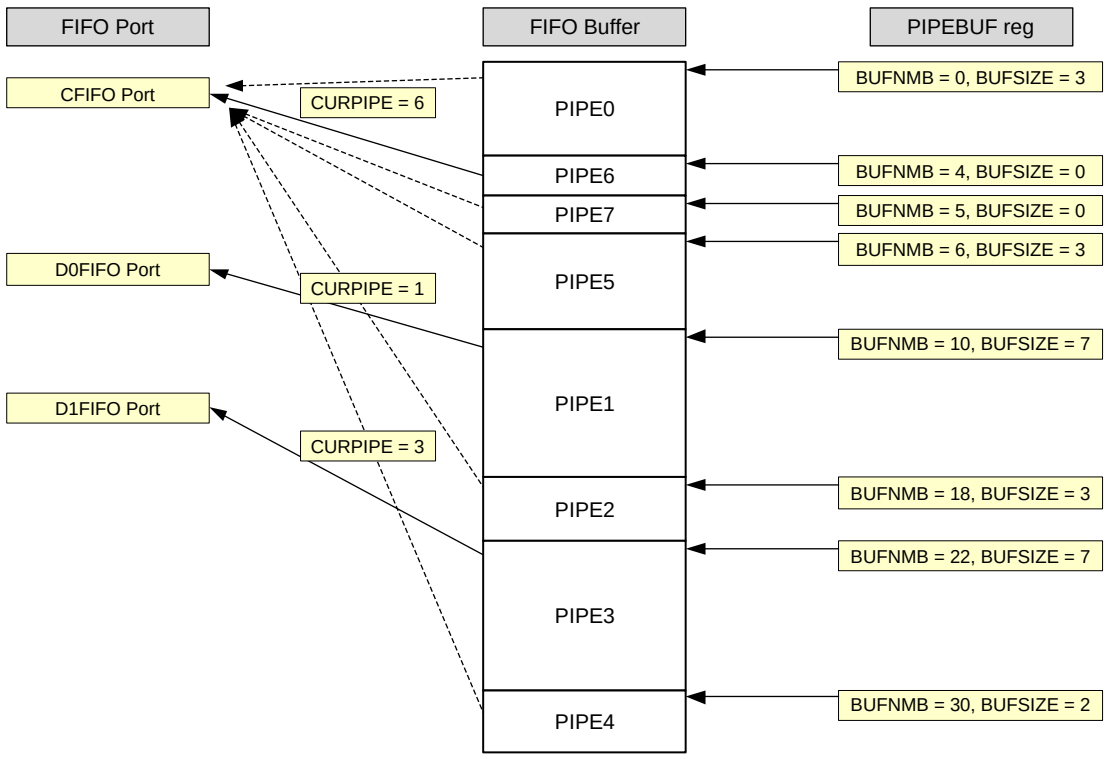
(1) FIFO buffer allocation

Figure 6.5-12 shows an example of FIFO buffer memory mapping of this module. The FIFO buffer area is shared by the CPU and this module. The FIFO buffers can be accessed by either the system (CPU) or this module (SIE).

An independent FIFO buffer area is allocated for each pipe. The memory area is made up of memory blocks (1 block = 64 bytes) and is specified by the starting block number and the number of blocks (specified by the BUFNMB and BUFSIZE bits in the PIPEBUF register). When the CNTMD bit in the PIPExCFG register is used to set "continuous transfer mode", the value set with the BUFSIZE bit must be an integral multiple of the maximum packet size. If the double-buffer configuration is selected by the DBLB bit in the PIPExCFG register, two memory areas the size of which is specified by the BUFSIZE bit in the PIPEBUF register are allocated to a single pipe.

FIFO ports are used to access the FIFO buffer (data read/write). The pipe number of a pipe to be assigned to a FIFO port is specified by the CURPIPE bits in the CFIFOSEL/DxFIFOSEL register.

The FIFO buffer status of each pipe can be confirmed by using the BSTS and INBUFM bits in the DCPCTR and PIPExCTR registers. The access right of a FIFO port can be confirmed by using the FRDY bit in the CFIFOCTR/ DxFIFOCTR register.



Note: If pipes 8 and 9 are not in use: Settings such as BUFSIZE are unspecified.

Figure 6.5-12 Example of FIFO Buffer Memory Mapping

(2) Clearing FIFO buffers

Table 6.5-33 lists the modes in which this module can clear the FIFO buffer. Clearing of the FIFO buffer is controlled by the following bits.

Table 6.5-33 List of FIFO Buffer Clearing Modes

Bit Name	BCLR	DCLRM	ACLRM
Register	CFIFOCTR register DxFIFOCTR register	DxFIFOSEL register	PIPExCTR register
Function	The FIFO buffer on the CPU side is cleared.	The FIFO buffer is automatically cleared after the data is read from the specified pipe.	The buffer is automatically cleared to discard all the received packets.
Clearing method	Write “1” to clear.	1: Mode enabled 0: Mode disabled	1: Mode enabled 0: Mode disabled

6.5.3.3.5 FIFO Port Functions

This section describes FIFO port functions. **Table 6.5-34** shows the FIFO port function settings for this module.

If data write access is enabled and data is written up to buffer full state (in non-continuous transfer: maximum packet size), the port automatically goes to the USB bus transmittable state. To enable transmission of data smaller than the buffer full (in non-continuous transfer: less than the maximum packet size), the BVAL bit in the CFIFOCTR/ DxFIFOCTR register must be used to set write end.

To send a zero-length packet, the BCLR bit of that register must be used to clear the buffer before the BVAL bit is used to write end.

When all the data is read in a read access, the port automatically enters the state in which new packets can be received. However, when a zero-length packet is received (DTLN = 0), no data can be read, and therefore the buffer must be cleared by using the BCLR bit.

The receive data length is confirmed with the DTLN bits in the CFIFOCTR/DxFIFOCTR register.

Table 6.5-34 FIFO Port Function Settings

Register Name	Bit Name	Function	Note
C/DxFIFOSEL	RCNT	Selects DTLN read mode	
	REW	Rewinds buffer memory (re-read, re-write)	
	DCLRM	Reads received data of the specified pipe, and then automatically clears the received data	DxFIFO only
	DREQE	Asserts DREQ signal	DxFIFO only
	MBW	Specifies FIFO port access bit width	
	BIGEND	Selects FIFO port endian	CFIFO only
	ISEL	Specifies FIFO port access direction	DCP only
	CURPIPE	Selects current pipe	
C/DxFIFOCTR	BVAL	Finishes buffer memory write	
	BCLR	Clears CPU-side buffer memory	
	FRDY	Monitors FIFO port ready	
	DTLN	Confirms received data length	

(1) FIFO port selection

Table 6.5-35 lists pipes that can be selected for each FIFO port.

Use the CURPIPE bits in the C/DxFIFOSEL register to select the pipe to be accessed. After selecting the pipe, confirm that the value written to the CURPIPE bits can be read correctly (if the previous pipe number is read out, this module is currently changing the pipe), confirm that FRDY = 1, and then access the FIFO port.

Figure 6.5-13 shows the procedure for switching the pipe when accessing a FIFO port.

In addition, use the MBW bit to select the bus width with which to access the FIFO port. If the target pipe is the default control pile (DCP), the ISEL bit determines the buffer memory access direction. If the target pipe is not the DCP, the DIR bit in the PIPExCFG register determines the buffer memory access direction.

Table 6.5-35 FIFO Port Access for Each Pipe

Pipe	Access Method	Usable Port
DCP	CPU access	CFIFO port register
Pipes 1 to 9	CPU access	CFIFO port register
	DMA access	DxFIFO port register

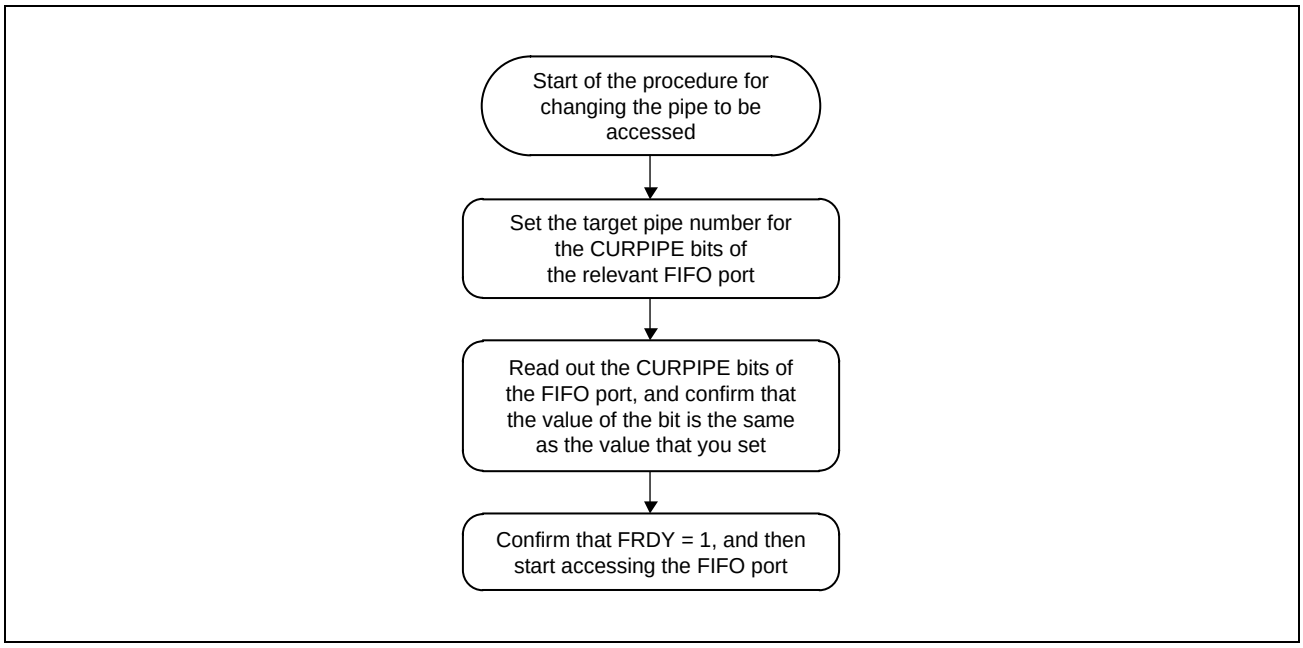


Figure 6.5-13 Pipe Switching Procedure for FIFO Port Access

(2) DxFIFO automatic clear mode (DxFIFO port read direction)

This module automatically clears the buffer memory for a pipe when data is read out from the buffer memory if the DCLRM bit in the DxFIFOSEL register is set to 1.

Table 6.5-36 shows the correspondence between packet reception and buffer memory clear processing by the software for each setting.

As shown in **Table 6.5-36**, the buffer clearing conditions vary with the value that is set for the BFRE bit. However, using the DCLRM bit eliminates the need for buffer clear by the software even in states where clearing is required, which enables DMA transfers without using the software.

Note that for this function, only the buffer memory read direction can be set.

Table 6.5-36 Relationship between Packet Reception and Buffer Memory Clear Processing by the Software

	Register Setting			
	DCLRM = 0		DCLRM = 1	
Buffer State during Packet Reception	BFRE = 0	BFRE = 1	BFRE = 0	BFRE = 1
Buffer full	Clearing unnecessary		Clearing unnecessary	
Zero-length packet received	Clearing necessary			
Normal short packet received	Clearing unnecessary	Clearing necessary		
Transaction count end				

(3) BRDY interrupt timing selection function

The BFRE bit in the PIPECFG register can be set so that a BRDY interrupt is not generated when a data packet of maximum packet size is received.

When using a DMA transfer, this function enables an interrupt to be generated only when the last data is received. The “last data” here indicates either a short packet reception or the transaction count end. If the BFRE bit is set to 1, a BRDY interrupt is generated after the received data is read. By reading the DTLN bits in the DxFIFOCTR register, the receive data length of last data packet received just before the BRDY interrupt is generated can be confirmed.

Table 6.5-37 shows when this module generates a BRDY interrupt.

Table 6.5-37 BRDY Interrupt Generation Timing

	Register Setting	
Buffer State during Packet Reception	BFRE = 0	BFRE = 1
Buffer full (normal packet received)	When packet is received	No interrupt generated
Zero-length packet received	When packet is received	When packet is received
Normal short packet received	When packet is received	When received data has been read from the buffer memory
Transaction count end	When packet is received	When received data has been read from the buffer memory

The BFRE bit function is valid only in the reading direction from the buffer memory. For the writing direction, fix the BFRE bit to 0.

6.5.3.3.6 Control Transfer (DCP)

In the data stage of control transfer, the default control pipe (DCP) is used to transfer data. For the DCP, a single 64-byte buffer is allocated as a fixed area that is used for both control reading and writing (in continuous transfer mode (CNTMD = 1), the size of this area is fixed to 256 bytes). The buffer memory can be accessed through the CFIFO port only.

(1) Control transfer

(a) Setup stage

This module always responds with ACK to any normal setup packet for this module. The following shows the behavior of this module in the setup stage:

1. When this module receives a new setup packet, this module sets the following bits:
   - VALID bit in the INTSTS0 register: 1
   - PID bit in the DCPCTR register: NAK
   - CCPL bit in the DCPCTR register: 0
2. When this module receives a data packet after receiving a setup packet, this module stores USB request parameters in the USBREQ, USBVAL, USBINDX, and USBLENG registers.

Response processing for control transfer must be performed after VALID is set to 0. While VALID is 1, PID cannot be set to BUF, and therefore the data stage cannot end.

By using a function of the VALID bit, when this module receives a new USB request during control transfer, this module can respond to the newest request, canceling the request that is being processed.

This module also automatically recognizes the type of transfer (Control Read, Control Write, or No-Data Control) from the direction bit (bit 8 of bmRequestType) and request data length (wLength) of the received USB request to manage stage transition. For an incorrect sequence, a control transfer stage transition interrupt occurs to report a sequence error to software. For details about stage management by this module, see **Figure 6.5-10**.

(b) Data stage

Use the DCP to transfer data in response to the received USB request. Before the DCP buffer memory is accessed, use the ISEL bit in the CFIFOSEL register to specify the access direction.

A transaction is executed by setting the PID bit in the DCPCTR register to BUF.

The end of data transfer is detected with a BRDY or BEMP interrupt. For Control Write transfer, use a BRDY interrupt. For Control Read transfer, use a BEMP interrupt.

For Control Write transfer in high-speed operation mode, an NYET handshake response is performed in accordance with the buffer memory state.

(c) Status stage

If the PID bit in the DCPCTR register is BUF, setting the CCPL bit to 1 terminates control transfer.

After control transfer is terminated by the above setting, this module automatically executes the status stage according to the data transfer direction determined in the setup stage, as shown below:

(1) For Control Read transfer
Upon receiving a zero-length packet from the USB Host Controller, this module sends an ACK response.

(2) For Control Write or No-Data Control transfer
This module sends a zero-length packet, and then receives an ACK response from the USB Host Controller.

(d) Control transfer automatic response function

This module automatically responds to any normal SET_ADDRESS request. However, if a SET_ADDRESS request has any of the following errors, software must respond, instead of this module:

- bmRequestType ≠ “00h”
- wIndex ≠ “00h”
- wLength ≠ “00h”
- wValue > “7Fh”
- DVSQ = “011 (Configured)”

Software must respond to all requests other than SET_ADDRESS.

6.5.3.3.7 Bulk Transfer (Pipes 1 to 5)

The user can select the buffer memory usage method (single/double buffer, continuous/non-continuous transfer mode) for bulk transfers. The buffer memory size can be set up to 2 KB. The controller manages the buffer memory state and automatically responds to PING packets and NYET handshakes.

(1) NYET handshake control

Table 6.5-38 lists responses to tokens received in a bulk or control transfer.

When an OUT token is received in a bulk or control transfer and there is only an open space for one packet in the buffer memory, this module sends a NYET response. However, when a short packet is received, this module sends an ACK response instead of a NYET response even under these conditions.

Table 6.5-38 List of Responses to Received Tokens

PID bit value	Buffer memory state*1	Received token	Response	Note
NAK/STALL	—	SETUP	ACK	—
	—	IN/OUT/PING	NAK/STALL	—
BUF	—	SETUP	ACK	—
	RCV-BRDY	OUT/PING	ACK	Receives data packet when OUT token is received*1
	RCV-BRDY	OUT	NYET	Receives data packet*2
	RCV-BRDY	OUT (Short)	ACK	Receives data packet*2
	RCV-BRDY	PING	ACK	*2
	RCV-NRDY	OUT/PING	NAK	
	TRN-BRDY	IN	DATA0/1	Sends data packet
	TRN-NRDY	IN	NAK	

Note: Details are described below.
RCV-BRDY*1: Buffer memory has a space for 2 packets or more when an OUT or PING token is received.
RCV-BRDY*2: Buffer memory has only a space for one packet when an OUT token is received.
RCV-NRDY: Buffer memory has no space for any packet when a PING token is received.
TRN-BRDY: Buffer memory has sent data when an IN token is received.
TRN-NRDY: Buffer memory has no send data when an IN token is received.

6.5.3.3.8 Interrupt Transfer (Pipes 6 to 9)

This module performs an interrupt transfer in accordance with the period managed by the host controller. This module ignores (no response) PING packets in interrupt transfers. In addition, this module does not send a NYET handshake, but sends an ACK, NAK or STALL response.

Note that this module does not support high-bandwidth interrupt transfers.

6.5.3.3.9 Isochronous Transfer (Pipes 1 and 2)

This module is provided with the following functions for isochronous transfers:

- Notification of error information about isochronous transfers
- Interval counter (IITV bit)
- Data setup control for isochronous IN transfers (IDLY function)
- Buffer flush function for isochronous IN transfers (IFIS bit)
- SOF pulse output function

This module does not support high-bandwidth Isochronous transfers.

(1) Isochronous transfer error detection

This module has the following error information detection functions for the software to manage errors that occur during isochronous transfer.

Table 6.5-39 and **Table 6.5-40** describe the error checking procedure and interrupts that are generated.

1. PID error
   The PID of the received packet is invalid.
2. CRC error and bit stuffing error
   The received packet has a CRC error or invalid bit stuffing.
3. Max packet size over
   The data size of the received packet is larger than the preset maximum packet size.
4. Overrun error and underrun error
   (a) The buffer memory has no data when an IN token is received during IN-direction (send) transfer.
   (b) The buffer memory has no space although an OUT token is received during OUT-direction (receive) transfer.
5. Interval error
   An interval error occurs in the following cases:
   (a) An IN token cannot be received within the interval frame during isochronous IN transfer.
   (b) An OUT token cannot be received within the interval frame during isochronous OUT transfer.

Table 6.5-39 Error Detection during Transmission/Reception of Token

Error Detection Priority	Error Type	Interrupt Generated at Error Detection and Status
1	PID error	No interrupt is generated (ignored as a corrupted packet).
2	CRC error, bit stuffing error	No interrupt is generated (ignored as a corrupted packet).
3	Overrun error, underrun error	An NRDY interrupt is generated and the OVRN bit is set. A zero-length packet is sent in response to an IN token. A data packet is not received in response to an OUT token.
4	Interval error	An NRDY interrupt is not generated.

Table 6.5-40 Error Detection during Reception of Data Packet

Error Detection Priority	Error Type	Interrupt Generated at Error Detection and Status
1	PID error	No interrupt is generated (ignored as a corrupted packet).
2	CRC error, bit stuffing error	An NRDY interrupt is generated and the CRCE bit is set.
3	Packet size error (too large packet)	A BEMP interrupt is generated and the PID bit is set to STALL.

(2) DATA-PID

This module does not support high-bandwidth transfers.

The following shows actions that can be taken in response to a received PID.

1. IN direction:

   DATA0: Used to send packets.
   DATA1: Not used to send packets.
   DATA2: Not used to send packets.
   mDATA: Not used to send packets.

2. OUT direction (in full-speed operation):

   DATA0: Packets are received normally.
   DATA1: Packets are received normally.
   DATA2: Packets are ignored.
   mDATA: Packets are ignored.

3. OUT direction (in high-speed operation):

   DATA0: Packets are received normally.
   DATA1: Packets are received normally.
   DATA2: Packets are received normally.
   mDATA: Packets are received normally.

(3) Interval counter

(a) Outline of operation

The IITV bit in the PIPEPERI register can be used to set the interval of isochronous transfer. The interval counter enables the functions listed in **Table 6.5-41**.

Table 6.5-41 Functions of the Interval Counter

Transfer Direction	Function	Detecting Condition
IN	Transmit buffer flush function	An IN token cannot successfully be received within the interval frame during isochronous IN transfer.
OUT	Notification of unreceived token	An OUT token cannot successfully be received within the interval frame during isochronous OUT transfer.

Counting of intervals is based on received SOF packets or interpolated SOFs. Therefore, even if SOF packets are damaged, the isochronism can still be maintained. Frame intervals are set as 2n (micro) frames, where n is the value of the IITV bit.

(b) Interval counter initialization

This module initializes the interval counter under the following conditions:

(1) Power on reset
The IITV bit is initialized.

(2) Clearing of the buffer memory by the ACLRM bit
The IITV bit is not initialized but the counter is initialized.

(3) USB bus reset

After the interval counter is initialized and a packet is successfully transferred, counting of intervals starts under the following conditions:

(1) An SOF packet is received after data is sent in response to an IN token when PID = BUF.

(2) An SOF packet is received after data is received in response to an OUT token when PID = BUF.

Note that the interval counter is not initialized in the following conditions:

(1) The PID bit is set to NAK or STALL.
The interval timer is not stopped at this interval. The transaction will be attempted at the next interval.

(2) USB bus reset or USB suspension
The IITV bit is not initialized. When an SOF packet is received, counting starts from the value existing before reception.

(4) Send data setup for isochronous transfer

This module becomes able to send data packets by isochronous transfer from the next frame after data is written to the buffer memory and then an SOF packet is detected. This is called “send data setup for isochronous transfer”.

This function can identify the frame with which data sending started.

If the buffer memory is in a double-buffer configuration and writing to both buffers has been completed, only the buffer to which writing finished earlier can transfer data. Therefore, even when several IN tokens are received within the same frame, only one packet of data is sent from the buffer memory.

When an IN token is received, if the buffer memory is ready for sending data, the data is transferred and a normal response is returned. However, if the buffer memory is not ready for sending data, a zero-length packet is sent and an underrun error occurs.

Figure 6.5-14 shows examples of sending using the send data setup function for isochronous transfer by setting “IITV = 0” (each frame) in this module.

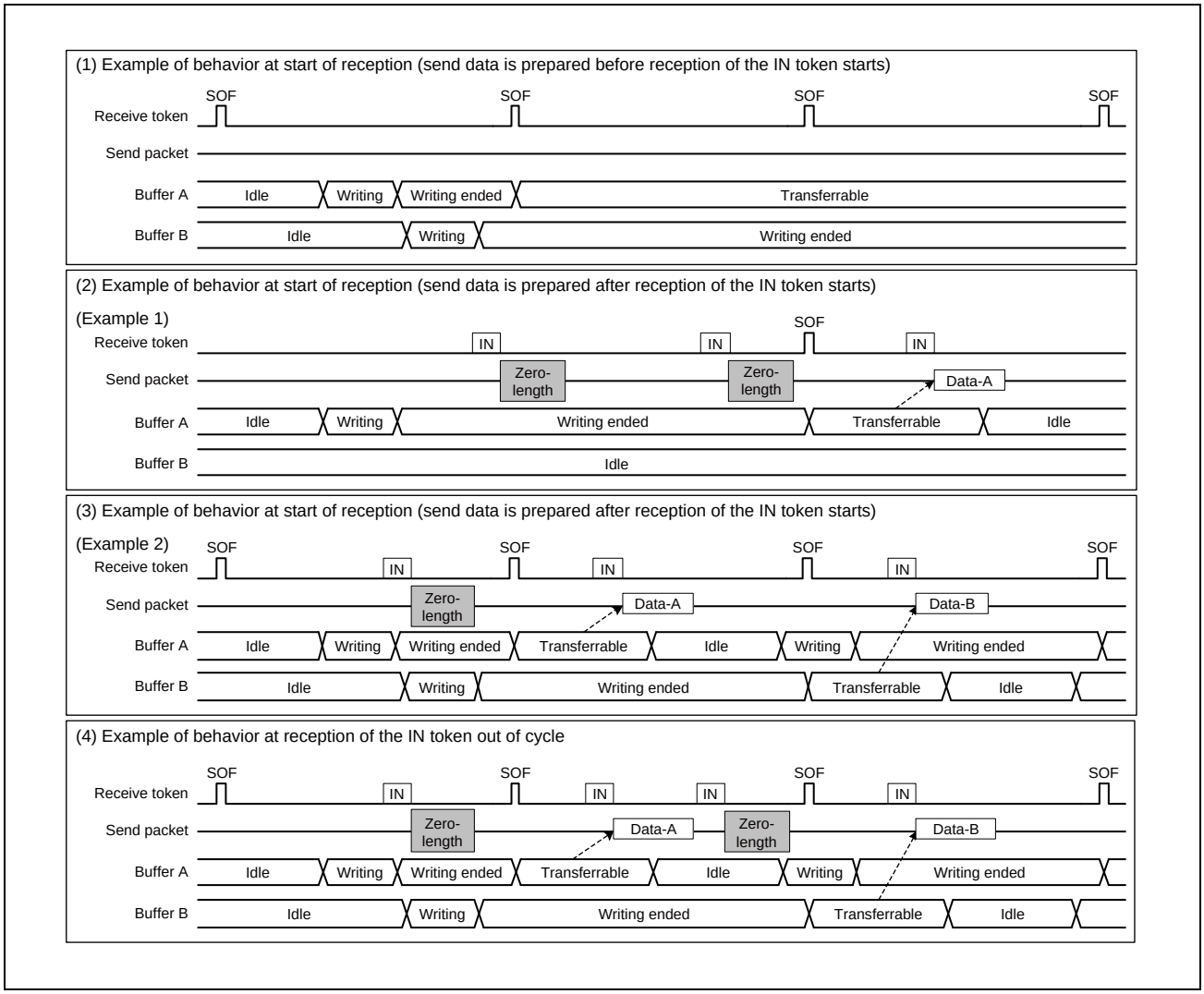


Figure 6.5-14 Examples of Data Setup Function Behavior

(5) Transmit buffer flush for isochronous transfer

If this module does not receive an IN token in an interval frame and receives a (micro) SOF packet in the next frame during isochronous data transfer, this module handles the IN token as a corrupted token and clears the buffer that can send data to make the buffer writable.

At this time, if the buffer memory is in a double-buffer configuration and writing to both buffers has been completed, this module assumes the discarded buffer memory to be sent within the same interval frame. As a result, the buffer memory that is not discarded by reception of a (micro) SOF packet becomes to be able to transfer data.

The operation start timing of the buffer flush function varies with the value of the IITV bit.

- If IITV is 0
  Buffer flush operation is performed from the first frame after the pipe is enabled.
- If IITV is not 0
  Buffer flush operation is performed after the first successful transaction.

Figure 6.5-15 shows an example of how the buffer flush function of this module behaves. For a token outside the set interval (token prior to the interval frame), however, this module sends the written data or a zero-length packet as an underrun error according to the data setup state.

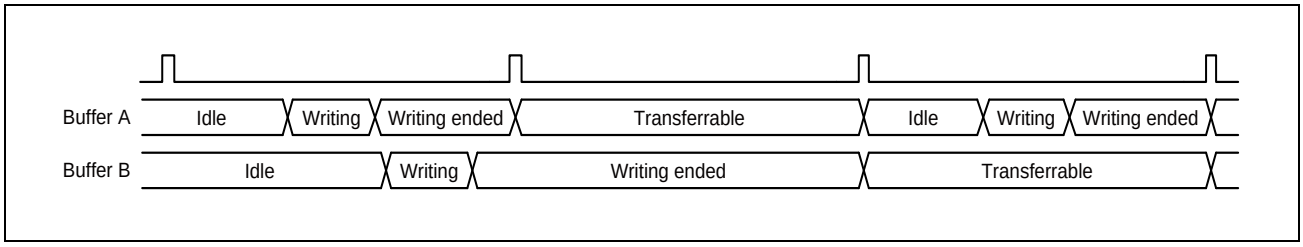


Figure 6.5-15 Example of Buffer Flush Function Behavior

Figure 6.5-16 shows an example of an interval error that occurs in this module.

There are five types of interval errors, as listed below. At timing (1) in the figure, an interval error occurs and the buffer flush function operates.

If an interval error occurs during IN transfer, the buffer flush function starts. If an interval error occurs during OUT transfer, an NRDY interrupt occurs.

Use the OVRN bit to determine whether an error is an NRDY interrupt (such as a receive packet error) or an overrun error.

In the figure, responses to tokens indicated as shaded boxes are made in accordance with the buffer memory state.

- IN direction:
  - If the buffer is ready to transfer data, data is transferred and a normal response is returned.
  - If the buffer is not ready to transfer data, a zero-length packet is sent and an underrun error occurs.
- OUT direction:
  - If the buffer is ready to receive data, data is received and a normal response is returned.
  - If the buffer is not ready to receive data, data is discarded and an overrun error occurs.

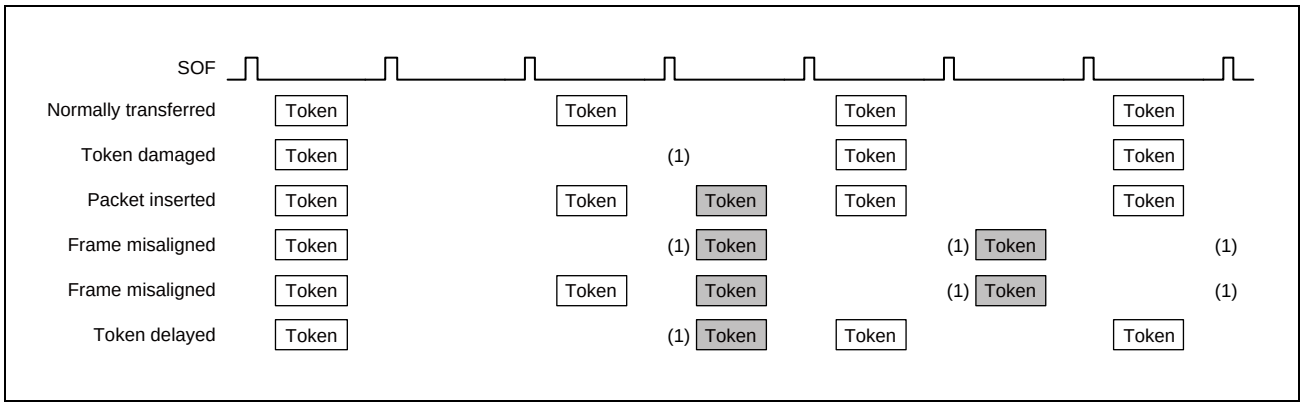


Figure 6.5-16 Example of Occurrence of Interval Error (when IITV is 1)

6.5.3.3.10 SOF Interpolation Function

If the controller cannot receive data at intervals of 1 ms (in full-speed operation) or 125 µs (in high-speed operation) due to corruption or missing of an SOF packet, the controller internally interpolates the SOF. The controller starts SOF interpolation upon receiving an SOF packet when both the USBE bit and SUSPM bit are set to 1.

The interpolation function is initialized under the following conditions:

1) Power on reset
2) USB bus reset
3) Suspended state detected

The SOF interpolation operates according to the following specifications:

1. Frame interval (125 µs or 1 ms) is based on the results of the reset handshake protocol.
2. The interpolation function does not operate until an SOF packet is received.
3. After receiving the first SOF packet, this module interpolates the SOF by using the 60-MHz internal clock to measure 125 µs or 1 ms.
4. After receiving the second or a subsequent SOF packet, this module interpolates the SOF by using the previous reception interval.
5. Interpolation is not performed in the suspended state or while a USB bus reset is being received.
   (If this module enters the suspended state in high-speed operation, interpolation continues for 3 ms after receiving the last packet.)

The SOF interpolation function operates with the following functions:

1) Updating of frame number or micro-frame number
2) SOFR interrupt and micro SOF lock
3) SOF pulse output
4) Counting of isochronous transfer intervals

If an SOF packet is lost in full-speed operation, the FRNM bit in the FRMNUM register is not updated.

If a micro SOF packet is lost in high-speed operation, the UFRNM bit in the uFRMNUM register is updated.

However, if a micro SOF packet for which “micro-FRNM = 000” is set is lost, the FRNM bit is not updated. In this case, the FRNM bit is not updated even if subsequent micro SOF packets for which “micro-FRNM = 000” is not set are successfully received.

6.5.3.3.11 Link Power Management Processing

According to the Link Power Management specification, the existing suspend state is redefined as the L2 state and a new L1 state is defined as a state where transition and resumption at a lower latency than L2 (suspend) are possible.

The table below compares the features of the L2 (suspend) state and L1 state.

Table 6.5-42 Comparison Between Suspend (L2) State and L1 State

Item	L1	Suspend (L2)
Transition	LPM Transaction	3-ms idle period
Host-activated resumption	(Host) Minimum drive period specifiable by the host. Specified between 75 µs to 1.175 ms (Device) 10-µs K drive	(Host) Min. 20-ms K drive (Device) 10-ms K drive
Device-activated resumption	(Device) 50-µs K drive (Host) 60- to 990-µs K drive (Device) 10-µs K drive	(Device) 1-ms to 15-ms K drive (Host) Min. 20-ms K drive (Device) 10-ms K drive
Signaling	Low and Full Speed Idle	Low and Full Speed Idle

The following describes the processing for transition to and resumption from the L1 state.

(1) Descriptor

This module must return its own descriptor when receiving the GetDescriptor command.

Whether the contents of the descriptor to be returned need to be modified is dependent on whether this module responds to the transition to and resumption from the L1 state with an LPM transaction. The details are summarized in the table below.

Table 6.5-43 Relationship between LPM Response and Descriptor

LPM Response	bcdUSB	Presence of USB 2.0 Ex.Desc	USB 2.0 Ex.Desc LPM	Response When LPM Is Received	Remarks
Not respond	0200	Not present	—	Not Respond	Standard action in the case where this module does not respond to LPM
	0201	Present	LPM = 0	STALL	This is when rejection of response to LPM is explicitly declared. In this case, it is necessary to send a STALL response instead of making no response.
Respond	0201	Present	LPM = 1	ACK or NYET	Standard action in the case where this module responds to LPM

Whether to respond to transition to and resumption from L1 is declared by the LPM bit of the USB 2.0 extension descriptor. To provide this module with the USB 2.0 extension descriptor, it is necessary to set the bcdUSB field of the device descriptor to 0201 or greater.

When not responding to LPM, set the bcdUSB value to 0200 without providing this module with the USB 2.0 extension descriptor. In this case, it is necessary to ignore any LPM token received.

When not responding to LPM, it is also possible to set bcdUSB to 0201 and set the LPM bit of the USB 2.0 extension descriptor to 0 (noncompliant). In this case, however, it is not allowed to ignore LPM and is necessary to send a STALL response.

When responding to LPM, set bcdUSB to 0201 and the LPM bit of the USB 2.0 extension descriptor to 1 (compliant). This grants this module to send an NYET or ACK in response to an LPM token.

(2) Basic processing

This module needs to execute the following processing.

1. Responds to the LPM token received from the host with “No response”, “ACK”, “NYET”, or “STALL” according to this module's own state.
2. Transitions to the L1 state if it fails to detect the retransmission of an LPM token for 8 μs after making an ACK response.
3. Detects a K drive of the host and performs resume processing to the idle state.
4. Performs resume processing to the idle state based on the Remote Wake signal.

For 1, the software specifies the response method according to the values of the L1RESPEN, L1RESPMD, and L1NEGOMD bits in the PL1CTRL register. The hardware makes the response that is designated by the software upon receiving an LPM token.

For 2, both retransmission control and transition to the L1 state are processed by the hardware. Transition to the L1 state can be identified through a DVST interrupt.

For 3, a RESM interrupt occurs on detection of host K in the L1 state.

For 4, starting the Remote Wake processing can be instructed to the hardware by setting the WKUP bit by the software. The specification stipulates that the software clears this bit on resumption from the L2 state. On the other hand, the hardware clears this bit on resumption from the L1 state.

(3) HIRD value negotiation

The HIRD value contained in the LPM token is the K period of the host on resumption from the L1 state.

This module can respond with ACK if the received HIRD value falls within the desired range as specified by the L1NEGOMD and HIRDTHR bits in the L1CTRL register; otherwise, this module can respond with NYET and request the host to modify the HIRD value.

Remark This HIRD value negotiation function must also be supported on the host side.

6.5.3.3.12 DMA Mode

(1) Register mode/link mode

The DMS bit in the CHCFG_n register can be used to switch between register mode and link mode.

Table 6.5-44 DMA mode settings

DMS (CHCFG)	Mode	Description
0	Register mode	Performs DMA transfer based on the values set by Next Register Set.
1	Link mode	Accesses the descriptor area, and executes DMA transfer based on the values set by descriptors. This module repeats descriptor reading and DMA transfer unless the descriptor settings are changed or the control register is used to stop the processing.

(a) Register mode

In register mode, this module executes DMA transfer based on the values set in internal registers.

Two sets of the transfer-source address, transfer-destination address, and number of bytes to be transferred can be held (in Next0 Register Set and Next1 Register Set registers). One of these Next registers can be used to execute transfer, and both Next registers can be used to execute continuous transfer.

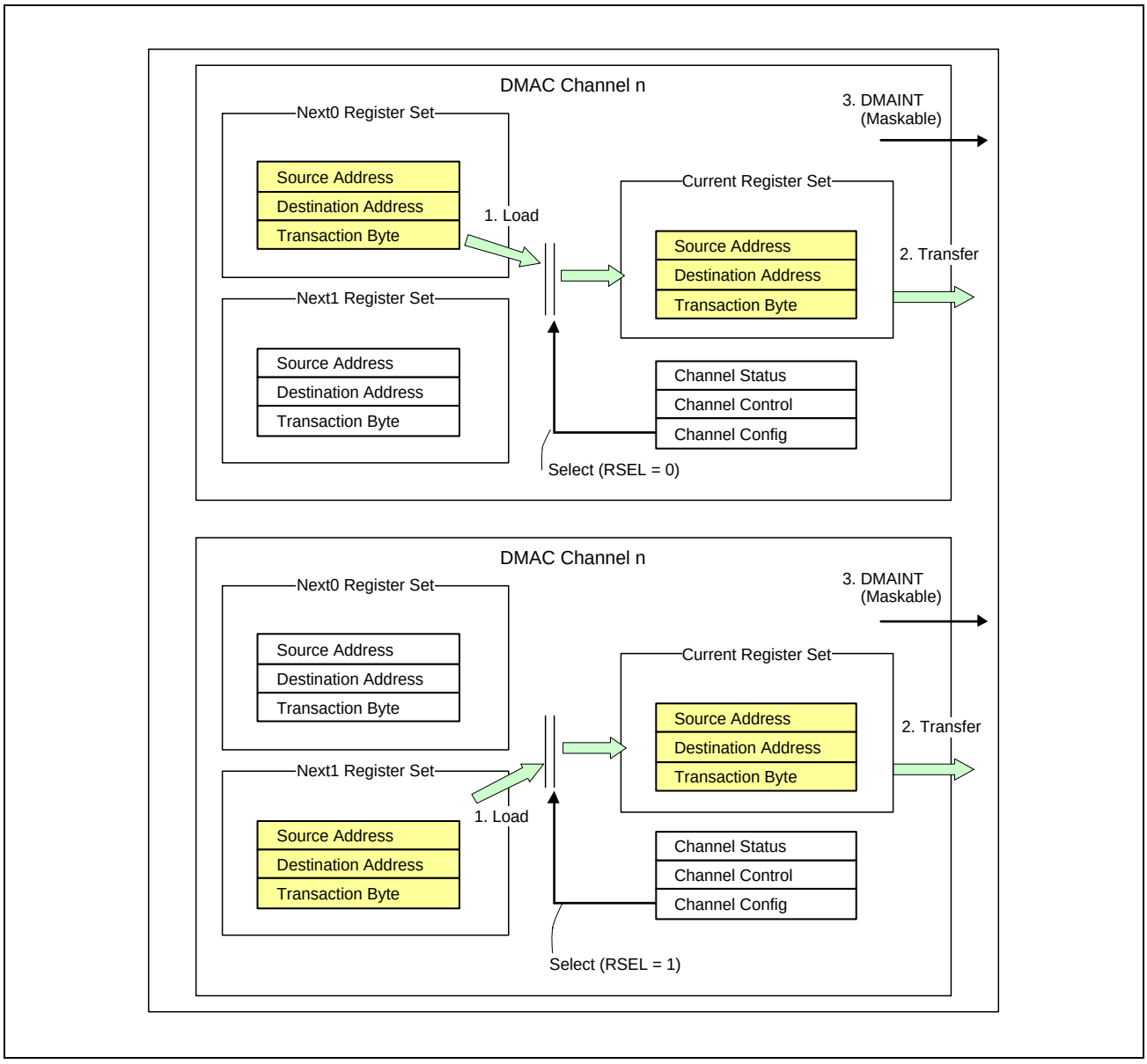


Figure 6.5-17 Overview of Normal Behavior of Register Mode

The upper part of the above figure indicates a case when Next0 Register Set is processed. The lower part of the above figure indicates a case when Next1 Register Set is processed.

a-1) Operation flow of register mode

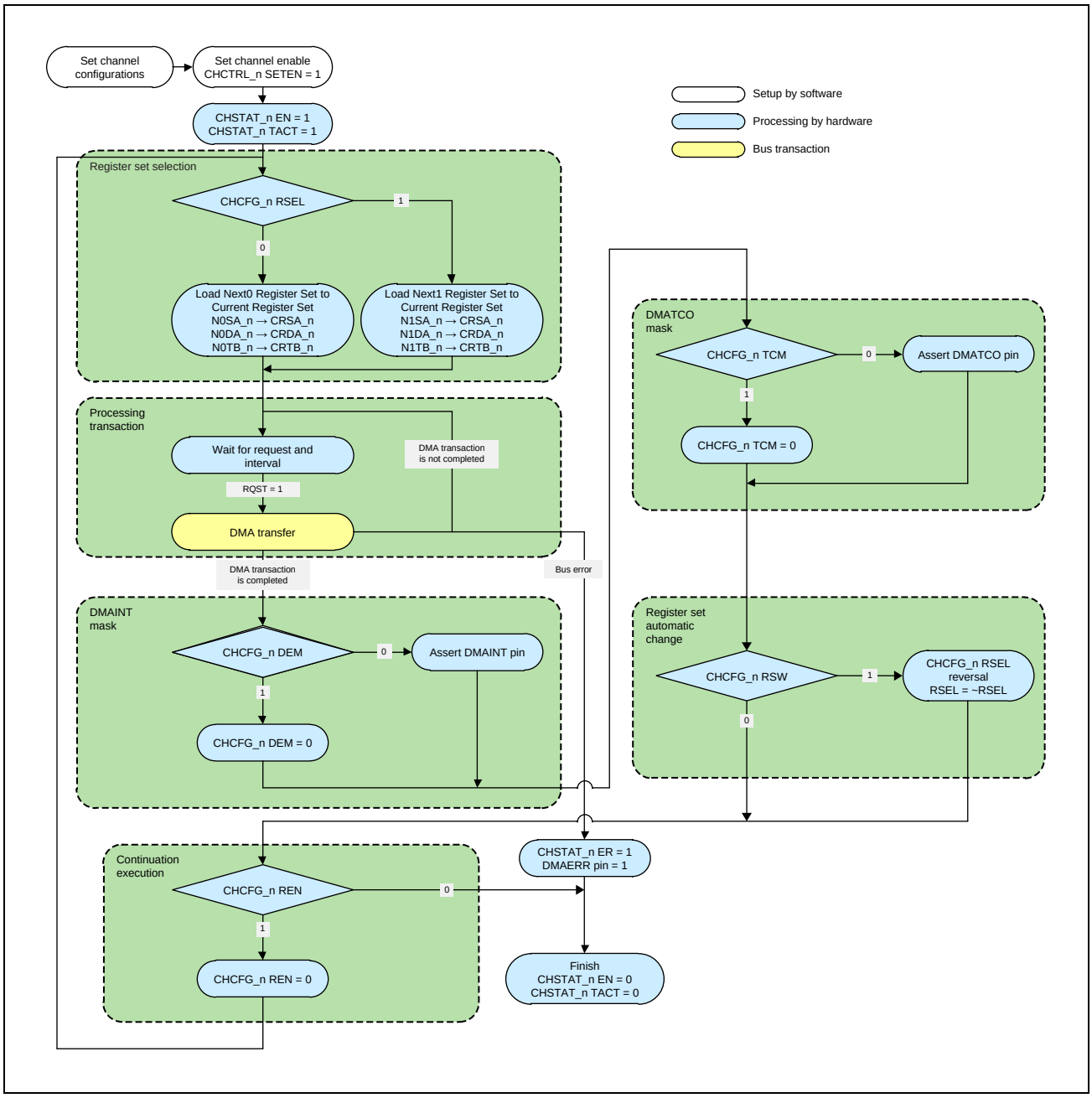


Figure 6.5-18 Register Mode Operation Flow

Description of the register mode operation flow:

1. Channel setting
   The Next0 Register Set or Next1 Register Set register is set (transfer-destination address, transfer-source address, and total number of bytes to be transferred). In addition, the FIFO channel, volume of transfer data, and other items are set for USB control that is used for the Channel Register Set register.
   No software settings are required in OUT transfer (reception in peripheral mode) or in IN transfer (transmission in peripheral mode) because the hardware sets the transfer source or destination address according to the setting of the CURPIPE bits in the DxFIFOSEL register.
   (see **6.5.3.3.13 DMA Transfer**).
2. Register set selection
   When 1 is written to the SETEN bit in the CHCTRL_n register, the EN and TACT bits in the CHSTAT_n register are set to 1. As a result, the values set by the Next Register Set register selected by the RSEL bit in the CHCFG_n register are loaded to the Current Register Set register.
3. DMA transaction
   A DMA transaction is executed based on the values that are set. For details about transfer, **6.5.3.3.13 DMA Transfer**.
4. The USBFDMAmn masking
   The USBFDMAmn interrupt is masked depending on the value of the DEM bit in the CHCFG_n register. If DEM = 1, the USBFDMAmn interrupt is masked, and the DEM bit is automatically cleared to 0.
5. DMATC masking
   DMATC from DMAC control to USB control is masked depending on the value of the TCM bit in the CHCFG_n register. If TCM = 1, DMATC is masked, and the TCM bit is automatically cleared to 0.
6. Automatic register set switchover
   Whether the current Next register set is to be switched to the other Next register set is determined by the value of the RSW bit in the CHCFG_n register.
7. Continuation of execution
   Whether to continue DMA transfer is determined by the value of the REN bit in the CHCFG_n register. If REN = 0, the EN and TACT bits in the CHSTAT_n register are cleared to 0, and DMAC operation stops. If REN = 1, DMAC operation continues, and the REN bit is automatically cleared to 0.

a-2) Register mode setting

(1) Register mode setting

The register set to be processed is selected.

Table 6.5-45 Register Mode Setting

DMS (CHCFG_n)	RSEL (CHCFG_n)	Description
0	0	Processes Next0 Register Set.
	1	Processes Next1 Register Set.

(2) USBFDMAmn masking

The USBFDMAmn interrupt can be masked.

Table 6.5-46 USBFDMAmn Mask Setting

DEM (CHCFG_n)	Description
0	Asserts the USBFDMAmn interrupt when the DMA transaction is completed.
1	Does not assert the USBFDMAmn interrupt even when the DMA transaction is completed. After the DMA transaction is completed, the DEM bit is cleared to 0.

(3) DMATC mask setting

DMATC from DMAC control to USB control can be masked.

Table 6.5-47 DMATC Mask Setting

DEM (CHCFG_n)	Description
0	Asserts DMATC when the DMA transaction is completed.
1	Does not assert DMATC even when the DMA transaction is completed. After the DMA transaction is completed, the TCM bit is cleared to 0.

(4) Automatic transaction execution for a register set

After a DMA transaction finishes, another DMA transaction can be executed.

Table 6.5-48 Automatic Execution Setting for a Register Set

REN (CHCFG_n)	Behavior	Remarks
0	The EN bit is cleared and DMA operation is terminated when the DMA transaction for the register set that is set by RSEL finishes.	Use this setting to execute a DMA transaction only once.
1	After a DMA transaction finishes, DMA transfer of the contents of the next register set continues. The REN bit is cleared to 0 when continuous transfer is successful.	Use this setting to continue processing of register set contents.

(5) Automatic register set switchover setting

After a DMA transaction finishes, the next register set to be processed can be switched.

Table 6.5-49 Automatic Execution Setting for a Register Set

RSW (CHCFG_n)	Behavior	Remarks
0	The register set is not switched when a DMA transaction finishes.	Use this setting to use only one register set.
1	When REN = 1 and a DMA transaction finishes, the RSEL setting is automatically reversed to select the other register set.	Use this setting to switch the register set.

a-3) Example of setting the register mode

(1) Example of setting the register mode when using only the Next0 register set

Table 6.5-50 Register Mode Setting Example

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	TCM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0	0	0	0	0	0
(register mode)	(Next0)	(not masked)	(not masked)	(not switched)	(continuous execution disabled)

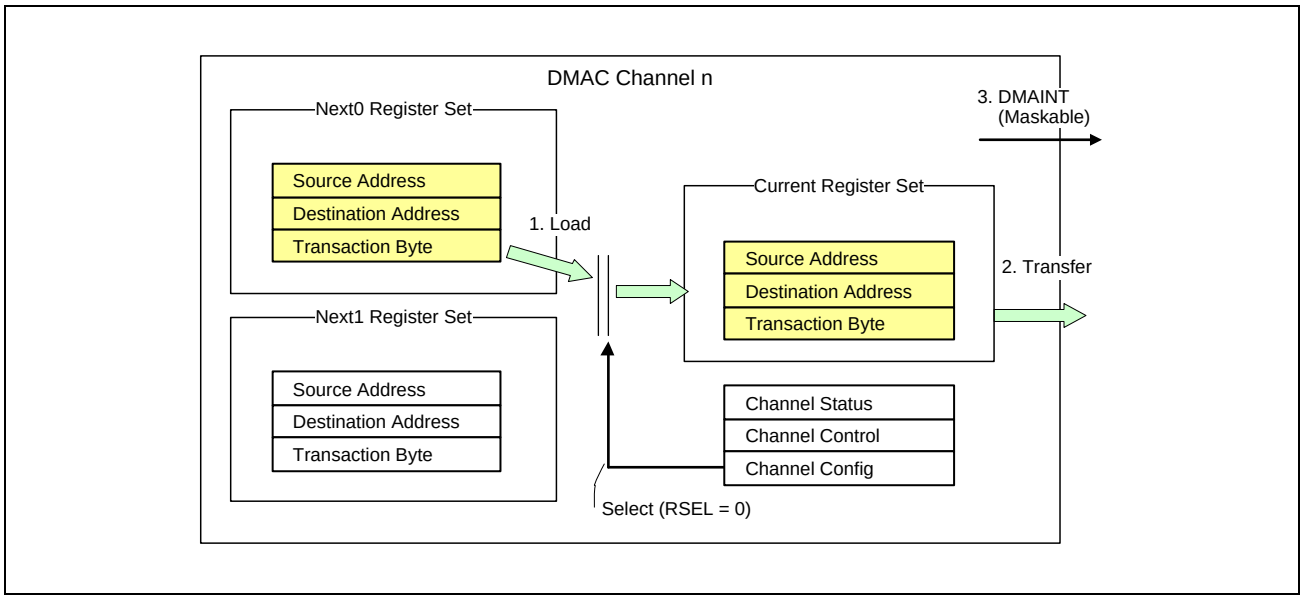


Figure 6.5-19 Register Mode Setting Example 1

1. By writing 1 to the SETEN bit in the CHCTRL_n register, the EN bit in the CHSTAT_n register is set to 1. As a result, the contents of Next0 Register Set are loaded to Current Register Set.
2. A DMA transaction is executed based on the values of Current Register Set and Channel Register Set.
3. Because the DEM bit in the CHCFG_n register is 0, the USBFDMAmn interrupt is asserted after the DMA transaction finishes.
4. Because the TCM bit in the CHCFG_n register is 0, DMATC is asserted after the DMA transaction finishes.
5. Because the REN bit in the CHCFG_n register is 0, the EN bit in the CHSTAT_n register is cleared to 0, and the operation ends.

(2) Example of setting the register mode when using two register sets continuously

Table 6.5-51 Automatic Register Set processing Setting

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	TCM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0 (register mode)	0 (Next0)	1 (masked)	0 (not masked)	1 (switched)	1 (continuous execution enabled)

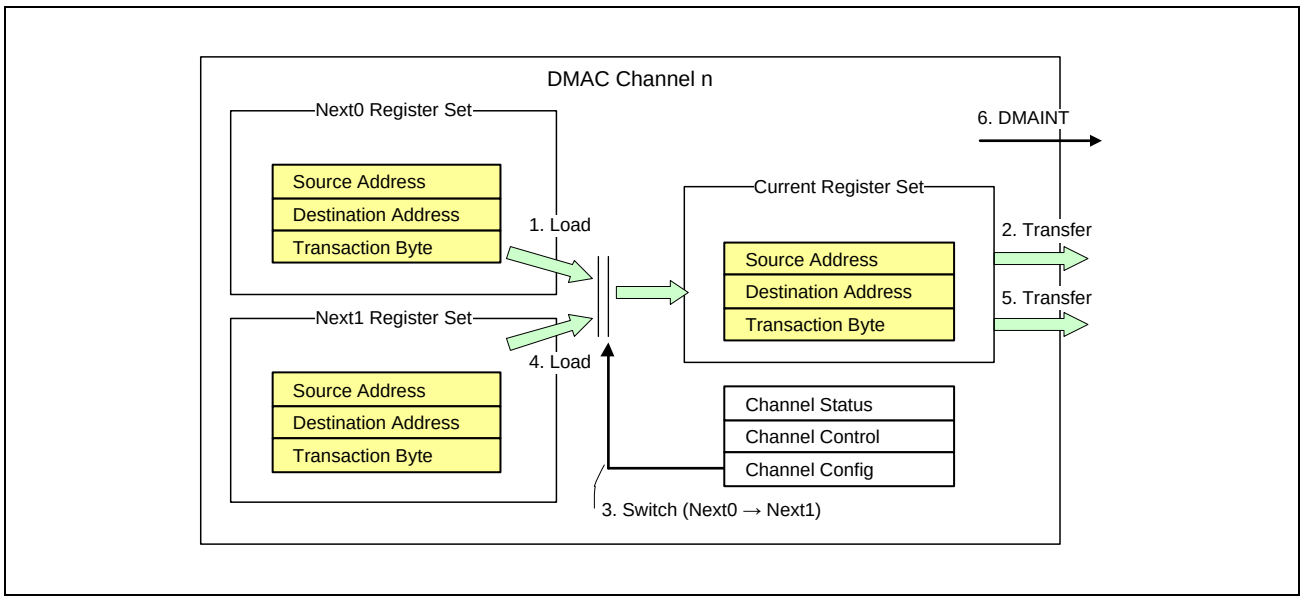


Figure 6.5-20 Register Mode Setting Example 2

1. By writing 1 to the SETEN bit in the CHCTRL_n register, the EN bit in the CHSTAT_n register is set to 1. As a result, the contents of Next0 Register Set are loaded to Current Register Set.
2. A DMA transaction is executed based on the values of Current Register Set and Channel Register Set.
3. Because the DEM bit in the CHCFG_n register is 1, the USBFDMAmn interrupt is not asserted after the DMA transaction finishes.
4. Because the REN bit in the CHCFG_n register is 1, operation continues. The REN bit is automatically cleared to 0.
5. Because the RSW bit in the CHCFG_n register is 1, the next register set to be processed is switched (RSEL = 0 to 1).
6. The contents of Next1 Register Set are loaded to Current Register Set.
7. A DMA transaction is executed based on the values of Current Register Set and Channel Register Set.
8. Because the DEM bit in the CHCFG_n register is 0, the USBFDMAmn interrupt is asserted after the DMA transaction finishes.
9. Because the TCM bit in the CHCFG_n register is 0, DMATC is asserted after the DMA transaction finishes.
10. Because the REN bit in the CHCFG_n register is 0, the EN bit in the CHSTAT_n register is cleared to 0, and the operation ends.

(b) Link mode

In link mode, this module reads the value set in a descriptor placed in an external storage area to execute a DMA transaction. In DMAC, there are Next Link Address (NXLA_n) and Current Link Address (CRLA_n) registers for each channel. The Next Link Address (NXLA_n) register is used to set the address of the descriptor to be read the next time. The Current Link Address (CRLA_n) register is used to display the descriptor address for the current DMA transaction.

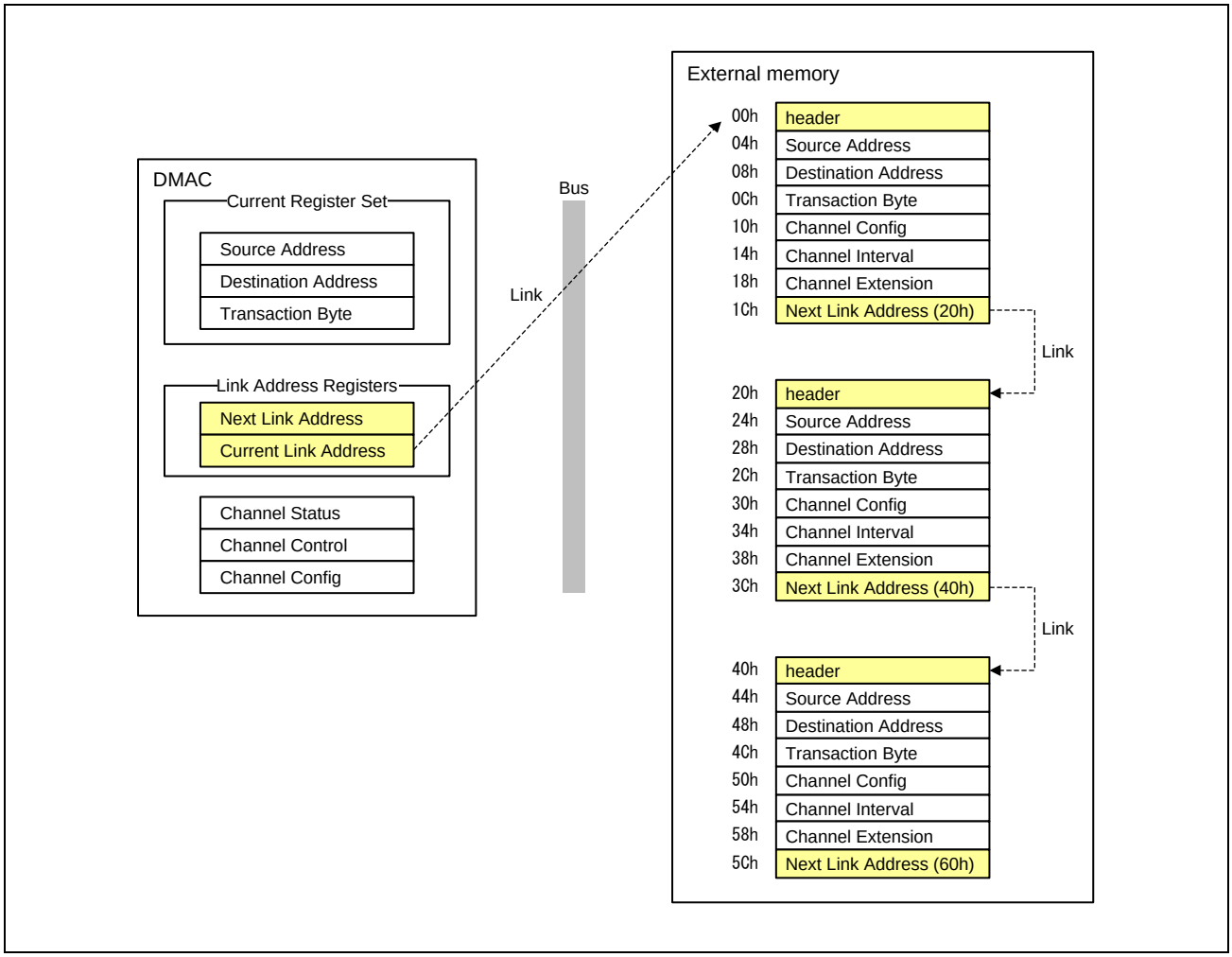


Figure 6.5-21 Overview of Link Mode

b-1) Operation flow of link mode

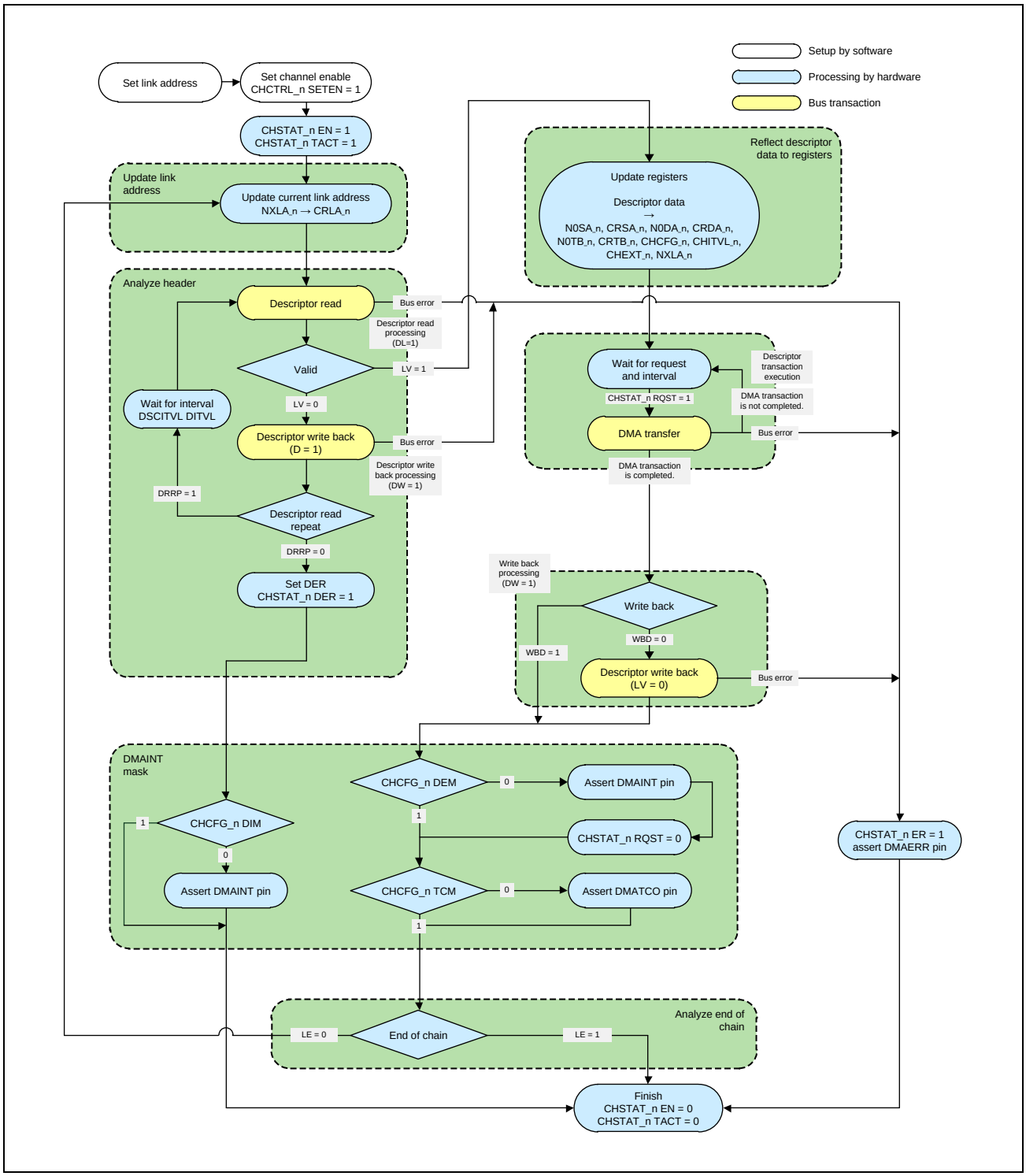


Figure 6.5-22 Link Mode Operation Flow

Description of the link mode operation flow:

1. Channel setting
   The beginning address of the link destination is set in the NXLA_n register.
2. Link address updating
   If 1 is written to the SETEN bit in the CHCTRL_n register, the EN and TACT bits of the CHSTAT_n register are set to 1. As a result, the address set in the NXLA_n register is loaded to the CRLA_n register.
3. Descriptor reading and header judgment
   A read of the descriptor starts, and DMAC checks the contents of header. If LV = 0, this module writes 1 back to the D bit of header. After that, if the DRRP bit in the CHCFG_n register is 1, this module waits for the time intervals set by the DSCITVL register, and then reads the same descriptor again. If DRRP = 0, the DER bit in the CHSTAT_n register is set to 1, and this module is placed in the end state (both EN and TACT bits in the CHSTAT_n register are 0). At this time, if the DIM bit in the CHCFG_n register is 0, this module asserts the USBFDMAmn interrupt.
4. Descriptor setting
   If LV = 1, the data read from the descriptor is loaded to Current Register Set and Channel Register Set. In addition, the next link target is loaded to the NXLA_n register.
5. DMA transaction
   A DMA transaction is executed based on the values that are set. For details about transfer, see **6.5.3.3.13 DMA Transfer**.
6. Header write-back
   If WBD of header is 0, DMAC writes LV = 0 to the header area.
7. USBFDMAmn masking
   If the DEM bit in the CHCFG_n register is 0, this module asserts the USBFDMAmn interrupt.
8. DMATC masking
   If the TCM bit in the CHCFG_n register is 0, this module asserts DMATC.
9. Link end judgment
   If LE of header is 1, the EN and TACT bits in the CHSTAT_n register are cleared to 0, and DMAC terminates operation. If LE is 0, this module updates Current Register Set, and then restarts reading the next descriptor.

b-2) Register setting

(1) Link mode setting

To use link mode, set the DMS bit in the CHCFG_n register to 1.

Table 6.5-52 Link Mode Setting

DMS (CHCFG_n)	Description
1	This module operates in link mode. The setting of this bit cannot be changed by using a descriptor.

(2) Link address setting

The Next Link Address (NXLA_n) and Current Link Address (CRLA_n) registers are used to indicate a link target. Before starting link mode, set the link target in the NXLA_n register.
After reading a descriptor, this module updates the NXLA_n register to the next link. Note that the CRLA_n register indicates the address of the link target that is being executed.

Table 6.5-53 Link Address Register Set

Register	Description
Next Link Address Register (NXLA_n)	This register is used to set and display the next link target. Before starting link mode, set the address of the link target in this register.
Current Link Address Register (CRLA_n)	This register is used to display the link target that is being executed. This is a read-only register.

b-3) Descriptor setting

DMAC supports multiple descriptor formats.

A switchover between formats is specified by using the DSCFM field of bits [31:28] of the 1st word (header) of the descriptor.

The following table shows the relationship between DSCFM values and descriptor formats.

Table 6.5-54 Descriptor Formats

DSCFM	Descriptor Size	Next Link Address	Channel Extension	Channel Interval	Channel Config	Transaction Size	Destination Address	Source Address	header
3	4 words	✓	— (reload)	— (reload)	— (reload)	— (header)	✓	✓	✓ (with STS)
1	8 words	✓	✓	✓	✓	✓	✓	✓	✓ (no STS)
Other than the above	If DSCFM is set to a value that is not 1 or 3, operation cannot be guaranteed. Make sure that DSCFM is set to 1 or 3.								

Table 6.5-55 Explanation of the Marks in **Table 6.5-54**

Field	Mark	Description	Remarks
Header	✓ (with STS)	The STS field of bits [15:0] in the header is valid. The value set in the STS field is used as the total number of transfer bytes (Transaction Size).	—
	✓ (no STS)	The STS field of bits [15:0] in the header is invalid. The value of "Transaction Size" in the descriptor is used as the total number of bytes.	
Source Address	✓	Specify the source address.	—
Destination Address	✓	Specify the destination address.	—
Transaction Size	✓	Specify the transaction size.	—
	— (header)	Omit the transaction size. The value set in the STS field is used as the total number of transfer bytes (Transaction Size)	Because the STS field is of 16 bits, a maximum of 65,535 bytes can be set.
Channel Config Channel Interval Channel Extension	✓	Specify Channel Config, Channel Interval, and Channel Extension.	—
	— (reload)	Omit Channel Config, Channel Interval, and Channel Extension. The previous settings (the values of the CHCFG_n, CHITVL_n, and CHEXT_n registers of the last time) are inherited.	—
Next Link Address	✓	Specify the next descriptor address (Next Link Address) to be read after DMA transfer of the descriptor.	—

DMAC sequentially interprets data read from descriptors. If the number of words specified for DSCFM is less than 8, place the data of descriptors that are indicated by "✓" in **Table 6.5-54** on memory.

No software settings are required in OUT transfer (reception in peripheral mode) or in IN transfer (transmission in peripheral mode) because the hardware sets the transfer source or destination address according to the setting of the CURPIPE bits in the DxFIFOSEL register.

Table 6.5-56 Example of Placing Descriptors

DSCFM	Address							
	Link Address + 1Ch	Link Address + 18h	Link Address + 14h	Link Address + 10h	Link Address + 0Ch	Link Address + 08h	Link Address + 04h	Link Address + 00h
3h	—	—	—	—	Next Link Address	Destination Address	Source Address	header
1h	Next Link Address	Extension	Interval	Config	Transaction Byte	Destination Address	Source Address	header

- header

The header area provides the descriptor status and other information as shown below.

DMAC reads this area before DMA transfer in link mode starts. After a DMA transaction terminates, DMAC writes the transfer status back to this area.

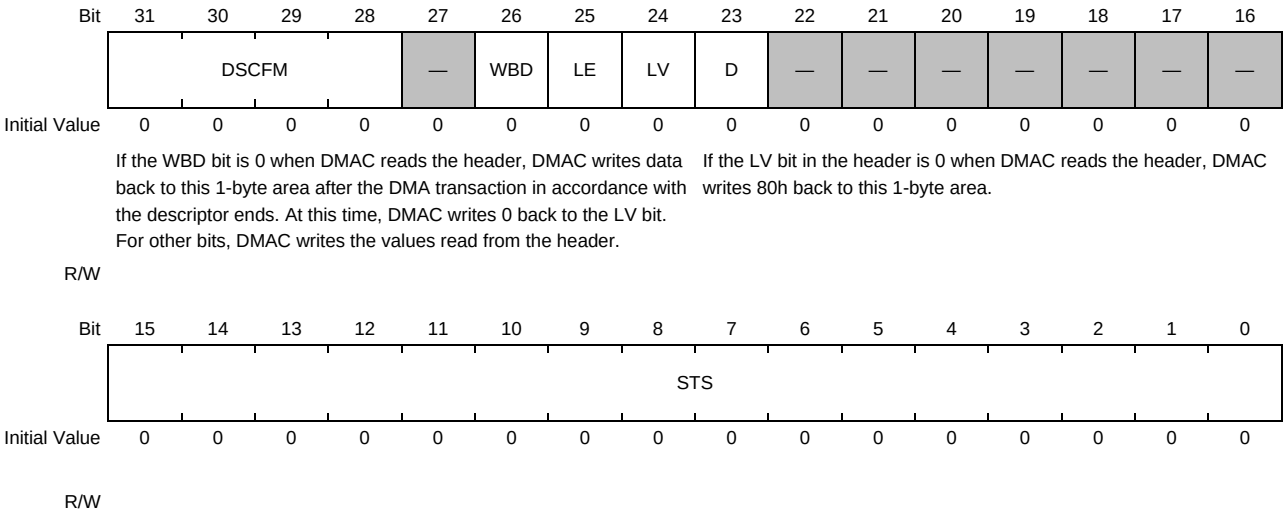


Figure 6.5-23 Header Area

Table 6.5-57 Header Area (1/2)

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DSCFM			Descriptor Format Specifies the descriptor format (length and combination of descriptors). For details see **Table 6.5-54**.
27	—			Reserved area. Set 0.
26	WBD			Write Back Disable Masks a write-back operation for the LV bit. If this bit is 1, DMAC does not perform a write-back operation. 0: Writes 0 back to the LV bit. 1: Does not perform a write-back operation for the LV bit.
25	LE			Link End Indicates that the link will end with the DMA transaction for this descriptor. Set this bit to 1 to indicate the end of link. 0: The link continues. 1: The link ends.

Table 6.5-57 Header Area (2/2)

Bit	Bit Name	Initial Value	R/W	Description
24	LV			Link Valid Indicates that this descriptor is valid. If WBD = 0, after DMAC executes the DMA transaction written in the descriptor, DMAC writes 0 to this bit. When header is set, set 1 to this bit. 0: This descriptor is invalid. 1: This descriptor is enabled.
23	D			Descriptor Error Indicates a descriptor access error. If LV is 0 when the descriptor is read, DMAC writes 1 back to this bit. 0: A descriptor error has not occurred. 1: LV was 0 when the descriptor was read.
22 to 16	—			Reserved area. Set 0.
15 to 0	STS			Short Transaction Size If DSCFM is 3, the transaction size is set (in bytes). The maximum number of transfer bytes that can be set is 65,535. If DSCFM is 3, Do not set 0 for STS. If 0 is set, operation cannot be guaranteed.

If descriptors are added sequentially while DMAC is operating, the access that the CPU sets 0 to the LV bit and the access that DMAC writes 1 back to the D bit might contend with each other. If this contention occurs, prior-written data is overwritten by latter-written data.

To prevent this problem from occurring, DMAC performs a write-back operation for the D bit in a byte-write manner. Therefore, the CPU must also set LV to 1 in a byte-write manner. Because the byte lanes for the D and LV bits are different, by writing data to different areas, occurrence of this problem can be prevented.

- Settings of descriptors other than header

  The specifications of data of the descriptors other than header are the same as the specifications of internal registers.

- Settings specified when descriptors are accessed

  The MHPROT pin output can be set for the LWPR and LDPR fields of the DCTRL register when descriptors are accessed. Set it according to the access target in which descriptors are deployed.

- Descriptor areas and DMA transfer areas

  The following provides an overview of the descriptor areas and DMA transfer areas that are accessed by DMAC.

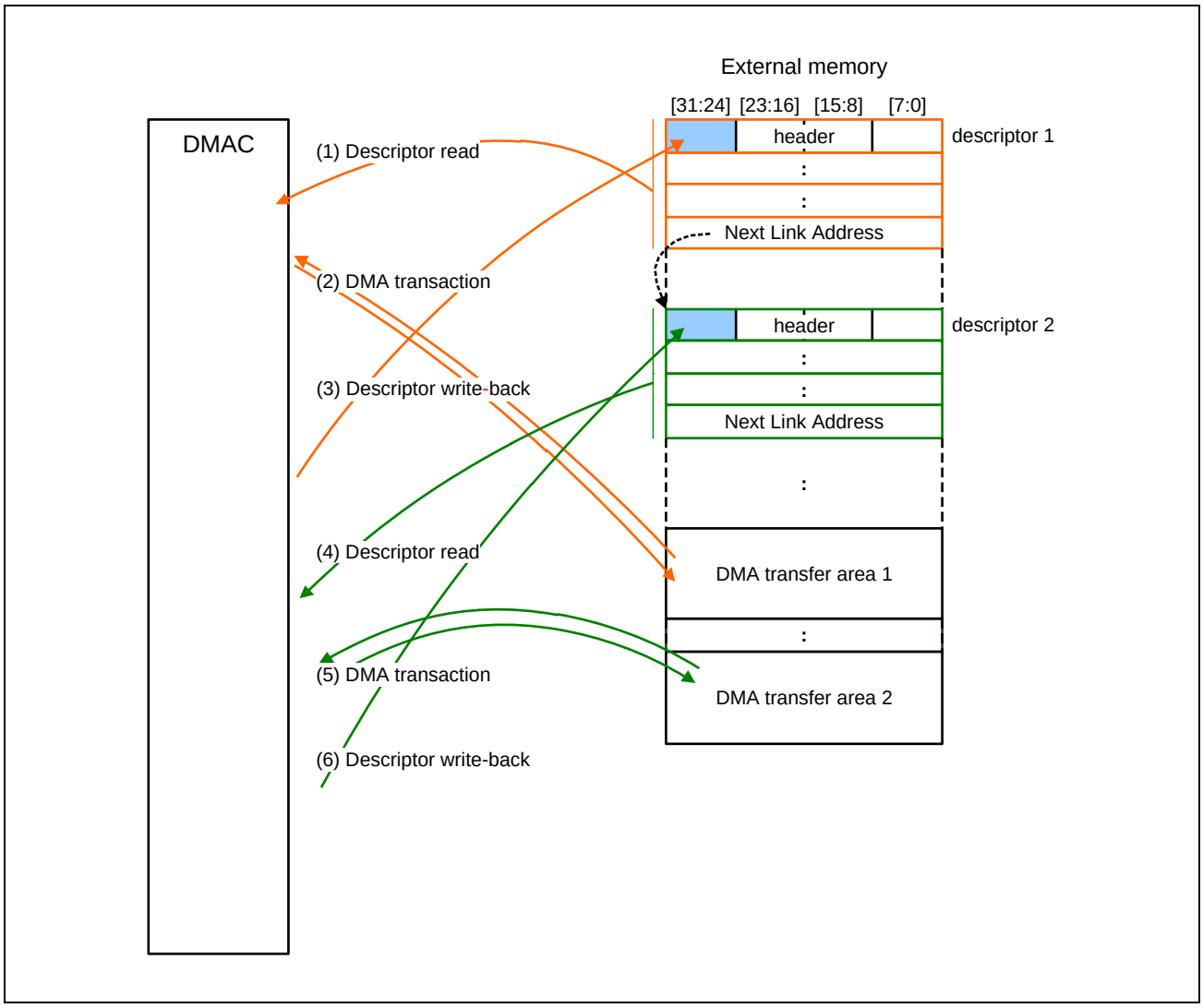


Figure 6.5-24 Overview of Descriptor Areas and DMA Transfer Areas

(1) Descriptor read
DMAC loads a value from the internal NXLA_n register to the CRLA_n register, and then reads a descriptor from the external memory space indicated by the CRLA_n register (descriptor 1).

(2) DMA transfer
If the LV bit of the header descriptor is 1, DMAC performs a DMA transfer in accordance with the descriptor information.

(3) Descriptor write-back
After performing a DMA transfer of data by the number of bytes that are set, if the WBD bit in header is 0, DMAC performs write-back for bits [31:24] in header of Descriptor 1. For the LV field, 0 is written back. For other fields, the values read in (1) are written back on a byte-size basis.

(4) Descriptor read
If the value of the LE bit in the header descriptor that was read previously (in (1)) is 0, DMAC reads the next descriptor from the address (descriptor2) indicated by Next Link Address in the current descriptor.

(5) DMA transfer
If the LV bit in the header descriptor is 1, DMAC performs a DMA transfer in accordance with the descriptor information.

(6) Descriptor write-back
After performing a DMA transfer of data by the number of bytes that are set, if the WBD bit in header is 0, DMAC performs write-back for bits [31:24] in header of Descriptor 2. For the LV field, 0 is written back. For other fields, the values read in (4) are written back on a byte-size basis.

(Steps (4) to (6) are repeated.)

If LE = 1 and WBD = 0 in header, DMAC performs a DMA transfer with the descriptor settings, writes 0 back to the LV bit in header, and then terminates processing.

If LE = 1 and WBD = 1 in header, DMAC performs a DMA transfer with the descriptor settings, and then terminates processing (without performing write-back).

If LV = 0 in header, DMAC writes 1 back to the D bit in header, and then, if the DRRP bit in the CHCFG_n register is 1, DMAC waits for the number of intervals specified by the DITVL field of the DSCITVL_n register, and then reads the descriptor again. If DRRP = 0, DMAC terminates processing.

- Notes on descriptors
  - In link mode, settings can be changed by reading descriptors. However, it is impossible to synchronize the setting change times and hardware requests. Therefore, to use hardware requests, before setting the SETEN bit in the CHCTRL_n register, set the AM, LVL, HIEN, LOEN, and SEL bits in the CHCFG_n register. Note that these setting bits must not be changed in descriptors.
  - Descriptors cannot be used to change the DMS field in the CHCFG_n register (DMAC is always placed in link mode). Although descriptors can be used to change the settings of the REN, RSW, and RSEL fields in the CHCFG_n register, changes of those fields do not affect operation.
  - The descriptor can be initialized by overwriting the memory area corresponding to the descriptor you intend to initialize while the DMAC is not operating. The DMAC determines whether or not the descriptor is valid by referring to the DSCFM field and LV bit in the header. Accordingly, set the areas in memory corresponding to the DSCFM field and LV bit to 1 or 3, and to 1, respectively, before enabling DMAC operation.
  - To set the next descriptor on memory while DMAC is operating, make sure that 1 is written to the LV bit after the descriptors subsequent to header (Source Address, Destination Address, ..., Next Link Address) are set. If this is not performed and descriptor setting by the CPU and descriptor reading by DMAC contend, DMAC performs a DMA transfer using the previous values of those descriptors (Source Address, Destination Address, ..., Next Link Address).
  - To leave the write-back information for the D bit of header, make sure that 1 is written to the LV bit of header in a byte-access manner.

b-4) Link configuration example

In link mode, descriptors can be configured as shown below.

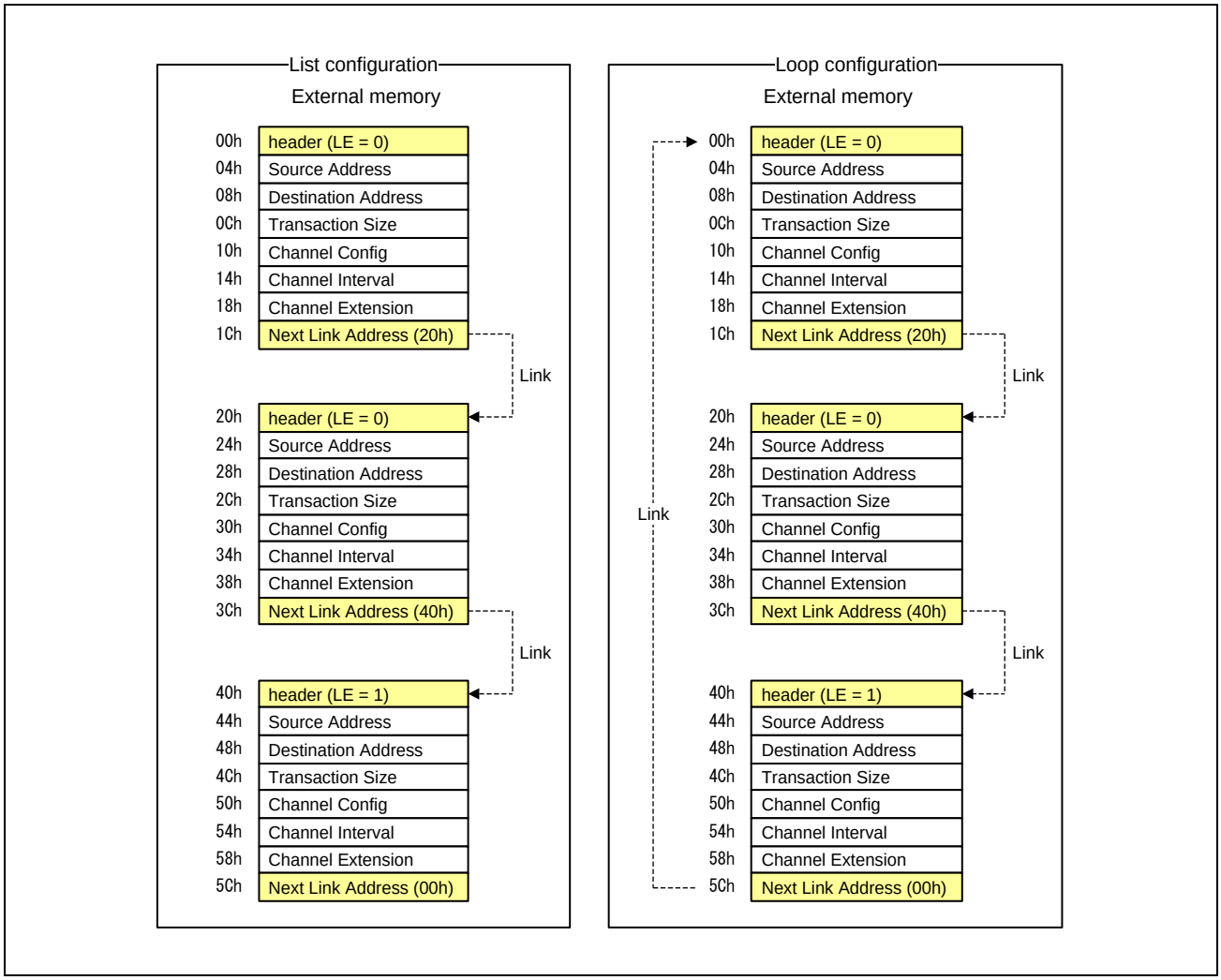


Figure 6.5-25 Link Mode Configuration Example

- List configuration

  The link ends by setting 1 for the LE bit in the header of the last descriptor.

- Loop configuration

  A loop of descriptors can be created by setting the link target of the last descriptor to the address of the first descriptor. To end the loop, change the value of the LE bit of header to 1 or use the transfer interrupt procedure.

(2) Write only mode

Write-only mode is enabled by setting 1 for the WONLY bit in the CHCFG_n register.

Table 6.5-58 Write-Only Mode Setting

WONLY (CHCFG)	Mode	Description
0	Normal mode	A DMA transfer is performed using the values set in Next Register Set.
1	Write-only mode	A DMA write transfer is performed without performing a DMA read transfer.

In write-only mode, no DMA read transfer is performed (note that descriptors are read in the same way as in normal mode). In register mode, the values set in the NxSA_n register (if RSEL = 0, x = 0; if RSEL = 1, x = 1) are used as write data. In link mode, the values of the SA fields of descriptors are used as write data.

Use this mode to, for example, initialize the memory area.

6.5.3.3.13 DMA Transfer

This section describes the basic operation of DMA transfer.

(1) Transfer modes

DMAC supports only single transfer mode.

Upon receiving a DMA transfer request from USB control, DMAC executes a single DMA transfer on the side (source or destination) indicated by the REQD bit in the CHCFG_n register. DMAC then asserts internal DMA permission from internal USB control to DMAC control. DMAC performs a single transfer each time a transfer is received. DMAC continues this operation by the transfer size loaded to the CRTB_n register (arbitration between channels is performed for each DMA transfer).

The timing of internal DMA permission from internal USB control to DMAC control differs depending on the setting of the REQD bit in the CHCFG_n register and the setting of the transfer size (DDS[2:0] and SDS[2:0] in the CHCFG_n register). For details, see **(7) Operational difference depending on the transfer size**.

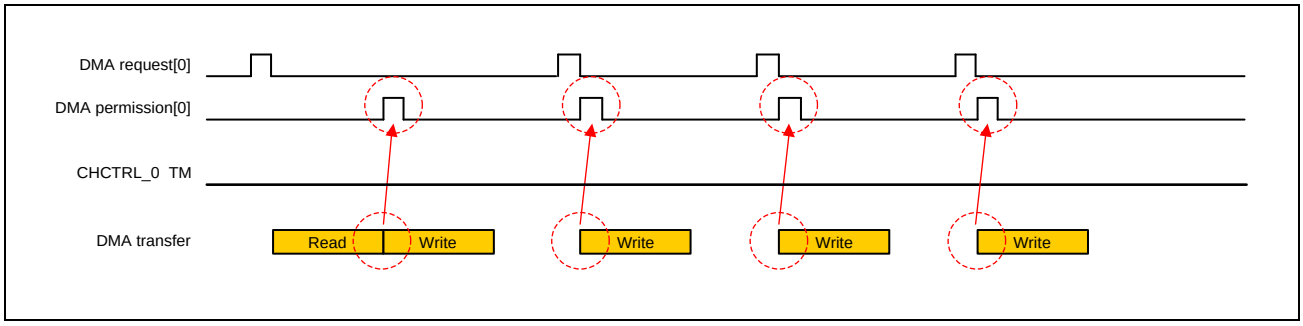


Figure 6.5-26 Single Transfer Mode (REQD = 1, SDS > DDS)

(2) DMA channel priority control

DMAC supports fixed-priority mode and round-robin mode as methods of arbitration between channels. The mode is selected by using the PR bit in the DCTRL register. If the PR bit is 0, fixed-priority mode is selected. If the PR bit is 1, round-robin mode is selected.

Table 6.5-59 Priority Control Setting

Mode	PR (DCTRL)	Description	Remarks
Fixed-priority	0	Controls requests based on fixed priority (CH0 > CH1 > ...).	Use this mode if channels have priority.
Round-robin	1	Controls requests in a round-robin manner.	Use this mode to execute requests equally.

(a) Fixed-priority mode

In fixed-priority mode, a fixed priority level is assigned to each channel as shown below.

(High) CH0 > CH1 (Low)

If DMA transfer requests simultaneously occur over multiple channels, the DMA transfer over the channel whose number is smallest is performed first.

The transfer over channel 0 is performed first. However, while a transfer switches to another transfer over channel 0, a transfer over the channel with the next highest priority level is performed in order to increase the bus usage rate.

(b) Round-Robin Mode

In round-robin mode, each time a transfer over a channel is received, the priority level of the channel that was used for the previous transfer is changed to the lowest level.

In the status immediately after the mode is reset, channels are assigned priority levels in the same way as fixed-priority mode as shown below.

(High) CH0 > CH1 (Low)

In this status, if a transfer request for DMA channel 0 does not occur and a transfer request for DMA channel 1 occurs, the transfer over DMA channel 1 is performed. When the transfer finishes, the channel priority is changed as follows.

(High) CH1 > CH0 (Low)

The following shows an example of DMA transfer in round-robin mode.

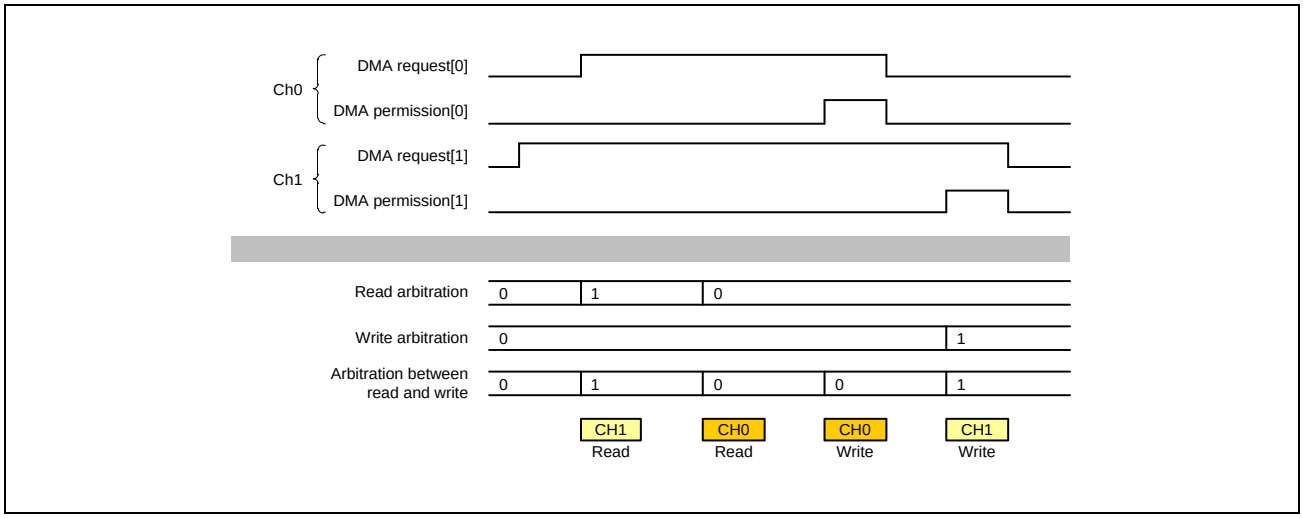


Figure 6.5-27 Example of Operation in Round-Robin Mode (with 4 channels, REQD = 1)

DMAC internally performs arbitration between read channels and arbitration between write channels, further performs arbitration between the arbitration results, and then issues bus access.

(3) Forced sweeping request

When a forced sweeping request is entered, DMAC transfers the data that is left untransferred in the buffer to the destination address. After the sweep processing finishes, DMAC continues DMA transfer.

The following provides notes on forced sweeping requests:

(1) If a forced sweeping request and a transfer request from USB control contend, DMAC performs forced sweeping first, and then performs a DMA transfer.

(2) If the USB-control-side system is the destination (REQD bit in the CHCFG_n register is 1) register REQD = 1, buffer overflow or another error might occur on the destination unit because data is transferred although no DMA transfer request is made on the USB control side. Therefore, the specifications physically prohibit DMAC from using forced sweeping if REQD = 1.

(3) The difference from ordinary sweeping mode described in **c-2) Transfer suspension (buffer sweeping enabled: SBE = 1)** (EN is cleared by setting 1 for the SBE bit in the CHCFG_n register) is as follows: DMAC stops operation after writing data in the buffer in ordinary sweeping mode, whereas DMAC can continue DMA transfer after sweeping the buffer in forced sweeping mode.

(a) Forced software sweeping request

The SETSSWPRQ bit in the CHCTRL_n register determines whether software-based forced sweeping requests can be used. To perform a forced sweeping request, write 1 to the SETSSWPRQ bit. DMAC then outputs the data in the buffer to the destination.

(4) DMA transfer completion interrupt (USBFDMAmn)

USBFDMAmn (m, n = 0, 1) is an interrupt signal that indicates termination of a DMA transaction.

If a transfer for the total number of transfer bytes loaded to the CRTB_n register is completed by an OKAY response, the END bit in the CHSTAT_n register is set to 1. At this time, if the DEM bit in the CHCFG_n register is 0, DMAC generates a USBFDMAmn (m, n = 0, 1) interrupt.
(If a write-back operation is performed in link mode, the interrupt is generated after the write-back operation finishes.)

In link mode, when the DRRP bit in the CHCFG_n register is 0, if the LV bit of the header of the descriptor that is read is 0, the DER bit in the CHSTAT_n register is set to 1. At this time, if the DIM bit in the CHCFG_n register is 0, DMAC generates a USBFDMAmn interrupt.

Use this signal to detect a transfer completion interrupt performed by the interrupt controller.

Table 6.5-60 USBFDMAmn Assertion Conditions

Cause	Condition	INT_DMA[n] mask signal
DMA transaction ended	A transfer of data by the number of transfer bytes loaded to the CRTB_n register is completed by an OKAY response (if a write-back operation is performed in link mode, after the operation finishes)	DEM bit in the CHCFG_n register
Descriptor was invalid	LV of header of the descriptor that is read is 0 when DRRP and DIM in the CHCFG_n register are both 0 in link mode	DIM bit in the CHCFG_n register

(5) DMA error interrupt (USBFDMAERRm)

If an error response is received for DMA transfer or descriptor access, this module stops transfer, assuming that an error occurred. When an error response is received, the EN bit in the CHSTAT_n register for channel n that is being used for transfer is cleared to 0, and the ER bit is set to 1 (n = 1, 0). Also, the USBFDMAERRm interrupt is asserted.

The USBFDMAERRm signal cannot be masked.

For a sequence of transfers for which an error occurred, data integrity cannot be guaranteed. Always use the following procedure to restart the transfer sequence from the beginning.

1. Set the SWRST bit in the CHCTRL_n register to 1.
2. Reset each register.

(6) Interval Count Function

The execution interval of a DMA transfer can be adjusted by using the ITVL field in the CHITVL_n register. This function prevents DMAC from continuously occupying the bus. If this function is enabled, DMAC does not perform a DMA transfer for the next request until the counter value becomes 0.

The following shows an operation example.

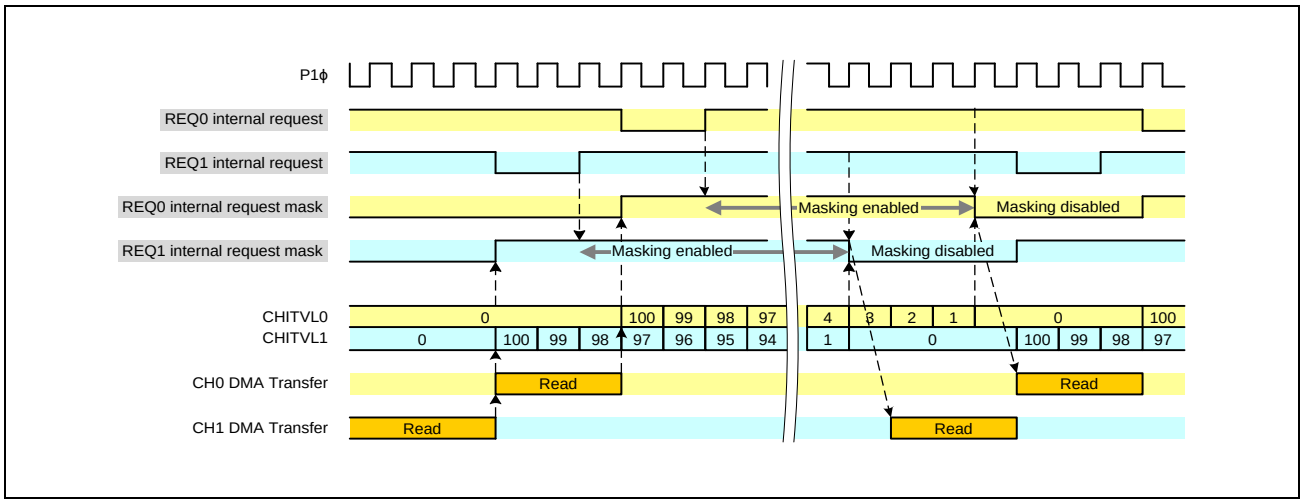


Figure 6.5-28 Example of Counting Intervals (REQD = 0, SDS < DDS)

An interval is inserted after a transfer is performed on the side specified by the REQD bit in the CHCFG_n register. The following shows how the REQD, SDS, and DDS values of the CHCFG_n register are related with the interval.

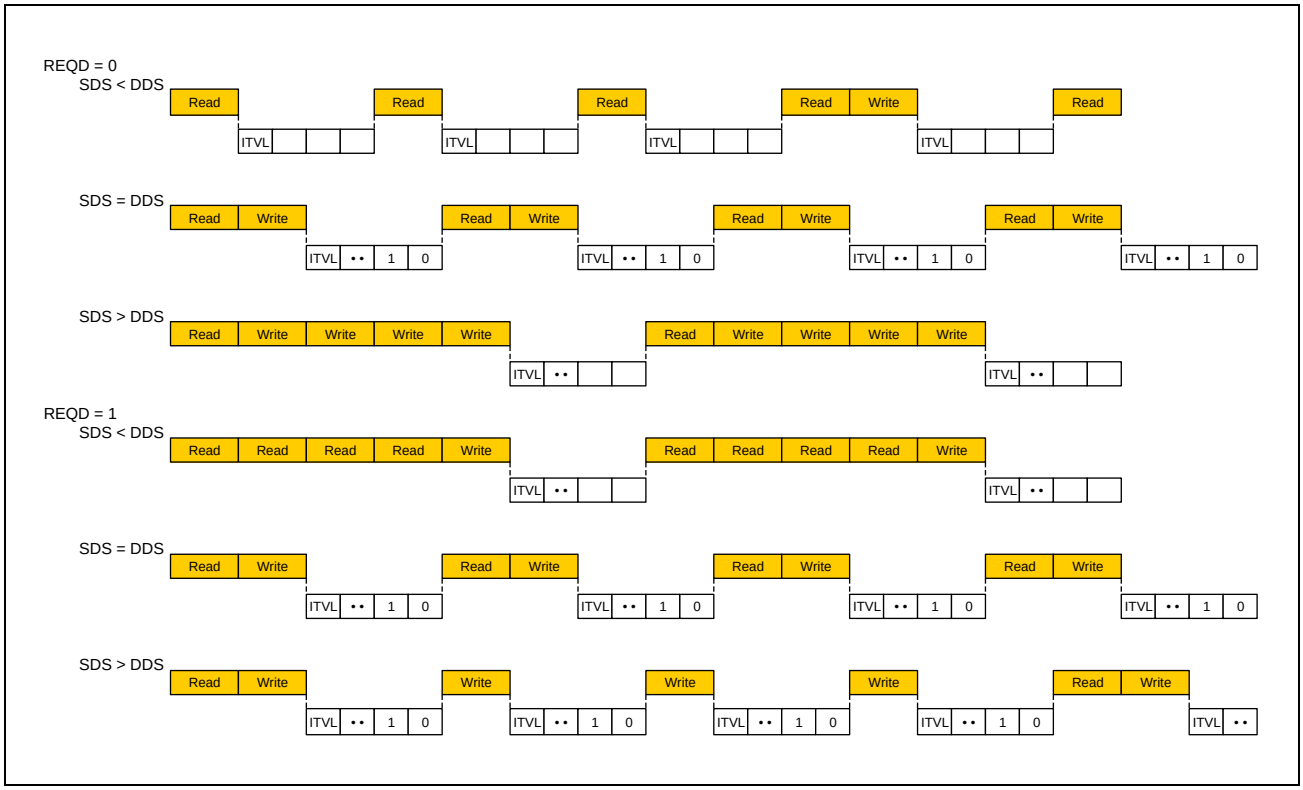


Figure 6.5-29 DMA Transfer Settings and Interval Count

(7) Operational difference depending on the transfer size

(a) If the transfer size on the source side is smaller

When reading of as much data as the destination data size finishes, a write to the destination starts.

The following figure is an example of the timing chart in the case where the source is an 8-bit field and the destination is a 32-bit field (SDS = 0 and DDS = 2 in the CHCFG_n register) (when the rising edge is detected).

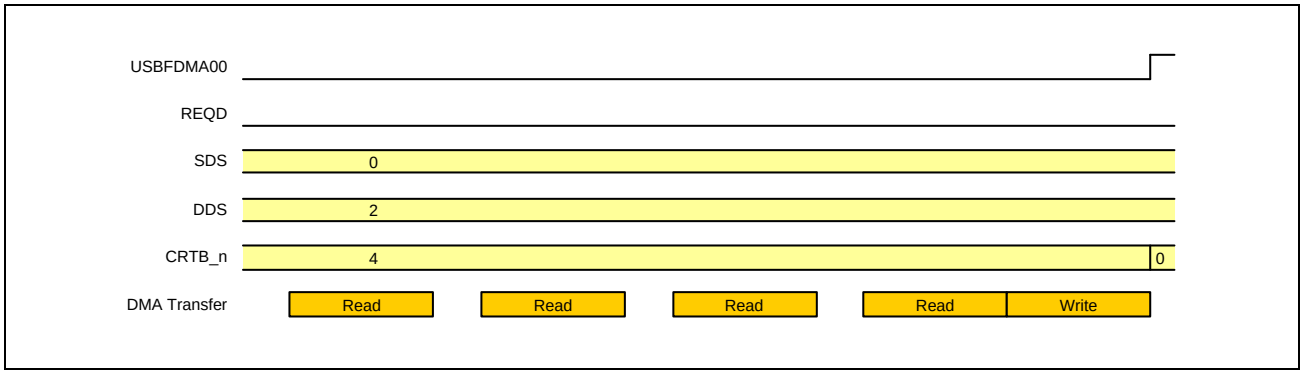


Figure 6.5-30 Example of Timing Chart in the Case Where the Source is Smaller
(LVL = 0, HIEN = 1, REQD = 0, and SDS < DDS)

(b) If the transfer size on the destination side is smaller

Because the source side is larger than the destination side, two or more destination write operations occur for one source read operation. The following is an example of the timing chart in the case where the source is a 64-bit field and the destination is a 16-bit field (SDS = 3 and DDS = 1 in the CHCFG_n register) (when the rising edge is detected).

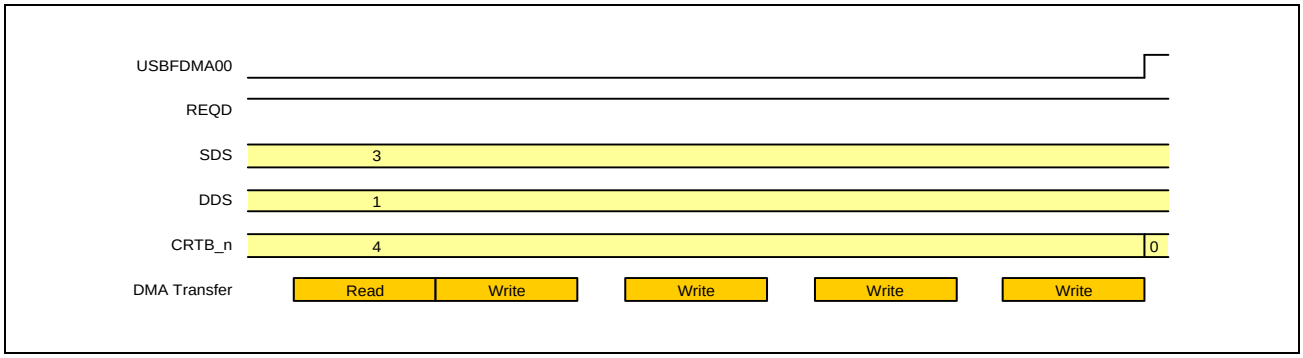


Figure 6.5-31 Example of Timing Chart in the Case Where the Destination Is Smaller
(LVL = 0, HIEN = 1, REQD = 1, and SDS > DDS in the CHCFG_n Register)

(c) If the Source and Destination Transfer Sizes Are the Same

Each time a DMA transfer request is detected, a source read operation and a destination write operation occur.

The following is an example of the timing chart in the case where the source and destination are 8-bit fields (SDS = 0 and DDS = 0 in the CHCFG_n register) (when the rising edge is detected).

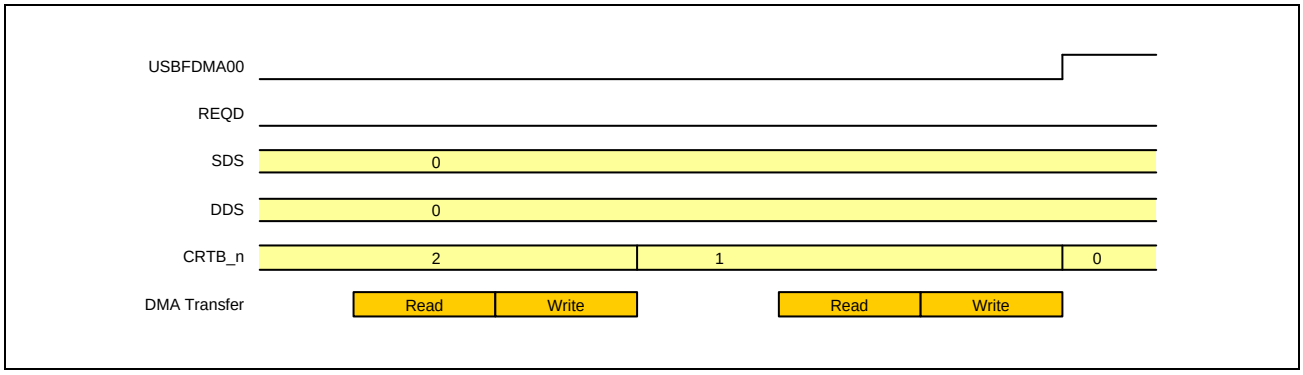


Figure 6.5-32 Example of Timing Chart in the Case where the Source and Destination Sizes are the Same (LVL = 0, HIEN = 1, REQD = 0, and SDS = DDS in the CHCFG_n Register)

(8) Transfer state

The CHSTAT_n register indicates the transfer state of a channel.

(a) Transfer state

The TACT bit in the CHSTAT_n register indicates that channel n is operating. When 1 is written to the SETEN bit in the CHCTRL_n register, the TACT bit is set to 1. The TACT bit continues to be 1 while DMAC is accessing a descriptor or waiting for a DMA request.

The TACT bit is cleared when the EN bit in the CHSTAT_n register is cleared (for details about the conditions in which the EN bit is cleared, see **(1) Current Source Address Register**) and the DMA transfer ends.

If the EN bit is not cleared when the DMA transaction ends (for example, when the REN bit of the CHCFG_n register is 1 in register mode or when DMAC accesses the next descriptor in link mode), the TACT bit is not cleared.

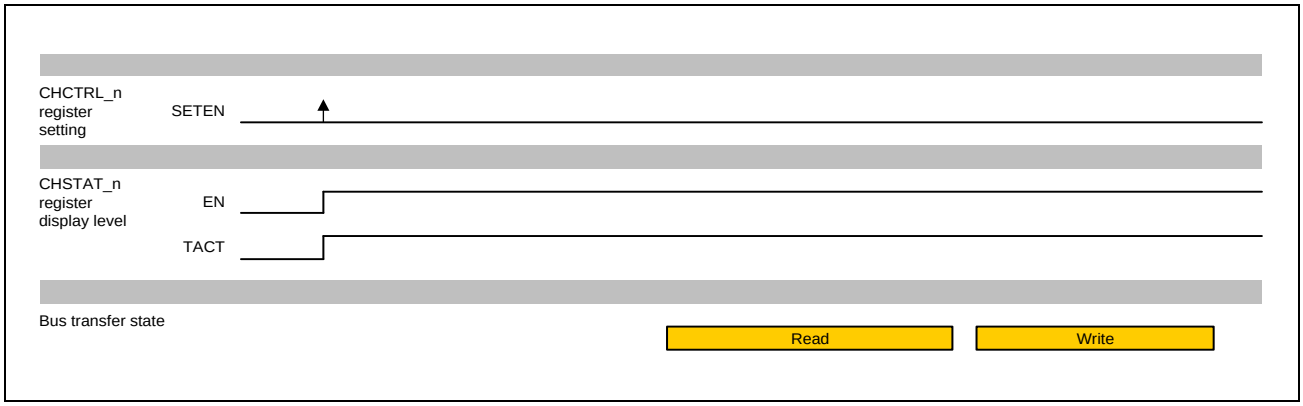


Figure 6.5-33 DMAC State Example 1 (Hardware Request)

(b) Suspension

The SETSUS bit in the CHCTRL_n register can be used to suspend a DMA transfer. If suspension of a DMA transfer is attempted when a bus cycle is running, the DMA transfer is suspended after the bus cycle finishes. The suspended transfer can be resumed by writing 1 to the CLRSUS bit in the CHCTRL_n register.

To check whether a DMA transfer is suspended, after setting the SETSUS bit in the CHCTRL_n register, check the SUS bit in the CHSTAT_n register or the SUS bit for the relevant channel in the DSTAT_SUS register. If The SUS bit is 1, the DMA transfer is currently suspended.

(c) Transfer suspension

By writing 1 to the CLREN bit in the CHCTRL_n register during a DMA transaction, the DMA transaction of the channel can be suspended. As the post-processing for suspension, the SBE bit in the CHCFG_n register can be used to select whether to sweep the data remaining in the buffer when the transaction is suspended. The default is SBE = 0 (do not sweep remaining data).

If the sweep mode is enabled and a transfer is suspended by setting 1 for the CLREN bit in the CHCTRL_n register, DMAC sweeps any data remaining in the buffer and stops operation.

c-1) Transfer suspension (buffer sweeping disabled: SBE = 0)

If 1 is written to the CLREN bit in the CHCTRL_n register during DMA transfer, DMAC suspends DMA transfer and then stops. The timing of stoppage depends on the value set for the REQD bit. After DMAC stops, write 1 to the SWRST bit in the CHCTRL_n register to clear the internal status of DMAC, and then specify the next transfer settings.

Complete deactivation of the channel can be confirmed when the value of the TACT bit in the CHSTAT_n register changes from 1 to 0.

If DMA transfer is suspended before it is completed, the USBFDMAmn interrupt is not asserted.

If the REQD bit in the CHCFG_n register is 0, DMAC stops when the next read operation is completed. However, if data that can be written exists in the buffer, DMAC writes the data and then stops.

If the REQD bit in the CHCFG_n register is 1, DMAC stops when the next read operation is completed.

c-2) Transfer suspension (buffer sweeping enabled: SBE = 1)

If 1 is written to the CLREN bit in the CHCTRL_n register during DMA transfer, DMAC suspends DMA transfer. If the REQD bit in the CHCFG_n register is 0, DMAC sweeps (writes) the already read data, and then stops DMA transfer. If the REQD bit is 1, sweep mode cannot be used physically.

After DMAC stops, set the SWRST bit in the CHCTRL_n register to clear the internal status of DMAC, and then specify the next transfer settings.

Complete deactivation of the channel can be confirmed when the value of the TACT bit in the CHSTAT_n register changes from 1 to 0.

c-3) How to confirm deactivation of the channel

Even when the EN bit of the CHSTAT_n register is cleared to 0 by writing 1 to the CLREN bit in the CHCTRL_n register, if a transfer has already been executed over the bus, DMAC cannot immediately stop. To check whether DMAC has stopped completely, check the EN and TACT bits in the CHSTAT_n register. If both bits are 0, DMAC has stopped completely.

c-4) Procedure for suspending transfer

To suspend transfer:

1. Write 1 to the CLREN bit in the CHCTRL_n register.
2. If the SBE bit in the CHCFG_n register is 0, DMAC stops according to the value of the REQD bit in the CHCFG_n register. If the SBE bit is 1, DMAC is placed in sweep mode.
3. Read the CHSTAT_n register to check whether the TACT bit is 0. If the TACT bit is 0, DMAC has stopped completely. If the TACT bit is 1, continue polling until the bit changes to 0.
4. To perform the next DMA transfer after it is suspended, make sure that the SWRST (software reset) bit in the CHCTRL_n register is turned on before the next DMA transfer starts.

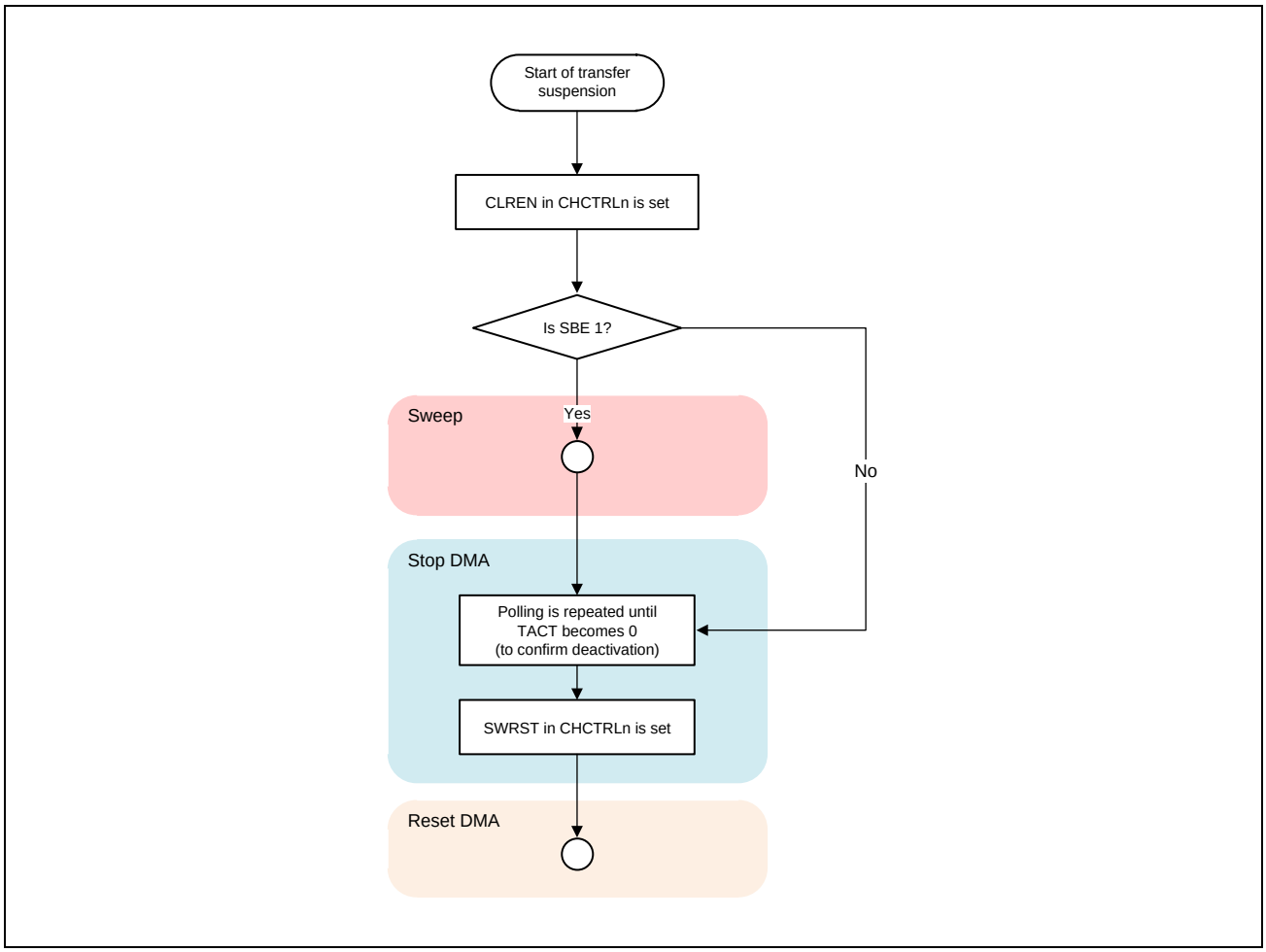


Figure 6.5-34 Transfer Suspension Flowchart

6.5.3.3.14 Access Type

(1) DMA master transfer combination list

(a) Read access

The following describes characteristics of the issuance type for DMA read access.

- An access is performed for a beat align space whose size is set by SDS[2:0] in the CHCFG_n register, including the source address indicated in the CRSA_n register. A beat unaligned transfer for the bus is not performed. An excess area is sometimes read depending on CRSA_n or SKIP settings. In this case, the necessary data is imported into the buffer from the read data.
- The size and burst length are determined based on the value set in the SDS[2:0] field.
  If the value set in the SDS[2:0] field is equal to or less than the bus width
  Size: Value set in SDS[2:0]
  Burst type: SINGLE
  If the value set in the SDS[2:0] field is larger than the bus width
  Size: Bus width
  Burst type: Fixed-length burst (burst length = SDS[2:0] value/bus width)

The following indicates the access types for the bus.

Table 6.5-61 DMA Read Transfer Combination List

SDS	Source Address	AHB transfer					
		First transfer			Second transfer		
		Address	Data Size	Burst Type	Address	Data Size	Burst Type
0 (8 bits)	-	addr	8	SINGLE			
1 (16 bits)	2 byte align	{addr[31:1], 0b}	16	SINGLE			
	2 byte unalign				{addr[31:1], 0b} + 2h	16	SINGLE
2 (32 bits)	4 byte align	{addr[31:2], 00b}	32	SINGLE			
	4 byte unalign				{addr[31:2], 00b} + 4h	32	SINGLE
4 (128 bits)	16 byte align	{addr[31:4], 0h}	32	INCR4			
	16 byte unalign				{addr[31:4], 0h} + 10h	32	INCR4
5 (256 bits)	32 byte align	{addr[31:5], 00h}	32	INCR8			
	32 byte unalign				{addr[31:5], 00h} + 20h	32	INCR8
6 (512 bits)	64 byte align	{addr[31:6], 00h}	32	INCR16			
	64 byte unalign				{addr[31:6], 00h + 40h	32	INCR16

Note: If EBT is detected in the middle of burst, 32-bit INCR burst is used to transfer the remaining data.

(b) Write access

The following describes characteristics of the issuance type for DMA write access.

- An access is performed from the destination address indicated by the CRDA_n register to the beat align boundary whose size is set by DDS[2:0] in the CHCFG_n register.
- The size and burst length are determined based on the value set in the DDS[2:0] field.
  If the value set in the DDS[2:0] field is equal to or less than the bus width
  Size: Value set in DDS[2:0]
  Burst type: SINGLE
  If the value set in the DDS[2:0] field is greater than the bus width
  Size: Bus width
  Burst type: Fixed-length burst (burst length = DDS[2:0] value/bus width)
- In write access, only the specified space is accessed. In the following cases, the combination of values smaller than the value set in the DDS[2:0] field is used for access.
  - The destination address is beat-unaligned for the value set in the DDS[2:0] field.
  - An access specified in the DDS field will be across the SKIP boundary.
  - The size specified in the DDS[2:0] field is too large for the number of remaining bytes to be transferred.

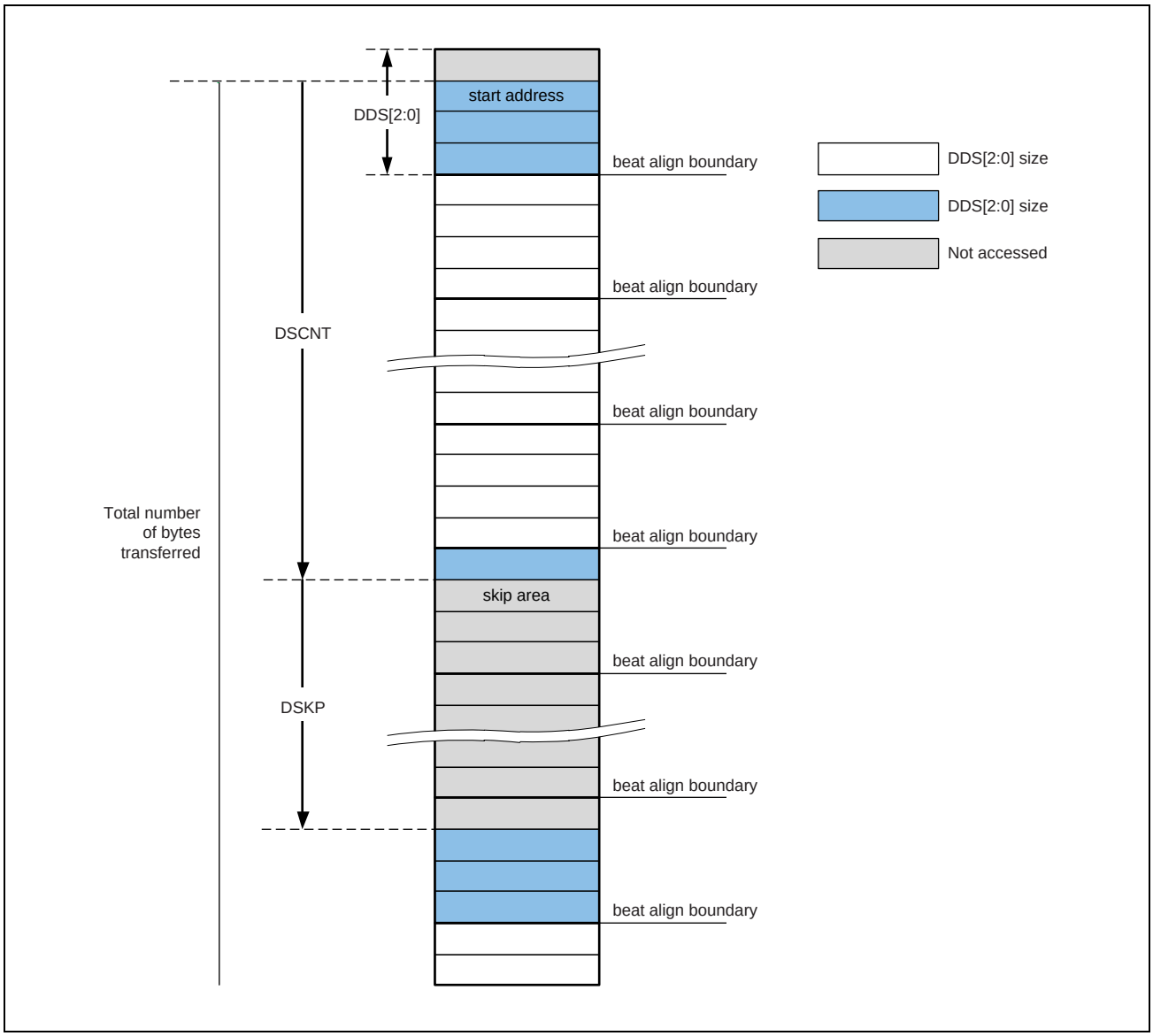


Figure 6.5-35 Example of DMA Write Access Space and Access Types

The following shows the access types for the bus in the case of beat-aligned.

Table 6.5-62 DMA Write Transfer Combination List

	AHB transfer		
	First transfer		
DDS[2:0]	Address	Data Size	Burst Type
0 (8 bits)	addr	8	SINGLE
1 (16 bits)	{addr[31:1], 0b}	16	SINGLE
2 (32 bits)	{addr[31:2], 00b}	32	SINGLE
4 (128 bits)	{addr[31:4], 0h}	32	INCR4
5 (256 bits)	{addr[31:5], 00h}	32	INCR8
6 (512 bits)	{addr[31:6], 00h}	32	INCR16

(2) DMA master descriptor combination list

(a) Read access

The following describes characteristics of a descriptor access.

- Accesses are performed for 8-word beat align space including the LINK address (the address indicated in the CRLA_n register). A beat unaligned transfer is not performed.
- The size and burst length set by INCR8 are used.
- The descriptor format (DSCFM) of the read header is analyzed, and then the descriptor data is set in an internal register.
- If the descriptor extends over the 8-word boundary, an additional read is performed on the succeeding eight words.

The following indicates the types of access to the bus.

Table 6.5-63 Descriptor Read Transfer Combination List

Descriptor Format	Address	AHB Transfer					
		First Transfer			Second Transfer		
	addr[4:0]	Address	Size	Burst	Address	Size	Burst
4 words	00h	{addr[31:4], 0000b}	32	INCR8			
	04h						
	08h						
	0Ch						
	10h						
	14h				{addr[31:4], 0000b} + 20h	32	INCR8
	18h						
	1Ch						
8 words	00h						
	04h				{addr[31:4], 0000b} + 20h	32	INCR8
	08h						
	0Ch						
	10h						
	14h						
	18h						
	1Ch						

Note: If EBT is detected in the middle of burst, 32-bit INCR burst is used to transfer the remaining data.

(b) Write access

Use single transfer as the issuance type for writing data back to a descriptor. The following indicates the types of access to the bus.

Table 6.5-64 Descriptor Write Transfer Combination List

Type	AHB transfer		
	Address	Size	Burst
Write-back in normal mode	{addr[31:2], 00b} + 3h	8	SINGLE
Write-back in the case of an error	{addr[31:2], 00b} + 2h	8	SINGLE

6.5.3.3.15 Arbitration between DMACs

Arbitration between internal DMACs is performed in round-robin mode.

In round-robin mode, the highest priority is given to the DMAC whose DMAC number is the DMC number being used for transfer + 1. Immediately after a reset, DMAC0 has the highest priority.

Table 6.5-65 Priority of a transfer request for DMACs that are performing transfer

	Next DMAC	
Current DMAC	DMAC0	DMAC1
DMAC0	2	1
DMAC1	1	2

Note: Priority: 1 (high), 2 (low)

6.5.3.3.16 Notes

(1) Access

During read access, a beat align area is accessed by one transfer. Therefore, if beat unaligned is set, the beat align area including the specified area is accessed.

For example, if the Source Address is 0000_1038h and SDS is 5 (256-bit), read starts from area 0000_1020h, not from address 0000_1038h. At this time, if the area from 0000_1020h to 0000_1037h contains a register whose value changes by read, the operation might be disrupted.

To prevent problems, use the beat align setting to access a register whose value changes by read access or access to an area adjacent to FIFO.

(2) Level Interrupt bit

This is a DMA interrupt output enable bit. Always set this bit to 1 irrespective of whether peripheral module interrupt USBFDMAmn (m, n = 0, 1) or USBFDMAERRm (m = 0, 1) is used.

To use peripheral module interrupt USBFDMAmn or USBFDMAERRm, set this bit to 1, and then set up the interrupt controller.

(3) Connection Diagram of Resistor to USB20_VUBUSIN

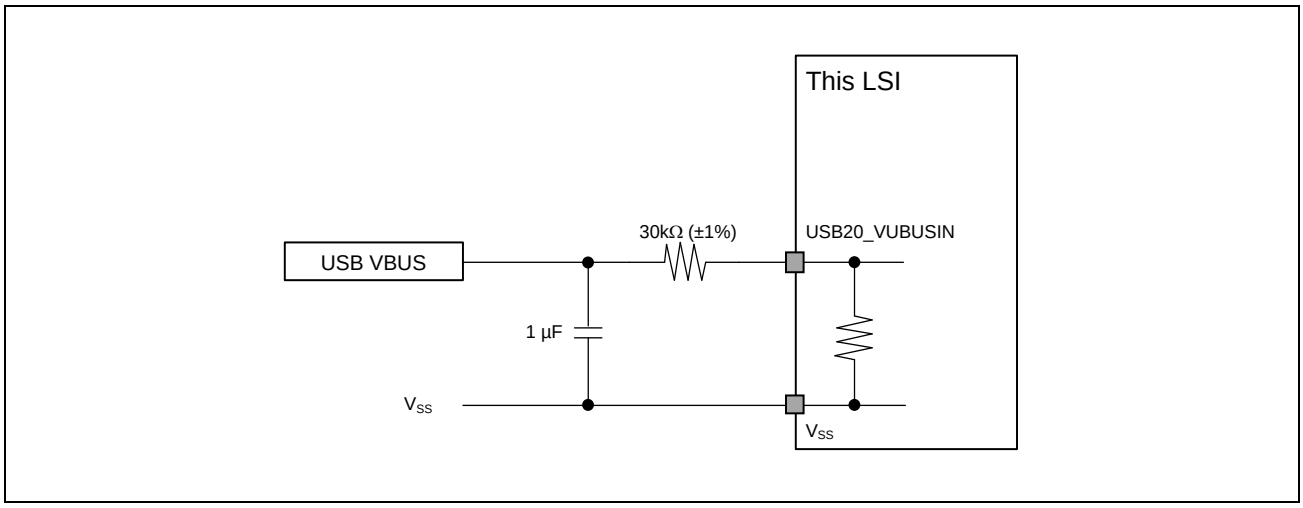


Figure 6.5-36 Connection Diagram of Resistor to USB20_VUBUSIN

SECTION 6 HIGH-SPEED INTERFACE

6.6 PCI Express 3.0 Interface (PCIe)

This section describes the functions of the PCI Express 3.0 interface (PCIe).

6.6.1 Overview

This unit has a dual structure so that the device incorporating it can operate as a root complex or endpoint device, and includes a Type 0/1 Configuration Register for this purpose. Furthermore, it has an internal DMA controller. However, limitations imposed by the AXI place restrictions on some specifications of PCI Express functionality.

■ Unit Function Specifications

AXI Interface Specification

Little endian is only supported.

- Master interface
  - 1 port
  - Bus width: 128 bits
  - ID width: 4 bits (fixed value)
  - Interleave not supported
  - Burst type: Incremental 1 beat (MAxSIZE = 1/2/4 bytes) 2 to 16 beats (MAxSIZE = 16 bytes)
  - Byte-lane transfer and narrow transfer (8, 16, 32-bit lengths, only in 1 dword) are supported. Non-aligned (“unaligned”) transfer is not supported.
  - Allowable number of read/write requests: Variable (1 to 16)
  - Write interleave depth: 1
  - Protection (data/user/non-secure) can be set by using registers.
  - Access type: Normal only (exclusive or locked access not supported)

- Slave interface
  - 1 port
  - Bus width: 128 bits
  - ID width: Up to 20 bits
  - Burst type: Incremental (TYPE = INCR, LENGTH = 1 to 16, SIZE = 1/2/4/8/16 bytes)
  - Support for byte-lane transfer (only when valid bytes are consecutive), unaligned transfer, and narrow transfer (1, 2, and 4 bytes)
  - Number of read transactions which can be accepted: 1 to 8
    Number of write transactions which can be accepted: 1

- Read data reordering depth: 1 to 8
  Write interleave depth: 1
- Required memory area: 16 KB or larger (variable)
- Protection is not distinguished (reception is possible even in the case of protection).
- Access type: Normal only (exclusive or locked access not supported)
- Cache signals (ARCACHE/AWCACHE) are only supported for bufferable bits for writing.

- DMAC
  - DMA method: Register control, descriptor control
  - Number of channels: 8
  - Allowable number of requests to be issued (the maximum total number: 8. When DMAC ch. 1 issues eight PCIe-MRd/MW requests, other channels cannot issue requests).
    - PCIe MRd: Up to 8/ch PCIe MWr: 1/ch
    - AXI Read: 1/ch AXI Write: 1/ch

PCI Express Specification (Compliant with the PCI Express Base Specification 4.0)

- PCI Express Gen1 (2.5 GT/s)/Gen2 (5.0 GT/s)/Gen3 (8.0 GT/s)
- Root Complex (RC) / Endpoint (EP) Applications, Type 0/1 Configuration Register
- Lane implementation x2 / x1 × 1ch
- Data payload: up to 256 bytes; read request size: up to 512 bytes
- Virtual channels are not supported (only VC0 supported)
- Number of outstanding transfers: 1 to 8
- Dynamic control of speed up/down configuration
- Clock Power Management is not supported (P1.CPM, P2.CPM not supported)
- Power Management (ASPM supported not support L1-Substate)
- Error handling/logging (AER supported)
- Replay FIFO with ECC
- Internal Memory without parity
- Number of Support Functions: 2
- Number of DMAC channels: 8
- Supported reference clock architecture
  - Common clock

The initial values of configuration registers for vendor ID, device ID, revision ID, class code, subsystem vendor ID, subsystem ID, and base address register mask are 0. Set appropriate values in the registers before the start of link up.

L1PM substates are not supported.

6.6.2 TOP Pins

6.6.2.1 List of external pins

Table 6.6-1 lists the external pins of the PCI.

Table 6.6-1 List of External Pins

Pin name	Input/Output	Function
PCIE_RXDPL0	Input	PCIe RX data (positive) of Lane 0
PCIE_RXDPL1	Input	PCIe RX data (positive) of Lane 1
PCIE_RXDNL0	Input	PCIe RX data (negative) of Lane 0
PCIE_RXDNL1	Input	PCIe RX data (negative) of Lane 1
PCIE_TXDPL0	Output	PCIe TX data (positive) of Lane 0
PCIE_TXDPL1	Output	PCIe TX data (positive) of Lane 1
PCIE_TXDNL0	Output	PCIe TX data (negative) of Lane 0
PCIE_TXDNL1	Output	PCIe TX data (negative) of Lane 1
PCIE_REFCLKP0	Input	Differential reference clock (positive)
PCIE_REFCLKN0	Input	Differential reference clock (negative)
PCIE0_RSTOUTB	Output	Output the reset signal • When using as RC: Connect to the PERST# pin on Endpoint (opposite device) and use when controlling the initialization of Endpoint from RC • When using as EP: Unused. Open. (High fixed output)

6.6.2.2 List of internal pins

Table 6.6-2 lists the internal pins of this unit.

Table 6.6-2 List of Internal Pins (1/5)

Pin name	Input/Output	Function
FLR_REQ[1:0]	Output	FLR request output
FLR_RESET[1:0]	Input	FLR input
INTX_EP_F0	Input	INTX interrupt signal input for Function #0 for Endpoint (Level signal input: 2 or more cycles of ACLK)
INTX_EP_F1	Input	INTX interrupt signal input for Function #1 for Endpoint (Level signal input: 2 or more cycles of ACLK)
UI_EXTMSI_VAL0	Input	MSI interrupt notification pin for Endpoint
UI_EXTMSI_VAL1	Input	MSI interrupt notification pin for Endpoint
UI_EXTMSI_VAL2	Input	MSI interrupt notification pin for Endpoint
UI_EXTMSI_VAL3	Input	MSI interrupt notification pin for Endpoint
UI_EXTMSI_VAL4	Input	MSI interrupt notification pin for Endpoint
UI_EXTMSI_VEC0 [4:0]	Input	MSI interrupt vector specification pin for Endpoint
UI_EXTMSI_VEC1 [4:0]	Input	MSI interrupt vector specification pin for Endpoint
UI_EXTMSI_VEC2 [4:0]	Input	MSI interrupt vector specification pin for Endpoint
UI_EXTMSI_VEC3 [4:0]	Input	MSI interrupt vector specification pin for Endpoint
UI_EXTMSI_VEC4 [4:0]	Input	MSI interrupt vector specification pin for Endpoint
UI_EXTMSI_FUNC0 [2:0]	Input	MSI interrupt function number specification pin for Endpoint
UI_EXTMSI_FUNC1 [2:0]	Input	MSI interrupt function number specification pin for Endpoint
UI_EXTMSI_FUNC2 [2:0]	Input	MSI interrupt function number specification pin for Endpoint
UI_EXTMSI_FUNC3 [2:0]	Input	MSI interrupt function number specification pin for Endpoint
UI_EXTMSI_FUNC4 [2:0]	Input	MSI interrupt function number specification pin for Endpoint
INTA_RC	Output	INTA interrupt output for Root Complex
INTB_RC	Output	INTB interrupt output for Root Complex
INTC_RC	Output	INTC interrupt output for Root Complex
INTD_RC	Output	INTD interrupt output for Root Complex
INTMSI_RC	Output	MSI Receive interrupt output for Root Complex
INT_LINK_BANDWIDTH	Output	Link Bandwidth change interrupt for Root Complex
INT_LINK_EQUALIZATION_REQUEST	Output	Link Equalization Request interrupt for Root Complex
INT_PM_PME	Output	PM_PME message Receive interrupt for Root Complex
INT_SERR	Output	Error detect interrupt for Root Complex
INT_SERR_COR	Output	Correctable Error detect interrupt for Root Complex
INT_SERR_NONFATAL	Output	Non Fatal Error detect interrupt for Root Complex
INT_SERR_FATAL	Output	Fatal Error detect interrupt for Root Complex
DMA_INT	Output	Event interrupt(DMA)
PCIE_EVT_INT	Output	Event interrupt(PCIe)
AXI_ERR_INT	Output	Event interrupt(AXI)
MSG_INT	Output	Event interrupt(Message Receive)
INT_ALL	Output	Event interrupt
ALLOW_ENTER_L1	Input	ASPM L1 state permission setting input
PME_TIM	Input	Clock input for PM_PME Message for Endpoint
TURN_OFF_EVENT	Output	PME_Turn_Off Msg. Reception flag output for Endpoint

Table 6.6-2 List of Internal Pins (2/5)

Pin name	Input/Output	Function
TURN_OFF_EVENT_ACK	Input	Acknowledge input for Endpoint
D3_EVENT_F0	Output	Non-D0 State transition request reception output for Function #0 for Endpoint
D3_EVENT_F1	Output	Non-D0 State transition request reception output for Function #1 for Endpoint
D3_EVENT_ACK_F0	Input	Acknowledgment input for Function #0 for Endpoint
D3_EVENT_ACK_F1	Input	Acknowledgment input for Function #1 for Endpoint
CFG_PMCSR_PME_STATUS_F0	Input	Power Management Event setting input for Function #0 for Endpoint
CFG_PMCSR_PME_STATUS_F1	Input	Power Management Event setting input for Function #1 for Endpoint
CFG_PMCSR_PME_STATUS_WRITECLEAR_F0	Output	PME_STATUS clear output for Function #0 for Endpoint
CFG_PMCSR_PME_STATUS_WRITECLEAR_F1	Output	PME_STATUS clear output for Function #1 for Endpoint
MODE_PORT	Input	Device Type setting 0b: Endpoint 1b: Root Complex
ACLK	Input	System clock input
CLK_PMU	Input	Power Management clock input
ARESETn	Input	System Reset input
MAWID[3:0]	Output	AXI Master Interface Signals-Write Address Channel Write address ID tag output
MAWADDR[63:0]	Output	AXI Master Interface Signals-Write Address Channel Write address output
MAWLEN[3:0]	Output	AXI Master Interface Signals-Write Address Channel Burst length output
MAWSIZE[2:0]	Output	AXI Master Interface Signals-Write Address Channel Burst size output
MAWBURST[1:0]	Output	AXI Master Interface Signals-Write Address Channel Burst type output (01b: Fixed output of incremental bursts)
MAWCACHE[3:0]	Output	AXI Master Interface Signals-Write Address Channel Cache type output
MAWPROT[2:0]	Output	AXI Master Interface Signals-Write Address Channel Protect type output
MAWVALID	Output	AXI Master Interface Signals-Write Address Channe Valid output
MAWREADY	Output	AXI Master Interface Signals-Write Address Channel Ready output
MWID[3:0]	Output	AXI Master Interface Signals-Write Channel Write Id-tag output
MWDATA[127:0]	Output	AXI Master Interface Signals-Write Channel Write Datta output
MWSTRB[15:0]	Output	AXI Master Interface Signals-Write Channel Write Strobe output
MWLST	Output	AXI Master Interface Signals-Write Channel Last of Write data signal output
MWVALID	Output	AXI Master Interface Signals-Write Channel Write Valid output
MBID[3:0]	Input	AXI Master Interface Signals-Write Response Channel Response ID tag input
MBRESP[1:0]	Input	AXI Master Interface Signals-Write Response Channel Response input

Table 6.6-2 List of Internal Pins (3/5)

Pin name	Input/Output	Function
MBVALID	Input	AXI Master Interface Signals-Write Response Channel Valid input
MBREADY	Output	AXI Master Interface Signals-Write Response Channel Ready output
MARID[3:0]	Output	AXI Master Interface Signals-Read Address Channel ID tag output
MARADDR[63:0]	Output	AXI Master Interface Signals-Read Address Channel Address output
MARLEN[3:0]	Output	AXI Master Interface Signals-Read Address Channel Length output
MARSIZE[2:0]	Output	AXI Master Interface Signals-Read Address Channel Size output
MARBURST[1:0]	Output	AXI Master Interface Signals-Read Address Channel Burst type output
MARLOCK[1:0]	Output	AXI Master Interface Signals-Read Address Channel Lock type output
MARCACHE[3:0]	Output	AXI Master Interface Signals-Read Address Channel Cache type output
MARPROT[2:0]	Output	AXI Master Interface Signals-Read Address Channel Protect output
MARVALID	Output	AXI Master Interface Signals-Read Address Channel Valid output
MARREADY	Input	AXI Master Interface Signals-Read Address Channel READY input
MRID[3:0]	Input	AXI Master Interface Signals-Read Data Channel ID tag input
MRDATA[d-1:0]	Input	AXI Master Interface Signals-Read Data Channel Data input
MRRESP[1:0]	Input	AXI Master Interface Signals-Read Data Channel Response input
MRLAST	Input	AXI Master Interface Signals-Read Data Channel LAST data input
MRVALID	Input	AXI Master Interface Signals-Read Data Channel Valid input
MRREADY	output	AXI Master Interface Signals-Read Data Channel READY output
SAWID[idlen-1:0] (idlen=20)	Input	AXI Slave Interface Signals-Write Address Channel Write address ID tag input
SAWADDR[63:0]	Input	AXI Slave Interface Signals-Write Address Channel Write address input
SAWLEN[3:0]	Input	AXI Slave Interface Signals-Write Address Channel Burst length input
SAWSIZE[2:0]	Input	AXI Slave Interface Signals-Write Address Channel Burst size input
SAWBURST[1:0]	Input	AXI Slave Interface Signals-Write Address Channel Burst type input
SAWVALID	Input	AXI Slave Interface Signals-Write Address Channel Valid signal input 1b: Valid 0b: Invalid

Table 6.6-2 List of Internal Pins (4/5)

Pin name	Input/Output	Function
SAWREADY	Output	AXI Slave Interface Signals-Write Address Channel Ready signal input 1b: The slave is ready. 0b: The slave is being prepared.
SWID[idlen-1:0]	Input	AXI Slave Interface Signals-Write Data Channel Write ID tag input
SWDATA[d-1:0](d=128)	Input	AXI Slave Interface Signals-Write Data Channel Write Data input
SWSTRB[d/8-1:0]	Input	AXI Slave Interface Signals-Write Data Channel Write strobe input
SWLAST	Input	AXI Slave Interface Signals-Write Data Channel Write LAST input
SWVALID	Input	AXI Slave Interface Signals-Write Data Channel Write Valid input
SWREADY	Output	AXI Slave Interface Signals-Write Data Channel Write Ready output
SBID[idlen-1:0]	Output	AXI Slave Interface Signals-Write Response Channel Write Response ID tag output
SBRESP[1:0]	Output	AXI Slave Interface Signals-Write Response Channel Write Response output
SBVALID	Output	AXI Slave Interface Signals-Write Response Channel Write Valid output
SBREADY	Input	AXI Slave Interface Signals-Write Response Channel Write Ready input
SARID[idlen-1:0]	Input	AXI Slave Interface Signals-Read Address Channel Read Address ID tag input
SARADDR[63:0]	input	AXI Slave Interface Signals-Read Address Channel Read Address input
SARLEN[3:0]	Input	AXI Slave Interface Signals-Read Address Channel Read length input
SARSIZE[2:0]	Input	AXI Slave Interface Signals-Read Address Channel Read Size input
SARBURST[1:0]	Input	AXI Slave Interface Signals-Read Address Channel Read Burst input
SARLOCK[1:0]	Input	AXI Slave Interface Signals-Read Address Channel Read Lock type input
SARCACHE[3:0]	Input	AXI Slave Interface Signals-Read Address Channel Read Cache type input
SARPROT[2:0]	Input	AXI Slave Interface Signals-Read Address Channel Read Protection type input
SARVALID	Input	AXI Slave Interface Signals-Read Address Channel Read Valid input
SARREADY	Output	AXI Slave Interface Signals-Read Address Channel Read Ready output
SRID[idlen-1:0]	Output	AXI Slave Interface Signals-Read Data Channel Read Data ID tag output
SRDATA[d-1:0]	Output	AXI Slave Interface Signals-Read Data Channel Read Data output
SRRESP[1:0]	Output	AXI Slave Interface Signals-Read Data Channel Read Response output

Table 6.6-2 List of Internal Pins (5/5)

Pin name	Input/Output	Function
SRLAST	Output	AXI Slave Interface Signals-Read Data Channel Raad Last output
SRVALID	Output	AXI Slave Interface Signals-Read Data Channel Read Valid output
SRREADY	Input	AXI Slave Interface Signals-Read Data Channel Read Ready input

6.6.3 Block Diagram

The figure below is a block diagram of the PCI.

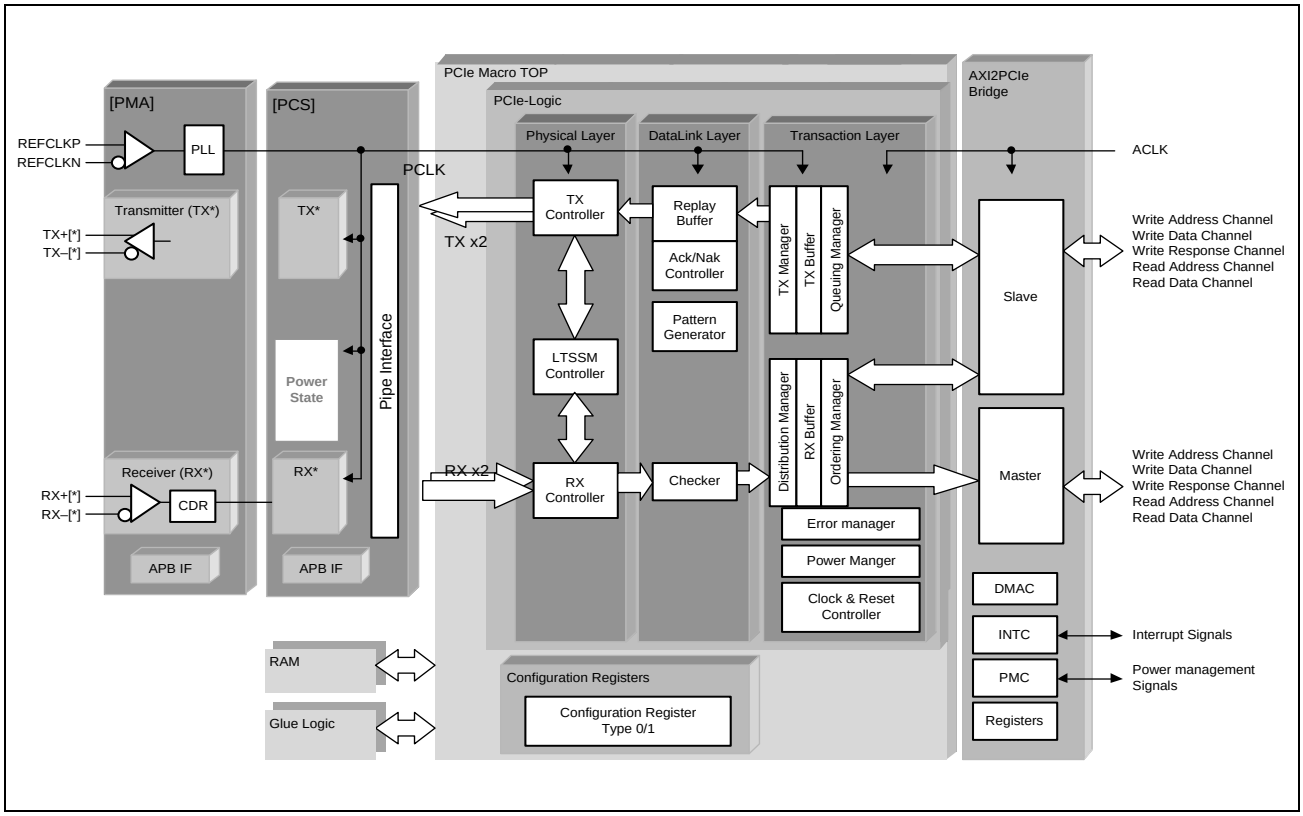


Figure 6.6-1 Block Diagram

6.6.3.1 Reset

A software reset by the register can be used for this unit. **Table 6.6-3** lists the software resets.

Table 6.6-3 List of Software Resets

Reset Name	Reset Control Register
RST_B	Reset Registers (offset:310h) bit 0
RST_GP_B	Reset Registers (offset:310h) bit 1
RST_RSM_B	Reset Registers (offset:310h) bit 2
RST_CFG_B	Reset Registers (offset:310h) bit 3
RST_LOAD_B	Reset Registers (offset:310h) bit 4
RST_PS_B	Reset Registers (offset:310h) bit 5
ARESETn (CPG)	Reset Control Register 3 bit 12

Note: In addition, there is a function level reset (FLR) which is only available in EP operation.

6.6.4 Registers

The register addresses of PCI are given as offsets from the individual base addresses <PCIn_base>. The register base addresses of each PCI are listed in the following table.

Table 6.6-4 Register Base Addresses

Base Address Name	Base Address
<PCI0_base>	0_1340_0000h (5340_0000h*1, 4340_0000h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

6.6.4.1 Register Descriptions (Root Complex Mode)

The following lists the registers incorporated in this unit.

CAUTION

The results of access to reserved bit areas, debug bit areas, and undefined areas are not guaranteed. The combination of reserved, debug, and undefined areas includes cases where the initial values will be non-zero, and the results of changes to such values are not guaranteed.

Registers with the HwInit attribute

Some registers with the RO attribute stated in the PCI Express Base Specification are writable at the time of initialization. When writing to these registers, CFG_HWINIT_EN (Permission Register (offset: 300h) bit [2]) must be set to 1b.

The following is the outline of categories of the register space within this unit.

AXI Bridge Registers	Address 0000h-1FFCh
Physical Layer Control/Status Registers	Address 2000h-5FFCh
PCI Express Configuration Registers (Type1)	Address 6000h-6FFCh
Reserved	Address 7000h-7FFCh

Access to the reserved spaces above is prohibited.

6.6.4.1.1 List of AXI Bridge Registers

The table below lists the AXI bridge registers. Unless specifically stated otherwise, byte, word, and double word access are all possible.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Request Data Register 0	PCI_RC_REQDATA0	xxxx_xxxxh	0080h	32
Request Data Register 1	PCI_RC_REQDATA1	xxxx_xxxxh	0084h	32
Request Data Register 2	PCI_RC_REQDATA2	xxxx_xxxxh	0088h	32
Request Receive Data Register	PCI_RC_REQRCVDAT	xxxx_xxxxh	008Ch	32
Request Address Register 1	PCI_RC_REQADR1	xxxx_xxxxh	0090h	32
Request Address Register 2	PCI_RC_REQADR2	xxxx_xxxxh	0094h	32
Request Byte Enable Register	PCI_RC_REQBE	0000_000Fh	0098h	32
Request Issue Register	PCI_RC_REQISS	0000_0000h	009Ch	32
MSI Receive Window Address (Lower) Register	PCI_RC_MSIRCVWADRL	0000_0000h	0100h	32
MSI Receive Window Address (Higher) Register	PCI_RC_MSIRCVWADRU	0000_0000h	0104h	32
MSI Receive Window Mask (Lower) Register	PCI_RC_MSIRCVWMSKL	0000_0003h	0108h	32
MSI Receive Window Mask (Higher) Register	PCI_RC_MSIRCVWMSKU	0000_0000h	010Ch	32
PCI INTx Receive Interrupt Enable Register	PCI_RC_PINTRCVIE	0000_0000h	0110h	32
PCI INTx Receive Interrupt Status Register	PCI_RC_PINTRCVIS	0000_0000h	0114h	32
Message Receive Interrupt Enable Register	PCI_RC_MSGRCVIE	0000_0000h	0120h	32
Message Receive Interrupt Status Register	PCI_RC_MSGRCVIS	0000_0000h	0124h	32
Message Code Register	PCI_RC_MSGCODE	0000_0000h	0130h	32
Message Data Register	PCI_RC_MSGDATA	0000_0000h	0134h	32
Message Header 3rdDW Register	PCI_RC_MSGH3DW	0000_0000h	0138h	32
Message Header 4thDW Register	PCI_RC_MSGH4DW	0000_0000h	013Ch	32
Interrupt Table Register	PCI_RC_INTTABLE	00xx_0000h	0140h	32
PCIe Event Interrupt Enable 0 Register	PCI_RC_PEIE0	0000_0000h	0200h	32
PCIe Event Interrupt Status 0 Register	PCI_RC_PEIS0	0000_0000h	0204h	32
PCIe Event Interrupt Enable 1 Register	PCI_RC_PEIE1	0000_0000h	0208h	32
PCIe Event Interrupt Status 1 Register	PCI_RC_PEIS1	0000_0000h	020Ch	32
AXI Master Error Interrupt Enable Register	PCI_RC_AMEIE	0000_0000h	0210h	32
AXI Master Error Interrupt Status Register	PCI_RC_AMEIS	0000_0000h	0214h	32
AXI Slave Error Interrupt Enable 1 Register	PCI_RC_ASEIE1	0000_0000h	0220h	32
AXI Slave Error Interrupt Status 1 Register	PCI_RC_ASEIS1	0000_0000h	0224h	32
AXI Slave Error Interrupt Status 3 Register	PCI_RC_ASEIS3	0000_0000h	0230h	32
Permission Register	PCI_RC_PERM	0000_0000h	0300h	32
Reset Register	PCI_RC_RESET	0000_00xxh	0310h	32
Mode Set 0 Register	PCI_RC_MSET0	2001_2000h	0314h	32
Mode Set 1 Register	PCI_RC_MSET1	0000_33F2h	0318h	32
Mode Set 3 Register	PCI_RC_MSET3	0000_0000h	0380h	32
Mode Set 4 Register	PCI_RC_MSET4	0000_0000h	0384h	32
Mode Set 5 Register	PCI_RC_MSET5	0000_0000h	0388h	32
Mode Status 0 Register	PCI_RC_MSTAT0	0000_0000h	0390h	32
PCIe Core Mode Set 1 Register	PCI_RC_PCMSET1	00FA_00F2h	0400h	32
PCIe Core Control 1 Register	PCI_RC_PCCTRL1	0000_0000h	0404h	32
PCIe Core Status 1 Register	PCI_RC_PCSTAT1	0000_0000h	0408h	32
PCIe Core Control 2 Register	PCI_RC_PCCTRL2	003E_0000h	0410h	32
PCIe Core Status 2 Register	PCI_RC_PCSTAT2	xxxx_xxxxh	0414h	32
PCIe Core Status 5 Register	PCI_RC_PCSTAT5	0000_0x00h	042Ch	32
DMA Interrupt Vector 0 Register	PCI_RC_DMAINTVEC0	0000_0000h	04D0h	32
DMA Interrupt Vector 1 Register	PCI_RC_DMAINTVEC1	0000_0000h	04D4h	32
MSI Receive Enable n Register	PCI_RC_MSIRCVEn	0000_0000h	0600h + n x 0010 (n = 0 to 15)	32
MSI Receive Message Data n Register	PCI_RC_MSIRCVMSGDATAn	0000_0000h	0604h + n x 0010 (n = 0 to 15)	32
MSI Receive Mask n Register	PCI_RC_MSIRCVMSKn	FFFF_FFFFh	0608h + n x 0010 (n = 0 to 15)	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
MSI Receive Status n Register	PCI_RC_MSIRCVSTATn	0000_0000h	060Ch + n x 0010 (n = 0 to 15)	32
DMAC Control Register	PCI_RC_DMACTRL	0000_0000h	0800h	32
DMAC Interrupt Enable Register	PCI_RC_DMAINTE	0000_0000h	0808h	32
DMAC Interrupt Status Register	PCI_RC_DMAINTS	0000_0000h	080Ch	32
DMAC Channel Control Register 0	PCI_RC_DMACHCTL0	0000_0000h	0900h	32
Descriptor Start Address (Lower) Register 0	PCI_RC_DPSADRL0	0000_0000h	0908h	32
Descriptor Start Address (Higher) Register 0	PCI_RC_DPSADRU0	0000_0000h	090Ch	32
QUE Entry Register 0	PCI_RC_QUEE0	0000_0000h	0910h	32
DMA Descriptor Control (Descriptor 00h) Register 0	PCI_RC_DMADPCTL0	0000_0000h	0920h	32
DMA Transaction Control (Descriptor 04h) Register 0	PCI_RC_DMATCTL0	0000_0000h	0924h	32
DMA Size (Descriptor 08h) Register 0	PCI_RC_DMASIZE0	0000_0000h	0928h	32
DMA Source Lower Address (Descriptor 10h) Register 0	PCI_RC_DMASLA0	0000_0000h	0930h	32
DMA Source Higher Address (Descriptor 14h) Register 0	PCI_RC_DMASUA0	0000_0000h	0934h	32
DMA Destination Lower Address (Descriptor 18h) Register 0	PCI_RC_DMADESTLA0	0000_0000h	0938h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 0	PCI_RC_DMADESTUA0	0000_0000h	093Ch	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 0	PCI_RC_DMADPLLP0	0000_0000h	0940h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 0	PCI_RC_DMADPULP0	0000_0000h	0944h	32
DMA Rest Size Register 0	PCI_RC_DMARESTSIZE0	0000_0000h	0950h	32
AXI Request Address (Lower) Register 0	PCI_RC_AREQAL0	0000_0000h	0960h	32
AXI Request Address (Higher) Register 0	PCI_RC_AREQAU0	0000_0000h	0964h	32
PCIe Request Address (Lower) Register 0	PCI_RC_PREQAL0	0000_0000h	0968h	32
PCIe Request Address (Higher) Register 0	PCI_RC_PREQAU0	0000_0000h	096Ch	32
QUE Status Register 0	PCI_RC_QUESTA0	0000_0000h	0970h	32
DMAC Error Status Register 0	PCI_RC_DMACESTA0	0000_0000h	0978h	32
DMAC Channel Control Register 1	PCI_RC_DMACHCTL1	0000_0000h	0980h	32
Descriptor Start Address (Lower) Register 1	PCI_RC_DPSADRL1	0000_0000h	0988h	32
Descriptor Start Address (Higher) Register 1	PCI_RC_DPSADRU1	0000_0000h	098Ch	32
QUE Entry Register 1	PCI_RC_QUEE1	0000_0000h	0990h	32
DMA Descriptor Control (Descriptor 00h) Register 1	PCI_RC_DMADPCTL1	0000_0000h	09A0h	32
DMA Transaction Control (Descriptor 04h) Register 1	PCI_RC_DMATCTL1	0000_0000h	09A4h	32
DMA Size (Descriptor 08h) Register 1	PCI_RC_DMASIZE1	0000_0000h	09A8h	32
DMA Source Lower Address (Descriptor 10h) Register 1	PCI_RC_DMASLA1	0000_0000h	09B0h	32
DMA Source Higher Address (Descriptor 14h) Register 1	PCI_RC_DMASUA1	0000_0000h	09B4h	32
DMA Destination Lower Address (Descriptor 18h) Register 1	PCI_RC_DMADESTLA1	0000_0000h	09B8h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 1	PCI_RC_DMADESTUA1	0000_0000h	09BCh	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 1	PCI_RC_DMADPLLP1	0000_0000h	09C0h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 1	PCI_RC_DMADPULP1	0000_0000h	09C4h	32
DMA Rest Size Register 1	PCI_RC_DMARESTSIZE1	0000_0000h	09D0h	32
AXI Request Address (Lower) Register 1	PCI_RC_AREQAL1	0000_0000h	09E0h	32
AXI Request Address (Higher) Register 1	PCI_RC_AREQAU1	0000_0000h	09E4h	32
PCIe Request Address (Lower) Register 1	PCI_RC_PREQAL1	0000_0000h	09E8h	32
PCIe Request Address (Higher) Register 1	PCI_RC_PREQAU1	0000_0000h	09ECh	32
QUE Status Register 1	PCI_RC_QUESTA1	0000_0000h	09F0h	32
DMAC Error Status Register 1	PCI_RC_DMACESTA1	0000_0000h	09F8h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
DMAC Channel Control Register 2	PCI_RC_DMACHCTL2	0000_0000h	0A00h	32
Descriptor Start Address (Lower) Register 2	PCI_RC_DPSADRL2	0000_0000h	0A08h	32
Descriptor Start Address (Higher) Register 2	PCI_RC_DPSADRU2	0000_0000h	0A0Ch	32
QUE Entry Register 2	PCI_RC_QUEE2	0000_0000h	0A10h	32
DMA Descriptor Control (Descriptor 00h) Register 2	PCI_RC_DMADPCTL2	0000_0000h	0A20h	32
DMA Transaction Control (Descriptor 04h) Register 2	PCI_RC_DMATCTL2	0000_0000h	0A24h	32
DMA Size (Descriptor 08h) Register 2	PCI_RC_DMASIZE2	0000_0000h	0A28h	32
DMA Source Lower Address (Descriptor 10h) Register 2	PCI_RC_DMASLA2	0000_0000h	0A30h	32
DMA Source Higher Address (Descriptor 14h) Register 2	PCI_RC_DMASUA2	0000_0000h	0A34h	32
DMA Destination Lower Address (Descriptor 18h) Register 2	PCI_RC_DMADESTLA2	0000_0000h	0A38h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 2	PCI_RC_DMADESTUA2	0000_0000h	0A3Ch	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 2	PCI_RC_DMADPLLP2	0000_0000h	0A40h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 2	PCI_RC_DMADPULP2	0000_0000h	0A44h	32
DMA Rest Size Register 2	PCI_RC_DMARESTSIZE2	0000_0000h	0A50h	32
AXI Request Address (Lower) Register 2	PCI_RC_AREQAL2	0000_0000h	0A60h	32
AXI Request Address (Higher) Register 2	PCI_RC_AREQAU2	0000_0000h	0A64h	32
PCIe Request Address (Lower) Register 2	PCI_RC_PREQAL2	0000_0000h	0A68h	32
PCIe Request Address (Higher) Register 2	PCI_RC_PREQAU2	0000_0000h	0A6Ch	32
QUE Status Register 2	PCI_RC_QUESTA2	0000_0000h	0A70h	32
DMAC Error Status Register 2	PCI_RC_DMACESTA2	0000_0000h	0A78h	32
DMAC Channel Control Register 3	PCI_RC_DMACHCTL3	0000_0000h	0A80h	32
Descriptor Start Address (Lower) Register 3	PCI_RC_DPSADRL3	0000_0000h	0A88h	32
Descriptor Start Address (Higher) Register 3	PCI_RC_DPSADRU3	0000_0000h	0A8Ch	32
QUE Entry Register 3	PCI_RC_QUEE3	0000_0000h	0A90h	32
DMA Descriptor Control (Descriptor 00h) Register 3	PCI_RC_DMADPCTL3	0000_0000h	0AA0h	32
DMA Transaction Control (Descriptor 04h) Register 3	PCI_RC_DMATCTL3	0000_0000h	0AA4h	32
DMA Size (Descriptor 08h) Register 3	PCI_RC_DMASIZE3	0000_0000h	0AA8h	32
DMA Source Lower Address (Descriptor 10h) Register 3	PCI_RC_DMASLA3	0000_0000h	0AB0h	32
DMA Source Higher Address (Descriptor 14h) Register 3	PCI_RC_DMASUA3	0000_0000h	0AB4h	32
DMA Destination Lower Address (Descriptor 18h) Register 3	PCI_RC_DMADESTLA3	0000_0000h	0AB8h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 3	PCI_RC_DMADESTUA3	0000_0000h	0ABCh	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 3	PCI_RC_DMADPLLP3	0000_0000h	0AC0h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 3	PCI_RC_DMADPULP3	0000_0000h	0AC4h	32
DMA Rest Size Register 3	PCI_RC_DMARESTSIZE3	0000_0000h	0AD0h	32
AXI Request Address (Lower) Register 3	PCI_RC_AREQAL3	0000_0000h	0AE0h	32
AXI Request Address (Higher) Register 3	PCI_RC_AREQAU3	0000_0000h	0AE4h	32
PCIe Request Address (Lower) Register 3	PCI_RC_PREQAL3	0000_0000h	0AE8h	32
PCIe Request Address (Higher) Register 3	PCI_RC_PREQAU3	0000_0000h	0AECh	32
QUE Status Register 3	PCI_RC_QUESTA3	0000_0000h	0AF0h	32
DMAC Error Status Register 3	PCI_RC_DMACESTA3	0000_0000h	0AF8h	32
DMAC Channel Control Register 4	PCI_RC_DMACHCTL4	0000_0000h	0B00h	32
Descriptor Start Address (Lower) Register 4	PCI_RC_DPSADRL4	0000_0000h	0B08h	32
Descriptor Start Address (Higher) Register 4	PCI_RC_DPSADRU4	0000_0000h	0B0Ch	32
QUE Entry Register 4	PCI_RC_QUEE4	0000_0000h	0B10h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
DMA Descriptor Control (Descriptor 00h) Register 4	PCI_RC_DMADPCTL4	0000_0000h	0B20h	32
DMA Transaction Control (Descriptor 04h) Register 4	PCI_RC_DMATCTL4	0000_0000h	0B24h	32
DMA Size (Descriptor 08h) Register 4	PCI_RC_DMASIZE4	0000_0000h	0B28h	32
DMA Source Lower Address (Descriptor 10h) Register 4	PCI_RC_DMASLA4	0000_0000h	0B30h	32
DMA Source Higher Address (Descriptor 14h) Register 4	PCI_RC_DMASUA4	0000_0000h	0B34h	32
DMA Destination Lower Address (Descriptor 18h) Register 4	PCI_RC_DMADESTLA4	0000_0000h	0B38h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 4	PCI_RC_DMADESTUA4	0000_0000h	0B3Ch	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 4	PCI_RC_DMADPLLP4	0000_0000h	0B40h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 4	PCI_RC_DMADPULP4	0000_0000h	0B44h	32
DMA Rest Size Register 4	PCI_RC_DMARESTSIZE4	0000_0000h	0B50h	32
AXI Request Address (Lower) Register 4	PCI_RC_AREQAL4	0000_0000h	0B60h	32
AXI Request Address (Higher) Register 4	PCI_RC_AREQAU4	0000_0000h	0B64h	32
PCIe Request Address (Lower) Register 4	PCI_RC_PREQAL4	0000_0000h	0B68h	32
PCIe Request Address (Higher) Register 4	PCI_RC_PREQAU4	0000_0000h	0B6Ch	32
QUE Status Register 4	PCI_RC_QUESTA4	0000_0000h	0B70h	32
DMAC Error Status Register 4	PCI_RC_DMACESTA4	0000_0000h	0B78h	32
DMAC Channel Control Register 5	PCI_RC_DMACHCTL5	0000_0000h	0B80h	32
Descriptor Start Address (Lower) Register 5	PCI_RC_DPSADRL5	0000_0000h	0B88h	32
Descriptor Start Address (Higher) Register 5	PCI_RC_DPSADRU5	0000_0000h	0B8Ch	32
QUE Entry Register 5	PCI_RC_QUEE5	0000_0000h	0B90h	32
DMA Descriptor Control (Descriptor 00h) Register 5	PCI_RC_DMADPCTL5	0000_0000h	0BA0h	32
DMA Transaction Control (Descriptor 04h) Register 5	PCI_RC_DMATCTL5	0000_0000h	0BA4h	32
DMA Size (Descriptor 08h) Register 5	PCI_RC_DMASIZE5	0000_0000h	0BA8h	32
DMA Source Lower Address (Descriptor 10h) Register 5	PCI_RC_DMASLA5	0000_0000h	0BB0h	32
DMA Source Higher Address (Descriptor 14h) Register 5	PCI_RC_DMASUA5	0000_0000h	0BB4h	32
DMA Destination Lower Address (Descriptor 18h) Register 5	PCI_RC_DMADESTLA5	0000_0000h	0BB8h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 5	PCI_RC_DMADESTUA5	0000_0000h	0BBCh	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 5	PCI_RC_DMADPLLP5	0000_0000h	0BC0h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 5	PCI_RC_DMADPULP5	0000_0000h	0BC4h	32
DMA Rest Size Register 5	PCI_RC_DMARESTSIZE5	0000_0000h	0BD0h	32
AXI Request Address (Lower) Register 5	PCI_RC_AREQAL5	0000_0000h	0BE0h	32
AXI Request Address (Higher) Register 5	PCI_RC_AREQAU5	0000_0000h	0BE4h	32
PCIe Request Address (Lower) Register 5	PCI_RC_PREQAL5	0000_0000h	0BE8h	32
PCIe Request Address (Higher) Register 5	PCI_RC_PREQAU5	0000_0000h	0BECh	32
QUE Status Register 5	PCI_RC_QUESTA5	0000_0000h	0BF0h	32
DMAC Error Status Register 5	PCI_RC_DMACESTA5	0000_0000h	0BF8h	32
DMAC Channel Control Register 6	PCI_RC_DMACHCTL6	0000_0000h	0C00h	32
Descriptor Start Address (Lower) Register 6	PCI_RC_DPSADRL6	0000_0000h	0C08h	32
Descriptor Start Address (Higher) Register 6	PCI_RC_DPSADRU6	0000_0000h	0C0Ch	32
QUE Entry Register 6	PCI_RC_QUEE6	0000_0000h	0C10h	32
DMA Descriptor Control (Descriptor 00h) Register 6	PCI_RC_DMADPCTL6	0000_0000h	0C20h	32
DMA Transaction Control (Descriptor 04h) Register 6	PCI_RC_DMATCTL6	0000_0000h	0C24h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
DMA Size (Descriptor 08h) Register 6	PCI_RC_DMASIZE6	0000_0000h	0C28h	32
DMA Source Lower Address (Descriptor 10h) Register 6	PCI_RC_DMASLA6	0000_0000h	0C30h	32
DMA Source Higher Address (Descriptor 14h) Register 6	PCI_RC_DMASUA6	0000_0000h	0C34h	32
DMA Destination Lower Address (Descriptor 18h) Register 6	PCI_RC_DMADESTLA6	0000_0000h	0C38h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 6	PCI_RC_DMADESTUA6	0000_0000h	0C3Ch	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 6	PCI_RC_DMADPLLP6	0000_0000h	0C40h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 6	PCI_RC_DMADPULP6	0000_0000h	0C44h	32
DMA Rest Size Register 6	PCI_RC_DMARESTSIZE6	0000_0000h	0C50h	32
AXI Request Address (Lower) Register 6	PCI_RC_AREQAL6	0000_0000h	0C60h	32
AXI Request Address (Higher) Register 6	PCI_RC_AREQAU6	0000_0000h	0C64h	32
PCIe Request Address (Lower) Register 6	PCI_RC_PREQAL6	0000_0000h	0C68h	32
PCIe Request Address (Higher) Register 6	PCI_RC_PREQAU6	0000_0000h	0C6Ch	32
QUE Status Register 6	PCI_RC_QUESTA6	0000_0000h	0C70h	32
DMAC Error Status Register 6	PCI_RC_DMACESTA6	0000_0000h	0C78h	32
DMAC Channel Control Register 7	PCI_RC_DMACHCTL7	0000_0000h	0C80h	32
Descriptor Start Address (Lower) Register 7	PCI_RC_DPSADRL7	0000_0000h	0C88h	32
Descriptor Start Address (Higher) Register 7	PCI_RC_DPSADRU7	0000_0000h	0C8Ch	32
QUE Entry Register 7	PCI_RC_QUEE7	0000_0000h	0C90h	32
DMA Descriptor Control (Descriptor 00h) Register 7	PCI_RC_DMADPCTL7	0000_0000h	0CA0h	32
DMA Transaction Control (Descriptor 04h) Register 7	PCI_RC_DMATCTL7	0000_0000h	0CA4h	32
DMA Size (Descriptor 08h) Register 7	PCI_RC_DMASIZE7	0000_0000h	0CA8h	32
DMA Source Lower Address (Descriptor 10h) Register 7	PCI_RC_DMASLA7	0000_0000h	0CB0h	32
DMA Source Higher Address (Descriptor 14h) Register 7	PCI_RC_DMASUA7	0000_0000h	0CB4h	32
DMA Destination Lower Address (Descriptor 18h) Register 7	PCI_RC_DMADESTLA7	0000_0000h	0CB8h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 7	PCI_RC_DMADESTUA7	0000_0000h	0CBCh	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 7	PCI_RC_DMADPLLP7	0000_0000h	0CC0h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 7	PCI_RC_DMADPULP7	0000_0000h	0CC4h	32
DMA Rest Size Register 7	PCI_RC_DMARESTSIZE7	0000_0000h	0CD0h	32
AXI Request Address (Lower) Register 7	PCI_RC_AREQAL7	0000_0000h	0CE0h	32
AXI Request Address (Higher) Register 7	PCI_RC_AREQAU7	0000_0000h	0CE4h	32
PCIe Request Address (Lower) Register 7	PCI_RC_PREQAL7	0000_0000h	0CE8h	32
PCIe Request Address (Higher) Register 7	PCI_RC_PREQAU7	0000_0000h	0CECh	32
QUE Status Register 7	PCI_RC_QUESTA7	0000_0000h	0CF0h	32
DMAC Error Status Register 7	PCI_RC_DMACESTA7	0000_0000h	0CF8h	32
AXI Window Base (Lower) Register m	PCI_RC_AWBASELm	0000_0000h	1000h + Offset: 00h/ 20h/ 40h/ … / E0h	32
AXI Window Base (Higher) Register m	PCI_RC_AWBASEUm	0000_0000h	1000h + Offset: 04h/ 24h/ 44h/ … / E4h	32
AXI Window Mask (Lower) Register m	PCI_RC_AWMASKLm	0000_0FFFh	1000h + Offset: 08h/ 28h/ 48h/ … / E8h	32
AXI Window Mask Higher Register m	PCI_RC_AWMASKUm	0000_0000h	1000h + Offset: 0Ch/ 2Ch/ 4Ch/ … / ECh	32
AXI Destination (Lower) Register m	PCI_RC_ADESTLm	0000_0000h	1000 + Offset: 10h/ 30h/ 50h/ 70h/ … / F0h	32
AXI Destination (Higher) Register m	PCI_RC_ADESTUm	0000_0000h	1000 + Offset: 14h/ 34h/ 54h/ 74h/ … / F4h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
PCIe Window Base (Lower) Register m	PCI_RC_PWBASELm	0000_0000h	1000 + Offset: 100h/ 120h/ 140h/ … / 1E0h	32
PCIe Window Base (Higher) Register m	PCI_RC_PWBASEUm	0000_0000h	1000 + Offset: 104h/ 124h/ 144h/ … / 1E4h	32
PCIe Window Mask (Lower) Register m	PCI_RC_PWMASKLm	0000_0FFFh	1000 + Offset: 108h/ 128h/ 148h/ … / 1E8h	32
PCIe Window Mask (Higher) Register m	PCI_RC_PWMASKUm	0000_0000h	1000 + Offset: 10Ch/ 12Ch/ 14Ch/ … / 1ECh	32
PCIe Destination (Lower) Register m	PCI_RC_PDESTLOm	0000_0000h	1000 + Offset: 110h/ 130h/ 150h/ 170h/ … / 1F0h	32
PCIe Destination (Higher) Register m	PCI_RC_PDESTUPm	0000_0000h	1000 + Offset: 114h/ 134h/ 154h/ 174h/ … / 1F4h	32

6.6.4.1.2 List of PCI Express Configuration Registers (Type1)

The table below lists the PCI Express configuration registers.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Vendor and Device ID Register	PCI_RC_VID	0000_0000h	6000h	32
Command and Status Register	PCI_RC_COM_STA	0010_0000h	6004h	32
Revision ID and Class Code Register	PCI_RC_RID_CC	0000_0000h	6008h	32
Cache Line and Header Type Register	PCI_RC_CL_HT	0001_0000h	600Ch	32
Base Address Register 0	PCI_RC_BAR0	0000_0004h	6010h	32
Base Address Register 1	PCI_RC_BAR1	0000_0000h	6014h	32
Bus Number Register	PCI_RC_BNR	0000_0000h	6018h	32
I/O Base/ Limit and Secondary Status	PCI_RC_IOBL_SS	0000_0000h	601Ch	32
Memory Base/ Limit Register	PCI_RC_MEMBL	FFF0_FFF0h	6020h	32
Prefetchable Memory Base/ Limit Register	PCI_RC_PMBL	0001_0001h	6024h	32
Prefetchable Base Higher 32Bits Register	PCI_RC_PBUP32	0000_0000h	6028h	32
Prefetchable Limit Higher 32Bits Register	PCI_RC_PLUP32	0000_0000h	602Ch	32
I/O Base/Limit Higher 16bits Register	PCI_RC_IOBLUP16	0000_0000h	6030h	32
Capabilities Pointer Register	PCI_RC_CP	0000_0040h	6034h	32
Expansion ROM Base Address	PCI_RC_EROMBA	0000_0000h	6038h	32
Bridge Control and Interrupt	PCI_RC_BC_INT	0000_0000h	603Ch	32
PM Capabilities	PCI_RC_PMC	4803_6001h	6040h	32
PM Status/Control Register	PCI_RC_PMSC	0000_0008h	6044h	32
PCI Express Capability Register	PCI_RC_PCIEC	0042_0010h	6060h	32
Device Capabilities Register	PCI_RC_DEVC	0000_8001h	6064h	32
Device Control/Status Register	PCI_RC_DEVCS	0000_2010h	6068h	32
Link Capabilities Register	PCI_RC_LINKC	0072_CC23h	606Ch	32
Link Control/Status Register	PCI_RC_LINKCS	1000_0008h	6070h	32
Slot Capabilities	PCI_RC_SLOTC	0000_0000h	6074h	32
Slot Control/Status	PCI_RC_SLOTCS	0040_0000h	6078h	32
Root Control/Capabilities	PCI_RC_ROOTCC	0000_0000h	607Ch	32
Root Status	PCI_RC_ROOTS	0000_0000h	6080h	32
Device Capabilities 2 Register	PCI_RC_DEVC2	0000_0012h	6084h	32
Device Control 2/Status 2 Register	PCI_RC_DEVCS2	0000_0000h	6088h	32
Link Capabilities 2 Register	PCI_RC_LINKC2	0000_000Eh	608Ch	32
Link Control 2/Status 2 Register	PCI_RC_LINCS2	0000_0003h	6090h	32
Slot Capabilities 2	PCI_RC_SLOTC2	0000_0000h	6094h	32
Slot Control 2 / Status 2	PCI_RC_SLOTCS2	0000_0000h	6098h	32
Base Address Register Mask00 (Lower)	PCI_RC_BARMSK00L	1FFF_FFFFh	60A0h	32
Base Address Register Mask00 (Higher)	PCI_RC_BARMSK00U	0000_0000h	60A4h	32
Base Size 00/01 Register	PCI_RC_BSIZE00_01	0000_0000h	60C8h	32
Type Supported 00/01/02 Register	PCI_RC_TSUPPORT00_01_02	0033_3333h	60D8h	32
Advanced Error Reporting Capability Register	PCI_RC_ADVERC	1501_0001h	6100h	32
Uncorrectable Error Status Register	PCI_RC_UNCESTS	0000_0000h	6104h	32
Uncorrectable Error Mask Register	PCI_RC_UNCEMASK	0000_0000h	6108h	32
Uncorrectable Error Severity Register	PCI_RC_UNCESVY	0046_2030h	610Ch	32
Correctable Error Status Register	PCI_RC_CESTS	0000_0000h	6110h	32
Correctable Error Mask Register	PCI_RC_CEMASK	0000_2000h	6114h	32
Advanced Error Capabilities and Control Register	PCI_RC_ADVECC	0000_00A0h	6118h	32
Header Log Register 0	PCI_RC_HLOG0	0000_0000h	611Ch	32
Header Log Register 1	PCI_RC_HLOG1	0000_0000h	6120h	32
Header Log Register 2	PCI_RC_HLOG2	0000_0000h	6124h	32
Header Log Register 3	PCI_RC_HLOG3	0000_0000h	6128h	32
Root Error Command	PCI_RC_ROOTEC	0000_0000h	612Ch	32
Root Error Status	PCI_RC_ROOTES	0000_0000h	6130h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Error source Identification Register	PCI_RC_ERRSI	0000_0000h	6134h	32
Device Serial Number Extended Capability Register	PCI_RC_DEVSNEXTC	1B01_0003h	6150h	32
Serial Number Register (Lower DW)	PCI_RC_SNL	0000_0000h	6154h	32
Serial Number Register (Higher DW)	PCI_RC_SNU	0000_0000h	6158h	32
Secondary PCI Express Extended Capability Header	PCI_RC_SPEECH	0001_0019h	61B0h	32
Link Control 3 Register	PCI_RC_LINC3	0000_0000h	61B4h	32
Lane Error Status Register	PCI_RC_LESTA	0000_0000h	61B8h	32
Lane Equalization Control Register	PCI_RC_LEQCTL	7F7F_7F7Fh	61BCh	32

6.6.4.1.3 AXI Bridge Register Descriptions

The function description of each register is given below.

(1) Request Data Register m (PCI_RC_REQDATAm) (m = 0 to 2)

This register issues various requests.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0080h/ 0084h/ 0088h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Request Data[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Request Data[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Request Data[31:0]	xh	RW	Issue various requests to PCIe

Table 6.6-5 Details of Each Requests

	Request Data Register 0	Request Data Register 1	Request Data Register 2
Offset Address	0080h	0084h	0088h
Zero-Length Read Request	Invalid	Invalid	Invalid
Config Write	Invalid	Invalid	Write data
Config Read	Invalid	Invalid	Invalid
Message Request	3rd Header	4th Header	Invalid
Message Request with data payload	3rd Header	4th Header	Message data

Note: The bits should be set to 0 for the requests indicated as “Invalid”.

(2) Request Receive Data Register (PCI_RC_REQRCVDAT)

This register indicates the data read on reception of the completion response after issuing a read request.

Access Size : 32 bits
Offset Address : <PCI0_base> + 008Ch
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Request Receive Data[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Request Receive Data[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Request Receive Data[31:0]	xh	R	After issuing a read request, the data read on reception of the completion response are set in these bits. However, these bits are invalid for Zero-Length Read and any kind of various write requests.

Note: x = Undefined

(3) Request Address Register 1 (PCI_RC_REQADR1)

This register issues requests to PCIe.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0090h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Request Address[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Request Address[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Request Address[31:0]	xh	RW	Set the Address, etc. when issuing a Request.

Table 6.6-6 Details of Each Requests

	[31:27]	[26:24]	[23:19]	[18:16]
Zero-Length Read Request	Address			
Config Write	Bus number		Device number	Function number
Config Read	Bus number		Device number	Function number
Message Request	Reserved	Routing type	Reserved	Reserved
Message Request with data payload	Reserved	Routing type	Reserved	Reserved

	[15:12]	[11:8]	[7:2]	[1:0]
Zero-Length Read Request	Address			Reserved
Config Write	Reserved	Ext. Reg. number	Register number	Reserved
Config Read	Reserved	Ext. Reg. number	Register number	Reserved
Message Request	Reserved	Reserved	Message code	
Message Request with data payload	Reserved	Reserved	Message code	

Note: The bits should be set to 0 for the requests indicated as “Reserved”.

(4) Request Address Register 2 (PCI_RC_REQADR2)

This register issues requests to PCIe.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0094h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Request Address63:32[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Request Address63:32[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Request Address63:32 [31:0]	xh	RW	Set the Address, etc. when issuing a Request.

Table 6.6-7 Details of Each Requests

	[31:0]
Zero-Length Read Request	Address
Config Write	Invalid
Config Read	Invalid
Message Request	Invalid
Message Request with data payload	Invalid

Note: The bits should be set to 0 for the requests indicated as “Invalid”.

(5) Request Byte Enable Register (PCI_RC_REQBE)

This register specifies the 1st DW byte enable bit of the TLP header when issuing a request to PCIe.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0098h
Initial Value : 0000_000Fh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	Request Byte Enable[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	Request Byte Enable[3:0]	Fh	RW	Specify Byte Enable when issuing Cfg Request as required. Normally use 1111b. 1b: Byte Enable enabled 0b: Byte Enable disabled

Table 6.6-8 Details of Each Requests

	[3:0]
Zero-Length Read Request	0000b
Config Write	Random (Usually 1111b)
Config Read	Random (Usually 1111b)
Message Request	Invalid (1111b)
Message Request with data payload	Invalid (1111b)

(6) Request Issue Register (PCI_RC_REQISS)

This register issues requests to PCIe.

Access Size : 32 bits
Offset Address : <PCI0_base> + 009Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	Request Rejection	MOR CD PERR	MOR CH PERR	MOR EP ERR	MOR STATUS[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	FUNC[2:0]			TR TYPE[3:0]				-	-	-	-	-	-	-	Request Issue
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	Request Rejection	0h	R	Indicates that the stopped or hibernated state of the PCI Express transmit (TX) side was detected and the process was forcibly terminated. 0b: Normal state (Request issued) 1b: Rejection
21	MOR CD PERR	0h	R	It is set to 1b when a data error occurs in the Completion TLP for the Non-Posted request issued by this register. Normally not used. It is not updated at the time of Posted request.
20	MOR CH PERR	0h	R	It is set to 1b when a header error occurs in the Completion TLP for the Non-Posted request issued by this register. Normally not used. It is not updated at the time of Posted request.
19	MOR EP ERR	0h	R	It is set to 1b when a Poisoned Completion TLP for a Non-Posted request issued by this register is received. Normally not used. It is not updated at the time of Posted request.
18 to 16	MOR STATUS[2:0]	0h	R	This register retains the MOR Status of Completion TLP for Non-Posted requests issued. It is not updated at the time of Posted request. 000b: Successful Completion (SC) 001b: Unsupported Request (UR) 010b: Configuration Request Retry Status (CRS) (not supported) 011b: Completion Timeout 100b: Completer Abort (CA) 101b: Unexpected Completion and mismatched type (Lock Completion responds to non-Lock Request) 110b: Reserved 111b: Overrun Completion length
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 12	FUNC[2:0]	0h	RW	Set the function of Request.
11 to 8	TR TYPE[3:0]	0h	RW	Sets the type of Request. Refer to **Table 6.6-9** for details.
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	Request Issue	0h	RW	When writing: 1b: Request issuance 0b: No operation When reading: 1b: Processing Request (Indicates that the issued Request is being processed.) 0b: Request can be accepted (Indicates that the processing of the issued Request has ended.)

Table 6.6-9 Details of Each Requests

	TR Type		Device Type	
	[11:8]	Posted/Non-Posted	Root Complex	Endpoint
Zero-Length Read Request	0000b (0h)	Non-posted	Issuable	Issuable
Configuration Read Type0	0100b (4h)	Non-posted	Issuable	Issuing prohibited
Configuration Write Type0	0101b (5h)	Non-posted	Issuable	Issuing prohibited
Configuration Read Type1	0110b (6h)	Non-posted	Issuable	Issuing prohibited
Configuration Write Type1	0111b (7h)	Non-posted	Issuable	Issuing prohibited
Message Request	1000b (8h)	Posted	Issuable	Issuable
Message Request with data payload	1001b (9h)	Posted	Issuable	Issuable
	Others	—	Issuing prohibited	Issuing prohibited

(7) MSI Receive Window Address (Lower) Register (PCI_RC_MSIRCVWADRL)

This register sets the start address of the MSI memory space for reception.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0100h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSI Receive Window Address (Lower)[28:13]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSI Receive Window Address (Lower)[12:0]													-	Message Data Enable	MSI Receive Window Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	MSI Receive Window Address (Lower)[28:0]	0h	RW	Set Start Address[31:3] of MSI receive window. However, it must be aligned with the size set by MSI Receive Window Mask. Even if you set a non-aligned Address, the Address Bit for which the MSI Receive Window Mask is set will be 0. *Make sure that MSI Receive Window Enable is 0b when changing this register.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	Message Data Enable	0h	RW	Control whether to use Message Data in MSI reception judgment. 0b: Not used (conventional mode) 1b: Used (multiple interrupts) The MSI reception judgment result selected by this register is reflected in the INTMSI_RC pin.
0	MSI Receive Window Enable	0h	RW	Enable setting of MSI Receive Window. 0b: Window is disabled. 1b: Window is enabled.

(8) MSI Receive Window Address (Higher) Register (PCI_RC_MSIRCVWADRU)

This register sets the start address of the MSI memory space to receive.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0104h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSI Receive Window Address (Higher)[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSI Receive Window Address (Higher)[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MSI Receive Window Address (Higher)[31:0]	0h	RW	Set the Start Address[63:32] of the MSI receive window. However, it must be aligned with the size set by MSI Receive Window Mask. Even if you set a non-aligned Address, the Address Bit for which the MSI Receive Window Mask is set will be 0b. *Make sure that MSI Receive Window Enable is 0b when changing this register.

(9) MSI Receive Window Mask (Lower) Register (PCI_RC_MSIRCVWMSKL)

This register specifies the size of the area from the address set in the MSI Receive Window Address bits.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0108h
Initial Value : 0000_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSI Receive Window Mask (Lower)[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSI Receive Window Mask (Lower)[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	MSI Receive Window Mask (Lower)[31:2]	0h	RW	Set the lower bits of the reserved area to 1b (min. 4-byte space; max. 4-Gbyte space). Example settings are as follows. 0h: 4-byte space 1h: 8-byte space 7fh: 512-byte space Note: To change this register, set the MSI Receive Window Enable bit to 0b.
1, 0	MSI Receive Window Mask (Lower)[1:0]	3h	R	11b fixed. MSI Receive Window Mask [1:0] should always be set to Mask.

(10) MSI Receive Window Mask (Higher) Register (PCI_RC_MSIRCVWMSKU)

This register specifies the size of the area from the address set in the MSI Receive Window Address bits.

Access Size : 32 bits
Offset Address : <PCI0_base> + 010Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	MSI Receive Window Mask (Higher)[30:16]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSI Receive Window Mask (Higher)[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 0	MSI Receive Window Mask (Higher)[30:0]	0h	RW	Set [62:32] for MSI Receive Window Mask.

(11) PCI INTx Receive Interrupt Enable Register (PCI_RC_PINTRCVIE)

This register enables INTx_RC interrupts.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0110h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	MSI Receive Interrupt Enable	INTD Receive Interrupt Enable	INTC Receive Interrupt Enable	INTB Receive Interrupt Enable	INTA Receive Interrupt Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	MSI Receive Interrupt Enable	0h	RW	Enable INTMSI_RC with MSI reception. 0b: disable 1b: enable
3	INTD Receive Interrupt Enable	0h	RW	Enable INTD_RC. 0b: disable 1b: enable
2	INTC Receive Interrupt Enable	0h	RW	Enable INTC_RC. 0b: disable 1b: enable
1	INTB Receive Interrupt Enable	0h	RW	Enable INTB_RC. 0b: disable 1b: enable
0	INTA Receive Interrupt Enable	0h	RW	Enable INTA_RC. 0b: disable 1b: enable

(12) PCI INTx Receive Interrupt Status Register (PCI_RC_PINTRCVIS)

This register indicates the INTx_RC interrupt factor. When Assert_INTx is received in response to a message request from PCIe, the corresponding bit in this register is set and interrupt INTx_RC is asserted. When Deassert_INTx is received in response to a message request, the corresponding bit in this register is cleared and interrupt INTx_RC is de-asserted. Although this register can be cleared by writing 1b by software, it is not recommended to clear this bit by the software itself during normal operation for interrupts on the bus for PCIe. The fields of this register are set if each factor is detected, regardless of the setting of the PCI_INTx_Receive_Interrupt_Enable register for RC.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0114h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	MSI Receive Interrupt Status	INTD Receive Interrupt Status	INTC Receive Interrupt Status	INTB Receive Interrupt Status	INTA Receive Interrupt Status
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW1	RW1	RW1	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	MSI Receive Interrupt Status	0h	RW1	MSI Receive Window Set when Memory Write Request is received from the PCI bus side in the set area. This bit is set when MSI is issued to AXI as a Write Transaction and the Response is returned.
3	INTD Receive Interrupt Status	0h	RW1	It is set when an Assert INTD Message is received, and cleared when a Deassert INTD Message is received.
2	INTC Receive Interrupt Status	0h	RW1	It is set when an Assert INTC Message is received, and cleared when a Deassert INTC Message is received.
1	INTB Receive Interrupt Status	0h	RW1	It is set when an Assert INTB Message is received, and cleared when a Deassert INTB Message is received.
0	INTA Receive Interrupt Status	0h	RW1	It is set when an Assert INTA Message is received, and cleared when a Deassert INTA Message is received.

(13) Message Receive Interrupt Enable Register (PCI_RC_MSGRCVIE)

This register controls enabling of MSG_INT in response to the reception of message requests other than INTx and error-related messages.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0120h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	Message Receive Interrupt Enable	-	-	-	-	PM_Active_State_Nak Receive Interrupt Enable	PM_PME Receive Interrupt Enable	PME_Turn_Off Receive Interrupt Enable	PME_TO_Ack Receive Interrupt Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	Message Receive Interrupt Enable	0h	RW	Enable control of MSG_INT assertion by message reception 0b: Disable 1b: Enable
23 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	PM_Active_State_Nak Receive Interrupt Enable	0h	RW	Enable control of MSG_INT assertion by PM_Active_State_Nak reception 0b: Disable 1b: Enable
18	PM_PME Receive Interrupt Enable	0h	RW	Enable control of MSG_INT assertion by PM_PME reception 0b: Disable 1b: Enable
17	PME_Turn_Off Receive Interrupt Enable	0h	RW	Enable control of MSG_INT assertion by PME_Turn_Off reception 0b: Disable 1b: Enable
16	PME_TO_Ack Receive Interrupt Enable	0h	RW	Enable control of MSG_INT assertion by PME_TO_Ack reception 0b: Disable 1b: Enable
15 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(14) Message Receive Interrupt Status Register (PCI_RC_MSGRCVIS)

This register is a status register that indicates the reception of message requests other than INTx and error-related messages. The value of this register is reflected in MSG_INT. Only the message code is used to judge the message type. The validity of routing and validity of Msg/MsgD selection are not verified. The corresponding message is considered to have been received.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0124h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	Message Receive Interrupt	-	-	-	-	PM_Active_State_Nak Receive Interrupt	PM_PME Receive Interrupt	PME_Turn_Off Receive Interrupt	PME_TO_Ack Receive Interrupt
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW1	R	R	R	R	RW1	RW1	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	Message Receive Interrupt	0h	RW1	Set when receiving a message (does not depend on the type of message)
23 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	PM_Active_State_Nak Receive Interrupt	0h	RW1	Disabled due to RC
18	PM_PME Receive Interrupt	0h	RW1	Set when receiving a PM_PME message
17	PME_Turn_Off Receive Interrupt	0h	RW1	Disabled due to RC
16	PME_TO_Ack Receive Interrupt	0h	RW1	Set when receiving a PM_TO_Ack message
15 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(15) Message Code Register (PCI_RC_MSGCODE)

This register stores the code and routing of the last received message.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0130h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Code[7:0]								Routing[2:0]			-	-	-	-	Message Payload
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 8	Message Code[7:0]	0h	R	Stores the Code of the last received Message.
7 to 5	Routing[2:0]	0h	R	Stores the Routing of the last received Message.
4 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	Message Payload	0h	R	Stores the presence or absence of the data payload of the last received Message. 1b: MsgD (with Payload) 0b: Msg (Without Payload) When Power Management Message (PME_TO_Ack Message , PME_Turn_Off Message , PM_PME Message , PM_Active_State_Nak Message) is received, this register is not written.

(16) Message Data Register (PCI_RC_MSGDATA)

This register stores the data of the last received message.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0134h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Message Data[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Data[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Message Data[31:0]	0h	R	Stores the 1DW data of the last received Message. It is updated only when MsgD(with Data) is received, and the previous value is retained when Msg(without Data) is received. When Power Management Message (PME_TO_Ack Message , PME_Turn_Off Message , PM_PME Message , PM_Active_State_Nak Message) is received, this register is not written.

(17) Message Header 3rdDW Register (PCI_RC_MSGH3DW)

This register stores the header (the 3rd DW) of the last received message.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0138h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Message Header 3rdDW[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Header 3rdDW[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Message Header 3rdDW[31:0]	0h	R	Stores the Header (3rdDW) of the last received Message.

Note: When a Power Management message (PME_TO_Ack Message, PME_Turn_Off Message, PM_PME Message, PM_Active_State_Nak Message) is received, this register is not written.

(18) Message Header 4thDW Register (PCI_RC_MSGH4DW)

This register stores the header (the 4th DW) of the last received message.

Access Size : 32 bits
Offset Address : <PCI0_base> + 013Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Message Header 4thDW Registers[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Header 4thDW Registers[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Message Header 4thDW Registers[31:0]	0h	R	Stores the Header (4thDW) of the last received Message.

Note: When a Power Management message (PME_TO_Ack Message, PME_Turn_Off Message, PM_PME Message, PM_Active_State_Nak Message) is received, this register is not written.

(19) Interrupt Table Register (PCI_RC_INTTABLE)

This register is an index of interrupt factors. The interrupt signal (active high) status of each category can be monitored in a list.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0140h
Initial Value : 00xx_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INT_LINK_EQUALIZATION_REQUEST	-	-	-	-	AXI_ERR_INT	PCIE_EVT_INT	MSG_INT	-	-	-	INTMSI_RC	INTD_RC	INTC_RC	INTB_RC	INTA_RC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	-	xh	R	Reserved Whenever it is read, xb is read. The written value will be ignored.
15	INT_LINK_EQUALIZATION_REQUEST	0h	R	LINK_EQUALIZATION_REQUEST Interrupt signal level monitor
14 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	AXI_ERR_INT	0h	R	Error interrupt signal monitor
9	PCIE_EVT_INT	0h	R	Event interrupt signal monitor
8	MSG_INT	0h	R	Message interrupt signal monitor
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	INTMSI_RC	0h	R	INT_MSI interrupt signal monitor
3	INTD_RC	0h	R	INTD_RC interrupt signal monitor
2	INTC_RC	0h	R	INTC_RC interrupt signal monitor
1	INTB_RC	0h	R	INTB_RC interrupt signal monitor
0	INTA_RC	0h	R	INTA_RC interrupt signal monitor

Note: x = Undefined

(20) PCIe Event Interrupt Enable 0 Register (PCI_RC_PEIE0)

This register enables interrupts of the various PCI Express event factors. This register enables writing to the PCIe Event Interrupt Status 0 Register. See the description of this status register for details of the factors.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0200h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	UI_LINK_WIDTH_CHANGE_DONE EN	UI_LINK_SPEED_CHANGE_DONE EN	Request Done EN	-	-	-	CA EN	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	R	R	RW	RW	RW	RW	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	RX_DLLP_PM_ENTER_L23 EN	-	ASPM L1 Rejected EN	DL_UpDown EN	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	R	RW	RW	RW	RW	R	RW	RW	RW	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30	UI_LINK_WIDTH_CHANGE_DONE EN	0h	RW	Up/Down Configure operation complete interrupt enable 0b: Disable 1b: Enable
29	UI_LINK_SPEED_CHANGE_DONE EN	0h	RW	Speed change operation completion interrupt enable 0b: Disable 1b: Enable
28	Request Done EN	0h	RW	Request complete interrupt enable 0b: Disable 1b: Enable
27, 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
24	CA EN	0h	RW	CA (Completer Abort) interrupt enable 0b: Disable 1b: Enable
23, 22	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
21	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20 to 14	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
13	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12	RX_DLLP_PM_ENTER_L23 EN	0h	RW	RX_DLLP_PM_ENTER_L23 interrupt enable 0b: Disable 1b: Enable
11	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
10	ASPM L1 Rejected EN	0h	RW	ASPM L1 Rejected interrupt enable 0b: Disable 1b: Enable
9	DL_UpDown EN	0h	RW	Interrupt enable on DL state change 0b: Disable 1b: Enable

Bit	Bit Name	Initial Value	R/W	Description
8	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 4	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.

(21) PCIe Event Interrupt Status 0 Register (PCI_RC_PEIS0)

This register is a status register that indicates each PCI Express event. Set to 1b by the factor in the table. After checking the factor, write 1b to clear it.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0204h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	UI_LINK_WIDTH_CHANGE_DONE	UI_LINK_SPEED_CHANGE_DONE	Request Done	-	-	-	CA	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW1	RW1	RW1	R	R	RW	RW1	RW	RW	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	RX_DLLP_PM_ENTER_L23	-	ASPM L1 Rejected	DL_Up Down	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	R	RW1	RW	RW1	RW1	R	RW	RW	RW	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30	UI_LINK_WIDTH_CHANGE_DONE	0h	RW1	Indicates completion of Up/Down Configure operation.
29	UI_LINK_SPEED_CHANGE_DONE	0h	RW1	Indicates completion of Speed Change operation.
28	Request Done	0h	RW1	For requests submitted in the Request Issue Registers (Offset: 9Ch): Non-Posted: Indicates that Completion has been received. Posted: Indicates that the request submission has completed..
27, 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
24	CA	0h	RW1	Indicates that the device has responded with CA (Completer Abort).
23, 22	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
21	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20 to 14	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
13	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12	RX_DLLP_PM_ENTER_L23	0h	RW1	Indicates transition to L2/L3 State in Power Management control.
11	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
10	ASPM L1 Rejected	0h	RW1	Indicates rejected ASPM L1 transition
9	DL_UpDown	0h	RW1	Set to 1b on transition from DL_Down state to DL_Up state, or DL_Up state to DL_Down state. Check the DL_Down/DL_Up status with PCIe Core Status 1 Registers (Offset: 408h).
8	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.

(22) PCIe Event Interrupt Enable 1 Register (PCI_RC_PEIE1)

This register enables parity error and ECC error interrupts. When each bit is set to the valid setting, the value of each corresponding status bit of the PCIe Event Interrupt Status 1 Register becomes valid.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0208h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TXB_PARITY_ERR EN	ERR_RPC_REPLAYFIFO_PERR EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	ERR_REPLAY_HIGHER_CORRECTABLE_ERROR EN	ERR_REPLAY_LOWER_CORRECTABLE_ERROR EN	-	-	-	-	-	-	ERR_REPLAY_HIGHER_UNCORRECTABLE_ERR OR EN	ERR_REPLAY_LOWER_UNCORRECTABLE_ERRO R EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	TXB_PARITY_ERR EN	0h	RW	Enable the TXB_PARITY_ERR interrupt. Parity error interrupt notification enable setting for TX Buffer installed in Transaction Layer 0b: Disable 1b: Enable
16	ERR_RPC_REPLAYFIFO_PERR EN	0h	RW	Enable ERR_RPC_REPLAYFIFO_PERR interrupts. Parity error interrupt notification enable setting for Replay FIFO installed in Data Link Layer 0b: Disable 1b: Enable
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	ERR_REPLAY_HIGHER_CORRECTABLE_ERROR EN	0h	RW	Enable ERR_REPLAY_HIGHER_CORRECTABLE_ERROR interrupts. Interrupt notification enable setting when an ECC 1-bit error (Correctable Error) occurs in the upper 64-bit data bus of the Replay FIFO installed in the Data Link Layer 0b: Disable 1b: Enable
8	ERR_REPLAY_LOWER_CORRECTABLE_ERROR EN	0h	RW	Enable ERR_REPLAY_LOWER_CORRECTABLE_ERROR interrupts. Interrupt notification enable setting when an ECC 1-bit error (Correctable Error) occurs in the lower 64-bit data bus of the Replay FIFO installed in the Data Link Layer 0b: Disable 1b: Enable
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	ERR_REPLAY_HIGHER_UNCORRECTABLE_ERROR EN	0h	RW	Enable ERR_REPLAY_HIGHER_UNCORRECTABLE_ERROR interrupts. Interrupt notification enable setting when an ECC 2-bit or more error (Uncorrectable Error) occurs in the upper 64-bit data bus of the Replay FIFO installed in the Data Link Layer 0b: Disable 1b: Enable

Bit	Bit Name	Initial Value	R/W	Description
0	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR EN	0h	RW	Enable ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR interrupts. Interrupt notification enable setting when an ECC 2-bit or more error (Uncorrectable Error) occurs in the lower 64-bit data bus of the Replay FIFO installed in the Data Link Layer 0b: Disable 1b: Enable

(23) PCIe Event Interrupt Status 1 Register (PCI_RC_PEIS1)

This register is a status register that indicates parity error and ECC error interrupts. After checking the factor, write 1b to the corresponding bit to clear it.

Access Size : 32 bits
Offset Address : <PCI0_base> + 020Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TXB_PARITY_ERR	ERR_RPC_REPLAYFIFO_PERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	ERR_REPLAY_HIGHER_CORRECTABLE_ERROR	ERR_REPLAY_LOWER_CORRECTABLE_ERROR	-	-	-	-	-	-	ERR_REPLAY_HIGHER_UNCORRECTABLE_ERROR	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW1	RW1	R	R	R	R	R	R	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	TXB_PARITY_ERR	0h	RW1	TXB_PARITY_ERR interrupt Parity error interrupt notification of TX Buffer installed in Transaction Layer
16	ERR_RPC_REPLAYFIFO_PERR	0h	RW1	ERR_RPC_REPLAYFIFO_PERR interrupt Parity error interrupt notification of Replay FIFO installed in Data Link Layer
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	ERR_REPLAY_HIGHER_CORRECTABLE_ERROR	0h	RW1	ERR_REPLAY_HIGHER_CORRECTABLE_ERROR interrupt Interrupt notification when an ECC 1-bit error (Correctable Error) occurs in the upper 64-bit data bus of the Replay FIFO installed in the Data Link Layer
8	ERR_REPLAY_LOWER_CORRECTABLE_ERROR	0h	RW1	ERR_REPLAY_LOWER_CORRECTABLE_ERROR interrupt Interrupt notification when an ECC 1-bit error (Correctable Error) occurs in the lower 64-bit data bus of the Replay FIFO installed in the Data Link Layer
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	ERR_REPLAY_HIGHER_UNCORRECTABLE_ERROR	0h	RW1	ERR_REPLAY_HIGHER_UNCORRECTABLE_ERROR interrupt Interrupt notification when an ECC 2-bit or more error (Uncorrectable Error) occurs in the upper 64-bit data bus of the Replay FIFO installed in the Data Link Layer
0	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR	0h	RW1	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR interrupt Interrupt notification when an ECC 2-bit or more error (Uncorrectable Error) occurs in the lower 64-bit data bus of the Replay FIFO installed in the Data Link Layer

(24) AXI Master Error Interrupt Enable Register (PCI_RC_AMEIE)

This register enables the AXI master error interrupt.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0210h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	Write MSTERR INT EN[3:0]				-	-	-	-	Read MSTERR INT EN[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	Write MSTERR INT EN[3:0]	0h	RW	Write MSTERR INT Enable Each corresponding bit can be turned on/off individually. 0b: Disable 1b: Enable
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	Read MSTERR INT EN[3:0]	0h	RW	Read MSTERR INT Enable Each corresponding bit can be turned on/off individually. 0b: Disable 1b: Enable

(25) AXI Master Error Interrupt Status Register (PCI_RC_AMEIS)

This register indicates the AXI master error interrupt status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0214h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	Write ERR ID[3:0]				-	-	-	-	Read ERR ID[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	Write MSTERR INT[3:0]				-	-	-	-	Read MSTERR INT[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW1	RW1	RW1	RW1	R	R	R	R	RW1	RW1	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27 to 24	Write ERR ID[3:0]	0h	R	Save the ID of the first DECERR/SLVERR received. When bits [11:8] are cleared, a new error ID can be saved. 0h: normal access
23 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 16	Read ERR ID[3:0]	0h	R	Save the ID of the first DECERR/SLVERR received. A new error ID can be saved when bits [3:0] are cleared. 0h: normal access
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	Write MSTERR INT[3:0]	0h	RW1	Indicates that an error was detected in the AXI Master Port. Only the first detected error is saved, and when bits [11:8] are cleared, a new error can be saved. bit[11]: length error When the length of the data sent by the TEF and the data channel do not match. bit[10]: ID mismatch When the MBID value received on the MAWID response channel is different. bit[9]: When DECERR is received bit[8]: When SLVERR is received
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	Read MSTERR INT[3:0]	0h	RW1	Indicates that an error was detected in the AXI Master Port. Only the first detected error is saved, and when bits [3:0] are cleared, a new error can be saved. bit[3]: length error When the length of the data received on the TER and the data channel do not match. bit[2]: ID mismatch When the MARID and MRID values received on the data channel are different. bit[1]: When DECERR is received bit[0]: When SLVERR is received

(26) AXI Slave Error Interrupt Enable 1 Register (PCI_RC_ASEIE1)

This register enables the AXI slave error interrupt.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0220h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	Write SLVERR INT EN[3:0]				-	-	-	-	-	-	Read SLVERR INT EN[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	Write SLVERR INT EN[3:0]	0h	RW	Enable Write SLVERR INT. Each corresponding bit can be turned on/off individually. 0b: Disable 1b: Enable
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	Read SLVERR INT EN[1:0]	0h	RW	Enable Read SLVERR INT. Each corresponding bit can be turned on/off individually. 0b: Disable 1b: Enable

(27) AXI Slave Error Interrupt Status 1 Register (PCI_RC_ASEIS1)

This register indicates the AXI slave error interrupt status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0224h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	Write SLVERR INT[3:0]				-	-	-	-	-	-	Read SLVERR INT[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW1	RW1	RW1	RW1	R	R	R	R	R	R	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	Write SLVERR INT[3:0]	0h	RW1	Indicates that an unrecoverable error was detected in the AXI Slave Port. (Transaction response will be SLVERR.) bit[11]: burst length error When the SAWLEN and the burst length of the data received on the data channel do not match. bit[10]: ID mismatch When the SAWID and SWID values received on the data channel are different. bit[9]: Burst type disabled When SAWBURST is 11b (undefined). When SAWBURST is 10b (wrapping) and burst length is other than 2, 4, 8, or 16. bit[8]: data size invalid When SAWSIZE is between 100b and 111b (Exceeding the AXI Bus width is not supported). Each bit means: 0b: No error detected 1b: error detection
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	Read SLVERR INT[1:0]	0h	RW1	Indicates that an unrecoverable error was detected in the AXI Slave Port. (Transaction response will be SLVERR.) bit1b: Burst type disabled When SARBURST is 11b (undefined). When SARBURST is 10b (wrapping) and burst length is other than 2, 4, 8, or 16. bit0b: Data size invalid When SARSIZE is between 100b and 111b (exceeding the AXI Bus width is not supported). Each bit means: 0b: No error detected 1b: error detection

(28) AXI Slave Error Interrupt Status 3 Register (PCI_RC_ASEIS3)

This register indicates the AXI ID at the time of the first AXI slave error.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0230h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ERR ID[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERR ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ERR ID[31:0]	0h	R	Save the ID when the first error occurred in the AXI Slave Error Interrupt Status1 register (Offset 224h). Only the first detected error is saved, and when bits [11:8] and bits [1:0] of the AXI Slave Error Interrupt Status1 register are all cleared, a new error ID can be saved.

(29) Permission Register (PCI_RC_PERM)

Access Size : 32 bits
Offset Address : <PCI0_base> + 0300h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	CFG_HWINIT_EN	PIPE PHY Register Enable	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	CFG_HWINIT_EN	0h	RW	Enable HwInit attribute registers to be rewritten from AXI. 0b: Disable 1b: Enable
1	PIPE PHY Register Enable	0h	RW	Access permission signal for Physical Layer Control/Status Registers. 0b: Disable register access in PIPE PHY space. 1b: Enable register access in PIPE PHY space.
0	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.

(30) Reset Register (PCI_RC_RESET)

This register resets the PCIe core. Supplied to the internal core by OR with the pin of the same name. For details of each pin, see the terminal information. The write value is saved when accessed from the AXI, but a low-level pulse is generated when the PCIe writes 0b. However, when the AXI has already written 0b, the signal remains at the low level. Writing 1b from the PCIe will be ignored.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0310h
Initial Value : 0000_00xxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	force to D0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RST_PREG_B	RST_OUT_B	RST_PS_B	RST_LOAD_B	RST_CFG_B	RST_RSM_B	RST_GP_B	RST_B
Initial Value	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	force to D0	0h	RW	After sending PME_TO_Ack, force PM Control to transition to D0 State. Auto-cleared when PM Control transitions to D0. This bit is normally prohibited as it can create inconsistencies with the power state of the entire system. 0b: No operation 1b: Transition to D0
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
7	RST_PREG_B	xh	RW	Not used in this macro. The setting value does not affect macro behavior. 0b: reset 1b: normal operation
6	RST_OUT_B	xh	RW	RST_OUT_B output 0b: reset 1b: normal operation
5	RST_PS_B	xh	RW	Reset to the PCI Express core part (PCLK domain) inside the macro 0b: reset 1b: normal operation
4	RST_LOAD_B	xh	RW	Reset to Configuration Register Reset to bits not initialized by RST_CFG_B. 0b: reset 1b: normal operation
3	RST_CFG_B	xh	RW	Reset to Configuration Register 0b: Reset 1b: normal operation
2	RST_RSM_B	xh	RW	POWERGOOD reset of AUX Power (AUX not supported) Reset to Sticky register. 0b: reset 1b: normal operation
1	RST_GP_B	xh	RW	Reset to the PCI Express core part (ACLK domain) inside the macro 0b: reset 1b: normal operation
0	RST_B	xh	RW	Reset to PCI Express core part inside macro 0b: reset 1b: normal operation

Note: x = Undefined

(31) Mode Set 0 Register (PCI_RC_MSET0)

This register sets AXI mode.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0314h
Initial Value : 2001_2000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	AWPROT[2:0]			AWCACHE_L[3:0]				-	-	AWLOCK[1:0]		AWCACHE_D[3:0]			
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	ARPROT[2:0]			-	-	-	-	-	-	ARLOCK[1:0]		ARCACHE[3:0]			
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 28	AWPROT[2:0]	2h	RW	Sets the protection type for PCIe→AXI transactions. This bit indicates whether the transaction's protection level is Normal, Privileged, or Secure, and whether the transaction is a data or instruction access. Bit[2] 1b: instruction access 0b: data access Bit[1] 1b: non-secure access 0b: secure access Bit[0] 1b: privileged access 0b: normal access
27 to 24	AWCACHE_L [3:0]	0h	RW	Indicates the value of MAWCACHE[3:0] to be issued to AXI. This setting is prohibited when issuing an AXI request containing the last byte. *Recommended value is 0000b. Bit[3] 1b: write allocatable 0b: not write allocatable Bit[2] 1b: Read assignable 0b: Read assignable Bit[1] 1b: cacheable 0b: non-cacheable Bit[0] 1b: Bufferable 0b: Not Bufferable
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21, 20	AWLOCK[1:0]	0h	RW	Lock type for PCIe-to-AXI transactions. This signal provides information about the atomic nature of the transfer. 00b: normal access, 01b: exclusive access, 10b: locked access, 11b: reserved
19 to 16	AWCACHE_D [3:0]	1h	RW	Indicates the value of MAWCACHE[3:0] to be issued to AXI. This setting is output when issuing an AXI request other than the output condition of AWCACHE_L. *Recommended value is 0001b. Bit[3] 1b: write allocatable 0b: not write allocatable Bit[2] 1b: Read assignable 0b: Read assignable Bit[1] 1b: cacheable 0b: non-cacheable Bit[0] 1b: Bufferable 0b: Not Bufferable
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 12	ARPROT[2:0]	2h	RW	Sets the protection type for PCIe→AXI transactions. This bit indicates whether the transaction's protection level is Normal, Privileged, or Secure, and whether the transaction is a data or instruction access. Bit[2] 1b: instruction access 0b: data access Bit[1] 1b: non-secure access 0b: secure access Bit[0] 1b: privileged access 0b: normal access
11 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5, 4	ARLOCK[1:0]	0h	RW	Lock type for PCIe-to-AXI transactions. This bit provides information about the atomic nature of the transfer. 00b: normal access, 01b: exclusive access, 10b: locked access, 11b: reserved
3 to 0	ARCACHE[3:0]	0h	RW	Cache type for PCIe-to-AXI transactions. These bits specify “bufferable”, “cacheable”, “writethrough”, “write-back”, or “allocation” as the attribute of the transaction.

(32) Mode Set 1 Register (PCI_RC_MSET1)

This register sets AXI mode.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0318h
Initial Value : 0000_33F2h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AXI Max Issue Write[3:0]				AXI Max Issue Read[3:0]				AXI Master Max Burst[3:0]				-	-	RAM Parity Enable	PCIe Request Order
Initial Value	0	0	1	1	0	0	1	1	1	1	1	1	0	0	1	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 12	AXI Max Issue Write[3:0]	3h	RW	Set the number of writes that can be issued by AXI Master. Set within the range allowed by Interconnect. 0h: possible number 1 1h: possible number 2 : Fh: Possible number 16
11 to 8	AXI Max Issue Read[3:0]	3h	RW	Set the number of read issues that can be issued by AXI Master. Set within the range allowed by Interconnect. 0h: possible number 1 1h: possible number 2 : Fh: Possible number 16
7 to 4	AXI Master Max Burst[3:0]	Fh	RW	Sets the maximum burst length as an AXI Master operation.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	RAM Parity Enable	1h	RW	Sets whether or not to check the parity of the internal SRAM. The default value is Enable, but it is ignored by non-Parity macros. 0b: RAM parity check disabled 1b: RAM parity check enabled
0	PCIe Request Order	0h	RW	Issue a Read Request to PCIe from the same AXI master without waiting for Completion. Set to 1 if you want to strictly follow the order of Requests to the Completer. 0b: Do not wait for Completion. 1b: Wait for Completion.

(33) Mode Set 3 Register (PCI_RC_MSET3)

This register outputs the setting value as the ASPM L1 Idle Time bit.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0380h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ASPM L1 Idle Time[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
7 to 0	ASPM L1 Idle Time[7:0]	0h	RW	Sets the idle period for AXI transactions that the macro checks on ASPM L1 transitions. One of the conditions for ASPM L1 transition is that the idle period is confirmed for the number of cycles of the 8 bits set by this bit plus 8'hFF to the lower 8 bits. 00h: 256 [ACLK] 01h: 512 [ACLK] : FFh: 65536 [ACLK]

(34) Mode Set 4 Register (PCI_RC_MSET4)

Access Size : 32 bits
Offset Address : <PCI0_base> + 0384h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MODE_EQ_RX_EVAL_WAIT[19:4]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MODE_EQ_RX_EVAL_WAIT[3:0]				-	-	-	-	MODE_8GT_EQTS2	MODE_EQ_PHASE23_ENABLE1	MODE_EQ_AUTONOMOUS1	-	-	UI_EQ_REQTYPE[1:0]		UI_EQ_REQUEST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	MODE_EQ_RX_EVAL_WAIT[19:0]	0h	RW	During RxEval (Downstream Port Phase3, Upstream Port Phase2), wait time setting until RxEval is executed when Block Alignment cannot be obtained. Set by the count number of PCLK. Note: It is necessary to set the register value (recommended: equivalent to 1msec) before starting. Please set from the local CPU.
11 to 8	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
7	MODE_8GT_EQTS2	0h	RW	8GT EQ TS2OS transmission enable/disable setting for Upstream Port 0b: Do not send, 1b: Send Note: Not used for Downstream Port. (fixed to 0b)
6	MODE_EQ_PHASE23_ENABLE1	0h	RW	Fixed to 0b
5	MODE_EQ_AUTONOMOUS1	0h	RW	Fixed to 0b
4, 3	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
2, 1	UI_EQ_REQTYPE[1:0]	0h	RW	Operation specification when sending Equalization Request with Quiesce Guarantee=1 in Recovery.RcvrCfg state 00b: Equalization Request transmission with Equalization Request Data Rate=0 (8G) 01b: Send Equalization Request with Equalization Request Data Rate=1 (16G) (Setting prohibited) 10b: Operation of “0” + set Link Equalization Request 8.0GT/s register 11b: Set “1” operation + Link Equalization Request 16.0GT/s register (setting prohibited)
0	UI_EQ_REQUEST	0h	RW	Set when sending an Equalization Request with Quiesce Guarantee=1 in the Recovery.RcvrCfg state. Setting 1b is prohibited except when MODE_QUIESCE_GUARANTEE=1. After setting 1b, hold until confirmation of UI_EQ_DONE=1. 0b: Do not send Equalization Request 1b: Send Equalization Request

(35) Mode Set 5 Register (PCI_RC_MSET5)

Access Size : 32 bits
Offset Address : <PCI0_base> + 0388h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MODE_8GT_EQTS2_PRESET[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
15 to 0	MODE_8GT_EQ TS2_PRESET [15:0]	0h	RW	Transmitter Preset value of 8GT EQ TS2OS to transmit at Upstream Port (USP). No setting is required if 8GT_EQTS2 is not sent from USP (DSP Preset is not specified). Settings must be made in advance if necessary. 4-bit configuration for each lane. Gen3/Gen4 supports up to x4 lane configurations. The correspondence between each bit and lane number is as follows. Bit[3:0] Lane #0 Bit[7:4] Lane #1 Bit[11:8] Setting Prohibited, other than default value is not writable. Bit[15:12] Setting Prohibited, other than default value is not writable. The above lane number is the default lane number for circuit implementation. Note that this is not a negotiated lane number. Note: Not used for Downstream Port. (fixed to 0000_0000h)

(36) Mode Status 0 Register (PCI_RC_MSTAT0)

Access Size : 32 bits
Offset Address : <PCI0_base> + 0390h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	RX_EQ_REQTYPE[1:0]		RX_EQ_REQUEST	UI_EQ_DONE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	RX_EQ_REQTYPE[1:0]	0h	R	Quiesce Guarantee and Equalization Request Data Rate bit display when receiving 8 consecutive TS2OS with Request Equalization bit (Symbol6, bit7) = 1 in Recovery. RcvrCfg state EP (USP) (1) When Request Equalization is received bit[0]: Quiesce Guarantee (TS2OS, Symbol6, bit6) (= 1 fixed) bit[1]: Equalization Request Data Rate (TS2OS, Symbol6, bit5) (2) When receiving EQTS2OS bit[0]: 0 (fixed) bit[1]: Equalization Request Data Rate (0b: EQTS2, 1b: 8GT EQTS2) RC (DSP) bit[0]: Quiesce Guarantee (TS2OS, Symbol6, bit6) bit[1]: Equalization Request Data Rate (TS2OS, Symbol6, bit5)
1	RX_EQ_REQUEST	0h	R	Status display when receiving 8 consecutive TS2OS with Request Equalization bit (Symbol6, bit7) = 1 in Recovery.RcvrCfg state 0b: No Equalization Request received 1b: Equalization Request received
0	UI_EQ_DONE	0h	R	Set Equalization Request with Quiesce Guarantee=1 in Recovery.RcvrCfg state when sending 0b: No Equalization Request sent 1b: Sent Equalization Request Related flow: "Set UI_EQ_REQUEST = 1b → Confirm UI_EQ_DONE = 1b → Clear UI_EQ_REQUEST = 0b"

(37) PCIe Core Mode Set 1 Register (PCI_RC_PCMSET1)

This register sets the operating mode of the PCI Express core.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0400h
Initial Value : 00FA_00F2h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	ASPM_L1_INTERVAL_TIME[11:0]											
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	MODE_LINK_UPCONFIGURE_CAPABILITY_DISABLED	-	MODE_TXSWING	-	-	-	-	-	-	-	-	MODE_PORT	-
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	0	0	1	0
R/W	R	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29, 28	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
27 to 16	ASPM_L1_INTERVAL_TIME [11:0]	0FAh	RW	Interval settings for ASPM L1 requests The PCIe Base Spec stipulates that ASPM L1 transition requests must not be accepted continuously within 10 us, and this field sets the timer value to guard against this. Set so that ACLK cycle x set value is 10 us or more. At ACLK: 400[MHz] or higher, this bit setting value should be the value (1/16) of the lower 4 bits of the setting value of the above specifications, and set the clock number as the timer value setting. Settings of this product (Default): When ACLK=400MHz(2.5ns): 4000(d)/16 = FA(h)
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14, 13	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
12	MODE_LINK_UPCONFIGURE_CAPABILITY_DISABLED	0h	RW	Setting of Link Upconfigure Capability bit of Training Sequence Ordered-set (TS-OS) 0b: Link Upconfigure Capability bit = 1b setting 1b: Link Upconfigure Capability bit = 0b setting (Gen1 x1) When connecting with a Gen1 PCIe device, Linkup may not occur unless this bit is 0b. In that case, change it to 0b in F/W.
11	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	MODE_TXSWING	0h	RW	SerDes serial output amplitude control 0b: Full swing mode (default) 1b: Half swing mode
9, 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 4	-	Fh	RW	Reserved These bits are read as Fh. The write value should always be Fh.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
1	MODE_PORT	1h	RW	Device type setting register. When the bit value of this register is fixed to the initial value 1b, the setting of the external pin MODE_PORT becomes valid. When the external pin MODE_PORT is fixed to 1b, the setting by this register bit becomes valid. 0b: Endpoint 1b: Root Complex
0	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.

(38) PCIe Core Control 1 Register (PCI_RC_PCCTRL1)

This register controls power management and LTSSM (Link Training Sequence State Machine) state transitions of the PCI Express core.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0404h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	BLB_RELAX_ORDERING_EN	-	-	-	-	-	UI_ENTER_L1S	-	RETURN_TO_L0	UI_RC_REJECT_ASPML1	Auto PM_Active_State_Nak	UI_ENTER_L2	UI_ENTER_TXL0S
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	MODE_QUIESCE_GUARANTEE	-	MODE_EQ_AUTONOMOUS	MODE_EQ_PHASE23_ENABLE	MODE_RESET_EIEOS_INTERVALL0S	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	BLB_RELAX_ORDERING_EN	0h	RW	Control of RO bit of Request to be sent 0b: RO bit of Request TLP to be sent is always 0b (default) 1b: A TLP can be sent with the RO bit of the Request TLP to be sent set to 1b.
27 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
23	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	UI_ENTER_L1S	0h	RW	L1SubState transition permission setting 0b : L1 Substate transition disabled (default) 1b: L1 Substate transition permission (setting prohibited)
21	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
20	RETURN_TO_L0	0h	RW	RC mode L1, L2 state to L0 state control (usually not used) 0b : Normal operation (default) 1b : Start return operation to L0 state when in L1 or L2 state in RC mode Cleared automatically after confirming PMU_LINKSTATE[0] =1.
19	UI_RC_REJECT_ASPML1	0h	RW	ASPM L1 transition rejection control 0b : Accept ASPM L1 transition request from EP device (default) 1b : Reject ASPM L1 transition request from EP device
18	Auto PM_Active_State_Nak	0h	RW	PM_ActiveState_Nak Message Transmission Mode for ASPM L1 Rejection Set to 1b if you want to reject ASPM L1 in RC. This Bit is automatically cleared when PM_ActiveState_Nak is automatically sent. Note: Auto-sent only once.
17	UI_ENTER_L2	0h	RW	RC mode L2 transition control Set to 1b when transitioning to L2 state in RC mode. When transitioning to the L2 state, the PCIe core must be reset by controlling the Reset register. When returning, this bit must be cleared to 0b after releasing the reset.
16	UI_ENTER_TXL0S	0h	RW	TxL0s transition control 0b: Do not perform ASPM L0s transition (default) 1b: Execute ASPM L0s transition when internal conditions are satisfied

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12	MODE_QUIESCE_GUARANTEE	0h	RW	Symbol6 bit6 Quiesce Guarantee control bit of TS2OS 0b: Set 0b to TS2OS (default) 1b: Set 1b to TS2OS
11	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
10	MODE_EQ_AUTONOMOUS	0h	RW	Gen3 feature: Autonomous Equalization Basically only changeable during the reset period 0b: Do not use Autonomous Mechanism 1b: Use Autonomous Mechanisms
9	MODE_EQ_PHASE23_ENABLE	0h	RW	Gen3 features: Setting whether to execute EQ PHASE2 and EQ PHASE3 in RC mode (MODE_PORT=1). 0b: Do not execute EQ PHASE2/3 1b: Execute EQ PHASE2/3
8	MODE_RESET_EIEOS_INTERVALL0S	0h	RW	Gen3 features: Reset EIEOS Interval information of bit 2, symbol 6 of TS1OS transmitted in Recovery.Equalization state
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.

(39) PCIe Core Status 1 Register (PCI_RC_PCSTAT1)

This register controls power management and LTSSM (Link Training Sequence State Machine) state transitions of the PCI Express core.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0408h
Initial Value : 000x_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	bme_down	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	LTSSM_STATE[6:0]							PMU_LINKSTATE[3:0]				-	-	STATE_VC0_NEGOTIATION_PENDING	DL_Down status
Initial Value	0	x	x	x	x	x	x	x	x	x	x	x	0	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

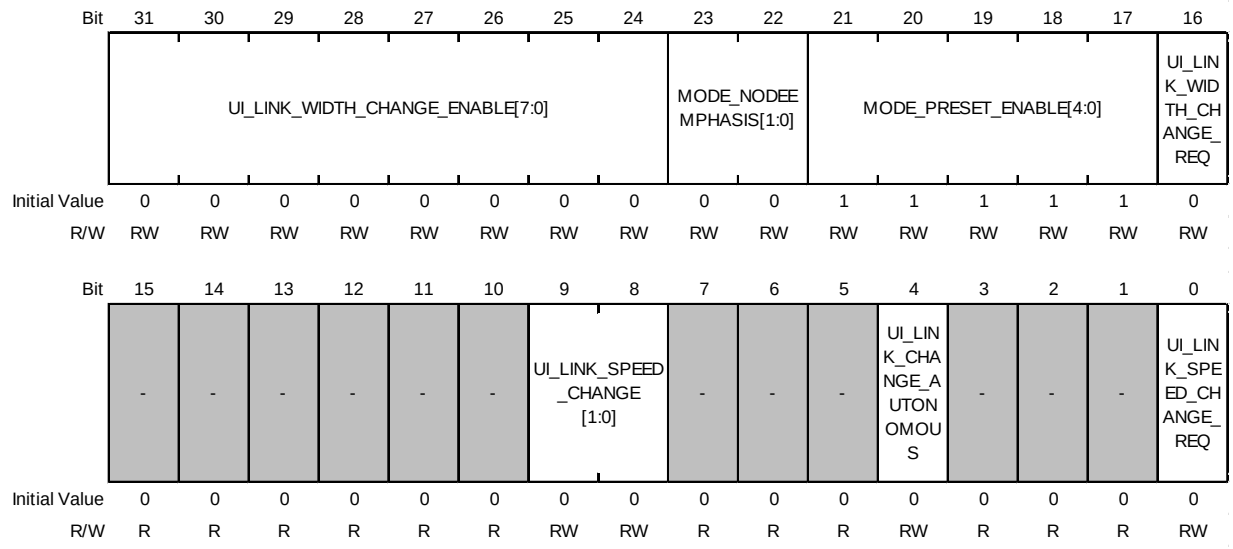
Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	bme_down	xh	R	Indicates that the PCIe core transmitter is in an unusable state.
16, 15	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 8	LTSSM_STATE[6:0]	xh	R	Indicates the state of the Link Training & Status State Machine in the PCIe Core Link. The following states are indicated by the upper 5 bits [14:10]. 000xxb: Detect 001xxb: Polling 010xxb: Config 01100b: L0 01101b: L1 0111xb: L2 100xxb: Recovery 101xxb: Disable 110xxb: Loopback
7 to 4	PMU_LINKSTATE[3:0]	xh	R	L-state monitor of power management control unit 0100b: L1 state 1000b: L2 state
3	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	-	xh	R	Reserved Whenever it is read, xb is read. The written value will be ignored.
1	STATE_VC0_NEGOTIATION_PENDING	xh	R	Flow Control initialization operation monitor If this bit is 1b, do not initiate a transaction from the AXI side. Check that this bit is 0b and DL_Down Status (bit [0]) is 0b. 0b: Indicates that Flow Control initialization is complete 1b: Indicates that Flow Control initialization has not completed
0	DL_Down status	xh	R	Indicates whether PCIe Core is in DL_Down or DL_Up state 0b: DL_Up Status 1b: DL_Down Status

Note: x = Undefined

(40) PCIe Core Control 2 Register (PCI_RC_PCCTRL2)

This register controls the link speed/width change in the PCI Express core.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0410h
Initial Value : 003E_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	UI_LINK_WIDTH_CHANGE_ENABLE[7:0]	0h	RW	Link Width setting to change Assert UI_LINK_WIDTH_CHANGE_REQ and set Lane to 1b to operate when Link Width change request is issued. The lower bit (bit [24]) drives Lane0 and the most significant bit (bit [31]) drives Lane7.
23, 22	MODE_NODEEMPHASIS[1:0]	0h	RW	No de-emphasis mode setting pin for Gen1/Gen2 operation Bit[0] Gen1 operation, 0b: Normal operation mode (default), 1b: No de-emphasis mode Bit[1] Gen2 operation, 0b: Normal operation mode (default), 1b: No de-emphasis mode
21 to 17	MODE_PRESET_ENABLE[4:0]	1Fh	RW	Reduced Swing mode setting pin for Gen3 operation Bit[0]: 8GT/s Preset P0, 0b: disable, 1b: enable (default) Bit[1]: 8GT/s Preset P2, 0b: disable, 1b: enable (default) Bit[2]: 8GT/s Preset P7, 0b: disable, 1b: enable (default) Bit[3]: 8GT/s Preset P8, 0b: disable, 1b: enable (default) Bit[4]: 8GT/s Preset P10, 0b: disable, 1b: enable (default)
16	UI_LINK_WIDTH_CHANGE_REQ	0h	RW	Link Width change request control Setting this bit to 1b issues a request to change the Link Width to the configuration set in bits [31:24] UI_LINK_WIDTHCHANGE_ENABLE field. By asserting it in the L0 state, it transitions from the Recovery state to the Configuration state, and performs negotiation with the other device. Set to 0b after confirming that PCIe Core Status 2 Register (Offset: 414h) bit [29] UI_LINK_WIDTH_CHANGE_DONE is asserted.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9, 8	UI_LINK_SPEED_CHANGE [1:0]	0h	RW	Link speed setting Set the Link Speed you want to change. 00b: 2.5 GT/s 01b: 5.0 GT/s 10b: 8.0 GT/s 11b: 16.0 GT/s (Setting prohibited)
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	UI_LINK_CHANGE_AUTONOMOUS	0h	RW	Link Width/Speed change reason setting 0b: reliability reason (change for reliability, direction of bandwidth reduction) 1b: autonomous reason (intentional change)
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	UI_LINK_SPEED_CHANGE_REQ	0h	RW	Link Speed change request control Setting this bit to 1b requests to change the Link Speed to the speed set in bit [8] UI_LINK_SPEED_CHANGE field. By asserting it in the L0 state, it transitions to the Recovery state and performs negotiation with the peer device. PCIe Core Status 2 Register (Offset: 414h) bit[28] Set to 0b after confirming that UI_LINK_SPEED_CHANGE_DONE has been asserted.

(41) PCIe Core Status 2 Register (PCI_RC_PCSTAT2)

This register indicates the status of the link speed/width change in the PCI Express core.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0414h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	UI_LINK_WIDTH_CHANGE_DONE	UI_LINK_SPEED_CHANGE_DONE	-	-	-	STATE_UPCONFIGURE_CAPABLE	-	STATE_NEGOTIATED_LANE_END[2:0]			-	STATE_NEGOTIATED_LANE_START[2:0]		
Initial Value	0	0	x	x	0	0	0	x	0	x	x	x	0	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STATE_RECEIVER_DETECTED[7:0]								STATE_DATA_RATE_IDENTIFIER_RECEIVED[7:0]							
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29	UI_LINK_WIDTH_CHANGE_DONE	xh	R	Link Width Change operation complete display Notifies completion of Width Change (1b) by setting PCIe Core Status 2 Register (Offset: 410h) bit [16] UI_LINK_WIDTH_CHANGE_REQ. It is set to 0b by setting UI_LINK_WIDTH_CHANGE_REQ to 0b.
28	UI_LINK_SPEED_CHANGE_DONE	xh	R	Link Speed Change operation complete display PCIe Core Status 2 Registers (Offset: 410h) bit[0] Notifies completion of Speed Change (1b) by setting UI_LINK_SPEED_CHANGE_REQ. It is set to 0b by setting UI_LINK_SPEED_CHANGE_REQ to 0b.
27 to 25	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	STATE_UPCONFIGURE_CAPABLE	xh	R	Upconfigure Capable bit display of opposite device Indicates whether the peer device supports changing the direction of widening the Link Width. If this bit is 0b, changing the Link Width will not restore the original Link Width.
23	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
22 to 20	STATE_NEGOTIATED_LANE_END[2:0]	xh	R	Displays the position of Lane Number (n-1) (meaning Lane1 when n=2) after Link Negotiation with the opposite device during n-lane operation. Used to check the state of the current working lane before changing the Link Width. 000b: Lane0 is Lane Number (n-1) 001b: Lane1 is Lane Number (n-1) Others: reserved
19	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
18 to 16	STATE_NEGOTIATED_LANE_START[2:0]	xh	R	Displays the position of Lane Number 0 after Link Negotiation with the opposite device during n-lane operation. Used to check the state of the current working lane before changing the Link Width. 000b: Lane0 is Lane Number 0 001b: Lane1 is Lane Number 0 Others: reserved
15 to 8	STATE_RECEIVER_DETECTED[7:0]	xh	R	Connection status display with other device Receiver Detection results are displayed. Bit[0] Detect opposite device on Lane0 Bit[1] Detect peer device on Lane1 (only for x2) Bit[2:7] reserved

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	STATE_DATA_ RATE_IDENTIFIE R_RECEIVED [7:0]	xh	R	Link Speed display supported by the opposite device Displays the TS-OS Data Rate Identifier feed received from the peer device. Bit[0] – Reserved Bit[1] – 2.5 GT/s Data Rate Supported. Must be set to 1b. Bit[2] – 5.0 GT/s Data Rate Supported. Must be set to 1b if Bit3 is 1b. Bit[3] – 8.0 GT/s Data Rate Supported. Bits[4:7] – Reserved

Note: x = Undefined

(42) PCIe Core Status 5 Register (PCI_RC_PCSTAT5)

This register indicates the status in the PCI Express core.

Access Size : 32 bits
Offset Address : <PCI0_base> + 042Ch
Initial Value : 0000_0x00h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ORT_TRANSACTION_PENDING
Initial Value	0	0	0	0	0	0	0	x	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	0000_00x0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	ORT_TRANSACTION_PENDING	0h	R	Indicates whether or not there are Outstanding Requests (a state in which all Completions corresponding to Non-Posted Requests sent from the AXI side have not been received). Check with this bit that there is no Outstanding Request before requesting/permitting a transition to TxL0s/L1/L2. 0b: State without Outstanding Request 1b: State with Outstanding Request

(43) DMA Interrupt Vector 0 Register (PCI_RC_DMAINTVEC0)

This register specifies the interrupt vectors for interrupt notification from AXI to PCIe (MSI) during DMA transfer.

Access Size : 32 bits
Offset Address : <PCI0_base> + 04D0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	DMA_CH3_MSI_EN	DMA_CH3_vec[4:0]					-	-	DMA_CH2_MSI_EN	DMA_CH2_vec[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	DMA_CH1_MSI_EN	DMA_CH1_vec[4:0]					-	-	DMA_CH0_MSI_EN	DMA_CH0_vec[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29	DMA_CH3_MSI_EN	0h	RW	DMA Ch3 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI For the RC function, MSI transmission is prohibited, so please fix it to 0b.
28 to 24	DMA_CH3_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch3
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21	DMA_CH2_MSI_EN	0h	RW	DMA Ch2 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI For the RC function, MSI transmission is prohibited, so please fix it to 0b.
20 to 16	DMA_CH2_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch2
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	DMA_CH1_MSI_EN	0h	RW	DMA Ch1 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI For the RC function, MSI transmission is prohibited, so please fix it to 0b.
12 to 8	DMA_CH1_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch1
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	DMA_CH0_MSI_EN	0h	RW	DMA Ch0 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI For the RC function, MSI transmission is prohibited, so please fix it to 0b.
4 to 0	DMA_CH0_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch0

Note 1. DMA_CHx_vec should be fixed to 0h in Root Complex mode.

(44) DMA Interrupt Vector 1 Register (PCI_RC_DMAINTVEC1)

This register specifies the interrupt vectors for interrupt notification from AXI to PCIe (MSI) during DMA transfer.

Access Size : 32 bits
Offset Address : <PCI0_base> + 04D4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	DMA_CH7_MSI_EN	DMA_CH7_vec[4:0]					-	-	DMA_CH6_MSI_EN	DMA_CH6_vec[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	DMA_CH5_MSI_EN	DMA_CH5_vec[4:0]					-	-	DMA_CH4_MSI_EN	DMA_CH4_vec[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29	DMA_CH7_MSI_EN	0h	RW	DMA Ch7 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI For the RC function, MSI transmission is prohibited, so please fix it to 0b.
28 to 24	DMA_CH7_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch7
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21	DMA_CH6_MSI_EN	0h	RW	DMA Ch6 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI For the RC function, MSI transmission is prohibited, so please fix it to 0b.
20 to 16	DMA_CH6_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch6
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	DMA_CH5_MSI_EN	0h	RW	DMA Ch5 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI For the RC function, MSI transmission is prohibited, so please fix it to 0b.
12 to 8	DMA_CH5_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch5
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	DMA_CH4_MSI_EN	0h	RW	DMA Ch4 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI For the RC function, MSI transmission is prohibited, so please fix it to 0b.
4 to 0	DMA_CH4_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch4

Note 1. DMA_CHx_vec should be fixed to 0h in Root Complex mode.

(45) MSI Receive Enable n Register (PCI_RC_MSIRCVEn)

This register enables control of MSI receive (Message Data discrimination) register group with the following as one set.

- MSI Receive Enable n Register (Offset 6x0h)
- MSI Receive message Data n Register (Offset 6x4h)
- MSI Receive Mask n Register (Offset 6x8h)
- MSI Receive Status n Register (Offset 6xCh)

Remark n is 0 to 15; x is the hexadecimal representation of n.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0600h/ 610h/ 620h/ 630h/ 640h/ … /6F0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	Enable	0h	RW	Set enable/disable of the MSI Receive (Message Data discrimination) register group, which sets 1 register for +16 bytes from Offset of this register. 0b: disabled 1b: Enabled

(46) MSI Receive Message Data n Register (PCI_RC_MSIRCVMSGDATAn)

This register is a Message data setting register for MSI reception judgment of registers enabled by MSI Receive Enable n Register.

Remark n is 0 to 15; x is the hexadecimal representation of n.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0604h/ 614h/ 624h/ 634h/ 644h/ … /6F4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Message Data[26:11]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Data[10:0]											-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	Message Data[26:0]	0h	RW	Set Message Data for MSI reception judgment
4 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(47) MSI Receive Mask n Register (PCI_RC_MSIRCVMSKn)

This register masks control of interrupt by the MSI reception status of the register group enabled by MSI Receive Enable n Register.

Remark n is 0 to 15; x is the hexadecimal representation of n.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0608h/ 618h/ 628h/ 638h/ 648h/ … /6F8h
Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSI Mask[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSI Mask[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MSI Mask[31:0]	FFFFFFFFh	RW	Set the interrupt Mask by each bit of MSI Status. 0b: Interrupt enabled 1b: Interrupt Mask

(48) MSI Receive Status n Register (PCI_RC_MSIRCVSTATn)

This register indicates the MSI receive status of the registers enabled by the MSI Receive Enable n Register.

Remark n is 0 to 15; x is the hexadecimal representation of n.

Access Size : 32 bits

Offset Address : <PCI0_base> + 060Ch/ 61Ch/ 62Ch/ 63Ch/ 64Ch/ … /6FCh

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSI Status[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSI Status[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MSI Status[31:0]	0h	RW1	MSI receive Status. The bit corresponding to Data[4:0] is set when Message Data[31:5] and MSI reception Data[31:5] match. (When Data[4:0]=00h, MSI Status n[0] is set. When Data[4:0]=01h, MSI Status n[1] is set.)

(49) DMAC Control Register (PCI_RC_DMACTRL)

This register sets the maximum size of read requests which can be issued to the PCIe core as a DMAC function. Use the initial setting (128 bytes). This setting is common to all DMA channels.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0800h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	D_PMRS[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	D_PMRS[2:0]	0h	RW	DMAC PCIe Max Read Request Size Set the upper limit of Read requests issued from the DMAC to PCIe. 000b: 128 bytes (default) 001b: 256 bytes 010b: 512 bytes (not supported) 011b: 1024 bytes (not supported) 100b: 2048 bytes (not supported) 101b: 4096 bytes (not supported) others: Reserved (prohibited)

(50) DMAC Interrupt Enable Register (PCI_RC_DMAINTE)

This register enables interrupts from the individual DMA channels.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0808h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CH7_ERR_EN	CH7_QUE_EMP_EN	CH7_STOP_EN	CH7_END_EN	CH6_ERR_EN	CH6_QUE_EMP_EN	CH6_STOP_EN	CH6_END_EN	CH5_ERR_EN	CH5_QUE_EMP_EN	CH5_STOP_EN	CH5_END_EN	CH4_ERR_EN	CH4_QUE_EMP_EN	CH4_STOP_EN	CH4_END_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH3_ERR_EN	CH3_QUE_EMP_EN	CH3_STOP_EN	CH3_END_EN	CH2_ERR_EN	CH2_QUE_EMP_EN	CH2_STOP_EN	CH2_END_EN	CH1_ERR_EN	CH1_QUE_EMP_EN	CH1_STOP_EN	CH1_END_EN	CH0_ERR_EN	CH0_QUE_EMP_EN	CH0_STOP_EN	CH0_END_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CH7_ERR_EN	0h	RW	CH7 Error Interrupt Enable 1b: Enable 0b: Disable
30	CH7_QUE_EMP_EN	0h	RW	CH7 Que Empty Interrupt Enable 1b: Enable 0b: Disable
29	CH7_STOP_EN	0h	RW	CH7 Stop Interrupt Enable 1b: Enable 0b: Disable
28	CH7_END_EN	0h	RW	CH7 Completion Interrupt Enable 1b: Enable 0b: Disable
27	CH6_ERR_EN	0h	RW	CH6 Error Interrupt Enable 1b: Enable 0b: Disable
26	CH6_QUE_EMP_EN	0h	RW	CH6 Que Empty Interrupt Enable 1b: Enable 0b: Disable
25	CH6_STOP_EN	0h	RW	CH6 Stop Interrupt Enable 1b: Enable 0b: Disable
24	CH6_END_EN	0h	RW	CH6 Completion Interrupt Enable 1b: Enable 0b: Disable
23	CH5_ERR_EN	0h	RW	CH5 Error Interrupt Enable 1b: Enable 0b: Disable
22	CH5_QUE_EMP_EN	0h	RW	CH5 Que Empty Interrupt Enable 1b: Enable 0b: Disable
21	CH5_STOP_EN	0h	RW	CH5 Stop Interrupt Enable 1b: Enable 0b: Disable
20	CH5_END_EN	0h	RW	CH5 Completion Interrupt Enable 1b: Enable 0b: Disable
19	CH4_ERR_EN	0h	RW	CH4 Error Interrupt Enable 1b: Enable 0b: Disable

Bit	Bit Name	Initial Value	R/W	Description
18	CH4_QUE_EMP_EN	0h	RW	CH4 Que Empty Interrupt Enable 1b: Enable 0b: Disable
17	CH4_STOP_EN	0h	RW	CH4 Stop Interrupt Enable 1b: Enable 0b: Disable
16	CH4_END_EN	0h	RW	CH4 Completion Interrupt Enable 1b: Enable 0b: Disable
15	CH3_ERR_EN	0h	RW	CH3 Error Interrupt Enable 1b: Enable 0b: Disable
14	CH3_QUE_EMP_EN	0h	RW	CH3 Que Empty Interrupt Enable 1b: Enable 0b: Disable
13	CH3_STOP_EN	0h	RW	CH3 Stop Interrupt Enable 1b: Enable 0b: Disable
12	CH3_END_EN	0h	RW	CH3 Completion Interrupt Enable 1b: Enable 0b: Disable
11	CH2_ERR_EN	0h	RW	CH2 Error Interrupt Enable 1b: Enable 0b: Disable
10	CH2_QUE_EMP_EN	0h	RW	CH2 Que Empty Interrupt Enable 1b: Enable 0b: Disable
9	CH2_STOP_EN	0h	RW	CH2 Stop Interrupt Enable 1b: Enable 0b: Disable
8	CH2_END_EN	0h	RW	CH2 Completion Interrupt Enable 1b: Enable 0b: Disable
7	CH1_ERR_EN	0h	RW	CH1 Error Interrupt Enable 1b: Enable 0b: Disable
6	CH1_QUE_EMP_EN	0h	RW	CH1 Que Empty Interrupt Enable 1b: Enable 0b: Disable
5	CH1_STOP_EN	0h	RW	CH1 Stop Interrupt Enable 1b: Enable 0b: Disable
4	CH1_END_EN	0h	RW	CH1 Completion Interrupt Enable 1b: Enable 0b: Disable
3	CH0_ERR_EN	0h	RW	CH0 Error Interrupt Enable 1b: Enable 0b: Disable
2	CH0_QUE_EMP_EN	0h	RW	CH0 Que Empty Interrupt Enable 1b: Enable 0b: Disable
1	CH0_STOP_EN	0h	RW	CH0 Stop Interrupt Enable 1b: Enable 0b: Disable
0	CH0_END_EN	0h	RW	CH0 Completion Interrupt Enable 1b: Enable 0b: Disable

(51) DMAC Interrupt Status Register (PCI_RC_DMAINTS)

This register indicates the state of interrupts from the individual DMA channels.

Access Size : 32 bits
Offset Address : <PCI0_base> + 080Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CH7_ERR	CH7_QUE_EMP	CH7_STOP	CH7_END	CH6_ERR	CH6_QUE_EMP	CH6_STOP	CH6_END	CH5_ERR	CH5_QUE_EMP	CH5_STOP	CH5_END	CH4_ERR	CH4_QUE_EMP	CH4_STOP	CH4_END
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH3_ERR	CH3_QUE_EMP	CH3_STOP	CH3_END	CH2_ERR	CH2_QUE_EMP	CH2_STOP	CH2_END	CH1_ERR	CH1_QUE_EMP	CH1_STOP	CH1_END	CH0_ERR	CH0_QUE_EMP	CH0_STOP	CH0_END
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31	CH7_ERR	0h	RW1	Set when an error occurs during DMA transfer.
30	CH7_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
29	CH7_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
28	CH7_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
27	CH6_ERR	0h	RW1	Set when an error occurs during DMA transfer.
26	CH6_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
25	CH6_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
24	CH6_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
23	CH5_ERR	0h	RW1	Set when an error occurs during DMA transfer.
22	CH5_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
21	CH5_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
20	CH5_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
19	CH4_ERR	0h	RW1	Set when an error occurs during DMA transfer.
18	CH4_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
17	CH4_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.

Bit	Bit Name	Initial Value	R/W	Description
16	CH4_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
15	CH3_ERR	0h	RW1	Set when an error occurs during DMA transfer.
14	CH3_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
13	CH3_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
12	CH3_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
11	CH2_ERR	0h	RW1	Set when an error occurs during DMA transfer.
10	CH2_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
9	CH2_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
8	CH2_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
7	CH1_ERR	0h	RW1	Set when an error occurs during DMA transfer.
6	CH1_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
5	CH1_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
4	CH1_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
3	CH0_ERR	0h	RW1	Set when an error occurs during DMA transfer.
2	CH0_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
1	CH0_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
0	CH0_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1

(52) DMAC Channel Control Register m (PCI_RC_DMACHCTLm) (m = 0 to 7)

This register sets the control method for each DMA channel. Set either register type or descriptor type.

- Setting QUE_EN = 1 and QUE_CLR = 1 is prohibited during register-type DMA transfer (RDMA_EN = 1).
- Setting RDMA_EN = 1 is prohibited during descriptor-type DMA transfer (QUE_EN = 1).

Access Size : 32 bits
Offset Address : <PCI0_base> + 0900h + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	QUE_CLR	-	-	-	-	-	-	QUE_EN	RDMA_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	QUE_CLR	0h	RW	QUE Clear Writing 1 clears QUE. All descriptor lists (waiting for execution and lists currently being executed) registered in QUE are cleared. Do not clear during DMA execution. The read value is always 0. Writing 1 at the same time as setting QUE_EN is prohibited.
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	QUE_EN	0h	RW	QUE Enable Setting this bit to 1 enables the descriptor list registered in the descriptor queue and starts DMA transfer (descriptor-type). It is automatically cleared to 0 when DMA stops (both normal and abnormal). It is also possible to stop DMA by writing 0. However, DMA will stop after finishing the currently executing request (both PCIe and AXI).
0	RDMA_EN	0h	RW	Register-type DMA transfer Enable Setting this bit to 1 starts DMA transfer (Register-type) and performs the transfer set by RDMA_SIZE. This bit is automatically cleared to 0 when DMA transfer is completed or an error is detected and DMA ends. It is also possible to stop DMA by writing 0. However, DMA will stop after completing the currently executing request (both PCIe and AXI). 1b: Register-type DMA transfer start 0b: Stop register-type DMA transfer

(53) Descriptor Start Address (Lower) Register m (PCI_RC_DPSADRL) (m = 0 to 7)

This register sets the descriptor queue list.

The setting value is registered as the lower 32 bits of DSA (DMA Start Address) in the queue list.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0908h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	QUE_ENTRY_DSA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	QUE_ENTRY_DSA[15:6]										QUE_ENTRY_DSA[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	QUE_ENTRY_DSA[31:6]	0h	RW	Descriptor list queue registration register. This area will be the DSA. Sets the lower 32 bits of the address where the first descriptor is stored.
5 to 0	QUE_ENTRY_DSA[5:0]	0h	R	Descriptor list queue registration register. This area will be the DSA. Sets the lower 32 bits of the address where the first descriptor is stored. (16-byte alignment: Lower 6 bits are fixed to 0.)

Note 1. When these bits are read, any of the following contents will be read depending on the state of DMA.
DMA transfer in progress: The descriptor list in progress
DMA suspended (when QUE_EN is automatically cleared): Last executed list
DMA suspended (when QUE_EN S/W is cleared): Suspended (executing) list

(54) Descriptor Start Address (Higher) Register m (PCI_RC_DPSADRU) (m = 0 to 7)

This register sets the descriptor queue list.

The setting value is registered as the higher 32 bits of DSA (DMA Start Address) in the queue list.

Access Size : 32 bits

Offset Address : <PCI0_base> + 090Ch + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	QUE_ENTRY_DSA[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	QUE_ENTRY_DSA[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	QUE_ENTRY_DSA[63:32]	0h	RW	Descriptor list queue registration register. This area will be the DSA. Set the upper 32 bits of the address where the first descriptor is stored.

Note 1. When these bits are read, any of the following contents will be read depending on the state of DMA.
DMA transfer in progress: The descriptor list in progress
DMA suspended (when QUE_EN is automatically cleared): Last executed list
DMA suspended (when QUE_EN S/W is cleared): Suspended (executing) list

(55) QUE Entry Register m (PCI_RC_QUEEm) (m = 0 to 7)

This register sets the descriptor queue list.

The setting value is registered as EI (End Interrupt), LS (List Stop), and LABEL of the queue list. Write to [31:24] to register to the queue.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0910h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	QUE_Registration[5:0]						QUE_ENTRY (EI)	QUE_ENTRY (LS)	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	QUE_ENTRY_LABEL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	QUE_Registration[5:0]	0h	RW	Enqueued by writing to [31:24]. The read value is always 0b.
25	QUE_ENTRY (EI)	0h	RW	Indicates whether an interrupt (Interrupt Status CHx_END) is sent when the processing of the descriptor list is completed. 1b: signal an interrupt 0b: Do not signal interrupts
24	QUE_ENTRY (LS)	0h	RW	Indicates whether to stop the DMA when processing of the descriptor list is complete. 1b: stop 0b: do not stop
23 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	QUE_ENTRY_LABEL[15:0]	0h	RW	There is no particular rule on how to set the labels in the list. You can set the value freely.

Note 1. When these bits are read, any of the following contents will be read depending on the state of DMA.
DMA transfer in progress: The descriptor list in progress
DMA suspended (when QUE_EN is automatically cleared): Last executed list
DMA suspended (when QUE_EN S/W is cleared): Suspended (executing) list

(56) DMA Descriptor Control (Descriptor 00h) Register m (PCI_RC_DMADPCTLm) (m = 0 to 7)

This register indicates the field value at offset 00h in the descriptor table. Only effective when the descriptor-type DMA transfer is selected (the value read has no meaning in the case of register-type transfer).

Access Size : 32 bits

Offset Address : <PCI0_base> + 0920h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSCFM[3:0]				-	WBD	LE	LV	D	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DSCFM[3:0]	0h	R	Shows the value of the DSCFM field in the running descriptor table.
27	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
26	WBD	0h	R	Shows the value of the WBD field in the running descriptor table.
25	LE	0h	R	Shows the value of the LE field in the running descriptor table.
24	LV	0h	R	Shows the value of the LV field in the running descriptor table.
23	D	0h	R	Shows the value of the D field in the running descriptor table.
22 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	STS[15:0]	0h	R	Shows the value of the STS field in the running descriptor table.

(57) DMA Transaction Control (Descriptor 04h) Register m (PCI_RC_DMATCTLm) (m = 0 to 7)

This register controls DMA transfer to the AXI and PCIe.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0924h + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	CCH_L[3:0]				CCH_D[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	DMA_TC[2:0]			-	-	DMA_ATB[1:0]		-	DMA_FUNC[2:0]			-	-	-	DMA_DIR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	R	R	RW	RW	R	RW	RW	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 20	CCH_L[3:0]	0h	RW	Indicates the value of A*CACHE[3:0] to be issued to AXI. CCH_L is output when issuing an AXI request containing the last byte in a transfer indicated by SIZE. Recommended value is 0000b. Bit[3] 1b: write allocatable 0b: not write allocatable Bit[2] 1b: read assignable 0b: not read assignable Bit[1] 1b: cacheable 0b: non-cacheable Bit[0] 1b: bufferable 0b: not bufferable
19 to 16	CCH_D[3:0]	0h	RW	Indicates the value of A*CACHE[3:0] to be issued to AXI. CCH_D is output when issuing an AXI request other than the output condition of CCH_L. The recommended value is 0001b when DIR=0 (PCIe→AXI) and 0000b when DIR=1 (AXI→PCIe). Bit[3] 1b: write allocatable 0b: not write allocatable Bit[2] 1b: read assignable 0b: not read assignable Bit[1] 1b: cacheable 0b: non-cacheable Bit[0] 1b: bufferable 0b: not bufferable
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 12	DMA_TC[2:0]	0h	RW	Traffic class to issue to PCIe. Specifies the traffic class value for requests issued to PCIe. Note: This IP does not support Virtual Channel, so please use it with 000b fixed.
11, 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9, 8	DMA_ATB[1:0]	0h	RW	Attributes to issue to PCIe. Bit[1]: Relaxed Ordering (Unsupported function: 0b fixed) Bit[0]: No Snoop (0b recommended)
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 4	DMA_FUNC [2:0]	0h	RW	Specify the function number of the request issued to PCIe.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	DMA_DIR	0h	RW	Sets the DMA transfer direction. 1b: AXI to PCIe 0b: PCIe to AXI

Note: When descriptor-type DMA transfer is in process, writing to these bits is prohibited and the value read indicates the value of the CCH_L, CCH_D, TC, ATB, and DIR fields in the descriptor table being executed.

(58) DMA Size (Descriptor 08h) Register m (PCI_RC_DMASIZEm) (m = 0 to 7)

This register sets the number of bytes for DMA transfer. The set value is reflected in the offset 0Ch field of the descriptor table.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0928h + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_SIZE[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_SIZE[15:3]													DMA_SIZE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DMA_SIZE [31:3]	0h	RW	Sets the number of DMA transfer bytes.
2 to 0	DMA_SIZE[2:0]	0h	R	Sets the number of DMA transfer bytes. The lower 3 bits are fixed to 0 because it is an 8-byte aligned setting.

Note: When descriptor-type DMA transfer is in process, writing to these bits is prohibited and the value read indicates the value of the SIZE field in the descriptor table being executed. The number of bytes for transfer when the setting is 0000_0000h is 1_0000_0000h.

(59) DMA Source Lower Address (Descriptor 10h) Register m (PCI_RC_DMASLAm) (m = 0 to 7)

This register sets the lower 32 bits of the source start address for DMA transfer. The set value is reflected in the offset 10h field of the descriptor table.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0930h + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_S_ADDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_S_ADDR[31:3]													DMA_S_ADDR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DMA_S_ADDR [31:3]	0h	RW	Set the lower 32 bits of the transfer source start address for DMA transfer. The lower 3 bits are fixed to 0 because it is an 8-byte aligned setting. (When the data bus width is 128 bits, 16-byte alignment: lower 4 bits are fixed to 0.)
2 to 0	DMA_S_ADDR [2:0]	0h	R	Set the lower 32 bits of the transfer source start address for DMA transfer. The lower 3 bits are fixed to 0 because it is an 8-byte aligned setting. (When the data bus width is 128 bits, 16-byte alignment: lower 4 bits are fixed to 0.)

Note 1. If the Source Address indicates the PCIe space (DIR=0), the PCIe request address is set in combination with the DMA PCIe Higher Address (Descriptor 10h). When descriptor-type DMA transfer is in process, writing to these bits is prohibited and the value read indicates the value of the SA field in the descriptor table being executed.

(60) DMA Source Higher Address (Descriptor 14h) Register m (PCI_RC_DMASUAm) (m = 0 to 7)

This register sets the higher 32 bits of the source start address for DMA transfer. The set value is reflected in the offset 14h field of the descriptor table.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0934h + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_S_ADDR[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_S_ADDR[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_S_ADDR [63:32]	0h	RW	Set the upper 32 bits of the transfer source start address for DMA transfer.

Note 1. If the Source Address indicates the PCIe space (DIR=0), the PCIe request address is set in combination with the DMA PCIe Higher Address (Descriptor 10h). When descriptor-type DMA transfer is in process, writing to these bits is prohibited and the value read indicates the value of the SA field in the descriptor table being executed.

(61) DMA Destination Lower Address (Descriptor 18h) Register m (PCI_RC_DMADESTLAm) (m = 0 to 7)

This register sets the lower 32 bits of the source start address for DMA transfer. The set value is reflected in the offset 18h field of the descriptor table.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0938h + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_D_ADDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_D_ADDR[15:3]													DMA_D_ADDR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DMA_D_ADDR [31:3]	0h	RW	Set the lower 32 bits of the transfer destination start address for DMA transfer. The lower 3 bits are fixed to 0 because it is an 8-byte aligned setting.
2 to 0	DMA_D_ADDR [2:0]	0h	R	Set the lower 32 bits of the transfer destination start address for DMA transfer. The lower 3 bits are fixed to 0 because it is an 8-byte aligned setting.

Note 1. If the Destination Address indicates the PCIe space (DIR=1), the PCIe request address is set in combination with the DMA PCIe Higher Address (Descriptor 10h). When descriptor-type DMA transfer is in process, writing to these bits is prohibited and the value read indicates the value of the DA field in the descriptor table being executed.

(62) DMA Destination Higher Address (Descriptor 1Ch) Register m (PCI_RC_DMADESTUAm) (m = 0 to 7)

This register sets the higher 32 bits of the source start address for DMA transfer. The set value is reflected in the offset 1Ch field of the descriptor table.

Access Size : 32 bits

Offset Address : <PCI0_base> + 093Ch + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_D_ADDR[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_D_ADDR[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_D_ADDR [63:32]	0h	RW	Set the upper 32 bits of the transfer destination start address for DMA transfer.

Note 1. If the Destination Address indicates the PCIe space (DIR=1), the PCIe request address is set in combination with the DMA PCIe Higher Address (Descriptor 10h). When descriptor-type DMA transfer is in process, writing to these bits is prohibited and the value read indicates the value of the DA field in the descriptor table being executed.

(63) DMA Descriptor Lower Link Pointer (Descriptor 20h) Register m (PCI_RC_DMADPLLPm) (m = 0 to 7)

This register indicates the field value at offset 20h in the descriptor table. Only effective when the descriptor-type DMA transfer is selected (the value read has no meaning in the case of register-type transfer).

Access Size : 32 bits
Offset Address : <PCI0_base> + 0940h + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_LP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_LP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_LP[31:0]	0h	R	Indicates the value of the LP field of the running descriptor table.

(64) DMA Descriptor Higher Link Pointer (Descriptor 24h) Register m (PCI_RC_DMADPULPm) (m = 0 to 7)

This register indicates the field value at offset 24h in the descriptor table. Only effective when the descriptor-type DMA transfer is selected (the value read has no meaning in the case of register-type transfer).

Access Size : 32 bits

Offset Address : <PCI0_base> + 0944h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_LP[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_LP[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_LP [63:32]	0h	R	Indicates the value of the LP field of the running descriptor table.

(65) DMA Rest Size Register m (PCI_RC_DMARESTSIZEm) (m = 0 to 7)

This register indicates the number of bytes for which DMA transfer has not yet been completed.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0950h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_REST_SIZE[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_REST_SIZE[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_REST_SIZE[31:0]	0h	R	Displays the number of bytes for which DMA transfer has not been completed. (Register/descriptor-type common)

(66) AXI Request Address (Lower) Register m (PCI_RC_AREQALm) (m = 0 to 7)

This register indicates the lower 32 bits of the address of the current or most recently completed AXI transfer.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0960h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AXI_REQ_ADDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AXI_REQ_ADDR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	AXI_REQ_ADDR[31:0]	0h	R	Displays the lower 32 bits of the address of the current or most recently completed AXI transfer. (Register/descriptor-type common)

(67) AXI Request Address (Higher) Register m (PCI_RC_AREQAUm) (m = 0 to 7)

This register indicates the higher 32 bits of the address of the current or most recently completed AXI transfer.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0964h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AXI_REQ_ADDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AXI_REQ_ADDR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	AXI_REQ_ADDR[31:0]	0h	R	Displays the upper 32 bits of the address of the current or most recently completed AXI transfer. (Register/descriptor-type common)

(68) PCIe Request Address (Lower) Register m (PCI_RC_PREQALm) (m = 0 to 7)

This register indicates the lower 32 bits of the address during PCIe transfer that is in progress or for the most recently completed transfer.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0968h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PCIE_REQ_ADDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCIE_REQ_ADDR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PCIE_REQ_ADDR[31:0]	0h	R	Displays the lower 32 bits of the address of the current or most recently completed PCIe transfer. (Register/descriptor-type common)

(69) PCIe Request Address (Higher) Register m (PCI_RC_PREQAUm) (m = 0 to 7)

This register indicates the higher 32 bits of the address during PCIe transfer that is in progress or for the most recently completed transfer.

Access Size : 32 bits
Offset Address : <PCI0_base> + 096Ch + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PCIE_REQ_ADDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCIE_REQ_ADDR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PCIE_REQ_ADDR[31:0]	0h	R	Displays the upper 32 bits of the address of the current or most recently completed PCIe transfer. (Register/descriptor-type common)

(70) QUE Status Register m (PCI_RC_QUESTAm) (m = 0 to 7)

This register indicates the state of the descriptor queue.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0970h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	GO_LIST	LIST_NUM[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	GO_LIST	0h	R	Shows whether there is a running descriptor list. 1b: Yes 0b: No
3 to 0	LIST_NUM[3:0]	0h	R	Displays the number of descriptor lists loaded on QUE (not including the list currently being executed). New registration to QUE (write access to QUE Entry) when this register indicates 8h is invalid (discarded).

(71) DMAC Error Status Register m (PCI_RC_DMACESTAm) (m = 0 to 7)

This register indicates the error status of the DMAC.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0978h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	CFG_BM_DIS_EP	BME_SUP	BME_DOWN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	MOR_CD_PERR	MOR_CH_PERR	MOR_EP_ERR	MOR_STATUS[2:0]			-	-	-	-	-	-	AXI_RESP[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

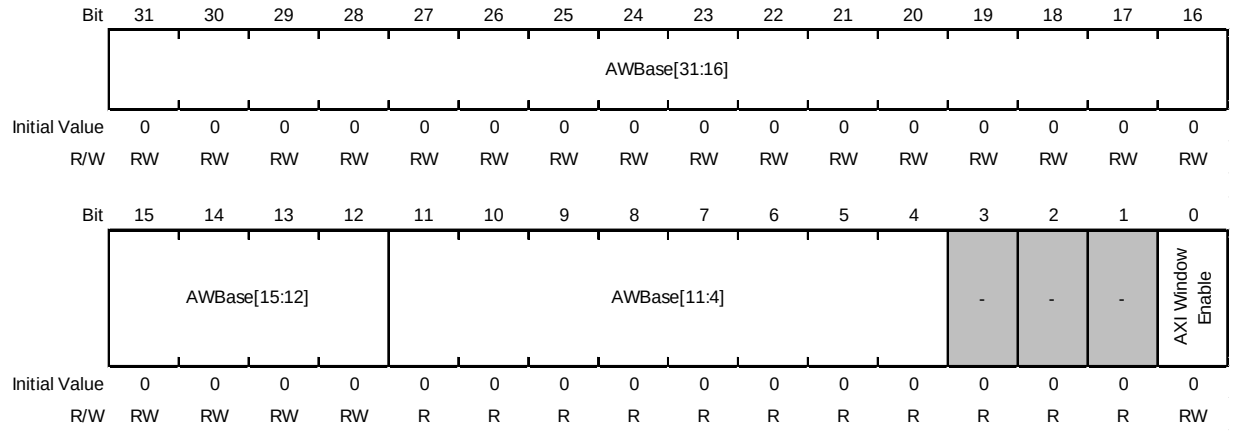
Bit	Bit Name	Initial Value	R/W	Description
31 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18	CFG_BM_DIS_EP	0h	R	CHx_ERR is set when the macro operates as an End Point and detects the state of Bus Master Enable Off (Configuration Space 004h bit [2]=0). Holds value until CHx_ERR is cleared.
17	BME_SUP	0h	R	It is set when a sleep signal from the PCIe core is detected as a cause for setting CHx_ERR. Holds value until CHx_ERR is cleared. (Valid only in Endpoint mode)
16	BME_DOWN	0h	R	It is set when a stop signal from the PCIe core is detected as a cause for setting CHx_ERR. Holds value until CHx_ERR is cleared.
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	MOR_CD_PERR	0h	R	It is set when MOR_CD_PERR is detected as a set factor for CHx_ERR. Holds value until CHx_ERR is cleared.
12	MOR_CH_PERR	0h	R	It is set when MOR_CH_PERR is detected as a set factor for CHx_ERR. Holds value until CHx_ERR is cleared.
11	MOR_EP_ERR	0h	R	It is set when it is Poisoned Completion as a set factor for CHx_ERR. Holds value until CHx_ERR is cleared.
10 to 8	MOR_STATUS[2:0]	0h	R	Indicates the value when MOR_STATUS is other than 000b (Success) as a set factor for CHx_ERR. Holds value until CHx_ERR is cleared. 000b: initial value 001b: Unsupported Request 010b: CRS 011b: Completion Timeout 100b: Completer Abort 101b: Unexpected Completion 110b: Reserved 111b: Mismatched Length (Length Overrun)
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	AXI_RESP[1:0]	0h	R	Displays the slave response during AXI Master transactions. It is updated when CHx_ERR is set and retains the value until the bit is cleared. 00b: Initial value 01b: Reserved 10b: SLVERR 11b: DECERR

(72) AXI Window Base (Lower) Register m (PCI_RC_AWBASELm) (m = 0 to 7)

This register is for setting windows for lower address conversion in access from PCIe to AXI. It sets the base address on the PCI. The areas are set in 4-Kbyte boundaries.

Access Size : 32 bits
Offset Address : <PCI0_base> + 1000h + Offset: 00h/ 20h/ 40h/ ・・・ / E0h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 12	AWBase[31:12]	0h	RW	Window setting register for address conversion when accessing AXI from PCIe. The configurable area is the 4K boundary.
11 to 4	AWBase[11:4]	0h	R	Fixed to 00h
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	AXI Window Enable	0h	RW	Enable setting of AXI Window. 0b: Window disabled 1b: Window enabled

(73) AXI Window Base (Higher) Register m (PCI_RC_AWBASEUm) (m = 0 to 7)

This register is for setting windows for higher address conversion in access from PCIe to AXI.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000h + Offset: 04h/ 24h/ 44h/ ・・・ / E4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AWBase[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AWBase[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	AWBase[63:32]	0h	RW	Window setting register for address conversion when accessing AXI from PCIe.

(74) AXI Window Mask (Lower) Register m (PCI_RC_AWMASKLm) (m = 0 to 7)

This register is for setting windows for lower address conversion in access from PCIe to AXI. The window is set as the area corresponding to the number of set bits from the address set in the AWBase register. The area which can be set is $4K \times 2^N$ bytes.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000h + Offset: 08h/ 28h/ 48h/ ・・・ / E8h

Initial Value : 0000_0FFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AWMask[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AWMask[15:12]				AWMask[11:0]											
Initial Value	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	AWMask[31:12]	0h	RW	Set the window in the area of the set number of bits from the address set in the AWBase register. Set 1 from the lower bit. Therefore, the area that can be set is 4K x 2^N bytes.
11 to 0	AWMask[11:0]	FFFh	R	Fixed to FFFh

(75) AXI Window Mask (Higher) Register m (PCI_RC_AWMASKUm) (m = 0 to 7)

This register is for setting windows for higher address conversion in access from PCIe to AXI.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000h + Offset: 0Ch/ 2Ch/ 4Ch/ ・・・ / Ech

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	AWMask[62:48]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AWMask[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 0	AWMask[62:32]	0h	RW	Set the window in the area of the set number of bits from the address set in the AWBase register. Set 1 from the lower bit.

(76) AXI Destination (Lower) Register m (PCI_RC_ADESTLm) (m = 0 to 7)

This register is for setting windows for lower address conversion in access from PCIe to AXI. The base address of the window in the address space on the AXI is set. The areas are set in 4-Kbyte boundaries.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000 + Offset: 10h/ 30h/ 50h/ 70h/ ・・・ / F0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADest[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADest[15:12]				Adest[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	ADest[31:12]	0h	RW	Sets the window base point in the address space from the AXI side. The configurable area is the 4K boundary.
11 to 0	Adest[11:0]	0h	R	Fixed to 000h

(77) AXI Destination (Higher) Register m (PCI_RC_ADESTUm) (m = 0 to 7)

This register is for setting windows for higher address conversion in access from PCIe to AXI.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000 + Offset: 14h/ 34h/ 54h/ 74h/ ・・・ / F4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADest[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADest[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ADest[63:32]	0h	RW	Sets the window base point in the address space from the AXI side.

(78) PCIe Window Base (Lower) Register m (PCI_RC_PWBASELm) (m = 0 to 7)

This register is for setting windows for lower address conversion in access from AXI to PCIe. It sets the base address on the AXI. The areas are set in 4-Kbyte boundaries.

Access Size : 32 bits
Offset Address : <PCI0_base> + 1000 + Offset: 100h/ 120h/ 140h/ ・・・ / 1E0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PWBase[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PWBase[15:12]				PWBase[11:4]								-	-	-	PCIe Window Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	PWBase[31:12]	0h	RW	Sets the base point of the address on the AXI side. The configurable area is the 4K boundary.
11 to 4	PWBase[11:4]	0h	R	Fixed to 00h
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	PCIe Window Enable	0h	RW	Enable setting of PCIe Window. 0b: Window disabled 1b: Window enabled

(79) PCIe Window Base (Higher) Register m (PCI_RC_PWBASEUm) (m = 0 to 7)

This register is for setting windows for higher address conversion in access from AXI to PCIe.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000 + Offset: 104h/ 124h/ 144h/ ・・・ / 1E4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PWBase[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PWBase[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PWBase[63:32]	0h	RW	Sets the base point of the address on the AXI side. The configurable area is the 4K boundary.

(80) PCIe Window Mask (Lower) Register m (PCI_RC_PWMASKLm) (m = 0 to 7)

This register is for setting windows for address conversion in access from AXI to PCIe. The window is set as the area corresponding to the number of set bits from the address set in the PWBase register.

Access Size : 32 bits
Offset Address : <PCI0_base> + 1000 + Offset: 108h/ 128h/ 148h/ ・・・ / 1E8h
Initial Value : 0000_0FFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PWMask[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PWMask[15:12]				PWMask[11:0]											
Initial Value	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	PWMask[31:12]	0h	RW	Set the window in the area of the set number of bits from the address set in the PWBase register. Set 1 from the lower bit.
11 to 0	PWMask[11:0]	FFFh	R	Fixed to FFFh

(81) PCIe Window Mask (Higher) Register m (PCI_RC_PWMASKUm) (m = 0 to 7)

This register is for setting windows for address conversion in access from AXI to PCIe. The window is set as the area corresponding to the number of set bits from the address set in the PWBase register.

Access Size : 32 bits
Offset Address : <PCI0_base> + 1000 + Offset: 10Ch/ 12Ch/ 14Ch/ ・・・ / 1ECh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	PWMask[62:48]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PWMask[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 0	PWMask[62:32]	0h	RW	Set the window in the area of the set number of bits from the address set in the PWBase register. Set 1 from the lower bit.

(82) PCIe Destination (Lower) Register m (PCI_RC_PDESTLOm) (m = 0 to 7)

This register is for setting windows for address conversion in access from AXI to PCIe. The base address of the window in the address space on the AXI is set. The areas are set in 4-Kbyte boundaries.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000 + Offset: 110h/ 130h/ 150h/ 170h/ ・・・ / 1F0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PDest[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PDest[15:12]				Pdest[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	PDest[31:12]	0h	RW	Set the window base point in the address space on the PCIe side. The configurable area is the 4k boundary.
11 to 0	Pdest[11:0]	0h	R	Fixed to 000h

(83) PCIe Destination (Higher) Register m (PCI_RC_PDESTUPm) (m = 0 to 7)

This register is for setting windows for higher address conversion in access from AXI to PCIe.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000 + Offset: 114h/ 134h/ 154h/ 174h/ ・・・ / 1F4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PDest[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PDest[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PDest[63:32]	0h	RW	Set the window base point in the address space on the PCIe side.

6.6.4.1.4 PCI Express Configuration Register Descriptions (Type1)

(1) Vendor and Device ID Register (PCI_RC_VID)

This register indicates the vendor and device ID. This register can be written during initialization.

Refer to **6.6.6.1.1 Changing the Initial Values of the Registers**.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Device ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Vendor ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Device ID[15:0]	0h	Hwinit	Used by the manufacturer specified by the Vendor ID to identify the manufactured device. Set a fixed value.
15 to 0	Vendor ID[15:0]	0h	Hwinit	Represents the manufacturer of the device. Set a fixed value.

Table 6.6-10 Valid Reset Signal

Reset Signal	Device ID	Vendor ID
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		

(2) Command and Status Register (PCI_RC_COM_STA)

This register specifies the command and the status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6004h
Initial Value : 0010_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	Signaled System Error	-	-	-	-	-	-	-	-	-	Capabilities List	Interrupt Status	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	RW1	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	Interrupt Disable	-	SERR# Enable	-	Parity Error Response	-	-	-	Bus Master Enable	Memory Space Enable	IO Space Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	R	RW	R	RW	R	R	R	RW	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30	Signaled System Error	0h	RW1	Set to 1b when the SERR Enable bit is 1b and this macro sends an ERR_FATAL or ERR_NONFATAL Message.
29 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	Capabilities List	1h	R	This bit is fixed to 1b because all PCI Express devices must implement PCI Express Capability.
19	Interrupt Status	0h	R	Indicates the interrupt of the device.
18 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	Interrupt Disable	0h	RW	Suppress transmission of Assert_INTx Message. Not used for Root Complex devices.
9	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	SERR# Enable	0h	RW	If set to 1b, the Root Complex is notified of Non-Fatal Errors and Fatal Errors by Message Transaction. Note: Even if this bit is not set, if a bit related to Error Reporting in the Device Control register of PCI Express Capability is set to “1”, an error is notified to the Root Complex by Message Transaction.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6	Parity Error Response	0h	RW	Controls the operation when transferring/receiving Poised TLP. Note: Error logging to the Detected Parity Error field of the Status register, the Device Status register of PCI Express Capability, and the Uncorrectable Error Status register of Advanced Error Reporting Capability is performed regardless of the setting of this bit.
5 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	Bus Master Enable	0h	RW	Controls whether to operate as a bus master.
1	Memory Space Enable	0h	RW	Controls whether to respond to accesses to memory space.
0	IO Space Enable	0h	R	0b fixed. It does not support access to the I/O space.

Table 6.6-11 Valid Reset Signal

Reset Signal	Signaled System Error	Capabilities List	Interrupt Status	Interrupt Disable	SERR# Enable	Parity Error Response
RST_LOAD_B						
RST_RSM_B						
RST_CFG_B	✓	✓	✓	✓	✓	✓

Reset Signal	Bus Master Enable	Memory Space Enable
RST_LOAD_B		
RST_RSM_B		
RST_CFG_B	✓	✓

(3) Revision ID and Class Code Register (PCI_RC_RID_CC)

This register indicates the revision ID and the class code.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Class Code[23:8]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Class Code[7:0]								Revision ID[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	Class Code[23:0]	0h	Hwinit	This is information that indicates the device type and function, and the definition of the value is divided into the following two bytes by the PCI SIG. Bit[31:24] base class Bit[23:16] sub-class Bit[15:8] programming interface Please set a fixed value.
7 to 0	Revision ID[7:0]	0h	Hwinit	An 8-bit ID used to represent the revision of a specific device specified by Vendor ID and Device ID. Please set a fixed value.

Table 6.6-12 Valid Reset Signal

Reset Signal	Class Code	Revision ID
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		

(4) Cache Line and Header Type Register (PCI_RC_CL_HT)

This register indicates the cache line and the header type.

Access Size : 32 bits
Offset Address : <PCI0_base> + 600Ch
Initial Value : 0001_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	Header Type[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	Cache Line Size[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	Header Type[7:0]	1h	R	Root Complex: 01h fixed.
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	Cache Line Size[7:0]	0h	R	Implemented as a read/write field for legacy compatibility, but the value set has no effect on this device.

Table 6.6-13 Valid Reset Signal

Reset Signal	Cache Line Size
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓

(5) Base Address Register 0 (PCI_RC_BAR0)

This register forms a 64-bit memory space in combination with Base Address Register 1 (BAR1).

Access Size : 32 bits
Offset Address : <PCI0_base> + 6010h
Initial Value : 0000_0004h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Base Address Register 0[27:12]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Base Address Register 0[11:0]												Prefetch	Type	-	Memory Space Indicator
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	Hwinit	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	Base Address Register 0[27:0]	0h	RW	Indicates the base address. Depending on the size of the address block required, some lower bits of this field are implemented as Read Only bits fixed at 0b. In this macro, the Read Only bits can be set with the Base Address Register Mask00 (Lower)(Offset: 0A0h).
3	Prefetch	0h	Hwinit	0b: disable, 1b: enable
2	Type	1h	R	0b: 32 bit address, 1b: 64-bit address 1b fixed to use 64-bit address
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	Memory Space Indicator	0h	R	Fixed to 0b to indicate the memory space.

Table 6.6-14 Valid Reset Signal

Reset Signal	Base Address Register 0	Prefetch
RST_LOAD_B		✓
RST_RSM_B		
RST_CFG_B	✓	

(6) Base Address Register 1 (PCI_RC_BAR1)

This register forms a 64-bit memory space in combination with Base Address Register 0 (BAR0).

Access Size : 32 bits
Offset Address : <PCI0_base> + 6014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Base Address Register 1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Base Address Register 1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Base Address Register 1[31:0]	0h	RW	Base Address Register 1 (64-bit Upper Address) Shows the upper 32 bits of the base address. In this macro, the Read Only bits can be set with the Base Address Register Mask00 (Higher)(Offset: 0A4h).

Table 6.6-15 Valid Reset Signal

Reset Signal	Base Address Register 1
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓

(7) Bus Number Register (PCI_RC_BNR)

Access Size : 32 bits
Offset Address : <PCI0_base> + 6018h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	Subordinate Bus Number[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Secondary Bus Number[7:0]								Primary Bus Number[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	Subordinate Bus Number[7:0]	0h	RW	Set the Subordinate Bus Number.
15 to 8	Secondary Bus Number[7:0]	0h	RW	Set the Secondary Bus Number.
7 to 0	Primary Bus Number[7:0]	0h	RW	Set the Primary Bus Number.

Table 6.6-16 Valid Reset Signal

Reset Signal	Subordinate Bus Number	Secondary Bus Number	Primary Bus Number
RST_LOAD_B			
RST_RSM_B			
RST_CFG_B	✓	✓	✓

(8) I/O Base/Limit and Secondary Status (PCI_RC_IOBL_SS)

Access Size : 32 bits
Offset Address : <PCI0_base> + 601Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Detected Parity Error	Received System Error	Received Master Abort	Received Target Abort	Signaled Target Abort	-	-	Master Data Parity Error	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	RW1	RW1	RW1	R	R	RW1	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IO Limit[7:0]								IO Base[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	Detected Parity Error	0h	RW1	Set to 1b when a Poisoned TLP is received, regardless of the setting of the Parity Error Response Enable bit in the Bridge Control and Interrupt Register (Offset:03Ch).
30	Received System Error	0h	RW1	Set to 1b when an ERR_FATAL or ERR_NONFATAL Message is received.
29	Received Master Abort	0h	RW1	Set to 1b when the Completion Status field receives a Completion of an Unsupported Request.
28	Received Target Abort	0h	RW1	Set to 1b when the Completion Status field receives a Completion for Completer Abort.
27	Signaled Target Abort	0h	RW1	The Completion Status field is set to 1b when sending a Completion of Completer Abort (Posted or Non-Posted Request).
26, 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	Master Data Parity Error	0h	RW1	Set to 1b when the Parity Error Response bit is set to 1b and the following two conditions occur: 1) Requester (BME) received a Poisoned Completion TLP. 2) Requester (BME) sent Poisoned Write Request TLP. This bit is not set to 1b if the Parity Error Response bit is 0b.
23 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 8	IO Limit[7:0]	0h	RW (R)	This is an unused field in this macro. (00h fixed)
7 to 0	IO Base[7:0]	0h	RW (R)	This is an unused field in this macro. (00h fixed)

Table 6.6-17 Valid Reset Signal

Reset Signal	Detected Parity Error	Received System Error	Received Master Abort	Received Target Abort	Signaled Target Abort	Master Data Parity Error
RST_LOAD_B						
RST_RSM_B						
RST_CFG_B	✓	✓	✓	✓	✓	✓

Reset Signal	I/O Limit	I/O Base
RST_LOAD_B		
RST_RSM_B		
RST_CFG_B	✓	✓

(9) Memory Base/Limit Register (PCI_RC_MEMBL)

Access Size : 32 bits
Offset Address : <PCI0_base> + 6020h
Initial Value : FFF0_FFF0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Memory Limit[11:0]												-	-	-	-
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Memory Base[11:0]												-	-	-	-
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	Memory Limit[11:0]	FFFh	RW	Initial value FFF0h
19 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 4	Memory Base[11:0]	FFFh	RW	Initial value FFF0h
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Table 6.6-18 Valid Reset Signal

Reset Signal	Memory Limit	Memory Base
RST_LOAD_B		
RST_RSM_B		
RST_CFG_B	✓	✓

(10) Prefetchable Memory Base/Limit Register (PCI_RC_PMBL)

Access Size : 32 bits
Offset Address : <PCI0_base> + 6024h
Initial Value : 0001_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Prefetchable Memory Limit[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Prefetchable Memory Base[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Prefetchable Memory Limit[15:0]	1h	RW (R)	Initial value 0001h
15 to 0	Prefetchable Memory Base[15:0]	1h	RW (R)	Initial value 0001h

Table 6.6-19 Valid Reset Signal

Reset Signal	Prefetchable Memory Limit	Prefetchable Memory Base
RST_LOAD_B		
RST_RSM_B		
RST_CFG_B	✓	✓

(11) Prefetchable Base Higher 32-Bits Register (PCI_RC_PBUP32)

Access Size : 32 bits
Offset Address : <PCI0_base> + 6028h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Prefetchable Base Higher 32bits[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Prefetchable Base Higher 32bits[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Prefetchable Base Higher 32bits[31:0]	0h	RW	Initial value 0000_0000h

Table 6.6-20 Valid Reset Signal

Reset Signal	Prefetchable Base Higher 32bits
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓

(12) Prefetchable Limit Higher 32-Bit Register (PCI_RC_PLUP32)

Access Size : 32 bits
Offset Address : <PCI0_base> + 602Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Prefetchable Limit Higher 32bits[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Prefetchable Limit Higher 32bits[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Prefetchable Limit Higher 32bits[31:0]	0h	RW	Initial value 0000_0000h

Table 6.6-21 Valid Reset Signal

Reset Signal	Prefetchable Limit Higher 32bits
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓

(13) I/O Base/Limit Higher 16-Bit Register (PCI_RC_IOBLUP16)

This register forms a 64-bit memory space in combination with Base Address Register 0 (BAR0).

Access Size : 32 bits
Offset Address : <PCI0_base> + 6030h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IO Limit Higher 16bits[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IO Base Higher 16bits[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	IO Limit Higher 16bits[15:0]	0h	RW	This is an unused field in this macro.
15 to 0	IO Base Higher 16bits[15:0]	0h	RW	This is an unused field in this macro.

Table 6.6-22 Valid Reset Signal

Reset Signal	I/O Limit Higher 16bits	I/O Base Higher 16bits
RST_LOAD_B		
RST_RSM_B		
RST_CFG_B	✓	✓

(14) Capabilities Pointer Register (PCI_RC_CP)

This register indicates the I/O base, limit, and secondary status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6034h
Initial Value : 0000_0040h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	Capability Pointer[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	Capability Pointer[7:0]	40h	Hwinit (R)	Capability implementation start address 40h PCI Power Management Capability is implemented from 40h. The lower 2 bits are fixed to 00b (reserved) and cannot be written from the UDL side.

Table 6.6-23 Valid Reset Signal

Reset Signal	Capability Pointer [7:2]
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	

(15) Expansion ROM Base Address (PCI_RC_EROMBA)

This register will not be used in Root Complex mode.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6038h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Expansion ROM Base Address[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Expansion ROM Base Address[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Expansion ROM Base Address[31:0]	0h	R	Reserved Doesn't apply to PCI Express.

(16) Bridge Control and Interrupt (PCI_RC_BC_INT)

Access Size : 32 bits
Offset Address : <PCI0_base> + 603Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	Secondary Bus Reset	-	VGA 16-bit Decode	VGA Enable	ISA Enable	SERR# Enable	Parity Error Response Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Interrupt Pin[7:0]								Interrupt Line[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	Secondary Bus Reset	0h	RW	1b Write to Hot Reset state
21	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	VGA 16-bit Decode	0h	RW	Initial value 0b Not used in this macro.
19	VGA Enable	0h	RW	Initial value 0b Not used in this macro.
18	ISA Enable	0h	RW	Initial value 0b Not used in this macro.
17	SERR# Enable	0h	RW	1b write enables INT_SERR* interrupt pin notification
16	Parity Error Response Enable	0h	RW	Writing 1b sets the Master Data Parity Error bit in the Secondary Status register when a Poisoned TLP is received.
15 to 8	Interrupt Pin[7:0]	0h	Hwinit	Fixed to 00h.
7 to 0	Interrupt Line[7:0]	0h	RW	Fixed to 00h.

Table 6.6-24 Valid Reset Signal

Reset Signal	Secondary Bus Reset	VGA 16-bit Decode	VGA Enable	ISA Enable	SERR# Enable	Parity Error Response Enable
RST_LOAD_B						
RST_RSM_B						
RST_CFG_B	✓	✓	✓	✓	✓	✓

Reset Signal	Interrupt Pin	Interrupt Line
RST_LOAD_B	✓	
RST_RSM_B		
RST_CFG_B		✓

(17) PM Capabilities (PCI_RC_PMC)

This register indicates various support information.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6040h
Initial Value : 4803_6001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PME Support[4:0]					D2 Support	D1 Support	AUX_Current[2:0]			Device Specific Initialization (DSI)	Immediate_Readiness_on_Return_to_D0	-	Version[2:0]		
Initial Value	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1	1
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	R	R	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Next Capability Pointer[7:0]								Capability ID[7:0]							
Initial Value	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	PME Support[4:0]	9h	Hwinit	Indicates whether PME is supported in each Device State. xxxx1b: supports D0 xxx1xb: supports D1 xx1xxb: supports D2 x1xxxb: supports D3hot 1xxxxb: D3cold supported (not supported)
26	D2 Support	0h	Hwinit	Indicates whether D2 Power Management State is supported. 0b: not supported 1b: Support
25	D1 Support	0h	Hwinit	Indicates whether the D1 Power Management State is supported. 0b: not supported 1b: Support
24 to 22	AUX_Current [2:0]	0h	Hwinit	Indicates the 3.3-V auxiliary current (the maximum current value supplied from the auxiliary power supply). 111b: 375 mA 110b: 320 mA 101b: 250 mA 100b: 220 mA 011b: 160 mA 010b: 100 mA 001b: 55 mA 000b: 0 (self powered) Reading returns 000b. (AUX not supported)
21	Device Specific Initialization (DSI)	0h	Hwinit	Indicates whether or not DSI (Device Specific Initialization) is used. 0b: not supported 1b: Support
20	Immediate_Readiness_on_Return_to_D0	0h	R	0b fixed
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	Version[2:0]	3h	Hwinit	011b is fixed. PCI Power Management Interface Specification Rev.1.2
15 to 8	Next Capability Pointer[7:0]	60h	R	Indicates the PCI Express Capability start address.
7 to 0	Capability ID[7:0]	1h	R	Indicates the PCI Power Management Capability. It is fixed at 01h.

Table 6.6-25 Valid Reset Signal

Reset Signal	PME Support	D2 Support	D1 Support	AUX_Current	DSI	Version
RST_LOAD_B	✓	✓	✓	✓	✓	✓
RST_RSM_B						
RST_CFG_B						

Reset Signal	Next Capability Pointer [7:2]
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	

(18) PM Status/Control Register (PCI_RC_PMSC)

This register indicates and controls the PME status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6044h
Initial Value : 0000_0008h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PME Status	-	-	-	-	-	-	PME Enable	-	-	-	-	No_Soft_Reset	-	PowerState [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	RW	R	R	R	R	R	R	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15	PME Status	0h	RW	Indicates that a PME assert factor has occurred. 1b indicates that there is a PME assertion factor.
14 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	PME Enable	0h	RW	Controls PME assertions. If 1b, PME assertion is enabled. Assert PME if PME_Status is set at this time. PCI Express performs Link Wake-up processing, and then performs PME assert processing by sending PM_PME Message. Note: The specifications are as follows depending on the value of the PM Capabilities register PME Support[4] (PME Support in D3cold). PME Support[4]=1b Reset: RST_RSM_B PME Support[4]=0b Reset: RST_CFG_B
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	No_Soft_Reset	1h	R	Indicates that internal reset is not performed inside the device at the power state transition from D3hot to D0.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	PowerState [1:0]	0h	RW	Set the PCI Device State. 00b: - D0 (Default) 01b: - D1 (Do not set) 10b: - D2 (Do not set) 11b: - D3hot

Table 6.6-26 Valid Reset Signal

Reset Signal	PME Enable	PowerState
RST_LOAD_B		
RST_RSM_B	✓	
RST_CFG_B		✓

(19) PCI Express Capability Register (PCI_RC_PCIEC)

This register indicates the PCIe capability.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6060h
Initial Value : 0042_0010h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	Interrupt Message Number[4:0]					Slot Implemented	DevicePort Type[3:0]				Capability Version[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	Hwinit	R	R	R	R	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Next Capability Pointer[7:0]								Capability ID[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29 to 25	Interrupt Message Number[4:0]	0h	R	00000b fixed.
24	Slot Implemented	0h	Hwinit	Setting it to 1b indicates that the PCI Express Link (Downstream Port) is connected to the Slot.
23 to 20	DevicePort Type[3:0]	4h	R	Indicates that it is a PCI Express Root Complex device. 0000b PCI Express Endpoint device 0001b Legacy PCI Express Endpoint device 0100b Root Port of PCI Express Root Complex (Default) 0101b Upstream Port of PCI Express Switch 0110b Downstream Port of PCI Express Switch 0111b PCI Express-to-PCI/PCI-X Bridge 1000b PCI/PCI-X-to-PCI Express Bridge 1001b Root Complex Integrated Endpoint Device 1010b Root Complex Event Collector All other encodings are reserved.
19 to 16	Capability Version[3:0]	2h	Hwinit	Indicates the version of PCI Express Capability Structure. 0010b fixed.
15 to 8	Next Capability Pointer[7:0]	0h	Hwinit (R)	Indicates that this Capability List is the final list (00h fixed). Lower 2 bits [9:8] are reserved and fixed to 00b.
7 to 0	Capability ID[7:0]	10h	R	Indicates PCI Express Capability. 10h fixed.

Table 6.6-27 Valid Reset Signal

Reset Signal	Slot Implemented	Next Capability Pointer
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		

(20) Device Capabilities Register (PCI_RC_DEVC)

This register indicates the device capability.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6064h
Initial Value : 0000_8001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Role-Based Error Reporting	-	-	-	-	-	-	-	-	-	-	-	-	Max_Payload_Size Supported[2:0]		
Initial Value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15	Role-Based Error Reporting	1h	R	Set to 1b if the Error Reporting feature (Compliant with Rev 1.1 or later) is implemented. Fixed to 1b in this core.
14 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	Max_Payload_ Size Supported[2:0]	1h	Hwinit	000b: 128B max payload size 001b: 256B max payload size (Default) 010b: 512B max payload size 011b: 1024B max payload size 100b: 2048B max payload size 101b: 4096B max payload size 110b: Reserved 111b: Reserved

Table 6.6-28 Valid Reset Signal

Reset Signal	Max_Payload_Size Support
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	

(21) Device Control/Status Register (PCI_RC_DEVCS)

This register controls the device and indicates the device status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6068h
Initial Value : 0000_2010h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	Unsupported Request Detected	Fatal Error Detected	Non-Fatal Error Detected	Correctable Error Detected
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW1	RW1	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	Max_Read_Request_Size[2:0]			-	-	-	-	Max_Payload_Size[2:0]			Enable Relaxed Ordering	Unsupported Request Reporting Enable	Fatal Error Reporting Enable	Non-Fatal Error Reporting Enable	Correctable Error Reporting Enable
Initial Value	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	RW	RW	RW	R	RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	Unsupported Request Detected	0h	RW1	Indicates that an Unsupported Request Error was detected. 1b indicates the detection of an error.
18	Fatal Error Detected	0h	RW1	Indicates that a Fatal Error was detected. 1b indicates the detection of an error.
17	Non-Fatal Error Detected	0h	RW1	Indicates that a Non-Fatal Error was detected. 1b indicates the detection of an error.
16	Correctable Error Detected	0h	RW1	Indicates that a correctable error was detected. 1b indicates the detection of an error.
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 12	Max_Read_Request_Size[2:0]	2h	RW	Set Max_Read_request_Size. 000b: 128B max read request size 001b: 256B max read request size 010b: 512B max read request size (Default) 011b: 1024B max read request size 100b: 2048B max read request size 101b: 4096B max read request size 110b: Reserved 111b: Reserved
11 to 8	-	All 0	R (RW)	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 5	Max_Payload_Size[2:0]	0h	RW	Set Max_Payload_Size. 000b: 128B max payload size (Default) 001b: 256B max payload size 010b: 512B max payload size 011b: 1024B max payload size 100b: 2048B max payload size 101b: 4096B max payload size 110b: Reserved 111b: Reserved

Bit	Bit Name	Initial Value	R/W	Description
4	Enable Relaxed Ordering	1h	RW	Set whether or not to use Relaxed Ordering as a Requester. 1b: Support 0b: not supported
3	Unsupported Request Reporting Enable	0h	RW	Controls the generation of ERR_NONFATAL and ERR_FATAL messages in response to the detection of unsupported requests. The value 1b enables generation of messages.
2	Fatal Error Reporting Enable	0h	RW	Controls the generation of ERR_FATAL Messages. 1b enables Message generation.
1	Non-Fatal Error Reporting Enable	0h	RW	Controls generation of ERR_NONFATAL Message. 1b enables Message generation.
0	Correctable Error Reporting Enable	0h	RW	Controls the generation of ERR_COR Messages. 1b enables Message generation.

Table 6.6-29 Valid Reset Signal

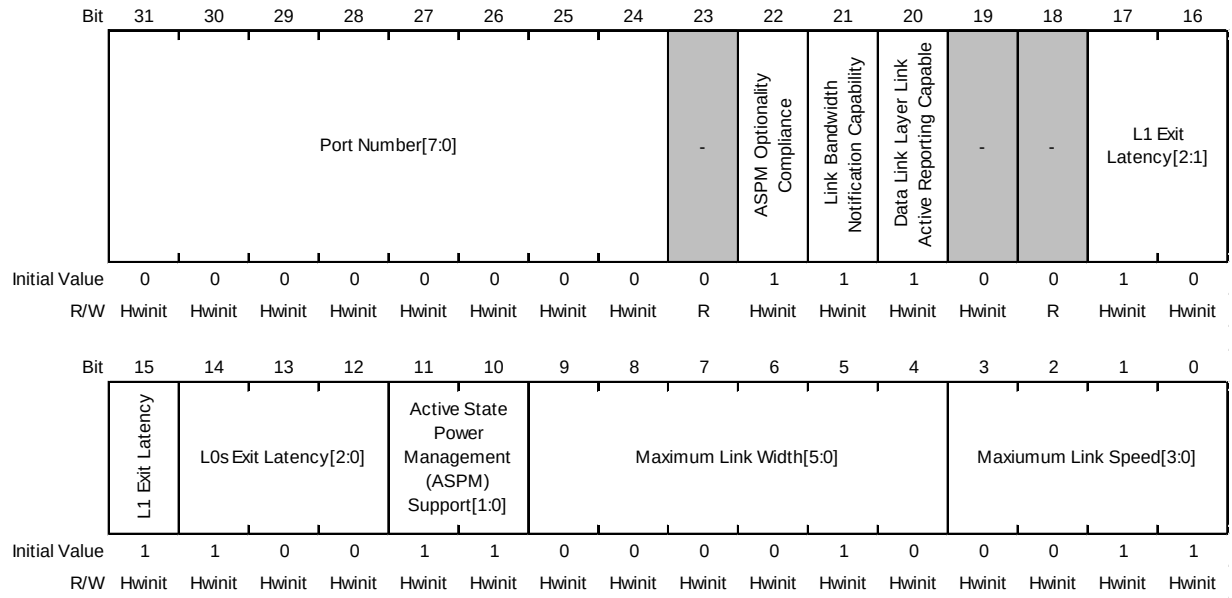
Reset Signal	Unsupported Request Detected	Fatal Error Detected	Non-Fatal Error Detected	Correctable Error Detected	Max_Read_Request_Size	Max_Payload_Size
RST_LOAD_B						
RST_RSM_B						
RST_CFG_B	✓	✓	✓	✓	✓	✓

Reset Signal	Enable Relaxed Ordering	Unsupported Request Reporting Enable	Fatal Error Reporting Enable	Non-Fatal Error Reporting Enable	Correctable Error Reporting Enable
RST_LOAD_B					
RST_RSM_B					
RST_CFG_B	✓	✓	✓	✓	✓

(22) Link Capabilities Register (PCI_RC_LINKC)

This register indicates the link capabilities. This register can be written during initialization.

Access Size : 32 bits
Offset Address : <PCI0_base> + 606Ch
Initial Value : 0072_CC23h



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Port Number[7:0]	0h	Hwinit	Indicates the port number of PCI Express Link.
23	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	ASPM Optionality Compliance	1h	Hwinit	1b fixed
21	Link Bandwidth Notification Capability	1h	Hwinit	Indicates support for Link Bandwidth Notification status and Interrupt functions.
20	Data Link Layer Link Active Reporting Capable	1h	Hwinit	Indicates support for the DL_Active state reporting feature in the Data Link Control and Management State Machine.
19	-	0h	Hwinit	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
18	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17 to 15	L1 Exit Latency[2:0]	5h	Hwinit	000b: Less than 1µs 001b: 1 µs to less than 2 µs 010b: 2 µs to less than 4 µs 011b: 4 µs to less than 8 µs 100b: 8 µs to less than 16 µs 101b: 16 µs to less than 32 µs (Default) 110b: 32 µs to 64 µs 111b: More than 64 µs
14 to 12	L0s Exit Latency[2:0]	4h	Hwinit	000b: Less than 64 ns 001b: 64 ns to less than 128 ns 010b: 128 ns to less than 256 ns 011b: 256 ns to less than 512 ns 100b: 512 ns to less than 1 µs (Default) 101b: 1 µs to less than 2 µs 110b: 2 µs-4 µs 111b: More than 4 µs

Bit	Bit Name	Initial Value	R/W	Description
11, 10	Active State Power Management (ASPM) Support[1:0]	3h	Hwinit	00b: Reserved 01b: L0s Entry Supported 10b: Reserved 11b: L0s and L1 Entry Supported (Default)
9 to 4	Maximum Link Width[5:0]	2h	Hwinit	000000b: Reserved 000001b: x1 (prohibited) 000010b: x2 (Default) 000100b: x4 (prohibited) 001000b: x8 (prohibited) 001100b: x12 (prohibited) 010000b: x16 (prohibited) 100000b: x32 (prohibited) Ch1 is set to 000010b (x2) in multilink configuration
3 to 0	Maximum Link Speed[3:0]	3h	Hwinit	0001b: 2.5 GT/s Link speed supported 0010b: 5.0 GT/s and 2.5 GT/s Link speeds supported 0011b: 8.0 GT/s Link speed supported (Default) 0100b: 16.0 GT/s Link speed supported (prohibited) All other encodings are reserved.

Table 6.6-30 Valid Reset Signal

Reset Signal	ASPM Optionality Compliance	Link Bandwidth Notification Capability	Data Link Layer Link Active Reporting Capable	Surprise Down Error Reporting Capable	L1 Exit Latency	L0s Exit Latency
RST_LOAD_B	✓	✓	✓	✓	✓	✓
RST_RSM_B						
RST_CFG_B						

Reset Signal	Active State Power Management Support	Maximum Link Width	Supported Link Speed
RST_LOAD_B	✓	✓	✓
RST_RSM_B			
RST_CFG_B			

(23) Link Control/Status Register (PCI_RC_LINKCS)

This register controls the link and indicates the link status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6070h
Initial Value : 1000_0008h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Link Autonomous Bandwidth Status	Link Bandwidth Management Status	Data Link Layer Link Active	Slot Clock Configuration	Link Training	-	Negotiated Link Width[5:0]						Current Link Speed[1:0]		-	-
Initial Value	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	R	Hwinit	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	Link Autonomous Bandwidth Interrupt Enable	Link Bandwidth Management Interrupt Enable	Hardware Autonomous Width Disable	-	Extended Synch	Common Clock Configuration	Retrain Link	Link Disable	Read Completion Boundary (RCB)	-	Active State Power Management (ASPM) Control[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	RW	RW	RW	R	RW	RW	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	Link Autonomous Bandwidth Status	0h	RW1	This bit is output when Link Autonomous Bandwidth Interrupt Enable is asserted.
30	Link Bandwidth Management Status	0h	RW1	This bit is output when Link Bandwidth Management Interrupt Enable is asserted.
29	Data Link Layer Link Active	0h	R	1b indicates that the Data Link Layer is in the Link Active state.
28	Slot Clock Configuration	1h	Hwinit	Indicates that the reference clock shared with the EP is used. 0b: Do not use Connector Reference Clock 1b: Use Connector Reference Clock (Default)
27	Link Training	0h	R	1b indicates that the Physical layer LTSSM is in Configuration state or Recovery state. This bit is cleared when exiting the Configuration/Recovery state.
26	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 20	Negotiated Link Width[5:0]	0h	R	Indicates the Link width established as a result of negotiation. 000001b: x1 000010b: x2 000100b: x4 001000b: x8 001100b: x12 010000b: x16 100000b: x32 Other encodings are Reserved.
19, 18	Current Link Speed[1:0]	0h	R	0001b: 2.5GT/s PCI Express Link 0010b: 5.0 GT/s PCI Express Link 0011b: 8.0 GT/s PCI Express Link 0100b: 16.0 GT/s PCI Express Link 0000b during reset period
17 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
11	Link Autonomous Bandwidth Interrupt Enable	0h	RW	Interrupt generation is controlled by the Link Autonomous Bandwidth Management Status bit (bit31). 0b: Interrupt Disable (Default) 1b: Interrupt Enable
10	Link Bandwidth Management Interrupt Enable	0h	RW	Interrupt generation is controlled by the Link Bandwidth Status bit (bit30). 0b: Interrupt Disable (Default) 1b: Interrupt Enable
9	Hardware Autonomous Width Disable	0h	RW	Set the Link Width Change function to Disable. 0b: Link Width Change Enable (Default) 1b: Link Width Change Disable
8	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	Extended Synch	0h	RW	If set to 1b, 4096 FTS Ordered-sets will be sent when transitioning from L0s to L0. Also, 1024 TS1 Ordered-sets are transmitted at the beginning of the Recovery state when transitioning from L1 to L0. The default value is 0b.
6	Common Clock Configuration	0h	RW	Sets whether the Common Reference Clock is used. 0b: Provided by non-Common Reference Clock (Default) 1b: Supplied by Common Reference Clock
5	Retrain Link	0h	RW	Setting it to 1b causes the LTSSM to transition to the Recovery state and start Link Retraining. Note that reading is always 0b.
4	Link Disable	0h	RW	Setting it to 1b causes the LTSSM to transition to the Disabled state.
3	Read Completion Boundary (RCB)	1h	R	0b: 64 bytes 1b: 128 bytes (Default)
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	Active State Power Management (ASPM) Control[1:0]	0h	RW	Sets the permission level for Active State Power Management. 00b Disabled (Default) 01b L0s Entry Supported 10b Reserved 11b L0s and L1 Entry Supported

Table 6.6-31 Valid Reset Signal

Reset Signal	Link Autonomous Bandwidth Status	Link Bandwidth Management Status	Slot Clock Configuration	Link Autonomous Bandwidth Interrupt Enable	Link Bandwidth Management Interrupt Enable	Hardware Autonomous Width Disable
RST_LOAD_B			✓			
RST_RSM_B						
RST_CFG_B	✓	✓		✓	✓	✓

Reset Signal	Extended Synch	Common Clock Configuration	Retrain Link	Link Disable	Active State Power Management Control
RST_LOAD_B					
RST_RSM_B					
RST_CFG_B	✓	✓	✓		✓

(24) Slot Capabilities (PCI_RC_SLOTC)

When PCI Express Capability Structure: Configuration Space Bit [24]: Slot Implemented is 0b, the slot is not implemented. Therefore, the bits in this register will be all '0'.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6074h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Physical Slot Number[12:0]													No Command Completed Support	Electromechanical Interlock Present	Slot Power Limit Scale
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Slot Power Limit Scale	Slot Power Limit Value[7:0]								Hot-Plug Capable	Hot-Plug Surprise	Power Indicator Present	Attention Indicator Present	MRL Sensor Present	Power Control Present	Attention Button Present
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	Physical Slot Number[12:0]	0h	Hwinit	Indicates the slot number connected to the port.
18	No Command Completed Support	0h	Hwinit	Indicates that the Slot will not issue a software notification if the issued command is completed by the Hot-Plug Controller.
17	Electromechanical Interlock Present	0h	Hwinit	Indicates that Electromechanical Interlock is implemented.
16, 15	Slot Power Limit Scale[1:0]	0h	Hwinit	Sets the Scale for the Slot Power Limit Value. This register is enabled when the Slot Implemented bit (Express Capability Register (offset: 060h), bit24) is set. 00b: 1.0x 01b: 0.1x 10b: 0.01x 11b: 0.001x
14 to 7	Slot Power Limit Value[7:0]	0h	Hwinit	In combination with the Slot Power Limit Scale Register value, sets the upper limit of Power for the Slot. This register is enabled when the Slot Implemented bit (Express Capability Register (offset: 060h), bit24) is set. F0h: 250W Slot Power Limit F1h: 275W Slot Power Limit F2h: 300W Slot Power Limit F3h to FFh: reserved
6	Hot-Plug Capable	0h	Hwinit	This bit indicates that this slot is capable of supporting hot-plug operations.
5	Hot-Plug Surprise	0h	Hwinit	This bit indicates that an adapter present in this slot might be removed from the system without any prior notification. This is a form factor specific capability. This bit is an indication to the operating system to allow for such removal without impacting continued software operation.
4	Power Indicator Present	0h	Hwinit	This bit indicates that a Power Indicator is electrically controlled by the chassis for this slot.
3	Attention Indicator Present	0h	Hwinit	This bit indicates that an Attention Indicator is electrically controlled by the chassis.

Bit	Bit Name	Initial Value	R/W	Description
2	MRL Sensor Present	0h	Hwinit	This bit indicates that an MRL Sensor is implemented on the chassis for this slot.
1	Power Control Present	0h	Hwinit	This bit indicates that a software programmable Power Controller is implemented for this slot/adapter (depending on form factor).
0	Attention Button Present	0h	Hwinit	This bit indicates that an Attention Button for this slot is electrically controlled by the chassis.

Table 6.6-32 Valid Reset Signal

Reset Signal	Physical Slot Number	No Command Completed Support	Electromechanical Interlock Present	Slot Power Limit Scale	Slot Power Limit Value	Hot-Plug Capable
RST_LOAD_B	✓	✓	✓	✓	✓	✓
RST_RSM_B						
RST_CFG_B						

Reset Signal	Hot-Plug Surprise	Power Indicator Present	Attention Indicator Present	MRL Sensor Present	Power Control Present	Attention Button Present
Present						
RST_LOAD_B	✓	✓	✓	✓	✓	✓
RST_RSM_B						

(25) Slot Control/Status (PCI_RC_SLOTCS)

This register is not implemented.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6078h
Initial Value : 0040_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Slot ControlStatus[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Slot ControlStatus[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Slot ControlStatus [31:0]	400000h	R	Since PCI Express Capability Structure: Configuration Space Bit [24]: Slot Implemented is 0b, the slot is not implemented, so all registers other than bit [22]: Presence Detect State are '0'.

Table 6.6-33 Valid Reset Signal

Reset Signal	Slot ControlStatus
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓

(26) Root Control/Capabilities (PCI_RC_ROOTCC)

Access Size : 32 bits
Offset Address : <PCI0_base> + 607Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	PME Interrupt Enable	System Error on Fatal Error Enable	System Error on Non-Fatal Error Enable	System Error on Correctable Error Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
3	PME Interrupt Enable	0h	RW	When a PME Message is received, it is possible to generate a PME interrupt along with setting the PME status bit. An interrupt will also occur if this bit is enabled after PME status is already set.
2	System Error on Fatal Error Enable	0h	RW	This bit makes it possible to generate a system error when ERR_FATAL occurs anywhere in the hierarchy below the root port.
1	System Error on Non-Fatal Error Enable	0h	RW	This bit makes it possible to generate a system error when NON_ERR_FATAL occurs anywhere in the hierarchy below the root port.
0	System Error on Correctable Error Enable	0h	RW	This bit makes it possible to generate a system error when ERR_COR occurs in any of the layers below the root port.

Table 6.6-34 Valid Reset Signal

Reset Signal	PME Interrupt Enable	System Error on Fatal Error Enable	System Error on Non-Fatal Error Enable	System Error on Correctable Error Enable
RST_LOAD_B				
RST_RSM_B				
RST_CFG_B	✓	✓	✓	✓

(27) Root Status (PCI_RC_ROOTS)

Access Size : 32 bits
Offset Address : <PCI0_base> + 6080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PME Pending	PME Status
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PME Requester ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	PME Pending	0h	R	This bit indicates that the PME Status bit is 1 and PME is pending. When the PME status bit is cleared by software, pending PMEs are resent in hardware by resetting the PME status bit and updating the Requester ID field appropriately. The PME pending bit is cleared by hardware when there are no more pending PMEs.
16	PME Status	0h	RW1	This bit indicates that a PME was received by the requestor indicated by the PME Requester ID field. Subsequent PMEs are held off until this bit is cleared by software.
15 to 0	PME Requester ID[15:0]	0h	R	This field indicates the PCI Requester ID of the most recent PME requestor. This field is only valid when the PME status bit is 1b.

Table 6.6-35 Valid Reset Signal

Reset Signal	PME Pending	PME Status	PME Requester ID
RST_LOAD_B			
RST_RSM_B			
RST_CFG_B	✓	✓	✓

(28) Device Capabilities 2 Register (PCI_RC_DEVC2)

This register indicates the device capability.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6084h
Initial Value : 0000_0012h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	Completion Timeout Disable Supported	Completion Timeout Ranges Supported[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
R/W	R	R	R	R	Hwinit	R	R	R	R	R	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 16	-	All 0	Hwinit	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11	-	All 0	Hwinit	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
10 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	Completion Timeout Disable Supported	1h	Hwinit	Set whether to support the Completion Timeout Disable function. 0b: not supported 1b: Supported (default)
3 to 0	Completion Timeout Ranges Supported[3:0]	2h	Hwinit	Set Completion Timeout Range. Range A: 50 µs to 10 ms Range B: 10 ms to 250 ms Range C: 250 ms to 4 s Range D: 4 s to 64 s The above four patterns are determined, and the following combinations can be set. 0000b: Program setting Not supported 0001b: Range A 0010b: Range B 0011b: Ranges A and B 0110b: Ranges B and C 0111b: Ranges A, B, and C 1110b: Ranges B, C and D 1111b: Ranges A, B, C, and D Other encodings are Reserved. Note: The initial value described in UM is "0010b", but please change the initial value according to the installed system/device performance, or specify the initial value as a product request.

Table 6.6-36 Valid Reset Signal

Reset Signal	Completion Timeout Disable Supported	Completion Timeout Ranges Supported
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		

(29) Device Control 2/Status 2 Register (PCI_RC_DEVCS2)

This register controls the device and indicates the device status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6088h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	Completion Timeout Disable	Completion Timeout Value[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	R	RW	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R (RW)	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	Completion Timeout Disable	0h	RW	Setting this bit enables the Completion Timeout Disable function.
3 to 0	Completion Timeout Value[3:0]	0h	RW	Set Completion Timeout Range. 0000b: 10 ms to 50 ms (default) 0001b: 50 µs to 100 µs 0010b: 1 ms to 10 ms 0101b: 16 ms to 55 ms 0110b: 65 ms to 210 ms 1001b: 260 ms to 900 ms 1010b: 1 s to 3.5 s 1101b: 4 s to 13 s 1110b: 17 s to 64 s Others: reserved (setting prohibited) Note: The default 0000b sets a longer time than the Base Spec's default 50usec lower limit, but this takes into account the Base Spec's recommended 10msec lower limit.

Table 6.6-37 Valid Reset Signal

Reset Signal	Completion Timeout Disable	Completion Timeout Value
RST_LOAD_B		
RST_RSM_B		
RST_CFG_B	✓	✓

(30) Link Capabilities 2 Register (PCI_RC_LINKC2)

This register indicates the link capabilities.

Access Size : 32 bits
Offset Address : <PCI0_base> + 608Ch
Initial Value : 0000_000Eh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	Supported Link Speeds Vector[6:0]							-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24 to 9	-	All 0	Hwinit	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 1	Supported Link Speeds Vector[6:0]	7h	Hwinit	The read value is undefined Indicates Support Link Speed Bit[0] 2.5 GT/s Bit[1] 5.0 GT/s Bit[2] 8.0 GT/s Bit[3] 16.0 GT/s (Setting prohibited) Bits[6:4] Reserved
0	-	All 0	R	Reserved Whenever it is read, undefined value is read. The written value will be ignored.

Table 6.6-38 Valid Reset Signal

Reset Signal	Supported Link Speeds Vector
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	

(31) Link Control 2/Status 2 Register (PCI_RC_LINCS2)

This register controls the link and indicates the link status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6090h
Initial Value : 0000_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	Link Equalization Request	Equalization Phase 3 Successful	Equalization Phase 2 Successful	Equalization Phase 1 Successful	Equalization Complete	Current De-emphasis Level
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW1	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Compliance PresetDe-emphasis[3:0]				Compliance SOS	Enter Modified Compliance	Transmit Margin[2:0]			Selectable De-emphasis	Hardware Autonomous Speed Disable	Enter Compliance	Target Link Speed[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	Hwinit	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21	Link Equalization Request	0h	RW1	This bit is Set by hardware to request the Link equalization process to be performed on the Link. The default value of this bit is 0b. For Multi-Function Upstream Port, this bit must be implemented in Function 0 and RsvdZ in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.
20	Equalization Phase 3 Successful	0h	R	When set to 1b, this bit indicates that Phase 3 of the Transmitter Equalization procedure has successfully completed. The default value of this bit is 0b. For Multi-Function Upstream Port, this bit must be implemented in Function 0 and RsvdZ in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.
19	Equalization Phase 2 Successful	0h	R	When set to 1b, this bit indicates that Phase 2 of the Transmitter Equalization procedure has successfully completed. The default value of this bit is 0b. For Multi-Function Upstream Port, this bit must be implemented in Function 0 and RsvdZ in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.
18	Equalization Phase 1 Successful	0h	R	When set to 1b, this bit indicates that Phase 1 of the Transmitter Equalization procedure has successfully completed. The default value of this bit is 0b. For Multi-Function Upstream Port, this bit must be implemented in Function 0 and RsvdZ in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.
17	Equalization Complete	0h	R	When set to 1b, this bit indicates that the Transmitter Equalization procedure has completed. The default value of this bit is 0b. For Multi-Function Upstream Port, this bit must be implemented in Function 0 and RsvdZ in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.

Bit	Bit Name	Initial Value	R/W	Description
16	Current De-emphasis Level	0h	R	This is a Status register that indicates the De-emphasis Level during Gen2 operation. 1b: -3.5dB 0b: -6 dB (Default) Note: The initial value of this bit is 0b at reset and after Gen2-Linkup. However, although it indicates 1b at Gen1-Linkup, ignore this bit as it has no meaning in Gen1.
15 to 12	Compliance PresetDe-emphasis[3:0]	0h	RW	Set Transmitter Preset(Gen3) and De-emphasis Level(Gen2) for Polling.Compliance State processing during Enter Compliance. (Gen3) 0000b: P0 (Default) 0001b: P1 0010b: P2 0011b: P3 0100b: P4 0101b: P5 0110b: P6 0111b: P7 1000b: P8 1001b: P9 1010b: P10 Others: Rsvd (Gen2) 0001b: -3.5 dB 0000b: -6 dB (Default)
11	Compliance SOS	0h	RW	When this bit is set to 1b, insert the SKP Ordered-set periodically during compliance pattern transmission.
10	Enter Modified Compliance	0h	RW	Setting bit for transmission of Modified Compliance pattern. 1b: Modified Compliance Pattern 0b: Compliance Pattern (Default)
9 to 7	Transmit Margin[2:0]	0h	RW	Adjust the voltage level of the Transmitter. 000b: Normal operating range 001b-111b: See Base Spec Section 6.6.6.4.4.
6	Selectable De-emphasis	0h	Hwinit	The read value is undefined De-emphasis setting register for RC Gen2 operation. 1b: -3.5dB 0b: -6dB
5	Hardware Autonomous Speed Disable	0h	RW	Controls the Link Speed Change function. 1b: Non-support of Link Speed Change (Disable) 0b: Support of Link Speed Change (Enable)
4	Enter Compliance	0h	RW	Setting to 1b allows transition to Compliance mode. At this time, the Link Speed will be the value set in the Target Link Speed field.
3 to 0	Target Link Speed[3:0]	3h	RW	Set the Link Speed value for notification to the opposite device during training. 0001b: 2.5 GT/s Target Link Speed 0010b: 5.0 GT/s Target Link Speed 0011b: 8.0 GT/s Target Link Speed (Default) 0100b: 16.0 GT/s Target Link Speed (prohibited) All other encodings are reserved.

Table 6.6-39 Valid Reset Signal

Reset Signal	Link Equalization Request	Compliance Preset/ De-emphasis	Compliance SOS	Enter Modified Compliance	Transmit Margin	Hardware Autonomous Speed Disable
RST_LOAD_B						
RST_RSM_B	✓	✓	✓	✓	✓	✓
RST_CFG_B						

Reset Signal	Enter Compliance	Target Link Speed
RST_LOAD_B		
RST_RSM_B	✓	✓
RST_CFG_B		

(32) Slot Capabilities 2 (PCI_RC_SLOTC2)

This register will not be used in Root Complex mode.

(33) Slot Control 2 / Status 2 (PCI_RC_SLOTCS2)

This register will not be used in Root Complex mode.

(34) Base Address Register Mask00 (Lower) (PCI_RC_BARMSK00L)

This register indicates mask information for Base Address Register 0 (BAR0).

Access Size : 32 bits
Offset Address : <PCI0_base> + 60A0h
Initial Value : 1FFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Base Address Register Mask00(Lower)[31:16]															
Initial Value	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Base Address Register Mask00(Lower)[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Base Address Register Mask00(Lower) [31:0]	1FFFFFFFh	Hwinit	Mask register for Base Address Register 0 (BAR0).

Table 6.6-40 Valid Reset Signal

Reset Signal	Base Address Register Mask00 (Lower)
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	

(35) Base Address Register Mask00 (Higher) (PCI_RC_BARMSK00U)

This register indicates mask information for Base Address Register 1 (BAR1).

Access Size : 32 bits
Offset Address : <PCI0_base> + 60A4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Base Address Register Mask00(Higher)[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Base Address Register Mask00(Higher)[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Base Address Register Mask00(Higher) [31:0]	0h	Hwinit	Mask register for Base Address Register 1 (BAR1).

Table 6.6-41 Valid Reset Signal

Reset Signal	Base Address Register Mask00 (Higher)
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	

(36) Base Address Register Mask01 (Lower) (PCI_RC_BARMSK01L)

This register will not be used in Root Complex mode.

(37) Base Address Register Mask01 (Higher) (PCI_RC_BARMSK01U)

This register will not be used in Root Complex mode.

(38) Base Address Register Mask02 (Lower) (PCI_RC_BARMSK02L)

This register will not be used in Root Complex mode.

(39) Base Address Register Mask02 (Higher) (PCI_RC_BARMSK02U)

This register will not be used in Root Complex mode.

(40) Base Size 00/01 Register (PCI_RC_BSIZE00_01)

This register sets the acceptable TLP size.

Access Size : 32 bits
Offset Address : <PCI0_base> + 60C8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Base Size 00[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 16	-	All 0	Hwinit	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	Base Size 00[9:0]	0h	Hwinit	Sets the size of the TLP (DW Size) that will be accepted in Address Space 00 set in the Base Address Register and Base Address Mask Register. If a TLP with a packet length larger than the size set here is received (even if it is smaller than the Max Payload Size), a Completer Abort Error (CA) will be detected. Note that the default value is 000h, which disables this function.

Table 6.6-42 Valid Reset Signal

Reset Signal	Base Size 00
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	

(41) Base Size 02/03 Register (PCI_RC_BSIZE02_03)

This register will not be used in Root Complex mode.

(42) Base Size 04/05 Register (PCI_RC_BSIZE04_05)

This register will not be used in Root Complex mode.

(43) Base Size 06 Register (PCI_RC_BSIZE06)

This register will not be used in Root Complex mode.

(44) Type Supported 00/01/02 Register (PCI_RC_TSUPPORT00_01_02)

This register indicates the transaction type which can be supported by the memory space.

Access Size : 32 bits
Offset Address : <PCI0_base> + 60D8h
Initial Value : 0033_3333h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	Type Supported 00[7:0]							
Initial Value	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	00_3333h	Hwinit	Reserved Whenever it is read, the initial value is read. The written value should always be the initial value.
7 to 0	Type Supported 00[7:0]	33h	Hwinit	Set the Transaction Types that can be supported by Space00 (CFG_SPACE00_BASE). Bit[0]: 32-bit memory read Bit[1]: 64-bit memory read Bit[2]: 32-bit memory read lock Bit[3]: 64-bit memory read lock Bit[4]: 32-bit memory write Bit[5]: 64-bit memory write Bit[6]: IO read Bit[7]: IO write

Table 6.6-43 Valid Reset Signal

Reset Signal	Type Supported 00
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	

(45) Advanced Error Reporting Capability Register (PCI_RC_ADVERC)

This register indicates the advanced error reporting capability.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6100h
Initial Value : 1501_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Next Capability Offset[11:0]												Capability Version[3:0]			
Initial Value	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	1
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	R	R	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCI Express Extended Capability ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	Next Capability Offset[11:0]	150h	Hwinit (R)	Indicates the starting address for Device Serial Number Capability
19 to 16	Capability Version[3:0]	1h	Hwinit	Indicates the version of the Capability Structure. Default: 0001b
15 to 0	PCI Express Extended Capability ID[15:0]	1h	R	Indicates the Advanced Error Reporting Capability. Default: 0001h

Table 6.6-44 Valid Reset Signal

Reset Signal	Next Capability Offset	Capability Version
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		

(46) Uncorrectable Error Status Register (PCI_RC_UNCESTS)

This register indicates the uncorrectable error status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6104h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	Unsupported Request Error Status	ECRC Error Status (Optional)	Malformed TLP Status	Receiver Overflow Status (Optional)	Unexpected Completion Status
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW1	RW1	RW1	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Completer Abort Status (Optional)	Completion Timeout Status	-	Poisoned TLP Received Status	-	-	-	-	-	-	-	Data Link Protocol Error Status	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	R	RW1	R	R	R	R	R	R	R	RW1	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	Unsupported Request Error Status	0h	RW1	Indicates that an unsupported TLP was received. 0b: No error detected 1b: error detected
19	ECRC Error Status (Optional)	0h	RW1	Indicates that an ECRC Error was received. 0b: No error detected 1b: error detected
18	Malformed TLP Status	0h	RW1	Indicates that a Malformed TLP was received. 0b: No error detected 1b: error detected
17	Receiver Overflow Status (Optional)	0h	RW1	Indicates that a TLP with a size larger than the free credits in the receive buffer was received. 0b: No error detected 1b: error detected
16	Unexpected Completion Status	0h	RW1	Indicates that a Completion was received, but there is no record of a corresponding Non-Posted Request sent (due to a mismatch with the Transaction Descriptor). 0b: No error detected 1b: error detected
15	Completer Abort Status (Optional)	0h	RW1	Indicates that a Completion whose Completion Status is Completer Abort (CA) was returned after receiving a Non-Posted Request. 0b: No error detected 1b: error detected
14	Completion Timeout Status	0h	RW1	Indicates that the corresponding Completion was not received within the specified time after sending a Non-Posted Request. 0b: No error detected 1b: error detected
13	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12	Poisoned TLP Received Status	0h	RW1	Indicates that a Poisoned TLP (with payload and the EP field in the header is 1b) has been received 0b: No error detected 1b: error detected

Bit	Bit Name	Initial Value	R/W	Description
11 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	Data Link Protocol Error Status	0h	RW1	Indicates that a Sequence Number error was detected in the Data Link Layer. 0b: No error detected 1b: error detected
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(47) Uncorrectable Error Mask Register (PCI_RC_UNCEMASK)

This register masks the uncorrectable error status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6108h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	Unsupported Request Error Mask	ECRC Error Mask (Optional)	Malformed TLP Mask	Receiver Overflow Mask (Optional)	Unexpected Completion Mask
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Completer Abort Mask (Optional)	Completion Timeout Mask	-	Poisoned TLP Received Mask	-	-	-	-	-	-	-	Data Link Protocol Error Mask	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	R	RW	R	R	R	R	R	R	R	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	Unsupported Request Error Mask	0h	RW	Masks the error notification when an Unsupported Request Error is detected. 0b: no mask 1b: Masks error message transmission, recording of the header in the Header Log register, and updating of the first error pointer.
19	ECRC Error Mask (Optional)	0h	RW	Masks error notification when ECRC Error is detected. 0b: no mask 1b: mask
18	Malformed TLP Mask	0h	RW	Masks error notification when Malformed TLP Error is detected. 0b: no mask 1b: Masks error message transmission, recording of the header in the Header Log register, and updating of the first error pointer.
17	Receiver Overflow Mask (Optional)	0h	RW	Masks error notification when Receiver Overflow Error is detected. 0b: no mask 1b: Masks error message transmission and updating of the first error pointer.
16	Unexpected Completion Mask	0h	RW	Masks error notification when Unexpected Completion Error is detected. 0b: no mask 1b: Masks error message transmission, recording of the header in the Header Log register, and updating of the first error pointer.
15	Completer Abort Mask (Optional)	0h	RW	Masks the error notification when a Completer Abort Error is detected. 0b: no mask 1b: Masks error message transmission, recording of the header in the Header Log register, and updating of the first error pointer.
14	Completion Timeout Mask	0h	RW	Masks the error notification when Completion Timeout Error is detected. 0b: no mask 1b: Masks error message transmission and updating of the first error pointer.
13	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12	Poisoned TLP Received Mask	0h	RW	Mask error notification when Poisoned TLP Error is detected. 0b: no mask 1b: Masks error message transmission, recording of the header in the Header Log register, and updating of the first error pointer.

Bit	Bit Name	Initial Value	R/W	Description
11 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	Data Link Protocol Error Mask	0h	RW	Masks error notifications when a Data Link Protocol Error is detected. 0b: no mask 1b: Masks error message transmission and updating of the first error pointer.
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Table 6.6-45 Valid Reset Signal

Reset Signal	Unsupported Request Error Mask	ECRC Error Mask	Malformed TLP Mask	Receiver Overflow Mask	Unexpected Completion Mask	Completer Abort Mask
RST_LOAD_B						
RST_RSM_B	✓	✓	✓	✓	✓	✓
RST_CFG_B						

Reset Signal	Completion Timeout Mask	Poisoned TLP Received Mask	Data Link Protocol Error Mask
RST_LOAD_B			
RST_RSM_B	✓	✓	✓
RST_CFG_B			

(48) Uncorrectable Error Severity Register (PCI_RC_UNCESVY)

This register sets the uncorrectable error severity.

Access Size : 32 bits
Offset Address : <PCI0_base> + 610Ch
Initial Value : 0046_2030h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	Unsupported Request Error Severity	ECRC Error Severity (Optional)	Malformed TLP Severity	Receiver Overflow Severity (Optional)	Unexpected Completion Severity
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	Hwinit	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Completer Abort Severity (Optional)	Completion Timeout Severity	-	Poisoned TLP Received Severity	-	-	-	-	-	-	-	Data Link Protocol Error Severity	-	-	-	-
Initial Value	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0
R/W	RW	RW	R	RW	R	R	R	R	R	R	R	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	-	1h	Hwinit	Reserved Whenever it is read, 1b is read. The write value should always be 1b.
21	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	Unsupported Request Error Severity	0h	RW	Set the Error Severity when Unsupported Request Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
19	ECRC Error Severity (Optional)	0h	RW	Sets the Error Severity when receiving an ECRC Error. 0b: Non-Fatal Error 1b: Fatal Error
18	Malformed TLP Severity	1h	RW	Sets the Error Severity when Malformed TLP Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
17	Receiver Overflow Severity (Optional)	1h	RW	Set the Error Severity when Receiver Overflow Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
16	Unexpected Completion Severity	0h	RW	Sets the Error Severity when an Unexpected Completion Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
15	Completer Abort Severity (Optional)	0h	RW	Sets the Error Severity when a Completer Abort Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
14	Completion Timeout Severity	0h	RW	Sets the Error Severity when Completion Timeout Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
13	-	1h	R	Reserved Whenever it is read, 1b is read. The written value will be ignored.
12	Poisoned TLP Received Severity	0h	RW	Sets the Error Severity when Poisoned TLP Error is detected. 0b: Non-Fatal Error 1b: Fatal Error

Bit	Bit Name	Initial Value	R/W	Description
11 to 5	-	1h	R	Reserved Whenever it is read, 1h is read. The written value will be ignored.
4	Data Link Protocol Error Severity	1h	RW	Sets the Error Severity when a Data Link Protocol Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Table 6.6-46 Valid Reset Signal

Reset Signal	Unsupported Request Error Severity	ECRC Error Severity	Malformed TLP Severity	Receiver Overflow Severity	Unexpected Completion Severity	Completer Abort Severity
RST_LOAD_B						
RST_RSM_B	✓	✓	✓	✓	✓	✓
RST_CFG_B						

Reset Signal	Completion Timeout Severity	Poisoned TLP Received Severity	Data Link Protocol Error Severity
RST_LOAD_B			
RST_RSM_B	✓	✓	✓
RST_CFG_B			

(49) Correctable Error Status Register (PCI_RC_CESTS)

This register indicates the correctable error status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6110h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Advisory Non-Fatal Error Status	Replay Timer Timeout Status	-	-	-	REPLAY_NUM Rollover Status	Bad DLLP Status	Bad TLP Status	-	-	-	-	-	Receiver Error Status (optional)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW1	RW1	R	R	R	RW1	RW1	RW1	R	R	R	R	R	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	Advisory Non-Fatal Error Status	0h	RW1	Indicates that an Advisory Non-Fatal Error was detected. 0b: No error detected 1b: error detected
12	Replay Timer Timeout Status	0h	RW1	Indicates that a Timeout error occurred when a TLP was sent and an Ack or Nak DLLP could not be received within the specified time. 0b: No error detected 1b: error detected
11 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	REPLAY_NUM Rollover Status	0h	RW1	Replay occurred four times in a row, indicating REPLAY_NUM rolled over from 11b to 00b. 0b: No error detected 1b: error detected
7	Bad DLLP Status	0h	RW1	Indicates that a DLLP CRC error was detected. 0b: No error detected 1b: error detected
6	Bad TLP Status	0h	RW1	Indicates that a TLP CRC error or Sequence Number error was detected. 0b: No error detected 1b: error detected
5 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	Receiver Error Status (optional)	0h	RW1	0b: No error detected 1b: error detected

Table 6.6-47 Valid Reset Signal

Reset Signal	Advisory Non-Fatal Error Status	Replay Timer Timeout Status	REPLAY_NUM Rollover Status	Bad DLLP Status	Bad TLP Status	Receiver Error Status
RST_LOAD_B						
RST_RSM_B	✓	✓	✓	✓	✓	✓
RST_CFG_B						

(50) Correctable Error Mask Register (PCI_RC_CEMASK)

This register masks the correctable error status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6114h
Initial Value : 0000_2000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Advisory Non-Fatal Error Mask	Replay Timer Timeout Mask	-	-	-	REPLAY_NUM Rollover Mask	Bad DLLP Mask	Bad TLP Mask	-	-	-	-	-	Receiver Error Mask (optional)
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	R	RW	RW	RW	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	Advisory Non-Fatal Error Mask	1h	RW	Masks error notification when Advisory non-fatal errors are detected. 0b: no mask 1b: Mask Advisory Non-Fatal Error handling (Mask updating First Error Pointer and Header Logging and sending Error Message)
12	Replay Timer Timeout Mask	0h	RW	Masks error notification when Replay Timer Timeout Error is detected. 0b: no mask 1b: Mask error message transmission
11 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	REPLAY_NUM Rollover Mask	0h	RW	REPLAY_NUM Mask error notification when Roll Over Error is detected. 0b: no mask 1b: Mask error message transmission
7	Bad DLLP Mask	0h	RW	Masks error notification when Bad DLLP Error is detected. 0b: no mask 1b: Mask error message transmission
6	Bad TLP Mask	0h	RW	Masks error notification when Bad TLP Error is detected. 0b: no mask 1b: Mask error message transmission
5 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	Receiver Error Mask (optional)	0h	RW	0b: no mask 1b: Mask error message transmission

Table 6.6-48 Valid Reset Signal

Reset Signal	Advisory Non-Fatal Error Mask	Replay Timer Timeout Mask	REPLAY_NUM Rollover Mask	Bad DLLP Mask	Bad TLP Mask	Receiver Error Mask
RST_LOAD_B						
RST_RSM_B	✓	✓	✓	✓	✓	✓
RST_CFG_B						

(51) Advanced Error Capabilities and Control Register (PCI_RC_ADVECC)

This register indicates and controls the advanced error capabilities.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6118h
Initial Value : 0000_00A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	ECRC Check Enable	ECRC Check Capable	ECRC Generation Enable	ECRC Generation Capable	First Error Pointer[4:0]				
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	RW	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	ECRC Check Enable	0h	RW	ECRC Check Enable setting 0b: Disable (Default) 1b: Enable
7	ECRC Check Capable	1h	R	Presence or absence of ECRC Check function 0b: Absent 1b: Present (Default)
6	ECRC Generation Enable	0h	RW	ECRC Generation Enable setting 0b: Disable (Default) 1b: Enable
5	ECRC Generation Capable	1h	R	Presence or absence of ECRC Generation function 0b: Absent 1b: Present (Default)
4 to 0	First Error Pointer[4:0]	0h	R	Indicates the field value of the Uncorrectable Error Status register for the first detected Uncorrectable Error Bit[4:0] First Error Pointer is reset by RST_GP_B.

Table 6.6-49 Valid Reset Signal

Reset Signal	ECRC Check Enable	ECRC Generation Enable
RST_LOAD_B		
RST_RSM_B	✓	✓
RST_CFG_B		

(52) Header Log Register 0 (PCI_RC_HLOG0)

This register indicates the header log.

Access Size : 32 bits
Offset Address : <PCI0_base> + 611Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Header of TLP associated with error 0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Header of TLP associated with error 0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Header of TLP associated with error 0[31:0]	0h	R	Indicates the 1st DW of Header for the first detected Uncorrectable Error.

(53) Header Log Register 1 (PCI_RC_HLOG1)

This register indicates the header log.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6120h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Header of TLP associated with error 1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Header of TLP associated with error 1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Header of TLP associated with error 1[31:0]	0h	R	Indicates the 2nd DW of Header for the first detected Uncorrectable Error.

(54) Header Log Register 2 (PCI_RC_HLOG2)

This register indicates the header log.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6124h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Header of TLP associated with error 2[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Header of TLP associated with error 2[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Header of TLP associated with error 2[31:0]	0h	R	Indicates the 3rd DW of Header for the first detected Uncorrectable Error.

(55) Header Log Register 3 (PCI_RC_HLOG3)

This register indicates the header log.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6128h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Header of TLP associated with error 3[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Header of TLP associated with error 3[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Header of TLP associated with error 3[31:0]	0h	R	Indicates the 4th DW of Header for the first detected Uncorrectable Error.

(56) Root Error Command (PCI_RC_ROOTEC)

Access Size : 32 bits
Offset Address : <PCI0_base> + 612Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	Fatal Error Reporting Enable	Non-Fatal Error Reporting Enable	Correctable Error Reporting Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	Fatal Error Reporting Enable	0h	RW	This bit enables an interrupt to be generated when ERR_FATAL occurs in any of the layers below the root port.
1	Non-Fatal Error Reporting Enable	0h	RW	This bit enables an interrupt to be generated when NON_ERR_FATAL occurs in any of the layers below the root port.
0	Correctable Error Reporting Enable	0h	RW	This bit enables an interrupt to be generated when ERR_COR occurs in any of the layers below the root port.

Table 6.6-50 Valid Reset Signal

Reset Signal	Fatal Error Reporting Enable	Non-Fatal Error Reporting Enable	Correctable Error Reporting Enable
RST_LOAD_B			
RST_RSM_B			
RST_CFG_B	✓	✓	✓

(57) Root Error Status (PCI_RC_ROOTES)

Access Size : 32 bits
Offset Address : <PCI0_base> + 6130h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	Fatal Error Message Received	Non-Fatal Error Message Received	First Uncorrectable Fatal	Multiple ERR_FATALNONFATAL Received	ERR_FATALNONFATAL Received	Multiple ERR_COR Received	ERR_COR Received
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW1	RW1	RW1	RW1	RW1	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6	Fatal Error Message Received	0h	RW1	Set when one or more ERR_FATAL is received.
5	Non-Fatal Error Message Received	0h	RW1	Set when one or more ERR_NON_FATAL is received.
4	First Uncorrectable Fatal	0h	RW1	Set when the first uncorrectable error is received and the error was ERR_FATAL.
3	Multiple ERR_FATALNONFATAL Received	0h	RW1	This means that ERR_FATAL or ERR_NON_FATAL was received while the ERR_FATAL/NONFATAL Received bit was already set.
2	ERR_FATALNONFATAL Received	0h	RW1	Indicates that ERR_FATAL or ERR_NON_FATAL was received with the ERR_FATAL/NONFATAL Received bit cleared.
1	Multiple ERR_COR Received	0h	RW1	It means that another ERR_COR was received while the ERR_COR Received bit was already set.
0	ERR_COR Received	0h	RW1	Indicates that ERR_COR was received with the ERR_COR Received bit cleared.

Table 6.6-51 Valid Reset Signal

Reset Signal	Fatal Error Message Received	Non-Fatal Error Message Received	First Uncorrectable Fatal	Multiple ERR_FATAL/NONFATAL Received	ERR_FATAL/NONFATAL Received	Multiple ERR_COR Received
RST_LOAD_B						
RST_RSM_B	✓	✓	✓	✓	✓	✓
RST_CFG_B						

Reset Signal	ERR_COR Received
RST_LOAD_B	
RST_RSM_B	✓
RST_CFG_B	

(58) Error source Identification Register (PCI_RC_ERRSI)

Access Size : 32 bits
Offset Address : <PCI0_base> + 6134h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ERR_FATALNONFATAL Source Identification[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERR_COR Source Identification[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	ERR_FATALNONFATAL Source Identification [15:0]	0h	R	When ERR_FATAL or ERR_NONFATAL is received when the ERR_FATAL/NONFATAL Received bit of the Root Error Status register is not set, the requester ID of the error is captured.
15 to 0	ERR_COR Source Identification [15:0]	0h	R	When the ERR_COR Received bit in the Root Error Status register is not set and ERR_COR is received, the error requester ID is captured.

Table 6.6-52 Valid Reset Signal

Reset Signal	ERR_FATAL/NONFATAL Source Identification	ERR_COR Source Identification
RST_LOAD_B		
RST_RSM_B	✓	✓
RST_CFG_B		

(59) Device Serial Number Extended Capability Register (PCI_RC_DEVSNEXTC)

This register specifies the device serial number extended capability.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6150h
Initial Value : 1B01_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Next Capability Offset[11:0]												Capability Version[3:0]			
Initial Value	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCI Express Extended Capability ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	Next Capability Offset[11:0]	1B0h	R	Indicates the starting address of the Secondary PCI Express Extended Capability Header. 1B0h fixed
19 to 16	Capability Version[3:0]	1h	Hwinit	Indicates the version of the Capability Structure. Default: 0001b
15 to 0	PCI Express Extended Capability ID[15:0]	3h	R	Indicates the Device Serial Number Extended Capability. Default: 0003h

Table 6.6-53 Valid Reset Signal

Reset Signal	Next Capability Offset[11:2]	Capability Version
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		

(60) Serial Number Register (Lower DW) (PCI_RC_SNL)

This register specifies the serial number of the device.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6154h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PCI Express Device Serial Number(Lower DW)[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCI Express Device Serial Number(Lower DW)[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PCI Express Device Serial Number(Lower DW)[31:0]	0h	Hwinit	The lower 32 bits of the IEEE standard 64-bit unique ID (EUI-64). EUI-64 consists of a 24-bit company ID and a 40-bit vendor-defined extension.

Table 6.6-54 Valid Reset Signal

Reset Signal	PCI Express Device Serial Number (Lower DW)
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	

(61) Serial Number Register (Higher DW) (PCI_RC_SNU)

This register specifies the serial number of the device.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6158h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PCI Express Device Serial Number(Higher DW)[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCI Express Device Serial Number(Higher DW)[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PCI Express Device Serial Number(Higher DW)[31:0]	0h	Hwinit	The Higher 32 bits of the IEEE standard 64-bit unique ID (EUI-64). EUI-64 consists of a 24-bit company ID and a 40-bit vendor-defined extension.

Table 6.6-55 Valid Reset Signal

Reset Signal	PCI Express Device Serial Number (Higher DW)
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	

(62) Secondary PCI Express Extended Capability Header (PCI_RC_SPEECH)

This register specifies the Secondary PCI Express Extended Capability Header.

Access Size : 32 bits
Offset Address : <PCI0_base> + 61B0h
Initial Value : 0001_0019h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Next Capability Offset[11:0]												Capability Version[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCI Express Extended Capability ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	Next Capability Offset[11:0]	0h	R	Indicates that this Capability List is the final List. 000h fixed
19 to 16	Capability Version[3:0]	1h	Hwinit	Indicates the version of the Capability Structure. Default: 0001b
15 to 0	PCI Express Extended Capability ID[15:0]	19h	R	Indicates the Secondary PCI Express Extended Capability Header. Default: 0019h

Table 6.6-56 Valid Reset Signal

Reset Signal	Capability Version
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	

(63) Link Control 3 Register (PCI_RC_LINC3)

Access Size : 32 bits
Offset Address : <PCI0_base> + 61B4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Link Equalization Request Interrupt Enable	Perform Equalization
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 9	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	Link Equalization Request Interrupt Enable	0h	RW	Link Equalization Request Interrupt Enable When Set, this bit enables the generation of an interrupt to indicate that the Link Equalization 8.0 GT/s Request bit or the Link Equalization Request 16.0 GT/s bit has been set. This bit is RW for Downstream Ports and for Upstream Ports when Crosslink Supported is 1b. This bit is not applicable and is RsvdP for Upstream Ports when the Crosslink Supported bit is 0b. The default value for this bit is 0b. RC: Default 0b EP: Fixed to 0 because Crosslink is not supported
0	Perform Equalization	0h	RW	Perform Equalization When this bit is 1b and a 1b is written to the Retrain Link bit with the Target Link Speed field set to 8.0 GT/s or higher, the Downstream Port must perform Link Equalization. This bit is RW for Downstream Ports and for Upstream Ports when Crosslink Supported is 1b. This bit is not applicable and is RsvdP for Upstream Ports when the Crosslink Supported bit is 0b. The default value is 0b. RC: Default 0b EP: Fixed to 0 because Crosslink is not supported

Table 6.6-57 Valid Reset Signal

Reset Signal	Enable Lower SKP OS Generation Vector	Link Equalization Request Interrupt Enable	Perform Equalization
RST_LOAD_B			
RST_RSM_B			
RST_CFG_B	✓	✓	✓

(64) Lane Error Status Register (PCI_RC_LESTA)

Access Size : 32 bits
Offset Address : <PCI0_base> + 61B8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Lane Error Status Bits[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	Lane Error Status Bits[1:0]	0h	RW1	Each bit indicates if the corresponding Lane detected a Lane-based error. A value of 1b indicates that a Lane based-error was detected on the corresponding Lane Number. The default value of each bit is 0b. For Links that are narrower than 32 bits, the unused upper bits [31:Bits Maximum Link Width] are RsvdZ Default 0

Table 6.6-58 Valid Reset Signal

Reset Signal	Lane Error Status Bits
RST_LOAD_B	
RST_RSM_B	✓
RST_CFG_B	

(65) Lane Equalization Control Register (PCI_RC_LEQCTL)

Access Size : 32 bits
Offset Address : <PCI0_base> + 61BCh
Initial Value : 7F7F_7F7Fh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	Upstream Port 8.0 GTs Receiver Preset Hint for Lane 1[2:0]			Upstream Port 8.0 GTs Transmitter Preset for Lane 1[3:0]				-	Downstream Port 8.0 GTs Receiver Preset Hint for Lane 1[2:0]			Downstream Port 8.0 GTs Transmitter Preset for Lane 1[3:0]			
Initial Value	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
R/W	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	Upstream Port 8.0 GTs Receiver Preset Hint for Lane 0[2:0]			Upstream Port 8.0 GTs Transmitter Preset for Lane 0[3:0]				-	Downstream Port 8.0 GTs Receiver Preset Hint for Lane 0[2:0]			Downstream Port 8.0 GTs Transmitter Preset for Lane 0[3:0]			
Initial Value	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
R/W	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, undefined value is read. The written value will be ignored.
30 to 28	Upstream Port 8.0 GTs Receiver Preset Hint for Lane 1[2:0]	1h	Hwinit	The read value is undefined This field contains the Receiver Preset Hint value sent or received during Link Equalization. Usage of this field varies as follows: A: Downstream Port This field contains the value sent on the associated Lane during Link Equalization. This field is HwInit. B: Upstream Port, Crosslink Supported = 0b This field is intended for debug and diagnostics. It contains the value captured from the associated Lane during Link Equalization. Field is RO. Note: When crosslinks are supported, case C (below) applies and this captured information is not visible to software. Vendors are encouraged to provide an alternate mechanism to obtain this information. C: Upstream Port, Crosslink Supported = 1b This field is not used or affected by the current Link Equalization. The field value will be used if a future crosslink negotiation switches the Operating Port Direction so that case A (above) applies. This field is HwInit. The default value is 111b. RC: Default 111b EP: Input signal

Bit	Bit Name	Initial Value	R/W	Description
27 to 24	Upstream Port 8.0 GTs Transmitter Presetfor Lane 1[3:0]	1h	Hwinit	The read value is undefined This field contains the Transmit Preset value sent or received during Link Equalization. Usage of this field varies as follows: A: Downstream Port This field contains the value sent on the associated Lane during Link Equalization. This field is Hwlnit. B: Upstream Port, Crosslink Supported = 0b This field is intended for debug and diagnostics. It contains the value captured from the associated Lane during Link Equalization. This field is RO. Note: When crosslinks are supported, case C (below) applies and this captured information is not visible to software. Vendors are encouraged to provide an alternate mechanism to obtain this information. C: Upstream Port, Crosslink Supported = 1b This field is not used or affected by the current Link Equalization. The field value will be used if a future crosslink negotiation switches the Operating Port Direction so that case A (above) applies. This field is Hwlnit. The default value is 1111b. RC: Default 1111b EP: Input signal
23	-	0h	R	Reserved Whenever it is read, undefined value is read. The written value will be ignored.
22 to 20	Downstream Port 8.0 GTs Receiver Preset Hint for Lane 1[2:0]	1h	Hwinit	The read value is undefined Receiver Preset Hint may be used as a hint for receiver equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port. For an Upstream Port if Crosslink Supported is 0b, this field is RsvdP. Otherwise, this field is Hwlnit. The default value is 111b. RC: Default 111b EP: Fixed to 0 because Crosslink is not supported
19 to 16	Downstream Port 8.0 GTs Transmitter Preset for Lane 1[3:0]	1h	Hwinit	The read value is undefined Transmitter Preset used for equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port. For an Upstream Port if Crosslink Supported is 0b, this field is RsvdP. Otherwise, this field is Hwlnit. The default value is 1111b. RC: Default 1111b EP: Fixed to 0 because Crosslink is not supported
15	-	0h	R	Reserved Whenever it is read, undefined value is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
14 to 12	Upstream Port 8.0 GTs Receiver Preset Hint for Lane 0[2:0]	1h	Hwinit	The read value is undefined This field contains the Receiver Preset Hint value sent or received during Link Equalization. Usage of this field varies as follows: A: Downstream Port This field contains the value sent on the associated Lane during Link Equalization. This field is Hwlnit. B: Upstream Port, Crosslink Supported = 0b This field is intended for debug and diagnostics. It contains the value captured from the associated Lane during Link Equalization. This field is RO. Note: When crosslinks are supported, case C (below) applies and this captured information is not visible to software. Vendors are encouraged to provide an alternate mechanism to obtain this information. C: Upstream Port, Crosslink Supported = 1b This field is not used or affected by the current Link Equalization. The field value will be used if a future crosslink negotiation switches the Operating Port Direction so that case A (above) applies. This field is Hwlnit. The default value is 111b. RC: Default 111b EP: Input signal
11 to 8	Upstream Port 8.0 GTs Transmitter Preset for Lane 0[3:0]	1h	Hwinit	The read value is undefined This field contains the Transmit Preset value sent or received during Link Equalization. Usage of this field varies as follows: A: Downstream Port This field contains the value sent on the associated Lane during Link Equalization. This field is Hwlnit. B: Upstream Port, Crosslink Supported = 0b This field is intended for debug and diagnostics. It contains the value captured from the associated Lane during Link Equalization. This field is RO. Note: When crosslinks are supported, case C (below) applies and this captured information is not visible to software. Vendors are encouraged to provide an alternate mechanism to obtain this information. C: Upstream Port, Crosslink Supported = 1b This field is not used or affected by the current Link Equalization. The field value will be used if a future crosslink negotiation switches the Operating Port Direction so that case A (above) applies. This field is Hwlnit. The default value is 1111b. RC: Default 1111b EP: Input signal
7	-	0h	R	Reserved Whenever it is read, undefined value is read. The written value will be ignored.
6 to 4	Downstream Port 8.0 GTs Receiver Preset Hint for Lane 0[2:0]	1h	Hwinit	The read value is undefined Receiver Preset Hint may be used as a hint for receiver equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port. For an Upstream Port if Crosslink Supported is 0b, this field is RsvdP. Otherwise, this field is Hwlnit. The default value is 111b. RC: Default 111b EP: Fixed to 0 because Crosslink is not supported

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	Downstream Port 8.0 GTs Transmitter Preset for Lane 0[3:0]	1h	Hwinit	The read value is undefined Transmitter Preset used for equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port. For an Upstream Port if Crosslink Supported is 0b, this field is RsvdP. Otherwise, this field is Hwlnit. The default value is 1111b. RC: Default 1111b EP: Fixed to 0 because Crosslink is not supported

Table 6.6-59 Valid Reset Signal

Reset Signal	Upstream Port 8.0GT/s Receiver Preset Hint	Upstream Port 8.0GT/s Transmitter Preset	Downstream Port 8.0GT/s Receiver Preset Hint	Downstream Port 8.0GT/s Transmitter Preset	Upstream Port 8.0GT/s Receiver Preset Hint	Upstream Port 8.0GT/s Transmitter Preset
RST_LOAD_B	✓	✓	✓	✓	✓	✓
RST_RSM_B						
RST_CFG_B						

Reset Signal	Downstream Port 8.0GT/s Receiver Preset Hint	Downstream Port 8.0GT/s Transmitter Preset
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		

6.6.4.2 Register Descriptions (Endpoint Mode)

The following lists the registers incorporated in the unit.

CAUTION

The results of access to reserved bit areas, debug bit areas, and undefined areas are not guaranteed. The combination of reserved, debug, and undefined areas includes cases where the initial values will be non-zero, and the results of changes to such values are not guaranteed.

The following is the outline of categories of the register space within this unit.

AXI Bridge Registers	Address 0000h-1FFCh
Physical Layer Control/Status Registers	Address 2000h-5FFCh
PCI Express Configuration Registers (Type0)	Address 6000h-6FFCh (Function #0)
PCI Express Configuration Registers (Type0)	Address 7000h-7FFCh (Function #1)

Access to the reserved spaces above is prohibited.

6.6.4.2.1 List of AXI Bridge Registers

Unless specifically stated otherwise, byte, word, and double word access are all possible.

Access to registers is possible from the AXI-bus side and the PCIe-bus side. Attributes may vary with the register and the direction of access. In the case of registers for which attributes vary, attributes in the higher parts of the cells in the R/W column are for access from the PCIe-bus side and those in the lower parts are for access from the AXI-bus side. However, the cells are not divided into higher and lower parts in cases where access does not vary with the side proceeding with access.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Request Data Register 0	PCI_EP_REQDATA0	xxxx_xxxxh	0080h	32
Request Data Register 1	PCI_EP_REQDATA1	xxxx_xxxxh	0084h	32
Request Data Register 2	PCI_EP_REQDATA2	xxxx_xxxxh	0088h	32
Request Receive Data Register	PCI_EP_REQRCVDAT	xxxx_xxxxh	008Ch	32
Request Address Register 1	PCI_EP_REQADR1	xxxx_xxxxh	0090h	32
Request Address Register 2	PCI_EP_REQADR2	xxxx_xxxxh	0094h	32
Request Byte Enable Register	PCI_EP_REQBE	0000_000Fh	0098h	32
Request Issue Register	PCI_EP_REQISS	0000_0000h	009Ch	32
PCI INTx Out Status Register	PCI_EP_INTXOUTS	0000_0000h	0118h	32
Message Receive Interrupt Enable Register	PCI_EP_MSGRCVIE	0000_0000h	0120h	32
Message Receive Interrupt Status Register	PCI_EP_MSGRCVIS	0000_0000h	0124h	32
Message Code Register	PCI_EP_MSGCODE	0000_0000h	0130h	32
Message Data Register	PCI_EP_MSGDATA	0000_0000h	0134h	32
Message Header 3rdDW Register	PCI_EP_MSGH3DW	0000_0000h	0138h	32
Message Header 4thDW Register	PCI_EP_MSGH4DW	0000_0000h	013Ch	32
Interrupt Table Register	PCI_EP_INTTABLE	00xx_0000h	0140h	32
PCIe Event Interrupt Enable 0 Register	PCI_EP_PEIE0	0000_0000h	0200h	32
PCIe Event Interrupt Status 0 Register	PCI_EP_PEIS0	0000_0000h	0204h	32
PCIe Event Interrupt Enable 1 Register	PCI_EP_PEIE1	0000_0000h	0208h	32
PCIe Event Interrupt Status 1 Register	PCI_EP_PEIS1	0000_0000h	020Ch	32
AXI Master Error Interrupt Enable Register	PCI_EP_AMEIE	0000_0000h	0210h	32
AXI Master Error Interrupt Status Register	PCI_EP_AMEIS	0000_0000h	0214h	32
AXI Slave Error Interrupt Enable 1 Register	PCI_EP_ASEIE1	0000_0000h	0220h	32
AXI Slave Error Interrupt Status 1 Register	PCI_EP_ASEIS1	0000_0000h	0224h	32
AXI Slave Error Interrupt Status 3 Register	PCI_EP_ASEIS3	0000_0000h	0230h	32
PCIe Event Interrupt Enable 2 Register	PCI_EP_PEIE2	0000_0000h	0240h	32
PCIe Event Interrupt Status 2 Register	PCI_EP_PEIS2	0000_0000h	0244h	32
Permission Register	PCI_EP_PERM	0000_0000h	0300h	32
Reset Register	PCI_EP_RESET	0000_00xxh	0310h	32
Mode Set 0 Register	PCI_EP_MSET0	2001_2000h	0314h	32
Mode Set 1 Register	PCI_EP_MSET1	0000_33F2h	0318h	32
Mode Set 3 Register	PCI_EP_MSET3	0000_0000h	0380h	32
Mode Set 4 Register	PCI_EP_MSET4	0000_0000h	0384h	32
Mode Set 5 Register	PCI_EP_MSET5	0000_0000h	0388h	32
Mode Status 0 Registers	PCI_EP_MSTA0	0000_0000h	0390h	32
PCIe Core Mode Set 1 Register	PCI_EP_PCMSET1	00FA_00F2h	0400h	32
PCIe Core Control 1 Register	PCI_EP_PCCTRL1	0000_0000h	0404h	32
PCIe Core Status 1 Register	PCI_EP_PCSTAT1	0x0x_xxxxh	0408h	32
PCIe Core Control 2 Register	PCI_EP_PCCTRL2	003E_0000h	0410h	32
PCIe Core Status 2 Register	PCI_EP_PCSTAT2	xxxx_xxxxh	0414h	32
PCIe Core Status 5 Register	PCI_EP_PCSTAT5	0000_0X00h	042Ch	32
DMA Interrupt Vector 0 Register	PCI_EP_DMAINTVEC0	0000_0000h	04D0h	32
DMA Interrupt Vector 1 Register	PCI_EP_DMAINTVEC1	0000_0000h	04D4h	32
DMAC Control Register	PCI_EP_DMACTRL	0000_0000h	0800h	32
DMAC Interrupt Enable Register	PCI_EP_DMAINTE	0000_0000h	0808h	32
DMAC Interrupt Status Register	PCI_EP_DMAINTS	0000_0000h	080Ch	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
DMAC Channel Control Register 0	PCI_EP_DMACHCTL0	0000_0000h	0900h	32
Descriptor Start Address (Lower) Register 0	PCI_EP_DPSADRL0	0000_0000h	0908h	32
Descriptor Start Address (Higher) Registers 0	PCI_EP_DPSADRU0	0000_0000h	090Ch	32
QUE Entry Register 0	PCI_EP_QUEE0	0000_0000h	0910h	32
DMA Descriptor Control (Descriptor 00h) Register 0	PCI_EP_DMADPCTL0	0000_0000h	0920h	32
DMA Transaction Control (Descriptor 04h) Register 0	PCI_EP_DMATCTL0	0000_0000h	0924h	32
DMA Size (Descriptor 08h) Register 0	PCI_EP_DMASIZE0	0000_0000h	0928h	32
DMA Source Lower Address (Descriptor 10h) Register 0	PCI_EP_DMASLA0	0000_0000h	0930h	32
DMA Source Higher Address (Descriptor 14h) Register 0	PCI_EP_DMASUA0	0000_0000h	0934h	32
DMA Destination Lower Address (Descriptor 18h) Register 0	PCI_EP_DMADESTLA0	0000_0000h	0938h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 0	PCI_EP_DMADESTUA0	0000_0000h	093Ch	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 0	PCI_EP_DMADPLLP0	0000_0000h	0940h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 0	PCI_EP_DMADPULP0	0000_0000h	0944h	32
DMA Rest Size Register 0	PCI_EP_DMARESTSIZ0	0000_0000h	0950h	32
AXI Request Address (Lower) Register 0	PCI_EP_AREQAL0	0000_0000h	0960h	32
AXI Request Address (Higher) Register 0	PCI_EP_AREQAU0	0000_0000h	0964h	32
PCIe Request Address (Lower) Register 0	PCI_EP_PREQAL0	0000_0000h	0968h	32
PCIe Request Address (Higher) Register 0	PCI_EP_PREQAU0	0000_0000h	096Ch	32
QUE Status Register 0	PCI_EP_QUESTA0	0000_0000h	0970h	32
DMAC Error Status Register 0	PCI_EP_DMACESTA0	0000_0000h	0978h	32
DMAC Channel Control Register 1	PCI_EP_DMACHCTL1	0000_0000h	0980h	32
Descriptor Start Address (Lower) Register 1	PCI_EP_DPSADRL1	0000_0000h	0988h	32
Descriptor Start Address (Higher) Registers 1	PCI_EP_DPSADRU1	0000_0000h	098Ch	32
QUE Entry Register 1	PCI_EP_QUEE1	0000_0000h	0990h	32
DMA Descriptor Control (Descriptor 00h) Register 1	PCI_EP_DMADPCTL1	0000_0000h	09A0h	32
DMA Transaction Control (Descriptor 04h) Register 1	PCI_EP_DMATCTL1	0000_0000h	09A4h	32
DMA Size (Descriptor 08h) Register 1	PCI_EP_DMASIZE1	0000_0000h	09A8h	32
DMA Source Lower Address (Descriptor 10h) Register 1	PCI_EP_DMASLA1	0000_0000h	09B0h	32
DMA Source Higher Address (Descriptor 14h) Register 1	PCI_EP_DMASUA1	0000_0000h	09B4h	32
DMA Destination Lower Address (Descriptor 18h) Register 1	PCI_EP_DMADESTLA1	0000_0000h	09B8h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 1	PCI_EP_DMADESTUA1	0000_0000h	09BCh	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 1	PCI_EP_DMADPLLP1	0000_0000h	09C0h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 1	PCI_EP_DMADPULP1	0000_0000h	09C4h	32
DMA Rest Size Register 1	PCI_EP_DMARESTSIZ1	0000_0000h	09D0h	32
AXI Request Address (Lower) Register 1	PCI_EP_AREQAL1	0000_0000h	09E0h	32
AXI Request Address (Higher) Register 1	PCI_EP_AREQAU1	0000_0000h	09E4h	32
PCIe Request Address (Lower) Register 1	PCI_EP_PREQAL1	0000_0000h	09E8h	32
PCIe Request Address (Higher) Register 1	PCI_EP_PREQAU1	0000_0000h	09ECh	32
QUE Status Register 1	PCI_EP_QUESTA1	0000_0000h	09F0h	32
DMAC Error Status Register 1	PCI_EP_DMACESTA1	0000_0000h	09F8h	32
DMAC Channel Control Register 2	PCI_EP_DMACHCTL2	0000_0000h	0A00h	32
Descriptor Start Address (Lower) Register 2	PCI_EP_DPSADRL2	0000_0000h	0A08h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Descriptor Start Address (Higher) Registers 2	PCI_EP_DPSADRU2	0000_0000h	0A0Ch	32
QUE Entry Register 2	PCI_EP_QUEE2	0000_0000h	0A10h	32
DMA Descriptor Control (Descriptor 00h) Register 2	PCI_EP_DMADPCTL2	0000_0000h	0A20h	32
DMA Transaction Control (Descriptor 04h) Register 2	PCI_EP_DMATCTL2	0000_0000h	0A24h	32
DMA Size (Descriptor 08h) Register 2	PCI_EP_DMASIZE2	0000_0000h	0A28h	32
DMA Source Lower Address (Descriptor 10h) Register 2	PCI_EP_DMASLA2	0000_0000h	0A30h	32
DMA Source Higher Address (Descriptor 14h) Register 2	PCI_EP_DMASUA2	0000_0000h	0A34h	32
DMA Destination Lower Address (Descriptor 18h) Register 2	PCI_EP_DMADESTLA2	0000_0000h	0A38h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 2	PCI_EP_DMADESTUA2	0000_0000h	0A3Ch	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 2	PCI_EP_DMADPLLP2	0000_0000h	0A40h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 2	PCI_EP_DMADPULP2	0000_0000h	0A44h	32
DMA Rest Size Register 2	PCI_EP_DMARESTSIZ2	0000_0000h	0A50h	32
AXI Request Address (Lower) Register 2	PCI_EP_AREQAL2	0000_0000h	0A60h	32
AXI Request Address (Higher) Register 2	PCI_EP_AREQAU2	0000_0000h	0A64h	32
PCIe Request Address (Lower) Register 2	PCI_EP_PREQAL2	0000_0000h	0A68h	32
PCIe Request Address (Higher) Register 2	PCI_EP_PREQAU2	0000_0000h	0A6Ch	32
QUE Status Register 2	PCI_EP_QUESTA2	0000_0000h	0A70h	32
DMAC Error Status Register 2	PCI_EP_DMACESTA2	0000_0000h	0A78h	32
DMAC Channel Control Register 3	PCI_EP_DMACHCTL3	0000_0000h	0A80h	32
Descriptor Start Address (Lower) Register 3	PCI_EP_DPSADRL3	0000_0000h	0A88h	32
Descriptor Start Address (Higher) Registers 3	PCI_EP_DPSADRU3	0000_0000h	0A8Ch	32
QUE Entry Register 3	PCI_EP_QUEE3	0000_0000h	0A90h	32
DMA Descriptor Control (Descriptor 00h) Register 3	PCI_EP_DMADPCTL3	0000_0000h	0AA0h	32
DMA Transaction Control (Descriptor 04h) Register 3	PCI_EP_DMATCTL3	0000_0000h	0AA4h	32
DMA Size (Descriptor 08h) Register 3	PCI_EP_DMASIZE3	0000_0000h	0AA8h	32
DMA Source Lower Address (Descriptor 10h) Register 3	PCI_EP_DMASLA3	0000_0000h	0AB0h	32
DMA Source Higher Address (Descriptor 14h) Register 3	PCI_EP_DMASUA3	0000_0000h	0AB4h	32
DMA Destination Lower Address (Descriptor 18h) Register 3	PCI_EP_DMADESTLA3	0000_0000h	0AB8h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 3	PCI_EP_DMADESTUA3	0000_0000h	0ABCh	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 3	PCI_EP_DMADPLLP3	0000_0000h	0AC0h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 3	PCI_EP_DMADPULP3	0000_0000h	0AC4h	32
DMA Rest Size Register 3	PCI_EP_DMARESTSIZ3	0000_0000h	0AD0h	32
AXI Request Address (Lower) Register 3	PCI_EP_AREQAL3	0000_0000h	0AE0h	32
AXI Request Address (Higher) Register 3	PCI_EP_AREQAU3	0000_0000h	0AE4h	32
PCIe Request Address (Lower) Register 3	PCI_EP_PREQAL3	0000_0000h	0AE8h	32
PCIe Request Address (Higher) Register 3	PCI_EP_PREQAU3	0000_0000h	0AECh	32
QUE Status Register 3	PCI_EP_QUESTA3	0000_0000h	0AF0h	32
DMAC Error Status Register 3	PCI_EP_DMACESTA3	0000_0000h	0AF8h	32
DMAC Channel Control Register 4	PCI_EP_DMACHCTL4	0000_0000h	0B00h	32
Descriptor Start Address (Lower) Register 4	PCI_EP_DPSADRL4	0000_0000h	0B08h	32
Descriptor Start Address (Higher) Registers 4	PCI_EP_DPSADRU4	0000_0000h	0B0Ch	32
QUE Entry Register 4	PCI_EP_QUEE4	0000_0000h	0B10h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
DMA Descriptor Control (Descriptor 00h) Register 4	PCI_EP_DMADPCTL4	0000_0000h	0B20h	32
DMA Transaction Control (Descriptor 04h) Register 4	PCI_EP_DMATCTL4	0000_0000h	0B24h	32
DMA Size (Descriptor 08h) Register 4	PCI_EP_DMASIZE4	0000_0000h	0B28h	32
DMA Source Lower Address (Descriptor 10h) Register 4	PCI_EP_DMASLA4	0000_0000h	0B30h	32
DMA Source Higher Address (Descriptor 14h) Register 4	PCI_EP_DMASUA4	0000_0000h	0B34h	32
DMA Destination Lower Address (Descriptor 18h) Register 4	PCI_EP_DMADESTLA4	0000_0000h	0B38h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 4	PCI_EP_DMADESTUA4	0000_0000h	0B3Ch	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 4	PCI_EP_DMADPLLP4	0000_0000h	0B40h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 4	PCI_EP_DMADPULP4	0000_0000h	0B44h	32
DMA Rest Size Register 4	PCI_EP_DMARESTSIZ4	0000_0000h	0B50h	32
AXI Request Address (Lower) Register 4	PCI_EP_AREQAL4	0000_0000h	0B60h	32
AXI Request Address (Higher) Register 4	PCI_EP_AREQAU4	0000_0000h	0B64h	32
PCIe Request Address (Lower) Register 4	PCI_EP_PREQAL4	0000_0000h	0B68h	32
PCIe Request Address (Higher) Register 4	PCI_EP_PREQAU4	0000_0000h	0B6Ch	32
QUE Status Register 4	PCI_EP_QUESTA4	0000_0000h	0B70h	32
DMAC Error Status Register 4	PCI_EP_DMACESTA4	0000_0000h	0B78h	32
DMAC Channel Control Register 5	PCI_EP_DMACHCTL5	0000_0000h	0B80h	32
Descriptor Start Address (Lower) Register 5	PCI_EP_DPSADRL5	0000_0000h	0B88h	32
Descriptor Start Address (Higher) Registers 5	PCI_EP_DPSADRU5	0000_0000h	0B8Ch	32
QUE Entry Register 5	PCI_EP_QUEE5	0000_0000h	0B90h	32
DMA Descriptor Control (Descriptor 00h) Register 5	PCI_EP_DMADPCTL5	0000_0000h	0BA0h	32
DMA Transaction Control (Descriptor 04h) Register 5	PCI_EP_DMATCTL5	0000_0000h	0BA4h	32
DMA Size (Descriptor 08h) Register 5	PCI_EP_DMASIZE5	0000_0000h	0BA8h	32
DMA Source Lower Address (Descriptor 10h) Register 5	PCI_EP_DMASLA5	0000_0000h	0BB0h	32
DMA Source Higher Address (Descriptor 14h) Register 5	PCI_EP_DMASUA5	0000_0000h	0BB4h	32
DMA Destination Lower Address (Descriptor 18h) Register 5	PCI_EP_DMADESTLA5	0000_0000h	0BB8h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 5	PCI_EP_DMADESTUA5	0000_0000h	0BBCh	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 5	PCI_EP_DMADPLLP5	0000_0000h	0BC0h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 5	PCI_EP_DMADPULP5	0000_0000h	0BC4h	32
DMA Rest Size Register 5	PCI_EP_DMARESTSIZ5	0000_0000h	0BD0h	32
AXI Request Address (Lower) Register 5	PCI_EP_AREQAL5	0000_0000h	0BE0h	32
AXI Request Address (Higher) Register 5	PCI_EP_AREQAU5	0000_0000h	0BE4h	32
PCIe Request Address (Lower) Register 5	PCI_EP_PREQAL5	0000_0000h	0BE8h	32
PCIe Request Address (Higher) Register 5	PCI_EP_PREQAU5	0000_0000h	0BECh	32
QUE Status Register 5	PCI_EP_QUESTA5	0000_0000h	0BF0h	32
DMAC Error Status Register 5	PCI_EP_DMACESTA5	0000_0000h	0BF8h	32
DMAC Channel Control Register 6	PCI_EP_DMACHCTL6	0000_0000h	0C00h	32
Descriptor Start Address (Lower) Register 6	PCI_EP_DPSADRL6	0000_0000h	0C08h	32
Descriptor Start Address (Higher) Registers 6	PCI_EP_DPSADRU6	0000_0000h	0C0Ch	32
QUE Entry Register 6	PCI_EP_QUEE6	0000_0000h	0C10h	32
DMA Descriptor Control (Descriptor 00h) Register 6	PCI_EP_DMADPCTL6	0000_0000h	0C20h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
DMA Transaction Control (Descriptor 04h) Register 6	PCI_EP_DMATCTL6	0000_0000h	0C24h	32
DMA Size (Descriptor 08h) Register 6	PCI_EP_DMASIZE6	0000_0000h	0C28h	32
DMA Source Lower Address (Descriptor 10h) Register 6	PCI_EP_DMASLA6	0000_0000h	0C30h	32
DMA Source Higher Address (Descriptor 14h) Register 6	PCI_EP_DMASUA6	0000_0000h	0C34h	32
DMA Destination Lower Address (Descriptor 18h) Register 6	PCI_EP_DMADESTLA6	0000_0000h	0C38h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 6	PCI_EP_DMADESTUA6	0000_0000h	0C3Ch	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 6	PCI_EP_DMADPLLP6	0000_0000h	0C40h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 6	PCI_EP_DMADPULP6	0000_0000h	0C44h	32
DMA Rest Size Register 6	PCI_EP_DMARESTSIZ6	0000_0000h	0C50h	32
AXI Request Address (Lower) Register 6	PCI_EP_AREQAL6	0000_0000h	0C60h	32
AXI Request Address (Higher) Register 6	PCI_EP_AREQAU6	0000_0000h	0C64h	32
PCIe Request Address (Lower) Register 6	PCI_EP_PREQAL6	0000_0000h	0C68h	32
PCIe Request Address (Higher) Register 6	PCI_EP_PREQAU6	0000_0000h	0C6Ch	32
QUE Status Register 6	PCI_EP_QUESTA6	0000_0000h	0C70h	32
DMAC Error Status Register 6	PCI_EP_DMACESTA6	0000_0000h	0C78h	32
DMAC Channel Control Register 7	PCI_EP_DMACHCTL7	0000_0000h	0C80h	32
Descriptor Start Address (Lower) Register 7	PCI_EP_DPSADRL7	0000_0000h	0C88h	32
Descriptor Start Address (Higher) Registers 7	PCI_EP_DPSADRU7	0000_0000h	0C8Ch	32
QUE Entry Register 7	PCI_EP_QUEE7	0000_0000h	0C90h	32
DMA Descriptor Control (Descriptor 00h) Register 7	PCI_EP_DMADPCTL7	0000_0000h	0CA0h	32
DMA Transaction Control (Descriptor 04h) Register 7	PCI_EP_DMATCTL7	0000_0000h	0CA4h	32
DMA Size (Descriptor 08h) Register 7	PCI_EP_DMASIZE7	0000_0000h	0CA8h	32
DMA Source Lower Address (Descriptor 10h) Register 7	PCI_EP_DMASLA7	0000_0000h	0CB0h	32
DMA Source Higher Address (Descriptor 14h) Register 7	PCI_EP_DMASUA7	0000_0000h	0CB4h	32
DMA Destination Lower Address (Descriptor 18h) Register 7	PCI_EP_DMADESTLA7	0000_0000h	0CB8h	32
DMA Destination Higher Address (Descriptor 1Ch) Register 7	PCI_EP_DMADESTUA7	0000_0000h	0CBCh	32
DMA Descriptor Lower Link Pointer (Descriptor 20h) Register 7	PCI_EP_DMADPLLP7	0000_0000h	0CC0h	32
DMA Descriptor Higher Link Pointer (Descriptor 24h) Register 7	PCI_EP_DMADPULP7	0000_0000h	0CC4h	32
DMA Rest Size Register 7	PCI_EP_DMARESTSIZ7	0000_0000h	0CD0h	32
AXI Request Address (Lower) Register 7	PCI_EP_AREQAL7	0000_0000h	0CE0h	32
AXI Request Address (Higher) Register 7	PCI_EP_AREQAU7	0000_0000h	0CE4h	32
PCIe Request Address (Lower) Register 7	PCI_EP_PREQAL7	0000_0000h	0CE8h	32
PCIe Request Address (Higher) Register 7	PCI_EP_PREQAU7	0000_0000h	0CECh	32
QUE Status Register 7	PCI_EP_QUESTA7	0000_0000h	0CF0h	32
DMAC Error Status Register 7	PCI_EP_DMACESTA7	0000_0000h	0CF8h	32
AXI Window Base (Lower) m Register (Function #n)	PCI_EP_AWBASELm_Fn	0000_0000h	1000h + Offset: 00h/ 20h/ 40h/ … / E0h	32
AXI Window Base (Higher) m Register (Function #n)	PCI_EP_AWBASEUm_Fn	0000_0000h	1000h + Offset: 04h/ 24h/ 44h/ … / E4h	32
AXI Window Mask (Lower) m Register (Function #n)	PCI_EP_AWMASKLm_Fn	0000_0FFFh	1000h + Offset: 08h/ 28h/ 48h/ … / E8h	32
AXI Window Mask (Higher) m Register (Function #n)	PCI_EP_AWMASKUm_Fn	0000_0000h	1000h + Offset: 0Ch/ 2Ch/ 4Ch/ … / ECh	32
AXI Destination (Lower) m Register (Function #n)	PCI_EP_ADESTLm_Fn	0000_0000h	1000h + Offset: 10h/ 30h/ 50h/ 70h/ … / F0h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
AXI Destination (Higher) m Register (Function #n)	PCI_EP_ADESTUm_Fn	0000_0000h	1000h + Offset: 14h/ 34h/ 54h/ 74h/ … / F4h	32
PCIe Window Base (Lower) m Register (Function #n)	PCI_EP_PWBASELm_Fn	0000_0000h	1000h + Offset: 100h/ 120h/ 140h/ … / 1E0h	32
PCIe Window Base (Higher) m Register (Function #n)	PCI_EP_PWBASEUm_Fn	0000_0000h	1000h + Offset: 104h/ 124h/ 144h/ … / 1E4h	32
PCIe Window Mask (Lower) m Register (Function #n)	PCI_EP_PWMASKLm_Fn	0000_0FFFh	1000h + Offset: 108h/ 128h/ 148h/ … / 1E8h	32
PCIe Window Mask (Higher) m Register (Function #n)	PCI_EP_PWMASKUm_Fn	0000_0000h	1000h + Offset: 10Ch/ 12Ch/ 14Ch/ … / 1ECh	32
PCIe Destination m (Lower) Register (Function #n)	PCI_EP_PDESTLOm_Fn	0000_0000h	1000h + Offset: 110h/ 130h/ 150h/ 170h/ … / 1F0h	32
PCIe Destination m (Higher) Register (Function #n)	PCI_EP_PDESTUPm_Fn	0000_0000h	1000h + Offset: 114h/ 134h/ 154h/ 174h/ … / 1F4h	32

6.6.4.2.2 List of PCI Express Configuration Registers (Type0)

The table lists the PCI Express configuration registers (Type 0).

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Vendor and Device ID (Function #n) Register	PCI_EP_VID_Fn	0000_0000h	6000h	32
Command and Status (Function #n)	PCI_EP_COM_STA_Fn	0010_0000h	6004h	32
Revision ID and Class Code (Function #n)	PCI_EP_RID_CC_Fn	0000_0000h	6008h	32
Cache Line and Header Type (Function #n)	PCI_EP_CL_HT	0080_0000h	600Ch	32
Base Address Register 0 (Function #n)	PCI_EP_BAR0_Fn	0000_0004h	6010h	32
Base Address Register 1 (Function #n)	PCI_EP_BAR1_Fn	0000_0000h	6014h	32
Base Address Register 2 (Function #n)	PCI_EP_BAR2_Fn	0000_000Ch	6018h	32
Base Address Register 3 (Function #n)	PCI_EP_BAR3_Fn	0000_0000h	601Ch	32
Base Address Register 4 (Function #n)	PCI_EP_BAR4_Fn	0000_0004h	6020h	32
Base Address Register 5 (Function #n)	PCI_EP_BAR5_Fn	0000_0000h	6024h	32
Subsystem ID (Function #n)	PCI_EP_SUBSID_Fn	0000_0000h	602Ch	32
Capabilities Pointer (Function #n)	PCI_EP_CP_Fn	0000_0040h	6034h	32
Interrupt Register (Function #n)	PCI_EP_INT_Fn	0000_0100h	603Ch	32
PM Capabilities (Function #n)	PCI_EP_PMC_Fn	4803_E001h	6040h	32
PM Status/Control (Function #n)	PCI_EP_PMSC_Fn	0000_0008h	6044h	32
PCI Express Capability (Function #n)	PCI_EP_PCIEC_Fn	0002_0010h	6060h	32
Device Capabilities (Function #n)	PCI_EP_DEVC_Fn	1000_8FC1h	6064h	32
Device Control/Status (Function #n)	PCI_EP_DEVCS_Fn	0000_2010h	6068h	32
Link Capabilities (Function #n)	PCI_EP_LINKC_Fn	0042_CC23h	606Ch	32
Link Control/Status (Function #n)	PCI_EP_LINKCS_Fn	1000_0000h	6070h	32
Device Capabilities 2 (Function #n)	PCI_EP_DEVC2_Fn	0000_0012h	6084h	32
Device Control 2/Status 2 (Function #n)	PCI_EP_DEVCS2_Fn	0000_0000h	6088h	32
Link Capabilities 2 (Function #n)	PCI_EP_LINKC2_Fn	0000_000Eh	608Ch	32
Link Control 2/Status 2 (Function #n)	PCI_EP_LINCS2_Fn	0000_0003h	6090h	32
Base Address Register Mask00 (Lower) (Function #n)	PCI_EP_BARMSK00L_Fn	1FFF_FFFFh	60A0h	32
Base Address Register Mask00 (Higher) (Function #n)	PCI_EP_BARMSK00U_Fn	0000_0000h	60A4h	32
Base Address Register Mask01 (Lower) (Function #n)	PCI_EP_BARMSK01L_Fn	FFFF_FFFFh	60A8h	32
Base Address Register Mask01 (Higher) (Function #n)	PCI_EP_BARMSK01U_Fn	F0: 0000_007Fh F1: 0000_00FFh	60ACh	32
Base Address Register Mask02 (Lower) (Function #n)	PCI_EP_BARMSK02L_Fn	0000_1FFFh	60B0h	32
Base Address Register Mask02 (Higher) (Function #n)	PCI_EP_BARMSK02U_Fn	0000_0000h	60B4h	32
Base Size 00/01 (Function #n)	PCI_EP_BSIZE00_01_Fn	0000_0000h	60C8h	32
Base Size 02/03 (Function #n)	PCI_EP_BSIZE02_03_Fn	0000_0000h	60CCh	32
Base Size 04/05 (Function #n)	PCI_EP_BSIZE04_05_Fn	0000_0000h	60D0h	32
Base Size 06 (Function #n)	PCI_EP_BSIZE06_Fn	0000_0000h	60D4h	32
Type Supported 00/01/02 (Function #n)	PCI_EP_TSUPPORT00_01_02_Fn	0033_3333h	60D8h	32
MSI Capability (Function #n)	PCI_EP_MSICAP_Fn	018A_6005h	60E0h	32
Message Address (Function #n)	PCI_EP_MSGADR_Fn	0000_0000h	60E4h	32
Message Higher Address (Function #n)	PCI_EP_MSGUADR_Fn	0000_0000h	60E8h	32
Message Data (Function #n)	PCI_EP_MSGDAT_Fn	0000_0000h	60ECh	32
Mask Bits (Function #n)	PCI_EP_MSKBIT_Fn	0000_0000h	60F0h	32
Pending Bits (Function #n)	PCI_EP_PENDBIT_Fn	0000_0000h	60F4h	32
Advanced Error Reporting Capability (Function #n)	PCI_EP_ADVERC_Fn	1501_0001h	6100h	32
Uncorrectable Error Status Register (Function #n)	PCI_EP_UNCESTS_Fn	0000_0000h	6104h	32
Uncorrectable Error Mask Register (Function #n)	PCI_EP_UNCEMASK_Fn	0000_0000h	6108h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Uncorrectable Error Severity Register (Function #n)	PCI_EP_UNCESVY_Fn	0046_2030h	610Ch	32
Correctable Error Status Register (Function #n)	PCI_EP_CESTS_Fn	0000_0000h	6110h	32
Correctable Error Mask Register (Function #n)	PCI_EP_CEMASK_Fn	0000_2000h	6114h	32
Advanced Error Capabilities and Control Register (Function #n)	PCI_EP_ADVECC_Fn	0000_00A0h	6118h	32
Header Log Register 0 (Function #n)	PCI_EP_HLOG0_Fn	0000_0000h	611Ch	32
Header Log Register 1 (Function #n)	PCI_EP_HLOG1_Fn	0000_0000h	6120h	32
Header Log Register 2 (Function #n)	PCI_EP_HLOG2_Fn	0000_0000h	6124h	32
Header Log Register 3 (Function #n)	PCI_EP_HLOG3_Fn	0000_0000h	6128h	32
Device Serial Number Extended Capability Register (Function #n)	PCI_EP_DEVSNEXTC_Fn	1B01_0003h	6150h	32
Serial Number Register (Lower DW) (Function #n)	PCI_EP_SNL_Fn	0000_0000h	6154h	32
Serial Number Register (Higher DW) (Function #n)	PCI_EP_SNU_Fn	0000_0000h	6158h	32
Secondary PCI Express Extended Capability Header (Function #0)	PCI_EP_SPEECH_F0	0001_0019h	61B0h	32
Link Control 3 Register (Function #0)	PCI_EP_LINC3_F0	0000_0000h	61B4h	32
Lane Error Status Register (Function #0)	PCI_EP_LESTA_F0	0000_0000h	61B8h	32
Lane Equalization Control Register (Function #0)	PCI_EP_LEQCTL_F0	xx00_xx00h	61BCh	32

6.6.4.2.3 AXI Bridge Register Descriptions

The function of each register is described below.

Registers can be accessed from the AXI and PCIe. Depending on the register, attributes may vary with the direction of access. In the case of registers for which attributes vary, attributes in the higher parts of the cells in the R/W column are for access from the PCIe-bus side and those in the lower parts are for access from the AXI-bus side. However, the cells are not divided into higher and lower parts in cases where access does not vary with the side proceeding with access.

(1) Request Data Register m (PCI_EP_REQDATAm) (m = 0 to 2)

This register issues various requests. Write access to this register is only possible from AXI. In case of write access from PCIe, it returns a response and ends normally without writing data to this register. Read access is possible from both AXI and PCIe.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0080h/ 84h/ 88h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Request Data[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(AXI)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Request Data[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(AXI)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Request Data[31:0]	xh	R (RW)	Issue various requests to PCIe

Note: x = Undefined

Table 6.6-60 Details of Each Requests

	Request Data Register 0	Request Data Register 1	Request Data Register 2
Zero-Length Read Request	Invalid	Invalid	Invalid
Message Request	3rd header	4th header	Invalid
Message Request with data payload	3rd header	4th header	Message data

Note: The bits should be set to 0 for the requests indicated as “Invalid”.

(2) Request Receive Data Register (PCI_EP_REQRCVDAT)

This register indicates the data read on reception of the completion response after issuing a read request.

Access Size : 32 bits
Offset Address : <PCI0_base> + 008Ch
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Request Receive Data[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Request Receive Data[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Request Receive Data[31:0]	xh	R	After issuing a read request, the data read on reception of the completion response are set in these bits. However, these bits are invalid for Zero-Length Read and any kind of various write requests.

Note: x = Undefined

(3) Request Address Register 1 (PCI_EP_REQADR1)

This register issues requests. Write access to this register is only possible from AXI. In case of write access from PCIe, it returns a response and ends normally without writing data to this register. Read access is possible from both AXI and PCIe.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0090h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Request Address[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(AXI)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Request Address[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(AXI)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Request Address[31:0]	xh	R (RW)	Set the Address, etc. when issuing a Request.

Note: x = Undefined

Table 6.6-61 Details of Each Requests

Request Address Register 1	[31:27]	[26:24]	[23:19]	[18:16]
Zero-Length Read Request	Address			
Message Request	Reserved	Routing Type	Reserved	Reserved
Message Request with data payload	Reserved	Routing Type	Reserved	Reserved

Request Address Register 1	[15:12]	[11:8]	[7:2]	[1:0]
Zero-Length Read Request	Address			Reserved
Message Request	Reserved	Reserved	Message Code	
Message Request with data payload	Reserved	Reserved	Message Code	

Note: The bits should be set to 0 for the requests indicated as “Reserved”.

(4) Request Address Register 2 (PCI_EP_REQADR2)

This register issues requests. Write access to this register is only possible from AXI. In case of write access from PCIe, it returns a response and ends normally without writing data to this register. Read access is possible from both AXI and PCIe.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0094h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Request Address63:32[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(AXI)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Request Address63:32[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(AXI)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Request Address63:32 [31:0]	xh	R (RW)	Set the Address, etc. when issuing a Request.

Note: x = Undefined

Table 6.6-62 Details of Each Requests

	[31:0]
Zero-Length Read Request	Address
Message Request	Invalid
Message Request with data payload	Invalid

Note: The bits should be set to 0 for the requests indicated as “Invalid”.

(5) Request Byte Enable Register (PCI_EP_REQBE)

This register specifies the first byte enable bit within the TLP header (1st DW byte) when issuing a request to PCIe. Write access to this register is only possible from AXI. In case of write access from PCIe, it returns a response and ends normally without writing data to this register. Read access is possible from both AXI and PCIe.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0098h
Initial Value : 0000_000Fh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(AXI)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	Request Byte Enable[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(AXI)	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	Request Byte Enable[3:0]	Fh	R (RW)	Specify Byte Enable when issuing Cfg Request as required. Normally use 1111b. 1b: Byte Enable enabled 0b: Byte Enable disabled

Table 6.6-63 Details of Each Requests

	[3:0]
Zero-Length Read Request	0000b
Message Request	Invalid (1111b)
Message Request with data payload	Invalid (1111b)

(6) Request Issue Register (PCI_EP_REQISS)

This register issues a request to PCIe. Write access to this register is only possible from AXI. In case of write access from PCIe, it returns a response and ends normally without writing data to this register. Read access is possible from both AXI and PCIe.

Access Size : 32 bits
Offset Address : <PCI0_base> + 009Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	Request Rejection	MOR CD PERR	MOR CH PERR	MOR EP ERR	MOR STATUS[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(AXI)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	FUNC[2:0]			TR TYPE[3:0]				-	-	-	-	-	-	-	Request Issue
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(AXI)	R	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	Request Rejection	0h	R	Indicates that the stopped or hibernated state of the PCI Express transmit (TX) side was detected and the process was forcibly terminated. 0b: Normal state (Request issued) 1b: Rejection
21	MOR CD PERR	0h	R	It is set to 1 when a data error occurs in the Completion TLP for the Non-Posted request issued by this register. Normally not used. It is not updated at the time of Posted request.
20	MOR CH PERR	0h	R	It is set to 1 when a header error occurs in the Completion TLP for the Non-Posted request issued by this register. Normally not used. It is not updated at the time of Posted request.
19	MOR EP ERR	0h	R	It is set to 1 when a Poisoned Completion TLP for a Non-Posted request issued by this register is received. Normally not used. It is not updated at the time of Posted request.
18 to 16	MOR STATUS[2:0]	0h	R	This register retains the MOR Status of Completion TLP for Non-Posted requests issued. It is not updated at the time of Posted request. 000b: Successful Completion (SC) 001b: Unsupported Request (UR) 010b: Configuration Request Retry Status (CRS) (not supported) 011b: Completion Timeout 100b: Completer Abort (CA) 101b: Unexpected Completion and mismatched type (Lock Completion responds to non-Lock Request) 110b: Reserved 111b: Overrun Completion length
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 12	FUNC[2:0]	0h	R (RW)	Set the function of Request.

Bit	Bit Name	Initial Value	R/W	Description
11 to 8	TR TYPE[3:0]	0h	R (RW)	Sets the type of Request. Refer to **Table 6.6-64** for details.
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	Request Issue	0h	R (RW)	When writing: 1b: Request issuance 0b: No operation When reading: 1b: Processing Request (Indicates that the issued Request is being processed.) 0b: Request can be accepted (Indicates that the processing of the issued Request has ended.)

Table 6.6-64 Details of Each Requests

	TR Type		Device Type	
	[11:8]	Posted/Non-Posted	Root Complex	Endpoint
Zero-Length Read Request	0000b (0h)	Non-posted	Issuable	Issuable
Configuration Read Type0	0100b (4h)	Non-posted	Issuable	Issuing prohibited
Configuration Write Type0	0101b (5h)	Non-posted	Issuable	Issuing prohibited
Configuration Read Type1	0110b (6h)	Non-posted	Issuable	Issuing prohibited
Configuration Write Type1	0111b (7h)	Non-posted	Issuable	Issuing prohibited
Message Request	1000b (8h)	Posted	Issuable	Issuable
Message Request with data payload	1001b (9h)	Posted	Issuable	Issuable
	Others	—	Issuing prohibited	Issuing prohibited

(7) PCI INTx Out Status Register (PCI_EP_INTXOUTS)

This register can confirm the PCI INTx status issued by the unit with the interrupt input signal (INTX_EP_F*: active high). In addition, the value of this register is not reflected if the INTx message is asserted or deasserted by issuing a special request (prohibited operation). This register is only valid in endpoint mode.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0118h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	INTD Status	INTC Status	INTB Status	INTA Status
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	INTD Status	0h	R	It is set by sending an Assert INTD Message and cleared by sending a Deassert INTD Message. 0b: Deassert 1b: Assert
2	INTC Status	0h	R	It is set by sending an Assert INTC Message and cleared by sending a Deassert INTC Message. 0b: Deassert 1b: Assert
1	INTB Status	0h	R	It is set by sending an Assert INTB Message and cleared by sending a Deassert INTB Message. 0b: Deassert 1b: Assert
0	INTA Status	0h	R	It is set by sending an Assert INTA Message and cleared by sending a Deassert INTA Message. 0b: Deassert 1b: Assert

(8) Message Receive Interrupt Enable Register (PCI_EP_MSGRCVIE)

This register controls enabling of MSG_INT in response to the reception of message requests other than INTx and error-related messages.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0120h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	Message Receive Interrupt Enable	-	-	-	-	PM_Active_State_Nak Receive Interrupt Enable	-	PME_Turn_Off Receive Interrupt Enable	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	Message Receive Interrupt Enable	0h	RW	Enable control of MSG_INT assertion by message reception 0b: Disable 1b: Enable
23 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	PM_Active_State_Nak Receive Interrupt Enable	0h	RW	Enable control of MSG_INT assertion by PM_Active_State_Nak reception 0b: Disable 1b: Enable
18	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
17	PME_Turn_Off Receive Interrupt Enable	0h	RW	Enable control of MSG_INT assertion by PME_Turn_Off reception 0b: Disable 1b: Enable
16	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
15 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(9) Message Receive Interrupt Status Register (PCI_EP_MSGRCVIS)

This register is a status register that indicates the reception of message requests other than INTx and error-related messages. The value of this register is reflected in MSG_INT. Only the message code is used to decide the message type, and the corresponding message is assumed to have been received while the validity of the routing and the validity of the Msg/MsgD selection are not verified.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0124h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	Message Receive Interrupt	-	-	-	-	PM_Active_State_Nak Receive Interrupt	-	PME_Turn_Off Receive Interrupt	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW1	R	R	R	R	RW1	RW1	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	Message Receive Interrupt	0h	RW1	Set when receiving a message (does not depend on the type of message)
23 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	PM_Active_State_Nak Receive Interrupt	0h	RW1	Disabled due to RC
18	-	0h	RW1	Reserved These bits are read as 0b. The write value should always be 0b.
17	PME_Turn_Off Receive Interrupt	0h	RW1	Disabled due to RC
16	-	0h	RW1	Reserved These bits are read as 0b. The write value should always be 0b.
15 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(10) Message Code Register (PCI_EP_MSGCODE)

This register stores the code and routing of the last received message.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0130h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Code[7:0]								Routing[2:0]			-	-	-	-	Message Payload
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 8	Message Code[7:0]	0h	R	Stores the Code of the last received Message.
7 to 5	Routing[2:0]	0h	R	Stores the Routing of the last received Message.
4 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	Message Payload	0h	R	Stores the presence or absence of the data payload of the last received Message. 1b: MsgD (with Payload) 0b: Msg (Without Payload) When Power Management Message (PME_TO_Ack Message, PME_Turn_Off Message, PM_PME Message, PM_Active_State_Nak Message) is received, this register is not written.

(11) Message Data Register (PCI_EP_MSGDATA)

This register stores the data of the last received message.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0134h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Message Data[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Data[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Message Data[31:0]	0h	R	Stores the 1DW data of the last received Message. It is updated only when MsgD(with Data) is received, and the previous value is retained when Msg(without Data) is received. When Power Management Message (PME_TO_Ack Message, PME_Turn_Off Message, PM_PME Message, PM_Active_State_Nak Message) is received, this register is not written.

(12) Message Header 3rdDW Register (PCI_EP_MSGH3DW)

This register stores the header (3rd DW) of the last received message.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0138h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Message Header 3rdDW[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Header 3rdDW[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Message Header 3rdDW[31:0]	0h	R	Stores the Header (3rdDW) of the last received Message.

Note: When a Power Management message (PME_TO_Ack Message, PME_Turn_Off Message, PM_PME Message, PM_Active_State_Nak Message) is received, this register is not written.

(13) Message Header 4thDW Register (PCI_EP_MSGH4DW)

This register stores the header (4th DW) of the last received message.

Access Size : 32 bits
Offset Address : <PCI0_base> + 013Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Message Header 4thDW Registers[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Header 4thDW Registers[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Message Header 4thDW Registers[31:0]	0h	R	Stores the Header (4thDW) of the last received Message.

Note: When a Power Management message (PME_TO_Ack Message, PME_Turn_Off Message, PM_PME Message, PM_Active_State_Nak Message) is received, this register is not written.

(14) Interrupt Table Register (PCI_EP_INTTABLE)

This register is an index of interrupt factors. The interrupt signal (active high) status of each category can be monitored in a list.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0140h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_INT[7:0]								-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	AXI_ERR_IN	PCIE_EVT_INT	MSG_INT	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DMA_INT[7:0]	0h	R	DMA_INT interrupt signal monitor
23 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	AXI_ERR_IN	0h	R	Error interrupt signal monitor
9	PCIE_EVT_INT	0h	R	Event interrupt signal monitor
8	MSG_INT	0h	R	Message interrupt signal monitor
7 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(15) PCIe Event Interrupt Enable 0 Register (PCI_EP_PEIE0)

This register enables interrupts of each PCI Express event factors. This register enables writing to the PCIe Event Interrupt Status 0 Register (address: <PCI_S0_REG_base> + 0204h).

Access Size : 32 bits
Offset Address : <PCI0_base> + 0200h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	UI_LINK_WIDTH_CHANGE_DONE EN	UI_LINK_SPEED_CHANGE_DONE EN	Request Done EN	-	-	AXI_PERR_EN	CA EN	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	R	R	RW	RW	RW	RW	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BME PERR EN	-	-	RX_DLLP_PM_ENTER_L23 EN	-	ASPM L1 Rejected EN	DL_Up Down EN	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30	UI_LINK_WIDTH_CHANGE_DONE EN	0h	RW	Up/Down Configure operation complete interrupt enable 0b: Disable 1b: Enable
29	UI_LINK_SPEED_CHANGE_DONE EN	0h	RW	Speed change operation completion interrupt enable 0b: Disable 1b: Enable
28	Request Done EN	0h	RW	Request complete interrupt enable 0b: Disable 1b: Enable
27, 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25	AXI_PERR_EN	0h	RW	Enable AXIM RAM parity error interrupt 0b: Disable 1b: Enable
24	CA EN	0h	RW	CA (Completer Abort) interrupt enable 0b: Disable 1b: Enable
23, 22	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
15	BME PERR EN	0h	RW	Enable BME Parity Error interrupt. 0b: Disable 1b: Enable (Note: Parity Error in AXI bridge internal RAM)
14, 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
12	RX_DLLP_PM_ENTER_L23 EN	0h	RW	RX_DLLP_PM_ENTER_L23 interrupt enable 0b: Disable 1b: Enable
11	-	0h	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
10	ASPM L1 Rejected EN	0h	RW	ASPM L1 Rejected interrupt enable Do not use this bit in Endpoint mode. 0b: Disable 1b: Enable
9	DL_UpDown EN	0h	RW	Interrupt enable on DL state change 0b: Disable 1b: Enable
8 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(16) PCIe Event Interrupt Status 0 Register (PCI_EP_PEIS0)

This register is a status register to indicate interrupts of each PCI Express event factors. Set to 1b by the factor in the table. After checking the factor, write 1b to the corresponding bit to clear it.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0204h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	UI_LINK_WIDTH_CHANGE_DONE	UI_LINK_SPEED_CHANGE_DONE	Request Done	-	-	AXIM PERR	CA	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW1	RW1	RW1	R	R	RW1	RW1	RW1	RW1	R	RW1	RW1	RW1	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BME_PERR	-	-	RX_DLLP_PM_ENTER_L23	-	ASPM L1 Rejected	DL_UpDown	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	R	R	R	RW1

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30	UI_LINK_WIDTH_CHANGE_DONE	0h	RW1	Indicates completion of Up/Down Configure operation.
29	UI_LINK_SPEED_CHANGE_DONE	0h	RW1	Indicates completion of Speed Change operation.
28	Request Done	0h	RW1	For requests submitted in the Request Issue Registers (Offset: 9Ch): Non-Posted: Indicates that Completion has been received. Posted: Indicates that the request submission has completed.
27, 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25	AXIM PERR	0h	RW1	Indicates that a parity error has occurred in the AXIM RAM. (Note: Parity Error in AXI bridge internal RAM)
24	CA	0h	RW1	Indicates that the device has responded with CA (Completer Abort).
23, 22	-	All 0	RW1	Reserved The write value should be 1b.
21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20 to 16	-	All 0	RW1	Reserved The write value should be 1b.
15	BME_PERR	0h	RW1	Indicates that a parity error has occurred in the BME RAM.
14, 13	-	All 0	RW1	Reserved These bits are read as 0b. The write value should always be 0b.
12	RX_DLLP_PM_ENTER_L23	0h	RW1	Indicates transition to L2/L3 State in Power Management control.
11	-	0h	RW1	Reserved These bits are read as 0b. The write value should always be 0b.
10	ASPM L1 Rejected	0h	RW1	It is recommended not to use (permit) in Endpoint mode.
9	DL_UpDown	0h	RW1	Set to 1b on transition from DL_Down state to DL_Up state, or DL_Up state to DL_Down state. Check the DL_Down/DL_Up status with PCIe Core Status 1 Registers (Offset: 408h).
8 to 4	-	All 0	RW1	Reserved The write value should be 1b.

Bit	Bit Name	Initial Value	R/W	Description
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	-	0h	RW1	Reserved These bits are read as 0b. The write value should always be 0b.

(17) PCIe Event Interrupt Enable 1 Register (PCI_EP_PEIE1)

This register enables parity error and ECC error interrupts. When each bit is set to the valid setting, the value of each corresponding status bit of the PCIe Event Interrupt Status 1 register (address: <PCI_S0_REG_base> + 020Ch) becomes valid.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0208h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TXB_PARITY_ERR EN	ERR_RPC_REPLAYFIFO_PERR EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	ERR_REPLAY_UPPER_CORRECTABLE_ERROR EN	ERR_REPLAY_LOWER_CORRECTABLE_ERROR EN	-	-	-	-	-	-	ERR_REPLAY_UPPER_UNCORRECTABLE_ERROR EN	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	TXB_PARITY_ERR EN	0h	RW	Enable the TXB_PARITY_ERR interrupt. Parity error interrupt notification enable setting for TX Buffer installed in Transaction Layer 0b: Disable 1b: Enable
16	ERR_RPC_REPLAYFIFO_PERR EN	0h	RW	Enable ERR_RPC_REPLAYFIFO_PERR interrupts. Parity error interrupt notification enable setting for Replay FIFO installed in Data Link Layer 0b: Disable 1b: Enable
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	ERR_REPLAY_UPPER_CORRECTABLE_ERROR EN	0h	RW	Enable ERR_REPLAY_UPPER_CORRECTABLE_ERROR interrupts. Interrupt notification enable setting when an ECC 1-bit error (Correctable Error) occurs in the upper 64-bit data bus of the Replay FIFO installed in the Data Link Layer 0b: Disable 1b: Enable
8	ERR_REPLAY_LOWER_CORRECTABLE_ERROR EN	0h	RW	Enable ERR_REPLAY_LOWER_CORRECTABLE_ERROR interrupts. Interrupt notification enable setting when an ECC 1-bit error (Correctable Error) occurs in the lower 64-bit data bus of the Replay FIFO installed in the Data Link Layer 0b: Disable 1b: Enable
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	ERR_REPLAY_UPPER_UNCORRECTABLE_ERROR EN	0h	RW	Enable ERR_REPLAY_UPPER_UNCORRECTABLE_ERROR interrupts. Interrupt notification enable setting when an ECC 2-bit or more error (Uncorrectable Error) occurs in the upper 64-bit data bus of the Replay FIFO installed in the Data Link Layer 0b: Disable 1b: Enable

Bit	Bit Name	Initial Value	R/W	Description
0	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR EN	0h	RW	Enable ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR interrupts. Interrupt notification enable setting when an ECC 2-bit or more error (Uncorrectable Error) occurs in the lower 64-bit data bus of the Replay FIFO installed in the Data Link Layer 0b: Disable 1b: Enable

(18) PCIe Event Interrupt Status 1 Register (PCI_EP_PEIS1)

This register is a status register that indicates parity error and ECC error interrupts. After checking the factor, write 1b to the corresponding bit to clear it.

Access Size : 32 bits
Offset Address : <PCI0_base> + 020Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TXB_PARITY_ERR	ERR_RPC_REPLAYFIFO_PERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	ERR_REPLAY_UPPER_CORRECTABLE_ERROR	ERR_REPLAY_LOWER_CORRECTABLE_ERROR	-	-	-	-	-	-	ERR_REPLAY_UPPER_UNCORRECTABLE_ERROR	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW1	RW1	R	R	R	R	R	R	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	TXB_PARITY_ERR	0h	RW1	TXB_PARITY_ERR interrupt Parity error interrupt notification of TX Buffer installed in Transaction Layer
16	ERR_RPC_REPLAYFIFO_PERR	0h	RW1	ERR_RPC_REPLAYFIFO_PERR interrupt Parity error interrupt notification of Replay FIFO installed in Data Link Layer
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	ERR_REPLAY_UPPER_CORRECTABLE_ERROR	0h	RW1	ERR_REPLAY_UPPER_CORRECTABLE_ERROR interrupt Interrupt notification when an ECC 1-bit error (Correctable Error) occurs in the upper 64-bit data bus of the Replay FIFO installed in the Data Link Layer
8	ERR_REPLAY_LOWER_CORRECTABLE_ERROR	0h	RW1	ERR_REPLAY_LOWER_CORRECTABLE_ERROR interrupt Interrupt notification when an ECC 1-bit error (Correctable Error) occurs in the lower 64-bit data bus of the Replay FIFO installed in the Data Link Layer
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	ERR_REPLAY_UPPER_UNCORRECTABLE_ERROR	0h	RW1	ERR_REPLAY_UPPER_UNCORRECTABLE_ERROR interrupt Interrupt notification when an ECC 2-bit or more error (Uncorrectable Error) occurs in the upper 64-bit data bus of the Replay FIFO installed in the Data Link Layer
0	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR	0h	RW1	ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR interrupt Interrupt notification when an ECC 2-bit or more error (Uncorrectable Error) occurs in the lower 64-bit data bus of the Replay FIFO installed in the Data Link Layer

(19) AXI Master Error Interrupt Enable Register (PCI_EP_AMEIE)

This register enables the AXI master error interrupt.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0210h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	Write MSTERR INT EN[3:0]				-	-	-	-	Read MSTERR INT EN[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	Write MSTERR INT EN[3:0]	0h	RW	Write MSTERR INT Enable Each corresponding bit can be turned on/off individually. 0b: Disable 1b: Enable
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	Read MSTERR INT EN[3:0]	0h	RW	Read MSTERR INT Enable Each corresponding bit can be turned on/off individually. 0b: Disable 1b: Enable

(20) AXI Master Error Interrupt Status Register (PCI_EP_AMEIS)

This register indicates the AXI master error interrupt status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0214h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	Write ERR ID[3:0]				-	-	-	-	Read ERR ID[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	Write MSTERR INT[3:0]				-	-	-	-	Read MSTERR INT[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW1	RW1	RW1	RW1	R	R	R	R	RW1	RW1	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27 to 24	Write ERR ID[3:0]	0h	R	Save the ID of the first DECERR/SLVERR received. When bits [11:8] are is cleared, a new error ID can be saved. 0h: normal access
23 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 16	Read ERR ID[3:0]	0h	R	Save the ID of the first DECERR/SLVERR received. A new error ID can be saved when bits [3:0] are cleared. 0h: normal access
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	Write MSTERR INT[3:0]	0h	RW1	Indicates that an error was detected in the AXI Master Port. Only the first detected error is saved, and when bits [11:8] are cleared, a new error can be saved. bit11: length error When the length of the data sent by the TEF and the data channel do not match. bit10: ID mismatch When the MBID value received on the MAWID response channel is different. bit9: When DECERR is received bit8: When SLVERR is received
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	Read MSTERR INT[3:0]	0h	RW1	Indicates that an error was detected in the AXI Master Port. Only the first detected error is saved, and when bits [3:0] are cleared, a new error can be saved. bit3: length error When the length of the data received on the TER and the data channel do not match. bit2: ID mismatch When the MARID and MRID values received on the data channel are different. bit1: When DECERR is received bit0: When SLVERR is received

(21) AXI Slave Error Interrupt Enable 1 Register (PCI_EP_ASEIE1)

This register enables the AXI slave error interrupt.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0220h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	Write SLVERR INT EN[3:0]				-	-	-	-	-	-	Read SLVERR INT EN[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	Write SLVERR INT EN[3:0]	0h	RW	Enable Write SLVERR INT. Each corresponding bit can be turned on/off individually. 0b: Disable 1b: Enable
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	Read SLVERR INT EN[1:0]	0h	RW	Enable Read SLVERR INT. Each corresponding bit can be turned on/off individually. 0b: Disable 1b: Enable

(22) AXI Slave Error Interrupt Status 1 Register (PCI_EP_ASEIS1)

This register indicates the AXI slave error interrupt status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0224h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	Write SLVERR INT[3:0]				-	-	-	-	-	-	Read SLVERR INT[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW1	RW1	RW1	RW1	R	R	R	R	R	R	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	Write SLVERR INT[3:0]	0h	RW1	Indicates that an unrecoverable error was detected in the AXI Slave Port. (Transaction response will be SLVERR.) bit11: burst length error When the SAWLEN and the burst length of the data received on the data channel do not match. bit10: ID mismatch When the SAWID and SWID values received on the data channel are different. bit9: Burst type disabled When SAWBURST is b11 (undefined). When SAWBURST is b10 (wrapping) and burst length is other than 2, 4, 8, or 16. bit8: data size invalid When SAWSIZE is between b100 and b111 (Exceeding the AXI Bus width is not supported). Each bit means: 0b: No error detected 1b: error detection
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	Read SLVERR INT[1:0]	0h	RW1	Indicates that an unrecoverable error was detected in the AXI Slave Port. (Transaction response will be SLVERR.) bit1b: Burst type disabled When SARBURST is b11 (undefined). When SARBURST is b10 (wrapping) and burst length is other than 2, 4, 8, or 16. bit0b: Data size invalid When SARSIZE is between b100 and b111 (exceeding the AXI Bus width is not supported). Each bit means: 0b: No error detected 1b: error detection

(23) AXI Slave Error Interrupt Status 3 Register (PCI_EP_ASEIS3)

This register indicates the AXI slave error interrupt status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0230h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ERR ID[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERR ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ERR ID[31:0]	0h	R	Save the ID when the first error occurred in the AXI Slave Error Interrupt Status1 register (Offset 224h). Only the first detected error is saved, and when bits [11:8] and bits [1:0] of the AXI Slave Error Interrupt Status1 register are all cleared, a new error ID can be saved.

(24) PCIe Event Interrupt Enable 2 Register (PCI_EP_PEIE2)

This register enables interrupts of the various PCI Express event factors. This register enables writing to the PCIe Event Interrupt Status 2 Register (address: <PCI_S0_REG_base> + 0244h).

Access Size : 32 bits
Offset Address : <PCI0_base> + 0240h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LTR_MECHANISM_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D3_hot_err_EN[7:0]								CFG_POWER_STATE_EN[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	LTR_MECHANISM_EN	0h	RW	Interrupt enable for the ability to detect changes in CFG_LTR_MECHANISM
15 to 8	D3_hot_err_EN[7:0]	0h	RW	When Completion is in Pending state (ORT_TRANSACTION_PENDING=1), transition to D3hot state (CFG_POWERSTATE!=0) or receive PME_TURN_OFF Message, detect the state when D3_EVENT_ACK/TURN_OFF_EVENT_ACK is asserted Interrupt enable for D3_hot_err status detection Bits [15:10] are not used. The lower 2 bits correspond to each function.
7 to 0	CFG_POWER_STATE_EN[7:0]	0h	RW	D_STATE_OUT_Fx(x=Function no.) Interrupt enable bits [7:2] of the function to detect pin change are not used. The lower 2 bits correspond to each function.

(25) PCIe Event Interrupt Status 2 Register (PCI_EP_PEIS2)

This register indicates the state of various PCI Express events.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0244h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LTR_MECHANISM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D3_hot_err[7:0]								CFG_POWER_STATE[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	LTR_MECHANISM	0h	RW1	Status to detect and notify changes in CFG_LTR_MECHANISM
15 to 8	D3_hot_err[7:0]	0h	RW1	Status that detects and notifies the state when D3_EVENT_ACK/TURN_OFF_EVENT_ACK is asserted when transition to D3hot state (CFG_POWERSTATE!=0) or PME_TURN_OFF Message is received in Completion Pending state (ORT_TRANSACTION_PENDING=1) Bit[15:10] are not used. Lower 2bit corresponds to each function.
7 to 0	CFG_POWER_STATE[7:0]	0h	RW1	D_STATE_OUT_Fx(x=Function no.) Status to notify pin change detection Bits [7:2] are not used. The lower 2 bits correspond to each function.

(26) Permission Register (PCI_EP_PERM)

Access Size : 32 bits
Offset Address : <PCI0_base> + 0300h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PIPE PHY Register Enable	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
1	PIPE PHY Register Enable	0h	RW	Access permission signal for Physical Layer Control/Status Registers. 0b: Disable register access in PIPE PHY space. 1b: Enable register access in PIPE PHY space.
0	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.

(27) Reset Register (PCI_EP_RESET)

This register resets the PCIe core. Supplied to the internal core by OR with the pin of the same name. For details of each pin, see the terminal information. The write value is saved when accessed from the AXI, but a low-level pulse is generated when the PCIe writes 0b. However, when the AXI has already written 0b, the signal remains at the low level. Writing 1b from the PCIe is ignored.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0310h
Initial Value : 0000_00xxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	force to D0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RST_PREG_B	RST_OUT_B	RST_PS_B	RST_LOAD_B	RST_CFG_B	RST_RSM_B	RST_GP_B	RST_B
Initial Value	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	force to D0	0h	RW	After sending PME_TO_Ack, force PM Control to transition to D0 State. Auto-cleared when PM Control transitions to D0. This bit is normally prohibited as it can create inconsistencies with the power state of the entire system. 0b: No operation 1b: Transition to D0
15 to 12	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
7	RST_PREG_B	xh	RW	Not used in this macro. The setting value does not affect macro behavior. 0b: reset 1b: normal operation
6	RST_OUT_B	xh	RW	RST_OUT_B output 0b: reset 1b: normal operation
5	RST_PS_B	xh	RW	Reset to the PCI Express core part (PCLK domain) inside the macro 0b: reset 1b: normal operation
4	RST_LOAD_B	xh	RW	Reset to Configuration Register Reset to bits not initialized by RST_CFG_B. 0b: reset 1b: normal operation
3	RST_CFG_B	xh	RW	Reset to Configuration Register 0b: Reset 1b: normal operation
2	RST_RSM_B	xh	RW	POWERGOOD reset of AUX Power (AUX not supported) Reset to Sticky register. 0b: reset 1b: normal operation
1	RST_GP_B	xh	RW	Reset to the PCI Express core part (ACLK domain) inside the macro 0b: reset 1b: normal operation
0	RST_B	xh	RW	Reset to PCI Express core part inside macro 0b: reset 1b: normal operation

Note: x = Undefined

(28) Mode Set 0 Register (PCI_EP_MSET0)

This register sets AXI mode.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0314h
Initial Value : 2001_2000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	AWPROT[2:0]			AWCACHE_L[3:0]				-	-	AWLOCK[1:0]		AWCACHE_D[3:0]			
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	ARPROT[2:0]			-	-	-	-	-	-	ARLOCK[1:0]		ARCACHE[3:0]			
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 28	AWPROT[2:0]	2h	RW	Sets the protection type for PCIe→AXI transactions. This bit indicates whether the transaction's protection level is Normal, Privileged, or Secure, and whether the transaction is a data or instruction access. Bit[2] 1b: instruction access 0b: data access Bit[1] 1b: non-secure access 0b: secure access Bit[0] 1b: privileged access 0b: normal access
27 to 24	AWCACHE_L [3:0]	0h	RW	Indicates the value of MAWCACHE[3:0] to be issued to AXI. This setting is prohibited when issuing an AXI request containing the last byte. Recommended value is 0000b. Bit[3] 1b: write allocatable 0b: not write allocatable Bit[2] 1b: read assignable 0b: not read assignable Bit[1] 1b: cacheable 0b: non-cacheable Bit[0] 1b: bufferable 0b: not bufferable Note: When issuing an MSI, the setting of bit 0 is ignored and buffering is forcibly disabled (MAWCACHE[0] = 0).
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21, 20	AWLOCK[1:0]	0h	RW	Lock type for PCIe-to-AXI transactions. This signal provides information about the atomic nature of the transfer. 00b: normal access, 01b: exclusive access, 10b: locked access, 11b: reserved
19 to 16	AWCACHE_D [3:0]	1h	RW	Indicates the value of MAWCACHE[3:0] to be issued to AXI. This setting is output when issuing an AXI request other than the output condition of AWCACHE_L. Recommended value is 0001b. Bit[3] 1b: write allocatable 0b: not write allocatable Bit[2] 1b: read assignable 0b: not read assignable Bit[1] 1b: cacheable 0b: non-cacheable Bit[0] 1b: bufferable 0b: not bufferable
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 12	ARPROT[2:0]	2h	RW	Sets the protection type for PCIe→AXI transactions. This bit indicates whether the transaction's protection level is Normal, Privileged, or Secure, and whether the transaction is a data or instruction access. Bit[2] 1b: instruction access 0b: data access Bit[1] 1b: non-secure access 0b: secure access Bit[0] 1b: privileged access 0b: normal access
11 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5, 4	ARLOCK[1:0]	0h	RW	Lock type for PCIe-to-AXI transactions. This bit provides information about the atomic nature of the transfer. 00b: normal access, 01b: exclusive access, 10b: locked access, 11b: reserved

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	ARCACHE[3:0]	0h	RW	Cache type for PCIe-to-AXI transactions. These bits specify “bufferable”, “cacheable”, “write-through”, “write-back”, or “allocation” as the attribute of the transaction.

(29) Mode Set 1 Register (PCI_EP_MSET1)

This register sets AXI mode.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0318h
Initial Value : 0000_33F2h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AXI Max Issue Write[3:0]				AXI Max Issue Read[3:0]				AXI Master Max Burst[3:0]				-	-	RAM Parity Enable	PCIe Request Order
Initial Value	0	0	1	1	0	0	1	1	1	1	1	1	0	0	1	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 12	AXI Max Issue Write[3:0]	3h	RW	Set the number of writes that can be issued by AXI Master. Set within the range allowed by Interconnect. 0h: possible number 1 1h: possible number 2 : Fh: Possible number 16
11 to 8	AXI Max Issue Read[3:0]	3h	RW	Set the number of read issues that can be issued by AXI Master. Set within the range allowed by Interconnect. 0h: possible number 1 1h: possible number 2 : Fh: Possible number 16
7 to 4	AXI Master Max Burst[3:0]	Fh	RW	Sets the maximum burst length as an AXI Master operation.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	RAM Parity Enable	1h	RW	Sets whether or not to check the parity of the internal SRAM. The default value is Enable, but it is ignored by non-Parity macros. 0b: RAM parity check disabled 1b: RAM parity check enabled
0	PCIe Request Order	0h	RW	Issue a Read Request to PCIe from the same AXI master without waiting for Completion. Set to 1 if you want to strictly follow the order of Requests to the Completer. 0b: Do not wait for Completion. 1b: Wait for Completion.

(30) Mode Set 3 Register (PCI_EP_MSET3)

This register outputs the setting value as the ASPM L1 Idle Time bit.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0380h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ASPM L1 Idle Time[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
7 to 0	ASPM L1 Idle Time[7:0]	0h	RW	Sets the idle period for AXI transactions that the macro checks on ASPM L1 transitions. One of the conditions for ASPM L1 transition is that the idle period is confirmed for the number of cycles of the 8 bits set by this bit plus 8'hFF to the lower 8 bits. 00h: 256 [ACLK] 01h: 512 [ACLK] : FFh : 65536 [ACLK]

(31) Mode Set 4 Register (PCI_EP_MSET4)

Access Size : 32 bits
Offset Address : <PCI0_base> + 0384h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MODE_EQ_RX_EVAL_WAIT[19:4]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MODE_EQ_RX_EVAL_WAIT[3:0]				-	-	-	-	MODE_8GT_EQTS2	MODE_EQ_PHASE23_ENABLE1	MODE_EQ_AUTONOMOUS1	-	-	UI_EQ_REQTYPE[1:0]		UI_EQ_REQUEST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	MODE_EQ_RX_EVAL_WAIT[19:0]	0h	RW	During RxEval (Downstream Port Phase3, Upstream Port Phase2), wait time setting until RxEval is executed when Block Alignment cannot be obtained. Set by the count number of PCLK. Note: It is necessary to set the register value (recommended: equivalent to 1msec) before starting. Please set from the local CPU.
11 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
7	MODE_8GT_EQTS2	0h	RW	8GT EQ TS2OS transmission enable/disable setting for Upstream Port 0b: Do not send, 1b: Send Note: Not used for Downstream Port. (fixed to 0b)
6	MODE_EQ_PHASE23_ENABLE1	0h	RW	Fixed to 0b
5	MODE_EQ_AUTONOMOUS1	0h	RW	Fixed to 0b
4, 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
2, 1	UI_EQ_REQTYPE[1:0]	0h	RW	Operation specification when sending Equalization Request with Quiesce Guarantee=1 in Recovery.RcvrCfg state 00b: Equalization Request transmission with Equalization Request Data Rate=0 (8G) 01b: Send Equalization Request with Equalization Request Data Rate=1 (16G) (Setting prohibited) 10b: Operation of “0” + set Link Equalization Request 8.0GT/s register 11b: Set “1” operation + Link Equalization Request 16.0GT/s register (setting prohibited)
0	UI_EQ_REQUEST	0h	RW	Set when sending an Equalization Request with Quiesce Guarantee=1 in the Recovery.RcvrCfg state. Setting 1b is prohibited except when MODE_QUIESCE_GUARANTEE=1. After setting 1b, hold until confirmation of UI_EQ_DONE=1. 0b: Do not send Equalization Request 1b: Send Equalization Request

(32) Mode Set 5 Register (PCI_EP_MSET5)

Access Size : 32 bits
Offset Address : <PCI0_base> + 0388h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MODE_8GT_EQTS2_PRESET[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
15 to 0	MODE_8GT_EQ TS2_PRESET [15:0]	0h	RW	Transmitter Preset value of 8GT EQ TS2OS to transmit at Upstream Port (USP). No setting is required if 8GT_EQTS2 is not sent from USP (DSP Preset is not specified). Settings must be made in advance if necessary. 4-bit configuration for each lane. Gen3/Gen4 supports up to x4 lane configurations. The correspondence between each bit and lane number is as follows. Bit[3:0] Lane #0 Bit[7:4] Lane #1 Bit[11:8] Setting prohibited, other than default value is not writable. Bit[15:12] Setting prohibited, other than default value is not writable. The above lane number is the default lane number for circuit implementation. Note that this is not a negotiated lane number. Note: Not used for Downstream Port. (fixed to 0000_0000h)

(33) Mode Status 0 Registers (PCI_EP_MSTA0)

Access Size : 32 bits
Offset Address : <PCI0_base> + 0390h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	RX_EQ_REQTYPE[1:0]		RX_EQ_REQUEST	UI_EQ_DONE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	RX_EQ_REQTYPE[1:0]	0h	R	Quiesce Guarantee and Equalization Request Data Rate bit display when receiving 8 consecutive TS2OS with Request Equalization bit (Symbol6, bit7) = 1b in Recovery. RcvrCfg state EP (USP) (1) When Request Equalization is received bit0: Quiesce Guarantee (TS2OS, Symbol6, bit6) (= 1 fixed) bit1: Equalization Request Data Rate (TS2OS, Symbol6, bit5) (2) When receiving EQTS2OS bit0: 0 (fixed) bit1: Equalization Request Data Rate (0b: EQTS2, 1b: 8GT EQTS2)
1	RX_EQ_REQUEST	0h	R	Status display when receiving 8 consecutive TS2OS with Request Equalization bit (Symbol6, bit7) = 1 in Recovery.RcvrCfg state 0b: No Equalization Request received 1b: Equalization Request received
0	UI_EQ_DONE	0h	R	Set Equalization Request with Quiesce Guarantee=1 in Recovery.RcvrCfg state when sending 0b: No Equalization Request sent 1b: Sent Equalization Request Related flow: "Set UI_EQ_REQUEST = 1b → Confirm UI_EQ_DONE = 1b → Clear UI_EQ_REQUEST = 0b"

(34) PCIe Core Mode Set 1 Register (PCI_EP_PCMSET1)

This register sets the operating mode of the PCI Express core.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0400h
Initial Value : 00FA_00F2h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	ASPM_L1_INTERVAL_TIME[11:0]											
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	MODE_LINK_UPCONFIGURE_CAPABILITY_DISABLED	-	MODE_TXSWING	-	MODE_SELECTABLE_DEEMPHASIS	-	-	-	-	-	-	MODE_PORT	-
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	0	0	1	0
R/W	R	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29, 28	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
27 to 16	ASPM_L1_INTERVAL_TIME [11:0]	0FAh	RW	Interval settings for ASPM L1 requests The PCIe Base Spec stipulates that ASPM L1 transition requests must not be accepted continuously within 10 us, and this field sets the timer value to guard against this. Set so that ACLK cycle x set value is 10 us or more. At ACLK: 400[MHz] or higher, this bit setting value should be the value (1/16) of the lower 4 bits of the setting value of the above specifications, and set the clock number as the timer value setting. Settings of this product (Default): When ACLK=400MHz(2.5ns): 4000(d)/16 = FA(h)
15	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14, 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
12	MODE_LINK_UPCONFIGURE_CAPABILITY_DISABLED	0h	RW	Setting of Link Upconfigure Capability bit of Training Sequence Ordered-set (TS-OS) 0b: Link Upconfigure Capability bit = 1b setting 1b: Link Upconfigure Capability bit = 0b setting (Gen1 x1) When connecting with a Gen1 PCIe device, Linkup may not occur unless this bit is 0b. In that case, change it to 0b in F/W.
11	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	MODE_TXSWING	0h	RW	SerDes serial output amplitude control 0b: Full swing mode (default) 1b: Half swing mode
9	-	0h	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
8	MODE_SELECTABLE_DEEMPHASIS	0h	RW	De-emphasis setting in Endpoint mode For Endpoint, set the De-emphasis value for 5.0 GT/s operation. Initial value of select_deemphasis variable described in PCIe Base Spec. 0b: -6 dB (default) 1b: -3.5dB

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	-	All 1	RW	Reserved Whenever it is read, 1111b is read. The write value should always be 1111b.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	MODE_PORT	1h	RW	Device type setting register. When the bit value of this register is fixed to the initial value 1b, the setting of the external pin MODE_PORT becomes valid. When the external pin MODE_PORT is fixed to 1b, the setting by this register bit becomes valid. 0b: Endpoint 1b: Root Complex
0	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(35) PCIe Core Control 1 Register (PCI_EP_PCCTRL1)

This register controls power management and LTSSM (Link Training Sequence State Machine) state transitions.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0404h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	BLB_RELAX_ORDERING_EN	-	-	-	-	-	UI_ENTER_L1S	-	-	-	-	-	UI_ENTER_TXL0S
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	MODE_QUIESCE_GUARANTEE	-	MODE_EQ_AUTONOMOUS	MODE_EQ_PHASE23_ENABLE	MODE_RESET_EIEOS_INTERVALL0S	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	BLB_RELAX_ORDERING_EN	0h	RW	Control of RO bit of Request to be sent 0b: RO bit of Request TLP to be sent is always 0b (default) 1b: A TLP can be sent with the RO bit of the Request TLP to be sent set to 1b.
27 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
23	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	UI_ENTER_L1S	0h	RW	L1SubState transition permission setting 0b : L1 Substate transition disabled (default) 1b: L1 Substate transition permission (setting prohibited)
21 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
16	UI_ENTER_TXL0S	0h	RW	TxL0s transition control 0b: Do not perform ASPM L0s transition (default) 1b: Execute ASPM L0s transition when internal conditions are satisfied
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12	MODE_QUIESCE_GUARANTEE	0h	RW	Symbol6 bit6 Quiesce Guarantee control bit of TS2OS 0b: Set 0b to TS2OS (default) 1b: Set 1b to TS2OS
11	-	0h	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
10	MODE_EQ_AUTONOMOUS	0h	RW	Gen3 feature: Autonomous Equalization Basically only changeable during the reset period 0b: Do not use Autonomous Mechanism 1b: Use Autonomous Mechanisms
9	MODE_EQ_PHASE23_ENABLE	0h	RW	Gen3 features: Setting whether to execute EQ PHASE2 and EQ PHASE3 in RC mode (MODE_PORT=1). 0b: Do not execute EQ PHASE2/3 1b: Execute EQ PHASE2/3

Bit	Bit Name	Initial Value	R/W	Description
8	MODE_RESET_EIEOS_INTERVALL0S	0h	RW	Gen3 features: Reset EIEOS Interval information of bit 2, symbol 6 of TS1OS transmitted in Recovery.Equalization state
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.

(36) PCIe Core Status 1 Register (PCI_EP_PCSTAT1)

This register indicates the status of the power management in the PCI Express core.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0408h
Initial Value : 0x0x_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	TURN_OFF_EVENT_ACK	TURN_OFF_EVENT	-	-	-	-	-	-	-	-	bme_down	-
Initial Value	0	0	0	0	x	0	0	0	0	0	0	0	0	0	x	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	LTSSM_STATE[6:0]							PMU_LINKSTATE[3:0]				-	-	STATE_VC0_NEGOTIATION_PENDING	DL_Down status
Initial Value	0	x	x	x	x	x	x	x	x	x	x	x	0	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27	TURN_OFF_EVENT_ACK	xh	R	TURN_OFF_EVENT_ACK input signal monitor
26	TURN_OFF_EVENT	0h	R	TURN_OFF_EVENT output signal monitor
25 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	bme_down	xh	R	Indicates that the PCIe core transmitter is in an unusable state.
16, 15	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 8	LTSSM_STATE[6:0]	xh	R	Indicates the state of the Link Training & Status State Machine in the PCIe Core Link. The following states are indicated by the upper 5 bits [14:10]. 000xxb: Detect 001xxb: Polling 010xxb: Config 01100b: L0 01101b: L1 0111xb: L2 100xxb: Recovery 101xxb: Disable 110xxb: Loopback
7 to 4	PMU_LINKSTATE[3:0]	xh	R	L-state monitor of power management control unit 0100b: L1 state 1000b: L2 state
3	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	-	xh	R	Reserved Whenever it is read, x is read. The written value will be ignored.
1	STATE_VC0_NEGOTIATION_PENDING	xh	R	Flow Control initialization operation monitor If this bit is 1b, do not initiate a transaction from the AXI side. Check that this bit is 0b and DL_Down Status (bit [0]) is 0b. 0b: Indicates that Flow Control initialization is complete 1b: Indicates that Flow Control initialization has not completed

Bit	Bit Name	Initial Value	R/W	Description
0	DL_Down status	xh	R	Indicates whether PCIe Core is in DL_Down or DL_Up state 0b: DL_Up Status 1b: DL_Down Status

Note: x = Undefined

(37) PCIe Core Control 2 Register (PCI_EP_PCCTRL2)

This register controls the link speed/width change in the PCI Express core.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0410h
Initial Value : 003E_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	UI_LINK_WIDTH_CHANGE_ENABLE[7:0]								MODE_NODEEMPHASIS[1:0]		MODE_PRESET_ENABLE[4:0]					UI_LINK_WIDTH_CHANGE_REQ
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	UI_LINK_SPEED_CHANGE[1:0]		-	-	-	UI_LINK_CHANGE_AUTONOMOUS	-	-	-	UI_LINK_SPEED_CHANGE_REQ
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	UI_LINK_WIDTH_CHANGE_ENABLE[7:0]	0h	RW	Link Width setting to change Assert UI_LINK_WIDTH_CHANGE_REQ and set Lane to 1b to operate when Link Width change request is issued. The lower bit (bit [24]) drives Lane0 and the most significant bit (bit [31]) drives Lane7. For this LSI effective bits are Bit-25 and Bit-24
23, 22	MODE_NODEEMPHASIS[1:0]	0h	RW	No de-emphasis mode setting pin for Gen1/Gen3 operation [0] Gen1 operation, 0b: Normal operation mode (default), 1b: No de-emphasis mode [1] Gen3 operation, 0b: Normal operation mode (default), 1b: No de-emphasis mode
21 to 17	MODE_PRESET_ENABLE[4:0]	1Fh	RW	Reduced Swing mode setting pin during Gen3 operation (unused) Bit[0]: 8GT/s Preset P0, 0b: disable, 1b: enable (default) Bit[1]: 8GT/s Preset P2, 0b: disable, 1b: enable (default) Bit[2]: 8GT/s Preset P7, 0b: disable, 1b: enable (default) Bit[3]: 8GT/s Preset P8, 0b: disable, 1b: enable (default) Bit[4]: 8GT/s Preset P10, 0b: disable, 1b: enable (default)
16	UI_LINK_WIDTH_CHANGE_REQ	0h	RW	Link Width change request control Setting this bit to 1b issues a request to change the Link Width to the configuration set in bits [31:24] UI_LINK_WIDTHCHANGE_ENABLE field. By asserting it in the L0 state, it transitions from the Recovery state to the Configuration state, and performs negotiation with the other device. Set to 0b after confirming that PCIe Core Status 2 Register (Offset: 414h) bit [29] UI_LINK_WIDTH_CHANGE_DONE is asserted.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9, 8	UI_LINK_SPEED_CHANGE[1:0]	0h	RW	Link speed setting Set the Link Speed you want to change. 00b: 2.5 GT/s 01b: 5.0 GT/s 10b: 8.0 GT/s 11b: 16.0 GT/s (Setting prohibited)
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	UI_LINK_CHANGE_AUTONOMOUS	0h	RW	Link Width/Speed change reason setting 0b: reliability reason (change for reliability, direction of bandwidth reduction) 1b: autonomous reason (intentional change)
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
0	UI_LINK_SPEED_CHANGE_REQ	0h	RW	Link Speed change request control Setting this bit to 1b requests to change the Link Speed to the speed set in bit [8] UI_LINK_SPEED_CHANGE field. By asserting it in the L0 state, it transitions to the Recovery state and performs negotiation with the peer device. PCIe Core Status 2 Register (Offset: 414h) bit[28] Set to 0b after confirming that UI_LINK_SPEED_CHANGE_DONE has been asserted.

(38) PCIe Core Status 2 Register (PCI_EP_PCSTAT2)

This register indicates the status of the link speed/width change in the PCI Express core.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0414h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	UI_LINK_WIDTH_CHANGE_DONE	UI_LINK_SPEED_CHANGE_DONE	-	-	-	STATE_UPCONFIGURE_CAPABLE	-	STATE_NEGOTIATED_LANE_END[2:0]			-	STATE_NEGOTIATED_LANE_START[2:0]		
Initial Value	0	0	x	x	0	0	0	x	0	x	x	x	0	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STATE_RECEIVER_DETECTED[7:0]								STATE_DATA_RATE_IDENTIFIER_RECEIVED[7:0]							
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29	UI_LINK_WIDTH_CHANGE_DONE	xh	R	Link Width Change operation complete display Notifies completion of Width Change (1b) by setting PCIe Core Status 2 Register (Offset: 410h) bit [16] UI_LINK_WIDTH_CHANGE_REQ. It is set to 0b by setting UI_LINK_WIDTH_CHANGE_REQ to 0b.
28	UI_LINK_SPEED_CHANGE_DONE	xh	R	Link Speed Change operation complete display PCIe Core Status 2 Registers (Offset: 410h) bit[0] Notifies completion of Speed Change (1b) by setting UI_LINK_SPEED_CHANGE_REQ. It is set to 0b by setting UI_LINK_SPEED_CHANGE_REQ to 0b.
27 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	STATE_UPCONFIGURE_CAPABLE	xh	R	Upconfigure Capable bit display of opposite device Indicates whether the peer device supports changing the direction of widening the Link Width. If this bit is 0b, changing the Link Width will not restore the original Link Width.
23	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22 to 20	STATE_NEGOTIATED_LANE_END[2:0]	xh	R	Displays the position of Lane Number (n-1) (meaning Lane1 when n=2) after Link Negotiation with the opposite device during n-lane operation. Used to check the state of the current working lane before changing the Link Width. 000b: Lane0 is Lane Number (n-1) 001b: Lane1 is Lane Number (n-1) Others: reserved
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	STATE_NEGOTIATED_LANE_START[2:0]	xh	R	Displays the position of Lane Number 0 after Link Negotiation with the opposite device during n-lane operation. Used to check the state of the current working lane before changing the Link Width. 000b: Lane0 is Lane Number 0 001b: Lane1 is Lane Number 0 Others: reserved
15 to 8	STATE_RECEIVER_DETECTED[7:0]	xh	R	Connection status display with other device Receiver Detection results are displayed. Bit[0] Detect opposite device on Lane0 Bit[1] Detect peer device on Lane1 (only for x2) Bit[2:7] reserved

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	STATE_DATA_RATE_IDENTIFIER_RECEIVED [7:0]	xh	R	Link Speed display supported by the opposite device Displays the TS-OS Data Rate Identifier feed received from the peer device. Bit[0] – Reserved Bit[1] – 2.5 GT/s Data Rate Supported. Must be set to 1b. Bit[2] – 5.0 GT/s Data Rate Supported. Must be set to 1b if Bit3 is 1b. Bit[3] – 8.0 GT/s Data Rate Supported. Bits[4:7] – Reserved

Note: x = Undefined

(39) PCIe Core Status 5 Register (PCI_EP_PCSTAT5)

This register indicates the status in the PCI Express core.

Access Size : 32 bits
Offset Address : <PCI0_base> + 042Ch
Initial Value : 0000_0x00h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Suspend_bme[7:0]								D3_EVENT[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D3_EVENT_ACK[7:0]								ORT_TRANSACTION_PENDING[7:0]							
Initial Value	0	0	0	0	0	0	x	x	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Suspend_bme [7:0]	0h	R	Indicates the state in which the use of the PCIe core transmitter should be suppressed for each function. Other than the lower 2 bits Reserved (fixed to 0b)
23 to 16	D3_EVENT [7:0]	0h	R	D3_EVENT signal monitor for each function Other than the lower 2 bits Reserved (fixed to 0b)
15 to 8	D3_EVENT_ACK[7:0]	0xh	R	D3_EVENT_ACK signal monitor for each function Other than the lower 2 bits Reserved (fixed to 0b)
7 to 0	ORT_TRANSACTION_PENDING [7:0]	0h	R	Outstanding Request Monitor by Function Other than the lower 2 bits Reserved (fixed to 0b) Indicates whether or not there are Outstanding Requests (a state in which all Completions corresponding to Non-Posted Requests sent from the AXI side have not been received). Check with this bit that there is no Outstanding Request before requesting/permitting a transition to TxL0s/L1/L2. 0b: State without Outstanding Request 1b: State with Outstanding Request

(40) DMA Interrupt Vector 0 Register (PCI_EP_DMAINTVEC0)

This register specifies the interrupt vectors for interrupt notification from AXI to PCIe (MSI) during DMA transfer.

Access Size : 32 bits
Offset Address : <PCI0_base> + 04D0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	DMA_CH3_MSI_EN	DMA_CH3_vec[4:0]					-	-	DMA_CH2_MSI_EN	DMA_CH2_vec[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	DMA_CH1_MSI_EN	DMA_CH1_vec[4:0]					-	-	DMA_CH0_MSI_EN	DMA_CH0_vec[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29	DMA_CH3_MSI_EN	0h	RW	DMA Ch3 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI
28 to 24	DMA_CH3_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch3
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21	DMA_CH2_MSI_EN	0h	RW	DMA Ch2 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI
20 to 16	DMA_CH2_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch2
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	DMA_CH1_MSI_EN	0h	RW	DMA Ch1 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI
12 to 8	DMA_CH1_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch1
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	DMA_CH0_MSI_EN	0h	RW	DMA Ch0 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI
4 to 0	DMA_CH0_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch0

(41) DMA Interrupt Vector 1 Register (PCI_EP_DMAINTVEC1)

This register specifies interrupt vectors for interrupt notification from AXI to PCIe (MSI) during DMA transfer.

Access Size : 32 bits
Offset Address : <PCI0_base> + 04D4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	DMA_CH7_MSI_EN	DMA_CH7_vec[4:0]					-	-	DMA_CH6_MSI_EN	DMA_CH6_vec[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	DMA_CH5_MSI_EN	DMA_CH5_vec[4:0]					-	-	DMA_CH4_MSI_EN	DMA_CH4_vec[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29	DMA_CH7_MSI_EN	0h	RW	DMA Ch7 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI
28 to 24	DMA_CH7_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch7
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21	DMA_CH6_MSI_EN	0h	RW	DMA Ch6 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI
20 to 16	DMA_CH6_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch6
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	DMA_CH5_MSI_EN	0h	RW	DMA Ch5 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI
12 to 8	DMA_CH5_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch5
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	DMA_CH4_MSI_EN	0h	RW	DMA Ch4 MSI Enable 0b: Do not use MSI (DMA interrupt notification by DMA_INT pin) 1b: Use MSI
4 to 0	DMA_CH4_vec[4:0]	0h	RW	Vector value of MSI interrupt transmitted by DMAC Ch4

(42) DMAC Control Register (PCI_EP_DMACTRL)

This register sets the maximum size of read requests which can be issued to the PCIe Core as a DMAC function. Use the initial setting (128-byte). This setting is common to all DMA channels.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0800h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	D_PMRS[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	D_PMRS[2:0]	0h	RW	DMAC PCIe Max Read Request Size Set the upper limit of Read requests issued from the DMAC to PCIe. 000b: 128 bytes (default) 001b: 256 bytes 010b: 512 bytes (not supported) 011b: 1024 bytes (not supported) 100b: 2048 bytes (not supported) 101b: 4096 bytes (not supported) others: Reserved (prohibited)

(43) DMAC Interrupt Enable Register (PCI_EP_DMAINTE)

This register enables interrupts from the individual DMA channels.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0808h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CH7_ERR_EN	CH7_QUE_EMP_EN	CH7_STOP_EN	CH7_END_EN	CH6_ERR_EN	CH6_QUE_EMP_EN	CH6_STOP_EN	CH6_END_EN	CH5_ERR_EN	CH5_QUE_EMP_EN	CH5_STOP_EN	CH5_END_EN	CH4_ERR_EN	CH4_QUE_EMP_EN	CH4_STOP_EN	CH4_END_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH3_ERR_EN	CH3_QUE_EMP_EN	CH3_STOP_EN	CH3_END_EN	CH2_ERR_EN	CH2_QUE_EMP_EN	CH2_STOP_EN	CH2_END_EN	CH1_ERR_EN	CH1_QUE_EMP_EN	CH1_STOP_EN	CH1_END_EN	CH0_ERR_EN	CH0_QUE_EMP_EN	CH0_STOP_EN	CH0_END_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CH7_ERR_EN	0h	RW	CH7 Error Interrupt Enable 1b: Enable 0b: Disable
30	CH7_QUE_EMP_EN	0h	RW	CH7 Que Empty Interrupt Enable 1b: Enable 0b: Disable
29	CH7_STOP_EN	0h	RW	CH7 Stop Interrupt Enable 1b: Enable 0b: Disable
28	CH7_END_EN	0h	RW	CH7 Completion Interrupt Enable 1b: Enable 0b: Disable
27	CH6_ERR_EN	0h	RW	CH6 Error Interrupt Enable 1b: Enable 0b: Disable
26	CH6_QUE_EMP_EN	0h	RW	CH6 Que Empty Interrupt Enable 1b: Enable 0b: Disable
25	CH6_STOP_EN	0h	RW	CH6 Stop Interrupt Enable 1b: Enable 0b: Disable
24	CH6_END_EN	0h	RW	CH6 Completion Interrupt Enable 1b: Enable 0b: Disable
23	CH5_ERR_EN	0h	RW	CH5 Error Interrupt Enable 1b: Enable 0b: Disable
22	CH5_QUE_EMP_EN	0h	RW	CH5 Que Empty Interrupt Enable 1b: Enable 0b: Disable
21	CH5_STOP_EN	0h	RW	CH5 Stop Interrupt Enable 1b: Enable 0b: Disable
20	CH5_END_EN	0h	RW	CH5 Completion Interrupt Enable 1b: Enable 0b: Disable
19	CH4_ERR_EN	0h	RW	CH4 Error Interrupt Enable 1b: Enable 0b: Disable

Bit	Bit Name	Initial Value	R/W	Description
18	CH4_QUE_EMP_EN	0h	RW	CH4 Que Empty Interrupt Enable 1b: Enable 0b: Disable
17	CH4_STOP_EN	0h	RW	CH4 Stop Interrupt Enable 1b: Enable 0b: Disable
16	CH4_END_EN	0h	RW	CH4 Completion Interrupt Enable 1b: Enable 0b: Disable
15	CH3_ERR_EN	0h	RW	CH3 Error Interrupt Enable 1b: Enable 0b: Disable
14	CH3_QUE_EMP_EN	0h	RW	CH3 Que Empty Interrupt Enable 1b: Enable 0b: Disable
13	CH3_STOP_EN	0h	RW	CH3 Stop Interrupt Enable 1b: Enable 0b: Disable
12	CH3_END_EN	0h	RW	CH3 Completion Interrupt Enable 1b: Enable 0b: Disable
11	CH2_ERR_EN	0h	RW	CH2 Error Interrupt Enable 1b: Enable 0b: Disable
10	CH2_QUE_EMP_EN	0h	RW	CH2 Que Empty Interrupt Enable 1b: Enable 0b: Disable
9	CH2_STOP_EN	0h	RW	CH2 Stop Interrupt Enable 1b: Enable 0b: Disable
8	CH2_END_EN	0h	RW	CH2 Completion Interrupt Enable 1b: Enable 0b: Disable
7	CH1_ERR_EN	0h	RW	CH1 Error Interrupt Enable 1b: Enable 0b: Disable
6	CH1_QUE_EMP_EN	0h	RW	CH1 Que Empty Interrupt Enable 1b: Enable 0b: Disable
5	CH1_STOP_EN	0h	RW	CH1 Stop Interrupt Enable 1b: Enable 0b: Disable
4	CH1_END_EN	0h	RW	CH1 Completion Interrupt Enable 1b: Enable 0b: Disable
3	CH0_ERR_EN	0h	RW	CH0 Error Interrupt Enable 1b: Enable 0b: Disable
2	CH0_QUE_EMP_EN	0h	RW	CH0 Que Empty Interrupt Enable 1b: Enable 0b: Disable
1	CH0_STOP_EN	0h	RW	CH0 Stop Interrupt Enable 1b: Enable 0b: Disable
0	CH0_END_EN	0h	RW	CH0 Completion Interrupt Enable 1b: Enable 0b: Disable

(44) DMAC Interrupt Status Register (PCI_EP_DMAINTS)

This register indicates the state of interrupts from the individual DMA channels.

Access Size : 32 bits
Offset Address : <PCI0_base> + 080Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CH7_ERR	CH7_QUE_EMP	CH7_STOP	CH7_END	CH6_ERR	CH6_QUE_EMP	CH6_STOP	CH6_END	CH5_ERR	CH5_QUE_EMP	CH5_STOP	CH5_END	CH4_ERR	CH4_QUE_EMP	CH4_STOP	CH4_END
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH3_ERR	CH3_QUE_EMP	CH3_STOP	CH3_END	CH2_ERR	CH2_QUE_EMP	CH2_STOP	CH2_END	CH1_ERR	CH1_QUE_EMP	CH1_STOP	CH1_END	CH0_ERR	CH0_QUE_EMP	CH0_STOP	CH0_END
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31	CH7_ERR	0h	RW1	Set when an error occurs during DMA transfer.
30	CH7_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
29	CH7_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
28	CH7_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
27	CH6_ERR	0h	RW1	Set when an error occurs during DMA transfer.
26	CH6_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
25	CH6_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
24	CH6_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
23	CH5_ERR	0h	RW1	Set when an error occurs during DMA transfer.
22	CH5_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
21	CH5_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
20	CH5_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
19	CH4_ERR	0h	RW1	Set when an error occurs during DMA transfer.
18	CH4_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
17	CH4_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.

Bit	Bit Name	Initial Value	R/W	Description
16	CH4_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
15	CH3_ERR	0h	RW1	Set when an error occurs during DMA transfer.
14	CH3_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
13	CH3_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
12	CH3_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
11	CH2_ERR	0h	RW1	Set when an error occurs during DMA transfer.
10	CH2_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
9	CH2_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
8	CH2_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
7	CH1_ERR	0h	RW1	Set when an error occurs during DMA transfer.
6	CH1_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
5	CH1_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
4	CH1_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1
3	CH0_ERR	0h	RW1	Set when an error occurs during DMA transfer.
2	CH0_QUE_EMP	0h	RW1	Set when the list is removed from the descriptor queue (transferred to the running descriptor list) and the QUE is empty.
1	CH0_STOP	0h	RW1	Set when the DMAC is prematurely stopped. DMA stops under the following conditions: - When RDMA_EN or QUE_EN is cleared to 0 by the S/W during DMA transfer execution and the processing of the request being issued ends.
0	CH0_END	0h	RW1	Set when the DMAC ends normally. Normal termination refers to the following conditions. - Transfer for the amount indicated by DMA_SIZE is completed. - At the end of the descriptor list when the EI field is 1

(45) DMAC Channel Control Register m (PCI_EP_DMACHCTLm) (m = 0 to 7)

This register sets the control method for each DMA channel. Set either register type or descriptor type.

- Setting QUE_EN = 1b and QUE_CLR = 1b is prohibited during register-type DMA transfer (TDMA_EN = 1b).
- Setting RDMA_EN = 1b is prohibited during descriptor-type DMA transfer (QUE_EN = 1b).

Access Size : 32 bits
Offset Address : <PCI0_base> + 0900h + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	QUE_CLR	-	-	-	-	-	-	QUE_EN	RDMA_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	QUE_CLR	0h	RW	QUE Clear Writing 1b clears QUE. All descriptor lists (waiting for execution and lists currently being executed) registered in QUE are cleared. Do not clear during DMA execution. The read value is always 0b. Writing 1b at the same time as setting QUE_EN is prohibited.
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	QUE_EN	0h	RW	QUE Enable Setting this bit to 1b enables the descriptor list registered in the descriptor queue and starts DMA transfer (descriptor-type). It is automatically cleared to 0 when DMA stops (both normal and abnormal). It is also possible to stop DMA by writing 0b. However, DMA will stop after finishing the currently executing request (both PCIe and AXI).
0	RDMA_EN	0h	RW	Register-type DMA transfer Enable Setting this bit to 1b starts DMA transfer (register-type) and performs the transfer set by RDMA_SIZE. This bit is automatically cleared to 0 when DMA transfer is completed or an error is detected and DMA ends. It is also possible to stop DMA by writing 0b. However, DMA will stop after completing the currently executing request (both PCIe and AXI). 1b: Register method DMA transfer start 0b: Stop register-type DMA transfer

(46) Descriptor Start Address (Lower) Register m (PCI_EP_DPSADRLm) (m = 0 to 7)

This register sets the descriptor queue list.

The setting value is registered as the lower 32 bits of DSA (DMA Start Address) in the queue list.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0908h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	QUE_ENTRY_DSA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	QUE_ENTRY_DSA[15:6]										QUE_ENTRY_DSA5:0[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	QUE_ENTRY_DSA[31:6]	0h	RW	Descriptor list queue registration register. This area will be the DSA. Sets the lower 32 bits of the address where the first descriptor is stored.
5 to 0	QUE_ENTRY_DSA[5:0]	0h	R	Descriptor list queue registration register. This area will be the DSA. Sets the lower 32 bits of the address where the first descriptor is stored. (16-byte alignment: Lower 6 bits are fixed to 0.)

Note 1. When these bits are read, any of the following contents will be read depending on the state of DMA.
DMA transfer in progress: The descriptor list in progress
DMA suspended (when QUE_EN is automatically cleared): Last executed list
DMA suspended (when QUE_EN S/W is cleared): Suspended (executing) list

(47) Descriptor Start Address (Higher) Registers m (PCI_EP_DPSADRUm) (m = 0 to 7)

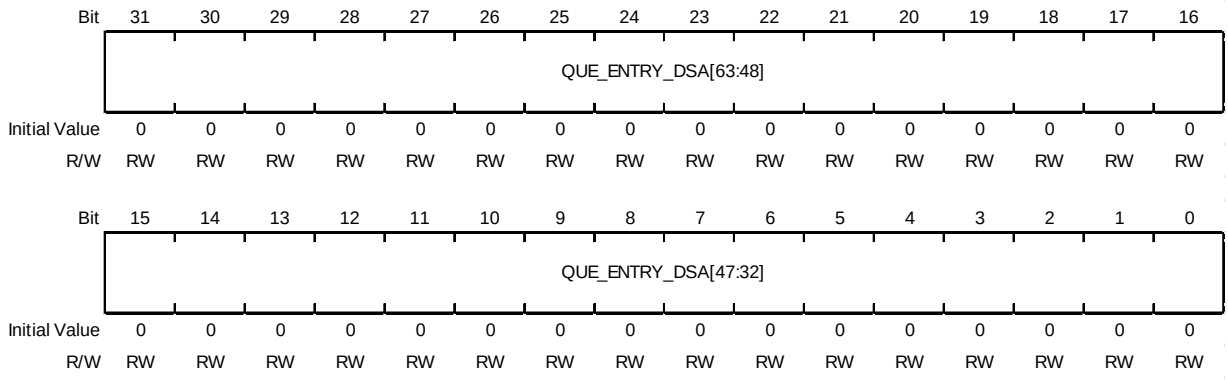
This register sets the descriptor queue list.

The setting value is registered as the higher 32 bits of DSA (DMA Start Address) in the queue list.

Access Size : 32 bits

Offset Address : <PCI0_base> + 090Ch + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	QUE_ENTRY_DSA[63:32]	0h	RW	Descriptor list queue registration register. This area will be the DSA. Set the upper 32 bits of the address where the first descriptor is stored.

Note 1. When these bits are read, any of the following contents will be read depending on the state of DMA.
DMA transfer in progress: The descriptor list in progress
DMA suspended (when QUE_EN is automatically cleared): Last executed list
DMA suspended (when QUE_EN S/W is cleared): Suspended (executing) list

(48) QUE Entry Register m (PCI_EP_QUEEm) (m = 0 to 7)

This register sets the descriptor queue list.

The setting value is registered as EI (End Interrupt), LS (List Stop), and LABEL of the queue list. Write to [31:24] to register to the queue.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0910h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	QUE_Registration[5:0]						QUE_ENTRY_EI	QUE_ENTRY_LS	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	QUE_ENTRY_LABEL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	QUE_Registration[5:0]	0h	RW	Enqueued by writing to [31:24]. The read value is always 0b.
25	QUE_ENTRY_EI	0h	RW	Indicates whether an interrupt (Interrupt Status CHx_END) is sent when the processing of the descriptor list is completed. 1b: signal an interrupt 0b: Do not signal interrupts
24	QUE_ENTRY_LS	0h	RW	Indicates whether to stop the DMA when processing of the descriptor list is complete. 1b: stop 0b: do not stop
23 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	QUE_ENTRY_LABEL[15:0]	0h	RW	There is no particular rule on how to set the labels in the list. You can set the value freely.

Note 1. When these bits are read, any of the following contents will be read depending on the state of DMA.
DMA transfer in progress: The descriptor list in progress
DMA suspended (when QUE_EN is automatically cleared): Last executed list
DMA suspended (when QUE_EN S/W is cleared): Suspended (executing) list

(49) DMA Descriptor Control (Descriptor 00h) Register m (PCI_EP_DMADPCTLm) (m = 0 to 7)

This register indicates the field value at offset 00h in the descriptor table. Only effective when the descriptor-type DMA transfer is selected (the value read has no meaning in the case of register-type transfer).

Access Size : 32 bits

Offset Address : <PCI0_base> + 0920h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSCFM[3:0]				-	WBD	LE	LV	D	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DSCFM[3:0]	0h	R	Shows the value of the DSCFM field in the running descriptor table.
27	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
26	WBD	0h	R	Shows the value of the WBD field in the running descriptor table.
25	LE	0h	R	Shows the value of the LE field in the running descriptor table.
24	LV	0h	R	Shows the value of the LV field in the running descriptor table.
23	D	0h	R	Shows the value of the D field in the running descriptor table.
22 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	STS[15:0]	0h	R	Shows the value of the STS field in the running descriptor table.

(50) DMA Transaction Control (Descriptor 04h) Register m (PCI_EP_DMATCTLm) (m = 0 to 7)

This register controls DMA transfer to the AXI and PCIe.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0924h + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	CCH_L[3:0]				CCH_D[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	DMA_TC[2:0]			-	-	DMA_ATB[1:0]		-	DMA_FUNC[2:0]			-	-	-	DMA_DIR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	R	R	RW	RW	R	RW	RW	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 20	CCH_L[3:0]	0h	RW	Indicates the value of A*CACHE[3:0] to be issued to AXI. CCH_L is output when issuing an AXI request containing the last byte in a transfer indicated by SIZE. Recommended value is 0000b. Bit[3] 1b: write allocatable 0b: not write allocatable Bit[2] 1b: read assignable 0b: not read assignable Bit[1] 1b: cacheable 0b: non-cacheable Bit[0] 1b: bufferable 0b: not bufferable
19 to 16	CCH_D[3:0]	0h	RW	Indicates the value of A*CACHE[3:0] to be issued to AXI. CCH_D is output when issuing an AXI request other than the output condition of CCH_L. The recommended value is 0001b when DIR=0 (PCIe→AXI) and 0000b when DIR=1 (AXI→PCIe). Bit[3] 1b: write allocatable 0b: not write allocatable Bit[2] 1b: read assignable 0b: not read assignable Bit[1] 1b: cacheable 0b: non-cacheable Bit[0] 1b: bufferable 0b: not bufferable
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 12	DMA_TC[2:0]	0h	RW	Traffic class to issue to PCIe. Specifies the traffic class value for requests issued to PCIe. Note: This IP does not support Virtual Channel, so please use it with 000b fixed.
11, 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9, 8	DMA_ATB[1:0]	0h	RW	Attributes to issue to PCIe. bit[1]: Relaxed Ordering (Unsupported function: 0b fixed) bit[0]: No Snoop (0b recommended)
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 4	DMA_FUNC [2:0]	0h	RW	Specify the function number of the request issued to PCIe.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	DMA_DIR	0h	RW	Sets the DMA transfer direction. 1b: AXI to PCIe 0b: PCIe to AXI

Note 1. When descriptor-type DMA transfer is in progress, writing is prohibited and the value read indicates the value of the CCH_L, CCH_D, TC, ATB, or DIR field of the descriptor table being executed.

(51) DMA Size (Descriptor 08h) Register m (PCI_EP_DMASIZEm) (m = 0 to 7)

This register sets the number of bytes for DMA transfer. The set value is reflected in the offset 0Ch field of the descriptor table.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0928h + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_SIZE[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_SIZE[15:3]													DMA_SIZE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DMA_SIZE [31:3]	0h	RW	Sets the number of DMA transfer bytes. The lower 3 bits are fixed to 0 because it is an 8-byte aligned setting.
2 to 0	DMA_SIZE [2:0]	0h	R	Sets the number of DMA transfer bytes. The lower 3 bits are fixed to 0 because it is an 8-byte aligned setting.

Note 1. When descriptor-type DMA transfer is in progress, writing is prohibited and the value read indicates the value of the SIZE field of the descriptor table being executed. The number of transfer bytes when 0000_0000h is set is 1_0000_0000h.

(52) DMA Source Lower Address (Descriptor 10h) Register m (PCI_EP_DMASLAm) (m = 0 to 7)

This register sets the lower 32 bits of the source start address for DMA transfer. The set value is reflected in the offset 10h field of the descriptor table.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0930h + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_S_ADDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_S_ADDR[15:3]													DMA_S_ADDR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DMA_S_ADDR [31:3]	0h	RW	Set the lower 32 bits of the transfer source start address for DMA transfer. The lower 3 bits are fixed to 0 because it is an 8-byte aligned setting. (When the data bus width is 128 bits, 16-byte alignment: lower 4 bits are fixed to 0.)
2 to 0	DMA_S_ADDR [2:0]	0h	R	Set the lower 32 bits of the transfer source start address for DMA transfer. The lower 3 bits are fixed to 0 because it is an 8-byte aligned setting. (When the data bus width is 128 bits, 16-byte alignment: lower 4 bits are fixed to 0.)

Note 1. If the Source Address indicates the PCIe space (DIR=0), the PCIe request address is set in combination with the DMA PCIe Higher Address (Descriptor 10h). When descriptor-type DMA transfer is in process, writing to these bits is prohibited and the value read indicates the value of the SA field in the descriptor table being executed.

(53) DMA Source Higher Address (Descriptor 14h) Register m (PCI_EP_DMASUAm) (m = 0 to 7)

This register sets the higher 32 bits of the source start address for DMA transfer. The set value is reflected in the offset 14h field of the descriptor table.

Access Size : 32 bits
Offset Address : <PCI0_base> + 0934h + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_S_ADDR[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_S_ADDR[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_S_ADDR [63:32]	0h	RW	Set the upper 32 bits of the transfer source start address for MA transfer.

Note 1. If the Source Address indicates the PCIe space (DIR=0), the PCIe request address is set in combination with the DMA PCIe Higher Address (Descriptor 10h). When descriptor-type DMA transfer is in process, writing to these bits is prohibited and the value read indicates the value of the SA field in the descriptor table being executed.

(54) DMA Destination Lower Address (Descriptor 18h) Register m (PCI_EP_DMADESTLAm) (m = 0 to 7)

This register sets the lower 32 bits of the source start address for DMA transfer. The set value is reflected in the offset 18h field of the descriptor table.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0938h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_D_ADDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_D_ADDR[15:3]													DMA_D_ADDR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DMA_D_ADDR [31:3]	0h	RW	Set the lower 32 bits of the transfer destination start address for DMA transfer. The lower 3 bits are fixed to 0 because it is an 8-byte aligned setting.
2 to 0	DMA_D_ADDR [2:0]	0h	R	Set the lower 32 bits of the transfer destination start address for DMA transfer. The lower 3 bits are fixed to 0 because it is an 8-byte aligned setting.

Note 1. If the Destination Address indicates the PCIe space (DIR=1), the PCIe request address is set in combination with the DMA PCIe Higher Address (Descriptor 10h). When descriptor-type DMA transfer is in process, writing to these bits is prohibited and the value read indicates the value of the DA field in the descriptor table being executed.

(55) DMA Destination Higher Address (Descriptor 1Ch) Register m (PCI_EP_DMADESTUAm) (m = 0 to 7)

This register sets the higher 32 bits of the source start address for DMA transfer. The set value is reflected in the offset 1Ch field of the descriptor table.

Access Size : 32 bits

Offset Address : <PCI0_base> + 093Ch + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_D_ADDR[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_D_ADDR[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_D_ADDR [63:32]	0h	RW	Set the upper 32 bits of the transfer destination start address for DMA transfer.

Note 1. If the Destination Address indicates the PCIe space (DIR=1), the PCIe request address is set in combination with the DMA PCIe Higher Address (Descriptor 10h). When descriptor-type DMA transfer is in process, writing to these bits is prohibited and the value read indicates the value of the DA field in the descriptor table being executed.

(56) DMA Descriptor Lower Link Pointer (Descriptor 20h) Register m (PCI_EP_DMADPLLPm) (m = 0 to 7)

This register indicates the field value at offset 20h in the descriptor table. Only effective when descriptor-type DMA transfer is selected (the value read has no meaning in the case of register-type transfer).

Access Size : 32 bits

Offset Address : <PCI0_base> + 0940h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_LP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_LP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_LP[31:0]	0h	R	Indicates the value of the LP field of the running descriptor table.

(57) DMA Descriptor Higher Link Pointer (Descriptor 24h) Register m (PCI_EP_DMADPULPm) (m = 0 to 7)

This register indicates the field value at offset 24h in the descriptor table. Only effective when descriptor-type DMA transfer is selected (the value read has no meaning in the case of register-type transfer).

Access Size : 32 bits

Offset Address : <PCI0_base> + 0944h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_LP[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_LP[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_LP[63:32]	0h	R	Indicates the value of the LP field of the running descriptor table.

(58) DMA Rest Size Register m (PCI_EP_DMARESTSIZm) (m = 0 to 7)

This register indicates the number of bytes for which DMA transfer has not yet been completed.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0950h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMA_REST_SIZE[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMA_REST_SIZE[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DMA_REST_SIZE[31:0]	0h	R	Displays the number of bytes for which DMA transfer has not been completed. (Register/descriptor-type common)

(59) AXI Request Address (Lower) Register m (PCI_EP_AREQALm) (m = 0 to 7)

This register indicates the lower 32 bits of the address of the current or most recently completed AXI transfer.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0960h + 080h x m (m = 0 to 7) (channel offset))

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AXI_REQ_ADDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AXI_REQ_ADDR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	AXI_REQ_ADDR[31:0]	0h	R	Displays the lower 32 bits of the address of the current or most recently completed AXI transfer. (Register/descriptor-type common)

(60) AXI Request Address (Higher) Register m (PCI_EP_AREQAUm) (m = 0 to 7)

This register indicates the higher 32 bits of the address of the current or most recently completed AXI transfer.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0964h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AXI_REQ_ADDR[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AXI_REQ_ADDR[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	AXI_REQ_ADDR[63:32]	0h	R	Displays the upper 32 bits of the address of the current or most recently completed AXI transfer. (Register/descriptor method common)

(61) PCIe Request Address (Lower) Register m (PCI_EP_PREQALm) (m = 0 to 7)

This register indicates the lower 32 bits of the address during PCIe transfer that is in progress or for the most recently completed transfer.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0968h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PCIE_REQ_ADDR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCIE_REQ_ADDR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PCIE_REQ_ADDR[31:0]	0h	R	Displays the lower 32 bits of the address of the current or most recently completed PCIe transfer. (Register/descriptor-type common)

(62) PCIe Request Address (Higher) Register m (PCI_EP_PREQAUm) (m = 0 to 7)

This register indicates the higher 32 bits of the address during PCIe transfer that is in progress or for the most recently completed transfer.

Access Size : 32 bits
Offset Address : <PCI0_base> + 096Ch + 080h x m (m = 0 to 7) (channel offset)
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PCIE_REQ_ADDR[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCIE_REQ_ADDR[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PCIE_REQ_ADDR[63:32]	0h	R	Displays the upper 32 bits of the address of the current or most recently completed PCIe transfer. (Register/descriptor-type common)

(63) QUE Status Register m (PCI_EP_QUESTAm) (m = 0 to 7)

This register indicates the state of the descriptor queue.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0970h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	GO_LIST	LIST_NUM[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	GO_LIST	0h	R	Shows whether there is a running descriptor list. 1b: Yes 0b: No
3 to 0	LIST_NUM[3:0]	0h	R	Displays the number of descriptor lists loaded on QUE (not including the list currently being executed). New registration to QUE (write access to QUE Entry) when this register indicates 8h is invalid (discarded).

(64) DMAC Error Status Register m (PCI_EP_DMACESTAm) (m = 0 to 7)

This register indicates the error status of the DMAC.

Access Size : 32 bits

Offset Address : <PCI0_base> + 0978h + 080h x m (m = 0 to 7) (channel offset)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	CFG_BM_DIS_EP	BME_SUP	BME_DOWN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	MOR_CD_PERR	MOR_CH_PERR	MOR_EP_ERR	MOR_STATUS[2:0]			-	-	-	-	-	-	AXI_RESP[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

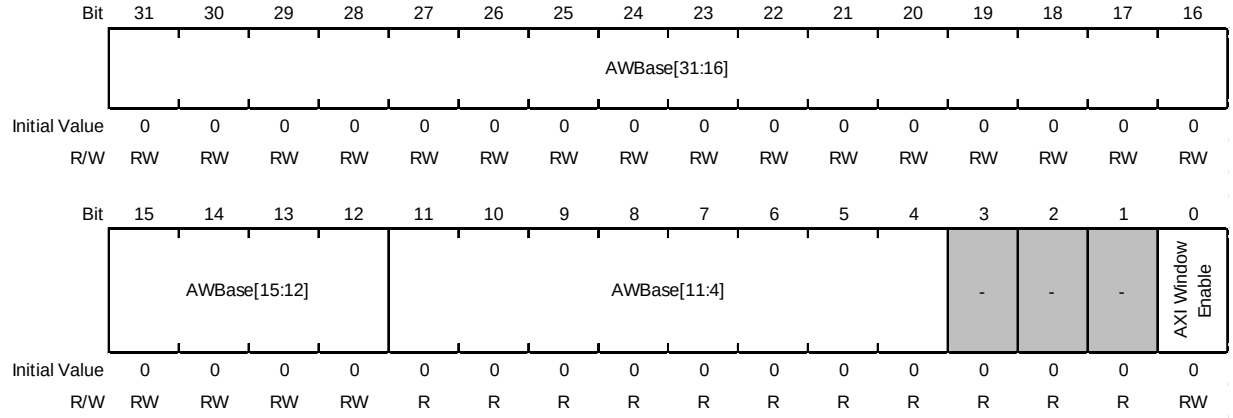
Bit	Bit Name	Initial Value	R/W	Description
31 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18	CFG_BM_DIS_EP	0h	R	CHx_ERR is set when the macro operates as an End Point and detects the state of Bus Master Enable Off (Configuration Space 004h bit [2]=0). Holds value until CHx_ERR is cleared.
17	BME_SUP	0h	R	It is set when a sleep signal from the PCIe core is detected as a cause for setting CHx_ERR. Holds value until CHx_ERR is cleared. (Valid only in Endpoint mode)
16	BME_DOWN	0h	R	It is set when a stop signal from the PCIe core is detected as a cause for setting CHx_ERR. Holds value until CHx_ERR is cleared.
15	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.
13	MOR_CD_PERR	0h	R	It is set when MOR_CD_PERR is detected as a set factor for CHx_ERR. Holds value until CHx_ERR is cleared.
12	MOR_CH_PERR	0h	R	It is set when MOR_CH_PERR is detected as a set factor for CHx_ERR. Holds value until CHx_ERR is cleared.
11	MOR_EP_ERR	0h	R	It is set when it is Poisoned Completion as a set factor for CHx_ERR. Holds value until CHx_ERR is cleared.
10 to 8	MOR_STATUS [2:0]	0h	R	Indicates the value when MOR_STATUS is other than 000b (Success) as a set factor for CHx_ERR. Holds value until CHx_ERR is cleared. 000b: initial value 001b: Unsupported Request 010b: CRS 011b: Completion Timeout 100b: Completer Abort 101b: Unexpected Completion 110b: Reserved 111b: Mismatched Length (Length Overrun)
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.
3	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	AXI_RESP[1:0]	0h	R	Displays the slave response during AXI Master transactions. It is updated when CHx_ERR is set and retains the value until the bit is cleared. 00b: Initial value 01b: Reserved

(65) AXI Window Base (Lower) m Register (Function #n) (PCI_EP_AWBASELm_Fn) (m = 0 to 7, n = 0, 1)

This register is for setting windows for address conversion in access from PCIe to AXI. It sets the base address on the PCI. The areas are set in 4-Kbyte boundaries.

Access Size : 32 bits
Offset Address : <PCI0_base> + 1000h + Offset: 00h/ 20h/ 40h/ ・・・ / E0h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 12	AWBase[31:12]	0h	RW	Window setting register for address conversion when accessing AXI from PCIe. The configurable area is the 4K boundary.
11 to 4	AWBase[11:4]	0h	R	Fixed to 00h
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	AXI Window Enable	0h	RW	Enable setting of AXI Window. 0b: Window disabled 1b: Window enabled

(66) AXI Window Base (Higher) m Register (Function #n) (PCI_EP_AWBASEUm_Fn) (m = 0 to 7, n = 0, 1)

This register is for setting windows for address conversion in access from PCIe to AXI.

Access Size : 32 bits
Offset Address : <PCI0_base> + 1000h + Offset: 04h/ 24h/ 44h/ ・・・ / E4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AWBase[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AWBase[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	AWBase[63:32]	0h	RW	Window setting register for address conversion when accessing AXI from PCIe.

(67) AXI Window Mask (Lower) m Register (Function #n) (PCI_EP_AWMASKLm_Fn) (m = 0 to 7, n = 0, 1)

This register is for setting windows for address conversion in access from PCIe to AXI. The window is set as the area corresponding to the number of set bits from the address set in the AWBase register. The area which can be set is 4K × 2^N bytes.

Access Size : 32 bits
Offset Address : <PCI0_base> + 1000h + Offset: 08h/ 28h/ 48h/ ・・・ / E8h
Initial Value : 0000_0FFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AWMask[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AWMask[15:12]				AWMask[11:0]											
Initial Value	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	AWMask [31:12]	0h	RW	Set the window in the area of the set number of bits from the address set in the AWBase register. Set 1b from the lower bit. Therefore, the area that can be set is 4K x 2^N bytes.
11 to 0	AWMask[11:0]	FFFh	R	Fixed to FFFh

(68) AXI Window Mask (Higher) m Register (Function #n) (PCI_EP_AWMASKUm_Fn) (m = 0 to 7, n = 0, 1)

This register is for setting windows for address conversion in access from PCIe to AXI.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000h + Offset: 0Ch/ 2Ch/ 4Ch/ ・・・ / Ech

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	AWMask[62:48]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AWMask[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 0	AWMask [62:32]	0h	RW	Set the window in the area of the set number of bits from the address set in the AWBase register. Set 1b from the lower bit.

(69) AXI Destination (Lower) m Register (Function #n) (PCI_EP_ADESTLm_Fn) (m = 0 to 7, n = 0, 1)

This register is for setting windows for address conversion in access from PCIe to AXI. It sets the base address on the AXI. The areas are set in 4-Kbyte boundaries.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000h + Offset: 10h/ 30h/ 50h/ 70h/ ・・・ / F0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADest[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADest[15:12]				ADest[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	ADest[31:12]	0h	RW	Sets the window base point in the address space from the AXI side. The configurable area is the 4K boundary.
11 to 0	ADest[11:0]	0h	R	Fixed to 000h

(70) AXI Destination (Higher) m Register (Function #n) (PCI_EP_ADESTUm_Fn) (m = 0 to 7, n = 0, 1)

This register is for setting windows for address conversion in access from PCIe to AXI.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000h + Offset: 14h/ 34h/ 54h/ 74h/ ・・・ / F4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADest[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADest[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ADest[63:32]	0h	RW	Sets the window base point in the address space from the AXI side.

(71) PCIe Window Base (Lower) m Register (Function #n) (PCI_EP_PWBASELm_Fn) (m = 0 to 7, n = 0, 1)

This register is for setting windows for address conversion in access from AXI to PCIe. It sets the base address on the AXI. The areas are set in 4-Kbyte boundaries.

Access Size : 32 bits
Offset Address : <PCI0_base> + 1000h + Offset: 100h/ 120h/ 140h/ ・・・ / 1E0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PWBase[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PWBase[15:12]				PWBase[11:4]								-	-	-	PCIe Window Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	PWBase[31:12]	0h	RW	Sets the base point of the address on the AXI side. The configurable area is the 4K boundary.
11 to 4	PWBase[11:4]	0h	R	Fixed to 00h
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	PCIe Window Enable	0h	RW	Enable setting of PCIe Window. 0b: Window disabled 1b: Window enabled

(72) PCIe Window Base (Higher) m Register (Function #n) (PCI_EP_PWBASEUm_Fn) (m = 0 to 7, n = 0, 1)

This register is for setting windows for address conversion in access from AXI to PCIe.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000h + Offset: 104h/ 124h/ 144h/ ・・・ / 1E4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PWBase[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PWBase[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PWBase[63:32]	0h	RW	Sets the base point of the address on the AXI side. The configurable area is the 4K boundary.

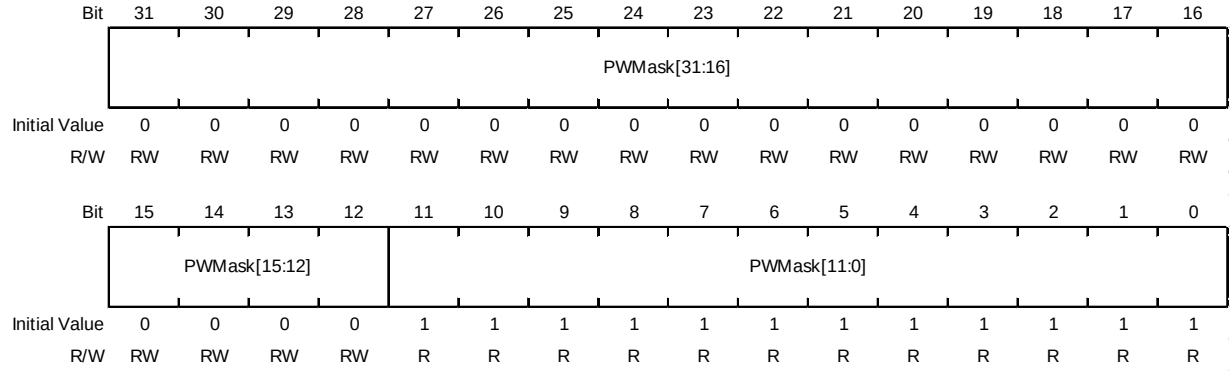
(73) PCIe Window Mask (Lower) m Register (Function #n) (PCI_EP_PWMASKLm_Fn) (m = 0 to 7, n = 0, 1)

This register is for setting windows for address conversion in access from AXI to PCIe. The window is set as the area corresponding to the number of set bits from the address set in the PWBase register.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000h + Offset: 108h/ 128h/ 148h/ ・・・ / 1E8h

Initial Value : 0000_0FFFh

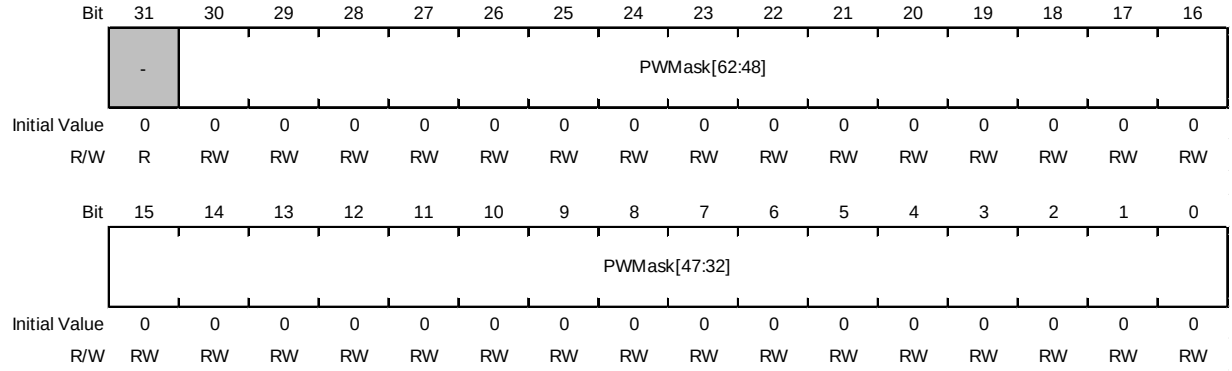


Bit	Bit Name	Initial Value	R/W	Description
31 to 12	PWMask [31:12]	0h	RW	Set the window in the area of the set number of bits from the address set in the PWBase register. Set 1b from the lower bit.
11 to 0	PWMask[11:0]	FFFh	R	Fixed to FFFh

(74) PCIe Window Mask (Higher) m Register (Function #n) (PCI_EP_PWMASKUm_Fn) (m = 0 to 7, n = 0, 1)

This register is for setting windows for address conversion in access from AXI to PCIe. The window is set as the area corresponding to the number of set bits from the address set in the PWBase register.

Access Size : 32 bits
Offset Address : <PCI0_base> + 1000h + Offset: 10Ch/ 12Ch/ 14Ch/ ・・・ / 1ECh
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 0	PWMask [62:32]	0h	RW	Set the window in the area of the set number of bits from the address set in the PWBase register. Set 1b from the lower bit.

(75) PCIe Destination m (Lower) Register (Function #n) (PCI_EP_PDESTLOm_Fn) (m = 0 to 7, n = 0, 1)

This register is for setting windows for address conversion in access from AXI to PCIe. It sets the base address on the PCIe. The areas are set in 4-Kbyte boundaries.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000h + Offset: 110h/ 130h/ 150h/ 170h/ ・・・ / 1F0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PDest[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PDest[15:12]				PDest[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	PDest[31:12]	0h	RW	Set the window base point in the address space on the PCIe side. The configurable area is the 4k boundary.
11 to 0	PDest[11:0]	0h	R	Fixed to 000h

(76) PCIe Destination m (Higher) Register (Function #n) (PCI_EP_PDESTUPm_Fn) (m = 0 to 7, n = 0, 1)

This register is for setting windows for address conversion in access from AXI to PCIe.

Access Size : 32 bits

Offset Address : <PCI0_base> + 1000h + Offset: 114h/ 134h/ 154h/ 174h/ ・・・ / 1F4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PDest[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PDest[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PDest[63:32]	0h	RW	Set the window base point in the address space on the PCIe side.

6.6.4.2.4 PCI Express Configuration Register Descriptions (Type0)

(1) Vendor and Device ID (Function #n) Register (PCI_EP_VID_Fn) (n = 0, 1)

This register indicates the vendor and device ID.

Refer to **6.6.6.1.1 Changing the Initial Values of the Registers**.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Device ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Vendor ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Device ID[15:0]	0h	R (RW)	Used by the manufacturer specified by the Vendor ID to identify the manufactured device. Set a fixed value.
15 to 0	Vendor ID[15:0]	0h	R (RW)	Represents the manufacturer of the device. Set a fixed value.

Table 6.6-65 Valid Reset Signal

Reset Signal	Device ID	Vendor ID
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		
FLR		

(2) Command and Status (Function #n) (PCI_EP_COM_STA_Fn) (n = 0, 1)

This register specifies the command and the status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6004h
Initial Value : 0010_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Detected Parity Error	Signaled System Error	Received Master Abort	Received Target Abort	Signaled Target Abort	-	-	Master Data Parity Error	-	-	-	Capabilities List	Interrupt Status	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W(PCIe)	RW1	RW1	RW1	RW1	RW1	R	R	RW1	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	R	R	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	Interrupt Disable	-	SERR# Enable	-	Parity Error Response	-	-	-	Bus Master Enable	Memory Space Enable	IO Space Enable
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	RW	R	RW	R	RW	R	R	R	RW	RW	R
R/W(UDL)	R	R	R	R	R	RW	R	RW	R	RW	R	R	R	RW	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31	Detected Parity Error	0h	RW1 (RW)	Set to 1b when a Poisoned TLP is received, regardless of the setting of the Parity Error Response bit.
30	Signaled System Error	0h	RW1 (RW)	Set to 1b when the SERR Enable bit is 1b and this macro sends an ERR_FATAL or ERR_NONFATAL Message.
29	Received Master Abort	0h	RW1 (RW)	Set to 1b when the Completion Status field receives a Completion of an Unsupported Request.
28	Received Target Abort	0h	RW1 (RW)	Set to 1b when the Completion Status field receives a Completion for Completer Abort.
27	Signaled Target Abort	0h	RW1 (RW)	Set to 1b when the Completion Status field has sent a Completion of Completer Abort.
26, 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	Master Data Parity Error	0h	RW1 (RW)	Set to 1b when the Parity Error Response bit is set to 1b and the following two conditions occur: 1) Requester (BME) received a Poisoned Completion TLP. 2) Requester (BME) sent Poisoned Write Request TLP. This bit is not set to 1b if the Parity Error Response bit is 0b.
23 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	Capabilities List	1h	R	This bit is fixed to 1b because all PCI Express devices must implement PCI Express Capability.
19	Interrupt Status	0h	R	Indicates the interrupt status of the device.
18, 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	Interrupt Disable	0h	RW	Suppress transmission of Assert_INTx Message. When set to 1b, INTx Message cannot be sent. If this bit is set to 1b while an Assert_INTx Message has been sent, a Deassert_INTx Message must be sent.
9	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
8	SERR# Enable	0h	RW	When set to 1b, the Root Complex is notified of Non-Fatal Errors and Fatal Errors by Message Transaction. Note: Even if this bit is not set, if a bit related to Error Reporting in the Device Control register of PCI Express Capability is set to "1", an error is notified to the Root Complex by Message Transaction.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6	Parity Error Response	0h	RW	Controls the behavior when sending/receiving Poisoned TLP. Note: Error logging to the Detected Parity Error field of the Status register, the Device Status register of PCI Express Capability, and the Uncorrectable Error Status register of Advanced Error Reporting Capability is performed regardless of the setting of this bit.
5 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	Bus Master Enable	0h	RW	Controls whether to operate as a bus master.
1	Memory Space Enable	0h	RW	Controls whether to respond to accesses to memory space.
0	IO Space Enable	0h	R	0b fixed. It does not support access to the I/O space.

Table 6.6-66 Valid Reset Signal

Reset Signal	Detected Parity Error	Signaled System Error	Received Master Abort	Received Target Abort	Signaled Target Abort	Master Data Parity Error
RST_LOAD_B						
RST_RSM_B						
RST_CFG_B	✓	✓	✓	✓	✓	✓
FLR	✓	✓	✓	✓	✓	✓

Reset Signal	Interrupt Disable	SERR# Enable	Parity Error Response	Bus Master Enable	Memory Space Enable
RST_LOAD_B					
RST_RSM_B					
RST_CFG_B	✓	✓	✓	✓	✓
FLR	✓	✓	✓	✓	✓

(3) Revision ID and Class Code (Function #n) (PCI_EP_RID_CC_Fn) (n = 0, 1)

This register indicates the revision ID and the class code.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Class Code[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Class Code[15:8]								Revision ID[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	Class Code[31:8]	0h	R (RW)	This is information that indicates the device type and function, and the definition of the value is divided into the following two bytes by the PCI SIG. Bit[31:24] base class Bit[23:16] sub-class Bit[15:8] programming interface Please set a fixed value.
7 to 0	Revision ID[7:0]	0h	R (RW)	An 8-bit ID used to represent the revision of a specific device specified by Vendor ID and Device ID. Please set a fixed value.

Table 6.6-67 Valid Reset Signal

Reset Signal	Class Code	Revision ID
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		
FLR		

(4) Cache Line and Header Type (Function #n) (PCI_EP_CL_HT) (n = 0, 1)

This register indicates the cache line size and the header type.

Access Size : 32 bits
Offset Address : <PCI0_base> + 600Ch
Initial Value : 0080_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	Header Type[7:0]							
Initial Value	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	Cache Line Size[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	Header Type[7:0]	80h	R	Multi-Function Device bit = 1b (80h fixed)
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	Cache Line Size[7:0]	0h	RW (R)	Implemented as a read/write field for legacy compatibility, but the value set has no effect on this device.

Table 6.6-68 Valid Reset Signal

Reset Signal	Cache Line Size
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓
FLR	✓

(5) Base Address Register 0 (Function #n) (PCI_EP_BAR0_Fn) (n = 0, 1)

This register forms a 64-bit memory space in combination with Base Address Register 1 (BAR1).

Access Size : 32 bits
Offset Address : <PCI0_base> + 6010h
Initial Value : 0000_0004h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAR0[27:12]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAR0[11:0]												Prefetch	Type	-	MSI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	BAR0[27:0]	0h	RW	Indicates the base address. Depending on the size of the address block required, some lower bits of this field are implemented as Read Only bits fixed at 0b. In this macro, the Read Only bits can be set with the Base Address Register Mask00 (Lower)(Offset: 0A0h).
3	Prefetch	0h	R (RW)	0b: disable, 1b: enable
2	Type	1h	R (RW)	0b: 32 bit address, 1b: 64-bit address 1b fixed to use 64-bit address
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	MSI	0h	R (RW)	Fixed to 0b to indicate the memory space.

Table 6.6-69 Valid Reset Signal

Reset Signal	BAR0	Prefetch	Type	MSI
RST_LOAD_B		✓	✓	✓
RST_RSM_B				
RST_CFG_B	✓			
FLR	✓			

(6) Base Address Register 1 (Function #n) (PCI_EP_BAR1_Fn) (n = 0, 1)

This register forms a 64-bit memory space in combination with Base Address Register 0 (BAR0).

Access Size : 32 bits
Offset Address : <PCI0_base> + 6014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAR1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAR1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BAR1[31:0]	0h	RW	Base Address Register 1 (64-bit Upper Address) Shows the upper 32 bits of the base address. In this macro, the Read Only bits can be set with the Base Address Register Mask00 (Higher)(Offset: 0A4h).

Table 6.6-70 Valid Reset Signal

Reset Signal	BAR1
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓
FLR	✓

(7) Base Address Register 2 (Function #n) (PCI_EP_BAR2_Fn) (n = 0, 1)

This register forms a 64-bit memory space in combination with Base Address Register 3 (BAR3).

Access Size : 32 bits
Offset Address : <PCI0_base> + 6018h
Initial Value : 0000_000Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAR2[27:12]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAR2[11:0]												Prefetch	Type	-	MSI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	BAR2[27:0]	0h	RW	Indicates the base address. Depending on the size of the address block required, some lower bits of this field are implemented as Read Only bits fixed at 0b. In this macro, the Read Only bits can be set with the Base Address Register Mask01 (Lower)(Offset: 0A8h).
3	Prefetch	1h	R (RW)	0b: disable, 1b: enable
2	Type	1h	R (RW)	0b: 32 bit address, 1b: 64-bit address 1b fixed to use 64-bit address
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	MSI	0h	R (RW)	Fixed to 0b to indicate the memory space.

Table 6.6-71 Valid Reset Signal

Reset Signal	BAR2	Prefetch	Type	MSI
RST_LOAD_B		✓	✓	✓
RST_RSM_B				
RST_CFG_B	✓			
FLR	✓			

(8) Base Address Register 3 (Function #n) (PCI_EP_BAR3_Fn) (n = 0, 1)

This register forms a 64-bit memory space in combination with Base Address Register 2 (BAR2).

Access Size : 32 bits
Offset Address : <PCI0_base> + 601Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAR3[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAR3[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BAR3[31:0]	0h	RW	Indicates the base address. Depending on the size of the address block required, some lower bits of this field are implemented as Read Only bits fixed at 0b. In this macro, the Read Only bits can be set with the Base Address Register Mask01 (Higher)(Offset: 0ACh).

Table 6.6-72 Valid Reset Signal

Reset Signal	BAR3
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓
FLR	✓

(9) Base Address Register 4 (Function #n) (PCI_EP_BAR4_Fn) (n = 0, 1)

This register forms a 64-bit memory space in combination with Base Address Register 5 (BAR5).

Access Size : 32 bits
Offset Address : <PCI0_base> + 6020h
Initial Value : 0000_0004h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAR4[27:12]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAR4[11:0]												Prefetch	Type	-	MSI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	BAR4[27:0]	0h	RW	Indicates the base address. Depending on the size of the address block required, some lower bits of this field are implemented as Read Only bits fixed at 0b. In this macro, the Read Only bits can be set with the Base Address Register Mask02 (Lower)(Offset: 0B0h).
3	Prefetch	1h	R (RW)	0b: disable, 1b: enable
2	Type	1h	R (RW)	0b: 32 bit address, 1b: 64-bit address 1b fixed to use 64bitAddress
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	MSI	0h	R (RW)	Fixed to 0b to indicate the memory space.

Table 6.6-73 Valid Reset Signal

Reset Signal	BAR4	Prefetch	Type	MSI
RST_LOAD_B		✓	✓	✓
RST_RSM_B				
RST_CFG_B	✓			
FLR	✓			

(10) Base Address Register 5 (Function #n) (PCI_EP_BAR5_Fn) (n = 0, 1)

This register forms a 64-bit memory space in combination with Base Address Register 4 (BAR4).

Access Size : 32 bits
Offset Address : <PCI0_base> + 6024h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAR5[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAR5[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BAR5[31:0]	0h	RW	Indicates the base address. Depending on the size of the address block required, some lower bits of this field are implemented as Read Only bits fixed at 0b. In this macro, the Read Only bits can be set with the Base Address Register Mask02 (Higher)(Offset: 0B4h).

Table 6.6-74 Valid Reset Signal

Reset Signal	BAR5
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓
FLR	✓

(11) Cardbus CIS Pointer (Function #n) (PCI_EP_CCISP_Fn) (n = 0, 1)

This function is not supported.

(12) Subsystem ID (Function #n) (PCI_EP_SUBSID_Fn) (n = 0, 1)

This register indicates the subsystem ID.

Access Size : 32 bits
Offset Address : <PCI0_base> + 602Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Subsystem ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Subsystem Vendor ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	Subsystem ID[15:0]	0h	R (RW)	Subsystem Vendor 16-bit ID used to identify subsystems manufactured by the manufacturer specified by the ID. Set fixed value
15 to 0	Subsystem Vendor ID[15:0]	0h	R (RW)	A 16-bit ID representing the manufacturer of the add-in card or subsystem that contains this device. Set fixed value

Table 6.6-75 Valid Reset Signal

Reset Signal	Subsystem ID	Subsystem Vendor ID
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		
FLR		

(13) Expansion ROM Base Address (Function #n) (PCI_EP_EROMBA_Fn) (n = 0, 1)

This function is not supported.

(14) Capabilities Pointer (Function #n) (PCI_EP_CP_Fn) (n = 0, 1)

This register indicates start address for implementing the capability.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6034h
Initial Value : 0000_0040h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	Capabilities Pointer[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	Capabilities Pointer[7:0]	40h	R	Capability implementation start address 40H PCI Power Management Capability is implemented from 40h.

(15) Interrupt Register (Function #n) (PCI_EP_INT_Fn) (n = 0, 1)

This register assigns an interrupt to the terminal.

Access Size : 32 bits
Offset Address : <PCI0_base> + 603Ch
Initial Value : 0000_0100h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Interrupt Pin[7:0]								Interrupt Line[7:0]							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 8	Interrupt Pin[7:0]	1h	R (RW)	Specifies the message to be issued on the INTX_EP_F* pin. 01h : Assert_INTA# (default) 02h : Assert_INTB# 03h : Assert_INTC# 04h : Assert_INTD# 05h-FFh: (Reserved)
7 to 0	Interrupt Line[7:0]	0h	RW	Indicates which line of the system's interrupt lines the device's interrupt output is connected to as a result of system configuration. Values are set by system initialization programs and read by device drivers and the OS as needed. This is a software register and there is no relation between device operation and register value.

Table 6.6-76 Valid Reset Signal

Reset Signal	Interrupt Pin	Interrupt Line
RST_LOAD_B	✓	
RST_RSM_B		
RST_CFG_B		✓
FLR		✓

(16) PM Capabilities (Function #n) (PCI_EP_PMC_Fn) (n = 0, 1)

This register indicates various support information.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6040h
Initial Value : 4803_E001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PME Support[4:0]					D2 Support	D1 Support	AUX_Current[2:0]			Device Specific Initialization	Immediate_Readiness_on_Return_to_D0	-	Version[2:0]		
Initial Value	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Next Capability Pointer[7:0]								Capability ID[7:0]							
Initial Value	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	PME Support[4:0]	9h	R (RW)	Indicates whether PME is supported in each Device State. xxxx1b: supports D0 xxx1xb: supports D1 xx1xxb: supports D2 x1xxxb: supports D3hot 1xxxxb: D3cold supported (not supported)
26	D2 Support	0h	R (RW)	Indicates whether D2 Power Management State is supported. 0b: not supported 1b: Support
25	D1 Support	0h	R (RW)	Indicates whether the D1 Power Management State is supported. 0b: not supported 1b: Support
24 to 22	AUX_Current [2:0]	0h	R (RW)	Indicates the 3.3-V auxiliary current (the maximum current value supplied from the auxiliary power supply). 111b: 375 mA 110b: 320 mA 101b: 250 mA 100b: 220 mA 011b: 160 mA 010b: 100 mA 001b: 55 mA 000b: 0 (self powered) Reading returns 000b. (AUX not supported)
21	Device Specific Initialization	0h	R (RW)	Indicates whether or not Device Specific Initialization is used. 0b: not supported 1b: Support
20	Immediate_Readiness_on_Return_to_D0	0h	R	0b fixed
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	Version[2:0]	3h	R (RW)	011b is fixed. PCI Power Management Interface Specification Rev.1.2
15 to 8	Next Capability Pointer[7:0]	E0h	R	Indicates the MSI Capability start address.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	Capability ID[7:0]	1h	R	Indicates the PCI Power Management Capability. It is fixed at 01h.

Table 6.6-77 Valid Reset Signal

Reset Signal	PME Support	D2 Support	D1 Support	AUX_Current	Device Specific Initialization	Version
RST_LOAD_B	✓	✓	✓	✓	✓	✓
RST_RSM_B						
RST_CFG_B						
FLR						

(17) PM Status/Control (Function #n) (PCI_EP_PMSC_Fn) (n = 0, 1)

This register indicates and controls the PME status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6044h
Initial Value : 0000_0008h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PME Status	-	-	-	-	-	-	PME Enable	-	-	-	-	No_Soft_Reset	-	PowerState [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W(PCIe)	RW	R	R	R	R	R	R	RW	R	R	R	R	R	R	RW	RW
R/W(UDL)	RW	R	R	R	R	R	R	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15	PME Status	0h	RW	Indicates that a PME assert factor has occurred. 1b indicates that there is a PME assertion factor.
14 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	PME Enable	0h	RW	Controls PME assertions. If 1b, PME assertion is enabled. Assert PME if PME_Status is set at this time. PCI Express performs Link Wake-up processing, and then performs PME assert processing by sending PM_PME Message. Note: The specifications are as follows depending on the value of the PM Capabilities register PME Status[4] (PME Support in D3cold). PME Status[4]=1b: Reset: RST_RSM_B EP UDL: RW EP PCIe: RWS PME Status[4]=0b: Reset: RST_RSM_B EP UDL: RW EP PCIe: RW
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	No_Soft_Reset	1h	R	Indicates that internal reset is not performed inside the device at the power state transition from D3hot to D0.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	PowerState [1:0]	0h	RW	Set the PCI Device State. 00b - D0 (Default) 01b - D1 (Do not set) 10b - D2 (Do not set) 11b - D3hot

Table 6.6-78 Valid Reset Signal

Reset Signal	PME Enable	PowerState
RST_LOAD_B		
RST_RSM_B	✓	
RST_CFG_B		✓
FLR	✓	✓

(18) PCI Express Capability (Function #n) (PCI_EP_PCIEC_Fn) (n = 0, 1)

This register indicates the PCIe capability.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6060h
Initial Value : 0002_0010h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	Interrupt Message Number[4:0]					-	Device Port Type[3:0]				Capability Version[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Next Capability Pointer[7:0]								Capability ID[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29 to 25	Interrupt Message Number[4:0]	0h	R (RW)	Indicates the MSI vector used in interrupt messages associated with any status bit in this Capability Structure
24	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 20	Device Port Type[3:0]	0h	R (RW)	Indicates that it is a PCI Express Endpoint device. 0000b PCI Express Endpoint device (Default) 0001b Legacy PCI Express Endpoint device 0100b Root Port of PCI Express Root Complex 0101b Upstream Port of PCI Express Switch 0110b Downstream Port of PCI Express Switch 0111b PCI Express-to-PCI/PCI-X Bridge 1000b PCI/PCI-X-to-PCI Express Bridge 1001b Root Complex Integrated Endpoint Device 1010b Root Complex Event Collector All other encodings are reserved.
19 to 16	Capability Version[3:0]	2h	R (RW)	Indicates the version of PCI Express Capability Structure. 0010b fixed.
15 to 8	Next Capability Pointer[7:0]	0h	R	Indicates that this Capability List is the final List. 00h fixed.
7 to 0	Capability ID[7:0]	10h	R	Indicates PCI Express Capability. 10h fixed.

Table 6.6-79 Valid Reset Signal

Reset Signal	Interrupt Message Number	Device/Port Type	Capability Version
RST_LOAD_B	✓	✓	✓
RST_RSM_B			
RST_CFG_B			
FLR			

(19) Device Capabilities (Function #n) (PCI_EP_DEVC_Fn) (n = 0, 1)

This register indicates the device capability.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6064h
Initial Value : 1000_8FC1h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	Function Level Reset Capability	Captured Slot Power Limit Scale[1:0]		Captured Slot Power Limit Value[7:0]								-	-
Initial Value	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Role-Based Error Reporting	-	-	-	Endpoint L1 Acceptable Latency[2:0]			Endpoint L0s Acceptable Latency[2:0]			Extended Tag Field Supported	-	-	Max_Payload_Size Supported[2:0]		
Initial Value	1	0	0	0	1	1	1	1	1	1	0	0	0	0	0	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	Function Level Reset Capability	1h	R (RW)	Set when the Function Level Reset function is implemented.
27, 26	Captured Slot Power Limit Scale[1:0]	0h	R (RW)	Indicates the Scale of the Captured Slot Power Limit Value. Set by the received Set_Slot_Power_Limit Message.
25 to 18	Captured Slot Power Limit Value[7:0]	0h	R (RW)	Indicates the Slot Power Limit (Watt) value. Set by the received Set_Slot_Power_Limit Message.
17, 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15	Role-Based Error Reporting	1h	R	Set to 1b if the Error Reporting ECN feature is implemented. Fixed to 1b for PCI Express Base Spec 1.1 and later.
14 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 9	Endpoint L1 Acceptable Latency[2:0]	7h	R (RW)	000b: max 1 us 001b: max 2 us 010b: max 4 us 011b: max 8 us 100b: max 16 us 101b: max 32 us 110b: max 64 us 111b: No limit (Default)
8 to 6	Endpoint L0s Acceptable Latency[2:0]	7h	R (RW)	000b: max 64 ns 001b: max 128 ns 010b: max 256 ns 011b: max 512 ns 100b: max 1 us 101b: max 2 us 110b: max 4 us 111b: No limit (Default)

Bit	Bit Name	Initial Value	R/W	Description
5	Extended Tag Field Supported	0h	R (RW)	Extended Tag support 0b: 5-bit Tag field supported (Default) 1b: 8-bit Tag field supported (not supported)
4, 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	Max_Payload_ Size Supported[2:0]	1h	R (RW)	000b: 128B max payload size 001b: 256B max payload size (Default) 010b: 512B max payload size 011b: 1024B max payload size 100b: 2048B max payload size 101b: 4096B max payload size 110b: Reserved 111b: Reserved

Table 6.6-80 Valid Reset Signal

Reset Signal	Function Level Reset Capability	Captured Slot Power Limit Scale	Captured Slot Power Limit Value	Endpoint L1 Acceptable Latency	Endpoint L0s Acceptable Latency	Extended Tag Field Supported
RST_LOAD_B	✓			✓	✓	✓
RST_RSM_B						
RST_CFG_B		✓	✓			

Reset Signal	Max_Payload_Size Supported
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	

(20) Device Control/Status (Function #n) (PCI_EP_DEVCS_Fn) (n = 0, 1)

This register controls the device and indicates the device status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6068h
Initial Value : 0000_2010h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	Transaction Pending	-	Unsupported Request Detected	Fatal Error Detected	Non-Fatal Error Detected	Correctable Error Detected
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	RW1	RW1	RW1	RW1
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Initiate Function Level Reset	Max_Read_Request_Size[2:0]			-	-	-	-	Max_Payload_Size[2:0]			Enable Relaxed Ordering	Unsupported Request Reporting Enable	Fatal Error Reporting Enable	Non-Fatal Error Reporting Enable	Correctable Error Reporting Enable
Initial Value	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	R	RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	R	RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21	Transaction Pending	0h	R	Indicates that the transaction is Pending because Completion has not been received for the sent Non-posted Request. 1b indicates Pending.
20	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	Unsupported Request Detected	0h	RW1 (RW)	Indicates that an Unsupported Request Error was detected. 1b indicates the detection of an error.
18	Fatal Error Detected	0h	RW1 (RW)	Indicates that a Fatal Error was detected. 1b indicates the detection of an error.
17	Non-Fatal Error Detected	0h	RW1 (RW)	Indicates that a Non-Fatal Error was detected. 1b indicates the detection of an error.
16	Correctable Error Detected	0h	RW1 (RW)	Indicates that a correctable error was detected. 1b indicates the detection of an error.
15	Initiate Function Level Reset	0h	RW	A write of 1b initiates Function Level Reset to the Function. The value read by software from this bit is always 0b.
14 to 12	Max_Read_Request_Size[2:0]	2h	RW	Set Max_Read_request_Size. 000b: 128B max read request size 001b: 256B max read request size 010b: 512B max read request size (Default) 011b: 1024B max read request size 100b: 2048B max read request size 101b: 4096B max read request size 110b: Reserved 111b: Reserved
11	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
7 to 5	Max_Payload_ Size[2:0]	0h	RW	Set Max_Payload_Size. 000b: 128B max payload size (Default) 001b: 256B max payload size 010b: 512B max payload size 011b: 1024B max payload size 100b: 2048B max payload size 101b: 4096B max payload size 110b: Reserved 111b: Reserved:
4	Enable Relaxed Ordering	1h	RW	Set whether or not to use Relaxed Ordering as a Requester. 1b: Support (Default) 0b: not supported
3	Unsupported Request Reporting Enable	0h	RW	ERR_NONFATAL due to Unsupported Request detection Controls the generation of ERR_FATAL Message. 1b: enables Message generation.
2	Fatal Error Reporting Enable	0h	RW	Controls the generation of ERR_FATAL Messages. 1b: enables Message generation.
1	Non-Fatal Error Reporting Enable	0h	RW	Controls generation of ERR_NONFATAL Message. 1b: enables Message generation.
0	Correctable Error Reporting Enable	0h	RW	Controls the generation of ERR_COR Messages. 1b: enables Message generation.

Table 6.6-81 Valid Reset Signal

Reset Signal	Unsupported Request Detected	Fatal Error Detected	Non-Fatal Error Detected	Correctable Error Detected	Initiate Function Level Reset	Max_Read_Request_Size
RST_LOAD_B						
RST_RSM_B						
RST_CFG_B	✓	✓	✓	✓	✓	✓
FLR	✓	✓	✓	✓		✓

Reset Signal	Max_Payload_Size	Enable Relaxed Ordering	Unsupported Request Reporting Enable	Fatal Error Reporting Enable	Non-Fatal Error Reporting Enable	Correctable Error Reporting Enable
RST_LOAD_B						
RST_RSM_B						
RST_CFG_B	✓	✓	✓	✓	✓	✓
FLR		✓	✓	✓	✓	✓

(21) Link Capabilities (Function #n) (PCI_EP_LINKC_Fn) (n = 0, 1)

This register indicates the link capabilities.

Access Size : 32 bits
Offset Address : <PCI0_base> + 606Ch
Initial Value : 0042_CC23h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Port Number[7:0]								-	ASPM Optionality Compliance	-	-	-	Clock Power Management	L1 Exit Latency[2:1]	
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	R	R	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	L1 Exit Latency	L0s Exit Latency[2:0]			Active State Power Management ASPM Support[1:0]		Maximum Link Width[5:0]						Max Link Speed[3:0]			
Initial Value	1	1	0	0	1	1	0	0	0	0	1	0	0	0	1	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Port Number[7:0]	0h	R (RW)	Indicates the port number of PCI Express Link.
23	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	ASPM Optionality Compliance	1h	R (RW)	1b fixed
21 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18	Clock Power Management	0h	R (RW)	Indicates whether the CLKREQ# mechanism is supported in L1 and L2/L3 Ready. 0b: CLKREQ# mechanism not supported. 1b: Supports CLKREQ# mechanism. (Setting prohibited)
17 to 15	L1 Exit Latency[2:0]	5h	R (RW)	000b: Less than 1µs 001b: 1 µs to less than 2 µs 010b: 2 µs to less than 4 µs 011b: 4 µs to less than 8 µs 100b: 8 µs to less than 16 µs 101b: 16 µs to less than 32 µs (Default) 110b: 32 µs to 64 µs 111b: More than 64 µs
14 to 12	L0s Exit Latency[2:0]	4h	R (RW)	000b: Less than 64 ns 001b: 64 ns to less than 128 ns 010b: 128 ns to less than 256 ns 011b: 256 ns to less than 512 ns 100b: 512 ns to less than 1 µs (Default) 101b: 1 µs to less than 2 µs 110b: 2 µs-4 µs 111b: More than 4 µs
11, 10	Active State Power Management ASPM Support[1:0]	3h	R (RW)	00b: Reserved 01b: L0s Entry Supported 10b: Reserved 11b: L0s and L1 Entry Supported (Default)

Bit	Bit Name	Initial Value	R/W	Description
9 to 4	Maximum Link Width[5:0]	2h	R (RW)	000000b: Reserved 000001b: x1 (prohibited) 000010b: x2 (Default) 000100b: x4 (prohibited) 001000b: x8 (prohibited) 001100b: x12 (prohibited) 010000b: x16 (prohibited) 100000b: x32 (prohibited) Ch1 is set to 000010b (x2) in multilink configuration
3 to 0	Max Link Speed[3:0]	3h	R (RW)	0001b: 2.5 GT/s Link speed supported 0010b: 5.0 GT/s and 2.5 GT/s Link speeds supported 0011b: 8.0 GT/s Link speed supported (Default) 0100b: 16.0 GT/s Link speed supported (prohibited) All other encodings are reserved.

Note: x = Undefined

Table 6.6-82 Valid Reset Signal

Reset Signal	ASPM Optionality Compliance	Clock Power Management	L1 Exit Latency	L0s Exit Latency	Active State Power Management Support	Maximum Link Width
RST_LOAD_B	✓	✓	✓	✓	✓	✓
RST_RSM_B						
RST_CFG_B						
FLR						

Reset Signal	Max Link Speed
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(22) Link Control/Status (Function #n) (PCI_EP_LINKCS_Fn) (n = 0, 1)

This register controls the link and indicates the link status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6070h
Initial Value : 1000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	Data Link Layer Link Active	Slot Clock Configuration	-	-	Negotiated Link Width[5:0]						Current Link Speed[1:0]		-	-
Initial Value	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	RW1	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	RW1	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Hardware Autonomous Width Disable	Enable Clock Power Management	Extended Synch	Common Clock Configuration	-	-	Read Completion Boundary (RCB)	-	Active State Power Management (ASPM) Control[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	RW	RW	RW	RW	R	R	RW	R	RW	RW
R/W(UDL)	R	R	R	R	R	R	RW	RW	RW	RW	R	R	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29	Data Link Layer Link Active	0h	RW1	1b indicates that the Data Link Layer is in the Link Active state.
28	Slot Clock Configuration	1h	R (RW)	Indicates whether or not the reference clock supplied to the Connector is used when the Add-in Card is used. 0b: Not using Connector Reference Clock 1b: Use Connector Reference Clock (Default)
27, 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 20	Negotiated Link Width[5:0]	0h	R	Indicates the Link width established as a result of negotiation. 000001b: x1 000010b: x2 000100b: x4 001000b: x8 001100b: x12 010000b: x16 100000b: x32 Other encodings are Reserved.
19, 18	Current Link Speed[1:0]	0h	R	0001b: 2.5GT/s PCI Express Link 0010b: 5.0 GT/s PCI Express Link 0011b: 8.0 GT/s PCI Express Link 0100b: 16.0 GT/s PCI Express Link 0000b during reset period
17 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	Hardware Autonomous Width Disable	0h	RW	Controls Link width change. 0b: Enable of Link Width Change 1b: Disable Note: Reserved in Function #1

Bit	Bit Name	Initial Value	R/W	Description
8	Enable Clock Power Management	0h	RW	Controls Enable/Disable of CLKREQ# mechanism in L1 and L2/L3 Ready.
7	Extended Synch	0h	RW	If set to 1b, 4096 FTS Ordered-sets is sent when transitioning from L0s to L0. Also, 1024 TS1 Ordered-sets are transmitted at the beginning of the Recovery state when transitioning from L1 to L0. The default value will be 0b.
6	Common Clock Configuration	0h	RW	Sets whether the Common Reference Clock is used. 0b: Provided by non-Common Reference Clock (Default) 1b: Supplied by Common Reference Clock
5, 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	Read Completion Boundary (RCB)	0h	RW	0b: 64 bytes (Default) 1b: 128 bytes
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	Active State Power Management (ASPM) Control[1:0]	0h	RW	Sets the permission level for Active State Power Management. 00b: Disabled (Default) 01b: L0s Entry Supported 10b: L1 Entry Enabled 11b: L0s and L1 Entry Supported

Table 6.6-83 Valid Reset Signal

Reset Signal	Slot Clock Configuration	Hardware Autonomous Width Disable	Enable Clock Power Management	Extended Synch	Common Clock Configuration	Read Completion Boundary
RST_LOAD_B	✓					
RST_RSM_B						
RST_CFG_B		✓	✓	✓	✓	✓
FLR						

Reset Signal	Active State Power Management Control
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓
FLR	

(23) Slot Capabilities (Function #n) (PCI_EP_SLOTC_Fn) (n = 0, 1)

This function is not supported.

(24) Slot Control/Status (Function #n) (PCI_EP_SLOTCS_Fn) (n = 0, 1)

This function is not supported.

(25) Root Control/Capabilities (Function #n) (PCI_EP_ROOTCC_Fn) (n = 0, 1)

This function is not supported.

(26) Root Status (Function #n) (PCI_EP_ROOTS_Fn) (n = 0, 1)

This function is not supported.

(27) Device Capabilities 2 (Function #n) (PCI_EP_DEVC2_Fn) (n = 0, 1)

This register indicates the device capability.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6084h
Initial Value : 0000_0012h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	Completion Timeout Disable Supported	Completion Timeout Ranges Supported[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	RW	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 16	-	All 0	R (RW)	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11	-	All 0	R (RW)	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
10 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	Completion Timeout Disable Supported	1h	R (RW)	Set whether to support the Completion Timeout Disable function. 0b: Not supported 1b: Supported (default)
3 to 0	Completion Timeout Ranges Supported[3:0]	2h	R (RW)	Set Completion Timeout Range Range A: 50 µs to 10 ms Range B: 10 ms to 250 ms Range C: 250 ms to 4 s Range D: 4 s to 64 s The above four patterns are determined, and the following combinations can be set. 0000b: Program setting Not supported 0001b: Range A 0010b: Range B 0011b: Ranges A and B 0110b: Ranges B and C 0111b: Ranges A, B, and C 1110b: Ranges B, C, and D 1111b: Ranges A, B, C, and D Other encodings are reserved. Note: Although the initial value described in the UM is "0010b", change the initial value according to the installed system/device performance, or specify the initial value as a product requirement.

Table 6.6-84 Valid Reset Signal

Reset Signal	Completion Timeout Disable Supported	Completion Timeout Ranges Supported
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		
FLR		

(28) Device Control 2/Status 2 (Function #n) (PCI_EP_DEVCS2_Fn) (n = 0, 1)

This register controls device and indicates the device status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6088h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	Completion Timeout Disable	Completion Timeout Value[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	RW	RW	RW	R	RW	R	R	R	R	R	RW	RW	RW	RW	RW
R/W(UDL)	R	RW	RW	RW	R	RW	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 12	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
11	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
9 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	Completion Timeout Disable	0h	RW	Setting this bit enables the Completion Timeout Disable function.
3 to 0	Completion Timeout Value[3:0]	0h	RW	Set Completion Timeout Range. 0000b: 10 ms to 50 ms (default) 0001b: 50 µs to 100 µs 0010b: 1 ms to 10 ms 0101b: 16 ms to 55 ms 0110b: 65 ms to 210 ms 1001b: 260 ms to 900 ms 1010b: 1 s to 3.5 s 1101b: 4 s to 13 s 1110b: 17 s to 64 s Others: reserved (setting prohibited) Note: The default 0000b sets a longer time than the Base Spec's default 50usec lower limit, but this takes into account the Base Spec's recommended 10msec lower limit.

Table 6.6-85 Valid Reset Signal

Reset Signal	Completion Timeout Disable	Completion Timeout Value
RST_LOAD_B		
RST_RSM_B		
RST_CFG_B	✓	✓
FLR	✓	✓

(29) Link Capabilities 2 (Function #n) (PCI_EP_LINKC2_Fn) (n = 0, 1)

This register indicates the link capabilities.

Access Size : 32 bits
Offset Address : <PCI0_base> + 608Ch
Initial Value : 0000_000Eh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	Supported Link Speeds Vector[6:0]							-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, undefined value is read. The written value will be ignored.
24 to 9	-	All 0	R (RW)	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
8	-	All 0	R	Reserved Whenever it is read, undefined value is read. The written value will be ignored.
7 to 1	Supported Link Speeds Vector[6:0]	7h	R (RW)	The read value is undefined. Indicates the supported link speed. Bit[0]: 2.5 GT/s Bit[1]: 5.0 GT/s Bit[2]: 8.0 GT/s Bit[3]: 16.0 GT/s (setting prohibited) Bits[6:4]: Reserved
0	-	All 0	R	Reserved Whenever it is read, undefined value is read. The written value will be ignored.

Table 6.6-86 Valid Reset Signal

Reset Signal	Supported Link Speeds Vector
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(30) Link Control 2/Status 2 (Function #n) (PCI_EP_LINCS2_Fn) (n = 0, 1)

This register controls the link and indicates the link status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6090h
Initial Value : 0000_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	Link Equalization Request	Equalization Phase 3 Successful	Equalization Phase 2 Successful	Equalization Phase 1 Successful	Equalization Complete	Current De-emphasis Level
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	RW1	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	RW1	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Compliance Preset De-emphasis[3:0]				Compliance SOS	Enter Modified Compliance	Transmit Margin[2:0]			-	Hardware Autonomous Speed Disable	Enter Compliance	Target Link Speed[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21	Link Equalization Request	0h	RW1	This bit is Set by hardware to request the Link equalization process to be performed on the Link. The default value of this bit is 0b. For Multi-Function Upstream Port, this bit must be implemented in Function 0 and RsvdZ in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.
20	Equalization Phase 3 Successful	0h	R	When set to 1b, this bit indicates that Phase 3 of the Transmitter Equalization procedure has successfully completed. The default value of this bit is 0b. For Multi-Function Upstream Port, this bit must be implemented in Function 0 and RsvdZ in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.
19	Equalization Phase 2 Successful	0h	R	When set to 1b, this bit indicates that Phase 2 of the Transmitter Equalization procedure has successfully completed. The default value of this bit is 0b. For Multi-Function Upstream Port, this bit must be implemented in Function 0 and RsvdZ in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.
18	Equalization Phase 1 Successful	0h	R	When set to 1b, this bit indicates that Phase 1 of the Transmitter Equalization procedure has successfully completed. The default value of this bit is 0b. For Multi-Function Upstream Port, this bit must be implemented in Function 0 and RsvdZ in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.
17	Equalization Complete	0h	R	When set to 1b, this bit indicates that the Transmitter Equalization procedure has completed. The default value of this bit is 0b. For Multi-Function Upstream Port, this bit must be implemented in Function 0 and RsvdZ in other Functions. Components that only support speeds below 8.0 GT/s are permitted to hardwire this bit to 0b.

Bit	Bit Name	Initial Value	R/W	Description
16	Current De-emphasis Level	0h	R	This is a Status register that indicates the De-emphasis Level during Gen3 operation. 1b: -3.5dB 0b: -6 dB (Default) Note: The initial value of this bit is 0b at reset and after Gen3-Linkup. However, although it indicates 1b at Gen1-Linkup, ignore this bit as it has no meaning in Gen1.
15 to 12	Compliance Preset De-emphasis[3:0]	0h	RW	Set Transmitter Preset(Gen3) and De-emphasis Level(Gen2) for Polling.Compliance State processing during Enter Compliance. (Gen3) 0000b: P0 (Default) 0001b: P1 0010b: P2 0011b: P3 0100b: P4 0101b: P5 0110b: P6 0111b: P7 1000b: P8 1001b: P9 1010b: P10 Others: Rsvd (Gen2) 0001b: -3.5 dB 0000b: -6 dB (Default) Note: Reserved in Function #1
11	Compliance SOS	0h	RW	If this bit is set to 1b, insert the SKP Ordered-set periodically during compliance pattern transmission. Note: Reserved in Function #1
10	Enter Modified Compliance	0h	RW	Setting bit for transmission of Modified Compliance pattern. 1b: Modified Compliance Pattern 0b: Compliance Pattern (Default) Note: Reserved in Function #1
9 to 7	Transmit Margin[2:0]	0h	RW	Adjust the voltage level of the Transmitter. 000b: Normal operating range 001b-111b: See Base Spec *Section 8.3.4 in PCIe standard*. Note: Reserved in Function #1
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	Hardware Autonomous Speed Disable	0h	RW	Controls the Link Speed Change function. 1b: Non-support of Link Speed Change (Disable) 0b: Support of Link Speed Change (Enable) Note: Reserved in Function #1
4	Enter Compliance	0h	RW	Setting to 1b allows transition to Compliance mode. At this time, the Link Speed will be the value set in the Target Link Speed field. Note: Reserved in Function #1
3 to 0	Target Link Speed[3:0]	3h	RW	Sets the Link Speed value for notification to the Root Complex during training. 0001b: 2.5 GT/s Target Link Speed 0010b: 5.0 GT/s Target Link Speed 0011b: 8.0 GT/s Target Link Speed (Default) 0100b: 16.0 GT/s Target Link Speed (Setting prohibited) All other encodings are reserved. Note: Reserved in Function #1

Table 6.6-87 Valid Reset Signal

Reset Signal	Link Equalization Request	Compliance Preset/De-emphasis	Compliance SOS	Enter Modified Compliance	Transmit Margin	Hardware Autonomous Speed Disable
RST_LOAD_B						
RST_RSM_B	✓	✓	✓	✓	✓	✓
RST_CFG_B						
FLR						

Reset Signal	Enter Compliance	Target Link Speed
RST_LOAD_B		
RST_RSM_B	✓	✓
RST_CFG_B		
FLR		

(31) Slot Capabilities 2 (Function #n) (PCI_EP_SLOTC2_Fn) (n = 0, 1)

This function is not supported.

(32) Slot Control 2 / Status 2 (Function #n) (PCI_EP_SLOTCS2_Fn) (n = 0, 1)

This function is not supported.

(33) Base Address Register Mask00 (Lower) (Function #n) (PCI_EP_BARMSK00L_Fn) (n = 0, 1)

This register indicates mask information for Base Address Register 0 (BAR0).

Access Size : 32 bits
Offset Address : <PCI0_base> + 60A0h
Initial Value : 1FFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BARM00L[31:16]															
Initial Value	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BARM00L[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BARM00L[31:0]	1FFF_FFFFh	R (RW)	Mask register for Base Address Register 0 (BAR0).

Table 6.6-88 Valid Reset Signal

Reset Signal	BARM 00L
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(34) Base Address Register Mask00 (Higher) (Function #n) (PCI_EP_BARMSK00U_Fn) (n = 0, 1)

This register indicates mask information for Base Address Register 1 (BAR1).

Access Size : 32 bits
Offset Address : <PCI0_base> + 60A4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BARM00U[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BARM00U[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BARM00U [31:0]	0h	R (RW)	Mask register for Base Address Register 1 (BAR1).

Table 6.6-89 Valid Reset Signal

Reset Signal	BARM 00U
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(35) Base Address Register Mask01 (Lower) (Function #n) (PCI_EP_BARMSK01L_Fn) (n = 0, 1)

Access Size : 32 bits
Offset Address : <PCI0_base> + 60A8h
Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BARM01L[31:16]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BARM01L[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BARM01L[31:0]	FFFF_FFFFh	R (RW)	Mask register for Base Address Register 2 (BAR2).

Table 6.6-90 Valid Reset Signal

Reset Signal	BARM 01L
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(36) Base Address Register Mask01 (Higher) (Function #n) (PCI_EP_BARMSK01U_Fn) (n = 0, 1)

Access Size : 32 bits
Offset Address : <PCI0_base> + 60ACh
Initial Value : F0: 0000_007Fh
F1: 0000_00FFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BARM01U[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BARM01U[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0/1	1	1	1	1	1	1	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BARM01U [31:0]	7Fh FFh	R (RW)	Mask register for Base Address Register 3 (BAR3). 0000007Fh: (Function0) 000000FFh: (Function1)

Table 6.6-91 Valid Reset Signal

Reset Signal	BARM 01U
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(37) Base Address Register Mask02 (Lower) (Function #n) (PCI_EP_BARMSK02L_Fn) (n = 0, 1)

Access Size : 32 bits
Offset Address : <PCI0_base> + 60B0h
Initial Value : 0000_1FFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BARM02L[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BARM02L[15:0]															
Initial Value	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BARM02L[31:0]	1FFFh	R (RW)	Mask register for Base Address Register 4 (BAR4).

Table 6.6-92 Valid Reset Signal

Reset Signal	BARM 02L
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(38) Base Address Register Mask02 (Higher) (Function #n) (PCI_EP_BARMSK02U_Fn) (n = 0, 1)

Access Size : 32 bits
Offset Address : <PCI0_base> + 60B4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BARM02U[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BARM02U[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BARM02U [31:0]	0h	R (RW)	Mask register for Base Address Register 5 (BAR5).

Table 6.6-93 Valid Reset Signal

Reset Signal	BARM 02U
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(39) Base Size 00/01 (Function #n) (PCI_EP_BSIZE00_01_Fn) (n = 0, 1)

This register sets the acceptable TLP size.

Access Size : 32 bits
Offset Address : <PCI0_base> + 60C8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	Base Size 01[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Base Size 00[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 16	Base Size 01[9:0]	0h	R (RW)	Sets the size of the TLP (DW Size) that will be accepted in Address Space 01 set in the Base Address Register and Base Address Mask Register. If a TLP with a packet length larger than the size set here is received (even if it is smaller than the Max Payload Size), a Completer Abort Error (CA) will be detected. Note that the default value is 000h, which disables this function.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	Base Size 00[9:0]	0h	R (RW)	Sets the size of the TLP (DW Size) that will be accepted in Address Space 00 set in the Base Address Register and Base Address Mask Register. If a TLP with a packet length larger than the size set here is received (even if it is smaller than the Max Payload Size), a Completer Abort Error (CA) will be detected. Note that the default value is 000h, which disables this function.

Table 6.6-94 Valid Reset Signal

Reset Signal	Base Size 01	Base Size 00
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		
FLR		

(40) Base Size 02/03 (Function #n) (PCI_EP_BSIZE02_03_Fn) (n = 0, 1)

This register sets the acceptable TLP size.

Access Size : 32 bits
Offset Address : <PCI0_base> + 60CCh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	Base Size 03[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Base Size 02[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 16	Base Size 03[9:0]	0h	R (RW)	Sets the size of the TLP (DW Size) that will be accepted in Address Space 03 set in the Base Address Register and Base Address Mask Register. If a TLP with a packet length larger than the size set here is received (even if it is smaller than the Max Payload Size), a Completer Abort Error (CA) will be detected. Note that the default value is 000h, which disables this function.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	Base Size 02[9:0]	0h	R (RW)	Sets the size of the TLP (DW Size) that will be accepted in Address Space 02 set in the Base Address Register and Base Address Mask Register. If a TLP with a packet length larger than the size set here is received (even if it is smaller than the Max Payload Size), a Completer Abort Error (CA) will be detected. Note that the default value is 000h, which disables this function.

Table 6.6-95 Valid Reset Signal

Reset Signal	Base Size 03	Base Size 02
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		
FLR		

(41) Base Size 04/05 (Function #n) (PCI_EP_BSIZE04_05_Fn) (n = 0, 1)

This register sets the acceptable TLP size.

Access Size : 32 bits
Offset Address : <PCI0_base> + 60D0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	Base Size 05[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Base Size 04[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 16	Base Size 05[9:0]	0h	R (RW)	Sets the size of the TLP (DW Size) that will be accepted in Address Space 05 set in the Base Address Register and Base Address Mask Register. If a TLP with a packet length larger than the size set here is received (even if it is smaller than the Max Payload Size), a Completer Abort Error (CA) will be detected. Note that the default value is 000h, which disables this function.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	Base Size 04[9:0]	0h	R (RW)	Sets the size of the TLP (DW Size) that will be accepted in Address Space 04 set in the Base Address Register and Base Address Mask Register. If a TLP with a packet length larger than the size set here is received (even if it is smaller than the Max Payload Size), a Completer Abort Error (CA) will be detected. Note that the default value is 000h, which disables this function.

Table 6.6-96 Valid Reset Signal

Reset Signal	Base Size 05	Base Size 04
RST_LOAD_B	✓	✓
RST_RSM_B		
RST_CFG_B		
FLR		

(42) Base Size 06 (Function #n) (PCI_EP_BSIZE06_Fn) (n = 0, 1)

This register sets the acceptable TLP size.

Access Size : 32 bits
Offset Address : <PCI0_base> + 60D4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	Base Size 06[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	Base Size 06[9:0]	0h	R (RW)	Sets the size of the TLP (DW Size) that will be accepted in Address Space 06 set in the Base Address Register and Base Address Mask Register. If a TLP with a packet length larger than the size set here is received (even if it is smaller than the Max Payload Size), a Completer Abort Error (CA) will be detected. Note that the default value is 000h, which disables this function.

Table 6.6-97 Valid Reset Signal

Reset Signal	Base Size 06
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(43) Type Supported 00/01/02 (Function #n) (PCI_EP_TSUPPORT00_01_02_Fn) (n = 0, 1)

This register indicates the transaction type which can be supported by the memory space.

Access Size : 32 bits
Offset Address : <PCI0_base> + 60D8h
Initial Value : 0033_3333h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	Type Supported 02[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Type Supported 01[7:0]								Type Supported 00[7:0]							
Initial Value	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R (RW)	Reserved Whenever it is read, 0b is read. The write value should always be 0b.
23 to 16	Type Supported 02[7:0]	33h	R (RW)	Set the transaction type that can be supported by Space00 (CFG_SPACE02_BASE). Bit[0]: 32-bit memory read Bit[1]: 64-bit memory read Bit[2]: 32-bit memory read lock Bit[3]: 64-bit memory read lock Bit[4]: 32-bit memory write Bit[5]: 64-bit memory write Bit[6]: IO read Bit[7]: IO write
15 to 8	Type Supported 01[7:0]	33h	R (RW)	Set the transaction type that can be supported by Space00 (CFG_SPACE01_BASE). Bit[0]: 32-bit memory read Bit[1]: 64-bit memory read Bit[2]: 32-bit memory read lock Bit[3]: 64-bit memory read lock Bit[4]: 32-bit memory write Bit[5]: 64-bit memory write Bit[6]: IO read Bit[7]: IO write
7 to 0	Type Supported 00[7:0]	33h	R (RW)	Set the transaction type that can be supported by Space00 (CFG_SPACE00_BASE). Bit[0]: 32-bit memory read Bit[1]: 64-bit memory read Bit[2]: 32-bit memory read lock Bit[3]: 64-bit memory read lock Bit[4]: 32-bit memory write Bit[5]: 64-bit memory write Bit[6]: IO read Bit[7]: IO write

Table 6.6-98 Valid Reset Signal

Reset Signal	Type Supported 02	Type Supported 01	Type Supported 00
RST_LOAD_B	✓	✓	✓
RST_RSM_B			
RST_CFG_B			
FLR			

(44) MSI Capability (Function #n) (PCI_EP_MSICAP_Fn) (n = 0, 1)

This register specifies the MSI Capability.

Access Size : 32 bits
Offset Address : <PCI0_base> + 60E0h
Initial Value : 018A_6005h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	Per-vector masking capable	64bit Address Capable	Multiple Message Enable[2:0]			Multiple Message Capable[2:0]			MSI Enable
Initial Value	0	0	0	0	0	0	0	1	1	0	0	0	1	0	1	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	RW	RW	RW	R	R	R	RW
R/W(UDL)	R	R	R	R	R	R	R	R	R	RW	RW	RW	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Next Capability Pointer[7:0]								Capability ID[7:0]							
Initial Value	0	1	1	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	Per-vector masking capable	1h	R	Indicates support for MSI per-vector masking.
23	64bit Address Capable	1h	R	Indicates that a 64-bit Address MSI Message can be generated. 1b fixed.
22 to 20	Multiple Message Enable[2:0]	0h	RW	Set the generation permission and number of Multiple Messages. Since 101b is specified in the Multiple Message Capable field, up to 101b (32) can be set as software settings. 000b: 1 (Default) 001b: 2 010b: 4 011b: 8 100b: 16 101b: 32 110b: Reserved 111b: Reserved
19 to 17	Multiple Message Capable[2:0]	5h	R	000b: 1 001b: 2 010b: 4 011b: 8 100b: 16 101b: 32(Default) 110b: Reserved 111b: Reserved
16	MSI Enable	0h	RW	Controls whether or not to generate MSI Messages. 1b enables MSI generation.
15 to 8	Next Capability Pointer[7:0]	60h	R	Indicates the starting address of the PCI Express Capability. Default: 60h
7 to 0	Capability ID[7:0]	5h	R	Indicates MSI Capability. 05h fixed.

Table 6.6-99 Valid Reset Signal

Reset Signal	Multiple Message Enable	MSI Enable
RST_LOAD_B		
RST_RSM_B		
RST_CFG_B	✓	✓
FLR	✓	✓

(45) Message Address (Function #n) (PCI_EP_MSGADR_Fn) (n = 0, 1)

This register sets the destination address of MSI messages.

Access Size : 32 bits
Offset Address : <PCI0_base> + 60E4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Message Address[29:14]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Address[13:0]														-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	Message Address[29:0]	0h	RW	Set the MSI Message Destination Address [31:2].
1, 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Table 6.6-100 Valid Reset Signal

Reset Signal	Message Address
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓
FLR	✓

(46) Message Higher Address (Function #n) (PCI_EP_MSGUADR_Fn) (n = 0, 1)

This register sets the higher destination address of MSI messages.

Access Size : 32 bits
Offset Address : <PCI0_base> + 60E8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Message Upper Address[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Upper Address[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Message Upper Address[31:0]	0h	RW	Set the MSI Message Destination Address [63:32].

Table 6.6-101 Valid Reset Signal

Reset Signal	Message Upper Address
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓
FLR	✓

(47) Message Data (Function #n) (PCI_EP_MSGDAT_Fn) (n = 0, 1)

Access Size : 32 bits
Offset Address : <PCI0_base> + 60ECh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Message Data[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	Message Data[15:0]	0h	RW	Set Data[15:0] to be set in the MSI Message. If the Multiple Message Enable field is 011b (8), the Function (UDL) side can change the lower 3 bits of the Message data, and 8 types of vectors can be notified to the system as Message_Data. Similarly, in the case of 010b (4), the lower 2 bits can be changed by the function (UDL) side, and 4 types of vector can be changed. In the case of 001b (2), the lower 1 bit can be changed by the function (UDL) side, and 2 types of vector are set. In the case of 000b (1), 1 type of vector is set for the set data only. Note: Specify 0b for invalid high-order bits (in the above example of Multiple Message Enable setting 4, high-order bits other than the low-order 2 bits to be changed on the Function side) for the vector specification value on each Function side.

Table 6.6-102 Valid Reset Signal

Reset Signal	Message Data
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓
FLR	✓

(48) Mask Bits (Function #n) (PCI_EP_MSKBIT_Fn) (n = 0, 1)

This register masks the transmission of messages as vectors.

Access Size : 32 bits
Offset Address : <PCI0_base> + 60F0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Mask Bits[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Mask Bits[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Mask Bits[31:0]	0h	RW	Bit set to “1b” to mask message transmission of vector

Table 6.6-103 Valid Reset Signal

Reset Signal	Mask Bits
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓
FLR	✓

(49) Pending Bits (Function #n) (PCI_EP_PENDBIT_Fn) (n = 0, 1)

This register indicates the pending state of an MSI message for the given function.

Access Size : 32 bits
Offset Address : <PCI0_base> + 60F4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Pending Bits[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Pending Bits[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Pending Bits[31:0]	0h	R	Indicates MSI Message pending status for each Function

(50) Advanced Error Reporting Capability (Function #n) (PCI_EP_ADVERC_Fn) (n = 0, 1)

This register indicates the advanced error reporting capability.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6100h
Initial Value : 1501_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Next Capability Offset[11:0]												Capability Version[3:0]			
Initial Value	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCI Express Extended Capability ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	Next Capability Offset[11:0]	150h	R	Indicates the starting address for DeviceSerialNumberCapability.
19 to 16	Capability Version[3:0]	1h	R (RW)	Indicates the version of the Capability Structure. Default: 0001b
15 to 0	PCI Express Extended Capability ID[15:0]	1h	R	Indicates the Advanced Error Reporting Capability. Default: 0001h

Table 6.6-104 Valid Reset Signal

Reset Signal	Capability Version
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(51) Uncorrectable Error Status Register (Function #n) (PCI_EP_UNCESTS_Fn) (n = 0, 1)

This register indicates the uncorrectable error status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6104h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	Unsupported Request Error Status	ECRC Error Status Optional	Malformed TLP Status	Receiver Overflow Status Optional	Unexpected Completion Status
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	RW1	RW1	RW1	RW1	RW1
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	RW1	RW1	RW1	RW1	RW1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Completer Abort Status Optional	Completion Timeout Status	-	Poisoned TLP Received Status	-	-	-	-	-	-	-	Data Link Protocol Error Status	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW1	RW1	R	RW1	R	R	R	R	R	R	R	RW1	R	R	R	R
R/W(UDL)	RW1	RW1	R	RW1	R	R	R	R	R	R	R	RW1	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	Unsupported Request Error Status	0h	RW1	Indicates that an unsupported TLP was received. 0b: No error detected 1b: error detected
19	ECRC Error Status Optional	0h	RW1	Indicates that an ECRC Error was received. 0b: No error detected 1b: error detected
18	Malformed TLP Status	0h	RW1	Indicates that a Malformed TLP was received. 0b: No error detected 1b: error detected
17	Receiver Overflow Status Optional	0h	RW1	Indicates that a TLP with a size larger than the free credits in the receive buffer was received. 0b: No error detected 1b: error detected
16	Unexpected Completion Status	0h	RW1	Indicates that a Completion was received, but there is no record of a corresponding Non-Posted Request sent (due to a mismatch with the Transaction Descriptor). 0b: No error detected 1b: error detected
15	Completer Abort Status Optional	0h	RW1	Indicates that a Completion whose Completion Status is Completer Abort (CA) was returned after receiving a Non-Posted Request. 0b: No error detected 1b: error detected
14	Completion Timeout Status	0h	RW1	Indicates that the corresponding Completion was not received within the specified time after sending a Non-Posted Request. 0b: No error detected 1b: error detected
13	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
12	Poisoned TLP Received Status	0h	RW1	Indicates that a Poisoned TLP (with payload and the EP field in the header is 1b) has been received 0b: No error detected 1b: error detected
11 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	Data Link Protocol Error Status	0h	RW1	Indicates that a Sequence Number error was detected in the Data Link Layer. 0b: No error detected 1b: error detected
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Table 6.6-105 Valid Reset Signal

Reset Signal	Unsupported Request Error Status	ECRC Error Status	Malformed TLP Status	Receiver Overflow Status	Unexpected Completion Status	Completer Abort Status
RST_LOAD_B						
RST_RSM_B	✓	✓	✓	✓	✓	✓
RST_CFG_B						
FLR						

Reset Signal	Completion Timeout Status	Poisoned TLP Received Status	Data Link Protocol Error Status
RST_LOAD_B			
RST_RSM_B	✓	✓	✓
RST_CFG_B			
FLR			

(52) Uncorrectable Error Mask Register (Function #n) (PCI_EP_UNCEMASK_Fn) (n = 0, 1)

This register masks the uncorrectable error status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6108h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	Unsupported Request Error Mask	ECRC Error Mask Optional	Malformed TLP Mask	Receiver Overflow Mask Optional	Unexpected Completion Mask
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Completer Abort Mask Optional	Completion Timeout Mask	-	Poisoned TLP Received Mask	-	-	-	-	-	-	-	Data Link Protocol Error Mask	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	RW	RW	R	RW	R	R	R	R	R	R	R	RW	R	R	R	R
R/W(UDL)	RW	RW	R	RW	R	R	R	R	R	R	R	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	Unsupported Request Error Mask	0h	RW	Masks error notification to Root Complex when Unsupported Request Error is detected. 0b: no mask 1b: Masks error message transmission, recording of the header in the Header Log register, and updating of the first error pointer.
19	ECRC Error Mask Optional	0h	RW	Masks error notification to Root Complex when ECRC Error is detected. 0b: no mask 1b: mask
18	Malformed TLP Mask	0h	RW	Masks error notification to Root Complex when Malformed TLP Error is detected. 0b: no mask 1b: Masks error message transmission, recording of the header in the Header Log register, and updating of the first error pointer.
17	Receiver Overflow Mask Optional	0h	RW	Masks error notification to Root Complex when Receiver Overflow Error is detected. 0b: no mask 1b: Masks error message transmission and updating of the first error pointer.
16	Unexpected Completion Mask	0h	RW	Masks error notification to Root Complex when Unexpected Completion Error is detected. 0b: no mask 1b: Masks error message transmission, recording of the header in the Header Log register, and updating of the first error pointer.
15	Completer Abort Mask Optional	0h	RW	Masks the error notification to the Root Complex when a Completer Abort Error is detected. 0b: no mask 1b: Masks error message transmission, recording of the header in the Header Log register, and updating of the first error pointer.
14	Completion Timeout Mask	0h	RW	Masks error notification to Root Complex when Completion Timeout Error is detected. 0b: no mask 1b: Masks error message transmission and updating of the first error pointer.
13	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
12	Poisoned TLP Received Mask	0h	RW	Masks error notification to Root Complex when Poisoned TLP Error is detected. 0b: no mask 1b: Masks error message transmission, recording of the header in the Header Log register, and updating of the first error pointer.
11 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	Data Link Protocol Error Mask	0h	RW	Masks error notification to Root Complex when Data Link Protocol Error is detected. 0b: no mask 1b: Masks error message transmission and updating of the first error pointer.
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Table 6.6-106 Valid Reset Signal

Reset Signal	Unsupported Request Error Mask	ECRC Error Mask	Malformed TLP Mask	Receiver Overflow Mask	Unexpected Completion Mask	Completer Abort Mask
RST_LOAD_B						
RST_RSM_B	✓	✓	✓	✓	✓	✓
RST_CFG_B						
FLR						

Reset Signal	Completion Timeout Mask	Poisoned TLP Received Mask	Data Link Protocol Error Mask
RST_LOAD_B			
RST_RSM_B	✓	✓	✓
RST_CFG_B			
FLR			

(53) Uncorrectable Error Severity Register (Function #n) (PCI_EP_UNCESVY_Fn) (n = 0, 1)

This register sets the uncorrectable error severity.

Access Size : 32 bits
Offset Address : <PCI0_base> + 610Ch
Initial Value : 0046_2030h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	Unsupported Request Error Severity	ECRC Error Severity Optional	Malformed TLP Severity	Receiver Overflow Severity Optional	Unexpected Completion Severity
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW
R/W(UDL)	R	R	R	R	R	R	R	R	R	RW	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Completer Abort Severity Optional	Completion Timeout Severity	-	Poisoned TLP Received Severity	-	-	-	-	-	-	-	Data Link Protocol Error Severity	-	-	-	-
Initial Value	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0
R/W(PCIe)	RW	RW	R	RW	R	R	R	R	R	R	R	RW	R	R	R	R
R/W(UDL)	RW	RW	R	RW	R	R	R	R	R	R	R	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	-	1h	R (RW)	Reserved Whenever it is read, 1b is read. The write value should always be 1b.
21	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	Unsupported Request Error Severity	0h	RW	Set the Error Severity when Unsupported Request Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
19	ECRC Error Severity Optional	0h	RW	Sets the Error Severity when receiving an ECRC Error. 0b: Non-Fatal Error 1b: Fatal Error
18	Malformed TLP Severity	1h	RW	Sets the Error Severity when Malformed TLP Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
17	Receiver Overflow Severity Optional	1h	RW	Set the Error Severity when Receiver Overflow Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
16	Unexpected Completion Severity	0h	RW	Sets the Error Severity when an Unexpected Completion Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
15	Completer Abort Severity Optional	0h	RW	Sets the Error Severity when a Completer Abort Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
14	Completion Timeout Severity	0h	RW	Sets the Error Severity when Completion Timeout Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
13	-	1h	R	Reserved Whenever it is read, 1b is read. The written value will be ignored.
12	Poisoned TLP Received Severity	0h	RW	Sets the Error Severity when Poisoned TLP Error is detected. 0b: Non-Fatal Error 1b: Fatal Error

Bit	Bit Name	Initial Value	R/W	Description
11 to 5	-	1h	R	Reserved Whenever it is read, 1h is read. The written value will be ignored.
4	Data Link Protocol Error Severity	1h	RW	Sets the Error Severity when a Data Link Protocol Error is detected. 0b: Non-Fatal Error 1b: Fatal Error
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Table 6.6-107 Valid Reset Signal

Reset Signal	Unsupported Request Error Severity	ECRC Error Severity	Malformed TLP Severity	Receiver Overflow Severity	Unexpected Completion Severity	Completer Abort Severity
RST_LOAD_B						
RST_RSM_B	✓	✓	✓	✓	✓	✓
RST_CFG_B						
FLR						

Reset Signal	Completion Timeout Severity	Poisoned TLP Received Severity	Data Link Protocol Error Severity
RST_LOAD_B			
RST_RSM_B	✓	✓	✓
RST_CFG_B			
FLR			

(54) Correctable Error Status Register (Function #n) (PCI_EP_CESTS_Fn) (n = 0, 1)

This register indicates the correctable error status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6110h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Advisory Non-Fatal Error Status	Replay Timer Timeout Status	-	-	-	REPLAY_NUM Rollover Status	Bad DLLP Status	Bad TLP Status	-	-	-	-	-	Receiver Error Status Optional
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	RW1	RW1	R	R	R	RW1	RW1	RW1	R	R	R	R	R	RW1
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	Advisory Non-Fatal Error Status	0h	RW1 (R)	Indicates that an Advisory Non-Fatal Error was detected. 0b: No error detected 1b: error detected
12	Replay Timer Timeout Status	0h	RW1 (R)	Indicates that a Timeout error occurred when a TLP was sent and an Ack or Nak DLLP could not be received within the specified time. 0b: No error detected 1b: error detected
11 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	REPLAY_NUM Rollover Status	0h	RW1 (R)	Replay occurred four times in a row, indicating REPLAY_NUM rolled over from 11b to 00b. 0b: No error detected 1b: error detected
7	Bad DLLP Status	0h	RW1 (R)	Indicates that a DLLP CRC error was detected. 0b: No error detected 1b: error detected
6	Bad TLP Status	0h	RW1 (R)	Indicates that a TLP CRC error or Sequence Number error was detected. 0b: No error detected 1b: error detected
5 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	Receiver Error Status Optional	0h	RW1 (R)	0b: No error detected 1b: error detected

Table 6.6-108 Valid Reset Signal

Reset Signal	Advisory Non-Fatal Error Status	Replay Timer Timeout Status	REPLAY_NUM Rollover Status	Bad DLLP Status	Bad TLP Status	Receiver Error Status
RST_LOAD_B						
RST_RSM_B	✓	✓	✓	✓	✓	✓
RST_CFG_B						
FLR						

(55) Correctable Error Mask Register (Function #n) (PCI_EP_CEMASK_Fn) (n = 0, 1)

This register masks the correctable error status.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6114h
Initial Value : 0000_2000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	Advisory Non-Fatal Error Mask	Replay Timer Timeout Mask	-	-	-	REPLAY_NUM Rollover Mask	Bad DLLP Mask	Bad TLP Mask	-	-	-	-	-	Receiver Error Mask Optional
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	RW	RW	R	R	R	RW	RW	RW	R	R	R	R	R	RW
R/W(UDL)	R	R	RW	RW	R	R	R	RW	RW	RW	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	Advisory Non-Fatal Error Mask	0h	RW	Masks error notification to Root Complex when an Advisory non-fatal error is detected. 0b: no mask 1b: Mask Advisory Non-Fatal Error handling (Mask updating First Error Pointer and Header Logging and sending Error Message)
12	Replay Timer Timeout Mask	0h	RW	Masks error notification to Root Complex when Replay Timer Timeout Error is detected. 0b: no mask 1b: Mask error message transmission
11 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	REPLAY_NUM Rollover Mask	0h	RW	REPLAY_NUM Mask error notification to Root Complex when Roll Over Error is detected. 0b: no mask 1b: Mask error message transmission
7	Bad DLLP Mask	0h	RW	Masks error notification to Root Complex when Bad DLLP Error is detected. 0b: no mask 1b: Mask error message transmission
6	Bad TLP Mask	0h	RW	Masks error notification to Root Complex when Bad TLP Error is detected. 0b: no mask 1b: Mask error message transmission
5 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	Receiver Error Mask Optional	0h	RW	0b: no mask 1b: Mask error message transmission

Table 6.6-109 Valid Reset Signal

Reset Signal	Advisory Non-Fatal Error Mask	Replay Timer Timeout Mask	REPLAY_NUM Rollover Mask	Bad DLLP Mask	Bad TLP Mask	Receiver Error Mask Optional
RST_LOAD_B						
RST_RSM_B	✓	✓	✓	✓	✓	✓
RST_CFG_B						
FLR						

(56) Advanced Error Capabilities and Control Register (Function #n) (PCI_EP_ADVECC_Fn) (n = 0, 1)

This register indicates and controls the advanced error capabilities.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6118h
Initial Value : 0000_00A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	ECRC Check Enable	ECRC Check Capable	ECRC Generation Enable	ECRC Generation Capable	First Error Pointer[4:0]				
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	RW	R	RW	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	RW	R	RW	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	ECRC Check Enable	0h	RW	ECRC Check Enable setting 0b: Disable (Default) 1b: Enable
7	ECRC Check Capable	1h	R	Presence or absence of ECRC Check function 0b: Absent 1b: Present (Default)
6	ECRC Generation Enable	0h	RW	ECRC Generation Enable setting 0b: Disable (Default) 1b: Enable
5	ECRC Generation Capable	1h	R	Presence or absence of ECRC Generation function 0b: Absent 1b: Present (Default)
4 to 0	First Error Pointer[4:0]	0h	R	Indicates the field value of the Uncorrectable Error Status register for the first detected Uncorrectable Error

Table 6.6-110 Valid Reset Signal

Reset Signal	ECRC Check Enable	ECRC Generation Enable
RST_LOAD_B		
RST_RSM_B	✓	✓
RST_CFG_B		
FLR		

(57) Header Log Register 0 (Function #n) (PCI_EP_HLOG0_Fn) (n = 0, 1)

This register indicates the header log.

Access Size : 32 bits
Offset Address : <PCI0_base> + 611Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Header of TLP associated with error 0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Header of TLP associated with error 0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Header of TLP associated with error 0[31:0]	0h	R	Indicates the 1st DW of Header for the first detected Uncorrectable Error.

(58) Header Log Register 1 (Function #n) (PCI_EP_HLOG1_Fn) (n = 0, 1)

This register indicates the header log.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6120h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Header of TLP associated with error 1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Header of TLP associated with error 1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Header of TLP associated with error 1[31:0]	0h	R	Indicates the 2nd DW of Header for the first detected Uncorrectable Error.

(59) Header Log Register 2 (Function #n) (PCI_EP_HLOG2_Fn) (n = 0, 1)

This register indicates the header log.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6124h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Header of TLP associated with error 2[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Header of TLP associated with error 2[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Header of TLP associated with error 2[31:0]	0h	R	Indicates the 3rd DW of Header for the first detected Uncorrectable Error.

(60) Header Log Register 3 (Function #n) (PCI_EP_HLOG3_Fn) (n = 0, 1)

This register indicates the header log.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6128h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Header of TLP associated with error 3[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Header of TLP associated with error 3[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	Header of TLP associated with error 3[31:0]	0h	R	Indicates the 4th DW of Header for the first detected Uncorrectable Error.

(61) Root Error Command (Function #n) (PCI_EP_ROOTEC_Fn) (n = 0, 1)

This function is not supported.

(62) Root Error Status (Function #n) (PCI_EP_ROOTES_Fn) (n = 0, 1)

This function is not supported.

(63) Error source Identification Register (Function #n) (PCI_EP_ERRSI_Fn) (n = 0, 1)

This function is not supported.

(64) Device Serial Number Extended Capability Register (Function #n) (PCI_EP_DEVSNEXTC_Fn) (n = 0, 1)

This register specifies the device serial number extended capability.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6150h
Initial Value : 1B01_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Next Capability Offset[11:0]												Capability Version[3:0]			
Initial Value	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCI Express Extended Capability ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	Next Capability Offset[11:0]	1B0h	R	Indicates the starting address of the Secondary PCI Express Extended Capability Header. 1B0h fixed. Function#1 is fixed at 000h.
19 to 16	Capability Version[3:0]	1h	R (RW)	Indicates the version of the Capability Structure. Default: 0001b
15 to 0	PCI Express Extended Capability ID[15:0]	3h	R	Indicates the Device Serial Number Extended Capability. Default: 0003h

Table 6.6-111 Valid Reset Signal

Reset Signal	Capability Version
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(65) Serial Number Register (Lower DW) (Function #n) (PCI_EP_SNL_Fn) (n = 0, 1)

This register specifies the serial number of the device.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6154h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PCI Express Device Serial Number Lower DW[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCI Express Device Serial Number Lower DW[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PCI Express Device Serial Number Lower DW[31:0]	0h	R (RW)	The lower 32 bits of the IEEE standard 64-bit unique ID (EUI-64). EUI-64 consists of a 24-bit company ID and a 40-bit vendor-defined extension. When using a unique ID, write from the UDL side.

Note 1. The “PCI Express Device Serial Number” value must be the same value for all functions including lower/higher DW.

Table 6.6-112 Valid Reset Signal

Reset Signal	PCI Express Device Serial Number (Lower DW)
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(66) Serial Number Register (Higher DW) (Function #n) (PCI_EP_SNU_Fn) (n = 0, 1)

This register specifies the serial number of the device.

Access Size : 32 bits
Offset Address : <PCI0_base> + 6158h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PCI Express Device Serial Number Upper DW[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCI Express Device Serial Number Upper DW[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PCI Express Device Serial Number Upper DW[31:0]	0h	R (RW)	Upper 32 bits of the IEEE standard 64-bit unique ID (EUI-64). EUI-64 consists of a 24-bit company ID and a 40-bit vendor-defined extension. When using a unique ID, write from the UDL side.

Table 6.6-113 Valid Reset Signal

Reset Signal	PCI Express Device Serial Number (Higher DW)
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(67) Secondary PCI Express Extended Capability Header (Function #0) (PCI_EP_SPEECH_F0)

Access Size : 32 bits
Offset Address : <PCI0_base> + 61B0h
Initial Value : 0001_0019h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Next Capability Offset[11:0]												Capability Version[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCI Express Extended Capability ID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	Next Capability Offset[11:0]	0h	R	Indicates that this Capability List is the final List. 000h fixed
19 to 16	Capability Version[3:0]	1h	R (RW)	Indicates the version of the Capability Structure. Default: 0001b
15 to 0	PCI Express Extended Capability ID[15:0]	19h	R	Indicates the Secondary PCI Express Extended Capability Header. Default: 0019h

Table 6.6-114 Valid Reset Signal

Reset Signal	Capability Version
RST_LOAD_B	✓
RST_RSM_B	
RST_CFG_B	
FLR	

(68) Link Control 3 Register (Function #0) (PCI_EP_LINC3_F0)

Access Size : 32 bits
Offset Address : <PCI0_base> + 61B4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 9	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Table 6.6-115 Valid Reset Signal

Reset Signal	Enable Lower SKP OS Generation Vector
RST_LOAD_B	
RST_RSM_B	
RST_CFG_B	✓
FLR	✓

(69) Lane Error Status Register (Function #0) (PCI_EP_LESTA_F0)

Access Size : 32 bits
Offset Address : <PCI0_base> + 61B8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Lane Error Status Bits[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(PCIe)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW
R/W(UDL)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	Lane Error Status Bits[1:0]	0h	RW (RW1)	Each bit indicates if the corresponding Lane detected a Lane-based error. A value of 1b indicates that a Lane based-error was detected on the corresponding Lane Number. The default value of each bit is 0b. For Links that are narrower than 32 bits, the unused upper bits [31:Bits Maximum Link Width] are RsvdZ Default 0h

Table 6.6-116 Valid Reset Signal

Reset Signal	Lane Error Status Bits
RST_LOAD_B	
RST_RSM_B	✓
RST_CFG_B	
FLR	

(70) Lane Equalization Control Register (Function #0) (PCI_EP_LEQCTL_F0)

Access Size : 32 bits

Offset Address : <PCI0_base> + 61BCh

Initial Value : xx00_xx00h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	Upstream Port 8.0 GTs Receiver Preset Hint for Lane 1[2:0]			Upstream Port 8.0 GTs Transmitter Preset for Lane 1[3:0]				-	Downstream Port 8.0 GTs Receiver Preset Hint for Lane 1[2:0]			Downstream Port 8.0 GTs Transmitter Preset for Lane 1[3:0]			
Initial Value	0	x	x	x	x	x	x	x	0	0	0	0	0	0	0	0
R/W(PCIe)	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit
R/W(UDL)	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	Upstream Port 8.0 GTs Receiver Preset Hint for Lane 0[2:0]			Upstream Port 8.0 GTs Transmitter Preset for Lane 0[3:0]				-	Downstream Port 8.0 GTs Receiver Preset Hint for Lane 0[2:0]			Downstream Port 8.0 GTs Transmitter Preset for Lane 0[3:0]			
Initial Value	0	x	x	x	x	x	x	x	0	0	0	0	0	0	0	0
R/W(PCIe)	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit
R/W(UDL)	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	R	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit	Hwinit

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, undefined value is read. The written value will be ignored.
30 to 28	Upstream Port 8.0 GTs Receiver Preset Hint for Lane 1[2:0]	xh	Hwinit	The read value is undefined This field contains the Receiver Preset Hint value sent or received during Link Equalization. Usage of this field varies as follows: A: Downstream Port This field contains the value sent on the associated Lane during Link Equalization. This field is HwInit. B: Upstream Port, Crosslink Supported = 0b This field is intended for debug and diagnostics. It contains the value captured from the associated Lane during Link Equalization. This field is RO. Note: When crosslinks are supported, case C (below) applies and this captured information is not visible to software. Vendors are encouraged to provide an alternate mechanism to obtain this information. C: Upstream Port, Crosslink Supported = 1b This field is not used or affected by the current Link Equalization. The field value will be used if a future crosslink negotiation switches the Operating Port Direction so that case A (above) applies. This field is HwInit. The default value is 111b. EP: input signal

Bit	Bit Name	Initial Value	R/W	Description
27 to 24	Upstream Port 8.0 GTs Transmitter Preset for Lane 1[3:0]	xh	Hwinit	The read value is undefined This field contains the Transmit Preset value sent or received during Link Equalization. Usage of this field varies as follows: A: Downstream Port This field contains the value sent on the associated Lane during Link Equalization. This field is Hwlnit. B: Upstream Port, Crosslink Supported = 0b This field is intended for debug and diagnostics. It contains the value captured from the associated Lane during Link Equalization. Field is RO. Note: When crosslinks are supported, case C (below) applies and this captured information is not visible to software. Vendors are encouraged to provide an alternate mechanism to obtain this information. C: Upstream Port, Crosslink Supported = 1b This field is not used or affected by the current Link Equalization. The field value will be used if a future crosslink negotiation switches the Operating Port Direction so that case A (above) applies. This field is Hwlnit. The default value is 1111b. EP: input signal
23	-	0h	R	Reserved Whenever it is read, undefined value is read. The written value will be ignored.
22 to 20	Downstream Port 8.0 GTs Receiver Preset Hint for Lane 1[2:0]	All 0	Hwinit	The read value is undefined Receiver Preset Hint may be used as a hint for receiver equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port. For an Upstream Port if Crosslink Supported is 0b, this field is RsvdP. Otherwise, this field is Hwlnit. The default value is 111b. EP: Fixed to 0 because Crosslink is not supported
19 to 16	Downstream Port 8.0 GTs Transmitter Preset for Lane 1[3:0]	All 0	Hwinit	The read value is undefined Transmitter Preset used for equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port. For an Upstream Port if Crosslink Supported is 0b, this field is RsvdP. Otherwise, this field is Hwlnit. The default value is 1111b. EP: Fixed to 0 because Crosslink is not supported
15	-	0h	R	Reserved Whenever it is read, undefined value is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
14 to 12	Upstream Port 8.0 GTs Receiver Preset Hint for Lane 0[2:0]	xh	Hwinit	The read value is undefined This field contains the Receiver Preset Hint value sent or received during Link Equalization. Usage of this field varies as follows: A: Downstream Port This field contains the value sent on the associated Lane during Link Equalization. This field is Hwlnit. B: Upstream Port, Crosslink Supported = 0b This field is intended for debug and diagnostics. It contains the value captured from the associated Lane during Link Equalization. This field is RO. Note: When crosslinks are supported, case C (below) applies and this captured information is not visible to software. Vendors are encouraged to provide an alternate mechanism to obtain this information. C: Upstream Port, Crosslink Supported = 1b This field is not used or affected by the current Link Equalization. The field value will be used if a future crosslink negotiation switches the Operating Port Direction so that case A (above) applies. This field is Hwlnit. The default value is 111b. EP: input signal
11 to 8	Upstream Port 8.0 GTs Transmitter Preset for Lane 0[3:0]	xh	Hwinit	The read value is undefined This field contains the Transmit Preset value sent or received during Link Equalization. Usage of this field varies as follows: A: Downstream Port This field contains the value sent on the associated Lane during Link Equalization. This field is Hwlnit. B: Upstream Port, Crosslink Supported = 0b This field is intended for debug and diagnostics. It contains the value captured from the associated Lane during Link Equalization. This field is RO. Note: When crosslinks are supported, case C (below) applies and this captured information is not visible to software. Vendors are encouraged to provide an alternate mechanism to obtain this information. C: Upstream Port, Crosslink Supported = 1b This field is not used or affected by the current Link Equalization. The field value will be used if a future crosslink negotiation switches the Operating Port Direction so that case A (above) applies. This field is Hwlnit. The default value is 1111b. EP: input signal
7	-	0h	R	Reserved Whenever it is read, undefined value is read. The written value will be ignored.
6 to 4	Downstream Port 8.0 GTs Receiver Preset Hint for Lane 0[2:0]	All 0	Hwinit	The read value is undefined Receiver Preset Hint may be used as a hint for receiver equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port. For an Upstream Port if Crosslink Supported is 0b, this field is RsvdP. Otherwise, this field is Hwlnit. The default value is 111b. EP: Fixed to 0 because Crosslink is not supported
3 to 0	Downstream Port 8.0 GTs Transmitter Preset for Lane 0[3:0]	All 0	Hwinit	The read value is undefined Transmitter Preset used for equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port. For an Upstream Port if Crosslink Supported is 0b, this field is RsvdP. Otherwise, this field is Hwlnit. The default value is 1111b. EP: Fixed to 0 because Crosslink is not supported

6.6.4.3 PHY Register Descriptions

The following lists the PHY registers. These registers control the PHY layer.

6.6.4.3.1 List of Physical Layer Control/Status Registers

The following is a list of physical layer control/status registers.

These registers control the pins for setting the PHY built into the macro.

When these registers are accessed for writing, PHY_REG_CLK_EN (permission register (offset: 300h) bit[1]) must be set to 1b before a write access is made. In addition, rewrite the registers while the PCIe logic and PHY resets (other than ARESETn, RST_CFG_B, and RST_LOAD_B) are asserted.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
XCFGD Setting Register	PCI_PHY_XCFGD	xxxx_xxxxh	2000h / 2010h / 2020h / … / 2230h	32
Reserved	-	-	2244h to 23FFh	-
XCFGA_CMN Setting Register	PCI_PHY_XCFGD_CMN	xxxx_xxxxh	2400h / 2410h / 2420h / … / 24F0h	32
Reserved	-	-	24F4h to 24FFh	-
XCFGA_L0 Setting Register	PCI_PHY_XCFGD_L0	xxxx_xxxxh	2500h / 2510h / 2520h / … / 2550h	32
Reserved	-	-	2554h to 255Fh	-
XCFGA_L1 Setting Register	PCI_PHY_XCFGD_L1	xxxx_xxxxh	2560h / 2570h / 2580h / … / 25B0h	32
Reserved	-	-	25B4h to 27FFh	-
XCFGO Monitor Register	PCI_PHY_XCFGO	xxxx_xxxxh	2800h / 2810h / 2820h / … / 28A0h	32
Reserved	-	-	28A4h to 2BFFh	-
MISC Setting Register	PCI_PHY_MISC	0000_0000h	2A00h	32
Reserved	-	-	2A04h to 2DFFh	-
PHY Monitor Register 1	PCI_PHY_MON1	xxxx_xxxxh	2E00h	32
Reserved	-	-	2E04h to 2E0Fh	-
PHY Monitor Register 2	PCI_PHY_MON2	0000_00x0h	2E10h	32

6.6.4.3.2 Register Descriptions

(1) XCFGD Setting Register (PCI_PHY_XCFGD)

This register is for setting the PHY setting pin XCFGD of the PCI Express core.

Do not change this register from its initial value.

Access Size : 32 bits
Offset Address : <PCI0_base> + 2000h / 2010h / 2020h / … / 2230h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	XCFGD[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	XCFGD[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	XCFGD[31:0]	xh	RW	Sets each bit of the PHY setting pin XCFGD.

Note: x = Undefined

(2) XCFGA_CMN Setting Register (PCI_PHY_XCFGD_CMN)

This register is for setting the PHY setting pin XCFGA_CMN of the PCI Express core.

Do not change this register from its initial value.

Access Size : 32 bits

Offset Address : <PCI0_base> + 2400h / 2410h / 2420h / … / 24F0h

Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	XCFGA_CMN[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	XCFGA_CMN[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	XCFGA_CMN [31:0]	xh	RW	Sets each bit of the PHY setting pin XCFGA_CMN

Note: x = Undefined

(3) XCFGA_L0 Setting Register (PCI_PHY_XCFGD_L0)

This register is for setting the PHY setting pin XCFGA_L0 of the PCI Express core.

Do not change this register from its initial value.

Access Size : 32 bits

Offset Address : <PCI0_base> + 2500h / 2510h / 2520h / … / 2550h

Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	XCFGA_L0[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	XCFGA_L0[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	XCFGA_L0 [31:0]	xh	RW	Sets each bit of the PHY setting pin XCFGA_L0

Note: x = Undefined

(4) XCFGA_L1 Setting Register (PCI_PHY_XCFGD_L1)

This register is for setting the PHY setting pin XCFGA_L1 of the PCI Express core.

Do not change this register from its initial value.

Access Size : 32 bits

Offset Address : <PCI0_base> + 2560h / 2570h / 2580h / … / 25B0h

Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	XCFGA_L1[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	XCFGA_L1[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	XCFGA_L1 [31:0]	xh	RW	Sets each bit of the PHY setting pin XCFGA_L1

Note: x = Undefined

(5) XCFGO Monitor Register (PCI_PHY_XCFGO)

This is a register for monitoring the PHY Debug pin XCFGO of the PCI Express core.

Do not change this register from its initial value.

Access Size : 32 bits

Offset Address : <PCI0_base> + 2800h / 2810h / 2820h / … / 29F0h

Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	XCFGO[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	XCFGO[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	XCFGO[31:0]	xh	R	Register for monitoring the PHY Debug pin XCFGO inside the macro

Note: x = Undefined

(6) MISC Setting Register (PCI_PHY_MISC)

This register is for setting the PHY sideband pin of the PCI Express core.

Do not change this register from its initial value.

Access Size : 32 bits
Offset Address : <PCI0_base> + 2A00h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VCONTROL[4:0]					-	DEBUG_SEL[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 8	VCONTROL [4:0]	0h	RW	This pin is for setting the PHY test mode. Fix to 0b for normal use.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 0	DEBUG_SEL [6:0]	0h	RW	Switches the PHY internal information output to PHY Monitor register 1 (Offset: 2E00h).

(7) PHY Monitor Register 1 (PCI_PHY_MON1)

This register is monitoring PHY of the PCI Express core.

Do not change this register from its initial value.

Access Size : 32 bits
Offset Address : <PCI0_base> + 2E00h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DEBUG_OUT[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEBUG_OUT[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DEBUG_OUT [31:0]	xh	R	PHY internal state/internal signal monitor area.

Note: x = Undefined

(8) PHY Monitor Register 2 (PCI_PHY_MON2)

This register is monitoring PHY of the PCI Express core.

Do not change this register from its initial value.

Access Size : 32 bits
Offset Address : <PCI0_base> + 2E10h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	RxElecIdle_L1	RxElecIdle_L0	BistOK_G3_L1	BistOK_G2_L1	BistOK_G3_L0	BistOK_G2_L0
Initial Value	0	0	0	0	0	0	0	0	0	0	x	x	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	RxElecIdle_L1	xh	R	Electrical idle signal for Lane 1 Receiver
4	RxElecIdle_L0	xh	R	Electrical idle signal for Lane 0 Receiver
3	BistOK_G3_L1	0h	R	BIST test result output pin for Lane1 in 8.0 GT/s mode
2	BistOK_G2_L1	0h	R	BIST test result output pin for Lane1 in 2.5 GT/s and 5.0 GT/s modes.
1	BistOK_G3_L0	0h	R	BIST test result output pin for Lane0 in 8.0 GT/s mode
0	BistOK_G2_L0	0h	R	BIST test result output pin for Lane0 in 2.5 GT/s and 5.0 GT/s modes.

Note: x = Undefined

6.6.5 Functional Description

6.6.5.1 PCIe Core Functions

The functional description of the unit is given below.

This unit is configured based on the Base Spec 4.0. For the detailed specification, refer to the Base Spec 4.0. This section mainly describes the parts specific to Renesas Electronics.

6.6.5.2 Issuing of PCIe Requests and Register Access (by AXI)

The following gives functional description on access to the PCI Express core by the AXI bus.

6.6.5.2.1 PCIe Requests which can be Issued (Supported Commands for TX)

Requests to be issued via pcie windows

MRd: Memory Read Request

MWr: Memory Write Request

Requests to be issued by the registers

MRd: Zero-Length Memory Read Request

CfgRd0: Configuration Read Type 0

CfgWr0: Configuration Write Type 0

CfgRd1: Configuration Read Type 1

CfgWr1: Configuration Write Type 1

Msg: Message Request

MsgD: Message Request with data payload

Unsupported requests (issuance prohibited)

IORd: I/O Read Request

IOWr: I/O Write Request

MRdLk: Memory Read Request-Locked

CplDLk: Completion for Locked Memory Read

CAUTION

If non-posted request with data (CfgWr0/CfgWr1/IOWr) which should not be transmitted is issued inadvertently in EPS mode, an MRd (non-posted request without data) which can be normally transmitted in EP mode may not also be transmitted after that.

6.6.5.2.2 Register Access

1) Internal register read/write

2) Configuration register read/write

The following restrictions apply to access to the configuration registers.

- Byte-lane transfer is only acceptable if the transfer consists of consecutive valid bytes.
- If the valid bytes are not consecutive, only values which are no greater than 32 bits (1 dword) and do not span the dword alignment are acceptable.

  Example) A value which can represent a dword TLP for PCIe

6.6.5.2.3 Issuing Memory Requests

Access from the AXI side is converted into a PCIe request and then issued. Up to eight windows can be allocated. Note that only incremental bursts and fixed bursts (of 1 beat) are acceptable. Wrapping bursts and fixed bursts (of 2 or more beats) are prohibited. If these prohibited bursts are received, the unit operates as follows.

- Although the response will in general be “OKAY” (except in cases that involved a protocol error of the AXI), this is beyond the scope of guaranteed operation.
- Unexpected requests may be issued to the PCIe bus.
- Unexpected registers may be modified.

To maintain the order between memory writing and memory reading, only issue next transactions after the reception of responses to writing. Buffering is not available in the issuing of memory write requests. (Refer to the description of AWCACHE* of Mode Set 0 Register (offset: 314h).)

(1) Memory Write Transaction from the AXI

A write transaction from the AXI via a window is converted into a MWr command and then issued.

- Number of write transactions which can be accepted at a time: 1
- Write data are held in an internal buffer.
- The order between memory write transactions is preserved (with the exception of transactions for messages and configuration).
- The order of memory write and other transactions is not preserved (a preceding memory read may be overtaken).
- To preserve the order between memory writing and memory reading or of Msg and MsgD, make sure that a next transaction is only issued after a response is returned. Buffering is not available in the issuing of memory write requests.
- Writing does not proceed if the PCI power state is not D0.
- Transactions where all bits of WSTRB are 0 during a burst return an “OKAY” response, but the written data are not reflected.
- Memory requests are not issued after an acknowledgement for PME_Turn_Off Message reception/Non-D0 State transition request reception has been asserted (Precautions: Only for Endpoint mode).

(2) Memory Read Transaction from the AXI

A read transaction from the AXI via a window is converted into an MRd command and then issued.

- Number of read transactions which can be accepted at a time: 1 to 8
- Read data are held in an internal buffer to preserve the order of transactions with the same ID.
- To preserve the order of memory read transactions from a given master ID, the system can also wait until the indicator of completion of a preceding read transaction is returned. The method of waiting is selectable as either of the above by the setting of an internal register as listed in the table below.

PCIe Request Order*1	Method of Waiting	Performance	Severity of Order
0 (initial value)	Data read are held in an internal buffer.	✓	—
1	Have the system wait for a read request to be issued.	—	✓

Note 1. The order can be set by using the PCIe Request Order bit (bit [0]) of Mode Set 1 Register (offset: 318h).

- The order of memory read transactions from a different master ID is not preserved.
- The order of memory read and other transactions is not preserved.
- To preserve the order between memory reading and memory writing, make sure that a next transaction is only issued after a response is returned. Buffering is not available in the issuing of memory write requests.
- MRdLk requests are not supported and therefore cannot be issued.
- Reading does not proceed if the PCI power state is not D0.
- Memory requests are not issued after an acknowledgement for PME_Turn_Off Message reception/Non-D0 State transition request reception has been asserted (Precautions: Only for Endpoint mode).

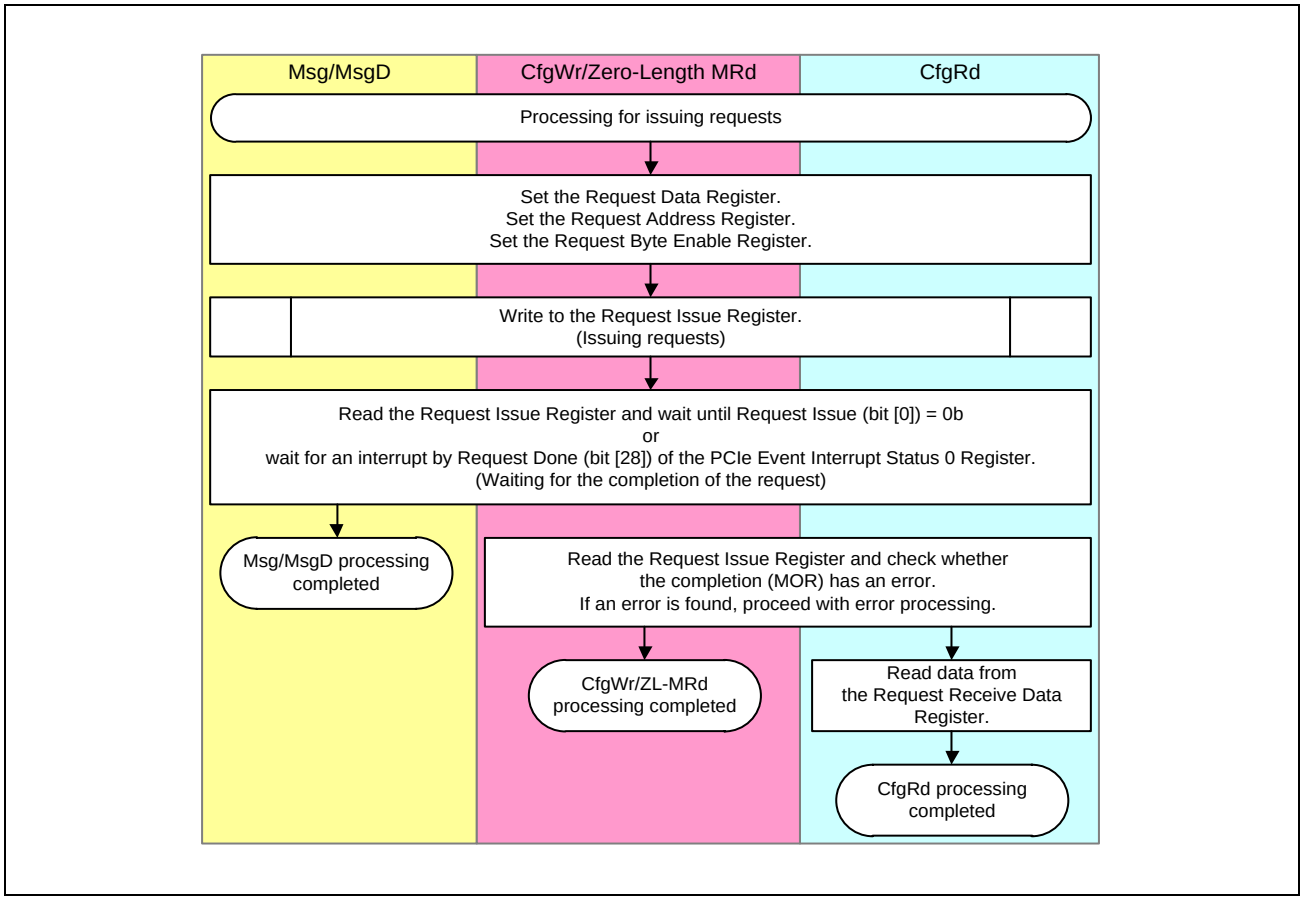
6.6.5.2.4 Issuing Special Requests

The following requests can be issued by controlling the internal registers.

- Configuration Read
- Configuration Write
- Zero-Length Memory Read Request
- Message Request
- Message Request with data payload

The internal registers are only writable from the AXI side. Attempted writing from the PCIe side is ignored and the write operation returns an “OKAY” response, but the written data are not reflected.

The figure below shows the flow of issuing requests by the registers.



(1) Issuing Message Requests

A message request is automatically issued by setting the Request Issue bit (bit [0]) of the Request Issue Register after access to an internal register through the AXI slave interface and setting a destination in Request Address Register 1.

(a) Flow of Issuing Message Requests

An example of the flow of issuing message requests is described below.

1. Set a destination in Request Address Register 1.

Bits [31:27]:	Reserved	Fixed to 0_0000b.
Bits [26:24]:	Routing Type	Specify the message routing.
Bits [23:8]:	Reserved	Fixed to 0000h.
Bits [7:0]:	Message Code	Specify the message code.

2. Set the 3rd header in Request Data Register 1 as required.
3. Set the 4th header in Request Data Register 2 as required.
4. For MsgD, set write data in Request Data Register 3 (not required in the case of Msg).
5. Message requests are automatically issued by setting the Request Issue bit (bit [0]) of the Request Issue Register to 1b (the TR type bits (bits [11:8]) of the Request Issue Register are used to set Msg/MsgD).
6. Read the Request Issue Register and wait (polling) or wait for an interrupt of Request Done (bit [28]) of PCIe Event Interrupt Status 0 Register until the Request Issue bit (bit [0]) is set to 0b, wait for the request to end.

(b) Notes when Issuing Message Requests

Points to note when issuing message requests are described below.

- INTx and error type messages are automatically issued. They do not require issuing as message requests by the user (prohibited).
- Set message headers appropriately.
- A value to be set in a message header must be 0 except when sending a vendor defined message.

(2) Issuing Zero-Length Read Requests

A zero-length read request is automatically issued by setting a destination in Request Address Register 1 after access to an internal register through the AXI slave interface and setting the Request Issue bit (bit [0]) of the Request Issue Register.

(a) Flow of Issuing Zero-Length Read Requests

An example of the flow of issuing zero-length read requests is described below.

1. Set a destination in Request Address Register 1.

   Bits [31:2]: Address Set the address.

   Bits [1:0]: Reserved Fixed to 00b.

2. Set a destination in Request Address Register 1.

   Bits [64:32]: Address Set the address.

3. Set Byte Enable (0000b) in Request Byte Enable Register.
4. Zero-length read requests are automatically issued by setting the Request Issue bit (bit [0]) of the Request Issue Register to 1b (the TR type bits (bits [11:8]) of the Request Issue Register are used to set zero-length read).
5. Read the Request Issue Register and wait (polling) or wait for an interrupt of Request Done (bit [28]) of PCIe Event Interrupt Status 0 Register until the Request Issue bit (bit [0]) is set to 0b, wait for the request to end.

6.6.5.3 Initiation of AXI Transactions and Register Access (by PCIe)

The following gives functional description on access to the AXI bus by the PCI Express core.

6.6.5.3.1 PCIe Requests which can be Received (Supported Commands for RX)

MRd: Memory Read Request

MWr: Memory Write Request

CfgRd0: Configuration Read Type 0

CfgWr0: Configuration Write Type 0

Msg: Message Request

MsgD: Message Request with data payload

Cpl: Completion

CplD: Completion with Data

The following requests are not supported.

MRdLk: Memory Read Request-Locked

IORd: I/O Read Request

IOWr: I/O Write Request

CplLK: Completion for Locked Memory Read without Data

CplDLk: Completion for Locked Memory Read

CfgRd1: Configuration Read Type 1

CfgWr1: Configuration Write Type 1

AtomicOP is not supported.

6.6.5.3.2 Initiation of AXI Transactions

Access from the PCIe side is converted into an AXI transaction and then issued. Up to eight windows can be allocated.

To avoid a read-after-write (RAW) hazard, only issue a next memory write request following reception of the completion TLP for the previous one.

Table 6.6-117 AXI Transaction from PCIe Request

PCIe Request	AXI Burst Type	AXI Bus Size	Division
MWr/MRd	Incremental	64	Yes
		32	No
IOWr/IORd	Unsupported request		
MRdLk	Unsupported request		

(1) Memory Write Transaction from the PCIe

- Allowable number of write requests to be issued: Variable (the value is set by the AXI Max Issue Write bits (bits [15:12]) of Mode Set 1 Register).
- Write data are not held in an internal buffer (buffering within the PCIe core).
- The order between memory write transactions is preserved.
- The order of memory writing and memory reading is not preserved.
- To preserve the order between memory writing and memory reading, make sure that a next request is only issued after a completion TLP is returned.
- AWCACHE and AWPROT can be set by using Mode Set 0 Register.
- In cases of contention between DMA transfer (from PCIe to AXI) and memory write transaction, requests are accepted in turn in round-robin fashion.

(2) Memory Read Transaction from the PCIe

- Allowable number of read requests to be issued: Variable (the value is set by the AXI Max Issue Read bits (bits [11:8]) of Mode Set 1 Register).
- Since the PCI specification requires dword-aligned memory addresses in transfer, invalid byte lanes may be read. If an AXI slave as a read destination has a register or FIFO buffer which has been cleared by being read, the transaction may be non-compliant.
- Data read are held in an internal buffer.
- The order between memory read operations is preserved.
- A preceding memory write is not overtaken.
  Data read are held in an internal buffer until a preceding memory write is completed.
  When a zero-length read request is received, an “OKAY” response is returned, but the written data are not reflected and a completion TLP is transmitted after waiting for the completion of the preceding memory write.
  Data read are held in an internal buffer until a preceding memory write is completed.
- ARCACHE can be set by using Mode Set 0 Register.
- In cases of contention between DMA transfer (from AXI to PCIe) and memory read transaction, requests are accepted in turn in round-robin fashion.

6.6.5.3.3 Narrow Transfer

Narrow transfer to be initiated by the AXI transfer only supports the following AXI transactions.

- MAxBURST = INCR
- MAxSIZE = 0h to 2h (8 bits / 16 bits / 32 bits)
- MAxLEN = 0h (1 beat)

The following restrictions apply to TLPs from the PCIe module in narrow transfer.

- The length is 1 dword.
- The First Byte Enable bit only supports the following.
  - MAxSIZE = 0h (8 bits): 1000b / 0100b / 0010b / 0001b
  - MAxSIZE = 1h (16 bits): 1100b / 0011b
  - MAxSIZE = 2h (32 bits): 1111b

6.6.5.4 DMAC Functions

This section explains functions of the DMAC within the unit. Control by registers and by descriptors are both supported as methods of DMA control. The method of control is independently selected per channel.

6.6.5.4.1 Register-Type Transfer

DMA transfer from AXI to PCIe or vice versa is handled by software making register settings. The table below lists the DMAC registers for execution of register-type DMA transfer.

Table 6.6-118 Registers Related to Register-Type DMA Transfer

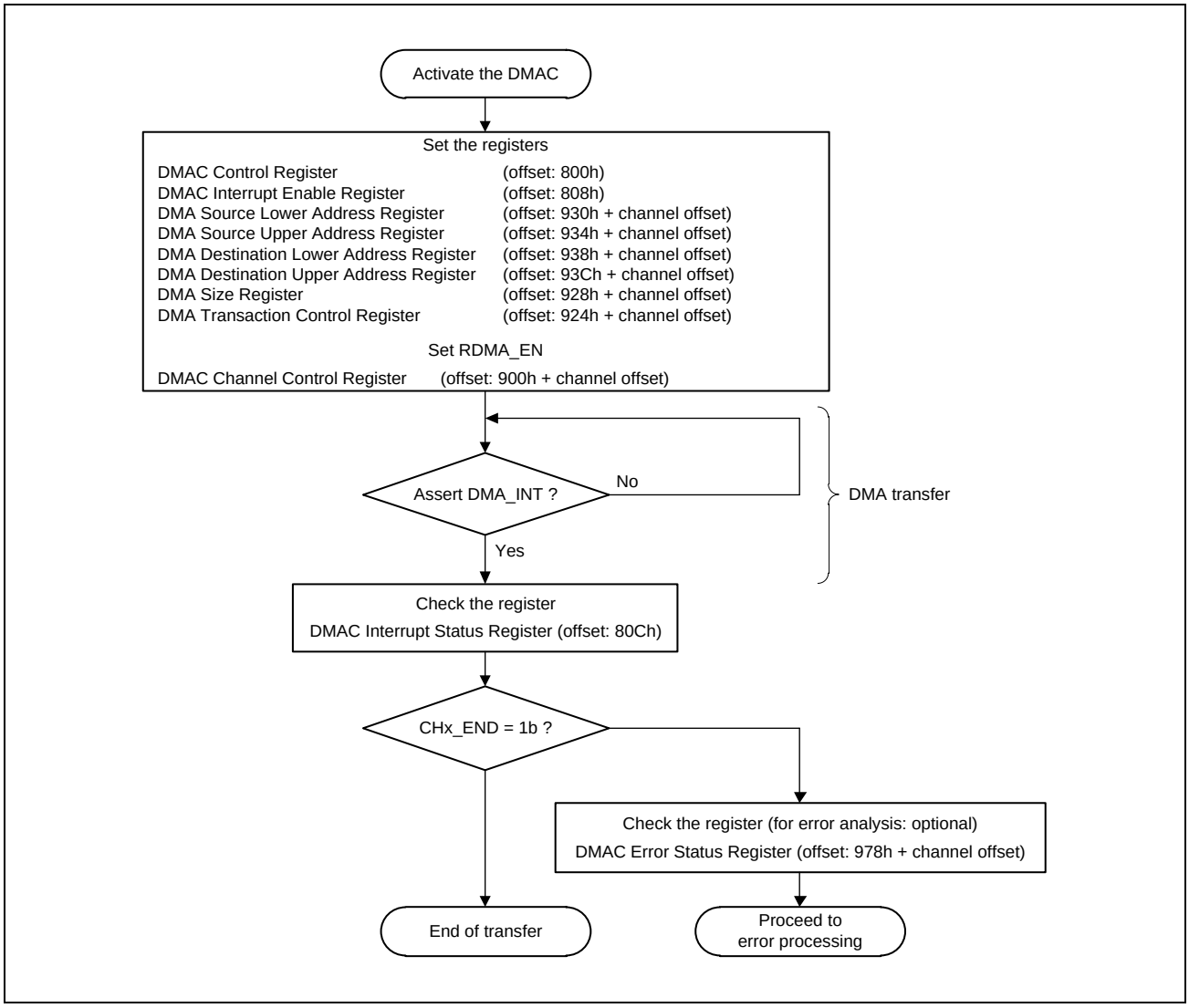
Common Control	
800h	DMAC Control Register
808h	DMAC Interrupt Enable Register
80Ch	DMAC Interrupt Status Register
Channel Control	
900h + channel offset	DMA Channel Control Register m
DMA Setting	
924h + channel offset	DMA Transaction Control Register m
928h + channel offset	DMA Size Register m
930h + channel offset	DMA Source Lower Address Register m
934h + channel offset	DMA Source Higher Address Register m
938h + channel offset	DMA Destination Lower Address Register m
93Ch + channel offset	DMA Destination Higher Address Register m
DMA Status	
950h + channel offset	DMA Rest Size Register m
960h + channel offset	AXI Request Address (Lower) Register m
964h + channel offset	AXI Request Address (Higher) Register m
968h + channel offset	PCIe Request Address(Lower) Register m
96Ch + channel offset	PCIe Request Address(Higher) Register m
978h + channel offset	DMAC Error Status Register m

(1) Flow of Operations

The following describes the procedures for settings to activate and stop the DMAC in the case of register-type transfer.

(a) Activation and Normal Operation

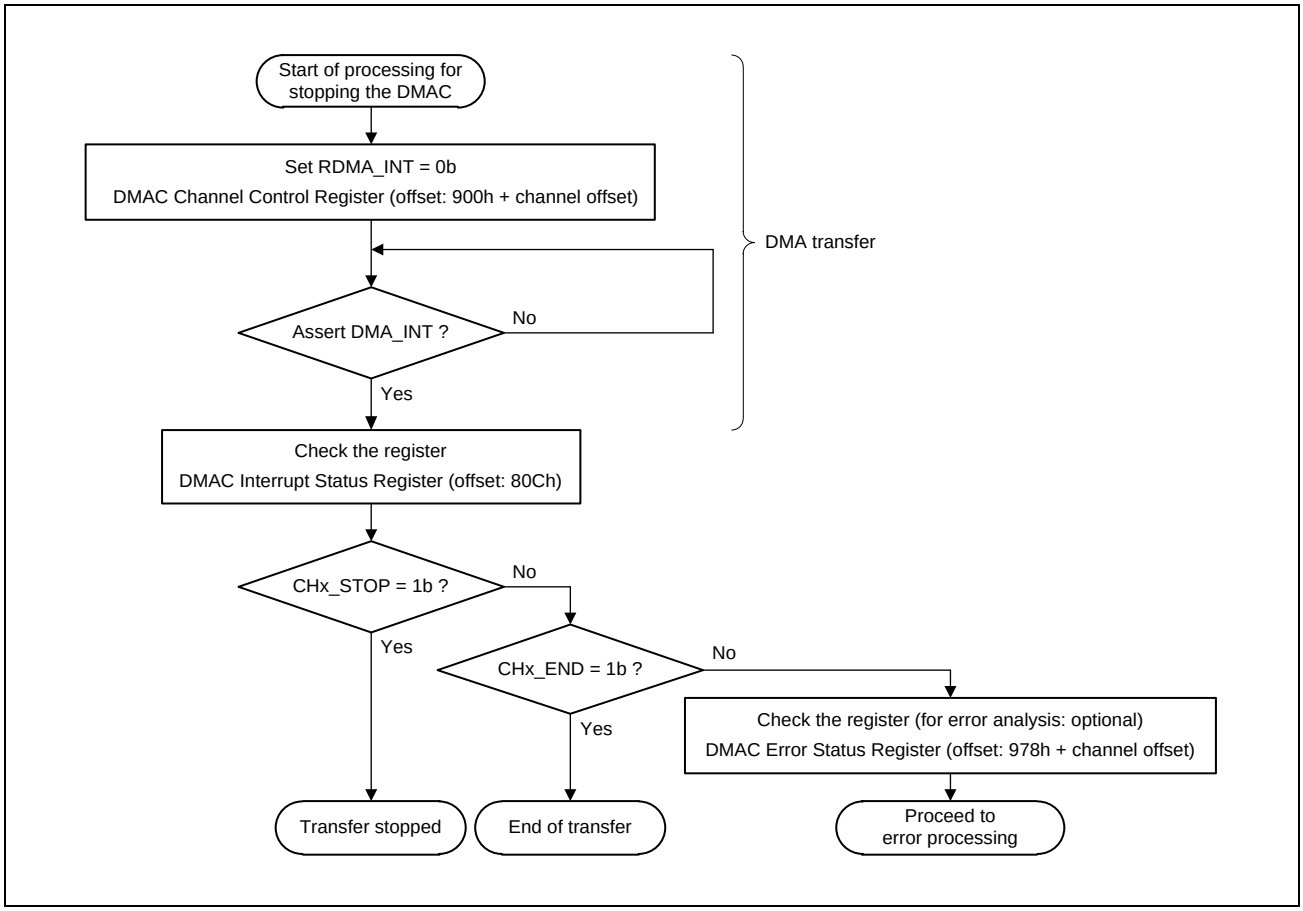
Before activating the DMAC, set the PCIe and AXI windows. After that, set the registers of the DMAC to start the DMAC.



(b) Stopping DMA Transfer

The following describes the procedure for settings to stop DMA transfer under software control.

- DMA transfer can be stopped by writing 0b to the RDMA_EN (register type DMA control enable) bit (bit [0]) of the DMAC channel control register (offset: 900h + channel offset).
- Following the detection of RDMA_EN being 0b, the unit waits for the completion of any request which has already been issued before asserting the interrupt signal (DMA_INT).
- After waiting for the assertion of DMA_INT, poll the CHx_STOP bit of the DMAC Interrupt Status Register (offset: 80Ch) to check that its value is 1b.
- This indicates that DMA transfer has been stopped. If the CHx_STOP bit is 0b, check the CHx_END bit for the given channel in the same register.
- If a last request is still being executed when RDMA_EN is set to 0b, the CHx_END bit indicates the completion of DMA transfer as in normal operation.
- The address to which transfer was most recently completed before it was stopped and the remaining number of bytes for transfer can be checked by reading the DMA Status registers (offset: 950h + channel offset to 978h + channel offset) for the given channel.

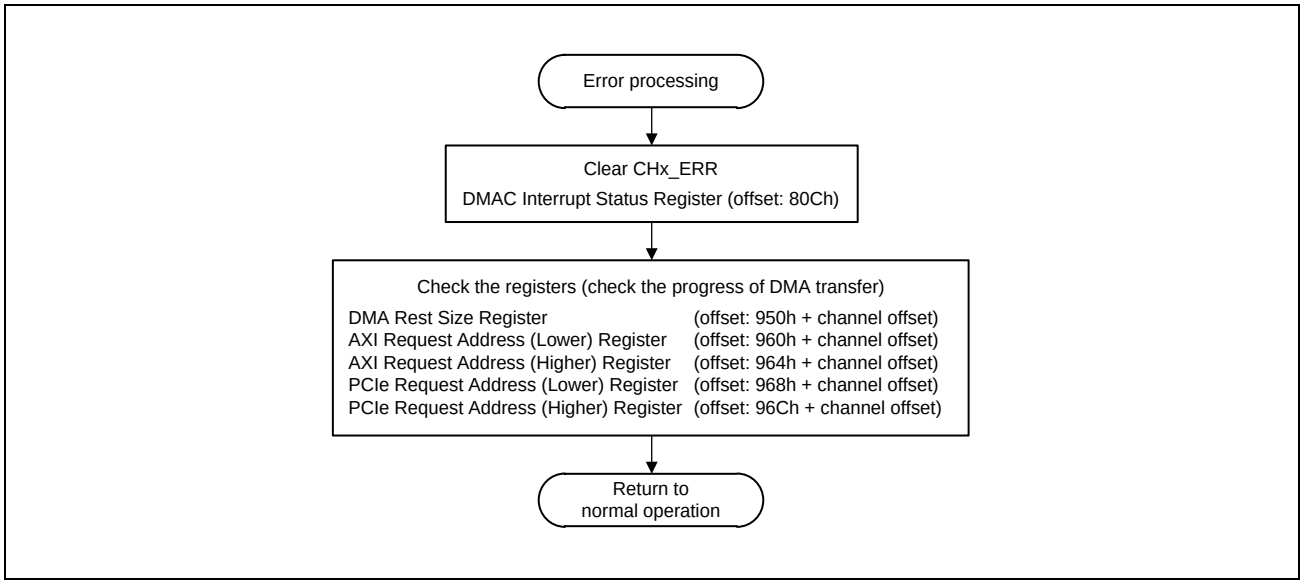


(c) Error Processing

If an error is detected in the AXI bus or a PCIe link while DMA transfer is in progress, the DMA_INT interrupt signal is asserted.

Check the channel where the error was found (CHx_ERR = 1b) by using the DMAC Interrupt Status Register (offset: 80Ch) and also the type of error by using the DMAC Error Status Register (offset: 978h + channel offset) as required.

Clear the CHx_ERR bit that is currently set in the DMAC interrupt status register (offset: 80Ch) by writing 0b to it and check the address to which transfer was most recently completed before it was stopped and the remaining number of bytes for transfer by reading the DMA Status registers (offset: 950h + channel offset to 978h + channel offset) for the given channel.



6.6.5.4.2 Descriptor-Type Transfer

Consecutive DMA transfer is achieved by sequentially reading descriptors which indicate parameters of DMA transfer. Descriptors are allocated to the AXI memory space and the descriptor lists indicate the addresses where the descriptors start. This DMAC has a queue for storing multiple descriptor lists and these descriptor lists are written by software. The first list loaded in the queue is executed after being moved to become the descriptor list for execution following the detection of the condition for starting DMA transfer (the DMAC retains up to nine lists, including the one currently being executed).

Descriptors can have a chained configuration; execution of a descriptor list ends on detection of the last descriptor, and if the queue has a next list, execution of the next descriptor follows.

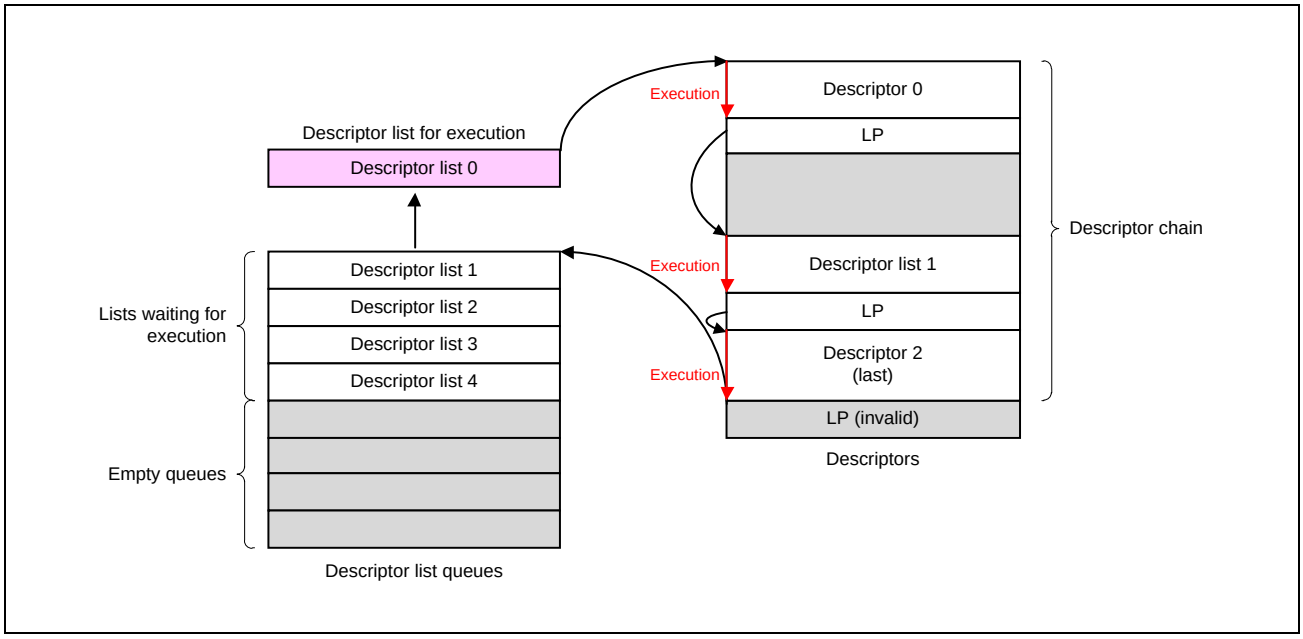


Figure 6.6-2 Descriptor Chain

The table below lists the DMAC registers for execution of descriptor-type DMA transfer.

Table 6.6-119 Registers Related to Descriptor-Type DMA Transfer

Common Control	
800h	DMAC Control Register
808h	DMAC Interrupt Enable Register
80Ch	DMAC Interrupt Status Register
Channel Control	
900h + channel offset	DMA Channel Control Register
908h + channel offset	Descriptor Start Address (Lower) Register m
90Ch + channel offset	Descriptor Start Address (Higher) Register m
910h + channel offset	QUE Entry Register
DMA Setting	
920h + channel offset	DMA Descriptor Control (Descriptor 00h) Register m
924h + channel offset	DMA Transaction Control (Descriptor 04h) Register m
928h + channel offset	DMA Size (Descriptor 08h) Register m
930h + channel offset	DMA Source Lower Address (Descriptor 10h) Register m
934h + channel offset	DMA Source Higher Address (Descriptor 14h) Register m
938h + channel offset	DMA Destination Lower Address (Descriptor 18h) Register m
93Ch + channel offset	DMA Destination Higher Address (Descriptor 1Ch) Register m
940h + channel offset	DMA Descriptor Lower Link Pointer (Descriptor 20h) Register m
944h + channel offset	DMA Descriptor Higher Link Pointer (Descriptor 24h) Register m
DMA Status	
950h + channel offset	DMA Rest Size Register m
960h + channel offset	AXI Request Address (Lower) Register m
964h + channel offset	AXI Request Address (Higher) Register m
968h + channel offset	PCIe Request Address (Lower) Register m
96Ch + channel offset	PCIe Request Address (Higher) Register m
970h + channel offset	QUE Status Register m
978h + channel offset	DMAC Error Status Register m

(1) Descriptor List Queues

The DMAC has queues for storing descriptor lists for each of the channels. The number of FIFO queue stages is eight, so it can hold up to eight descriptor list entries (not including the one currently being executed).

Software places descriptor list entries in the queue. The descriptor lists to be entered are set by the QUE_ENTRY register.

Since the target registers are also accessible in 8-bit units, making the settings in byte units through multiple accesses is also possible. A descriptor list entry is placed in the queue at the time when the QUE_ENTRY (higher) bits [31:24] are written. When making the settings and entering lists in the queue, take care with the order of register access (values written to the QUE_ENTRY (higher) bits [31:26] will not affect the register values and settings because they are read-only bits).

When the DMAC detects the presence of a list which has been entered in the queue, the first list entry is moved to become the pointer to the next descriptor for execution and DMA transfer starts.

(2) Descriptor List Format

The table below is the format of a descriptor list.

Table 6.6-120 Descriptor List Format

Offset	Byte 3			Byte 2	Byte 1	Byte 0
00h	DSA					
04h	Reserved	EI	LS	Reserved	LABEL	

Field Name	Description
DSA[31:0]	Descriptor start address for DMA. This field indicates the address where the first descriptor to be executed is stored. Since the setting is for 8-byte alignment and allocation of a single descriptor (00h to 1Ch) to straddle a 4-K boundary is prohibited, the 5 lower-order bits [4:0] are fixed to 0b.
EI	End Interrupt bit. This bit indicates whether an interrupt (Interrupt Status CHx_END of the DMAC Interrupt Status Register (offset: 80Ch)) is or is not conveyed when processing of this descriptor list is completed. 1b: The interrupt is conveyed. 0b: The interrupt is not conveyed.
LS	List Stop bit. This bit indicates whether DMA transfer is or is not to be stopped on completion of processing for this descriptor list entry. 1b: Stopped 0b: Not stopped On completion of the list in which the setting of this bit is 1b, the QUE_EN bit of the DMAC control register is cleared.
LABEL[15:0]	This field represents the label of the list. There are no special rules for the setting procedures. The value can be set as desired.

Operations by the settings of the EI and LS bits following the completion of the descriptor list are as follows.

Table 6.6-121 Details of the EI/LS Bits

EI	LS	Description
0b	0b	No interrupt. DMA transfer is not stopped. If the queue has a next list, execution of the next descriptor follows.
0b	1b	No interrupt. DMA transfer is stopped (QUE_EN is cleared).
1b	0b	Interrupt (DMAC Interrupt Status Register (Offset: 80Ch), Interrupt Status CHx_END) is present. DMA transfer is not stopped. If the queue has a next list, execution of the next descriptor follows.
1b	1b	Interrupt (DMAC Interrupt Status Register (Offset: 80Ch), Interrupt Status CHx_END) is present. DMA transfer is stopped (QUE_EN is cleared).

(3) Descriptor Format

The table below is the format of descriptors.

Table 6.6-122 Descriptor List Format

Offset	Byte 3								Byte 2								Byte 1								Byte 0							
00h	DSCFM			—	WBD	LE	LV		D	—	—	—	—	—	—	—	STS (not used)															
04h	SA																															
08h	DA																															
0Ch	SIZE																															
10h	PUA																															
14h	—	—	—	—	—	—	—	—	CCH_L				CCH_D				—	TC			—	—	ATB		—	FUNC				—	—	—	DIR
18h	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
1Ch	LP																															

Field Name	Description
DSCFM[3:0]	The Descriptor Format field specifies the format of descriptors. This DMAC only supports 0001b. Do not set any other value.
WBD	Write Back Disable bit Indicates whether the DMAC writes 0b back to the LV bit when DMA transfer specified by a single descriptor is completed. 1b: The LV bit is not written back. 0b: The LV bit is written back.
LE	List End bit Indicates the end of the current descriptor chain. 1b: The current descriptor is the last of the chain. 0b: The current descriptor is not the last of the chain.
LV	Link Valid bit Indicates that the descriptor is valid. When DMA transfer is completed, the DMAC writes 0b back to this bit. When DMA transfer ends due to an error, write-back does not proceed. 1b: The descriptor is valid (DMA transfer specified by the descriptor is not completed). 0b: The descriptor is not valid (DMA transfer specified by the descriptor is completed).
D	Descriptor error bit Indicates a descriptor access error. If LV = 0 when the descriptor is read, the DMAC writes 1b back to the LV bit. When a descriptor error occurs, if the setting of the LE bit is 0 (indicating that the current descriptor is not the last of the chain), the DMAC continues transfer in accord with the descriptor the LP bits indicate. 1b: A descriptor access error occurred. 0b: No error
STS[15:0]	This field has no effect when DSCFM = 0001b. The DMAC does not use this field.
SA[31:0]	The Source Address field indicates the address of the source data for transfer. Since the setting is for 8-byte alignment, do not set a value other than the 3 lower-order bits ([2:0]).
DA[31:0]	The Destination Address field indicates the destination address. Since the setting is for 8-byte alignment, do not set a value other than the 3 lower-order bits ([2:0]).
SIZE[31:0]	This field indicates the number of bytes for transfer. Since the setting is for 8-byte alignment, do not set a value other than the 3 lower-order bits ([2:0]).
PUA[63:32]	The PCIe Higher Address field indicates the higher-order part ([63:32]) of the address in the PCL2 memory space. In the case of DMA transfer from PCIe to AXI (DIR = 0b), this field indicates the source address; in the case of DMA transfer from AXI to PCIe (DIR = 1b), it indicates the destination address.

Field Name	Description
CCH_L[3:0]	This field indicates the value of A*CACHE[3:0] to be issued through the AXI. The value of CCH_L[3:0] is output when an AXI request including the last byte is issued in transfer specified in the SIZE bits. Bit [3]: Write allocation Bit [2]: Read allocation Bit [1]: Cache enabled Bit [0]: Buffer enabled The recommend value is 0000b regardless of whether DIR is 0 or 1.
CCH_D[3:0]	This field indicates the value of A*CACHE[3:0] to be issued through the AXI. The value of CCH_D[3:0] is output when an AXI request other than the condition for the output of CCH_L[3:0] is issued. Bit [3]: Write allocation Bit [2]: Read allocation Bit [1]: Cache enabled Bit [0]: Buffer enabled The recommended value is 0001b when DIR = 0b (PCIe → AXI) and 0000b when DIR = 1b (AXI → PCIe).
TC[2:0]	This field specifies the value of the traffic class of the request to be issued through the PCIe interface. The value must be fixed to 000b.
ATB[1:0]	This field indicates the value of the attribute to be issued through the PCIe interface. Bit [1]: Relaxed ordering Bit [0]: No-snoop If neither relaxed ordering nor no-snoop is used, this field should be set to 00b (recommended).
FUNC[2:0]	This field specifies the function number of the request to be issued through the PCIe interface.
DIR	This bit indicates the direction of data transfer. 1b: AXI → PCIe 0b: PCIe → AXI
LP[31:0]	This field indicates the address where a next descriptor is stored. Since the setting is for 8-byte alignment, do not set a value other than the 3 lower-order bits ([2:0]).

The conditions where the descriptor is written back after the descriptor is read, the timings, and the corresponding bits are listed below.

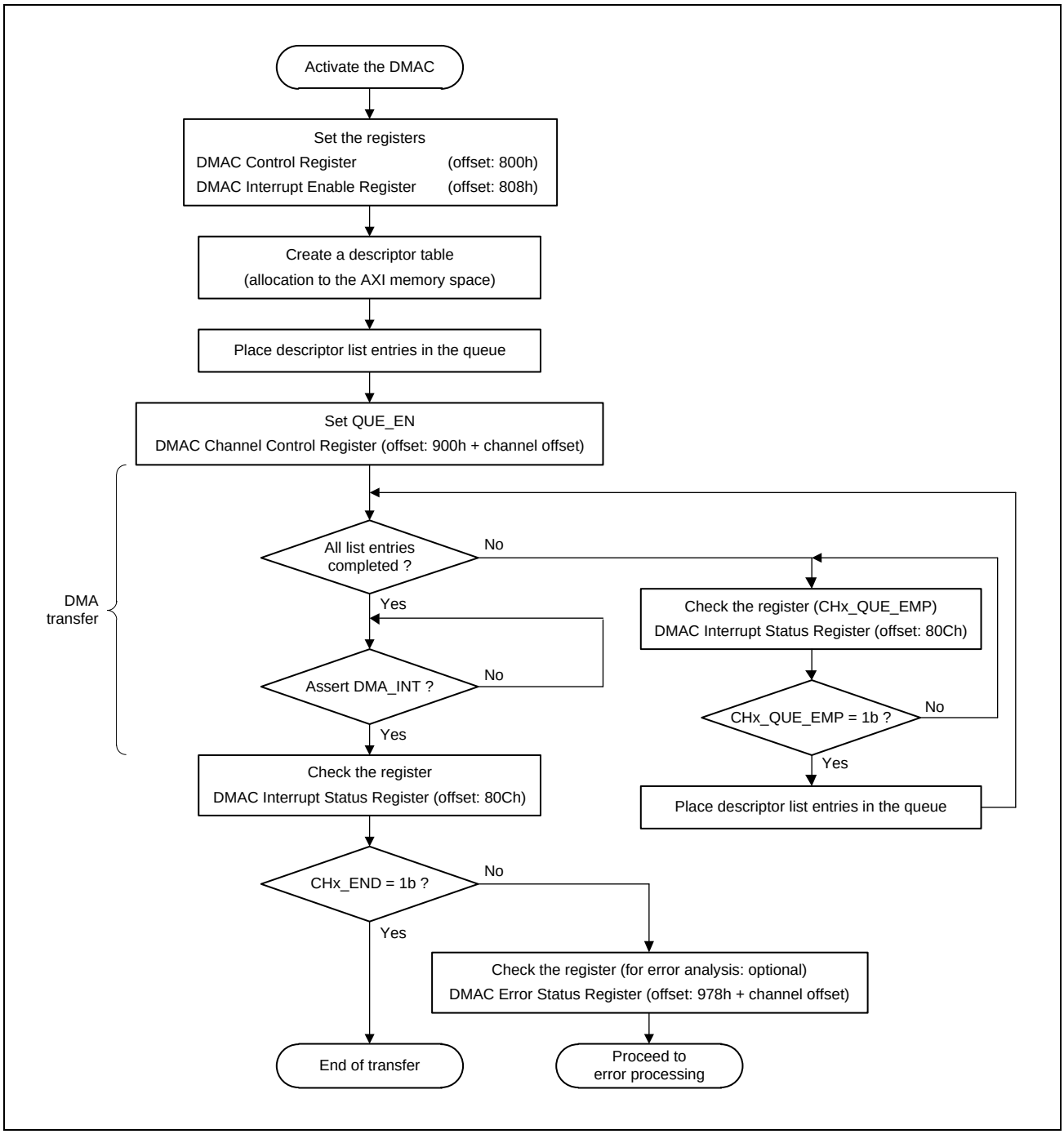
Condition	Timing	Corresponding Bit and Value
WBD = 0b && LV = 1b	After DMA transfer specified by the descriptor is completed	LV = 0b
WBD = (1b or 0b) && LV = 0b	After the descriptor is read (DMA transfer does not proceed)	D = 1b
WBD = 1b && LV = 1b	Write-back does not proceed.	—

(4) Flow of Operations

The following describes the procedures for settings to activate and stop the DMAC in the case of descriptor-type transfer.

(a) Activation and Normal Operation

Before activating the DMAC, set the PCIe and AXI windows. After that, set the registers of the DMAC to start the DMAC.

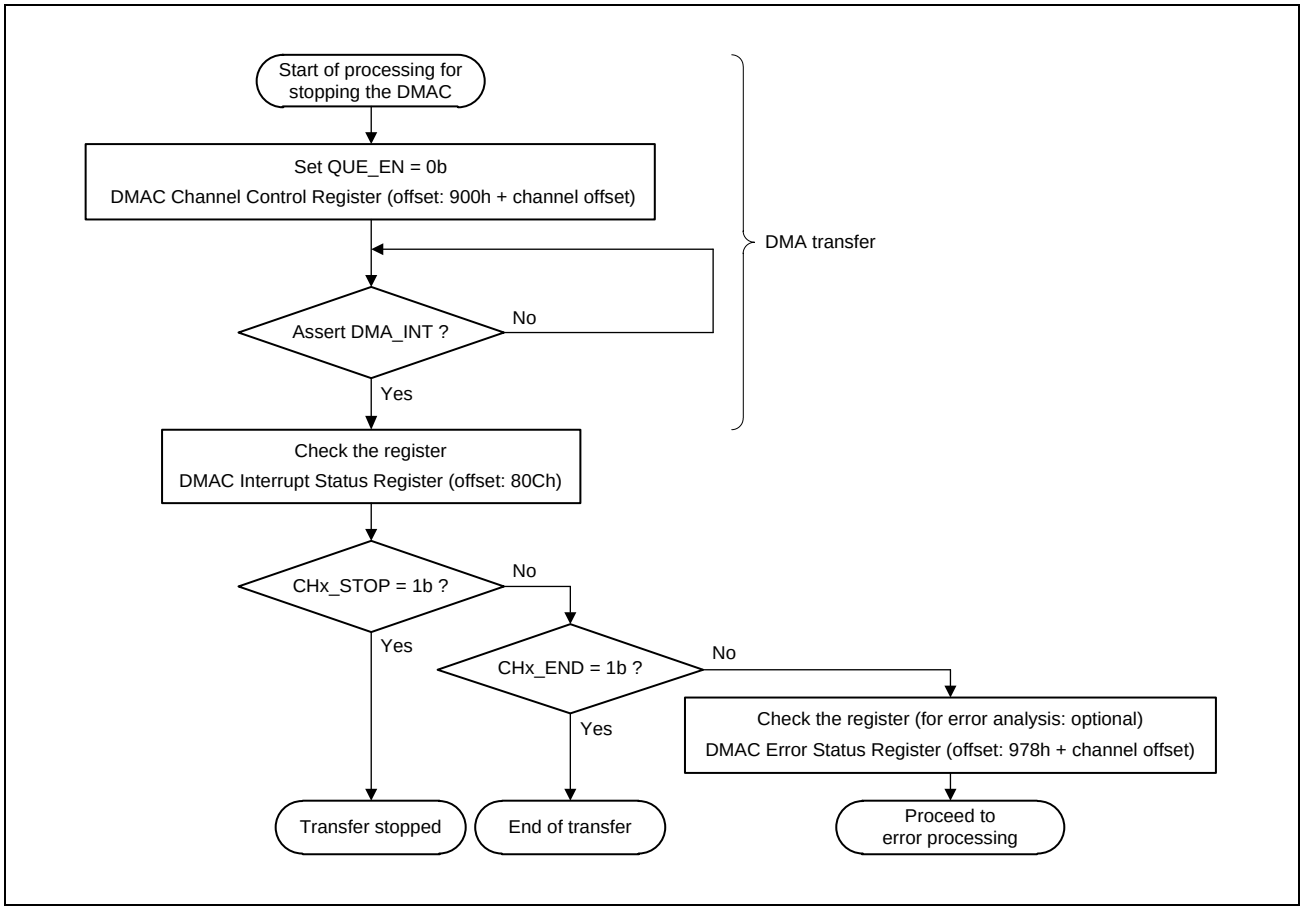


(b) Stopping DMA Transfer

The following describes the procedure for settings to stop DMA transfer under software control.

DMA transfer can be stopped by writing 0b to the QUE_EN (queue enable) bit (bit [1]) of the DMAC channel control register (offset: 900h + channel offset). Following the detection of QUE_EN being 0b, the unit waits for the completion of any request which has already been issued before asserting the interrupt signal (DMA_INT). After waiting for the assertion of DMA_INT, poll the CHx_STOP bit of the DMA interrupt status register (offset: 80Ch) to check that its value is 1b. This indicates that DMA transfer has been stopped. If the CHx_STOP bit is 0b, check the CHx_END bit for the given channel in the same register. If a last request is still being executed when QUE_EN is set to 0b, the CHx_END bit indicates the completion of DMA transfer as in normal operation.

The address to which transfer was most recently completed before it was stopped and the remaining number of bytes for transfer can be checked by reading the DMA Status registers (offset: 950h + channel offset to 978h + channel offset) for the given channel.

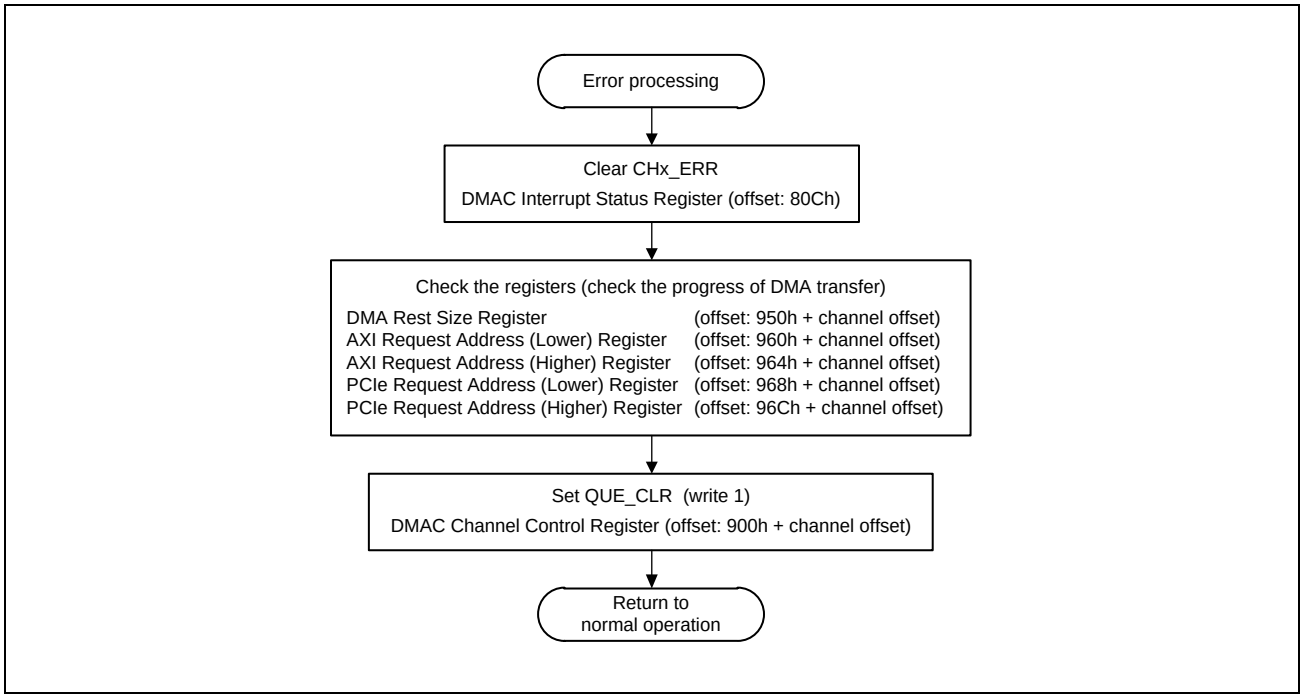


(c) Error Processing

If an error is detected in the AXI bus or a PCIe link while DMA transfer is in progress, the DMA_INT interrupt signal is asserted. Check the channel where the error was found (CHx_ERR = 1b) by using the DMAC interrupt status register and also the type of error by using the DMAC error status register (offset: 978h + channel offset) as required.

Clear the CHx_ERR bit that is currently set in the DMAC interrupt status register (offset: 80Ch) to 0b by writing to it. The address to which transfer was most recently completed before it was stopped and the remaining number of bytes for transfer can be checked by reading the DMA Status registers (offset: 950h + channel offset to 978h + channel offset) for the given channel.

Based on that information, setting the registers as described in **6.6.5.4.2(4)(a) Activation and Normal Operation** and create a descriptor table again. At this time, write 1b to the QUE_CLR bit (bit [8]) of the DMAC channel control register (offset: 900h + channel offset). Writing 1b to it leads to clearing of the queue. The descriptor lists registered in the queue (those waiting for execution and the one currently being executed) are all cleared.



6.6.5.4.3 Method of Transfer

This section describes the issuing of requests to the AXI or PCIe and data transfer in the case of DMA transfer. In either direction of transfer (from AXI to PCIe or vice versa), a request is issued to the reading side. After confirming the completion of preparation (reception) of the data to be read, a request is issued to the writing side for data transfer.

(1) DMA Transfer from PCIe to AXI

An MRd is issued to the PCIe module and an AXI write request is transferred following the reception of CplD from the PCIe. MRd requests corresponding to the number of outstanding transfers can be issued regardless of the reception of CplD. The received CplD (read data) is stored in the data buffer (RAM) once and then transferred by the DMAC to the AXI.

(2) DMA Transfer from AXI to PCIe

After the DMAC is activated, a read request (address channel) is issued to the AXI. The reception of data for reading from the given AXI slave starts and an MWr is issued to the PCIe. Data are repeatedly transferred until the completion of DMA data transfer corresponding to the amount of data set in the DMA Size bits.

6.6.5.4.4 Inter-Channel Arbitration

The DMAC arbitrates the following three types of request between channels.

1) PCIe Read Request
2) AXI Read Request
3) AXI Write Request

PCIe write requests do not require arbitration because data read over the AXI are returned one by one.

In the case of arbitration on the AXI side, not only requests for DMA data, but also reading and writing of descriptor data are subject to arbitration (each channel has a single source of requests for arbitration since a channel will not output a descriptor and request for data at the same time).

(1) Method of Arbitration

An MRd is issued to the PCIe module and an AXI write request is transferred following the reception of CplD from the PCIe. MRd requests corresponding to the number of outstanding transfers (eight) can be issued regardless of the reception of CplD. The received CplD (read data) is stored in the data buffer (RAM) once and then transferred by the DMAC to the AXI.

Arbitration for each type of request 1) to 3) above proceeds in round-robin fashion. A channel for which execution has just been completed is given the lowest priority and each channel which had a lower priority than that channel is raised by one rank in the order of priority.

The initial order of priority is ch. 0 > ch. 1 > ch. 2 > ch. 3 > ch. 4 > ch. 5 > ch. 6 > ch. 7.

The order of priority is changed every time a single request is completed.

There are two types of request for the AXI, requests for DMA data and requests for descriptors, but they are not distinguished for purposes of arbitration.

6.6.5.4.5 DMA Completed Interrupt

The DMAC has interrupt functions. It supports two types of interrupt source: one set is for normal operation and the other is for errors. For details, see **6.6.5.6.5 DMA Interrupt**.

6.6.5.5 Reception of PCIe Commands

6.6.5.5.1 Reception of MSIs (Root Complex)

When an MSI is received (see the judgment conditions below), it is forwarded to AXI as a write transaction and reflected in the MSI reception status (see the description of [Status]). An interrupt signal (INTMSI_RC) is asserted when the MSI reception interrupt generation condition is met (see the description of [Interrupt conditions]). The software identifies an MSI by the interrupt factor. If Message Data Enable=0, the factor can be identified by reading the data on the memory.

[Related registers]

- MSI Receive Window Address (Lower) Registers (Offset: 100h): Message Data Enable
- PCI INTx Receive Interrupt Enable Registers (Offset: 110h): MSI Receive Interrupt Enable
- PCI INTx Receive Interrupt Status Registers (Offset:114h): MSI Receive Interrupt Status
- MSI Receive Mast n Registers (Offset: 6x8h): MSI Mask
- MSI receive Status n Registers (Offset: 6xCh): MSI Status

The conditions for judging a request to be an MSI:

When all of the following conditions are satisfied, a request is judged to be an MSI.

(1) The request is a memory write request.

(2) The request is entered for any area within an AXI window.

(3) The address of an MWr from the PCIe interface must be within the area set by the MSI Reception Window Address register and the MSI Reception Window Mask register.
If a memory read request is received in the MSI reception area, the request is not judged to be an MSI so an interrupt is not generated.

(4) The length of memory write requests is 1 dword

(5) Matching of message data (only when Message Data Enable=1)

[Status]

(1) When Message Data Enable = 0
Set MSI Receive Interrupt Status (bit [4])

(2) When Message Data Enable = 1
Set the corresponding Vector bit in MSI Status

[Interrupt condition]

(1) When Message Data Enable = 0
When MSI Receive Interrupt Enable (bit [4]) is set

(2) When Message Data Enable = 1
When bit 0 of vector corresponding to MSI Mask is set

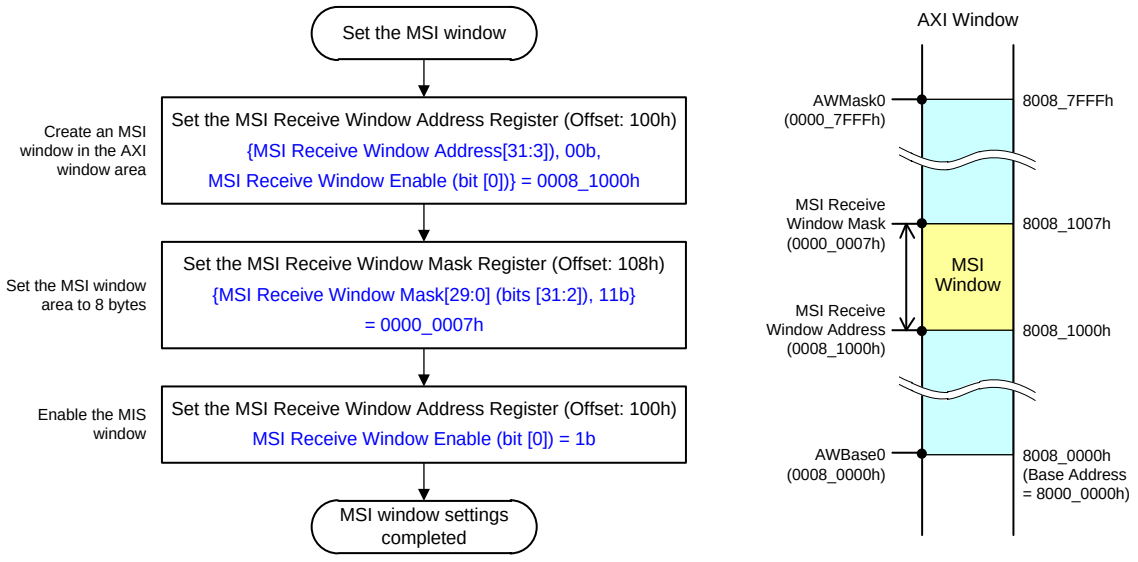
Notes on the MSI:

(1) When a response to the MSI write transaction is returned, this unit judges execution of the MSI to have been completed and asserts an interrupt signal. Depending on the system, however, the MSI memory write transaction may not be completed due to the latency over the AXI to the actual memory even if the interrupt signal is asserted. To avoid this problem, buffering of AXI write transactions must be disabled (MAWCACHE[0] = 0b) at the time an MSI is issued.
For MAWCACHE[3:1], the setting of AWCACHE_L (bits [27:24]) of Mode Set 0 Register is used.

(2) If an MSI is received, the MSI write transaction is executed when MAWID[3:0] = 0001b. This is required for the unit to recognize the reception of the response to the MSI and for assertion of INTMSI_RC.

6.6.5.5.2 Setting the MSI Window

To enable MSI interrupts, the MSI window must be set. **Figure 6.6-3** shows example settings. The MSI window can be allocated within any AXI window.



Note: The values assigned in the figure (values in blue) are examples.

Figure 6.6-3 Settings for the MSI Window

6.6.5.5.3 Reception of an Interrupt in Response to a Message Request (Root Complex)

When Assert_INTx is received in response to an Msg request from the PCIe interface, the corresponding interrupt bit (bits [3] to [0]) is set in the PCI INTx Receive Interrupt Status Register (offset: 114h) and an interrupt (INTx_RC) is asserted. When Deassert_INTx is received in response to a message request, the corresponding interrupt register (bit) is cleared and INTx_RC is de-asserted. The PCI INTx Receive Interrupt Status Register (offset: 114h) can be cleared by writing to it (RW1C) by software. In the PCIe, however, we do not recommend using software to clear an interrupt bit which has been set in response to an Msg in normal operation.

6.6.5.5.4 Reception of Message Requests

When a message is received through the PCIe interface, the relevant information is stored in the following registers.

Table 6.6-123 Message Related Registers

	Related Register
Message Code/Routing	Message Code Register (Offset: 130h)
Message Data	Message Data Register (Offset: 134h)
Message 3rd Header	Message Header 3rdDW Register (Offset: 138h)
Message 4th Header	Message Header 4thDW Register (Offset: 13Ch)

The corresponding bit in the Message Receive Interrupt Status Register is set at the same time as the reception of the following messages, and this is indicated by an interrupt.

Received Message		Corresponding Bit in Message Receive Interrupt Status Register
Type of Message	Message Code	
PM_Active_State_Nak	0001_0100h	Bit [19]
PM_PME	0001_1000h	Bit [18]
PME_Turn_Off	0001_1001h	Bit [17]
PME_TO_Ack	0001_1011h	Bit [16]

In the reception of PM messages other than the above, the relevant information is not written to the registers listed in **Table 6.6-123**.

- Unsupported messages:
  - Unlock
  - Vendor Defined Type 0
  - Vendor Defined Type 1

6.6.5.6 Interrupt

This unit prepares the output of interrupt signals in the ways stated below. The states of all interrupt signals are indicated at a glance in the Interrupt Table Register.

6.6.5.6.1 Error and Event Interrupt Notification

Table 6.6-124 Error/Event Interrupt Outputs

Interrupt Source Name	Active	Attribute	Description
DMA_INT	High	Level	DMA-related event
PCIE_EVT_INT	High	Level	PCIe-related event
MSG_INT	High	Level	Reception of messages
AXI_ERR_INT	High	Level	AXI-related error

Interrupt Output Related Status Registers

DMA_INT: DMAC Interrupt Status Register (Offset: 80Ch)
DMAC Error Status Register (Offset: 978h / + Channel Offset)

PCIE_EVT_INT: PCIe Event Interrupt Status 0 Register (Offset: 204h)
PCIe Event Interrupt Status 1 Register (Offset: 20Ch)
PCIe Event Interrupt Status 2 Register (Offset: 244h)

MSG_INT: Message Receive Interrupt Status Register (Offset: 124h)

AXI_ERR_INT: AXI Master Error Interrupt Status Register (Offset: 214h)
AXI Slave Error Interrupt Status 1 Register (Offset: 224h)

(1) Unit-Internal RAM Parity Error Interrupt Notification

The detection of parity errors in the internal RAM of this unit is indicated by the output of interrupt signals in the ways described below. The type of parity error detecting RAM can be confirmed from the corresponding register bit. A breakdown of the device is a possibility.

PCIE_EVT_INT: PCIe Event Interrupt Status 0 Register (Offset: 204h)
PCIe Event Interrupt Status 1 Register (Offset: 20Ch)

DMA_INT: DMAC Interrupt Status Register (Offset: 80Ch)
DMAC Error Status Register (Offset: 968h / + Channel Offset)

CAUTION

A RAM parity function is not available because it is not implemented in this core.

6.6.5.6.2 INTx/MSI Interrupt Notification (Root Complex)

Table 6.6-125 INTx/MSI Interrupt Outputs

Interrupt Source Name	Active	Attribute	Description
INTA_RC	High	Level	Set in response to the reception of an assert INTA message and cleared in response to the reception of a deassert INTA message.
INTB_RC	High	Level	Set in response to the reception of an assert INTB message and cleared in response to the reception of a deassert INTB message.
INTC_RC	High	Level	Set in response to the reception of an assert INTC message and cleared in response to the reception of a deassert INTC message.
INTD_RC	High	Level	Set in response to the reception of an assert INTD message and cleared in response to the reception of a deassert INTD message.
INTMSI_RC	High	Level	Set when a memory write request comes from the PCIe to the area set in the MSI reception window.
INT_LINK_BANDWIDTH	High	Level	Set when the link width has been changed.
INT_PM_PME	High	Level	Set in response to the reception of PME event notification (PM_PME message).
INT_SERR_COR/ INT_SERR_NONFATAL/ INT_SERR_FATAL	High	Level	Set in response to the reception of a correctable error message, non-fatal error message, or fatal error message.

6.6.5.6.3 Issuing INTx/MSI Interrupts (Endpoint)

An endpoint indicates two types of interrupt to the root complex.

(1) Legacy interrupt (Assert INTx Message/Deassert INTx Message)

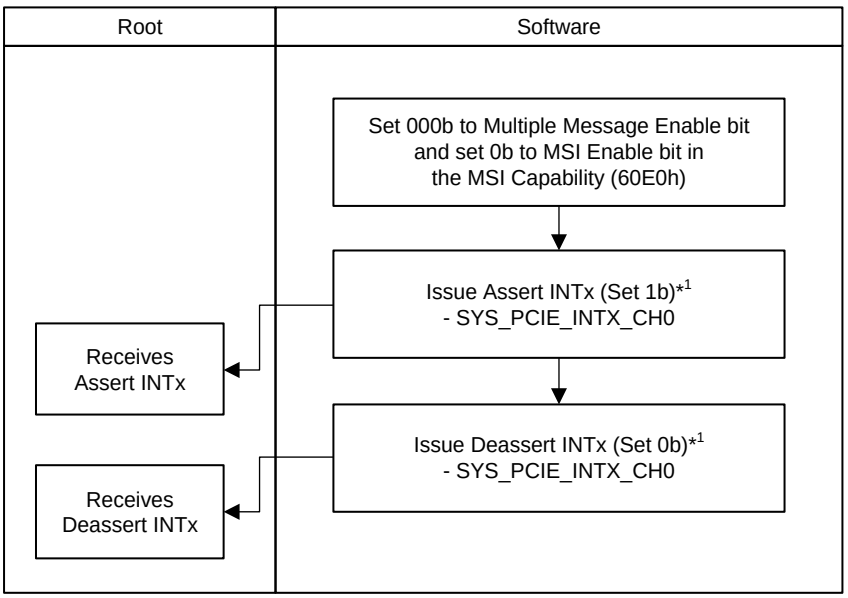
- Issuing INTx Msg specified by an interrupt register by controlling an interrupt input signal (INTX_EP_F*) (recommended)
- Issuing Assert INTx and Deassert INTx messages through the issuing of special requests.

(2) MSI

- Issuing an MSI by generating a memory write transaction directly.

The MSI Enable bit in the MSI Capability Register (PCIe configuration Register) can be used to switch exclusive operation of two interrupts on or off.

The legacy interrupt issuing flow is shown in **Figure 6.6-4** and the MSI issuing flow is shown in **Figure 6.6-5**.



Note 1. For details, refer to **(1) Issuing a Legacy PCI INTx Emulation Interrupt (when MSI Enable = 0b)**.

Figure 6.6-4 Legacy Interrupt Issuing Flow

To issue a legacy interrupt, the Interrupt Disable bit in the Command and Status Register (PCIe Configuration Register) must be cleared.

Furthermore, if writing proceeds in a power state other than D0 or a PME_TURN_OFF message is received, or if FLR_REQ is received, Deassert INTx Msg is automatically generated in response to the corresponding function asserting an INTx, and all active interrupts are withdrawn without waiting for the PME_Turn_Off Message reception/Non-D0 State transition request reception acknowledge/FLR_RESET response. However, when multiple functions share an INTx, the logical OR of the signals from the sharing functions is taken, a Deassert INTx Msg is generated, and the interrupts are withdrawn. If INTX_EP_F0/F1 is being asserted at the time of returning to the D0 state

once more or an FLR sequence is completed (indicated by the de-assertion of FLR_REQ), Assert INTx is generated again in response.

Table 6.6-126 INTx/MSI Interrupt Inputs

Interrupt Factor Name	Active	Attribute	Description
INTX_EP_F0	High	Level	This is the trigger for Assert INTx and Deassert INTx messages.
INTX_EP_F1	High	Level	

[Related registers]

PCI INTx Out Status Register (Offset: 118h)

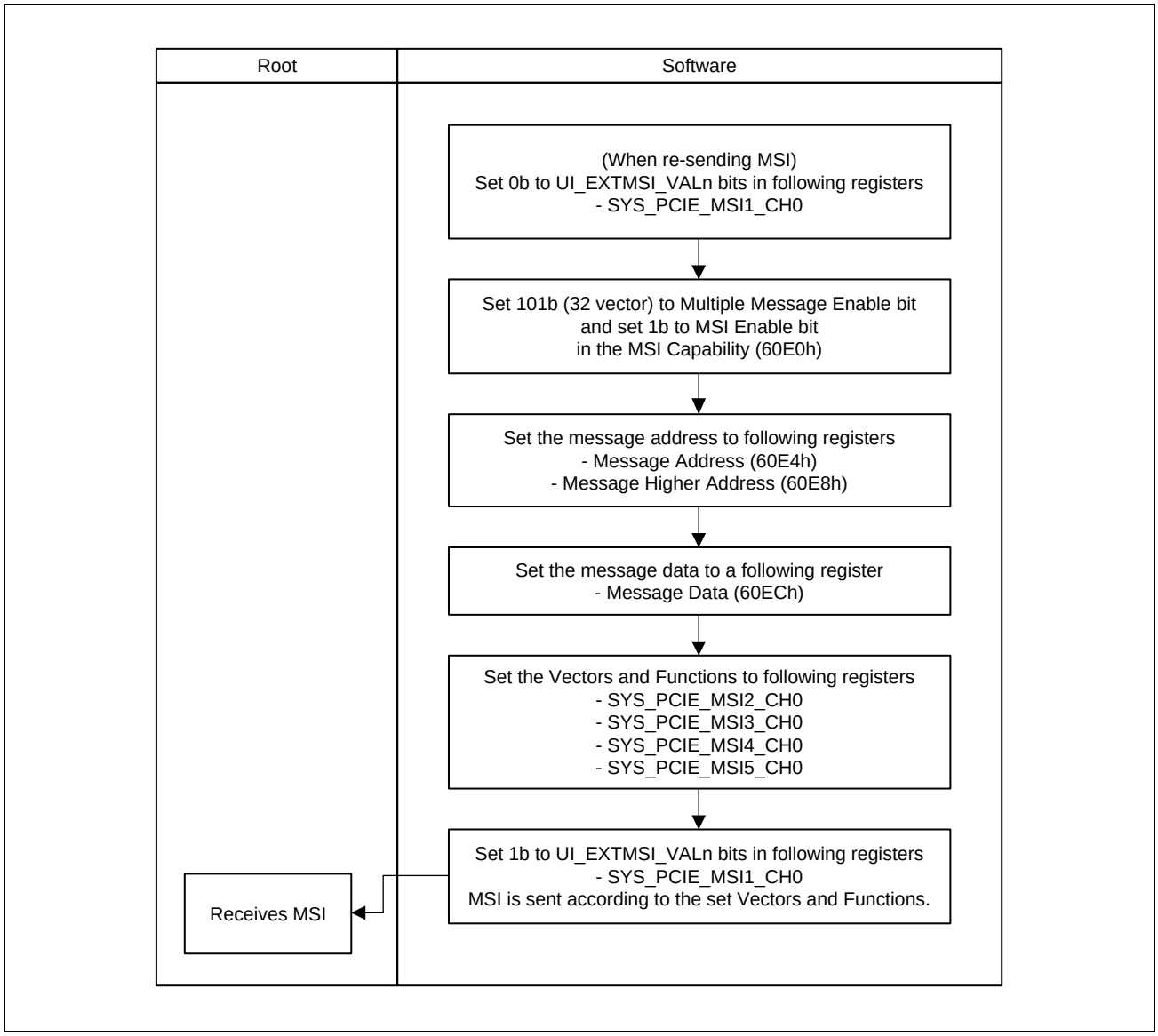


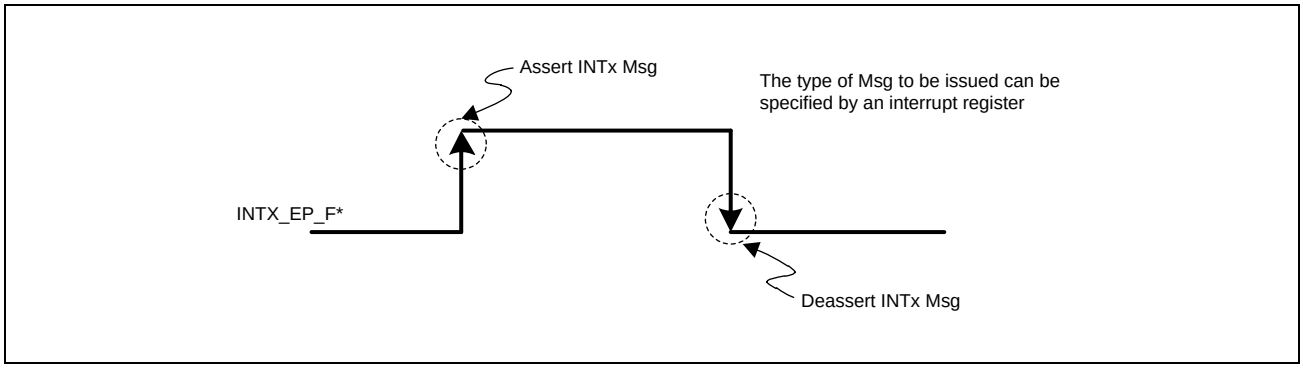
Figure 6.6-5 MSI Issuing Flow

(1) Issuing a Legacy PCI INTx Emulation Interrupt (when MSI Enable = 0b)

A legacy PCI INTx emulation interrupt is issued by controlling INTX_EP_F*.

The type of Assert INTx and Deassert INTx Msg to be issued by controlling an interrupt signal is specified by an interrupt register provided for each of the functions. Assert INTx Msg is issued by setting INTX_EP_F* to 1, and Deassert INTx Msg is issued by clearing INTX_EP_F* to 0.

Make sure that an interrupt signal successfully arrives on the receiving side following the de-assertion of an interrupt signal. After completion of processing for an interrupt from a given source, negating the signal is recommended.



Changing the allocation of the legacy interrupt lines is possible by changing the interrupt register setting of the configuration register. If the same interrupt is shared by multiple functions, the logical OR of INTX_EP_F0 and INTX_EP_F1 is taken to issue an interrupt.

(2) MSI Transmission by Interrupt Input (when MSI Enable = 1b)

Five inputs are used in specifying the interrupt (MSI). In the same way, each interrupt indication has its corresponding specified interrupt vector.

Table 6.6-127 External Interrupt Notification

Interrupt Notification Name	Input/ Output	Note
UI_EXTMSI_VAL0/1/2/3/4	Input	Interrupt notification
UI_EXTMSI_VEC0/1/2/3/4 [4:0]	Input	Interrupt vector specification
UI_EXTMSI_FUNC0/1/2/3/4 [2:0]	Input	Interrupt function number specification

Note: Multiple Message Enable in the Base Spec.: Implemented as a specification that supports 32 (5 bits).
Multiple Message Enable on the RC side: Support for 32 is also required.

CAUTION

Take care that the specified vector value does not impose a related restriction on the Multiple Message Enable register. For example, if the number set for Multiple Message Enable is 4, the two lower-order bits within the value set for the Message Data are overwritten by the value specified for the UI_EXTMSI_VEC[*] bits, and at that time, specify the value 0 for the higher-order bits that are not overwritten by UI_EXTMSI_VEC[*] bits. Convey interrupts in the state where the vector value and function number have been specified.

(3) Flow of Asserting MSI Enable

The following is the flow of asserting the MSI Enable field (bit 16) of the MSI Capability register.

1. Negate all INTx_EP_F0/F1 (processing by the endpoint).
2. Wait until the Interrupt Status field (bit 19) of the Command and Status register becomes 0b (confirm that the device is not in the waiting state for INTx interrupt messages).
   (Processing by the root complex: CfgRd)
3. Write 1b to the MSI Enable field (bit 16) of the MSI Capability register.
   (Processing by the root complex: CfgWr)

6.6.5.6.4 Root Complex Interrupt Notification

When this unit is in the root complex configuration, it has the following interrupt sources.

- INT_LINK_BAND_WIDTH
- INT_PM_PME
- INT_SERR
- INT_SERR_COR
- INT_SERR_FATAL
- INT_SERR_NONFATAL

(1) Interrupt due to the Change to the Link Bandwidth (INT_LINK_BAND_WIDTH)

This interrupt source is conveyed in response to the change to the bandwidth in link negotiation of PCI Express.

Note that this interrupt source is enabled when the root complex in use and it is fixed to “0: Low” when the endpoint is in use.

This interrupt source is only enabled when the Link Bandwidth Notification Capability bit (bit 21) of the Link Capabilities register (offset: 60Ch) is set to 1b. This interrupt signal can also be disabled by setting the Link Bandwidth Notification Capability bit to 0b even in operation as the root complex.

If the Link Bandwidth Notification Capability bit (bit 21) is set to 1b, the Link Bandwidth Management Status bit (bit 30) and the Link Autonomous Bandwidth Status bit (bit 31) in the Link Status register (offset 70h) serve as a source of the interrupt. These bits are set to 1b when the bandwidth is changed.

Whether to enable this interrupt source corresponds to the respective states of the Link Bandwidth Management Interrupt Enable bit (bit 10) and the Link Autonomous Bandwidth Interrupt Enable (bit 11) of the Link Control register (offset: 70h).

[Related Registers]

Offset	Bit	Description
PCI Express Capability Structure: Link Control / Status		
06Ch	21	Link Bandwidth Notification Capability
070h	10	Link Bandwidth Management Interrupt Enable
	11	Link Autonomous Bandwidth Interrupt Enable
	30	Link Bandwidth Management Status
	31	Link Autonomous Bandwidth Status

(2) PM-PME Reception Interrupt (INT_PM_PME)

This interrupt source is conveyed when the notification of a PME event (PM_PME message) is received from the other-party device.

Note that this interrupt source is enabled when the root complex in use and it is fixed to “0: Low” when the endpoint is in use.

This interrupt source is only conveyed when the PME Interrupt Enable bit (bit 3) of the Root Control/Capabilities Register (offset: 7Ch) is set to 1b. This interrupt source is not conveyed when the PME Interrupt Enable bit (bit 3) is set to 0b.

Note that the reception state of the received PM_PME message and the ID information, etc. are stored in the Root Status Register (offset: 80h) regardless of the setting of the PME Interrupt Enable bit mentioned above.

[Related Registers]

Offset	Bit	Description
PCI Express Capability Structure: Root Control Capabilities		
07Ch	3	PME Interrupt Enable
PCI Express Capability Structure: Root Status		
080h	15:0	PME Requester ID
	16	PME Status
	17	PME Pending

(3) System Error Interrupt (INT_SERR_xxx)

This interrupt is conveyed when a correctable error message, non-fatal error message, or fatal error message is received.

INT_SERR: The system error notification mentioned in the Base Spec. is obtained as the logical OR of the following three signals.

INT_SERR_COR: System Error on Correctable Error

INT_SERR_FATAL: System Error on Fatal Error

INT_SERR_NONFATAL: System Error on Non-Fatal Error

These interrupt sources are enabled when the root complex is in use and these are fixed to “0: Low” when the endpoint is in use.

(a) Correctable Error Interrupt (INT_SERR_COR)

To proceed with interrupt control due to a correctable error, set the interrupt enable bits of the following registers to 1b.

No.	Offset	Bit	Description
1	PCI Compatible Configuration: Bridge Control and Interrupt		
	03Ch	17	SERR# Enable
2	PCI Express Capability Structure: Device Control		
	068h	0	Correctable Error Reporting Enable
3	Advanced Error Reporting (AER) Capability: Root Error Command		
	12Ch	0	Correctable Error Reporting Enable

<Interrupt Source>

(1) ERR_COR Message Reception

If an ERR_COR message is received from the other-party device (endpoint), the INT_SERR_COR interrupt will be asserted. The following status register will be set at the same time.

Offset	Bit	Description
Advanced Error Reporting (AER) Capability: Root Error Status		
130h	0	ERR_COR Received

Note: If the Enable bits in no. 1, 2, and 3 are not set, the interrupt source will not be generated.
If the Enable bits in no. 1 and 2 are not set, writing will be masked.

(2) Correctable Error Detection

Any of the following correctable errors being detected within the unit will be written to the corresponding status register.

Error	Offset	Bit	Description
	Advanced Error Reporting (AER) Capability: Correctable Error Status		
8B10B Decode Error 8B10B RD Error	110h	0	Receiver Error Status
Bad TLP		6	Bad TLP Status
Bad DLLP		7	Bad DLLP Status
REPLAY_NUM Roll over		8	REPLAY_NUM Rollover Status
Replay Timer Timeout		12	Replay Timer Timeout Status
Advisory Non-Fatal Error		13	Advisory Non-Fatal Error Status

Note: If the Mask bit (Correctable Error Mask Register: Offset 114h) corresponding to each bit is set (masked), the corresponding error will not be detected and the interrupt source will not be generated.

At the same time, the INT_SERR_COR interrupt will be generated and the following status registers will be set to 1b.

Offset	Bit	Description
PCI Express Capability Structure: Device Status		
068h	16	Correctable Error Detect
Advanced Error Reporting (AER) Capability : Root Error Status		
130h	0	ERR_COR Received

Note: If the Enable bit in no. 2 is not set, the interrupt source will not be generated.
If the Enable bit in no. 2 is not set, writing to the Root Error Status Register will be masked.

If the Correctable Error Reporting Enable bit listed in no. 3 is not set, the source due to (1) and (2) stated above will be generated and INT_SERR_COR will not be asserted even if writing to the Root Error Register has proceeded.

(b) Fatal Error Interrupt (INT_SERR_FATAL)

To proceed with interrupt control due to a fatal error, set the interrupt enable bits of the following registers to 1b.

No.	Offset	Bit	Description
1	PCI Compatible Configuration: Bridge Control and Interrupt		
	03Ch	17	SERR# Enable
2	PCI Express Capability Structure: Device Control		
	068h	2	Fatal Error Reporting Enable
3	Common Configuration Space: Command and Status		
	004h	8	SERR# Enable
4	Advanced Error Reporting (AER) Capability: Root Error Command		
	12Ch	2	Fatal Error Reporting Enable

<Interrupt Source>

(1) ERR_FATAL Message Reception

If an ERR_FATAL message is received from the other-party device (endpoint), the INT_SERR_FATAL interrupt will be asserted (if the Enable bits in no.1 and 2 or no.3 and 4 are not set, the interrupt source will not be generated). The following status registers will be set at the same time.

Offset	Bit	Description	Remarks
Type1 Configuration Space: Secondary Status			Writing to this status register proceeds at the point a message is received regardless of the settings of the Enable bits.
01Ch	30	Received System Error	
Common Configuration Space: Status Register			If the Enable bits in no. 1 and 2 are not set, writing will be masked.
004h	30	Signaled System Error	
Advanced Error Reporting (AER) Capability: Root Error Status			If the Enable bit in no. 1, 2, or 3 is not set, writing will be masked.
130h	6	Fatal Error Message Received	

(2) Fatal Error Detection

Any of the following fatal errors being detected within the unit will be written to the corresponding status register.

Error	Offset	Bit	Description
	Advanced Error Reporting (AER) Capability: Uncorrectable Error Status		
Data Link Protocol Error	104h	4	Data Link Protocol Error Status
Poisoned TLP		12	Poisoned TLP Status
Completion Timeout		14	Completion Timeout Status
Completer Abort		15	Completer Abort Status
Unexpected Completion		16	Unexpected Completion Status
Receiver Overflow		17	Receiver Overflow Status
Malformed TLP		18	Malformed TLP Status
ECRC Error		19	ECRC Error Status
Unsupported Request		20	Unsupported Request Error Status

Note: If the Mask bit (Uncorrectable Error Mask Register: Offset 108h) corresponding to each bit is set (masked), the corresponding error will not be detected and the interrupt source will not be generated.
Also, the setting of the corresponding Severity bit (Uncorrectable Error Severity: Offset 10Ch) must be 1b (Fatal).

The INT_SERR_FATAL interrupt will be generated at the same time (if the Enable bits in no. 2 or no.3 and 4 are not set, the interrupt source will not be generated). The following status registers will be set at the same time.

Offset	Bit	Description	Remarks
PCI Express Capability Structure: Device Status			
068h	18	Fatal Error Detect	
Common Configuration Space: Status Register			If the Enable bit in no. 3 is not set, writing will be masked.
004h	30	Signaled System Error	
Advanced Error Reporting (AER) Capability: Root Error Status			If the Enable bit in no. 2 or 3 is not set, writing will be masked.
130h	6	Fatal Error Message Received	

If the Fatal Error Reporting Enable bit listed in no. 4 is not set, the source due to (1) and (2) stated above will be generated and INT_SERR_FATAL will not be generated even if writing to the Root Error Status Register has proceeded.

(c) Non-Fatal Error Interrupt (INT_SERR_NONFATAL)

To proceed with interrupt control due to a non-fatal error, set the interrupt enable bits of the following registers to 1b.

No.	Offset	Bit	Description
1	PCI Compatible Configuration: Bridge Control and Interrupt		
	03Ch	17	SERR# Enable
2	PCI Express Capability Structure: Device Control		
	068h	1	Non-Fatal Error Reporting Enable
3	Common Configuration Space: Command and Status		
	004h	8	SERR# Enable
4	Advanced Error Reporting (AER) Capability: Root Error Command		
	12Ch	1	Non-Fatal Error Reporting Enable

<Interrupt Source>

(1) ERR_FATAL Message Reception

If an ERR_NONFATAL message is received from the other-party device (endpoint), the INT_SERR_NONFATAL interrupt will be generated (if the Enable bits in no.1 and 2 or no.3 and 4 are not set, the interrupt source will not be generated). The following status registers will be set at the same time.

Offset	Bit	Description	Remarks
Type1 Configuration Space: Secondary Status			Writing to this status register proceeds at the point a message is received regardless of the settings of the Enable bits.
01Ch	30	Received System Error	
Common Configuration Space: Status Register			If the Enable bits in no. 1 and 3 are not set, writing will be masked.
004h	30	Signaled System Error	
Advanced Error Reporting (AER) Capability: Root Error Status			If the Enable bit in no. 1, 2, or 3 is not set, writing will be masked.
130h	5	Non-Fatal Error Message Received	

(2) Non-Fatal Error Detection

Any of the following non-fatal errors being detected within the unit will be written to the corresponding status register.

Error	Offset	Bit	Description
	Advanced Error Reporting (AER) Capability: Uncorrectable Error Status		
Data Link Protocol Error	104h	4	Data Link Protocol Error Status
Poisoned TLP		12	Poisoned TLP Status
Completion Timeout		14	Completion Timeout Status
Completer Abort		15	Completer Abort Status
Unexpected Completion		16	Unexpected Completion Status
Receiver Overflow		17	Receiver Overflow Status
Malformed TLP		18	Malformed TLP Status
ECRC Error		19	ECRC Error Status
Unsupported Request		20	Unsupported Request Error Status

Note: If the Mask bit (Uncorrectable Error Mask Register: Offset 108h) corresponding to each bit is set (masked), the corresponding error will not be detected and the interrupt source will not be generated.
Also, the setting of the corresponding Severity bit (Uncorrectable Error Severity: Offset 10Ch) must be 0b (Non-fatal).

The INT_SERR_NONFATAL interrupt will be generated at the same time (if the Enable bits in no. 2 or no.3 and 4 are not set, the interrupt source will not be generated). The following status registers will be set at the same time.

Offset	Bit	Description	Remarks
PCI Express Capability Structure: Device Status			—
068h	17	Non-Fatal Error Detect	
Common Configuration Space: Status Register			If the Enable bit in no. 3 is not set, writing will be masked.
004h	30	Signaled System Error	
Advanced Error Reporting (AER) Capability: Root Error Status			If the Enable bit in no. 2 or 3 is not set, writing will be masked.
130h	5	Non-Fatal Error Message Received	

If the Non-Fatal Error Reporting Enable bit listed in no. 4 is not set, the source due to (1) and (2) stated above will be generated and INT_SERR_NONFATAL will not be generated even if writing to the Root Error Status Register has proceeded.

6.6.5.6.5 DMA Interrupt

The DMAC has interrupt functions. It supports two types of interrupt source: one set is for normal operation and the other is for errors. Use the DMAC Interrupt Enable Register to select the interrupt sources and control masking. DMA interrupt notification includes notification by DMA_INT in the direction of the AXI bus (local bus direction) and notification by issuing an MSI to an other-party RC.

On completion of DMA transfer, either conveying a DMA_INT interrupt or transmitting an MSI to the RC, but not both, is selected per-channel by a register setting (transmitting an MSI is prohibited in RC mode).

The DMA_CH*_MSI_EN bits of the DMA Interrupt Vector 0 Register and the DMA Interrupt Vector 1 Register are used to switch the setting (“*” corresponds to the channel number of a given DMA and per-channel control is possible). If an MSI is to be issued, the value of its traffic class (TC) bits is fixed to 0. The vector value can be specified by using the DMA_CH*_vec bits of these registers.

CAUTION

Take care that the specified vector value does not impose a related restriction on the Multiple Message Enable register. For example, if the number set for Multiple Message Enable is 4, the two lower-order bits within the value set for the Message Data are overwritten by the value specified for the DMA_CH*_vec bits, and at that time, specify the value 0b for the higher-order bits that are not overwritten by DMA_CH*_vec bits.

6.6.5.7 Power Management (Root Complex Mode)

6.6.5.7.1 PCI Power Management (PCI-PM)

An endpoint is placed in non-D0 (the D3 hot state) by changing the setting of the Power State field register of the endpoint by the root complex., which causes a transition of the link state of both RC and EP to L1.

This unit does not support the AUX power supply function. D3 cold is not supported. Implementing the L2 function goes to implementation of L2/L3 ready.

(1) Flow of Transition to the L2/L3 Ready State (RC)

< For transition via PCIPM L1 >

1. A root complex (RC) issues a config write to change the D-state of an endpoint (EP) to D3 (D3 hot).
2. The RC is automatically placed in L1 by automatic response of the other-party EC.
3. The RC transmits a PME_Turn_Off message.
4. Confirm the reception of a PME_TO_Ack Message from the EP by reading the value of the PME_TO_Ack Receive Interrupt bit (bit 16) in the Message Receive Interrupt Status Register of the RC.
5. Set UI_ENTER_L2 of PCIe Core Control 1 Registers to 1b.
6. Confirm the transition to the L2/L3 ready state by reading the value of the LTSSM_STATE bits (bits [14:8]) in the PCIe Core Status 1 Register (5 higher-order bits [14:10] = 0111xb: L2).

< For direct transition to the L2/L3 ready state >

Steps 3 to 6 above apply.

(2) Flow of Return from the L2/L3 Ready State (RC)

< Return from the L2/L3 ready state by the RC >

1. Assert the following reset signals.

   RST_B, RST_GP_B, RST_PS_B, RST_CFG_B
2. Deassert the reset signals at a desired time and wait for return to L0 (wait for linkup).
3. If the link is not up, repeat steps 1 and 2.

< Return from the L2/L3 ready state by the EP >

1. Wait for the reception of a beacon by the EP. Read the LINKDN bit (0204h, bit 9) in the PCIe Event Interrupt Status register regularly to check whether ELECTRICAL_IDLE_BROKEN has been generated.
2. Assert the following reset signals.

   RST_B, RST_GP_B, RST_PS_B, RST_CFG_B
3. Deassert the reset signals at a desired time, clear UI_ENTER_L2 of PCIe Core Control 1 Registers to 0b, and wait for it to return to L0 (wait for linkup).
4. If the link is not up, repeat steps 2 and 3.

6.6.5.7.2 Active State Power Management (ASPM)

The ASPM L0s and L1 states can be used by the setting of the Active State PM Control bits (bits [1:0]) in the Link Control/Status Register (PCIe Configuration Register: 6070h).

ASPM L0s: The module is automatically placed in the ASPM L0s state following setting of the Active State PM Control bits to 01b or 11b. When to initiate a transition to this state is determined by judgment of the idle state by the PCIe core.

ASPM L1: An RC is placed in the ASPM L1 state in response a request for transition to the ASPM L1 state from an EP following setting of the Active State PM Control bit to 11b. When to initiate a transition to this state is determined by judgment of the idle state by the PCIe core.
A transition to the ASPM L1 state is enabled by setting the UI_RC_REJECT_ASPM_L1 bit (bit [19]) of the PCIe Core Control 1 Register (offset: 404h) to 0b. To reject a transition to the ASPM L1 state on the RC side, set the UI_RC_REJECT_ASPM_L1 bit to 1b.

Return to the L0 state from the L1 state:

Resuming access from the AXI master side:

Return to the L0 state is initiated in response to the other-party PCIe device resuming access.

Resuming access from the AXI slave side:

The AXI can be returned to the L0 state in response to the AXI starting access to a PCIe device. Access to an AXI configuration register also returns the AXI to the L0 state.

6.6.5.8 Power Management (Endpoint Mode)

6.6.5.8.1 PCI Power Management (PCI-PM)

The endpoint unit can be placed in the D3 hot state by changing the setting of the Power State field register by the root complex. A PME_Turn_Off message can be received even after the transition to the D3 state. The turning-off procedure described below starts in such cases.

D0: This is a full-power state in which no restrictions apply. The link state is basically L0. However, if the ASPM generates a source for a transition, the latter takes priority.

D3hot: The link state basically changes to L1. By changing the value of the Power State bits of the PM Status/Control register in response to the reception of CfgWr, the unit sets D3_EVENT to 1 when a transition to the non-D0 state is requested per function, and issues Deassert_INTx in response to the assertion of any of the INTx signals. The EP should set D3_EVENT_ACK to 1 after preparation for low power consumption. After D3_EVEMT_ACK is set to 1, issuing of the AXI transaction to the PCIe is prohibited.(this does not apply to access to internal registers which include the configuration register).

This unit does not support the AUX power supply function. D3 cold is not supported. Implementing the L2 function goes to implementation of L2/L3 ready.

(1) Flow of Transition to the L2/L3 Ready State (EP)

< For transition via PCIPM L1 >

1. The D-state of an endpoint (EP) is changed to D3 (D3 hot) in response to the reception of a config write from a root complex (RC).
2. Check that D3_EVENT has been set to 1, and clear D3_EVENT_ACK to 0 after setting it to 1.
3. A transition to the L1 state is automatically initiated.
4. The EC receives a PME_Turn_Off message from the RC.
5. Check that TURN_OFF_EVENT has been set to 1, and clear TURN_OFF_EVENT_ACK to 0 after setting it to 1.
6. A transition to the L2/L3 ready state is automatically initiated.

< For direct transition to the L2/L3 ready state >

Steps 4 to 6 above apply.

(2) Flow of Return from the L2/L3 Ready State (EP)

< Return from to the L2/L3 ready state by the RC >

1. Wait for an Electrical Idle exit from the RC.
2. After that, follow the instruction by the RC to return to L0 (the following is an example).
   - Example 2-1. Follow the instruction by the RC to reset the EP (assert RST_B, RST_GP_B, RST_PS_B, and RST_CFG_B).
   - Example 2-2. Follow the instruction by the RC to release the EP from the reset state at a desired time.
   - Example 2-3. Return to L0.

< Return from to the L2/L3 ready state by the EP >

1. Check that the value of the LTSSM_STATE bits (bits [14:8]) in the PCIe Core Status 1 Register is 3Ah.

   Note: As a criterion for judgment at this time, checking the value read from this register requires consecutively checking the above value several times. Since the value of the LTSSM bits is checked by direct reference to the internal circuit state information, an undefined state value may be read depending on the timing of the operation for transition. To make sure that the current state is the L2 idle state, we recommend checking the state by access more than once. Only check all seven bits of LTSSM_STATE (bits [14:8]) of the PCIe Core Status 1 Register at this time in the flow of return.

2. Assert the Power Management Event setting input pin to 1b.
3. Assert RST_B (the EP transmits a beacon).
4. After that, follow the instruction by the RC to return to L0 (the following is an example).
   - Example 4-1. The RC detects the reception of a beacon from the EP and initiates return to L0 on the RC side.
   - Example 4-2. Follow the instruction by the RC to reset the EP (assert RST_B, RST_GP_B, and RST_PS_B, RST_CFG_B).
   - Example 4-3. Follow the instruction by the RC to release the EP from the reset state at a desired time.
   - Example 4-4. Return to L0.

6.6.5.8.2 Active State Power Management (ASPM)

The ASPM L0 and L1 states can be used by setting the Active State PM Control bits (bits [1:0]) of the Link Control/Status Register (PCIe Configuration Register: 6070h).

Permit a transition to ASPM L1 by setting ALLOW_ENTER_L1 to 1b.

The ASPM L1 Idle Time bits of the registers must be set in advance to suit the AXI bus specifications (the range of settings is for from 256 to 65536 cycles of ACLK). If transfer does not proceed for the time set in the ASPM L1 Idle Time bits within the unit, the module is automatically placed in the ASPM L1 state.

ASPM L0s: A transition to the ASPM L0s state is automatically initiated by setting the Active State PM Control bits to 01b or 11b. When to initiate a transition to this state is determined by judgment of the idle state by the PCIe core, so your operation will not be especially required.

A transition to L0s can be controlled by setting the UI_ENTER_TXL0S bit (bit [16]) of the PCIe Core Control 1 Register.

ASPM L1: A transition to the ASPM L1 state can be initiated by setting the Active State PM Control bits to 10b or 11b. When to initiate a transition to this state is determined by judgment of the idle state by the PCIe core. A transition to the L1 state is enabled by setting ALLOW_ENTER_L1 to 1b.

Return from the L1 state to the L0 state

Resuming access from the AXI master side:

Return to the L0 state is initiated in response to the other-party PCIe device resuming access.

Resuming access from the AXI slave side:

The AXI can be returned to the L0 state in response to the AXI starting access to a PCIe device. Access to an AXI configuration register also returns the AXI to the L0 state.

(1) ASPM L1 Idle Time Setting

The Mode Set 3 registers (offset: 380h) are defined as the ASPM L1 Idle Time registers. These registers are used to set the counter for monitoring the state of the AXI bus or internal transactions in terms of remaining activity when this unit is placed in the ASPM L1 state. The counter monitors whether the AXI bus and transactions within this unit are idle based on one of the following settings and uses this as the condition for triggering the transition to the ASPM L1 state.

“ASPM L1 Idle Time[7:0]” = 00h: 256 cycles of ACLK
“ASPM L1 Idle Time[7:0]” = 01h: 512 cycles of ACLK
:
“ASPM L1 Idle Time[7:0]” = FFh: 65536 cycles of ACLK

6.6.5.9 Power Management (Common)

6.6.5.9.1 Turn Off (PME Turn Off Message Recept)

Upon the reception of a PME_Turn_Off message, the TURN_OFF_EVENT signal is asserted and Deassert_INTx is issued for all of the INTx that are asserted. Moreover, preparation for sending PME_TO_Ack message is automatically started.

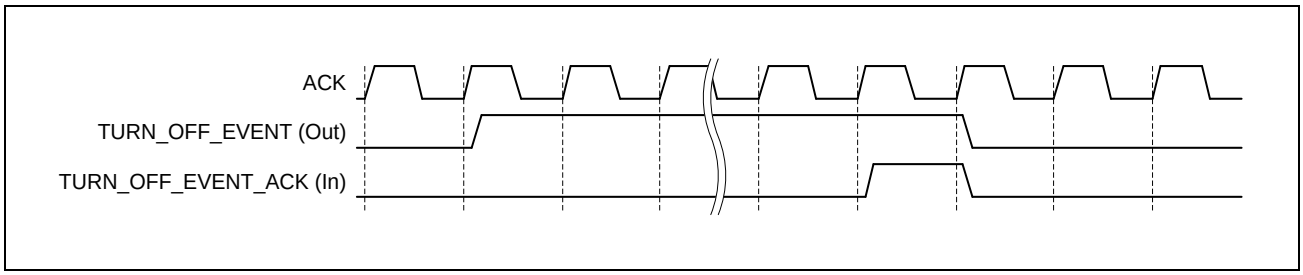


Figure 6.6-6 Relationship between TURN_OFF_EVENT and TURN_OFF_EVENT_ACK

6.6.5.9.2 Wake Up (PME Message transmit)

When CFG_PMCSR_PME_STATUS is 1b, preparation for automatic transmission of PME message is made. The transmission timing is set by the edge detection after the toggle timing of the PME_TIM signal is synchronized by ACLK.

Note that when a PME Turn Off message is received, automatic transmission of PME message will be stopped until ARESETn is generated or the DL_Up state is reached once after the DL_Down state is reached.

<Control procedure>

1. Among the configuration registers, when PME Enable = 1b in the PCI power management capability structure register, a PM_PME message can be sent from the EP to the RC by asserting the input pin, CFG_PMCSR_PME_STATUS, high and toggling the PME_TIM input.
2. The RC issues CfgWr to write 1b to and clear the PME_STATUS field of the device that has received the PM_PME message. At the same time, it sends CfgWr0 to change the Power State bit to D0.
3. When the EP enters the D0 state, the CFG_PMCSR_PME_STATUS_WRITECLEAR output signal of the EP is asserted.
4. By toggling the PME_TIM signal every 100 to 150 ms, the PM_PME Msg retransmission can be instructed in accordance with the following base spec. regulations.

Note that the PME_TIM, CFG_PMCSR_PME_STATUS, and CFG_PMCSR_PME_STATUS_WRITECLEAR signals are not used when there is no need to request the endpoint side to change the power state to D0.

6.6.5.10 Error Processing

Two error reporting paradigms are defined for PCI Express: baseline error reporting capability, which is the minimal requirement, and advanced error reporting (AER) capability, which can provide greater stability, as an optional error reporting facility. Our unit supports both error reporting functions.

6.6.5.10.1 Error Classification

PCI Express errors are of two types, correctable and uncorrectable. Uncorrectable errors are further classified into two types, non-fatal and fatal.

Error Type	Description
Correctable Error	An error which cannot be recovered by hardware
Uncorrectable Error (Non-Fatal)	An error which causes a particular transaction in PCI Express to be unreliable (but the PCI Express link itself is functional).
Uncorrectable Error (Fatal)	An error which causes the PCI Express link itself to be unreliable

In operation as a root complex, the above types of error are indicated by the individual interrupt names.

- INT_SERR_COR
- INT_SERR_NONFATAL
- INT_SERR_FATAL

6.6.5.10.2 Error Checking Mechanism

(1) Physical Layer Error List

The following type of error is to be detected in the physical layer.

- Receiver Error

(2) Data Link Layer Error List

The following type of error is to be detected in the link layer.

- Bad TLP Error
- Bad DLLP Error
- REPLAY_NUM Rollover
- Replay Timer Timeout
- Receiver Overflow Error

(3) Transaction Layer Error List

The following type of error is to be detected in the transaction layer.

- Completion Timeout
- Completer Abort
- Unsupported Request
- Unexpected Completion
- Malformed TLP
- Poisoned TLP

CAUTION

The specifications of the receiver error detection function in each LTSSM state are as listed below.

LTSSM State	Unit Specification	Base Spec Specifications
Configuration	Support	Gen1/Gen3 (Must)
Recovery	Non-Support	Option
L0	Support	Must
Disabled	Support	Option
Hot Reset	Support	Option

6.6.5.10.3 Error Message

The PCI Express Base Specification defines error messages as one of the mechanisms for notifying the system or another device of an error when it is detected by a PCI Express agent.

Error Message	Description
ERR_COR	Used when a correctable error is detected
ERR_NONFATAL	Used when a non-fatal, uncorrectable error is detected
ERR_FATAL	Used when a fatal, uncorrectable error is detected

Advisory Non-Fatal Error

The PCI Express Base Specification states that when a PCI Express agent as the detecting agent detects a non-fatal error, it handles the error as an advisory non-fatal error whether it does or does not support AER. In handling a non-fatal error as an advisory non-fatal error, the agent sends an ERR_COR message instead of an ERR_NONFATAL message and sends an advisory notification to the software. At this time, the advisory non-fatal error status bit of the correctable error status register is set to indicate the error state. Note that subsequent setting of the first error pointer register, logging of the header, and message transmission only proceed if the Advisory Non-Fatal Error Mask bit of the Correctable Error Mask Register is clear (no masking). They do not proceed if the bit is set.

The error cases which are handled as advisory non-fatal errors are as follows.

- Reception of unsupported non-posted requests
- Reception of a non-posted request with a completer abort completion response
- Reception of an unexpected completion
- Reception of a poisoned TLP (this is not handled as an advisory non-fatal error in this core)
- Detection of a completion timeout (this is not handled as an advisory non-fatal error in this core)

This unit may merge multiple error messages with the same ID. This would occur when multiple errors are detected during the wait for the unit to be ready for the transmission of messages after an error condition is detected. However, messages will not always be merged in such cases.

6.6.6 Operation

6.6.6.1 Setting Up

Set the internal registers of the unit including the configuration registers.

This section explains the procedure for setting up until the PCI Express link is up (the module is ready for data transfer).

For detailed procedures on releasing the reset and setting registers, see **6.6.6.5 PCIe Initialization Procedure**.

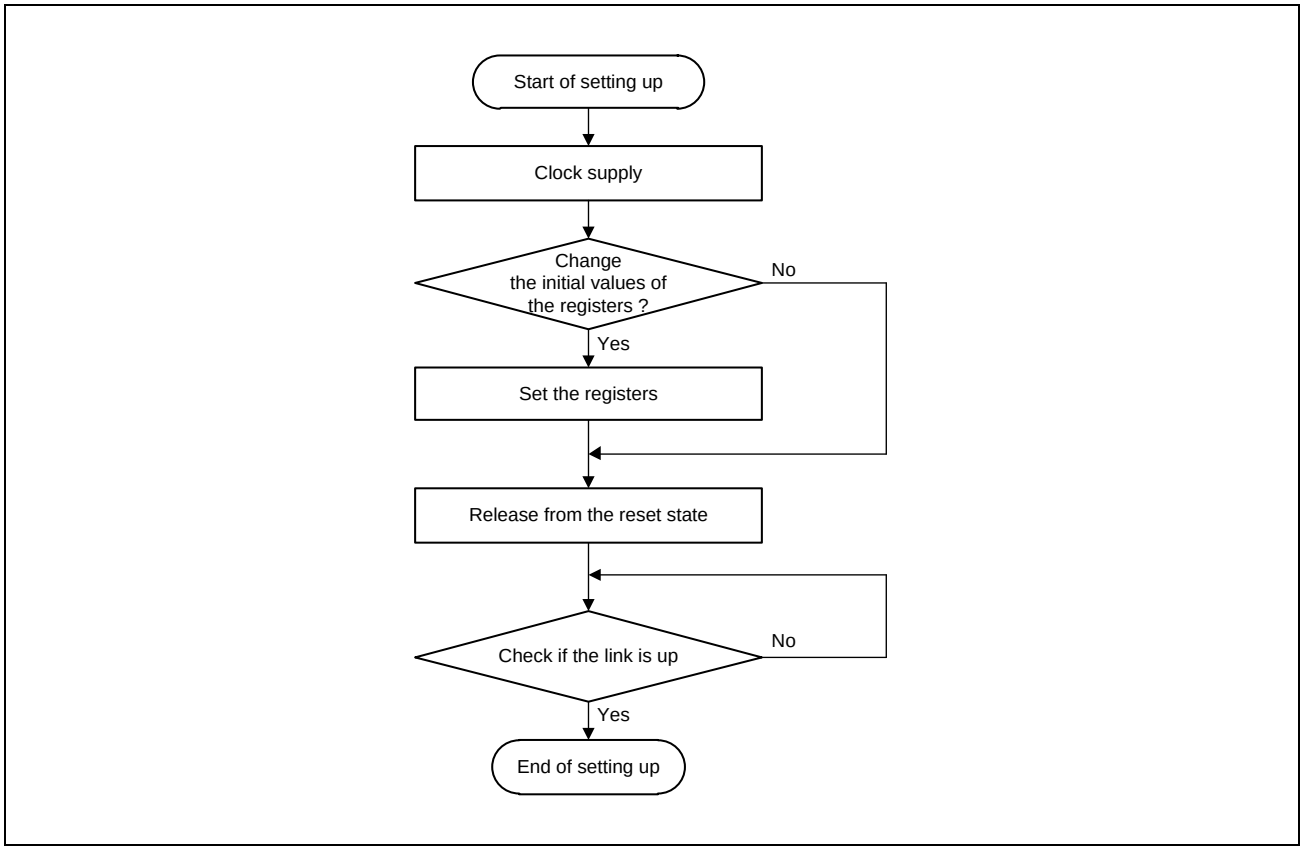


Figure 6.6-7 Setting Up

6.6.6.1.1 Changing the Initial Values of the Registers

Of the internal registers, the initial values of the configuration registers can be changed via the AXI slave interface.

1) AXI Bridge Registers

De-asserting ARESETn following supply of CLK allows access to these registers. In the case of resetting of the PCIe core, write to the registers while the reset signal is being asserted.

2) PCIe Configuration Register

De-asserting ARESETn following supply of CLK allows access to these registers. Of the reset signals of the PCIe core, those related to the configuration register must be de-asserted.

3) Physical Layer Control/Monitor Registers

De-asserting ARESETn following supply of CLK allows access to these registers. In the case of resetting of the PCIe core, write to the registers while the reset signal is being asserted. It is necessary to set the Permission Registers (<PCI0_base> + 0300h) bit[1] to 1b beforehand to allow access to the PIPE_PHY Register.

(1) Setting the initial values of the registers

The initial values of the registers listed below are 0. Set appropriate values in the registers before the start of link up.

- Device ID: Vendor and Device ID Register (<PCI0_base> + 6000h)
- Vendor ID: Vendor and Device ID Register (<PCI0_base> + 6000h)
- Class Code (base class/sub-class/programming interface):
  Revision ID and Class Code Register (<PCI0_base> + 6008h)
- Revision ID: Revision ID and Class Code Register (<PCI0_base> + 6008h)
- Subsystem ID: Subsystem ID (Function #n) (<PCI0_base> + 602Ch)
- Subsystem Vendor ID: Subsystem ID (Function #n) (<PCI0_base> + 602Ch)

(2) Setting the initial values of the registers

Make the following settings.

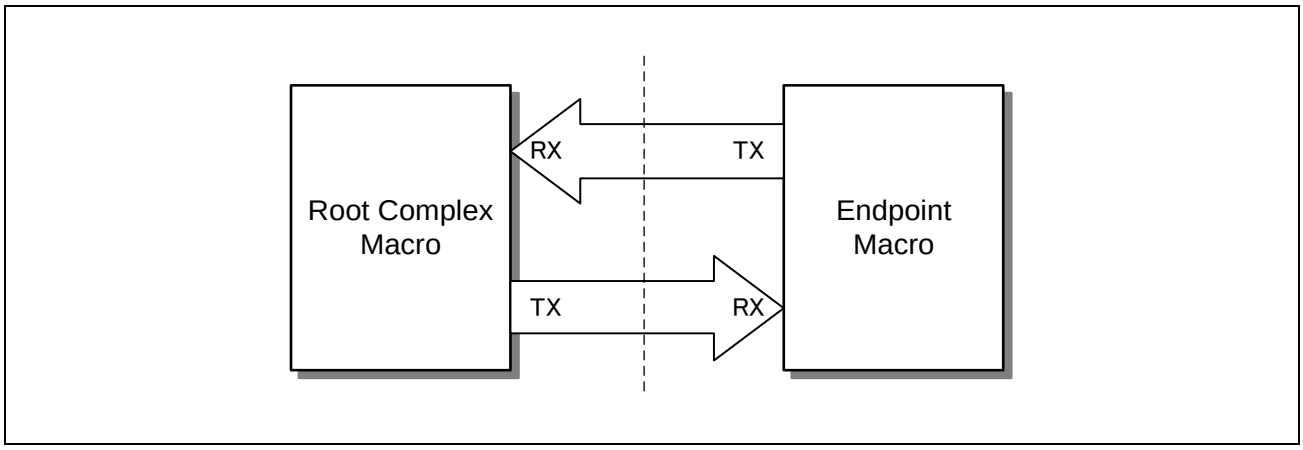
- Set the Upstream Port 8.0 GT/s Transmitter Preset and Downstream Port 8.0 GT/s Transmitter Preset in the Lane Equalization Control Register as follows:
  - <PCI0_base> + 61BCh: 0808_0808h (n = 0)
- Set the XCFGD Setting Resister.
  - <PCI0_base> + 20A0h: 05DB_B800h
- Set Device ID and Vendor ID of the manufacture device to Vendor and Device ID Register.
  Device ID of this LSI and Vendor ID of Renesas are below.
  - Device ID: 0038h (Endpoint) / 0039h (Root Complex)
  - Vendor ID: 1912h

6.6.6.1.2 De-asserting the Reset

In this unit, software reset control by the register is possible.

De-asserting of the reset of the core (the reset register in the case of register control) automatically starts operations such as receiver detection and the training sequence with the other-party device.

Note that our unit requires a wait of at least 5 ms from power-on to de-assertion of the reset.



Check that link negotiation with the other party has been completed and the link with the unit is up.

6.6.6.1.3 Checking if the Link is Up

Checking if the link is up can be done through either of the following methods. This is usually done by the root complex.

1) Polling

Have the CPU of the chip poll the DL_Down Status bit (bit [0]) of the PCIe core Status 1 Register in AXI bridge registers until the value of the bit is 0 (indicating the DL_Up state).

2) Interrupt

After assertion of the PCIE_EVT_INT pin of the interrupt signal, check the interrupt source by reading the DL_UpDown bit (bit [9]) in the PCIe Event Interrupt Status 0 register in AXI Bridge Registers. After that, check that the DL_Down Status bit (bit [0]) of the PCIe Core Status 1 Register is 0 (indicating the DL_Up state)

The settings above make the unit ready for transfer with the other party device.

However, data transfer such as reading or writing memory is not yet possible in this state. This requires subsequent window settings.

6.6.6.2 Setting the Windows (Root Complex Mode)

When using the window settings in the root complex configuration, the recommended settings in the BAR and BAR Mask registers are “BAR: All 0s” and “BAR Mask: All 1s”.

6.6.6.3 Setting the Windows (Endpoint Mode)

All PCI Express commands are issued via the set memory spaces or register spaces. After making the window settings and the configuration register settings, data such as PCIe MRd and MWr can be transferred. All PCI Express commands are issued via the set memory spaces or register spaces. After making the window settings and the configuration register settings, data can be transferred in response to PCIe MRd and MWr requests and so on.

6.6.6.3.1 BAR Specification (Two-Function Configuration)

Our unit has functionality for dividing the 64-bit memory space into window spaces for AXI access. Windows for access from PCIe to AXI and from AXI to PCIe are respectively referred to as AXI windows and PCI windows.

The AXI Bridge Registers (unit-specific registers) are accessible regardless of the function being used.

The 64-bit memory spaces must be set exclusively of each other and have no overlaps.

Set the AXI window spaces such that they do not overlap.

Table 6.6-128 Example BAR Settings (64 bits × 3, Function = 2)

Function	BAR (Memory Space)	AXI Window	Note
Function #0	64-bit memory space (1) {BAR1, BAR0}	AXI #0	
		AXI #1	
		AXI #2	
		AXI #3	
	64-bit memory space (2) {BAR3, BAR2}	AXI #4	
		AXI #5	
		AXI #6	
		AXI #7	
	64-bit memory space (3) {BAR5, BAR4}	—	AXI Bridge Registers
Function #1	64-bit memory space (1) {BAR1, BAR0}	AXI #0	
		AXI #1	
		AXI #2	
		AXI #3	
		AXI #4	
		AXI #5	
		AXI #6	
		AXI #7	
	64-bit memory space (2) {BAR3, BAR2}	—	Not allocated
	64-bit memory space (3) {BAR5, BAR4}	—	AXI Bridge Registers

Remarks: BAR: Abbreviation of Base Address Register

Note: The address spaces are invalid when the corresponding BAR values are all 0s. The BAR settings are required for each function.

The base addresses of the 64-bit memory spaces (1), (2), and (3) above are set by using the following configuration registers.

64-bit memory space (1): Configuration registers {BAR1, BAR0}
64-bit memory space (2): Configuration registers {BAR3, BAR2}
64-bit memory space (3): Configuration registers {BAR5, BAR4}

CAUTION

- Up to eight AXI windows can be set by dividing the 64-bit memory space (1).
- Up to eight AXI windows can be set by dividing the 64-bit memory space (2).
- The number of AXI windows allocated to each of the BAR areas is as follows.
  64-bit memory space (1): 64-bit memory space (2) = 8:0 or 4:4;
  Select either of the above settings. Note that all of the allocated AXI windows do not need to be used effectively.
- The 64-bit memory space (3) is dedicated for access to the AXI Bridge Registers.
- The base addresses of each of the AXI windows are limited within the 4-Gbyte space from the BAR base address.

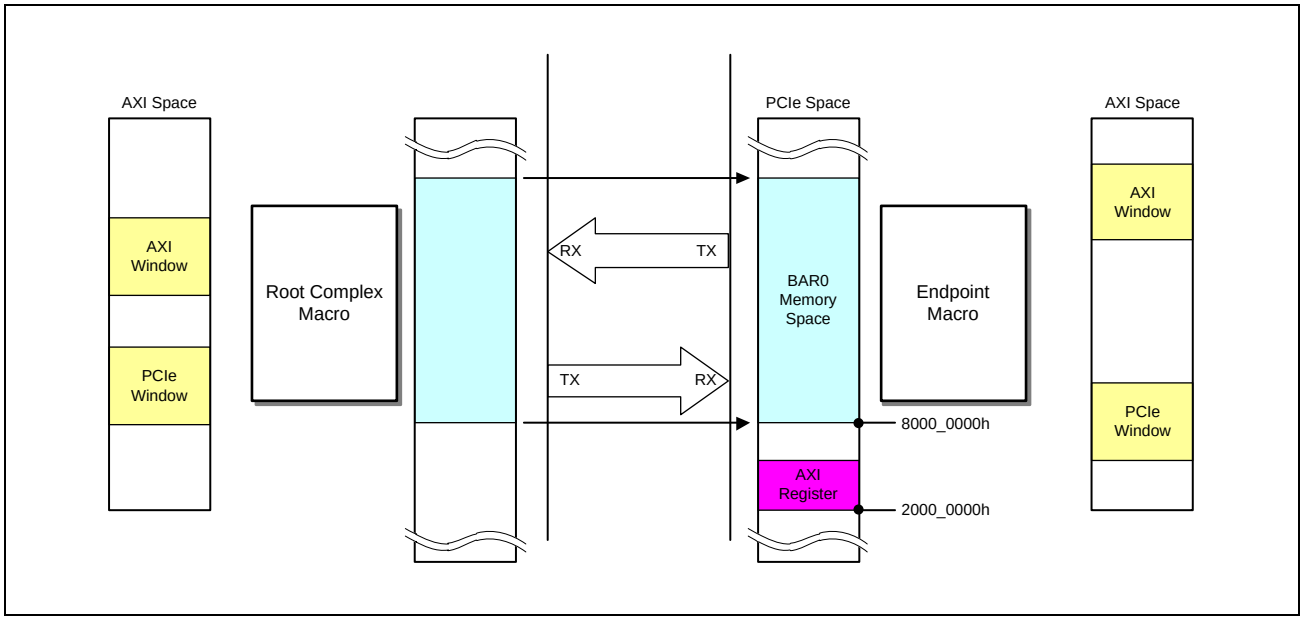
6.6.6.3.2 Setting the AXI Windows

In operation as an endpoint device, the AXI windows can be set in the following two ways.

1. The root complex issuing memory read and memory write requests.
2. Setting from the CPU of the endpoint under software control.

The following shows the procedure for making the settings by method 1.

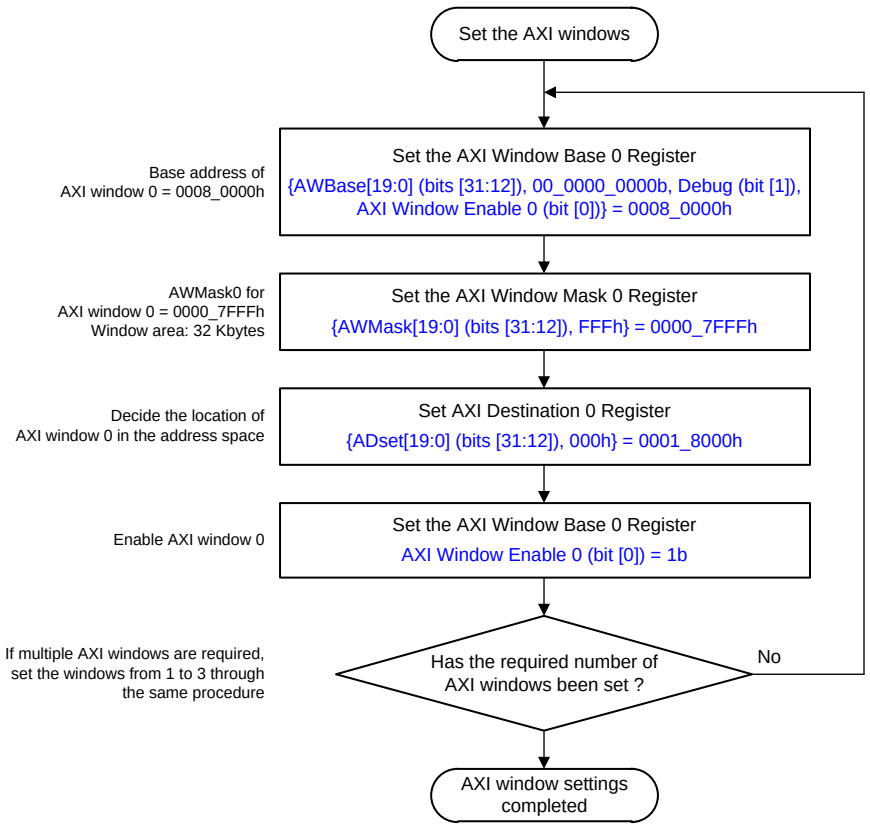
Set the AXI windows on the endpoint side by issuing memory read (MRd) and memory write (MWr) requests to the AXI Bridge Register access space set in the previous section.



Data transfer is possible when setting of the windows is completed and the Bus Master Enable bit and the Memory Space Enable bit of the given configuration register are set to the enabled state.

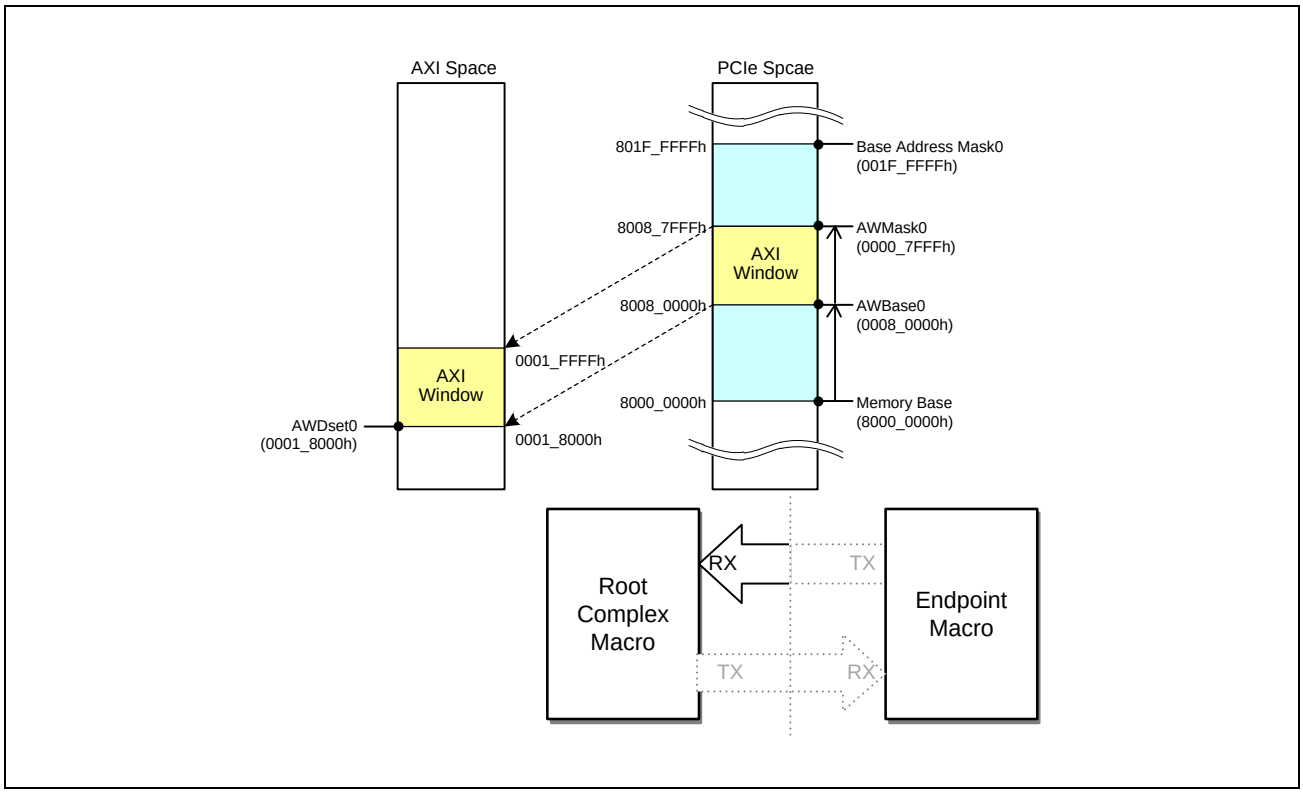
The following shows the procedure for making the settings by method 2.

The AXI windows can be set by write access to the registers through the AXI slave interface.



Note: The values assigned in the figure (values in blue) are examples.

Figure 6.6-8 Example Settings of the AXI Windows



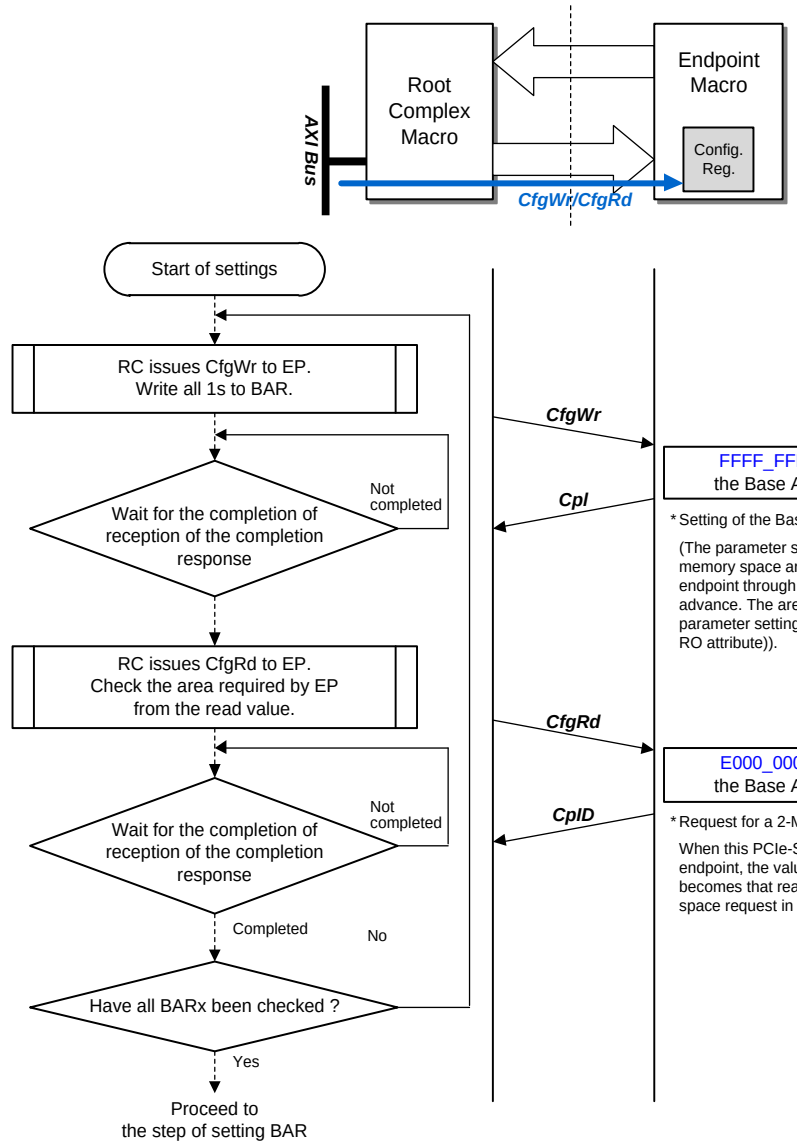
At this time, three address space settings, namely the AXI address space for the root complex device, the AXI address space for the endpoint device, and the PCI address space between the root complex and the endpoint, must be consistent within the system. If they are not, data cannot be transferred. Take care with the settings.

6.6.6.3.3 Setting the PCIe Address Space

A PC or similar system recognizes information on the areas of the memory space which are required by other-party endpoint devices by access to the configuration registers of the endpoints through software processing by the CPU on the root-complex side. In the case of embedded systems, etc., the areas required by the other-party devices will generally be known in advance, so this processing is not usually required.

The following is an example procedure for setting of the PCIe memory space for an other-party endpoint by the root complex.

Configuration requests are made to be issued to the other-party endpoint devices by register access through the AXI slave interface.



Note: The values assigned in the figure (values in blue) are examples.

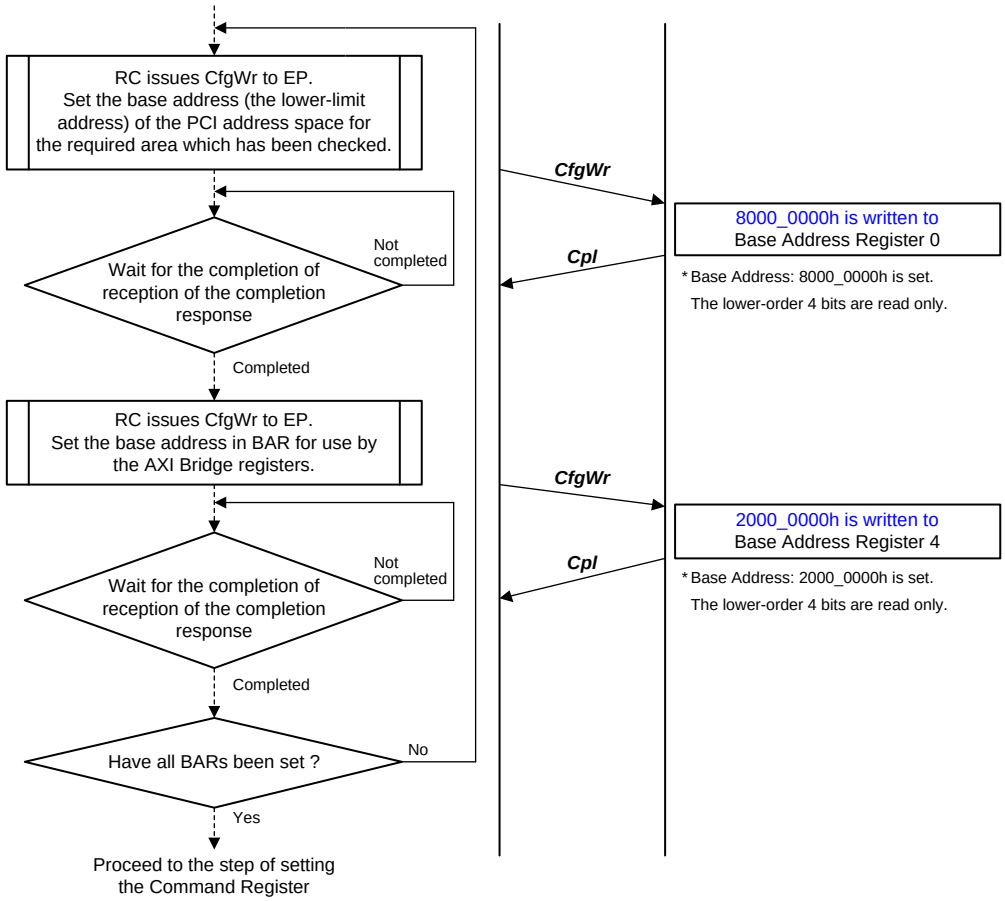
Figure 6.6-9 Example Settings for the PCIe Address Space

After issuing a configuration write request for writing all 1s (FFFF_FFFFh), a configuration read request is issued and if the result of reading is all 0s (0000_0000h), the given base address register (BAR) is judged to be reserved and the corresponding space is considered to be unused.

From the result of reading, software on the root complex side writes the base address (the lower-limit address) of the PCI address space to be allocated to the endpoint device by issuing a configuration write request.*[1] At this time, setting of a BAR for which the area requested by the endpoint device was found to be reserved and thus unused is not required.

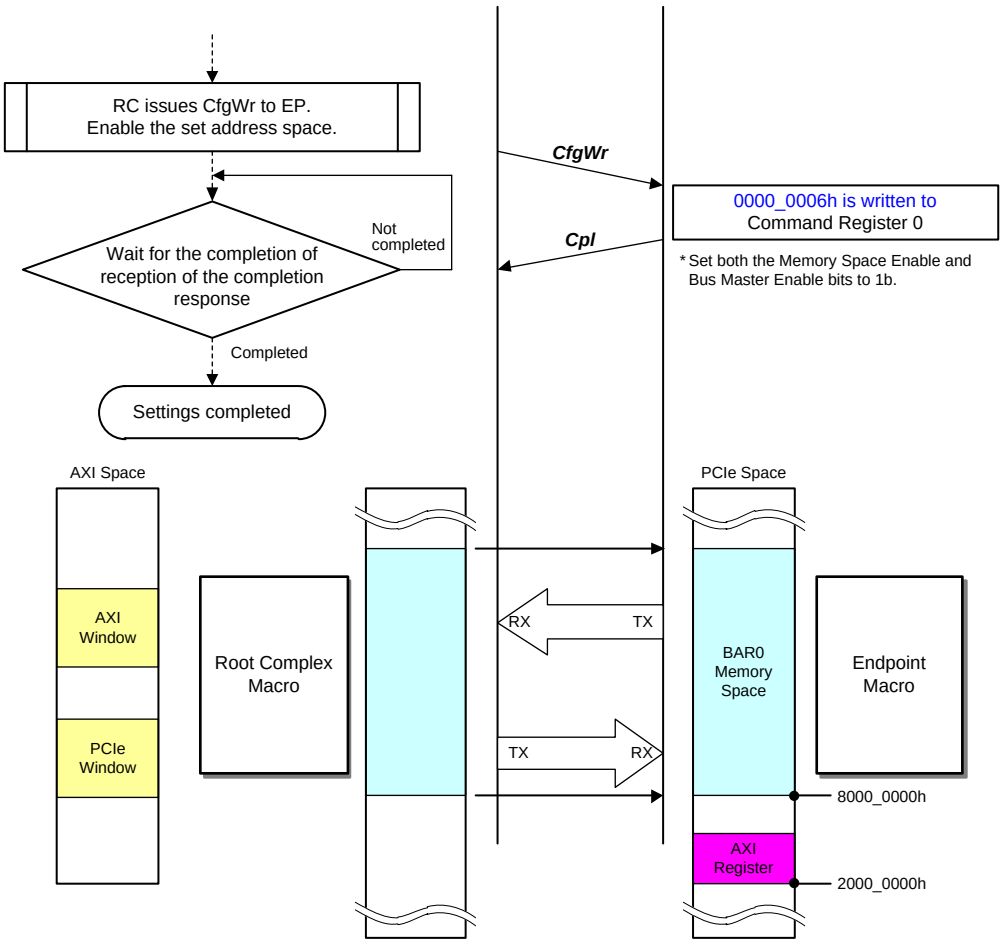
Also, if an address space which includes the other party has been prepared in advance, the above confirmation is not required. Set the base address directly.

Note 1. Setting a base address register to all 0s is prohibited.



Note: The values assigned in the figure (values in blue) are examples.

The completion of settings for the PCIe address space means that the root complex is ready for data transfer to the corresponding endpoint. Finally, set the Bus Master Enable bit and the Memory Space Enable bit of the given command register to enable the memory space.



Note: The values assigned in the figure (values in blue) are examples.

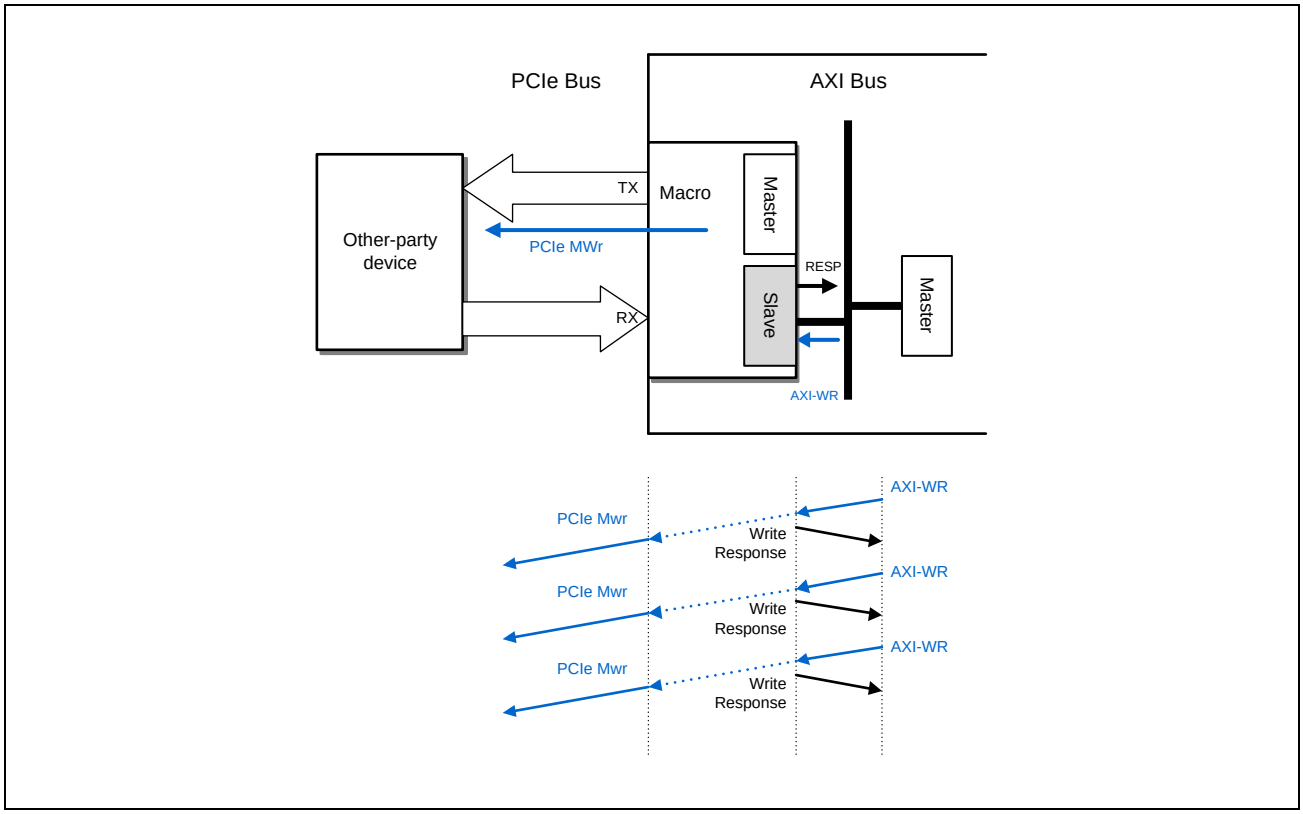
6.6.6.4 Data Transfer

This unit has one port each for master and slave operation as an AXI interface. For PCIe requests which can be issued through the master and slave ports, see **6.6.5.2 Issuing of PCIe Requests and Register Access (by AXI)** and **6.6.5.3 Initiation of AXI Transactions and Register Access (by PCIe)**.

The focus of this explanation is on normal memory data transfer.

6.6.6.4.1 PCIe MWr (when the AXI Slave is in Use)

A write transaction from the AXI bus (AXI-WR) via a window set as a PCIe window is converted into an MWr command (PCIe MWr) and then issued. If a PCIe MWr request is to be issued through the AXI interface slave port of the unit by using a DMAC, etc. external to the unit, the number of write transactions which can be accepted at a time is one, so the operation is as follows.



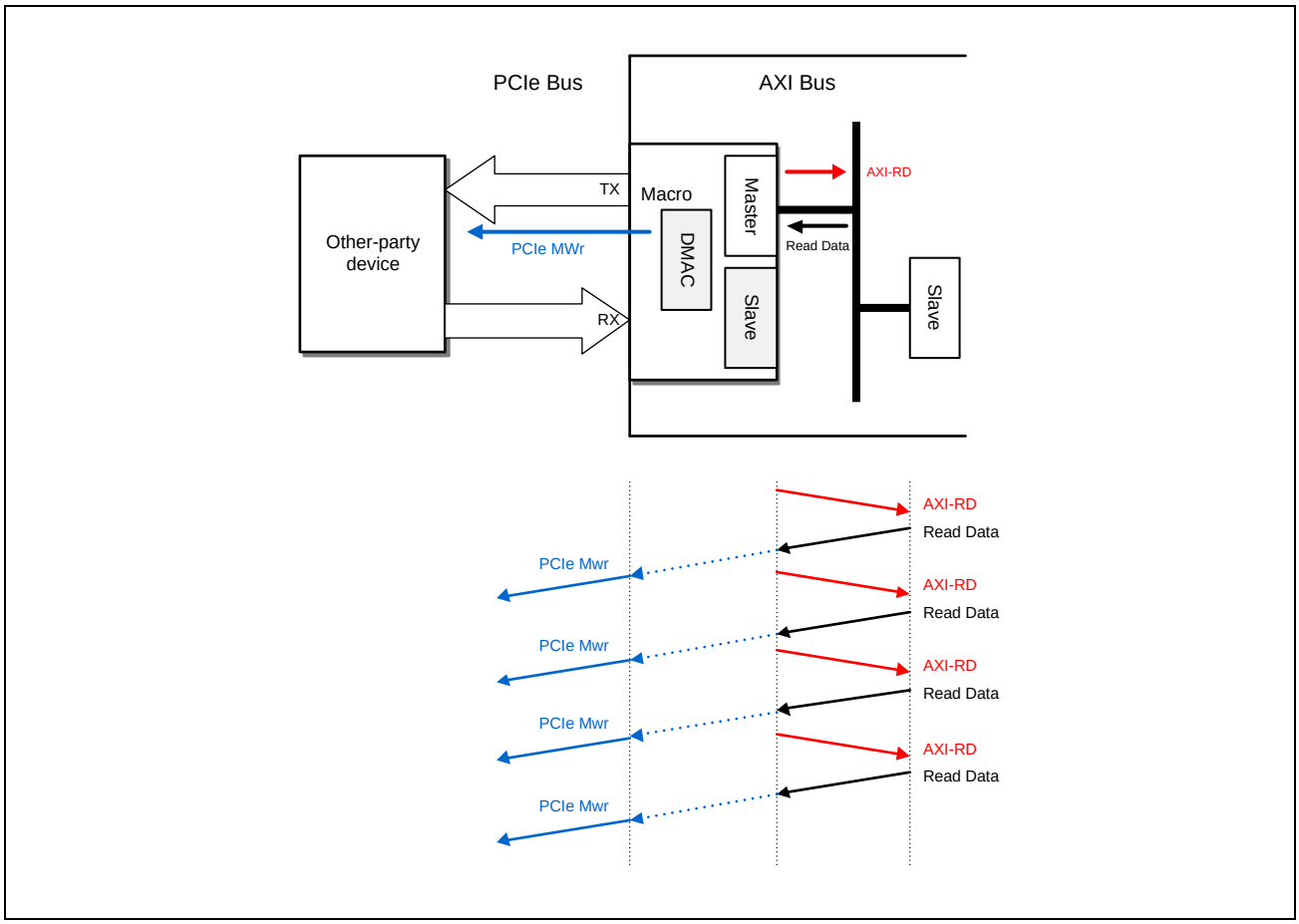
1. A write transaction from the AXI bus is issued through the AXI slave interface (write address channel, write data channel).
2. The transaction is converted into a PCIe MWr command and data are transferred to the other-party device via the PCIe bus.
3. After a wait for a response from the AXI slave interface (write response channel), a next AXI write transaction is issued.

A write response is issued after writing data to the transmission buffer (SRAM) within the unit. If this transmission buffer is full, the unit receives as much data as it can capture and then places the READY signal at the low level. Accordingly, the AXI bus may be placed in the hold state during transfer.

Since this depends not only on the specifications of the PCIe module (the number of lanes, rate, maximum payload size, etc.) but also on the size of the transmission buffer, the size of the reception buffer of the other-party device, and the external system configuration (the time for access to the external DRAM, etc.), care should be taken when considering the system configuration in general.

6.6.6.4.2 PCIe MWr (when the DMAC is in Use)

If a PCIe MWr request is to be issued through the AXI master port by using the DMAC within the unit, the number of requests which can be read by the internal DMAC is one transaction per channel, so the operation is as follows.



1. An AXI read (AXI-RD) request is issued through the AXI master interface (read address channel).
2. After the AXI master interface (read data channel) receives read data, this is converted into a PCIe MWr command and data are transferred to the other-party device via the PCIe bus.
3. Following the completion of the reception of read data, a next AXI-RD is issued.
4. The above steps are repeated until transfer of all bytes set in the DMA Size bits is completed.

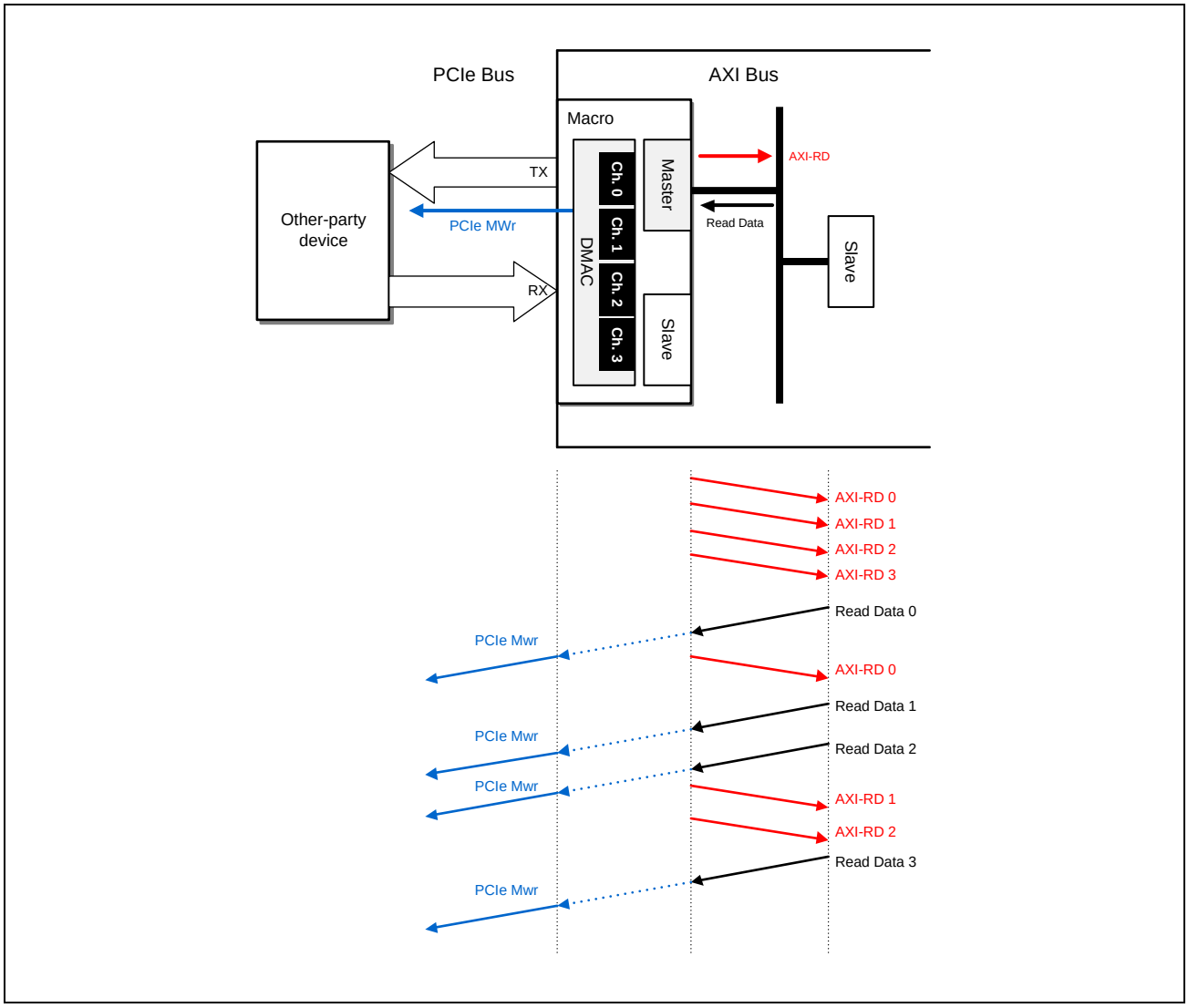
The read data channel checks that the data that have been read are valid and indicates the readiness of the data for reception. Even when the transmission buffer (SRAM) within the unit is full, the unit issues read requests through the read address channel. Also, it receives as much data as it can capture and places subsequent data in the non-receivable state. Accordingly, depending on the state of the transmission buffer, data may not be transferred regardless of a request having been issued, or the AXI bus may be placed in the hold state during transfer. This lengthens the period of waiting for data that have been read, leading to the deterioration of transfer performance as well as the deterioration of overall system performance.

Likewise, contention with write access by the slave interface and contention in write access between channels may decrease performance in transfer.

Since this depends not only on the specifications of the PCIe module (the number of lanes, rate, maximum payload size, etc.) but also on the size of the transmission buffer, the size of the reception buffer of the other-party device, and the

external system configuration (the time for access to the external DRAM, etc.), care should be taken when considering the system configuration in general.

The following is an example where the number of requests which can be read by the AXI master interfaced through four DMAC channels is 4.



1. An AXI read (AXI-RD0: DMAC ch. 0) request is issued through the AXI master interface (read address channel).
2. Since the number of requests which can be read by the unit = 4, AXI read requests are subsequently issued through DMAC ch. 1, ch. 2, and ch. 3.

   Note: No order of priority applies to the issuing of requests through these channels.

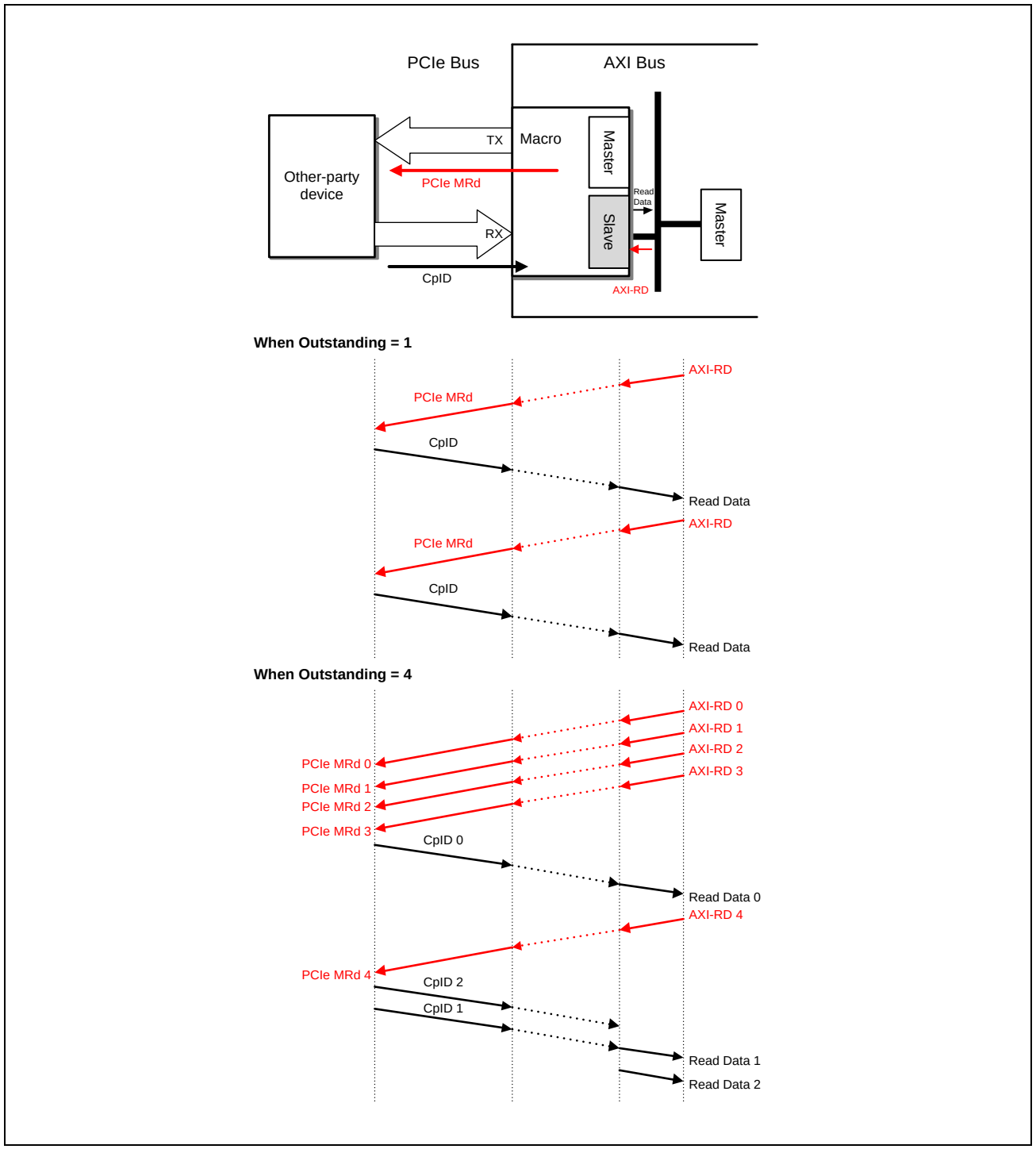
3. Following the completion of the reception of data read in response to the read request through ch. 0, ch. 0 is able to issue a next AXI-RD0.
4. The above steps are repeated until transfer of all bytes set in the DMA Size bits for each channel is completed.

Configuring data transfer as described above allows increased performance, although such a configuration complicates software control.

6.6.6.4.3 PCIe MRd (when the AXI Slave is in Use)

In usage as an AXI slave, memory read requests (PCIe MRd) are issued to the PCIe interface, a completion (CplD) is received from the PCIe interface, and an AXI write transfer (AXI-WR) is initiated (this unit supports eight outstanding transfers). Therefore, only the number of PCIe MRd requests that corresponds to this set number of outstanding transfers can be issued first, regardless of the reception of CplD. The received CplDs are stored in a data buffer (RAM) and then transferred through the AXI bus.

If a PCIe MRd is issued through the AXI interface slave port of the unit, the number of read requests which can be accepted by the AXI slave at a time = 1 to 8 corresponding to the PCIe section.



When outstanding = 1

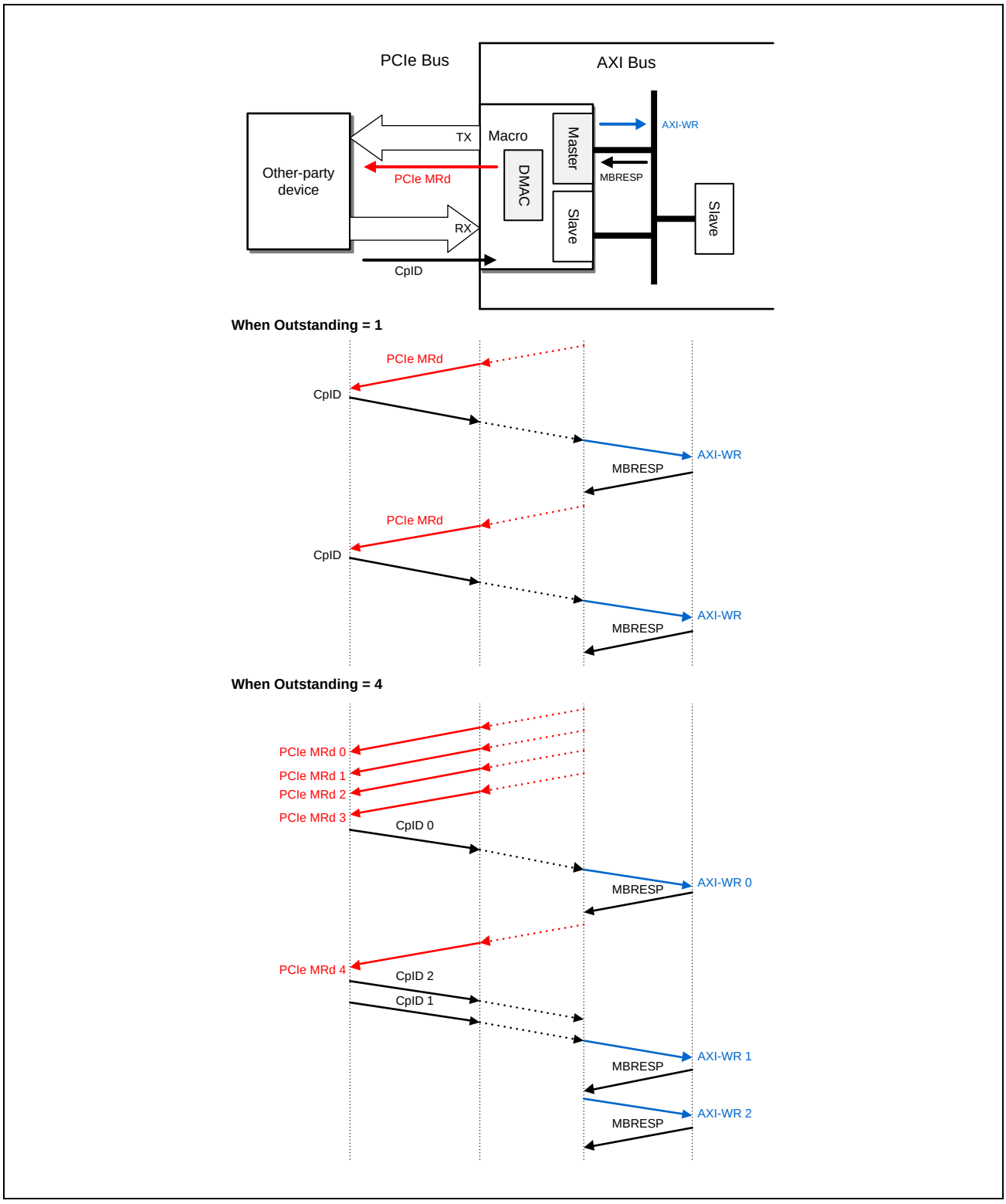
1. An AXI read transaction is issued through the AXI slave interface (read address channel).
2. A transaction is converted into a PCIe MRd command and the request for data reading is issued to the other-party device via the PCIe bus.
3. CplD is received from the other-party device and an AXI read response is issued through the AXI slave interface (read data channel).
4. Steps 1 to 3 are repeated.

When outstanding = 4

1. An AXI read transaction is issued through the AXI slave interface (read address channel).
2. A transaction is converted into a PCIe MRd command and the request for data reading is issued to the other-party device via the PCIe bus.
   Steps 1 and 2 can be repeated until up to four consecutive requests have been issued.
3. CplD is received from the other-party device and an AXI read response is issued through the AXI slave interface (read data channel).
   At this time, the PCIe module can operate with out-of-order transactions, so the order of CplDs from the other-party device may not be preserved. The unit also includes a buffer to support in-order transactions with the AXI bus. If this is used and the CplD for a preceding PCIe MRd is never returned, however, the next request cannot be issued and performance deteriorates accordingly.
   After that, steps 1 to 3 are repeated.

6.6.6.4.4 PCIe MRd (when the DMAC is in Use)

If a PCIe MRd request is to be issued by using the DMAC within the unit, the overall number of requests is also 1 to 8 when the DMAC is incorporated depending on the number of outstanding transfers.



1. Activate the internal DMAC through the AXI slave interface or from an other-party device.

When outstanding = 1

2. The DMAC issues a read request and this is converted to a PCIe MRd command within the unit. The request for data reading is issued to the other-party device via the PCIe bus.
3. CplD is received from the other-party device and an AXI write transaction is issued through the AXI master interface (write address channel, write data channel).
4. Following the completion of the AXI write transfer, a next request is issued.
5. Steps 2 to 4 are repeated until transfer of all bytes set in the DMA Size bits is complete

When outstanding = 4

2. The DMAC issues a read request and this is converted to a PCIe MRd command within the unit. The request for data reading is issued to the other-party device via the PCIe bus. Up to four consecutive requests can be issued.
3. CplD is received from the other-party device and an AXI write transaction is issued through the AXI master interface (write address channel, write data channel).
   At this time, the PCIe module can operate with out-of-order transactions, so the order of CplDs from the other-party device may not be preserved. The unit also includes a buffer to support in-order transactions with the AXI bus. If this is used and the CplD for a preceding PCIe MRd is never returned, however, the next request cannot be issued and performance deteriorates accordingly.
4. Following the completion of the AXI write transfer, a next request is issued.
5. Steps 2 to 4 are repeated until transfer of all bytes set in the DMA Size bits is completed.

6.6.6.5 PCIe Initialization Procedure

6.6.6.5.1 Initialization Procedure

(1) Root Complex Mode

An example of the initialization procedure for PCIe Root Complex mode is described in steps 1 to 13.

Table 6.6-129 Initialization Procedure (RC) (1/2)

Step 1	Set the Root Complex mode by the PCI Device Type setting register. SYS: Write 0000_0001h to the SYS_PCIE_MODE_CH0 register (1024h) (Root Complex mode)
Step 2	Set to the reset state. CPG: Write 0004_0000h to the CPG_RST11 register (92Ch) (PCIE_0_ARESETN (TYPE-A))
Step 3	Clock OFF setting CPG: Write 0030_0000h to the CPG_CLKON_12 register (630h) (PCIE_0_ACLK, PCIE_0_CLK_PMU)
Step 4	Release the reset. CPG: Write 0004_0004h to the CPG_RST11 register (92Ch) (PCIE_0_ARESETN (TYPE-A))
Step 5	Set the clock output to “On”. CPG: Write 0030_0030h to the CPG_CLKON_12 register (630h) (PCIE_0_ACLK, PCIE_0_CLK_PMU)
Step 6	Set to the PCIe reset state. PCI: Set bits [6:0] = 000_0000b in the PCI_RC_RESET register (310h) (RST_OUT_B, RST_PS_B, RST_LOAD_B, RST_CFG_B, RST_RSM_B, RST_GP_B, RST_B)
Step 7	Release the PCIe reset. PCI: Set bit 4 = 1b and bit 3 = 1b in the PCI_RC_RESET register (0310h) (RST_LOAD_B, RST_CFG_B)
Step 8*1	Setting of HWINT and PIPE_PHY related registers PCI: Set bit 2 = 1b in the PCI_RC_PERM register (300h) (CFG_HWINIT_EN) (access enable setting) PCI: Set bit 1 = 1b in the PCI_RC_PERM register (300h) (PIPE PHY Register Enable) (access enable setting) PCI: Write xxxx_xxxxh to the PCI_RC_VID register (6000h) (Device ID, Vendor ID) PCI: Write FFFF_FFDFh to the PCI_RC_RID_CC register (6008h) (Revision ID, Class Code) PCI: Write FFFF_FFFFh to the PCI_RC_BARMSK00L register (60A0h) PCI: Write FFFF_FFFFh to the PCI_RC_BARMSK00U register (60A4h) PCI: Write 0000_0000h to the PCI_RC_BSIZE00_01 register (60C8h) PCI: Write 0808_0808h to the PCI_RC_LEQCTL register (61BCh)*3 PCI: Write 05DB_B800h to the PCI_PHY_XCFGD register (20A0h)*3 PCI: Set bit 2 = 0b in the PCI_RC_PERM register (300h) (CFG_HWINIT_EN) (access disable setting) PCI: Set bit 1 = 0b in the PCI_RC_PERM register (300h) (PIPE PHY Register Enable) (access disable setting)
Step 9	SYS setting (ALLOW_ENTER_L1) SYS: Set bit 0 = 1b in the SYS_PCIE_MISC_CH0 register (1020h) SYS: Set bit 0 = 1b in the SYS_PCIE_MISC_CH1 register (1050h)

Table 6.6-129 Initialization Procedure (RC) (2/2)

Step 10*2	Interrupt settings PCI: Write 0000_1200h to the PCI_RC_PEIS0 register (0204h) PCI: Set bit 30 = 1b, bit 29 = 1b, bit 12 = 1b, and bit 9 = 1b in the PCI_RC_PEIE0 register (0200h) (UI_LINK_WIDTH_CHANGE_DONE EN, UI_LINK_SPEED_CHANGE_DONE EN, RX_DLLP_PM_ENTER_L23 EN, DL_UpDown EN) PCI: Write 0003_0303h to the PCI_RC_PEIS1 register (020ch) PCI: Set bit 17 = 1b, bit 16 = 1b, bit 9 = 1b, bit 8 = 1b, bit 1 = 1b, and bit 0 = 1b in the PCI_RC_PEIE1 register (0208h) (TXB_PARITY_ERR EN, ERR_RPC_REPLAYFIFO_PERR EN, ERR_REPLAY_HIGHER_CORRECTABLE_ERROR EN, ERR_REPLAY_LOWER_CORRECTABLE_ERROR EN, ERR_REPLAY_HIGHER_UNCORRECTABLE_ERROR EN, ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR EN) PCI: Write 0000_0F0Fh to the PCI_RC_AMEIS register (0214h) PCI: Set bits [11:8] = 1111b and bits [3:0] = 1111b in the PCI_RC_AMEIE register (0210h) (Write MSTERR INT EN [3:0], Read MSTERR INT EN [3:0]) PCI: Write 0000_0F03h to the PCI_RC_ASEIS1 register (0224h) PCI: Set bits [11:8] = 1111b, bit 1 = 1b, and bit 0 = 1b in the PCI_RC_ASEIE1 register (0220h) (Write SLVERR INT EN [3:0], Read SLVERR INT EN [1:0]) PCI: Write 010F_0000h to the PCI_RC_MSGRCVIS register (0124h) PCI: Write 0105_0000h to the PCI_RC_MSGRCVIE register (0120h)
Step 11	Release the reset. PCI: Set bit 5 = 1b, bit 1 = 1b, and bit 0 = 1b in the PCI_RC_RESET register (0310h) (RST_PS_B, RST_GP_B, RST_B)
Step 12	Wait for 500 µs or more
Step 13	Release the reset. PCI: Set bit 6 = 1b and bit 2 = 1b in the PCI_RC_RESET register (0310h) (RST_OUT_B, RST_RSM_B)

Note 1. Change the setting values according to the operating conditions.

Note 2. Set the corresponding interrupt handler prior to using this function.

Note 3. For details, refer to **6.6.6.1.1 Changing the Initial Values of the Registers**.

(2) Endpoint Mode

An example of the initialization procedure for PCIe End Point mode is described in steps 1 to 14.

Table 6.6-130 Initialization Procedure (EP) (1/2)

Step 1	Set the End Point mode by the PCI Device Type setting register SYS: Write 0000_0000h to the SYS_PCIE_MODE_CH0 register (1024h) (End Point mode)
Step 2	Set to the reset state. CPG: Write 0004_0000h to the CPG_RST11 register (92Ch) (PCIE_0_ARESETN (TYPE-A))
Step 3	Clock OFF setting CPG: Write 0030_0000h to the CPG_CLKON_12 register (630h) (PCIE_0_ACLK, PCIE_0_CLK_PMU)
Step 4	Release the reset. CPG: Write 0004_0004h to the CPG_RST11 register (92Ch) (PCIE_0_ARESETN (TYPE-A))
Step 5	Set the clock output to “On”. CPG: Write 0030_0030h to the CPG_CLKON_12 register (630h) (PCIE_0_ACLK, PCIE_0_CLK_PMU)
Step 6	Set to the PCIe reset state. PCI: Set bits [6:0] = 000_0000b in the PCI_EP_RESET register (310h) (RST_OUT_B, RST_PS_B, RST_LOAD_B, RST_CFG_B, RST_RSM_B, RST_GP_B, RST_B)
Step 7	Release the PCIe reset. PCI: Set bit 4 = 1b and bit 3 = 1b in the PCI_EP_RESET register (0310h) (RST_LOAD_B, RST_CFG_B)
Step 8	Setting of HWINT and PIPE_PHY related registers (Function #0) PCI: Set bit 2 = 1b in the PCI_EP_PERM register (300h) (CFG_HWINIT_EN) (access enable setting) PCI: Set bit 1 = 1b in the PCI_EP_PERM register (300h) (PIPE PHY Register Enable) (access enable setting) PCI: Write xxxx_xxxxh to the PCI_EP_VID_F0 register (6000h) (Device ID, Vendor ID) PCI: Write FFFF_FFFFh to the PCI_EP_RID_CC_F0 register (6008h) (Revision ID, Class Code) PCI: Write FFDF_FFFFh to the PCI_EP_SSID_F0 register (602Ch) (Subsystem ID, Subsystem Vendor ID) PCI: Write 1FFF_FFFFh to the PCI_EP_BARMSK00L_F0 register (60A0h) PCI: Write 0000_0000h to the PCI_EP_BARMSK00U_F0 register (60A4h) PCI: Write 0000_0000h to the PCI_EP_BSIZE00_01_F0 register (60C8h) PCI: Write 0000_0000h to the PCI_EP_BARMSK01L_F0 register (60A8h) PCI: Write 0000_0000h to the PCI_EP_BARMSK01U_F0 register (60ACh) PCI: Write 0000_1FFFh to the PCI_EP_BARMSK02L_F0 register (60B0h) PCI: Write 0000_0000h to the PCI_EP_BARMSK02U_F0 register (60B4h) PCI: Write 0000_0000h to the PCI_EP_BSIZE02_03_F0 register (60CCh) PCI: Write 0000_0000h to the PCI_EP_BSIZE04_05_F0 register (60D0h) PCI: Write 0000_0000h to the PCI_EP_BSIZE06_F0 register (60D4h) PCI: Write 0808_0808h to the PCI_EP_LEQCTL_F0 register (61BCh)*3 PCI: Write 05DB_B800h to the PCI_PHY_XCFGD register (20A0h)*3 PCI: Set bit 2 = 0b in the PCI_EP_PERM register (300h) (CFG_HWINIT_EN) (access disable setting) PCI: Set bit 1 = 0b in the PCI_EP_PERM register (300h) (PIPE PHY Register Enable) (access disable setting)

Table 6.6-130 Initialization Procedure (EP) (2/2)

Step 9*1	Setting of HWINT related registers (Function #1) PCI: Set bit 2 = 1b in the PCI_EP_PERM register (300h) (CFG_HWINIT_EN) (access enable setting) PCI: Write xxxx_xxxxh to the PCI_EP_VID_F1 register (7000h) (Device ID, Vendor ID) PCI: Write FFFF_FFFFh to the PCI_EP_RID_CC_F1 register (7008h) (Revision ID, Class Code) PCI: Write FDDF_FFFFh to the PCI_EP_SSID_F1 register (702Ch) (Subsystem ID, Subsystem Vendor ID) PCI: Write 1FFF_FFFFh to the PCI_EP_BARMSK00L_F1 register (70A0h) PCI: Write 0000_0000h to the PCI_EP_BARMSK00U_F1 register (70A4h) PCI: Write 0000_0000h to the PCI_EP_BSIZE00_01_F1 register (70C8h) PCI: Write 0000_0000h to the PCI_EP_BARMSK01L_F1 register (70A8h) PCI: Write 0000_0000h to the PCI_EP_BARMSK01U_F1 register (70ACh) PCI: Write 0000_1FFFh to the PCI_EP_BARMSK02L_F1 register (70B0h) PCI: Write 0000_0000h to the PCI_EP_BARMSK02U_F1 register (70B4h) PCI: Write 0000_0000h to the PCI_EP_BSIZE02_03_F1 register (70CCh) PCI: Write 0000_0000h to the PCI_EP_BSIZE04_05_F1 register (70D0h) PCI: Write 0000_0000h to the PCI_EP_BSIZE06_F1 register (70D4h) PCI: Set bit 2 = 0b in the PCI_EP_PERM register (300h) (CFG_HWINIT_EN) (access disable setting)
Step 10	SYS setting (ALLOW_ENTER_L1) SYS: Set bit 0 = 1b in the SYS_PCIE_MISC_CH0 register (1020h) SYS: Set bit 0 = 1b in the SYS_PCIE_MISC_CH1 register (1050h)
Step 11*2	Interrupt settings PCI: Write 0000_1200h to the PCI_EP_PEIS0 register (0204h) PCI: Set bit 30 = 1b, bit 29 = 1b, bit 12 = 1b, and bit 9 = 1b in the PCI_EP_PEIE0 register (0200h) (UI_LINK_WIDTH_CHANGE_DONE EN, UI_LINK_SPEED_CHANGE_DONE EN, RX_DLLP_PM_ENTER_L23 EN, DL_UpDown EN) PCI: Write 0003_0303h to the PCI_EP_PEIS1 register (020Ch) PCI: Set bit 17 = 1b, bit 16 = 1b, bit 9 = 1b, bit 8 = 1b, bit 1 = 1b, and bit 0 = 1b in the PCI_EP_PEIE1 register (0208h) (TXB_PARITY_ERR EN, ERR_RPC_REPLAYFIFO_PERR EN, ERR_REPLAY_HIGHER_CORRECTABLE_ERROR EN, ERR_REPLAY_LOWER_CORRECTABLE_ERROR EN, ERR_REPLAY_HIGHER_UNCORRECTABLE_ERROR EN, ERR_REPLAY_LOWER_UNCORRECTABLE_ERROR EN) PCI: Write 0000_0F0Fh to the PCI_EP_AMEIS register (0214h) PCI: Set bits [11:8] = 1111b and bits [3:0] = 1111b in the PCI_EP_AMEIE register (0210h) (Write MSTERR INT EN [3:0], Read MSTERR INT EN [3:0]) PCI: Write 0000_0F03h to the PCI_EP_ASEIS1 register (0224h) PCI: Set bits [11:8] = 1111b, bit 1 = 1b, and bit 0 = 1b in the PCI_EP_ASEIE1 register (0220h) (Write SLVERR INT EN [3:0], Read SLVERR INT EN [1:0]) PCI: Write 010F_0000h to the PCI_EP_MSGRCVIS register (0124h) PCI: Write 010A_0000h to the PCI_EP_MSGRCVIE register (0120h)
Step 12	Release the reset. PCI: Set bit 5 = 1b, bit 1 = 1b, and bit 0 = 1b in the PCI_EP_RESET register (0310h) (RST_PS_B, RST_GP_B, RST_B)
Step 13	Wait for 500 µs or more
Step 14	Release the reset. PCI: Set bit 6 = 1b and bit 2 = 1b in the PCI_EP_RESET register (0310h) (RST_OUT_B, RST_RSM_B)

Note 1. Change the setting values according to the operating conditions.

Note 2. Set the corresponding interrupt handler prior to using this function.

Note 3. For details, refer to **6.6.6.1.1 Changing the Initial Values of the Registers**.

6.6.7 Points for Caution and Restrictions

6.6.7.1 Points for Caution and Prohibited Items in the Issuing of Requests

The following describes the restrictions in the issuing of various requests.

(1) Prohibition of fixed bursts of 2 or more beats

Fixed bursts for 2 or more beats are prohibited. If an attempt to use these is made, an OKAY response is returned to the AXI bus, but this may lead to issuing of the unexpected requests to the PCIe side, or unexpected register access which will change register values, and so on.

(2) Points for caution in the issuing of special requests

The requests listed below are issued by access to the internal registers. The registers are only accessible from the AXI side and writing to them while requests are being processed is prohibited. Attempted access from the PCIe side will be ignored.

[Special Requests]

— Zero-Length Memory Read Request
— Configuration Read
— Configuration Write
— Message Request
— Message Request with data payload

CAUTION

Do not issue a special request for which issuing of the request is prohibited. Operation is not guaranteed if this is done.

6.6.7.2 Ordering Specifications of Received Non-Posted and Posted Requests

In the specifications of this unit, a non-posted request is not overtaken by a posted request on the receiving side.

Requests received from another party are output to the higher-level bus in the order of reception.

6.6.7.3 Caution when Changing the Speed Spontaneously by the EP Unit

Follow the procedure below. The following control bits are present in the PCI Core Control 2 Register and PCI Core Status 2 Register among the AXI-Bridge registers.

1. Wait until the home node is placed in the L0 state.
2. Read the value of the STATE_DATA_RATE_IDENTIFIER_RECEIVED bit to check the supported speed of the other-party node.
3. If the home node supports the speed to which the other-party node wants to change, set the UI_LINK_SPEED_CHANGE[1:0] bits and assert the UI_LINK_SPEED_CHANGE_REQ bit.
4. Wait for the UI_LINK_SPEED_CHANGE_DONE bit to be asserted (wait for the completion).

Remark UI_LINK_SPEED_CHANGE_REQ is retained until UI_LINK_SPEED_CHANGE_DONE is asserted. It should be de-asserted after checking the assertion of UI_LINK_SPEED_CHANGE_DONE.

6.6.7.4 Error Processing of Unsupported Requests

The following describes the flow of processing for error reporting and error logging.

(1) Access that straddles a 4-KB boundary

Data transfer to a memory space which straddles a 4-KB boundary cannot proceed (this is stipulated by the PCI Express Base Specification). In our unit, the reception of a memory write or read request for a memory space which straddles a 4-KB boundary is handled as a malformed TLP.

(2) RCB violations

The PCI Express Base Specification states that error processing in response to violations of the read completion boundary (RCB) is optional. The PCIe module of this LSI chip does not support the detection of RCB violations at the time of the reception of completion responses.

(3) Error processing in response to byte enable fields

The PCI Express Base Specification states that error processing in response to Byte Enable fields is optional. Our unit does not support the detection of errors in the form of violations of Byte Enable rules.

(4) Request that extends beyond the base address boundary

This core does not detect a request that starts within the range set in the Base Address Register but extends beyond the boundary from the base address as an error.

(5) Processing when a TPL that is ready to be transferred by the transmitter is a malformed TLP

Even if malformation of a TLP that is ready to be transferred by the transmitter is detected, due to the detection of parity errors when it is read from the FIFO buffer and so on, this unit does not suspend the transmission of such a TLP. Received malformed TLPs must be handled appropriately by other-party devices.

6.6.7.5 Processing on Reception of the Message

The following lists the messages to be detected as a silent drop or UR on reception of the given message.

Table 6.6-131 Operations on Message Reception

Received Message	Root Complex	Endpoint
Assert_INTx	Normal processing	Silent Drop
Deassert_INTx	Normal processing	Silent Drop
ERR_COR	Normal processing	Silent Drop
ERR_NONFATAL	Normal processing	Silent Drop
ERR_FATAL	Normal processing	Silent Drop
UNLOCK	Silent Drop	Silent Drop
Set_Slot_Power_Limit	Silent Drop	Normal processing
Vendor_Defined_Type0	UR	UR
Vendor_Defined_Type1	Silent Drop	Silent Drop
Ignore	Silent Drop	Silent Drop
LTR	UR	UR
OBFF	UR	UR
PM_PME	Normal processing	Silent Drop
PME_TO_Ack	Normal processing	Silent Drop
PM_Active_State_Nak	Silent Drop	Normal processing
PME_Turn_Off	Silent Drop	Normal processing
PTM_Request	UR	UR
PTM_Response	UR	UR
PTM_ResponseD	UR	UR
Invalidate_Request	Silent Drop	UR
Invalidate_Completion	Normal processing	Silent Drop
Page_Request	UR	UR
Page_Response	UR	UR

Note: Silent Drop: Normal completion, data are not reflected
UR: Unsupported Request

6.6.7.6 Notes on Window Settings

Note the following precautions and limitations when setting the window address.

- When adding the setting values of BAR* and the corresponding BAR MASK* register, make sure that bit carry does not occur.
- When adding the setting values of PCIe Window Base* and the corresponding PCIe Window Mask* register, make sure that bit carry does not occur.
- When adding the setting values of AXI Window Mask* and the corresponding AXI Destination* register, make sure that bit carry does not occur.
- When adding the setting values of PCIe Window Mask* and the corresponding PCIe Destination* register, make sure that bit carry does not occur.
- The values of the BAR Mask*, AXI Window Mask* and PCIe Window Mask* registers should be set to $2n - 1$.
- Each window should be a single memory size with a power of 2.
- The minimum area between BAR and BAR Mask is 4 Kbytes.
- Make sure that each window area and AXI Bridge register area (BAR4/5) do not overlap.

6.6.7.7 Other Points for Caution

(1) Access after the de-assertion of the reset signal

When a cycle of writing starts before the value of the SAWREADY bit has become 1b following the de-assertion of the reset signal, access in the second and subsequent cycles produces slave errors.

After the reset signal has been de-asserted, do not assert the SAWVALID signal until SAWREADY has become 1b.

(2) Point for caution at times of register writing

In some cases of writing 2DW or more and skipping over with the use of byte enable to a register from the AXI side, writing might not proceed as expected.

Using a pin reset or register reset from the AXI side as a non-consecutive SWSTRB, restricts writing to no more than 1DW (32 bits).

(3) Point for caution on register reading

In the case of reading a register from the AXI side, values read from invalid byte lanes are from undefined outputs (meaningless garbage data).

(4) Generation of unexpected correctable errors

A correctable error may be detected at the time of EIOS reception following low power state transitions of this unit such as from L0s to L1. If this happens, processing to send a message or assert an interrupt flag signal is to proceed. Take care on this point and respond appropriately so that a correctable error is not handled as a fatal error (a mask setting by a register to switch the notification of unexpected correctable errors off is recommended).

(5) Reset interval in transitions from a hot reset to detection in RC mode

In generations of PCI Express after Gen1, at the time of transitions from a hot reset to detection in accord with the operating rate, the base specification prescribes securing a 1-ms waiting interval for changes to the rate, so secure a reset interval of 1 ms.

SECTION 7 LOW-SPEED INTERFACE

7.1 Low-Speed Interface Overview

This section describes the Low-Speed Interface unit of this LSI. For details, refer to the sections of each unit.

■ Expanded Serial Peripheral Interface (xSPI) (See 7.2)

Supports connection of Serial flash memory. By setting the MD_BOOT[0:2] pin, it can be specified as a boot device.

■ Serial Communications Interface (RSCI) (See 7.3)

This LSI equips 10ch of SCI. SCI can communicate with UART, Simple SPI, Simple I2C.

■ Serial Communications Interface with FIFO (SCIF) (See 7.4)

In this LSI, SCIF executes UART communication. By setting the MD_BOOT[0:2] pin, download boot can be specified.

■ Serial Peripheral Interface (RSPI) (See 7.5)

This LSI equips 3ch of RSPI.

■ CRC Operation Unit (CRC) (See 7.6)

This unit generates the cyclic redundancy check (CRC) codes.

■ I^2C Bus Interface (RIIC) (See 7.7)

This LSI equips 9ch of RIIC.

RIIC8 is located in PD_AWO and can be used for PMIC control during boot.

■ I^3C Bus Interface (I3C) (See 7.8)

■ CAN-FD Interface (See 7.9)

■ A/D Converter (ADC) (See 7.10)

For this unit, analog inputs of up to 8 channels are selectable.

A signal from GPT or ELC can be used as an A/D conversion start trigger.

■ Temperature Sensor Unit (TSU) (See 7.11)

This LSI equips 1ch of TSU.

SECTION 7 LOW-SPEED INTERFACE

7.2 Expanded Serial Peripheral Interface (xSPI)

7.2.1 Overview

The xSPI protocol specifies the interface for Memory Devices, which provides high data throughput, low signal count, and limited backward compatibility with legacy SPI devices. The electrical interface can deliver up to 200 MB per second raw data throughput.

Table 7.2-1 lists the xSPI specifications, **Figure 7.2-1** shows a block diagram, and **Table 7.2-2** lists the I/O pins.

Table 7.2-1 xSPI Specifications

Item	Description
Number of channels	1 unit
Protocol	Compliant for the xSPI protocol
Data transmission and reception	Issue the transaction for up to 2 Slave as Master
Transfer speed	Support the transfer at xSPI200
Mode	• Support Protocol modes below – 1/4/8pin with SDR/DDR (1S-1S-1S, 4S-4D-4D*1, 8D-8D-8D*1) – 2/4pin with SDR (1S-2S-2S, 2S-2S-2S, 1S-4S-4S, 4S-4S-4S) • Configurable address length • Configurable initial access latency cycle • Support XiP mode
xSPI function	• Support Write Data Mask • Support In-band Reset (JESD252) • Memory-mapping – Support up to 256 MB address space (128 MB for each CS, or 256 MB for CS0 only) – Prefetch function for burst-read with low latency – Outstanding buffer for burst-write with high throughput • Manual command – Configurable up to 4 commands – Status Register Polling function • Input Strobe port timing shift
Interrupt source	2 interrupts

Note 1. DDR access without XSPI0_DS is not supported.

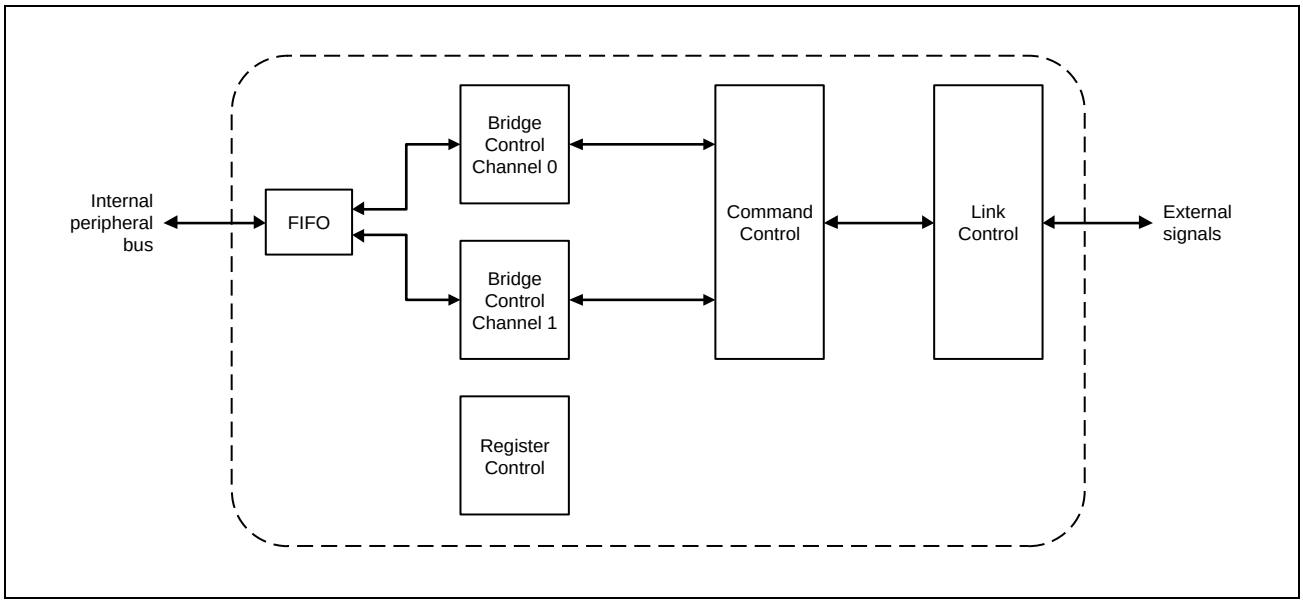


Figure 7.2-1 Block Diagram

Table 7.2-2 xSPI I/O Pins

Channel	Pin Name	Input/Output	Function
xSPIm (m = 0)	XSPIm_CKP	Output	Clock Positive
	XSPIm_CKN	Output	Clock Negative
	XSPIm_CS0N	Output	Chip Select for slave0
	XSPIm_CS1N	Output	Chip Select for slave1
	XSPIm_DS	I/O	Read Data Strobe / Write Data Mask
	XSPIm_IO0	I/O	Data 0 input/output
	XSPIm_IO1	I/O	Data 1 input/output
	XSPIm_IO2	I/O	Data 2 input/output
	XSPIm_IO3	I/O	Data 3 input/output
	XSPIm_IO4	I/O	Data 4 input/output
	XSPIm_IO5	I/O	Data 5 input/output
	XSPIm_IO6	I/O	Data 6 input/output
	XSPIm_IO7	I/O	Data 7 input/output
	XSPIm_RESET0N	Output	Master reset status for slave0
	XSPIm_RESET1N	Output	Master reset status for slave1
	XSPIm_RSTO0N	Input	Slave reset status for slave0
	XSPIm_RSTO1N	Input	Slave reset status for slave1
	XSPIm_INT0N	Input	Interrupt for slave0
	XSPIm_INT1N	Input	Interrupt for slave1
	XSPIm_ECS0N	Input	Error Correction Status for slave0
	XSPIm_ECS1N	Input	Error Correction Status for slave1
	XSPIm_WP0N	Output	Write Protect for slave0
	XSPIm_WP1N	Output	Write Protect for slave1

Table 7.2-3 Address Map

Space	Internal Address
CS0	0_2000_0000h (7000_0000h*1, 6000_0000h*2)
CS1	0_2800_0000h (7800_0000h*1, 6800_0000h*2)

Note: This table shows the default address map. The address map can be changed by SYS_SPI_STAADDCS1 and SYS_SPI_ENDADDCS0-1 registers.

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

7.2.2 Registers

Table 7.2-4 Register Base Address

Base Address Name	Base Address
<XSPIm_base>	0_1103_0000h (5103_0000h*1, 4103_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

7.2.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
xSPI Wrapper Configuration Register	XSPIm_WRAPCFG	0000_0000h	0000h	32
xSPI Common Configuration Register	XSPIm_COMCFG	0000_0000h	0004h	32
xSPI Bridge Map Configuration Register CH0	XSPIm_BMCFGCH0	0000_0000h	0008h	32
xSPI Bridge Map Configuration Register CH1	XSPIm_BMCFGCH1	0000_0000h	000Ch	32
xSPI Command Map Configuration Register 0 CS0	XSPIm_CMCFG0CS0	0000_0000h	0010h	32
xSPI Command Map Configuration Register 1 CS0	XSPIm_CMCFG1CS0	0008_0000h	0014h	32
xSPI Command Map Configuration Register 2 CS0	XSPIm_CMCFG2CS0	0008_0000h	0018h	32
Reserve	-	-	001Ch to 001Fh	-
xSPI Command Map Configuration Register 0 CS1	XSPIm_CMCFG0CS1	0000_0000h	0020h	32
xSPI Command Map Configuration Register 1 CS1	XSPIm_CMCFG1CS1	0008_0000h	0024h	32
xSPI Command Map Configuration Register 2 CS1	XSPIm_CMCFG2CS1	0008_0000h	0028h	32
Reserve	-	-	002Ch to 004Fh	-
xSPI Link I/O Configuration Register CS0	XSPIm_LIOCFGCS0	0007_0000h	0050h	32
xSPI Link I/O Configuration Register CS1	XSPIm_LIOCFGCS1	0007_0000h	0054h	32
Reserve	-	-	0058h to 005Fh	-
xSPI Bridge Map Control Register 0	XSPIm_BMCTL0	0000_00FFh	0060h	32
xSPI Bridge Map Control Register 1	XSPIm_BMCTL1	xxxx_xxxxh	0064h	32
xSPI Command Map Control Register CH0	XSPIm_CMCTLCH0	0000_0000h	0068h	32
xSPI Command Map Control Register CH1	XSPIm_CMCTLCH1	0000_0000h	006Ch	32
xSPI Command Manual Control Register 0	XSPIm_CDCTL0	0000_0000h	0070h	32
xSPI Command Manual Control Register 1	XSPIm_CDCTL1	0000_0000h	0074h	32
xSPI Command Manual Control Register 2	XSPIm_CDCTL2	0000_0000h	0078h	32
Reserve	-	-	007Ch to 007Fh	-
xSPI Command Manual Type Buf 0	XSPIm_CDTBUF0	0000_0000h	0080h	32
xSPI Command Manual Address Buf 0	XSPIm_CDABUF0	0000_0000h	0084h	32
xSPI Command Manual Data 0 Buf 0	XSPIm_CDD0BUF0	0000_0000h	0088h	32
xSPI Command Manual Data 1 Buf 0	XSPIm_CDD1BUF0	0000_0000h	008Ch	32
xSPI Command Manual Type Buf 1	XSPIm_CDTBUF1	0000_0000h	0090h	32
xSPI Command Manual Address Buf 1	XSPIm_CDABUF1	0000_0000h	0094h	32
xSPI Command Manual Data 0 Buf 1	XSPIm_CDD0BUF1	0000_0000h	0098h	32
xSPI Command Manual Data 1 Buf 1	XSPIm_CDD1BUF1	0000_0000h	009Ch	32
xSPI Command Manual Type Buf 2	XSPIm_CDTBUF2	0000_0000h	00A0h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
xSPI Command Manual Address Buf 2	XSPIm_CDABUF2	0000_0000h	00A4h	32
xSPI Command Manual Data 0 Buf 2	XSPIm_CDD0BUF2	0000_0000h	00A8h	32
xSPI Command Manual Data 1 Buf 2	XSPIm_CDD1BUF2	0000_0000h	00ACh	32
xSPI Command Manual Type Buf 3	XSPIm_CDTBUF3	0000_0000h	00B0h	32
xSPI Command Manual Address Buf 3	XSPIm_CDABUF3	0000_0000h	00B4h	32
xSPI Command Manual Data 0 Buf 3	XSPIm_CDD0BUF3	0000_0000h	00B8h	32
xSPI Command Manual Data 1 Buf 3	XSPIm_CDD1BUF3	0000_0000h	00BCh	32
Reserve	-	-	00C0h to 00FFh	-
xSPI Link Pattern Control Register 0	XSPIm_LPCTL0	0000_0000h	0100h	32
xSPI Link Pattern Control Register 1	XSPIm_LPCTL1	0000_0000h	0104h	32
xSPI Link I/O Control Register	XSPIm_LIOCTL	0003_0003h	0108h	32
Reserve	-	-	010Ch to 012Fh	-
xSPI Command Calibration Control Register 0 CS0	XSPIm_CCCTL0CS0	1F00_0000h	0130h	32
xSPI Command Calibration Control Register 1 CS0	XSPIm_CCCTL1CS0	0000_0000h	0134h	32
xSPI Command Calibration Control Register 2 CS0	XSPIm_CCCTL2CS0	0000_0000h	0138h	32
xSPI Command Calibration Control Register 3 CS0	XSPIm_CCCTL3CS0	0000_0000h	013Ch	32
xSPI Command Calibration Control Register 4 CS0	XSPIm_CCCTL4CS0	0000_0000h	0140h	32
xSPI Command Calibration Control Register 5 CS0	XSPIm_CCCTL5CS0	0000_0000h	0144h	32
xSPI Command Calibration Control Register 6 CS0	XSPIm_CCCTL6CS0	0000_0000h	0148h	32
xSPI Command Calibration Control Register 7 CS0	XSPIm_CCCTL7CS0	0000_0000h	014Ch	32
xSPI Command Calibration Control Register 0 CS1	XSPIm_CCCTL0CS1	1F00_0000h	0150h	32
xSPI Command Calibration Control Register 1 CS1	XSPIm_CCCTL1CS1	0000_0000h	0154h	32
xSPI Command Calibration Control Register 2 CS1	XSPIm_CCCTL2CS1	0000_0000h	0158h	32
xSPI Command Calibration Control Register 3 CS1	XSPIm_CCCTL3CS1	0000_0000h	015Ch	32
xSPI Command Calibration Control Register 4 CS1	XSPIm_CCCTL4CS1	0000_0000h	0160h	32
xSPI Command Calibration Control Register 5 CS1	XSPIm_CCCTL5CS1	0000_0000h	0164h	32
xSPI Command Calibration Control Register 6 CS1	XSPIm_CCCTL6CS1	0000_0000h	0168h	32
xSPI Command Calibration Control Register 7 CS1	XSPIm_CCCTL7CS1	0000_0000h	016Ch	32
Reserve	-	-	0170h to 017Fh	-
xSPI Version Register	XSPIm_VERSTT	0000_0000h	0180h	32
xSPI Common Status Register	XSPIm_COMSTT	0070_0000h	0184h	32
xSPI Calibration Status Register CS0	XSPIm_CASTTCS0	0000_0000h	0188h	32
xSPI Calibration Status Register CS1	XSPIm_CASTTCS1	0000_0000h	018Ch	32
xSPI Interrupt Status Register	XSPIm_INTS	0000_0000h	0190h	32
xSPI Interrupt Clear Register	XSPIm_INTC	xxxx_xxxxh	0194h	32
xSPI Interrupt Enable Register	XSPIm_INTE	0000_0000h	0198h	32

7.2.2.2 Register Description

The prefix (XSPIm_) of the register names is omitted in this and subsequent sections.

7.2.2.2.1 xSPI Configuration Registers

These registers configure xSPI Master function. These registers should be configured in the initialization phase. When the setting is needed to be changed after beginning xSPI transaction, stop all communicaitons (see **7.2.3.6.3 Flow of communication stop**) before changing the value of xSPI Configuration Register.

NOTE

The initialization phase means the period between reset releasing and issuing 1st xSPI transaction by memory mapping, manual-command or calibration.

(1) xSPI Wrapper Configuration Register (XSPIm_WRAPCFG)

This register has functions to configure xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	DSSFTCS1[4:0]					-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	DSSFTCS0[4:0]					-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
28 to 24	DSSFTCS1 [4:0]	0h	RW	DS shift for slave1 The function is same as one of slave0.
23 to 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12 to 8	DSSFTCS0 [4:0]	0h	RW	DS shift for slave0 This field configures the number of delay cells for XSPIm_DS pin. It is used to adjust the DS sampling timing. 1 cell delay is around 125 ps. When automatic calibration is enabled, it can be updated automatically. 00h: No shift 01h: Add a delay of 1 cell : 1Eh: Add a delay of 30 cells 1Fh: Add a delay of 31 cells
7 to 0	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

(2) xSPI Common Configuration Register (XSPIm_COMCFG)

This register has functions to configure xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OENEGEX	OEASTEX
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ARBMD[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
17	OENEGEX	0h	RW	Output Enable Negating extension This field extends 1 cycle output enable of Data and DS during output enable negating. This bit should not be used in case of no latency cycle, because xSPI output data could be conflicted with xSPI input data. 0b: No extend 1 cycle Output enable 1b: Extend 1 cycle Output enable
16	OEASTEX	0h	RW	Output Enable Asserting extension This field extends 1 cycle output enable of Data and DS during output enable asserting. When set to 1, CS asserting should be extended (LIOCFGCSn.CSASTEX = 1 (n = 0, 1)). This field should not be used in case of no latency cycle, because xSPI output data could be conflicted with xSPI input data. 0b: No extend 1 cycle Output enable 1b: Extend 1 cycle Output enable
15 to 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1, 0	ARBMD[1:0]	0h	RW	Channel arbitration mode This field selects the behavior when system bus accesses from ch0 and ch1 to slave devices occurred simultaneously. It is used only for memory-mapping mode. 00b: Round-Robbin (ch0-ch1-ch0-ch1…) 01b: Always ch0 win 10b: Always ch1 win 11b: Reserved

(3) xSPI Bridge Map Configuration Register CHn (XSPIm_BMCFGCHn) (n = 0, 1)

This register has functions to configure xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0008h + n x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMBTIM[7:0]								-	-	-	-	-	-	-	PREEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MWRSIZE[7:0]								MWRCOMB	-	-	-	-	-	-	WRMD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CMBTIM[7:0]	0h	RW	Combination timer This field specifies the expiration period of the combination timer. 00h means disabling the combination timer. When the timer is expired, the data in the combination buffer is pushed to memory device.
23 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	PREEN	0h	RW	Prefetch enable This field enables prefetch function for read transaction in memory-mapping mode. It could reduce the latency for read transaction with incremental address. 0b: Disable prefetch function 1b: Enable prefetch function
15 to 8	MWRSIZE[7:0]	0h	RW	Memory Write Size This field selects the size to combine incremental address in memory-mapping mode. It transmits an xSPI frame with the data combined up to the configured size while the address is incremental. Before the target size is reached, when non-incremental address or a read transaction are detected, pending data is transmitted to xSPI bus. 00h: Combine incremental address up to 4 bytes 01h: Combine incremental address up to 8 bytes : 0Eh: Combine incremental address up to 60 bytes 0Fh: Combine incremental address up to 64 bytes FFh: Combine incremental address up to 2 bytes Others: Setting prohibited
7	MWRCOMB	0h	RW	Memory Write Combination mode This field selects to combine the xSPI data in write access of memory-mapping mode. When this field is set to 0, xSPI data size depends on system bus's burst type and size. When this field is set to 1, the data size depends on MWRSIZE field. When this field is set to 1, any write transaction could be held in this xSPI master temporarily. 0b: Disable combination mode 1b: Enable combination mode
6 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	WRMD	0h	RW	System bus Write Response mode This field selects the timing of system bus write response in memory-mapping mode. When this field is set to 1, it returns the response after transmitting a frame on xSPI bus. When this mode is enabled, Memory Write Combination mode must be disabled. 0b: Return response after storing to Internal Write Buffer 1b: Return response after issuing write transaction to xSPI bus

(4) xSPI Command Map Configuration Register 0 CSn (XSPIm_CMCFG0CSn) (n = 0, 1)

This register has functions to configure xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0010h + n x 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADDRPCD[7:0]								ADDRPEN[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	ARYAMD	WPBSTMD	ADDSIZE[1:0]		FFMT[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ADDRPCD [7:0]	0h	RW	Address Replace Code This field configures the code to replace the MSByte of system bus address in memory-mapping mode. It replaces the corresponding bits when Address Replace Enable bit is set to 1b.
23 to 16	ADDRPEN[7:0]	0h	RW	Address Replace Enable This field selects the bits to replace for the MSByte of system bus address in memory-mapping mode. 0b: No replacement (xSPI frame address field is same as system bus address) 1b: Replacement
15 to 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	ARYAMD	0h	RW	Array address mode When this field is set to 1b, address for memory is mapped as {A[25:4], 00_0000b, A[3:0]} where A[25:0] is normal address. This field is effective only when FFMT=01b. 0b : Normal address mode 1b : Array address mode
4	WPBSTMD	0h	RW	Wrapping burst mode When this field is set to 1b, the wrapping boundary between system bus access and xSPI memory should be matched. 0b: Separate xSPI transfer at the wrapping address boundary 1b: Not separate xSPI transfer at the wrapping address boundary
3, 2	ADDSIZE[1:0]	0h	RW	Address size This field configures the number of address bytes in memory-mapping mode. In case of 8D-8D-8D profile 2.0, it should be configured to 4 bytes. 00b: 1 byte (256-byte address space) 01b: 2 bytes (64 KB address space) 10b: 3 bytes (16 MB address space) 11b: 4 bytes (4 GB address space)
1, 0	FFMT[1:0]	0h	RW	Frame format This field configures xSPI frame format in memory-mapping mode. See **Table 7.2-11** for detail. 00b: Normal format: Command 1 byte, Address ADDSIZE, Data up to system bus transaction. 01b: 8D-8D-8D profile 1.0 format: Command 2 bytes, Address ADDSIZE, Data up to system bus transaction 10b: 8D-8D-8D profile 2.0 Command Modifier format: Command & Modifier 6 bytes, Data up to system bus transaction 11b: 8D-8D-8D profile 2.0 Commands with Extended Command Modifier format: Command & Modifier 6 bytes, Data up to system bus transaction

(5) xSPI Command Map Configuration Register 1 CSn (XSPIm_CMCFG1CSn) (n = 0, 1)

This register has functions to configure xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0014h + n x 0010h
Initial Value : 0008_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	RDLATE[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDCMD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20 to 16	RDLATE[4:0]	8h	RW	Read latency cycle This field configures the latency cycle of read transaction in memory-mapping mode. 00h: No latency 01h: 1 cycle : 1Eh: 30 cycles 1Fh: 31 cycles
15 to 0	RDCMD[15:0]	0h	RW	Read command This field configures the command field of read transaction in memory-mapping mode. Normal format and 8D-8D-8D profile 2.0 format use only upper 1 byte. 8D-8D-8D profile 1.0 format uses 2 bytes.

(6) xSPI Command Map Configuration Register 2 CSn (XSPIm_CMCFG2CSn) (n = 0, 1)

This register has functions to configure xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0018h + n x 0010h
Initial Value : 0008_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	WRLATE[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WRCMD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20 to 16	WRLATE[4:0]	8h	RW	Write latency cycle This field configures the latency cycle of write transaction in memory-mapping mode. 00h: No latency 01h: 1 cycle : 1Eh: 30 cycles 1Fh: 31 cycles
15 to 0	WRCMD[15:0]	0h	RW	Write command This field configures the command field of write transaction in memory-mapping mode. Normal format and 8D-8D-8D profile 2.0 format use only upper 1 byte. 8D-8D-8D profile 1.0 format uses 2 bytes.

(7) xSPI Link I/O Configuration Register CSn (XSPIm_LIOCFGCSn) (n = 0, 1)

This register has functions to configure xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0050h + n x 0004h
Initial Value : 0007_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DDRSMPEX[3:0]				SDRSMPSFT[3:0]				SDRSMPMD	SDRDRV	CSNEGEX	CSASTEX	CSMIN[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	WRMSKMD	LATEMD	PRTMD[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DDRSMPEX [3:0]	0h	RW	DDR sampling window extend This field configures the cycle of extending the sampling window in DDR. In DDR, the input data is sampled during the expected cycle soon after latency cycle. The input data out of range is ignored. It can be configured depending on DS propagation delay. 0h: Expand no cycle 1h: Expand 1 cycle : 6h: Expand 6 cycles 7h: Expand 7 cycles Others: Setting prohibited
27 to 24	SDRSMPSFT [3:0]	0h	RW	SDR Sampling window shift This field shifts the timing of CK sampling in SDR. In case of using DS in SDR, there is no influence on the behavior. In case of DDR or using DS in SDR, it should be set to 0. 0h: Sample without delay 1h: Sample at 1 cycle delay : 6h: Sample at 6 cycle delay 7h: Sample at 7 cycle delay Others: Setting prohibited
23	SDRSMPMD	0h	RW	SDR Sampling mode This field selects the edge of sampling in SDR. In DDR, regardless of this setting, it samples data input ports with both edges of DS. When this field is set to 1b, data is sampled at rising-edge before falling-edge. 0b: Samples data input at falling-edge 1b: Samples data input at rising-edge
22	SDRDRV	0h	RW	SDR driving timing This field configures the timing of data output in SDR. This field should not be set to 1b in case of no latency cycle, because xSPI output data could be conflicted with xSPI input data. 0b: Drive at 1/2 cycle before CK rising-edge 1b: Drive at CK rising-edge
21	CSNEGEX	0h	RW	CS negating extension This field extends 1 cycle chip select pins when negating. 0b: No extension 1b: Extend 1 cycle

Bit	Bit Name	Initial Value	R/W	Description
20	CSASTEX	0h	RW	CS asserting extension This field extends 1 cycle chip select pins when asserting. 0b: No extension 1b: Extend 1 cycle
19 to 16	CSMIN[3:0]	7h	RW	CS minimum idle term This field configures the minimum cycle between xSPI frames. 0h: 1 cycle 1h: 2 cycles : Eh: 15 cycles Fh: 16 cycles
15 to 12	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
11	WRMSKMD	0h	RW	Write mask mode This field selects to use Data strobe port as write data mask. It is used only for 8D-8D-8D protocol mode. 0b: Write mask disable 1b: Write mask enable
10	LATEMD	0h	RW	Latency mode This field selects the behavior of initial access latency phase for both direct-manual mode and memory-mapping mode. When this field is set to 0b, the latency cycle is equal to each configured cycle from transmitting address field. When this field is set to 1b, the latency cycle is incremented from the last byte-pair of Address field and is extended 2 times of each configured cycle depending on data strobe port. It is used only for profile 2.0 frame format of 8D-8D-8D protocol mode with 6 bytes command/address field. Refer to xSPI protocol for detail. For this bit, refer to **Table 7.2-5**.
9 to 0	PRTMD[9:0]	0h	RW	Protocol mode This field configures the protocol mode and the pin to sample data inputs. In case of using not xSPI clock but Data strobe for sampling in SDR mode, it is required to set PRTMD[9] to 1. 000h: 1S-1S-1S 3B2h: 4S-4D-4D 3FFh: 8D-8D-8D 048h: 1S-2S-2S 049h: 2S-2S-2S 090h: 1S-4S-4S 092h: 4S-4S-4S Others: Setting prohibited

Table 7.2-5 LATEMD Bit Description

Value	Frame format	Usage
0b: Configurable latency	Profile 2.0	The configurable latency cycle should be set as minus 1.
	Others	Latency cycle increments after address field.
1b: Variable latency	Profile 2.0	Latency cycle increments from address [23:16]. And it should not be set to 1.
	Others	Not supported

7.2.2.2.2 xSPI Control Registers

These registers control xSPI Master function.

(1) xSPI Bridge Map Control Register 0 (XSPIm_BMCTL0)

This register has functions to control xSPI Master function.

This register should be configured in the initialization phase. When the setting is needed to be changed after beginning xSPI transaction, stop all communications (see **7.2.3.6.3 Flow of communication stop**) before changing the value of BMCTL0.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0060h
Initial Value : 0000_00FFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CH1CS1ACC [1:0]		CH1CS0ACC [1:0]		CH0CS1ACC [1:0]		CH0CS0ACC [1:0]	
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7, 6	CH1CS1ACC [1:0]	3h	RW	System bus ch1 to slave1 memory area access enable This field enables the access from ch1 to CS1 memory. 00b: Read/Write disable 01b: Read enable, Write disable 10b: Read disable, Write enable 11b: Read/Write enable
5, 4	CH1CS0ACC [1:0]	3h	RW	System bus ch1 to slave0 memory area access enable This field enables the access from ch1 to CS0 memory. 00b: Read/Write disable 01b: Read enable, Write disable 10b: Read disable, Write enable 11b: Read/Write enable
3, 2	CH0CS1ACC [1:0]	3h	RW	System bus ch0 to slave1 memory area access enable This field enables the access from ch0 to CS1 memory. 00b: Read/Write disable 01b: Read enable, Write disable 10b: Read disable, Write enable 11b: Read/Write enable
1, 0	CH0CS0ACC [1:0]	3h	RW	System bus ch0 to slave0 memory area access enable This field enables the access from ch0 to CS0 memory. 00b: Read/Write disable 01b: Read enable, Write disable 10b: Read disable, Write enable 11b: Read/Write enable

(2) xSPI Bridge Map Control Register 1 (XSPIm_BMCTL1)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0064h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	PBUFCLRCH1	PBUFCLRCH0	MWRPUSHCH1	MWRPUSHCH0	-	-	-	-	-	-	-	-
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	x	W	Reserved Whenever it is read, undefined value is read. The written value should always be 0b.
11	PBUFCLRCH1	x	W	The read value is undefined Prefetch Buffer clear for ch1 This function is same as PBUFCLRCH0. 0b: No command 1b: Clear request
10	PBUFCLRCH0	x	W	The read value is undefined Prefetch Buffer clear for ch0 This field requests to clear the prefetch buffer when the prefetch function is enabled. It should not be set during memory access (COMSTT.MEMACCCHn = 1). 0b: No command 1b: Clear request
9	MWRPUSHCH1	x	W	The read value is undefined Memory Write Data Push for ch1 The function is same as MWRPUSHCH0. 0b: No command 1b: Push request
8	MWRPUSHCH0	x	W	The read value is undefined Memory Write Data Push for ch0 This field requests to push the pending data in combination mode. 0b: No command 1b: Push request
7 to 0	-	x	W	Reserved Whenever it is read, undefined value is read. The written value should always be 0b.

(3) xSPI Command Map Control Register CHn (XSPIm_CMCTLCHn) (n = 0, 1)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0068h + n x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	XIPEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	XIPEXCODE[7:0]								XIPENCODE[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	XIPEN	0h	RW	XiP mode enable This field enables XiP mode in memory-mapping mode. When this field is set to 1, XiP enter code is inserted in the latency field, and the command field in next transaction is omitted. When this field is set to 0, XiP exit code is inserted in the latency field. And it is set to 0 automatically when transmitting XiP disable pattern. It should not be used for 8D-8D-8D protocol mode profile 2.0 frame format. 0b: Disable XiP mode 1b: Enable XiP mode
15 to 8	XIPEXCODE [7:0]	0h	RW	XiP mode exit code This field configures the code to exit XiP mode in memory-mapping mode.
7 to 0	XIPENCODE [7:0]	0h	RW	XiP mode enter code This field configures the code to enter XiP mode in memory-mapping mode.

(4) xSPI Command Manual Control Register 0 (XSPIm_CDCTL0)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0070h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	PERREP[3:0]				-	-	-	PERITV[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	TRNUM[1:0]		CSSEL	-	PERMD	TRREQ
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
27 to 24	PERREP[3:0]	0h	RW	Periodic transaction repeat This field configures the number of transaction repetitions in periodic manual-command mode. 0h: 1 (= 2^0) time 1h: 2 (= 2^1) times : Eh: 16384 (= 2^{14}) times Fh: 32768 (= 2^{15}) times
23 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20 to 16	PERITV[4:0]	0h	RW	Periodic transaction interval This field configures the interval of transaction in periodic manual-command mode. Too short interval compared with CPU bus cycle could result in no store into command buffer0. The interval should be longer than 4 times the system bus clock cycle. 00h: 2 (= 2^1) cycles 01h: 4 (= 2^2) cycles : 1Eh: 2,147,483,648 (= 2^{31}) cycles 1Fh: 4,294,967,296 (= 2^{32}) cycles
15 to 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5, 4	TRNUM[1:0]	0h	RW	Transaction number This field configures the number of transactions in normal manual-command mode. In periodic manual-command, the read data of last command is compared. 00b: Issue 1 command (using command buffer 0) 01b: Issue 2 commands (using command buffer 0-1) 10b: Issue 3 commands (using command buffer 0-2) 11b: Issue 4 commands (using command buffer 0-3)
3	CSSEL	0h	RW	Chip select This field selects a target memory to issue manual-command. 0b: CS0 1b: CS1
2	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
1	PERMD	0h	RW	Periodic mode This field enables the periodic transaction mode. When set to 1b, it repeats a transaction periodically and compares the read value with the expected value. It alternates the status polling operation for external memory. 0b: Direct manual-command mode 1b: Periodic manual-command mode
0	TRREQ	0h	RW	Transaction request This field requests to issue the transaction of manual-command. When this field is set to 1b, the transaction is started. This field is cleared to 0 when the transaction is completed. The transaction is canceled by clearing to 0 while the transaction is ongoing. 0b: No transaction 1b: Request transaction

(5) xSPI Command Manual Control Register 1 (XSPIm_CDCTL1)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0074h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PEREXP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEREXP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PEREXP[31:0]	0h	RW	Periodic transaction expected value This field configures the expected value to compare with the read value in periodic manual-command mode. For example, in case of comparing 1 byte, the lower byte should be configured.

(6) xSPI Command Manual Control Register 2 (XSPIm_CDCTL2)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0078h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PERMSK[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PERMSK[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PERMSK[31:0]	0h	RW	Periodic transaction masked value This field configures the masked value for the expected value in periodic manual-command mode. When this field is set 1 to any bit, the corresponding bit configured as expected value (CDCTL1.PEREXP[31:0]) is ignored. In 8D-8D-8D, the data bytes are transferred only in byte pairs on xSPI bus. It means the dummy read data could be stored. It should be masked for unused bits. For example, in case of read lower 1 byte, it should be configured to FFFF_FF00h.

(7) xSPI Command Manual Type Buf n (XSPIm_CDTBUFn) (n = 0 to 3)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0080h + n x 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRTYPE	-	LATE[4:0]					DATASIZE[3:0]				ADDSIZE[2:0]			CMDSIZE[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	CMD[15:0]	0h	RW	Command (1-2 bytes) This field configures the command field in manual-command mode. The number of bytes configured in Command Size bit is transferred. 1S-1S-1S, 4S-4D-4D: CMD[15:8] is command field, CMD[7:0] is not used. 8D-8D-8D profile 1.0: CMD[15:8] is command field, CMD[7:0] is extension field. 8D-8D-8D profile 2.0: CMD[15:0] is upper 2 bytes of command & modifier field. (bit 47-32 in xSPI protocol)
15	TRTYPE	0h	RW	Transaction Type This field selects the type of transaction. 0b: Read transaction (Readout data from slave device) 1b: Not read transaction
14	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
13 to 9	LATE[4:0]	0h	RW	Latency cycle This field configures the latency cycle in manual-command mode. 0h: No latency 1h: 1 cycle : 1Eh: 30 cycles 1Fh: 31 cycles
8 to 5	DATASIZE[3:0]	0h	RW	Write/Read Data Size This field configures the size of data field. In 8D-8D-8D, the data bytes are transferred only in byte pairs on xSPI bus. For example, even if configuring 1 byte for read, 2 bytes data is received. The last byte should be ignored. The 0 bytes must not configured for read transaction. 0h: 0 bytes (No data phase) 1h: 1 byte : 7h: 7 bytes 8h: 8 bytes Others: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
4 to 2	ADDSIZE[2:0]	0h	RW	Address size This field configures the size of address field. 000b: 0 bytes (No address phase) 001b: 1 byte 010b: 2 bytes 011b: 3 bytes 100b: 4 bytes Others: Setting prohibited
1, 0	CMDSIZE[1:0]	0h	RW	Command Size This field configures the size of command field. In case of 8D-8D-8D, it should be fixed to 10b. It should not be configured both command size and address size to zero. 00b: 0 bytes (No command phase) 01b: 1 byte 10b: 2 bytes Others: Setting prohibited

(8) xSPI Command Manual Address Buf n (XSPIm_CDABUFn) (n = 0 to 3)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0084h + n x 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADD[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ADD[31:0]	0h	RW	Address This field configures the address field in manual-command mode. 1S-1S-1S, 4S-4D-4D, 8D-8D-8D profile 1.0: It is address field. 8D-8D-8D profile 2.0: It is lower 4 bytes of command & modifier field. (bit 31-0 in xSPI protocol)

(9) xSPI Command Manual Data 0 Buf n (XSPIm_CDD0BUFn) (n = 0 to 3)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0088h + n x 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	0h	RW	Write/Read Data This field configures the data field in manual-command mode. In case of write transaction, the write data should be configured. In case of read transaction, the read data is stored.

(10) xSPI Command Manual Data 1 Buf n (XSPIm_CDD1BUFn) (n = 0 to 3)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 008Ch + n x 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	0h	RW	Write/Read Data This field configures the data field in manual-command mode. In case of write transaction, the write data should be configured. In case of read transaction, the read data is stored.

(11) xSPI Link Pattern Control Register 0 (XSPIm_LPCTL0)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0100h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	XD2VAL	-	-	XD2LEN[4:0]					XD1VAL	-	-	XD1LEN[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	XDPIN[1:0]		CSSEL	-	-	PATREQ
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	XD2VAL	0h	RW	XiP Disable pattern 2nd phase value This field selects the value of 2nd phase in XiP disable pattern. 0b: Low drive 1b: High drive
30, 29	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
28 to 24	XD2LEN[4:0]	0h	RW	XiP Disable pattern 2nd phase length This field selects the length of 2nd phase in XiP disable pattern. 00h: 0 cycles 01h: 1 cycle : 1Eh: 30 cycles 1Fh: 31 cycles
23	XD1VAL	0h	RW	XiP Disable pattern 1st phase value This field selects the value of 1st phase in XiP disable pattern. 0b: Low drive 1b: High drive
22, 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20 to 16	XD1LEN[4:0]	0h	RW	XiP Disable pattern 1st phase length This field selects the length of 1st phase in XiP disable pattern. The pattern with zero-length both 1st phase and 2nd phase should not be configured. 0h: 0 cycles 1h: 1 cycle : 1Eh: 30 cycles 1Fh: 31 cycles
15 to 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5, 4	XDPIN[1:0]	0h	RW	XiP Disable pattern pin This field selects the data output pins to transmit XiP Disable pattern. 00b: 1 pin 01b: 2 pins 10b: 4 pins 11b: 8 pins

Bit	Bit Name	Initial Value	R/W	Description
3	CSSEL	0h	RW	Chip select This field selects a target memory to issue a pattern. 0b: slave0 (CS0) 1b: slave1 (CS1)
2, 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	PATREQ	0h	RW	Pattern request This field requests to issue the pattern. When set to 1, it starts the pattern. It is cleared to 0 when the pattern completed. 0b: No request XiP Disable pattern 1b: Request XiP Disable pattern

(12) xSPI Link Pattern Control Register 1 (XSPIm_LPCTL1)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0104h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	RSTSU[2:0]			-	RSTWID[2:0]			-	-	RSTREP[1:0]		CSSEL	-	PATREQ[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
14 to 12	RSTSU[2:0]	0h	RW	Reset pattern data output setup time This field configures the number of setup cycles for data output based on the edge of CS in reset pattern. It needs enough setup time because xSPI slave samples any data at the rising edge of CS. This cycle of setup time should be less than the cycle of reset pattern width (RSTWID[2:0]). 000b: 1 cycle 001b: 2 cycles : 110b: 7 cycles 111b: 8 cycles
11	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10 to 8	RSTWID[2:0]	0h	RW	Reset pattern width This field configures the width of cycle in reset pattern and CS only pattern. It toggles CS with the configured cycle. 000b: 2 (= 2^1) cycles 001b: 4 (= 2^2) cycles : 110b: 128 (= 2^7) cycles 111b: 256 (= 2^8) cycles
7, 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5, 4	RSTREP[1:0]	0h	RW	Reset pattern repeat This field selects the repeating time to toggle CS from LOW to HIGH. 00b: 4 times (Specified on Reset Signaling Protocol) 01b: 5 times 10b: 6 times 11b: 7 times
3	CSSEL	0h	RW	Chip select This field selects a target memory to issue a pattern. 0b: slave0 (CS0) 1b: slave1 (CS1)
2	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PATREQ[1:0]	0h	RW	Pattern request This field requests to issue the pattern. When set to 01b or 10b, it starts the pattern. It is cleared to 00b when the pattern completed. 00b: No request 01b: Request Reset pattern 10b: Request CS only pattern 11b: Setting prohibited

(13) xSPI Link I/O Control Register (XSPIm_LIOCTL)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0108h
Initial Value : 0003_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RSTCS1	RSTCS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WPCS1	WPCS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
17	RSTCS1	1h	RW	Reset drive for slave1 This function is same as RSTCS0. 0b: Drive Low level 1b: Drive High level
16	RSTCS0	1h	RW	Reset drive for slave0 This field controls the value of xSPI reset port. It can be useful only for xSPI slave with reset port. 0b: Drive Low level 1b: Drive High level
15 to 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	WPCS1	1h	RW	WP drive for slave1 This function is same as WPCS0. 0b: Drive Low level 1b: Drive High level
0	WPCS0	1h	RW	WP drive for slave0 This field controls the value of Write Protect port. It can be useful only for xSPI slave with write protect port. 0b: Drive Low level 1b: Drive High level

(14) xSPI Command Calibration Control Register 0 CSn (XSPIm_CCCTL0CSn) (n = 0, 1)

This register has functions to control xSPI Master function.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0130h + n x 0020h
Initial Value : 1F00_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	CASFTEND[4:0]					-	-	-	CASFTSTA[4:0]				
Initial Value	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	CAITV[4:0]					-	-	-	-	-	-	CANOWR	CAEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
28 to 24	CASFTEND [4:0]	1Fh	RW	Calibration DS shift end value This field configures the end value of DS shift. It should be equal or more than the start value (CASFTSTA).
23 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20 to 16	CASFTSTA [4:0]	0h	RW	Calibration DS shift start value This field configures the start value of DS shift.
15 to 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12 to 8	CAITV[4:0]	0h	RW	Calibration interval This field configures the interval between calibration patterns. 00h: 2 (= 2^1) cycle wait 01h: 4 (= 2^2) cycle wait : 1Eh: 2,147,483,648 (= 2^{31}) cycle wait 1Fh: 4,294,967,296 (= 2^{32}) cycle wait
7 to 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	CANOWR	0h	RW	Calibration no write mode This field selects to omit write command in calibration sequence. It can be used for any slave device with fixed calibration pattern data. 0b: Calibration sequence with write command 1b: Calibration sequence without write command
0	CAEN	0h	RW	Automatic Calibration Enable This field enables the automatic calibration. When this field is set to 1, it transmits the calibration sequence periodically and adjusts the value of phase shift. When this field is set to 0 during the calibration sequence, it stops after completed ongoing calibration sequence, and then this field is cleared. 0b: Disable automatic calibration 1b: Enable automatic calibration

(15) xSPI Command Calibration Control Register 1 CSn (XSPIm_CCCTL1CSn) (n = 0, 1)

This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0134h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	CARDLATE[4:0]					-	-	-	CAWRLATE[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CADATASIZE[3:0]				CAADDSIZE[2:0]			CACMDSIZE [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
28 to 24	CARDLATE [4:0]	0h	RW	Read Latency cycle This field configures the latency cycle in calibration frame. 00h: No latency 01h: 1 cycle : 1Eh: 30 cycles 1Fh: 31 cycles
23 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20 to 16	CAWRLATE [4:0]	0h	RW	Write Latency cycle This field configures the latency cycle in calibration frame. 00h: No latency 01h: 1 cycle : 1Eh: 30 cycles 1Fh: 31 cycles
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8 to 5	CADATASIZE [3:0]	0h	RW	Write/Read Data Size This field configures the size of data field. In 8D-8D-8D, it should be configured with even byte. 0h: 1 byte 1h: 2 bytes : Eh: 15 bytes Fh: 16 bytes
4 to 2	CAADDSIZE [2:0]	0h	RW	Address size This field configures the size of address field. 000b: 0 bytes (No address phase) 001b: 1 byte 010b: 2 bytes 011b: 3 bytes 100b: 4 bytes Others: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
1, 0	CACMDSIZE [1:0]	0h	RW	Command Size This field configures the size of command field. In case of 8D-8D-8D, it should be fixed to 10b. It should not be configured both command size and address size to zero. 00b: 0 bytes (No command phase) 01b: 1 byte 10b: 2 bytes 11b: Setting prohibited

(16) xSPI Command Calibration Control Register 2 CSn (XSPIm_CCCTL2CSn) (n = 0, 1)

This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0138h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CARDCMD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAWRCMD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	CARDCMD [15:0]	0h	RW	Calibration pattern read command This field configures the calibration pattern read command.
15 to 0	CAWRCMD [15:0]	0h	RW	Calibration pattern write command This field configures the calibration pattern write command.

(17) xSPI Command Calibration Control Register 3 CSn (XSPIm_CCCTL3CSn) (n = 0, 1)

This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 013Ch + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAADD[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAADD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CAADD[31:0]	0h	RW	Calibration pattern address This field configures the calibration pattern address.

(18) xSPI Command Calibration Control Register 4 CSn (XSPIm_CCCTL4CSn) (n = 0, 1)

This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.

Access Size : 32 bits

Offset Address : <XSPIm_base> + 0140h + n x 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CADATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CADATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CADATA[31:0]	0h	RW	Calibration pattern data This field configures the calibration pattern data.

(19) xSPI Command Calibration Control Register 5 CSn (XSPIm_CCCTL5CSn) (n = 0, 1)

This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0144h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CADATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CADATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CADATA[31:0]	0h	RW	Calibration pattern data This field configures the calibration pattern data.

(20) xSPI Command Calibration Control Register 6 CSn (XSPIm_CCCTL6CSn) (n = 0, 1)

This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.

Access Size : 32 bits

Offset Address : <XSPIm_base> + 0148h + n x 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CADATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CADATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CADATA[31:0]	0h	RW	Calibration pattern data This field configures the calibration pattern data.

(21) xSPI Command Calibration Control Register 7 CSn (XSPIm_CCCTL7CSn) (n = 0, 1)

This register has functions to control xSPI Master function. It should be updated while Automatic Calibration Enable is disabled.

Access Size : 32 bits

Offset Address : <XSPIm_base> + 014Ch + n x 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CADATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CADATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CADATA[31:0]	0h	RW	Calibration pattern data This field configures the calibration pattern data.

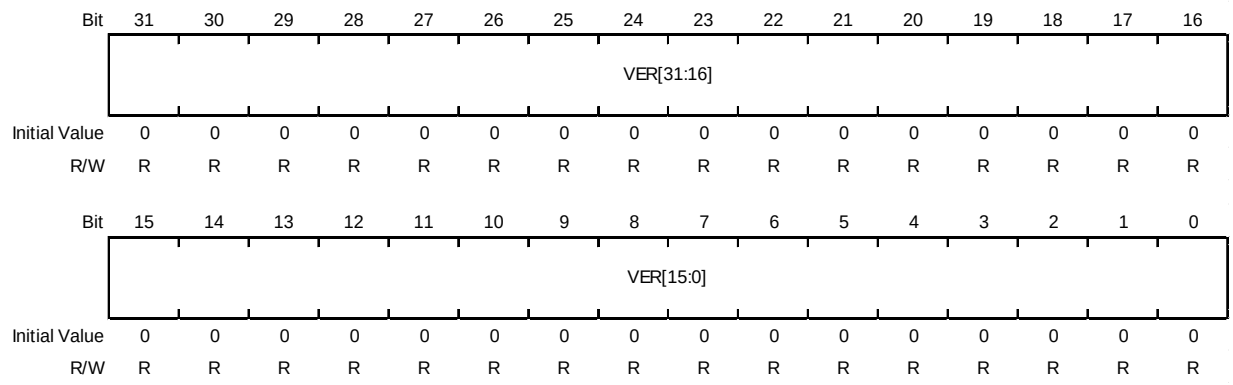
7.2.2.2.3 xSPI Status Registers

These registers monitor the status of xSPI Master.

(1) xSPI Version Register (XSPIm_VERSTT)

This register indicates the status of xSPI Master.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0180h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	VER[31:0]	0h	R	Version This field indicates this IP version.

(2) xSPI Common Status Register (XSPIm_COMSTT)

This register indicates the status of xSPI Master.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0184h
Initial Value : 0070_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	RSTOCS1	INTCS1	ECSCS1	-	RSTOCS0	INTCS0	ECSCS0
Initial Value	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	WRBUFNECH1	WRBUFNECH0	PBUFNECH1	PBUFNECH0	-	-	MEMACCCH1	MEMACCCH0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	-	All 0	R	Reserved Whenever it is read, 0b is read.
22	RSTOCS1	1h	R	RSTO monitor for slave1 This function is same as RSTOCS0. 0b: Low level 1b: High level
21	INTCS1	1h	R	INT monitor for slave1 This function is same as INTCS0. 0b: Low level 1b: High level
20	ECSCS1	1h	R	ECS monitor for slave1 This function is same as ECSCS0. 0b: Low level 1b: High level
19	-	0h	R	Reserved Whenever it is read, 0b is read.
18	RSTOCS0	0h	R	RSTO monitor for slave0 This field indicates the value of RSTO port. 0b: Low level 1b: High level
17	INTCS0	0h	R	INT monitor for slave0 This field indicates the value of INT port. 0b: Low level 1b: High level
16	ECSCS0	0h	R	ECS monitor for slave0 This field indicates the value of ECS port. 0b: Low level 1b: High level
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read.
7	WRBUFNECH1	0h	R	Write Buffer Not Empty for ch1 This function is same as WRBUFNECH0. 0b: Empty 1b: Not empty

Bit	Bit Name	Initial Value	R/W	Description
6	WRBUFNECH0	0h	R	Write Buffer Not Empty for ch0 This bit is only valid in the **7.2.3.6.3** and **7.2.3.6.7** flows. 0b: Empty 1b: Not empty
5	PBUFNECH1	0h	R	Prefetch Buffer Not Empty for ch1 0b: Empty 1b: Not empty
4	PBUFNECH0	0h	R	Prefetch Buffer Not Empty for ch0 0b: Empty 1b: Not empty
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read.
1	MEMACCCH1	0h	R	Memory access ongoing from ch1 0b: System bus bridge ch1 is not accessing memory. 1b: System bus bridge ch1 is accessing memory.
0	MEMACCCH0	0h	R	Memory access ongoing from ch0 0b: System bus bridge ch0b is not accessing memory. 1b: System bus bridge ch0 is accessing memory.

(3) xSPI Calibration Status Register CSn (XSPIm_CASTTCSn) (n = 0, 1)

This register indicates the status of xSPI Master.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0188h + n x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CASUC[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CASUC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CASUC[31:0]	0h	R	Calibration Success This field indicates the calibration success for each DS shift value. It is updated when each calibration sequence is completed. CASUC[x] indicates calibration success in DS shift value = x.

7.2.2.2.4 xSPI Interrupt Registers

These registers control the interrupt function of xSPI Master.

(1) xSPI Interrupt Status Register (XSPIm_INTS)

This register indicates the status of interrupt. The bits in this register are cleared to 0 when writing 1 on the corresponding bit of INTC register.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0190h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CASUCCS1	CASUCCS0	CAFAILCS1	CAFAILCS0	-	-	-	-	-	-	BUSERRCH1	BUSERRCH0	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	INTCS1	INTCS0	-	-	ECSCS1	ECSCS0	-	-	DSTOCS1	DSTOCS0	PERTO	INICMP	PATCMP	CMDCMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	CASUCCS1	0h	R	Calibration success for slave1 This function is same as CASUCCS0. 0b: No detection 1b: Detection
30	CASUCCS0	0h	R	Calibration success for slave0 This field is set to 1b when calibration succeeded. 0b: No detection 1b: Detection
29	CAFAILCS1	0h	R	Calibration failed for slave1 This function is same as CAFAILCS0. 0b: No detection 1b: Detection
28	CAFAILCS0	0h	R	Calibration failed for slave0 This field is set to 1b when calibration failed. 0b: No detection 1b: Detection
27 to 22	-	All 0	R	Reserved Whenever it is read, 0b is read.
21	BUSERRCH1	0h	R	System bus error for ch1 This field is set to 1b when system bus channel1 responds error.
20	BUSERRCH0	0h	R	System bus error for ch0 This field is set to 1b when system bus channel0 responds error.
19 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read.
13	INTCS1	0h	R	Interrupt detection for slave1 This function is same as INTCS0. 0b: No detection 1b: Detection

Bit	Bit Name	Initial Value	R/W	Description
12	INTCS0	0h	R	Interrupt detection for slave0 This field is set to 1b when the falling edge on INT port is detected. It can be useful only for xSPI slave with interrupt function. 0b: No detection 1b: Detection
11, 10	-	All 0	R	Reserved Whenever it is read, 0b is read.
9	ECSCS1	0h	R	ECC error detection for slave1 This function is same as ECSCS0. 0b: No detection 1b: Detection
8	ECSCS0	0h	R	ECC error detection for slave0 This field is set to 1b when the falling edge on ECS port is detected. It can be useful only for xSPI slave with ECC detection function. 0b: No detection 1b: Detection
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read.
5	DSTOCS1	0h	R	DS timeout for slave1 This function is same as DSTOCS0. 0b: No detection 1b: Detection
4	DSTOCS0	0h	R	DS timeout for slave0 This field is set to 1b when DS is lost in read transaction with using DS. It means not receiving the data during expected read phase. In this case, xSPI master stops the read transaction and the following transaction. This error may issue an error response to system bus. 0b: No detection 1b: Detection
3	PERTO	0h	R	Periodic transaction timeout This field is set to 1b when the read value does not match with the expected value in periodic manual-command mode. 0b: No detection 1b: Detection
2	INICMP	0h	R	Initial Sequence Completed This field is set to 1b when the initial sequence is completed. 0b: No detection 1b: Detection
1	PATCMP	0h	R	Pattern Completed This field is set to 1b when the requested pattern is completed. 0b: No detection 1b: Detection
0	CMDCMP	0h	R	Command Completed This field is set to 1b when the requested manual-command is completed. In direct manual-command, it means all transactions completed. In periodic manual-command, it means the read data matches with the expected data. 0b: No detection 1b: Detection

(2) xSPI Interrupt Clear Register (XSPIm_INTC)

This register clears the status of interrupt.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0194h
Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CASUCCS1C	CASUCCS0C	CAFAILCS1C	CAFAILCS0C	-	-	-	-	-	-	BUSERRCH1C	BUSERRCH0C	-	-	-	-
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	INTCS1C	INTCS0C	-	-	ECSCS1C	ECSCS0C	-	-	DSTOCS1C	DSTOCS0C	PERTOC	INICMPC	PATCMPC	CMDCMPC
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31	CASUCCS1C	x	W	The read value is undefined Calibration success for slave1 interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
30	CASUCCS0C	x	W	The read value is undefined Calibration success for slave0 interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
29	CAFAILCS1C	x	W	The read value is undefined Calibration failed for slave1 interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
28	CAFAILCS0C	x	W	The read value is undefined Calibration failed for slave0 interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
27 to 22	-	x	W	Reserved Whenever it is read, undefined value is read. The written value should always be 0b.
21	BUSERRCH1C	x	W	The read value is undefined System bus error for ch1 interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
20	BUSERRCH0C	x	W	The read value is undefined System bus error for ch0 interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
19 to 14	-	x	W	Reserved Whenever it is read, undefined value is read. The written value should always be 0b.
13	INTCS1C	x	W	The read value is undefined Interrupt detection for slave1 interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status

Bit	Bit Name	Initial Value	R/W	Description
12	INTCS0C	x	W	The read value is undefined Interrupt detection for slave0 interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
11, 10	-	x	W	Reserved Whenever it is read, undefined value is read. The written value should always be 0b.
9	ECSCS1C	x	W	The read value is undefined ECC error detection for slave1 interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
8	ECSCS0C	x	W	The read value is undefined ECC error detection for slave0 interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
7, 6	-	x	W	Reserved Whenever it is read, undefined value is read. The written value should always be 0b.
5	DSTOCS1C	x	W	The read value is undefined DS timeout for slave1 interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
4	DSTOCS0C	x	W	The read value is undefined DS timeout for slave0 interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
3	PERTOC	x	W	The read value is undefined Periodic transaction timeout interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
2	INICMPC	x	W	The read value is undefined Initial Sequence Completed interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
1	PATCMPC	x	W	The read value is undefined Pattern Completed interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status
0	CMDCMPC	x	W	The read value is undefined Command Completed interrupt clear 0b: No change in interrupt status 1b: Clear interrupt status

(3) xSPI Interrupt Enable Register (XSPIm_INTE)

This register enables the interrupt.

Access Size : 32 bits
Offset Address : <XSPIm_base> + 0198h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CASUCCS1E	CASUCCS0E	CAFAILCS1E	CAFAILCS0E	-	-	-	-	-	-	BUSERRCH1E	BUSERRCH0E	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	INTCS1E	INTCS0E	-	-	ECSCS1E	ECSCS0E	-	-	DSTOCS1E	DSTOCS0E	PERTOE	INICMPE	PATCMPE	CMDCMPE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CASUCCS1E	0h	RW	Calibration success for slave1 interrupt enable 0b: Disabled 1b: Enabled
30	CASUCCS0E	0h	RW	Calibration success for slave0 interrupt enable 0b: Disabled 1b: Enabled
29	CAFAILCS1E	0h	RW	Calibration failed for slave1 interrupt enable 0b: Disabled 1b: Enabled
28	CAFAILCS0E	0h	RW	Calibration failed for slave0 interrupt enable 0b: Disabled 1b: Enabled
27 to 22	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
21	BUSERRCH1E	0h	RW	System bus error for ch1 interrupt enable 0b: Disabled 1b: Enabled
20	BUSERRCH0E	0h	RW	System bus error for ch0 interrupt enable 0b: Disabled 1b: Enabled
19 to 14	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
13	INTCS1E	0h	RW	Interrupt detection for slave1 interrupt enable 0b: Disabled 1b: Enabled
12	INTCS0E	0h	RW	Interrupt detection for slave0 interrupt enable 0b: Disabled 1b: Enabled
11, 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	ECSCS1E	0h	RW	ECC error detection for slave1 interrupt enable 0b: Disabled 1b: Enabled

Bit	Bit Name	Initial Value	R/W	Description
8	ECSCS0E	0h	RW	ECC error detection for slave0 interrupt enable 0b: Disabled 1b: Enabled
7, 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	DSTOCS1E	0h	RW	DS timeout for slave1 interrupt enable 0b: Disabled 1b: Enabled
4	DSTOCS0E	0h	RW	DS timeout for slave0 interrupt enable 0b: Disabled 1b: Enabled
3	PERTOE	0h	RW	Periodic transaction timeout interrupt enable 0b: Disabled 1b: Enabled
2	INICMPE	0h	RW	Initial Sequence Completed interrupt enable 0b: Disabled 1b: Enabled
1	PATCMPE	0h	RW	Pattern Completed interrupt enable 0b: Disabled 1b: Enabled
0	CMDCMPE	0h	RW	Command Completed interrupt enable 0b: Disabled 1b: Enabled

7.2.3 Operation

xSPI Master interface has the functions to issue the transaction for external memory with xSPI Slave interface. It allows to write to registers in external memory or read from it.

This xSPI Master mainly has two modes to issue the transaction. One is a manual-command mode; Software configures all fields of xSPI frame and starts the transaction by software request. The other is a memory-mapping mode; it automatically converts system bus access for pre-configured memory area into xSPI transaction. It enables to access from system bus to external memory area outside of chip via xSPI bus.

This section describes the xSPI bus operation, the direct control of xSPI frame (manual-command), the control of memory access (memory-mapping), the error operation, and the flow to operation.

7.2.3.1 xSPI Bus

This section describes the xSPI bus operation.

7.2.3.1.1 Supported protocol mode

This xSPI Master supports various protocol modes. It is configured by Protocol mode bits (LIOCFGCSn.PRTMD[9:0]). The table below shows the summary of protocol modes.

Table 7.2-6 Supported Protocol Mode

Protocol Mode	Function	PRTMD[9:0]	Note
1S-1S-1S	Command, Address, and Data fields are transferred at SDR with using 1 data input pin and 1 data output pin. Read data is sampled with CK.	0x000	Specified by xSPI protocol
4S-4D-4D	Command field is transferred at SDR with using 4 data pins. Address and Data fields are transferred at DDR with using 4 data pins. Read data is sampled with DS.	0x3B2	Specified by xSPI protocol
8D-8D-8D	Command, Address, and Data fields are transferred at DDR with using 8 data pins. Read data is sampled with DS.	0x3FF	Specified by xSPI protocol
1S-2S-2S	Command field is transferred at SDR with using 1 data pin. Address and Data fields are transferred at SDR with using 2 data pins. Read data is sampled with CK.	0x048	—
2S-2S-2S	Command, Address, and Data fields are transferred at SDR with using 2 data pins. Read data is sampled with CK.	0x049	—
1S-4S-4S	Command field is transferred at SDR with using 1 data pin. Address and Data fields are transferred at SDR with using 4 data pins. Read data is sampled with CK.	0x090	—
4S-4S-4S	Command, Address, and Data fields are transferred at SDR with using 4 data pins. Read data is sampled with CK.	0x092	—

Note: In case of XiP mode enable, XiP code is inserted in Latency field. It is valid only for memory-mapping mode.

The table below shows a description of the main internal signals.

Table 7.2-7 Internal Signals

Signal Name	I/O	Function
clk_spi	I	xSPI control clock
spi_ck	O	xSPI Clock (Same as XSPIm_CKP pin)
spi_cs0	O	xSPI Chip Select for slave0 (Same as XSPIm_CS0# pin)
spi_cs1	O	xSPI Chip Select for slave1 (Same as XSPIm_CS1# pin)
spi_doe[7:0]	O	xSPI Data Output Enable
spi_do[7:0]	O	xSPI Data Output
spi_di[7:0]	I	xSPI Data Input
spi_dsoe	O	xSPI Data Strobe Output Enable
spi_dso	O	xSPI Data Strobe Output
spi_dsi	I	xSPI Data Strobe Input

The bytes of Command and Address fields are transferred in highest order to lowest order sequence. The sequential bytes of Data field are transferred in lowest address to highest address order. In case of using multiple pins, the least significant bit of each byte is placed on spi_do/di[0] with each higher order bit on the successively higher numbered spi_do/di signals.

The figure below shows timing-chart for 1S-1S-1S protocol mode. The spi_do[0] signal is used for output data and the spi_di[1] signal is used for input data.

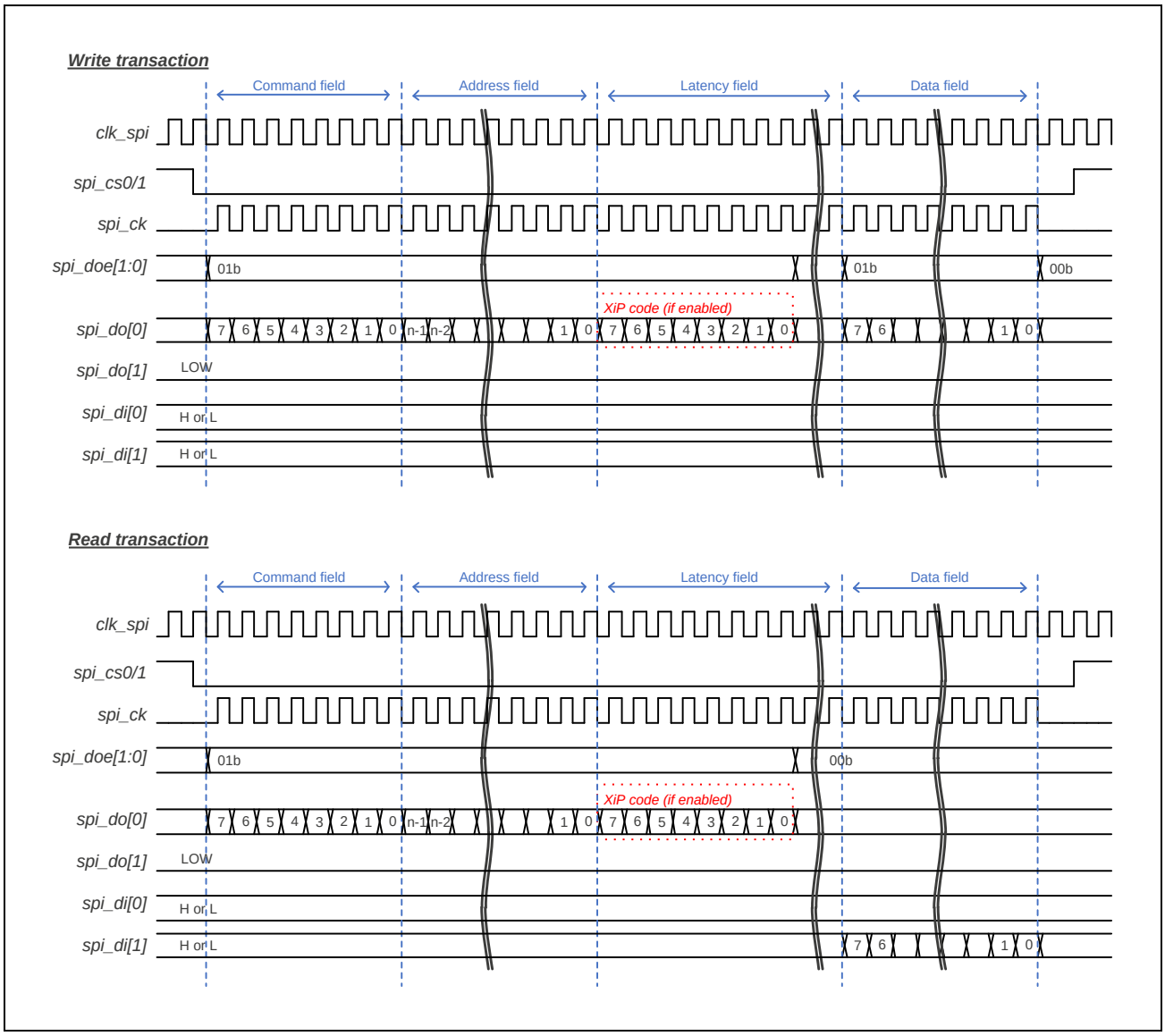


Figure 7.2-2 1S-1S-1S Timing-Chart

The figure below shows timing-chart for 1S-2S-2S protocol mode.

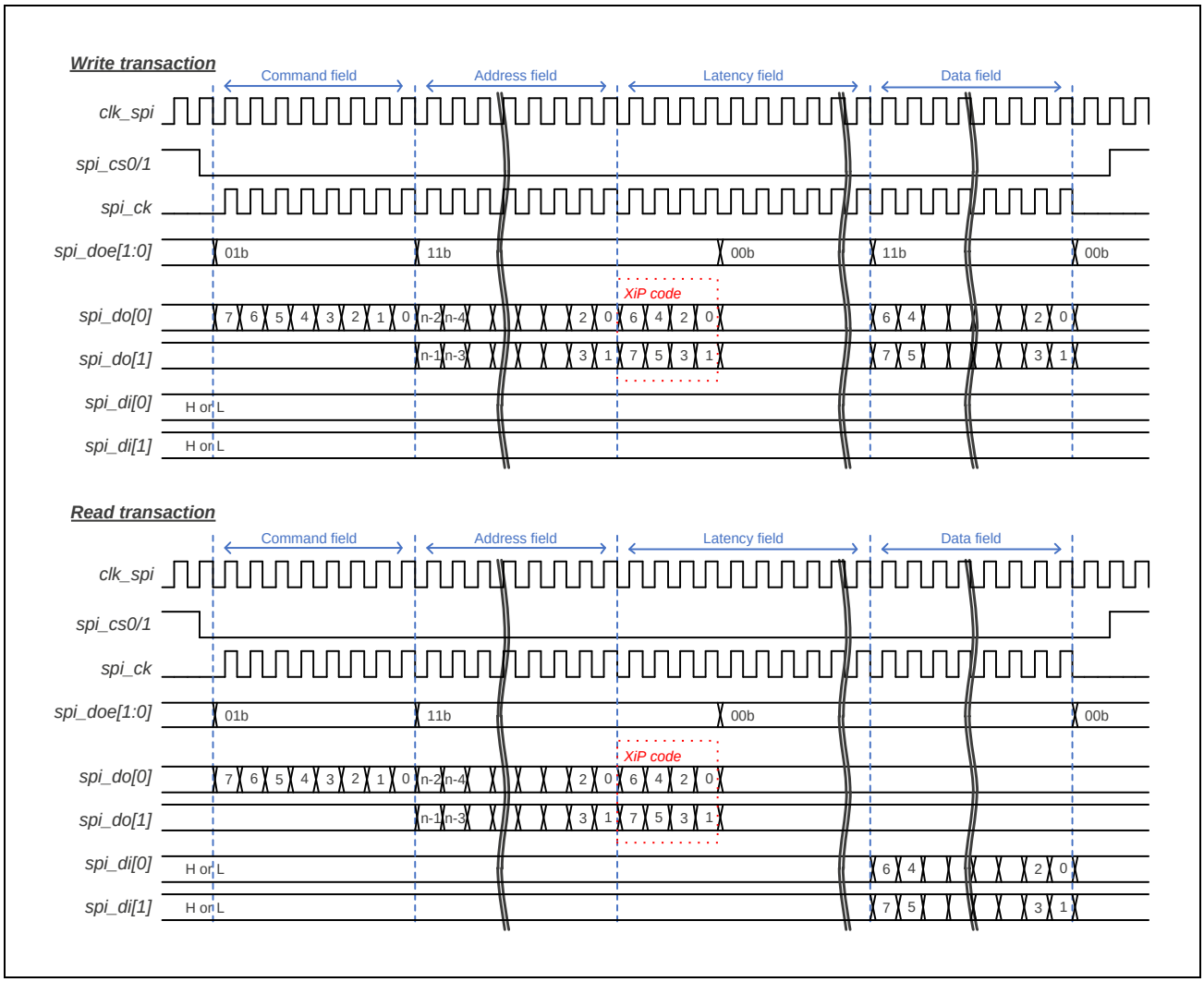


Figure 7.2-3 1S-2S-2S Timing-Chart

The figure below shows timing-chart for 4S-4D-4D protocol mode.

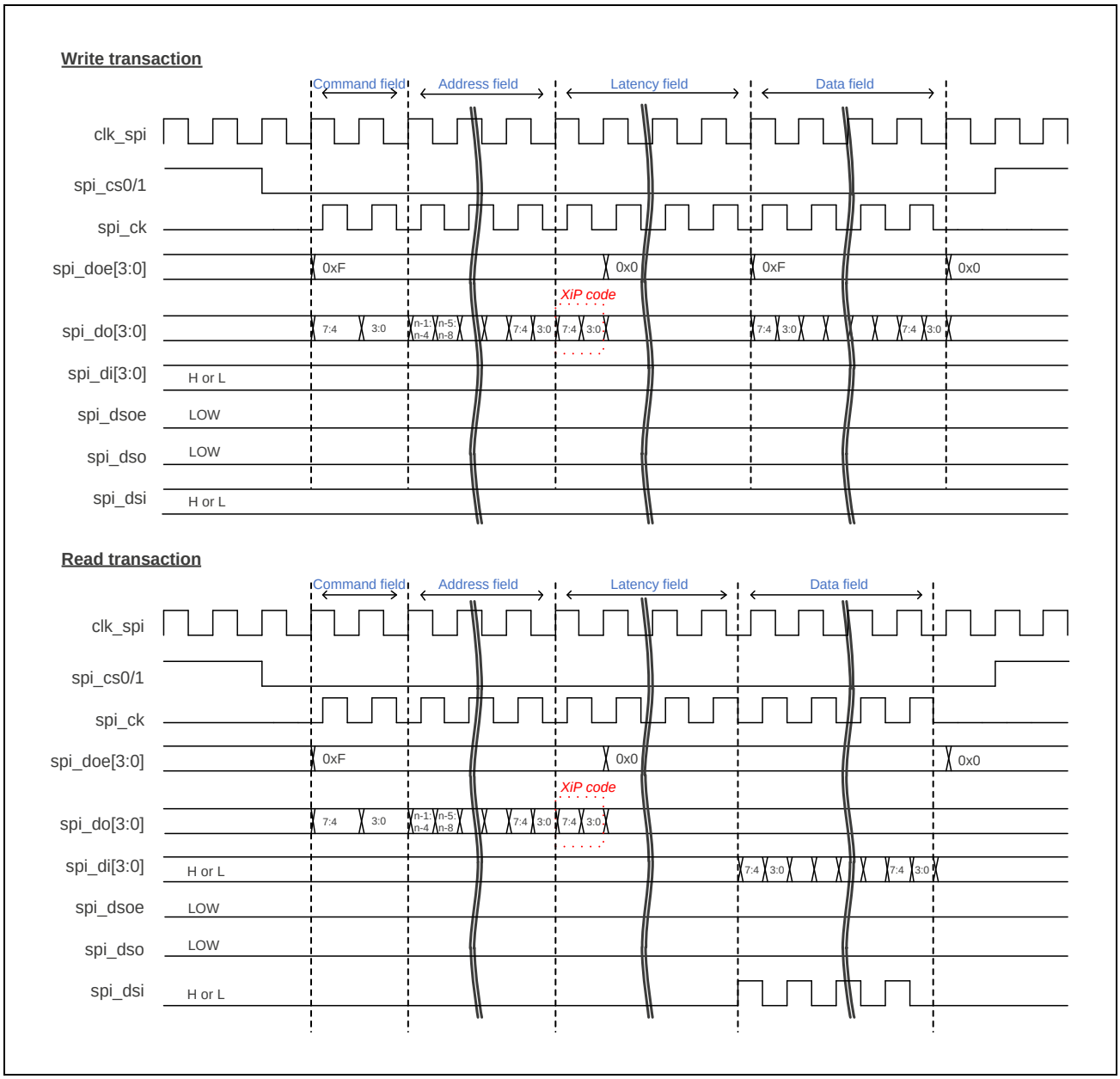


Figure 7.2-4 4S-4D-4D Timing-Chart

The figure below shows timing-chart for 8D-8D-8D profile 1.0 protocol mode. In 8D-8D-8D, the data is always transmitted with byte-pair. In case each field is odd byte, last one byte is padded with invalid data.

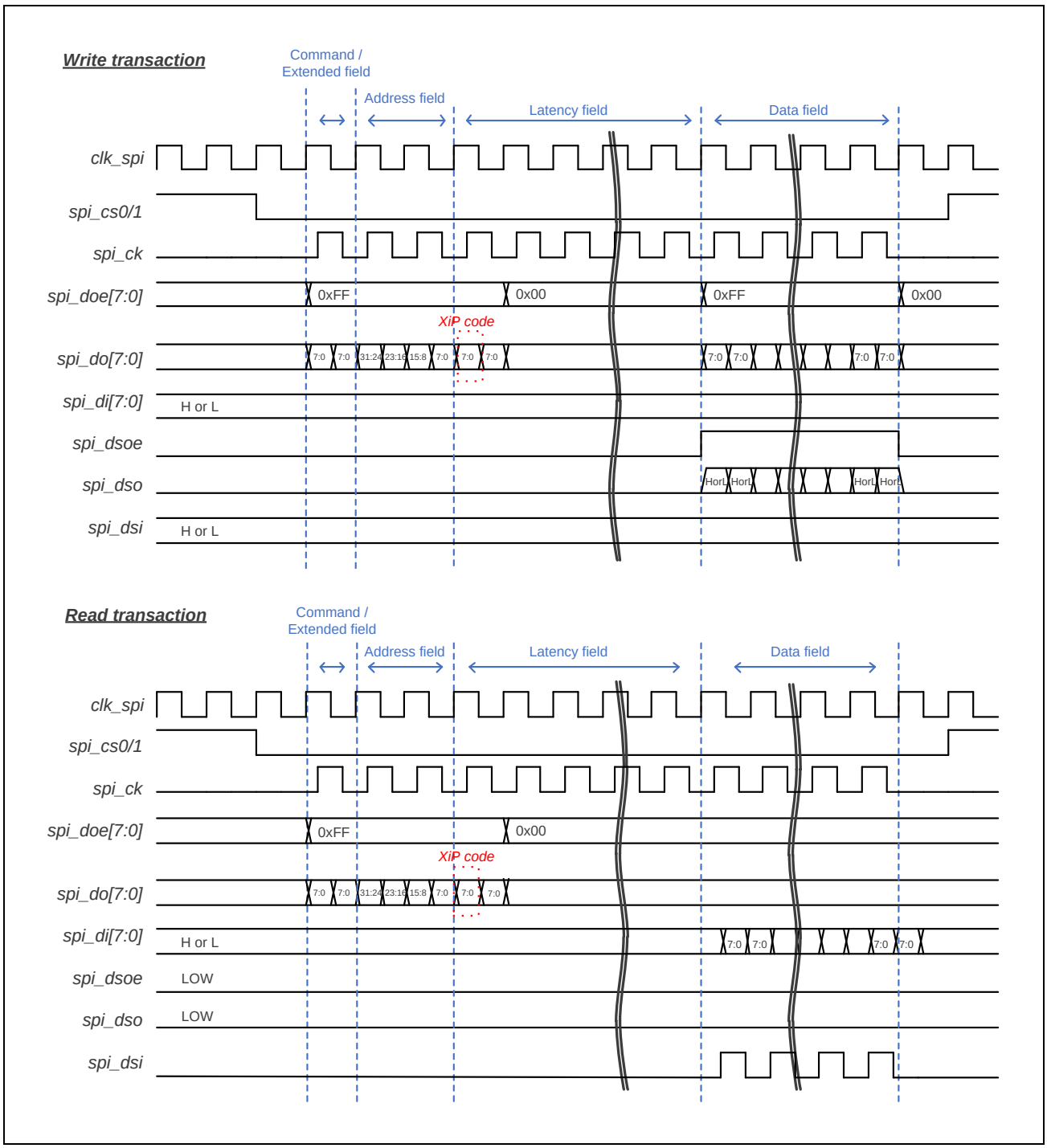


Figure 7.2-5 8D-8D-8D profile 1.0 Timing-Chart

The figure below shows timing-chart for 8D-8D-8D profile 2.0 protocol mode. In 8D-8D-8D, the data is always transmitted with byte-pair. In case each field is odd byte, last one byte is padded with invalid data.

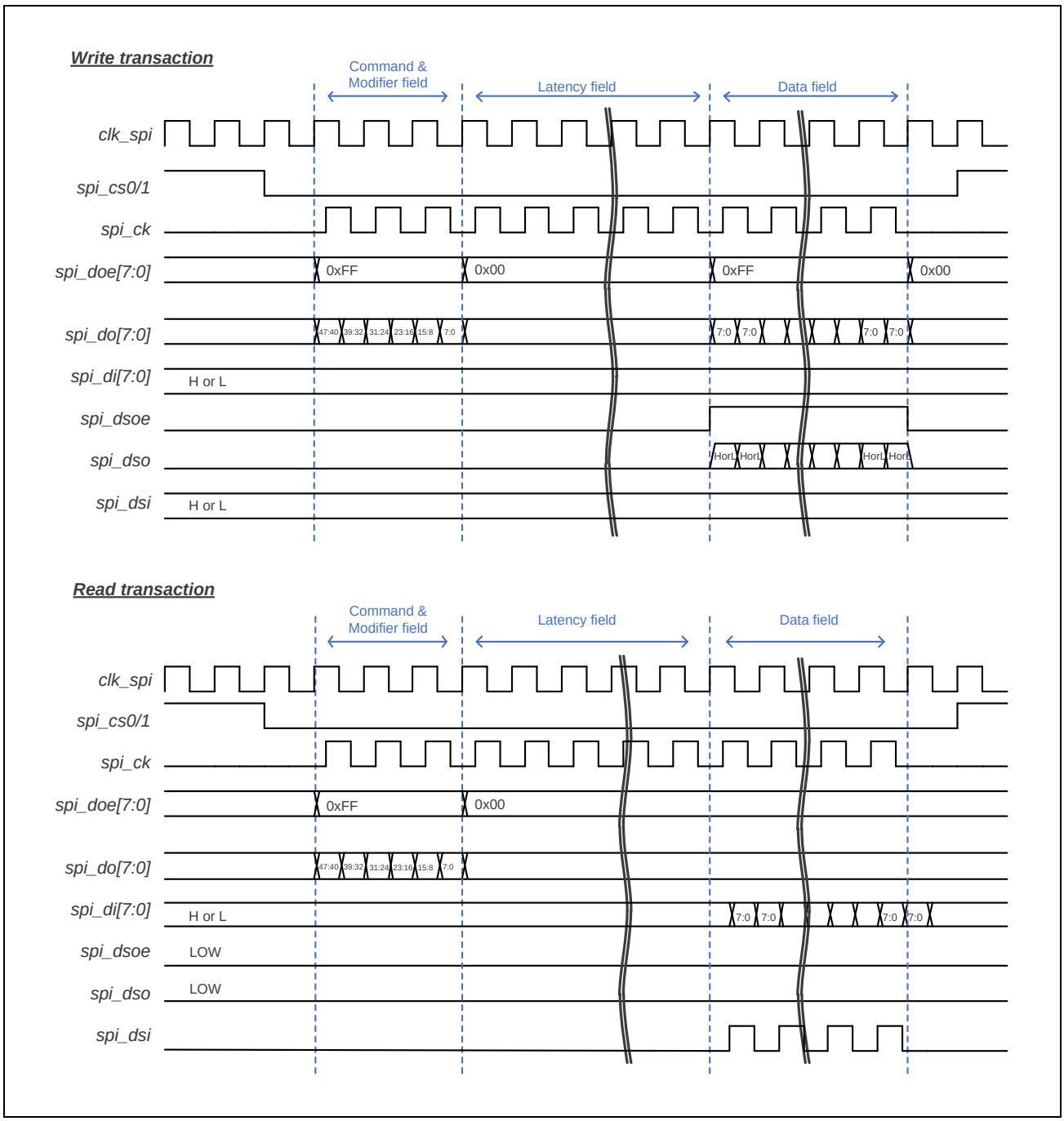


Figure 7.2-6 8D-8D-8D Profile 2.0 Timing-Chart

7.2.3.1.2 xSPI frame interval

The interval between xSPI frames is configured with CS minimum idle term bits (LIOCFGCSn.CSMIN[3:0]). It depends on the specification of xSPI slave device.

7.2.3.1.3 xSPI signals timing control

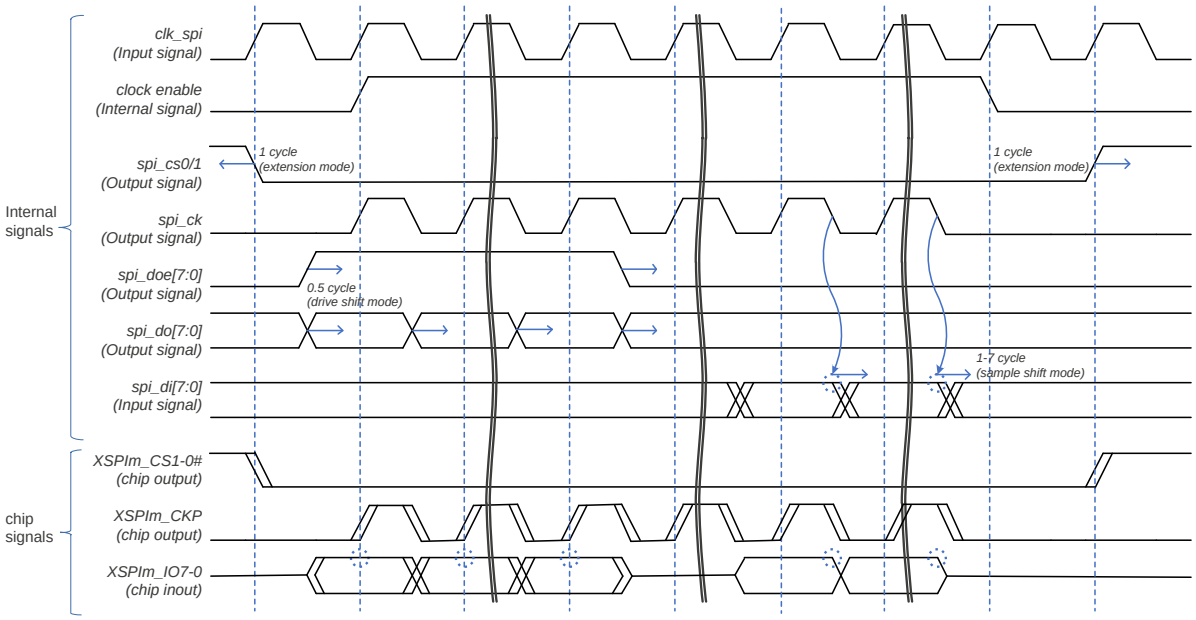
This xSPI Master supports both SDR and DDR. And it is possible to sample input data with Data-Strobe (DS) signal at SDR. For support of various modes and easy implementation, this xSPI Master could adjust the timing to drive/sample xSPI interface signals statically. The table below shows the summary of xSPI Interface signal timing control.

Table 7.2-8 Summary of xSPI Signals Timing Control

Signal	Mode	Default Operation	Timing Control (n = 0, 1)
CS (spi_cs) drive	—	Asserting 1 cycle before the rising-edge of first spi_ck	1 cycle extension for asserting with LIOCFGCSn.CSASTEX bit
		Negating 1.5 cycle after the falling-edge of last spi_ck	1 cycle extension for negating with LIOCFGCSn.CSNEGEX bit
CK (spi_ck) drive	SDR without DS	Reference point	—
	SDR with DS		
	DDR with DS	Reference point	—
DO/DOE (spi_do/doe) drive	SDR without DS	Falling edge of clk_spi	0 or 0.5 cycle shift with LIOCFGCSn.SDRDRV bit
	SDR with DS		
	DDR with DS	Both edges of clk_spi	—
DI (spi_di) sample	SDR without DS	Falling edge of spi_ck on expected data size	0 to 7 cycle shift (1 cycle unit) with LIOCFGCSn.SDRSMPSFT[3:0] bits 0 or 0.5 cycle shift with LIOCFGCSn.SDRSMPMD bit
	SDR with DS	Falling edge of spi_dsi signal on expected data size	Sample at rising edge with LIOCFGCSn.SDRSMPMD bit 0 to 1 cycle phase shift with WRAPCFG.DSSFTCSn[4:0] bits 0 to 7 cycle extension with LIOCFGCSn.DDRSMPEX[3:0] bits
	DDR with DS	Both edges of spi_dsi signal on expected data size	0 to 1 cycle phase shift with WRAPCFG.DSSFTCSn[4:0] bits 0 to 7 cycle extension with LIOCFGCSn.DDRSMPEX[3:0] bits

Note: In DDR on xSPI protocol, CK or DS should be aligned for center of data. It means to shift the phase by 0.25 cycle (90 degrees). This xSPI master supports to adjust this phase depending on the usage conditions.

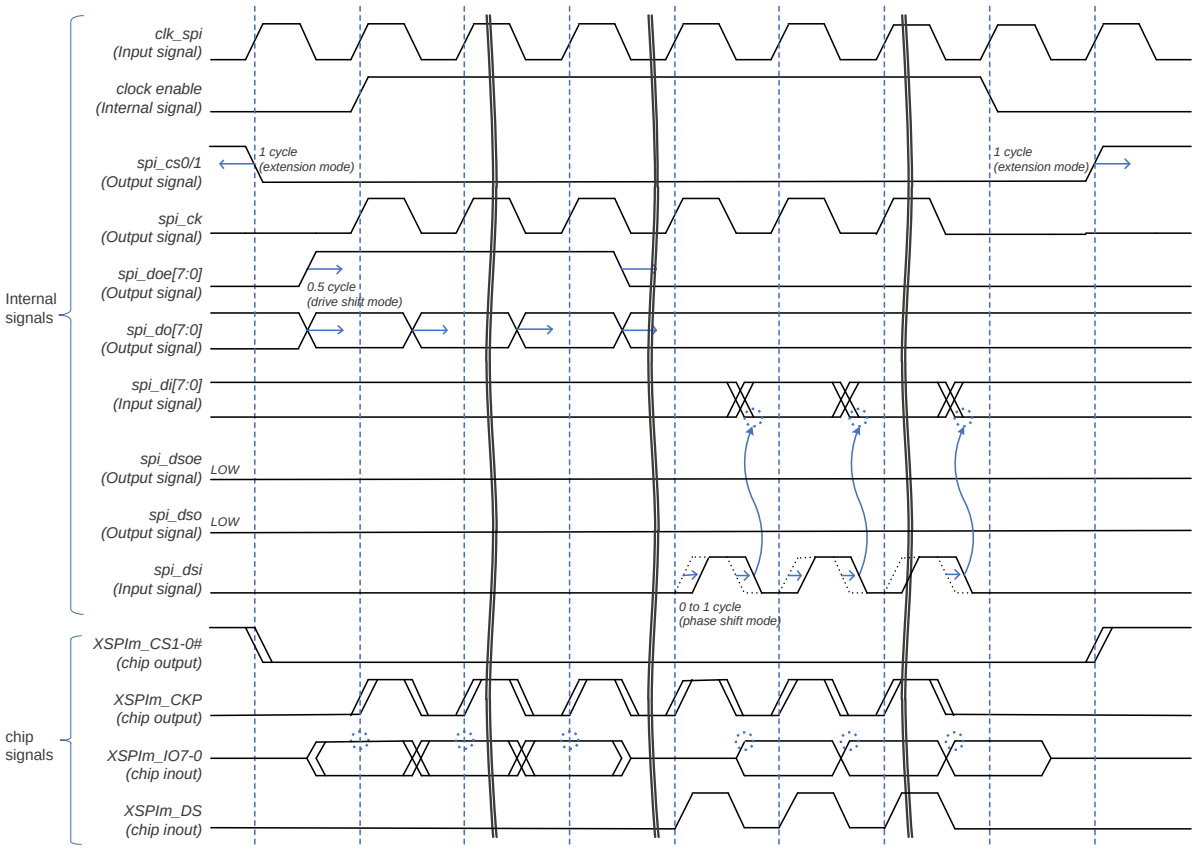
The figure below shows the default operation and timing control for SDR without DS.



Note: m = 0

Figure 7.2-7 Timing Control for SDR without DS

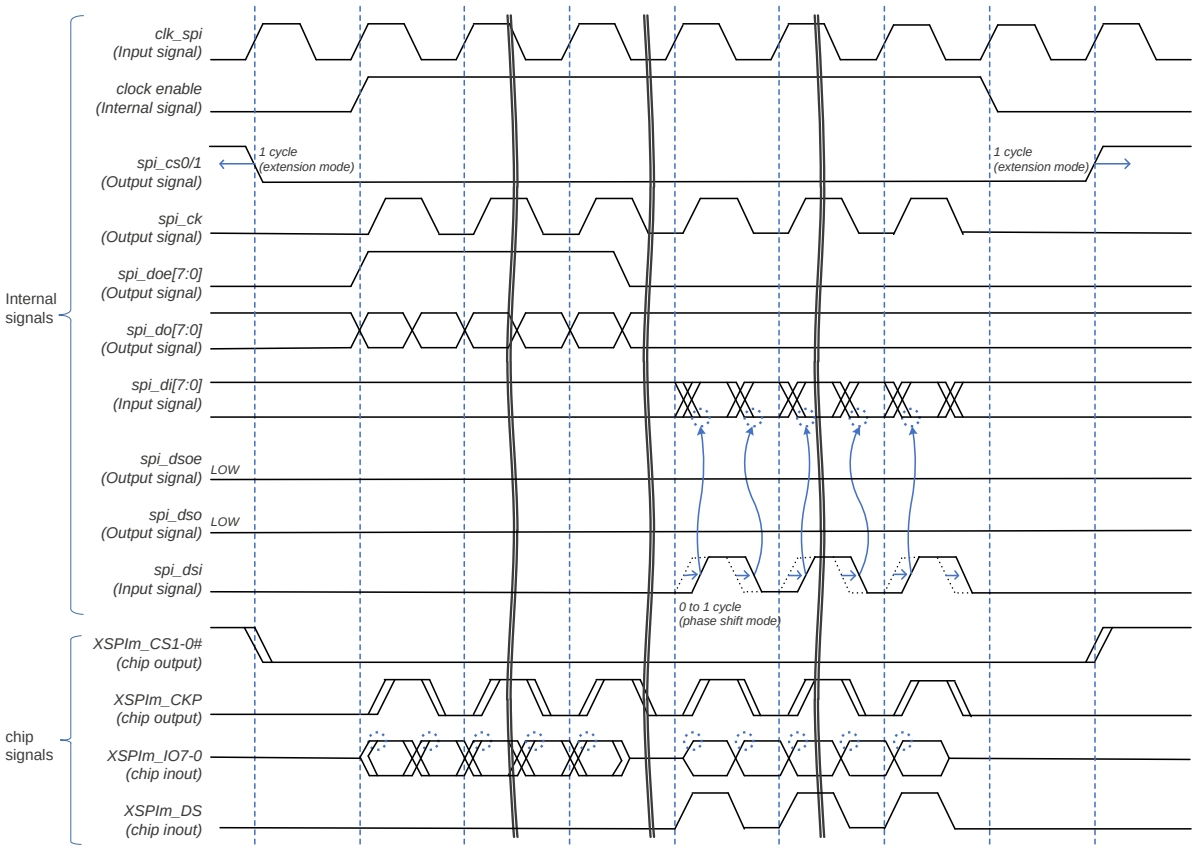
The figure below shows the default operation and timing control for SDR with DS.



Note: m = 0

Figure 7.2-8 Timing Control for SDR with DS

The figure below shows the default operation and timing control for DDR with DS.



Note: m = 0

Figure 7.2-9 Timing Control for DDR with DS

7.2.3.1.4 Automatic calibration

This xSPI master supports the function to adjust DS shift value (WRAPCFG.DSSFTCSn) automatically. When this function is enabled (CCCTL0CSn.CAEN = 1b), this xSPI master transmits the calibration sequence periodically and adjusts the value of phase shift.

When all read compare is mismatched in read transaction during the calibration sequence, the calibration fail bit (INTS.CAFAILCSn) is asserted and DS shift value is not updated. When at least one read compare is matched, the calibration success bit (INTS.CASUCCSn) is asserted and DS shift value is updated. The result of each DS shift value could be monitored by Calibration Status register (CASTTCSn).

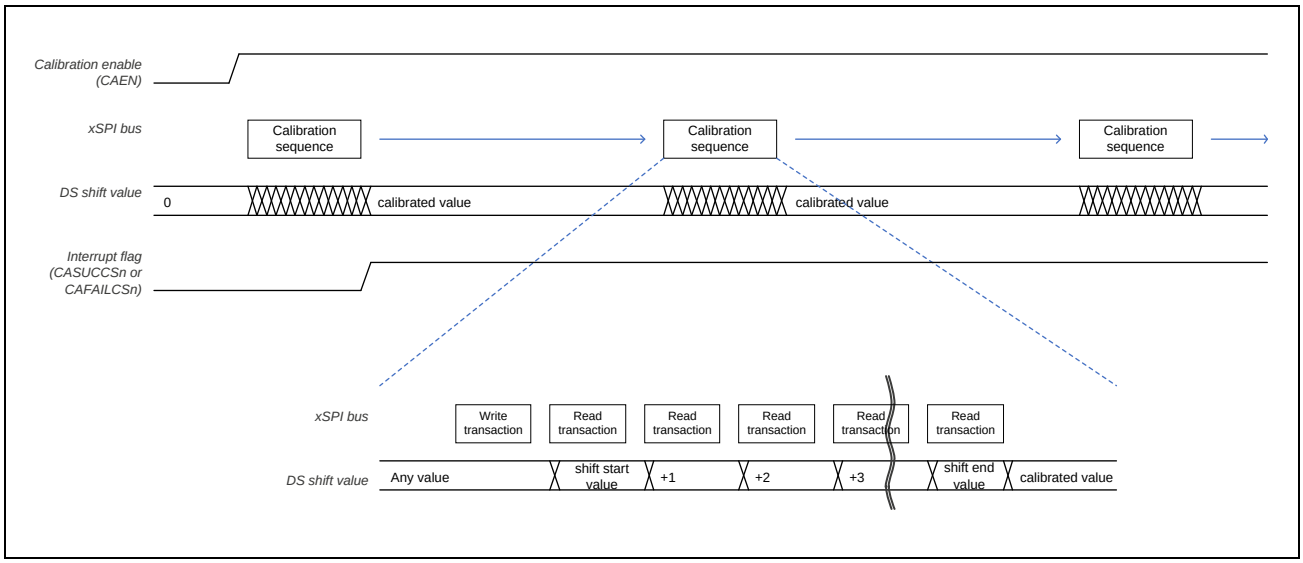


Figure 7.2-10 Automatic Calibration

7.2.3.2 Manual-command

This section describes the manual-command mode. The manual-command has two functional modes: direct mode and periodic mode.

7.2.3.2.1 Direct mode

This mode sequentially can issue up to four xSPI transactions configured and requested by Software. A series of transaction can be issued by a transaction request (CDCTL0.TRREQ = 1b with PERMD = 0b). The number of transactions (CDCTL0.TRNUM[1:0]) can be configured up to 4. It can be used to change the mode or read the status of xSPI slave device.

The table below shows the configured register bits for direct manual-command. The operating flow is illustrated in **Figure 7.2-25**.

Table 7.2-9 Manual-Command Configuration for Direct Mode

Transaction	Transaction Type	Command	Command Size	Address	Address Size	Data (x = 0, 1)	Data Size	Latency Cycle
1st Transaction	CDTBUF0.TRTYPE	CDTBUF0.CMD[15:0]	CDTBUF0.CMDSIZE[1:0]	CDABUF0.ADD[31:0]	CDTBUF0.ADDSIZE[2:0]	CDDxBUF0.DATA[31:0]	CDTBUF0.DATASIZE[3:0]	CDTBUF0.LATE[4:0]
2nd Transaction	CDTBUF1.TRTYPE	CDTBUF1.CMD[15:0]	CDTBUF1.CMDSIZE[1:0]	CDABUF1.ADD[31:0]	CDTBUF1.ADDSIZE[2:0]	CDDxBUF1.DATA[31:0]	CDTBUF1.DATASIZE[3:0]	CDTBUF1.LATE[4:0]
3rd Transaction	CDTBUF2.TRTYPE	CDTBUF2.CMD[15:0]	CDTBUF2.CMDSIZE[1:0]	CDABUF2.ADD[31:0]	CDTBUF2.ADDSIZE[2:0]	CDDxBUF2.DATA[31:0]	CDTBUF2.DATASIZE[3:0]	CDTBUF2.LATE[4:0]
4th Transaction	CDTBUF3.TRTYPE	CDTBUF3.CMD[15:0]	CDTBUF3.CMDSIZE[1:0]	CDABUF3.ADD[31:0]	CDTBUF3.ADDSIZE[2:0]	CDDxBUF3.DATA[31:0]	CDTBUF3.DATASIZE[3:0]	CDTBUF3.LATE[4:0]

7.2.3.2.2 Periodic mode

This mode periodically issues an xSPI read transaction configured and requested by Software. And it can compare the read value up to 4 bytes with expected value. The transaction is issued by a transaction request (CDCTL0.TRREQ = 1b with PERMD = 1b). It can be used to alternate the status polling operation of xSPI slave device.

The periodic term is configured in Periodic transaction interval bits (CDCTL0.PERITV[4:0]). The expected value is configured in Periodic transaction expected and masked value bits (CDCTL1.PEREXP[31:0] and CDCTL2.PERMSK[31:0]). The table below shows the configured register bits for periodic manual-command. The operating flow is illustrated in **Figure 7.2-26**.

Table 7.2-10 Manual-Command Configuration for Periodic Mode

Transaction	Transaction Type	Command	Command Size	Address	Address Size	Data (x = 0, 1)	Data Size	Latency Cycle
Read Transaction	CDTBUF0.TRTYPE = 0b	CDTBUF0.CMD[15:0]	CDTBUF0.CMDSIZE[1:0]	CDABUF0.ADD[31:0]	CDTBUF0.ADDSIZE[2:0]	CDDxBUF0.DATA[31:0]	CDTBUF0.DATASIZE[3:0]	CDTBUF0.LATE[4:0]

7.2.3.3 Memory-mapping

This section describes the memory-mapping mode. This mode automatically converts system bus access for pre-configured memory area into xSPI transaction.

7.2.3.3.1 Configuration

In this operation, the payload of address and data field are delivered from system bus signals. The information of command field and size are delivered from the configured register bits. The table below shows the register bits configured for memory- mapping.

Table 7.2-11 Memory-Mapping Configuration for Memory Area Access (n = 0, 1)

AHB Transaction	Format Change Mode	Command	Command Size	Address Size	Latency Cycle
Write for slave n memory area	Normal	CMCFG2CSn.WRCMD [15:8]	1 byte	CMCFG0CSn.ADDSIZE[1:0]	CMCFG2CSn.WRLATE[4:0]
	8D-8D-8D profile 1.0	CMCFG2CSn.WRCMD [15:0]	2 bytes		
	8D-8D-8D profile 2.0 Command Modifier	CMCFG2CSn.WRCMD [15:8]	1 byte	5 bytes	
	8D-8D-8D profile 2.0 Extended Command Modifier	CMCFG2CSn.WRCMD [15:13]	3 bits	45 bits	
Read for slave n memory area	Normal	CMCFG1CSn.RDCMD [15:8]	1 byte	CMCFG0CSn.ADDSIZE[1:0]	CMCFG1CSn.RDLATE[4:0]
	8D-8D-8D profile 1.0	CMCFG1CSn.RDCMD [15:0]	2 bytes		
	8D-8D-8D profile 2.0 Command Modifier	CMCFG1CSn.RDCMD [15:8]	1 byte	5 bytes	
	8D-8D-8D profile 2.0 Extended Command Modifier	CMCFG1CSn.RDCMD [15:13]	3 bits	45 bits	

Note: The MSByte of Address can be replaced with Address Replace Enable and Code bits (CMCFG0CSn.ADDRPEN[7:0]/ADDRPCD[7:0]).

7.2.3.3.2 Write access operation

When write access to memory area from system bus is accepted, this xSPI Master stores all payload data in internal bridge buffer and then issues a write transaction to xSPI slave. The figure below shows the operation summary.

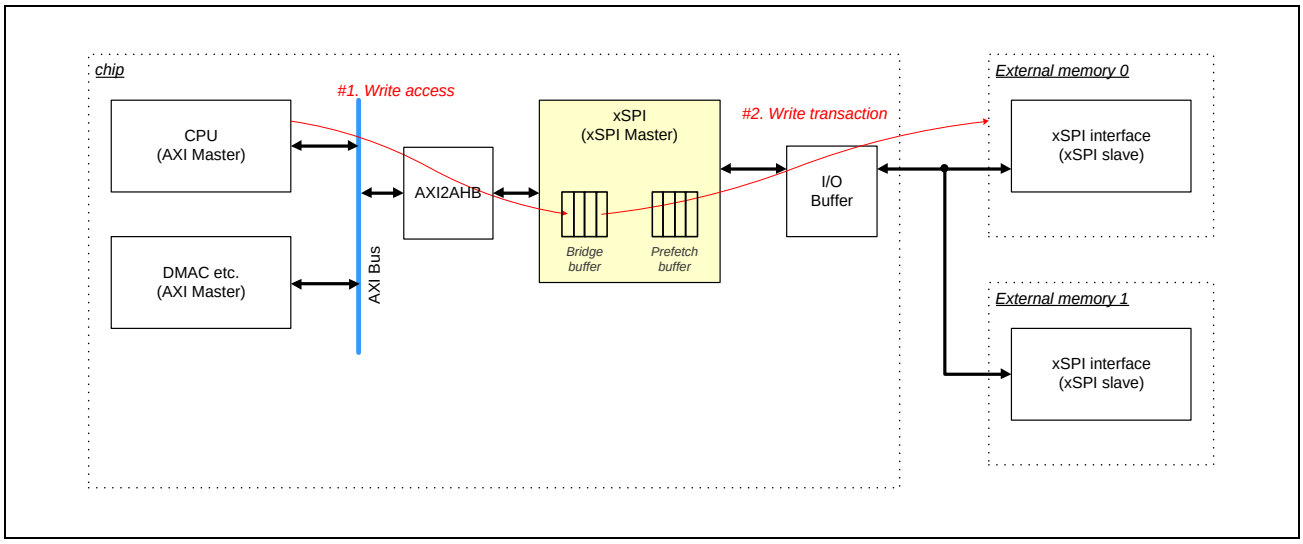


Figure 7.2-11 Write Access for Memory Area

The operation of xSPI bus changes depending on the system bus's burst type. When the burst type is single type or incremental type, one system bus transaction triggers one xSPI frame. When the burst type is wrap type and CMCFG0CSn.WPBSTMD is 0, one system bus transaction triggers two xSPI frames. The figure below shows the relationship between AXI and xSPI frame.

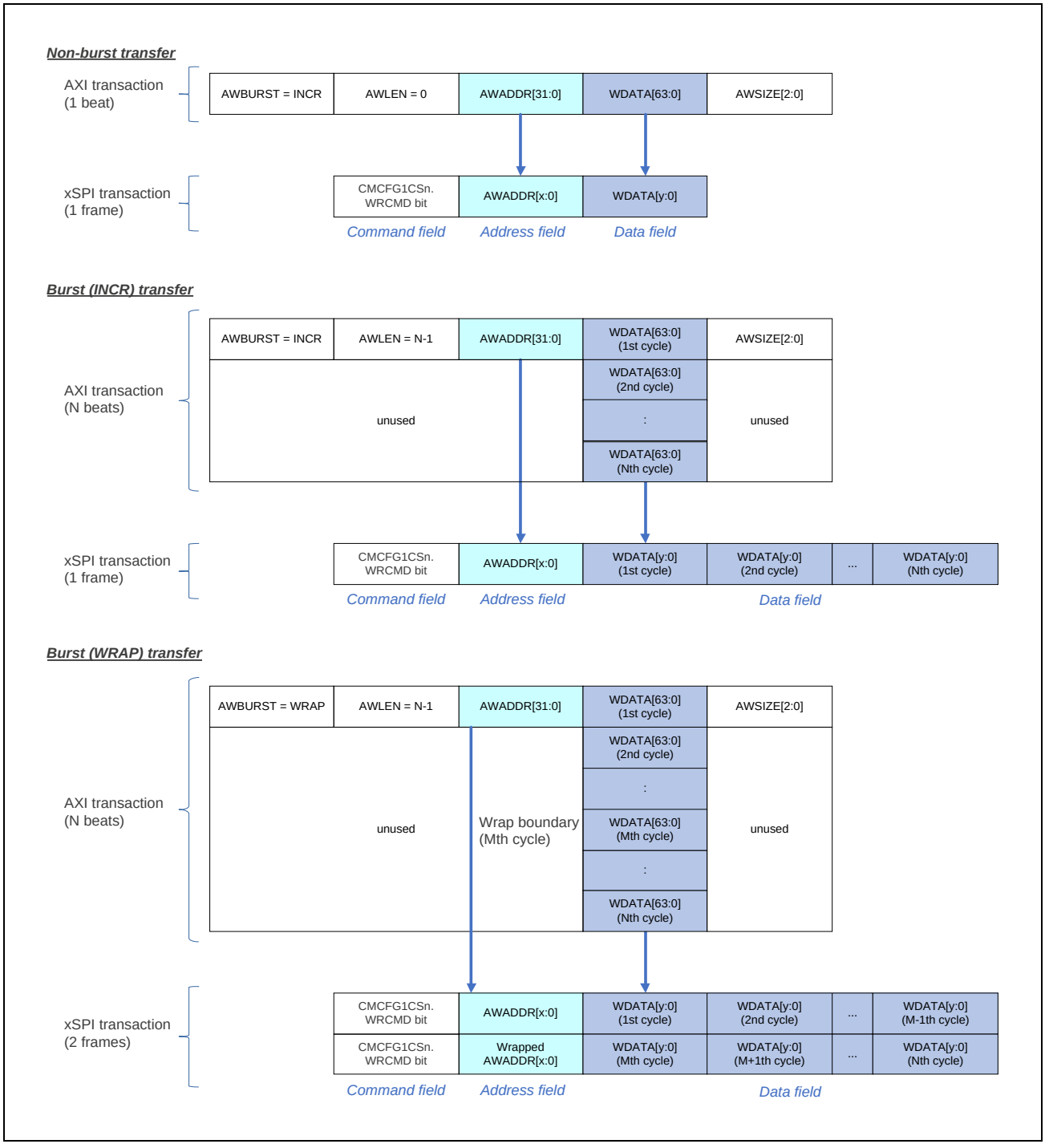


Figure 7.2-12 xSPI Frame Format in Write Access (Normal Format)

7.2.3.3.3 Combination function

At system bus write access to memory area, this xSPI master has the function to combine the write data for high throughput on xSPI bus. When this function is enabled (BMCFGCHn.MWRCOMB = 1b), this xSPI master transmits an xSPI frame with the selected size while the sequential address is incremental. When one of the below conditions is detected, even though the target size (BMCFGCHn.MWRSIZE[7:0]) is not reached, this xSPI master transmits the pending data to xSPI bus.

- Non-incremental address is detected.
- Different burst type is detected.
- Read transaction is detected.
- Access to a different slave is detected.
- Memory Write Data Push bit (BMCTL1.MWRPUSHCHx) is set.

This function could be useful for any slave device to request a chunk of data at a time. In this case, the system bus master should continue to provide the fixed data size with incremental address, e.g. if there is any device to request to write in page unit. The figure below shows the operation when the combination function is enabled.

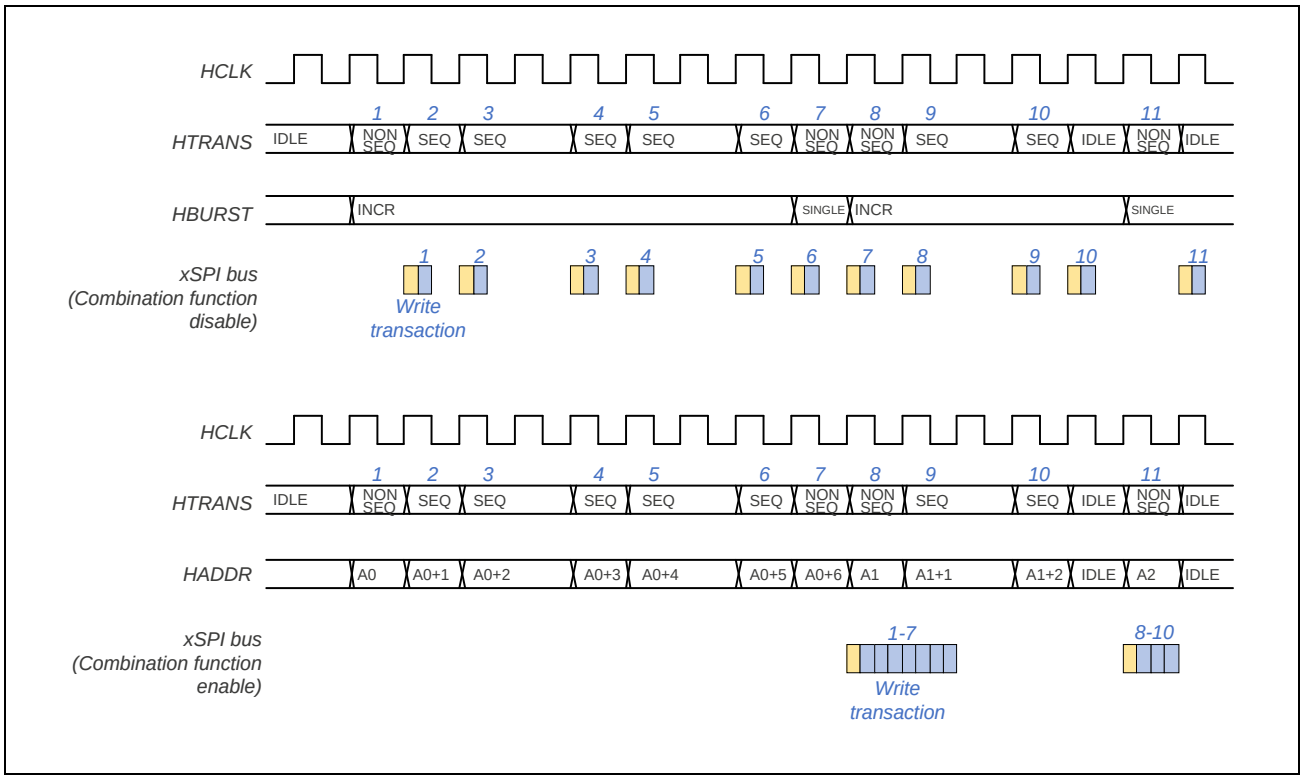


Figure 7.2-13 Combination Function

NOTE

The access which comply with any condition below is considered as incremental address.

- Transaction type is INCR.
- The access's start address is continuous to the previous last write address.
  - The access's start position of WSTRB is treated as the start address.
  - Previous access's last WSTRB is treated as the last write address.

The figure below shows a data combined example with AXI access.

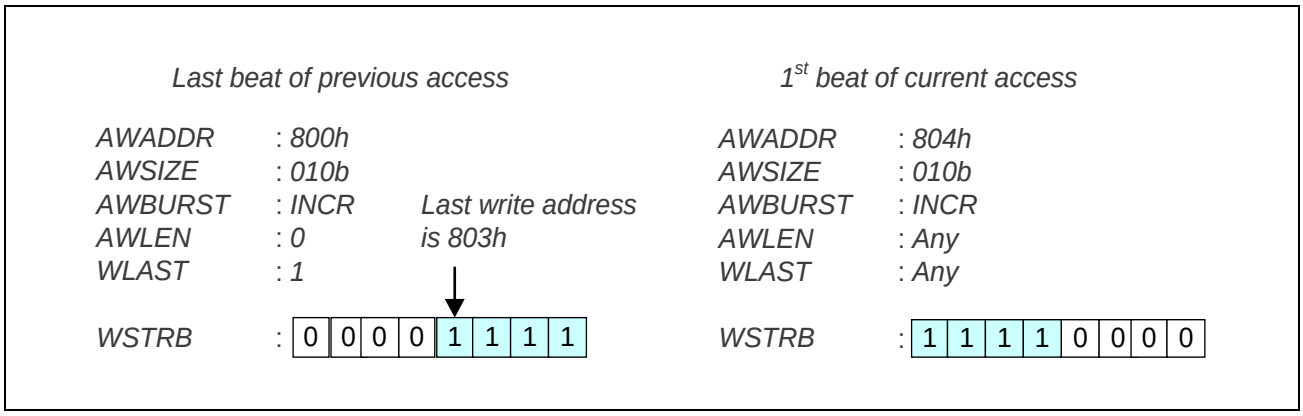


Figure 7.2-14 Data combined example with AXI access

7.2.3.3.4 Read access operation

At read access to memory area, soon after the read access is detected, this xSPI Master issues a read transaction to xSPI slave. The figure below shows the operation summary.

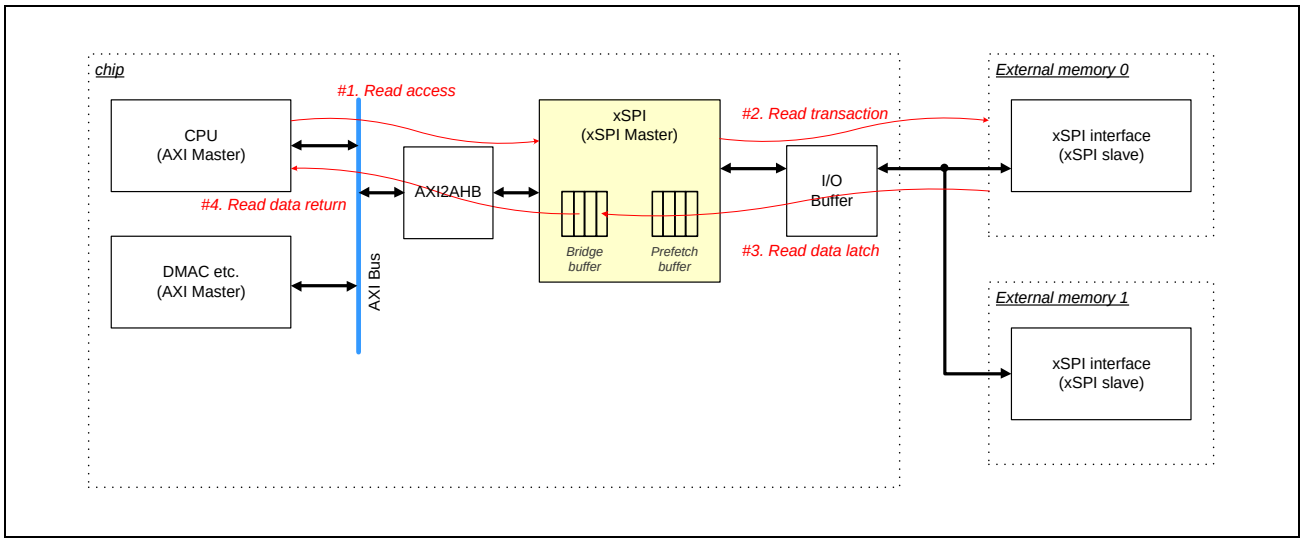


Figure 7.2-15 Read Access for Memory Area

The operation of xSPI bus changes depending on burst type. When the type is single or increment type, one system bus read transaction triggers one xSPI frame. When the type is wrap type and CMCFG0CSn.WPBSTMD is 0b, one system bus read transaction triggers two xSPI frame. The figure below shows the relationship between AXI and xSPI frame.

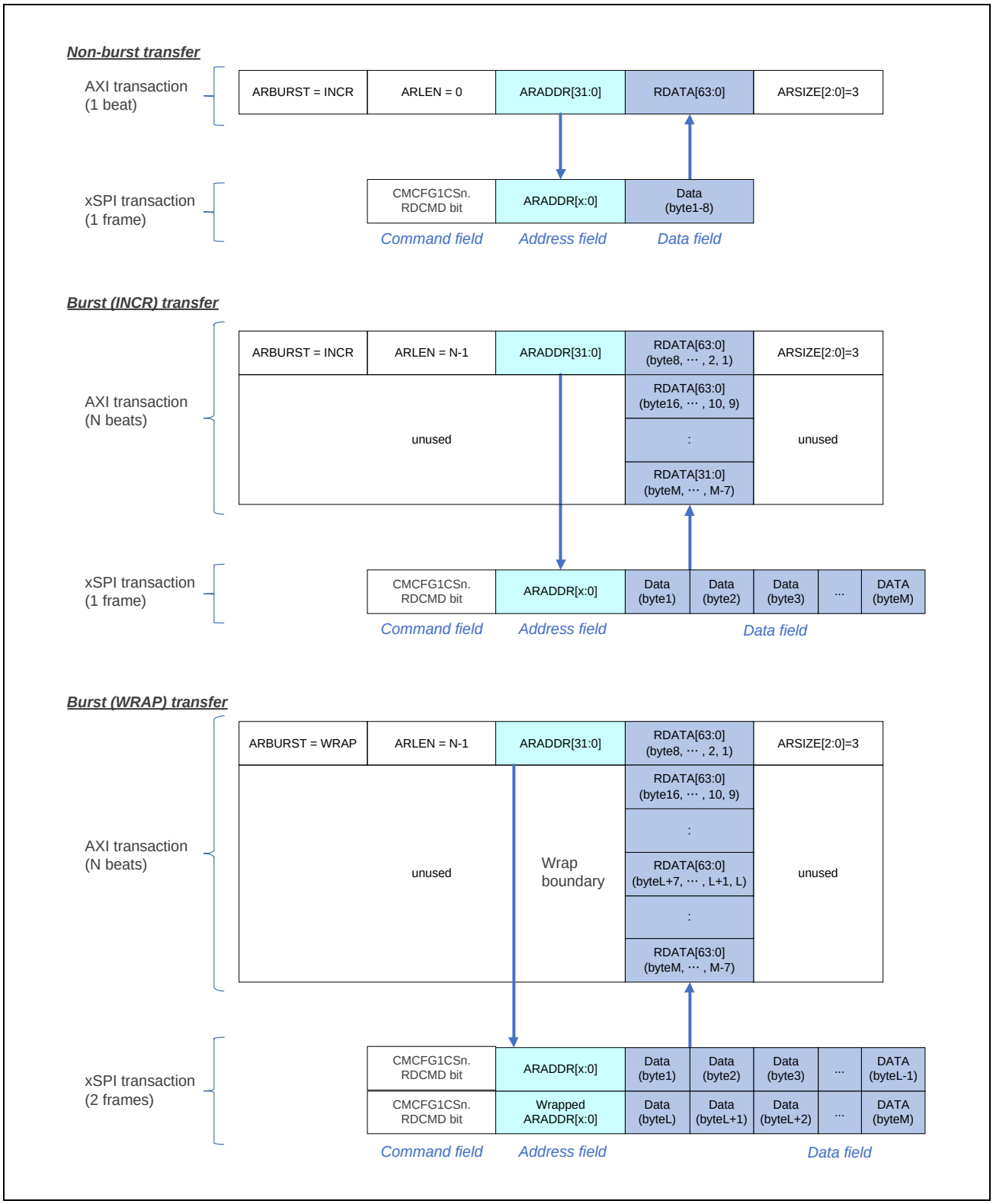


Figure 7.2-16 xSPI Frame Format in Read Access (Normal Format)

7.2.3.3.5 Prefetch function

At read access to memory area from system bus, this xSPI Master has the function to prefetch the read data to reduce the latency. When this function is enabled (BMCFGCHn.PREEN = 1b), this xSPI Master continues to read the incremental address and store the read data from xSPI slave in the internal prefetch buffer. And this xSPI master searches in the prefetch buffer for the following read access from system bus. If the target read data is found in the prefetch buffer, this xSPI Master returns the data from the prefetch buffer. If it is not found, this xSPI Master clears the prefetch buffer and newly issues a read transaction to xSPI slave. This function is effective in application such as the consecutive read addresses are close. But it is not effective in application such as the consecutive read addresses are not incremental because xSPI read frame for prefetch uses xSPI bus. The figure below shows the operation summary.

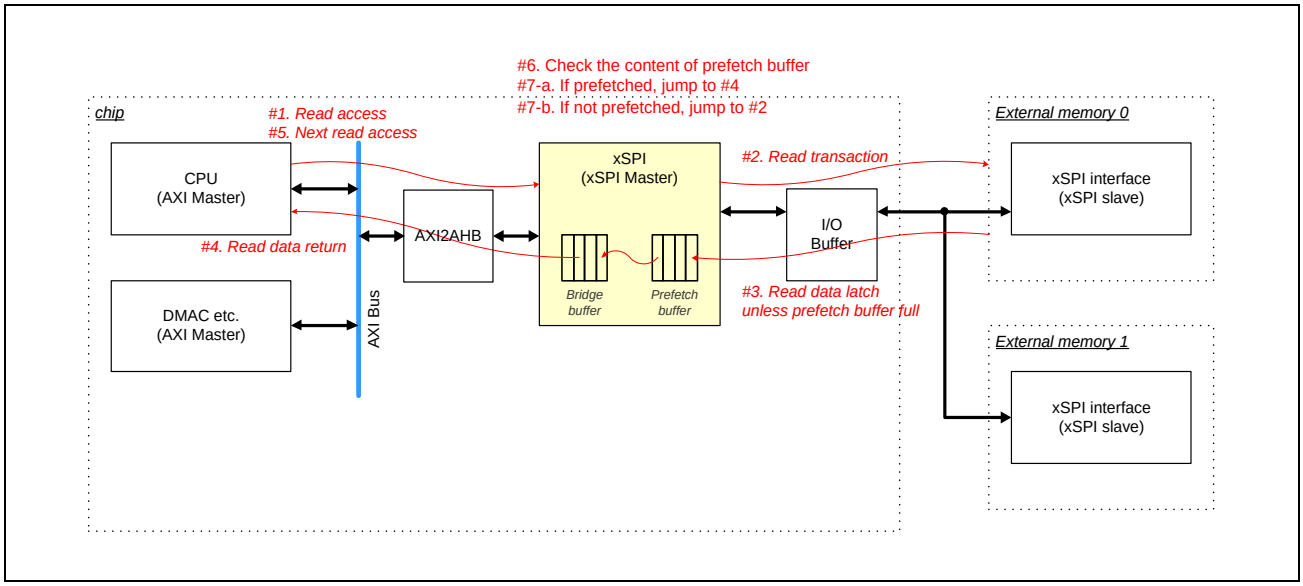


Figure 7.2-17 Read Access for Memory Area with Prefetch Enabled

NOTE

- When this prefetch function is enabled, Bus Master could read from not a slave device but the internal prefetch buffer. When accessed to the same address from multiple Bus Masters, this xSPI Master does not guarantee to read the latest data. If Bus Master intends to read the latest data from a slave device, the Master should clear the prefetch buffer (BMCTL1.PBUFCLRCHn) before issuing read.
- Prefetch buffer is implemented as FIFO-based, and when read access is issued, the data before the access address is discarded from the buffer. And when next access is issued to the region which is discarded at previous access, this module issues the xSPI read access again to fill the prefetch buffer.
- This module has the 1-line buffer which keeps the last 8-byte read data from the prefetch buffer. When read access is issued to the same address to the data in the 1-line buffer, this module returns read data from the 1-line buffer.

7.2.3.3.6 XiP mode

Some slave devices have XiP mode in which the command phase can be omitted for low latency. While in this mode, the xSPI master skips sending the command and the slave device implicitly performs the command that was executed in the previous transaction. When XiP mode bit is enabled (CMCTLCHn.XIPEN = 1b), this xSPI master inserts XiP enter code (CMCTLCHn.XIPENCODE[7:0]) in latency field. When XiP mode bit is disabled (CMCTLCHn.XIPEN = 0b), this xSPI master inserts XiP exit code (CMCTLCHn.XIPEXCODE[7:0]) in latency field. This function is available only for memory-mapping mode.

And when this xSPI master transmits XiP disable pattern, this master clears XiP mode bit and disables XiP mode configured for both channels. Note that it is not possible to disable XiP mode for only one channel by transmitting XiP disable pattern.

NOTE

- When enough latency cycle does not exist for XiP code, this xSPI Master could not insert XiP code.
- XiP mode could be used only for unidirectional access to a slave. The write transaction and read transaction should be separated.

7.2.3.4 Pattern Control

This xSPI Master has the function to transmit 3 types of pattern which are not in xSPI frame format. The pattern is triggered by setting trigger bits (LPCTL0-1.PATREQ[1:0]).

7.2.3.4.1 XiP disable pattern

XiP Disable pattern's length and value are configured by LPCTL0.XD1LEN[4:0] / XD1VAL / XD2LEN[4:0] / XD2VAL. It uses spi_ck, spi_doe/do signals. The number of output pins can be configured by XiP Disable pattern pin bits (LPCTL0.XDPIN[1:0]). It may be used to disable XiP mode for legacy SPI. The figure below shows the timing-chart.

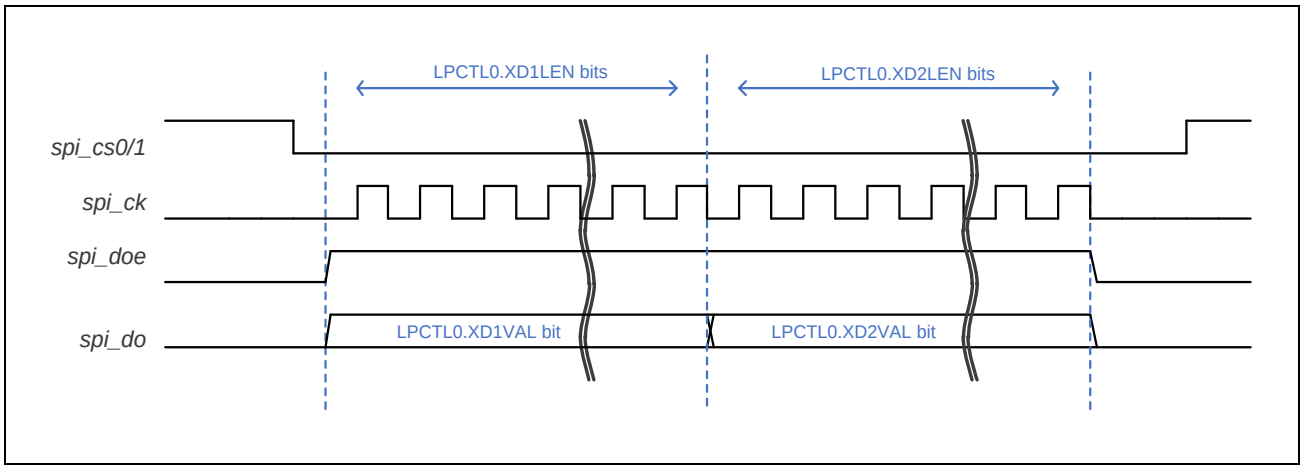


Figure 7.2-18 XiP Disable Pattern

7.2.3.4.2 Reset pattern

Reset pattern is specified in Serial Flash Reset Signaling Protocol (JESD252). CS Low/High width is configured with Reset Pattern Length bits (LPCTL1.RSTWID[2:0]). xSPI slave will sample the data input at the rising edge of CS. Setup time for data output is configured with Reset pattern data output setup time bits (LPCTL1.RSTSU[2:0]). The setup time should be always less than Reset pattern width. The figure below shows the timing-chart.

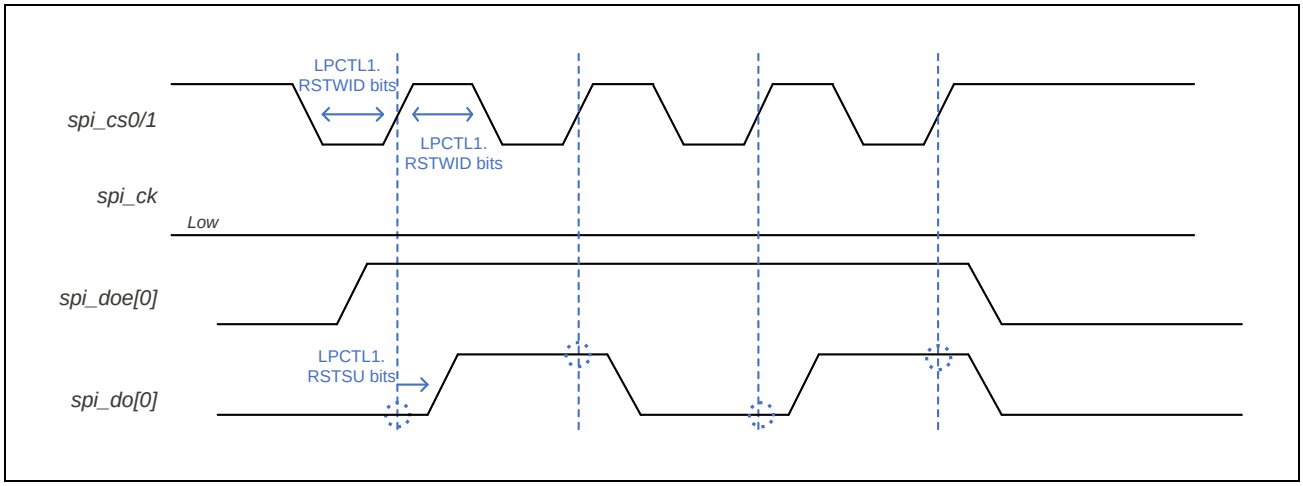


Figure 7.2-19 Reset Pattern

NOTE

In the protocol specification, CS Low/High width is defined as minimum 500 ns and Setup time is defined as minimum 6 ns.

7.2.3.4.3 CS only pattern

CS Only pattern activates CS port with the configured length bits (LPCTL1.RSTWID[2:0]). It may be used to resume from Deep Power Down state. The figure below shows the timing-chart.

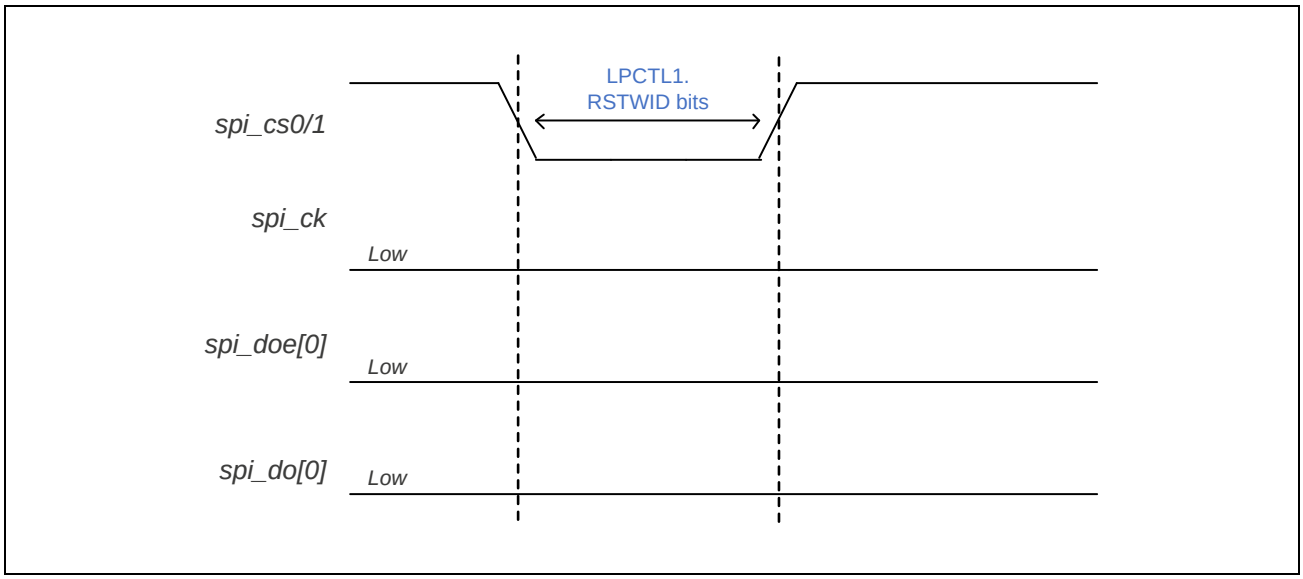


Figure 7.2-20 CS Only Pattern

7.2.3.5 Integrity Checking

This xSPI Master can detect some errors. The table below shows the error list and the detail behavior.

When reset of xSPI master is needed due to fatal error, set RESET ON/OFF control register (CPG_RST_10) to reset xSPI master. After confirming it is reset surely, release the reset.

Table 7.2-12 Error List

Error Type	Event	Flag Bit	Note (Action)
Calibration failed	The read data does not match the expected value during automatic calibration.	INTS.CAFAILCS0-1	It results in writing unexpected data to xSPI slave.
System bus error	This xSPI Master responds error on system bus for memory-mapping.	INTS.BUSERRCH0-1	This xSPI master should be reset for fatal error.
ECC error detection	Falling edge on ECS port is detected. It could be useful only for xSPI slave with ECC detection function.	INTS.ECSCS0-1	Only notify the error event of xSPI slave.
DS timeout	DS does not toggle in read transaction with using DS.	INTS.DSTOCS0-1	Both xSPI master and xSPI slave should be reset for fatal error.
Periodic transaction timeout	Read value does not match with the expected value in periodic manual-command mode.	INTS.PERTO	Depending on the status of function

7.2.3.5.1 Interrupts

This xSPI Master has interrupt ports.

It can be monitored with Interrupt Status Register (INTS). And it can be programmable with Interrupt Enable register (INTE). The table below shows the related register bits.

Table 7.2-13 xSPI Interrupt Sources

Name	ICU input	Interrupt Sources	CA55 GIC Request	CM33 GIC Request
XSPI0_INT	int_spi_pulse	Interrupt	Possible	Possible
XSPI0_INTERR	int_spi_err_pulse	Error interrupt	Possible	Possible

Table 7.2-14 Interrupt Register Bit

Flag Bit	Enable Bit	Clear Bit	Interrupt Sources (m = 0)
CASUCCS1	CASUCCS1E	CASUCCS1C	XSPIm_INT
CASUCCS0	CASUCCS0E	CASUCCS0C	
CAFAILCS1	CAFAILCS1E	CAFAILCS1C	XSPIm_INTERR
CAFAILCS0	CAFAILCS0E	CAFAILCS0C	
BUSERRCH1	BUSERRCH1E	BUSERRCH1C	XSPIm_INTERR
BUSERRCH0	BUSERRCH0E	BUSERRCH0C	
INTCS1	INTCS1E	INTCS1C	XSPIm_INTERR
INTCS0	INTCS0E	INTCS0C	
ECSCS1	ECSCS1E	ECSCS1C	XSPIm_INTERR
ECSCS0	ECSCS0E	ECSCS0C	
DSTOCS1	DSTOCS1E	DSTOCS1C	XSPIm_INTERR
DSTOCS0	DSTOCS0E	DSTOCS0C	
PERTO	PERTOE	PERTOC	XSPIm_INTERR
INICMP	INICMPE	INICMPC	XSPIm_INT
PATCMP	PATCMPE	PATCMPC	XSPIm_INT
CMDCMP	CMDCMPE	CMDCMPC	XSPIm_INT

7.2.3.6 Flows of Operations

7.2.3.6.1 Flow of startup

The figure below shows flow of startup.

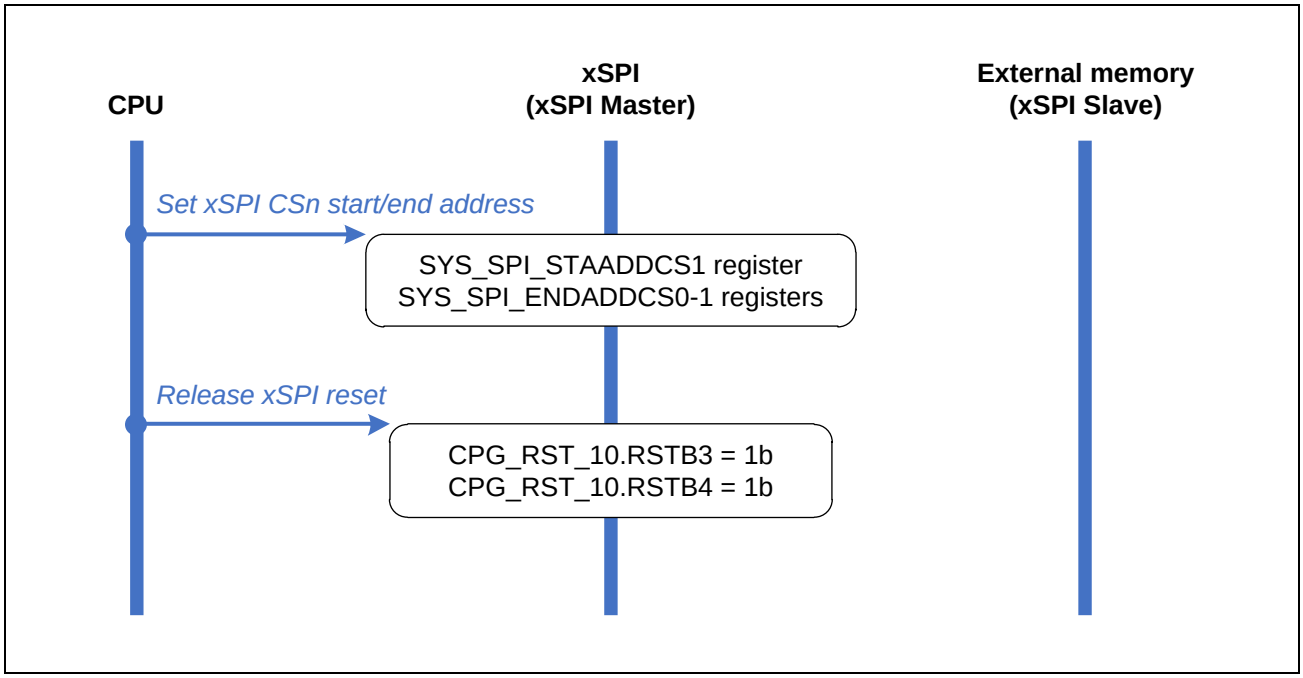


Figure 7.2-21 Flow of Startup

NOTE

SYS_SPI_STAADDCS1 and SYS_SPI_ENDADDCS0-1 registers should be set while xSPI reset is asserted.

7.2.3.6.2 Flow of configuration

The figure below shows flow of configuration.

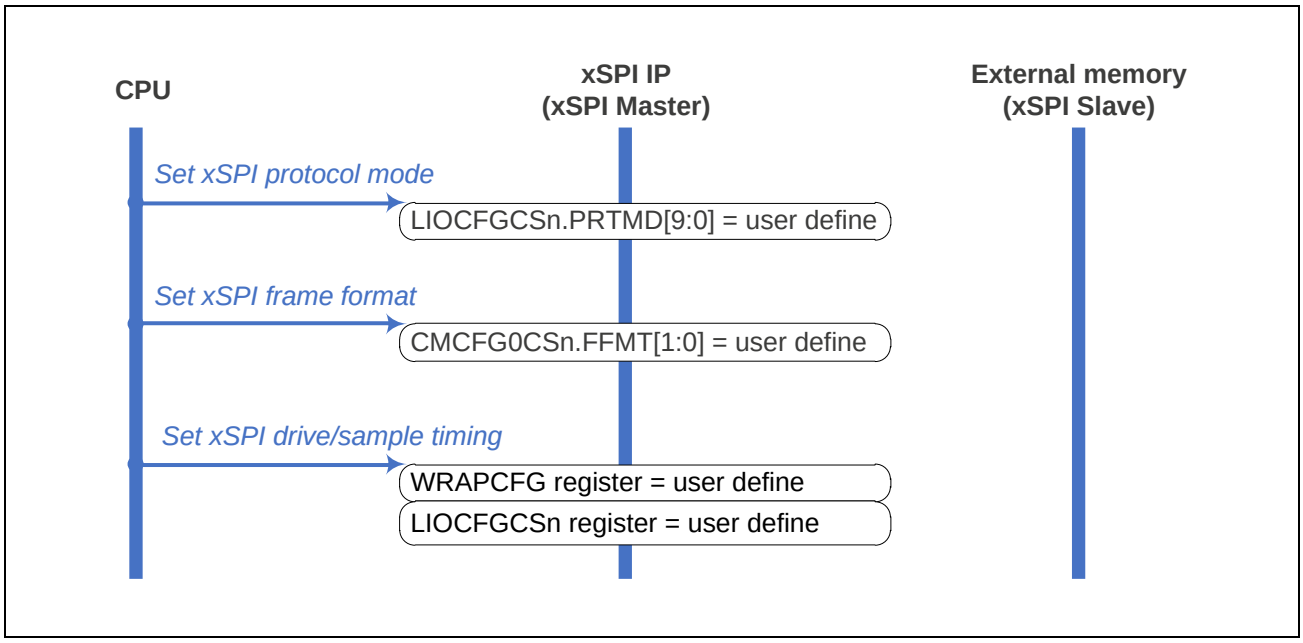


Figure 7.2-22 Flow of Configuration

7.2.3.6.3 Flow of communication stop

The figure below shows flow of communication stop.

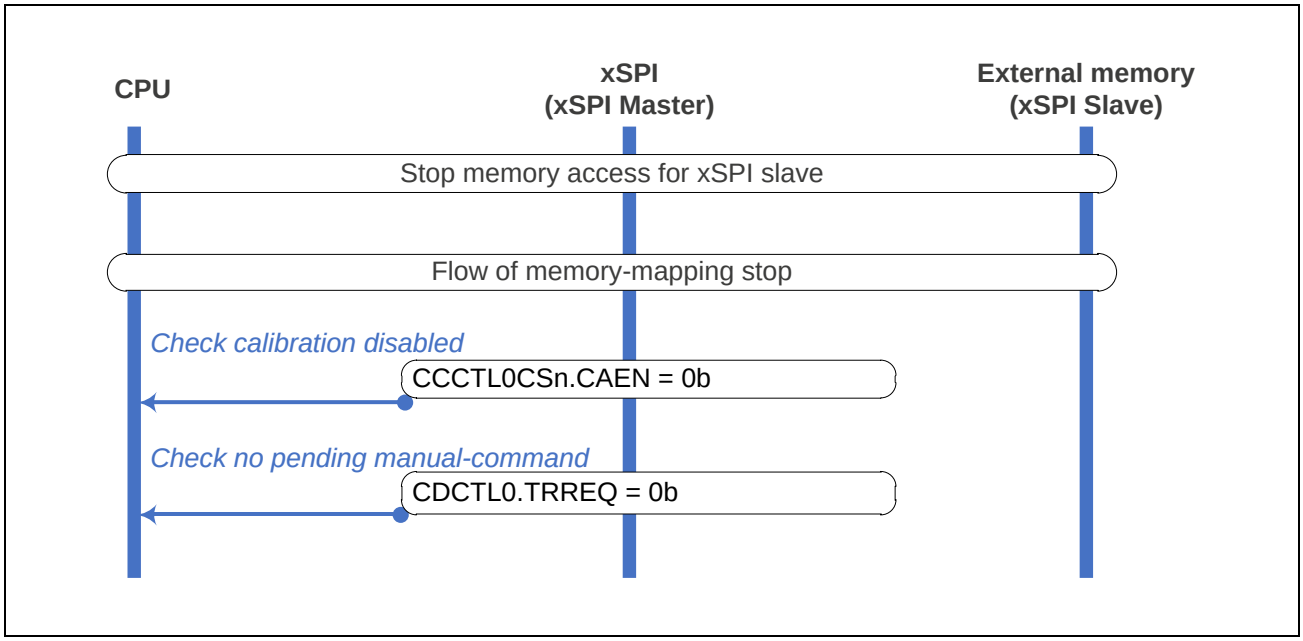


Figure 7.2-23 Flow of Communication Stop

NOTE

In case of re-config of any configuration register, all communications with xSPI slave should be stopped surely to avoid race condition between register setting and memory access. It means that automatic calibration is disabled, and there is no pending manual-command and memory-mapping access.

7.2.3.6.4 Flow of automatic calibration

The figure below shows flow of automatic calibration.

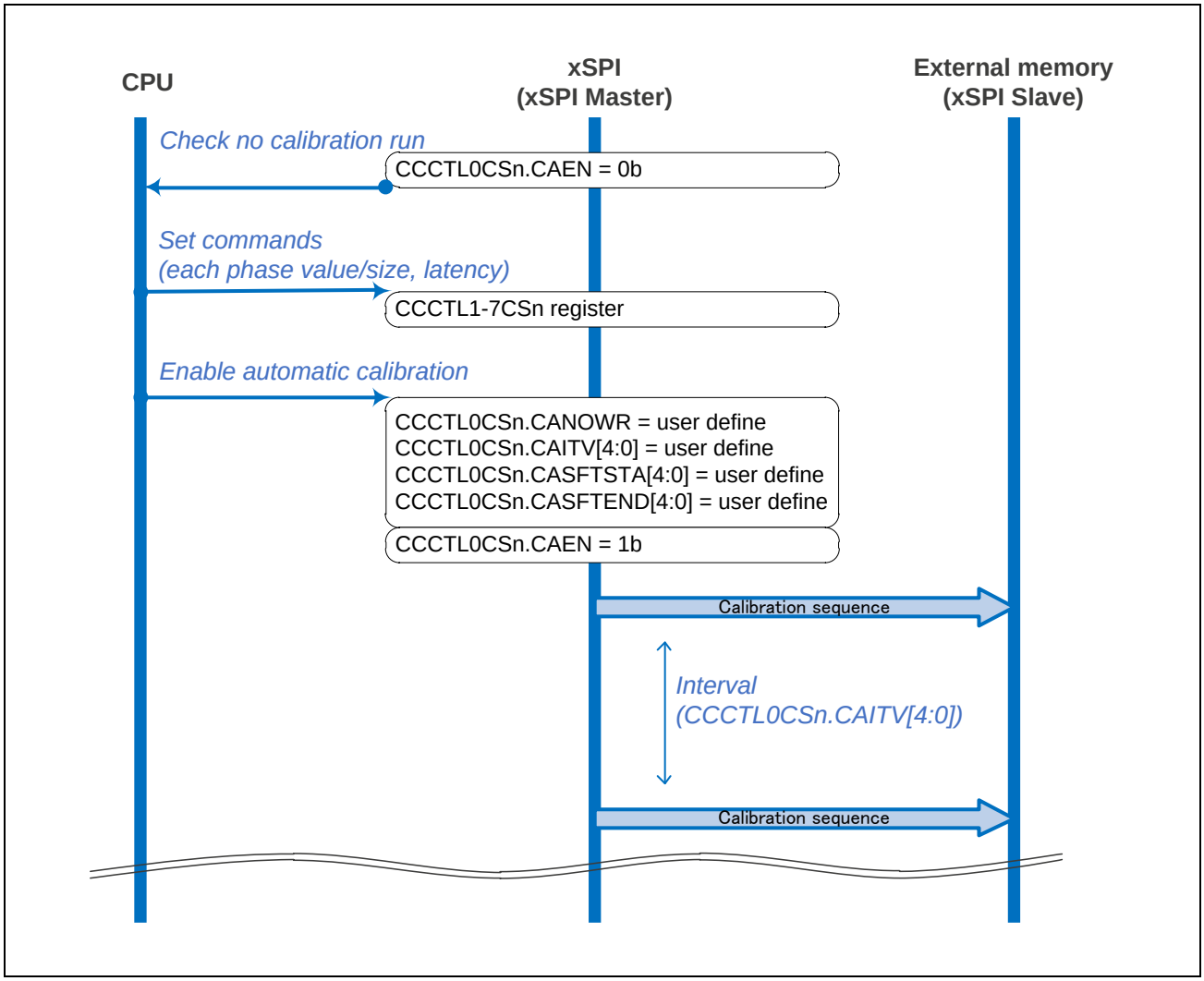


Figure 7.2-24 Flow of Automatic Calibration

7.2.3.6.5 Flow of manual-command procedure

The figure below shows manual-command procedure for direct mode.

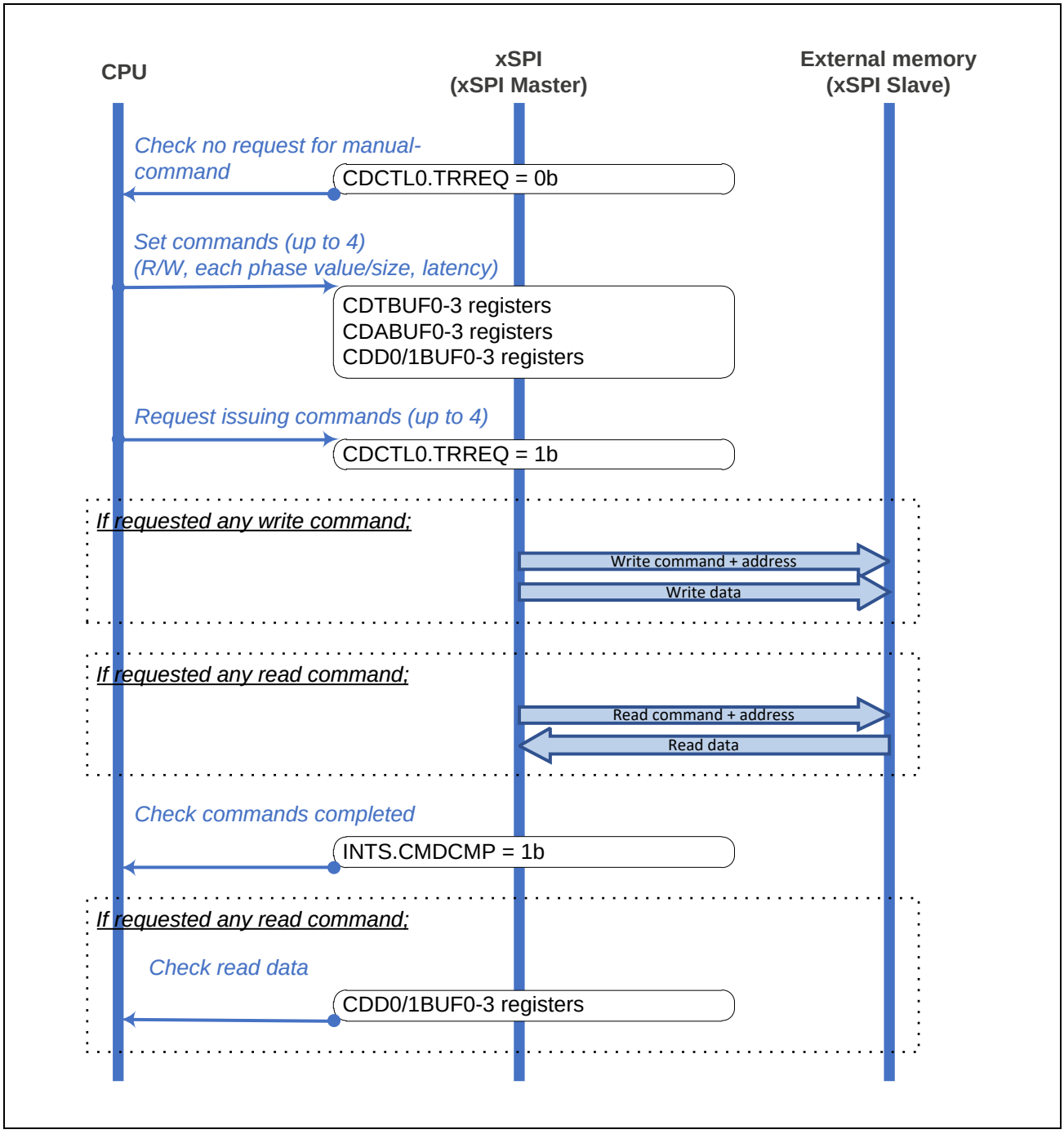


Figure 7.2-25 Flow of Manual-Command Procedure for Direct Mode

The figure below shows manual-command procedure for periodic mode.

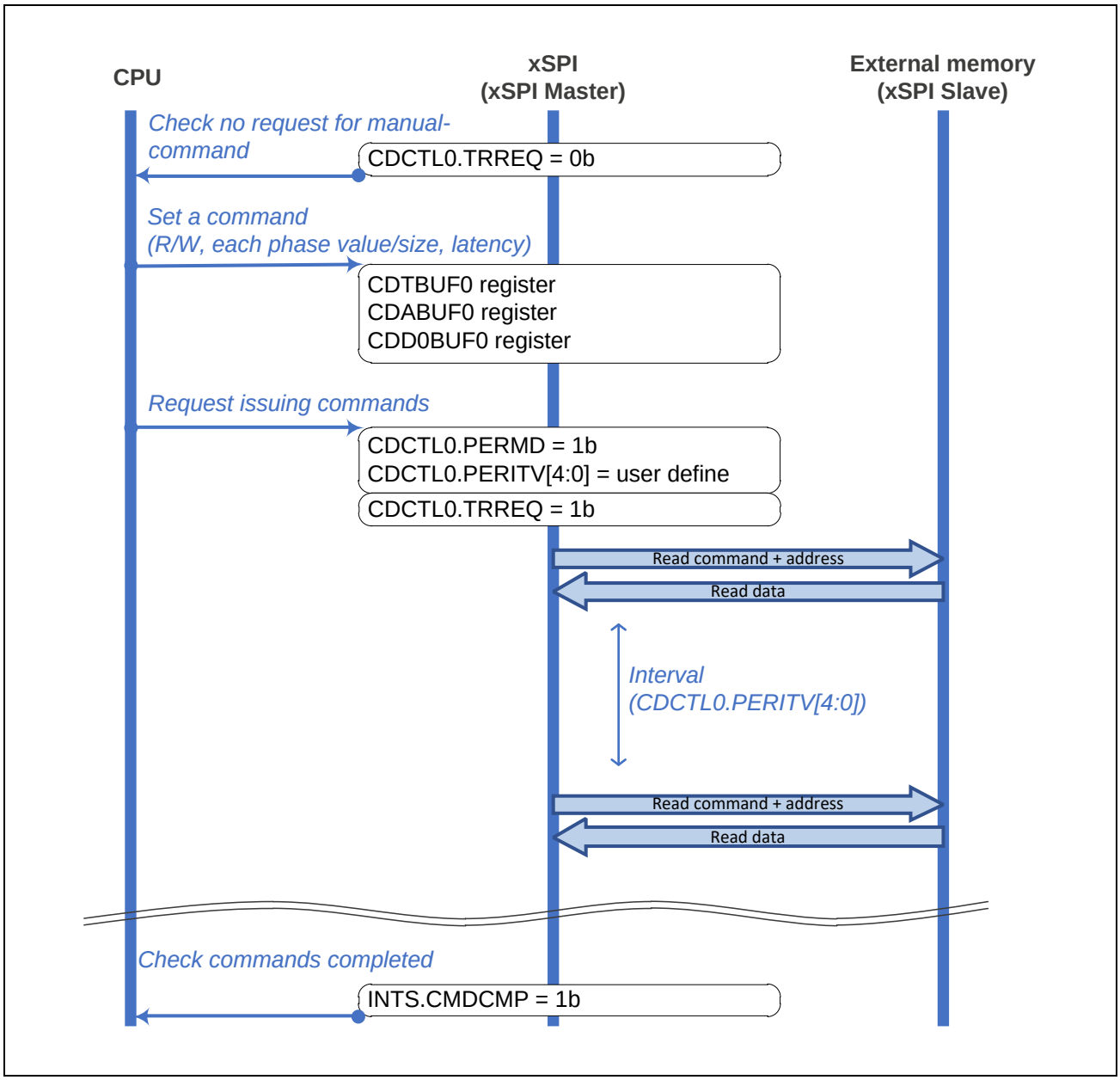


Figure 7.2-26 Flow of Manual-Command Procedure for Periodic Mode

7.2.3.6.6 Flow of memory-mapping

The figure below shows flow of memory-mapping.

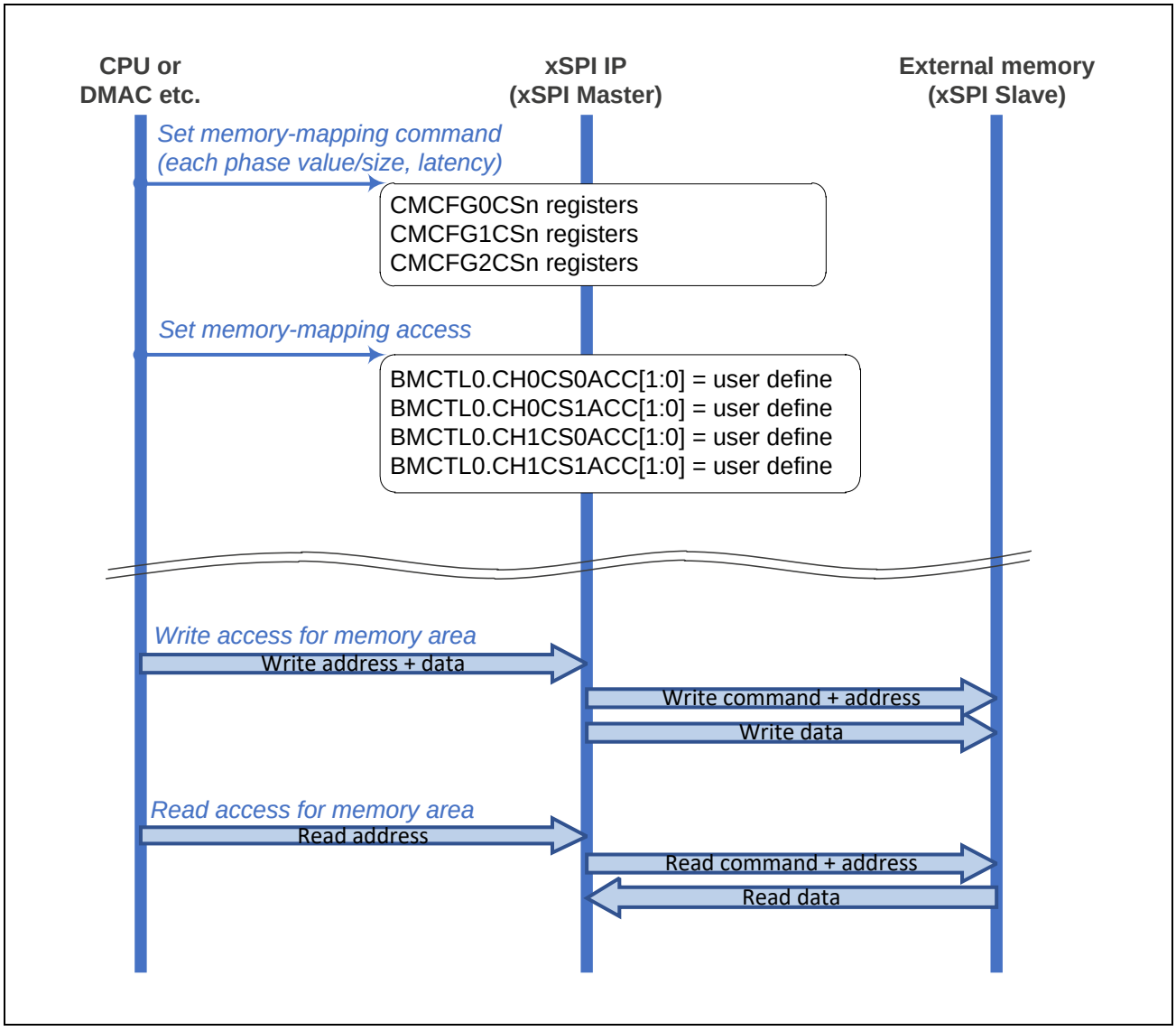


Figure 7.2-27 Flow of Memory-Mapping

7.2.3.6.7 Flow of memory-mapping stop

The figure below shows flow of memory-mapping stop. This flow should be used in the flow of **7.2.3.6.3 Flow of communication stop**.

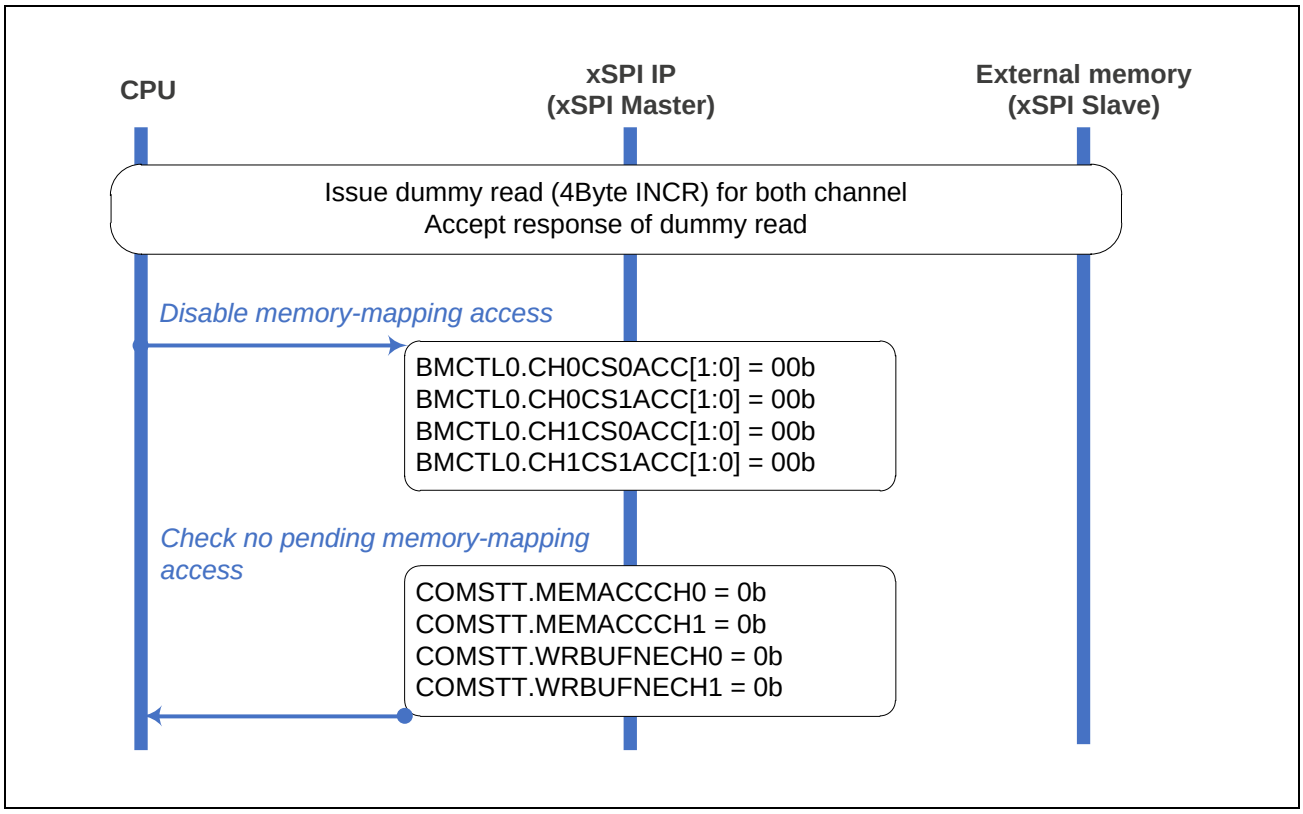


Figure 7.2-28 Flow of Memory-Mapping Stop

7.2.3.6.8 Flow of pattern request

The figure below shows flow of pattern request. Before requesting any pattern, any ongoing commands should be completed or canceled.

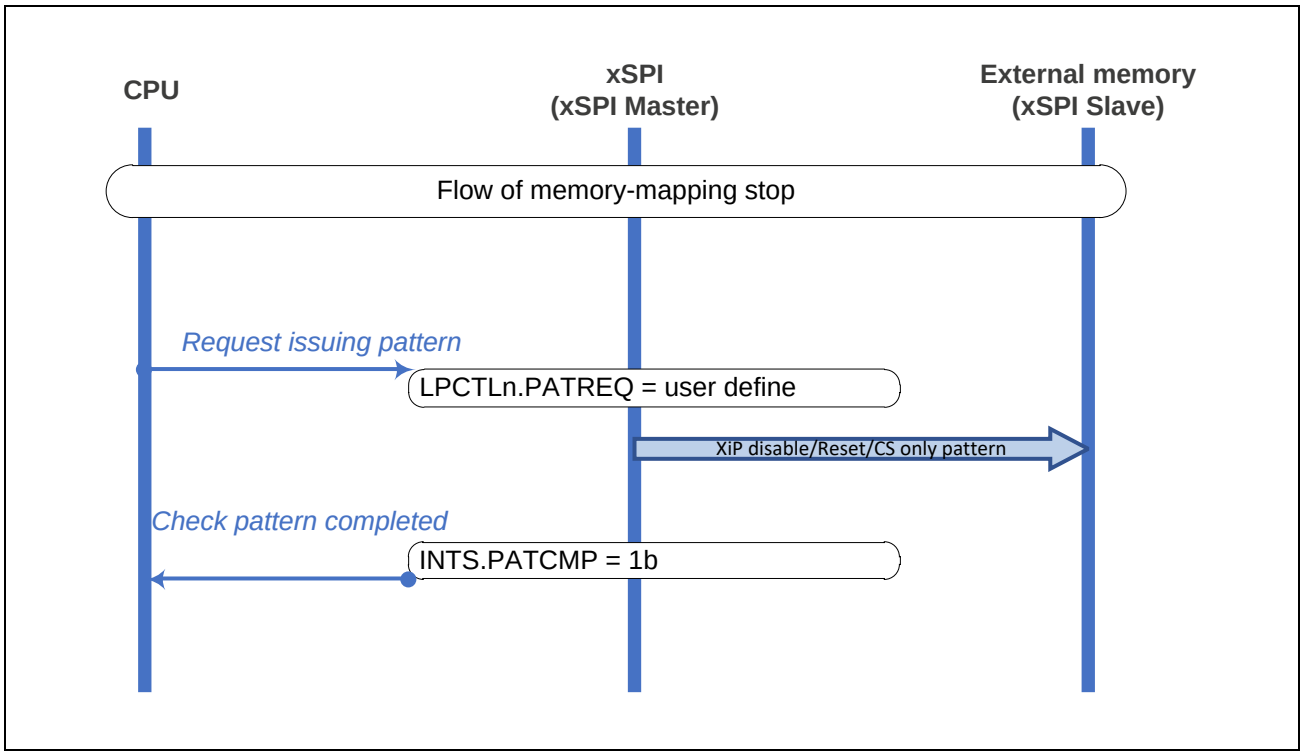


Figure 7.2-29 Flow of Pattern Request

7.2.3.6.9 Flow of XiP mode

The figure below shows flow of XiP mode enable/disable.

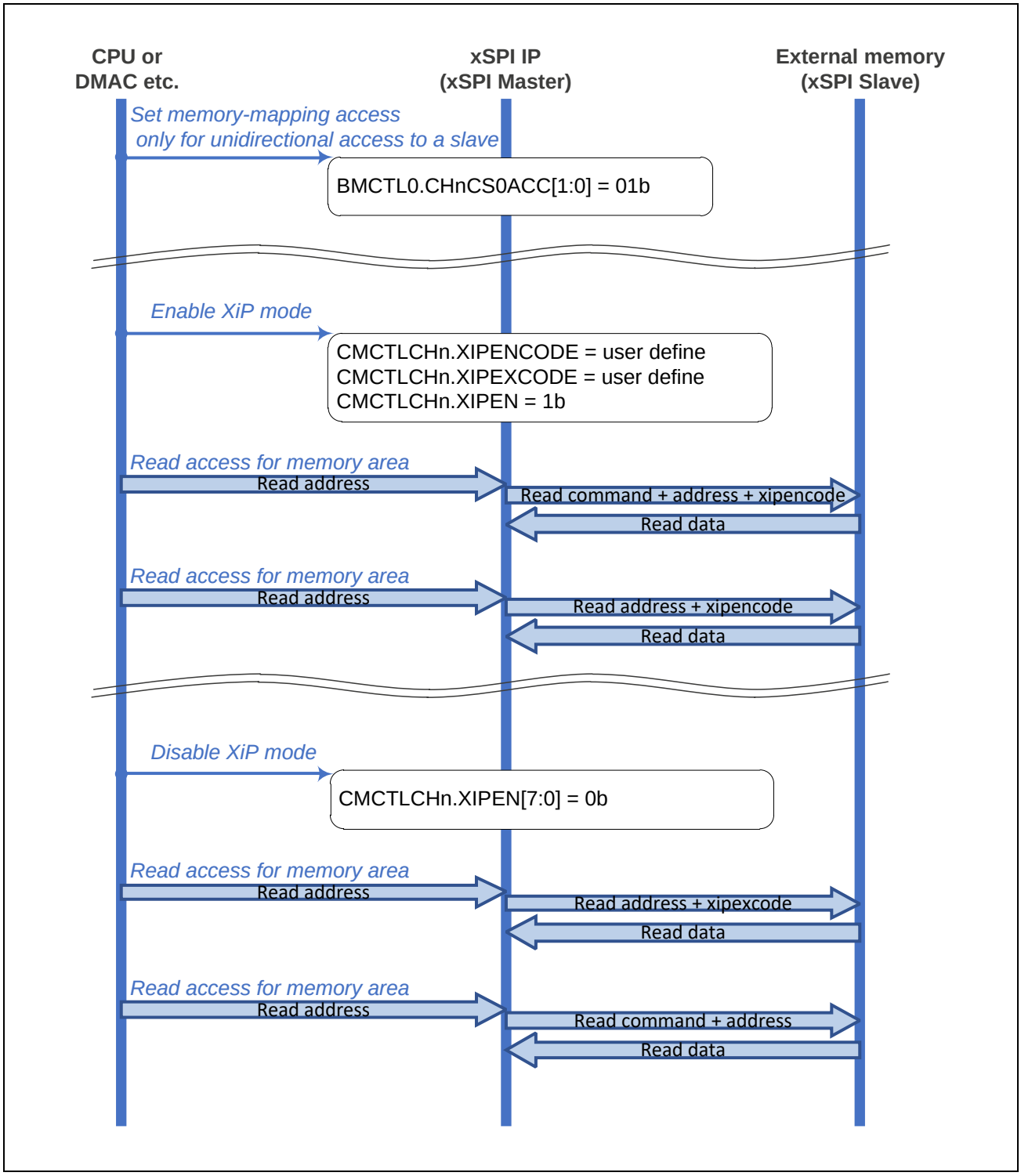


Figure 7.2-30 Flow of XiP Mode Enable/Disable

7.2.3.7 Usage Notes

- If prefetch function is enabled and CSn end address register (SYS_SPI_ENDADDCSn) is not set correctly compared to size of connected slave device, xSPI master may access outside of the slave device address. It may cause DS timeout in case of 8D-8D-8D. Make sure to set the correct slave device size to SYS_SPI_ENDADDCSn register.

- DDR access without DS is not supported.

SECTION 7 LOW-SPEED INTERFACE

7.3 Serial Communications Interface (RSCI)

7.3.1 Overview

This LSI has ten independent serial communications interface (RSCI) channels. The RSCI (hereinafter referred to as SCI) is configurable to six asynchronous and synchronous serial interfaces:

- Asynchronous interfaces (UART and Asynchronous Communications Interface Adapter (ACIA))
- 8-bit clock synchronous interface
- Simple I2C (master-only)
- Simple SPI
- Smart card interface
- Simple LIN interface

The smart card interface complies with the ISO/IEC 7816-3 standard for electronic signals and transmission protocol. Each SCI has FIFO buffers to enable continuous and full-duplex communication, and the data transfer speed can be configured independently using a baud rate generator.

Table 7.3-1 lists the SCI specifications, **Figure 7.3-1** shows a block diagram of SCI channel n, and **Table 7.3-2** lists the I/O pins.

Table 7.3-1 SCI Specifications (1/3)

Item	Specifications
Number of channels	10 channels
Serial communication modes	• Asynchronous • Clock synchronous • Smart card interface • Simple I2C • Simple SPI • Simple LIN
Transfer speed	Bit rate specifiable with the baud rate generator
Full-duplex communications	Transmitter: Continuous transmission possible using double- buffering Receiver: Continuous reception possible using double-buffering
Half-duplex communication	Possible by using TXD pins
Data transfer	Selectable as LSB-first or MSB-first transfer
RXD/TXD polarity	Selectable polarity for TXD and RXD independently
Interrupt sources	• Transmit end, transmit data empty, receive data full, receive error, receive data ready, and address match, break field detection/output, bus collision detection, active edge detection • Completion of generation of a start condition, restart condition, or stop condition (for simple I2C mode)
RS-485 driver control function	Output DE pin to enable transmission mode of external transceiver
Loopback function	Self-diagnosis of communication function inside module supported
Synchronizing circuit bypass function	Bus clock (RSCI_m_PCLK) can be used as operation clock (RSCI_m_TCLK) when bypass function is enabled.

Table 7.3-1 SCI Specifications (2/3)

Item		Specifications
Asynchronous mode	Data length	7, 8, or 9 bits
	Transmission stop bit	1 or 2 bits
	Reception sampling timing adjustment	Adjustable reception sampling timing from default timing
	Transmission timing adjustment	Adjustable timing of transmission wave edge by register
	Parity	Even parity, odd parity, or no parity
	Receive error detection	Parity, overrun, and framing errors
	Hardware flow control	Transmission and reception controllable with CTSm# and RTSm# pins
	Transmission/ reception	Selectable to 1-stage register or 32-stage FIFO
	Address match	Interrupt request/event output can be issued upon detecting a match between received data and the value in the compare match register.
	Start-bit detection	Selectable to low level or falling edge detection
	Break detection	Breaks from framing errors detectable by reading register
	Clock source	Selectable to internal or external clock
	Double-speed mode	Baud rate generator double-speed mode is selectable.
	Multi-processor communications function	Serial communication enabled between multiple processors
	Noise cancellation	Digital noise filters included on signal paths from RXDn pin inputs
	Half data communication function	Half data communication function makes data “0” low pulse only in the first half of 1-bit period
Simple-LIN	Start frame transmission/ Start frame reception	• Break field output possible • Break field output complete interrupt output possible • Bus collision detection possible • Bus collision detection interrupt output possible • Break field detectable • Break field detected interrupt output possible • Control field 0/1 data comparison function • Control field 1 can set two types of comparison data of primary and secondary. • Priority interrupt bit can be set in control field 1b. • Bit rate measurement function
	Input/output control function	When the simple-LIN function is OFF, the RXD receive signal can be input to the core section.
Clock synchronous mode	Data length	8 bits
	Reception sampling timing adjustment	Adjustable reception sampling timing to backward from default timing in case internal clock used
	Receive error detection	Overrun error
	Clock source	Selectable to internal clock (master mode) or external clock (slave mode)
	Double-speed mode	Baud rate generator double-speed mode is selectable.
	Hardware flow control	Transmission and reception controllable with CTSm# and RTSm# pins
	Transmission/ reception	Selectable to 1-stage register or 32-stage FIFO

Table 7.3-1 SCI Specifications (3/3)

Item		Specifications
Smart card interface mode	Error processing	Error signal can be automatically transmitted on detecting a parity error during reception.
		Data can be automatically retransmitted on receiving an error signal during transmission.
	Data type	Both direct and inverse convention supported
Simple I2C mode	Transfer format	I2C bus format (MSB-first only)
	Operating mode	Master (single-master operation only)
	Transfer rate	Up to 400 kbps
	Noise cancellation	The signal paths from input on the SCLn and SDAn pins incorporate digital noise filters and provide an adjustable interval for noise cancellation.
Simple SPI mode	Data length	8 bits
	Error detection	Overrun error
	Clock source	Selectable to internal clock (master mode) or external clock (slave mode)
	Double-speed mode	Baud rate generator double-speed mode is selectable.
	Reception sampling timing adjustment	Adjustable reception sampling timing to backward from default timing in case internal clock used
	SS input pin function	High impedance state can be invoked on the output pins by driving the SSn# pin high.
	Clock settings	Configurable between four clock phase and clock polarity settings
	Transmission/ reception	Selectable to 1-stage register or 32-stage FIFO
Bit rate modulation function		Error reduction through correction of outputs from the baud rate generator

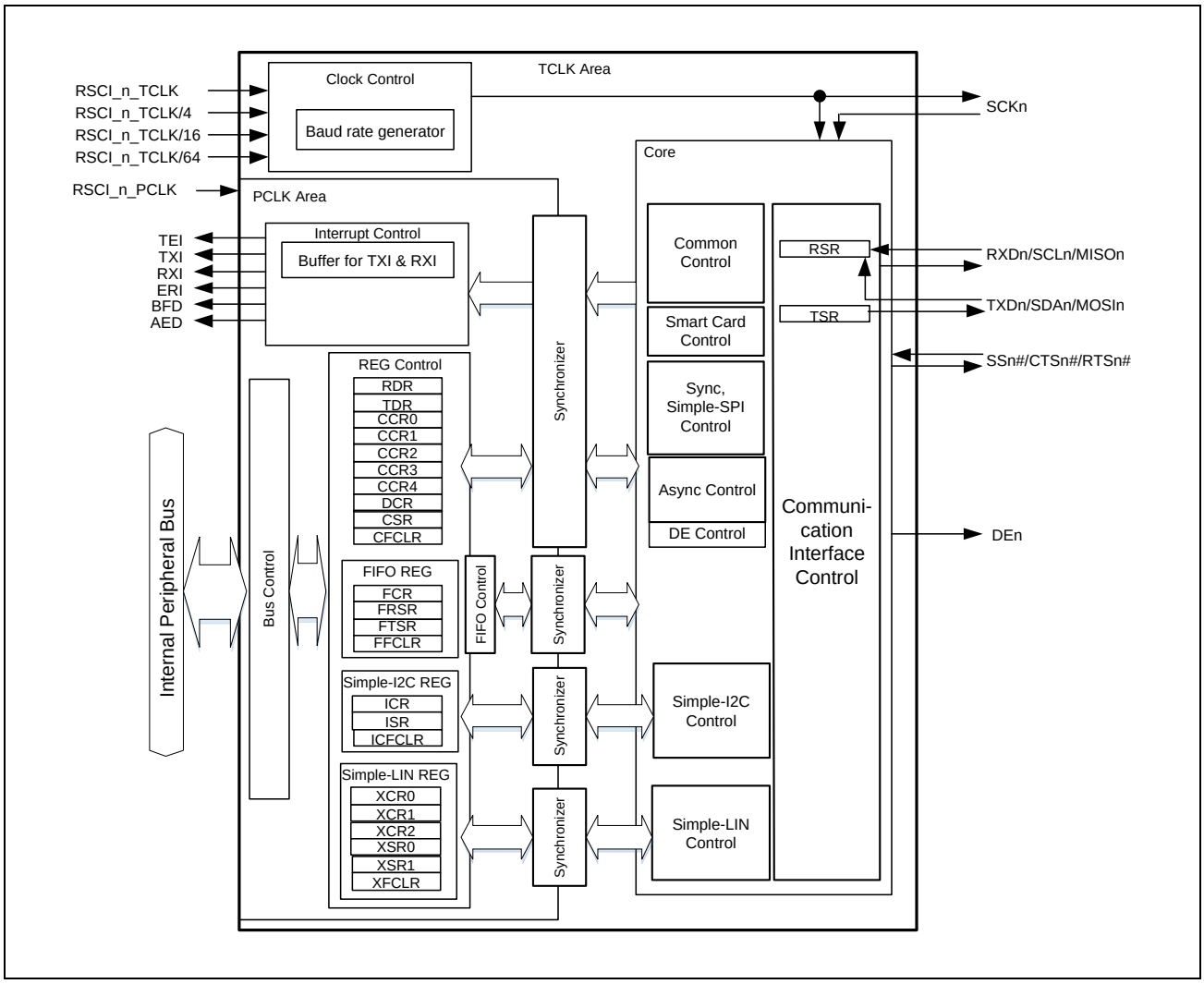


Figure 7.3-1 RSCI Block Diagram

Table 7.3-2 SCI Input/Output Pins

Channel	Pin Name	Signal name	Input/ Output	Function
RSCIm	SCKm	SCKm	I/O	RSCIm clock input/output
	RXDm_MISOm_SCLm	RXDm/SCLm/ MISOm	I/O	RSCIm receive data input RSCIm I2C clock input/output RSCIm data input (master), data output (slave)
	TXDm_MOSIm_SDAm	TXDm/SDAm/ MOSIm	I/O	RSCIm transmit data output RSCIm I2C data input/output RSCIm data output (master), data input (slave)
	SSm_CTSmN_RTSmN	SSm#/CTSm#/ RTSm#	I/O	RSCIm chip select input RSCIm transfer start control output RSCIm transfer start control input
	CTSmN	CTSm#	Input	RSCIm transfer start control input (exclusive use)
	DEm	DEm	Output	RSCIm Driver Enable output

Note: m = 0 to 9

7.3.2 SCI Registers

This section explains each register specification.

Table 7.3-3 shows a list of registers/bits that cannot be rewritten during communication. Also, check the notes on register rewriting described in **7.3.18 Usage Notes**.

Table 7.3-3 Bits that cannot be Rewritten during Communication

Register	Bit	Write Constraint
CCR0	RE	In Clock synchronous mode, Simple-SPI and Simple-I2C, 1b can be written only when TE = 0b and RE = 0b. After setting TE or RE to 1b, only 0b can be written in TE and RE.
CCR0	TE	
CCR0	IDSEL	Writable only when TE = 0b and RE = 0b.
CCR0	SSE	Writable only when TE = 0b and RE = 0b.
CCR1	CTSE	Writable only when TE = 0b and RE = 0b.
CCR1	CTSPEN	Writable only when TE = 0b and RE = 0b.
CCR1	PE	Writable only when TE = 0b and RE = 0b.
CCR1	PM	Writable only when TE = 0b and RE = 0b.
CCR1	TINV	Writable only when TE = 0b and RE = 0b.
CCR1	RINV	Writable only when TE = 0b and RE = 0b.
CCR1	SPLP	Writable only when TE = 0b and RE = 0b.
CCR1	SHARPS	Writable only when TE = 0b and RE = 0b.
CCR1	NFCS[2:0]	Writable only when TE = 0b and RE = 0b.
CCR1	NFEN	Writable only when TE = 0b and RE = 0b.
CCR2	All bits	Writable only when TE = 0b and RE = 0b.
CCR3	All bits	Writable only when TE = 0b and RE = 0b.
CCR4	All bits	Writable only when TE = 0b and RE = 0b.
ICR	IICDL[4:0]	Writable only when TE = 0b and RE = 0b.
ICR	IICINTM	Writable only when TE = 0b and RE = 0b.
ICR	IICCSC	Writable only when TE = 0b and RE = 0b.
FCR	All bits	Writable only when TE = 0b and RE = 0b.
DCR	All bits	Writable only when TE = 0b and RE = 0b.
XCR0	All bits	Writable only when TE = 0b and RE = 0b.
XCR1	TCST	Writable only when TE = 0b and RE = 0b.
XCR1	SDST	Writable only when TE = 0b and RE = 0b.
XCR1	PCF1D [7:0]	Writable only when TE = 0b and RE = 0b.
XCR1	SCF1D [7:0]	Writable only when TE = 0b and RE = 0b.
XCR1	CF1CE [7:0]	Writable only when TE = 0b and RE = 0b.
XCR2	All bits	Writable only when TE = 0b and RE = 0b.
HCH	All bits	Writable only when TE = 0b and RE = 0b.

The base addresses for the respective channels are as follows.

Table 7.3-4 Register Base Addresses

Base Register Name	Ch	Base Address
<RSCI0_base>	RSCI0	0_1280_0C00h (5280_0C00h*1, 4280_0C00h*2)
<RSCI1_base>	RSCI1	0_1280_1000h (5280_1000h*1, 4280_1000h*2)
<RSCI2_base>	RSCI2	0_1280_1400h (5280_1400h*1, 4280_1400h*2)
<RSCI3_base>	RSCI3	0_1280_1800h (5280_1800h*1, 4280_1800h*2)
<RSCI4_base>	RSCI4	0_1280_1C00h (5280_1C00h*1, 4280_1C00h*2)
<RSCI5_base>	RSCI5	0_1280_2000h (5280_2000h*1, 4280_2000h*2)
<RSCI6_base>	RSCI6	0_1280_2400h (5280_2400h*1, 4280_2400h*2)
<RSCI7_base>	RSCI7	0_1280_2800h (5280_2800h*1, 4280_2800h*2)
<RSCI8_base>	RSCI8	0_1280_2C00h (5280_2C00h*1, 4280_2C00h*2)
<RSCI9_base>	RSCI9	0_1280_3000h (5280_3000h*1, 4280_3000h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

7.3.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
Receive Data Register	RSCIm_RDR	0000_0000h	0000h	32
Transmit Data Register	RSCIm_TDR	FFFF_FFFFh	0004h	8, 16, 32
Common Control Register 0	RSCIm_CCR0	0000_0000h	0008h	8, 16, 32
Common Control Register 1	RSCIm_CCR1	0000_0000h	000Ch	8, 16, 32
Common Control Register 2	RSCIm_CCR2	FF00_FF04h	0010h	8, 16, 32
Common Control Register 3	RSCIm_CCR3	0000_1203h	0014h	8, 16, 32
Common Control Register 4	RSCIm_CCR4	0000_0000h	0018h	8, 16, 32
Communication Enable Status Register	RSCIm_CESR	00h	001Ch	8
Reserve	-	-	001Dh	-
Half data communication Control Register	RSCIm_HCR	00h	001Eh	8
Reserve	-	-	001Fh	-
Simple I2C Control Register	RSCIm_ICR	0000_0000h	0020h	8, 16, 32
FIFO Control Register	RSCIm_FCR	1F1F_0000h	0024h	8, 16, 32
Reserve	-	-	0028h to 002Fh	-
Driver Control Register	RSCIm_DCR	0000_0000h	0030h	8, 16, 32
Simple-LIN(SCIX) Control Register 0	RSCIm_XCR0	0000_0000h	0034h	8, 16, 32
Simple-LIN(SCIX) Control Register 1	RSCIm_XCR1	0000_0000h	0038h	8, 16, 32
Simple-LIN(SCIX) Control Register 2	RSCIm_XCR2	FFFE_0000h	003Ch	8, 16, 32
Reserve	-	-	0040h to 0047h	-
Common Status Register	RSCIm_CSR	6000_8000h	0048h	32
Simple I2C Status Register	RSCIm_ISR	0000_00xxh	004Ch	32
FIFO Receive Status Register	RSCIm_FRSR	0000_0000h	0050h	32
FIFO Transmit Status Register	RSCIm_FTSR	0000_0000h	0054h	32
Reserve	-	-	0058h to 005Bh	-
Simple-LIN(SCIX) Status Register 0	RSCIm_XSR0	0000_0000h	005Ch	32
Simple-LIN(SCIX) Status Register 1	RSCIm_XSR1	0000_0000h	0060h	32
Reserve	-	-	0064h to 0067h	-
Common Flag Clear Register	RSCIm_CFCLR	0000_0000h	0068h	8, 16, 32
Simple I2C Flag Clear Register	RSCIm_ICFCLR	0000_0000h	006Ch	8, 16, 32
FIFO Flag Clear Register	RSCIm_FFCLR	0000_0000h	0070h	8, 16, 32
Reserve	-	-	0074h to 0077h	-
Simple-LIN(SCIX) Flag Clear Register	RSCIm_XFCLR	0000_0000h	0078h	8, 16, 32

Note 1. The read access size is fixed at 32 bits.

7.3.2.2 SCI Register Descriptions

The prefix (RSCIm_) of the register names is omitted in this and subsequent sections.

7.3.2.2.1 Receive Shift Register (RSCIm_RSR)

RSR is a shift register which is used to receive serial data input from the RXDn pin and converts it into parallel data. When one frame of data has been received, it is automatically transferred to the RDR register.

The RSR register cannot be directly accessed by the CPU.

7.3.2.2.2 Receive Data Register (RSCIm_RDR)

In FIFO mode (CCR3.FM = 1b), this register has 32-stage FIFO structure.

Access Size : 32 bits
Address : <RSCIm_base> + 0000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	FER	PER	-	-	ORER	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	FFER	FPER	DR	MPB	RDAT[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read.
28	FER	0h	R	Framing error flag CSR.FER can be read.
27	PER	0h	R	Parity error flag CSR.PER can be read.
26, 25	-	All 0	R	Reserved Whenever it is read, 0b is read.
24	ORER	0h	R	Overrun Error flag CSR.ORER can be read.
23 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read.
12	FFER	0h	R	FIFO framing error flag (Valid only in Asynchronous mode) 0b: There is no framing error in the data read from the receive FIFO. 1b: There is framing error in the data read from the receive FIFO.
11	FPER	0h	R	FIFO parity error flag (Valid only in Asynchronous mode) 0b: There is no parity error in the data read from the receive FIFO. 1b: There is parity error in the data read from the receive FIFO.
10	DR	0h	R	Receive data ready flag FRSR.DR can be read.
9	MPB	0h	R	Multi-processor flag 0b: Data transmission cycles 1b: ID transmission cycles
8 to 0	RDAT[8:0]	0h	R	Serial receive data RDAT is a 9-bit register for storing received data. Received data is stored in [6:0] when 7-bit data is selected, in [7:0] when 8-bit data is selected, and in [8:0] when 9-bit data is selected. And 0 is stored in the unused bit.

RDAT[8:0] bits (Serial receive data)

After one frame of data is received, the received data is transferred from the RSR register to this registers, thus allowing the RSR register to receive the next data.

The RSR and RDR registers have a double-buffered structure to enable continuous reception.

In Non-FIFO mode (CCR3.FM=0b), read RDR only once when a receive data full interrupt (RXI) request is issued. Without reading received data from RDR, if the next one frame is received, an overrun error occurs.

In FIFO mode, continuous reception is executed until 32 stages are stored. If data is read when there is no received data in the receive FIFO (RDR), the value is undefined. When the receive FIFO (RDR) are full of received data, subsequent serial receive data is lost.

The CPU cannot write to RDR.

0b is stored in the bit position which is not received (bit 8 or bit 7 of RDR) at the time of 7-bit or 8-bit communication in asynchronous mode.

MPB bit (Multi-processor flag)

In asynchronous mode, during multi-processor communication (CCR3.MP = 1b), the value of the multi-processor bit corresponding to the received data (RDAT[8:0]) can be read.

FPER bit (FIFO parity error flag)

Indicates whether the data read from the receive FIFO has a parity error.

In Non-FIFO mode (CCR3.FM=0b), 0b is stored.

FFER bit (FIFO framing error flag)

Indicates whether the data read from receive FIFO has a framing error.

In Non-FIFO mode (CCR3.FM=0b), 0b is stored.

7.3.2.2.3 Transmit Data Register (RSCIm_TDR)

In FIFO mode (CCR3.FM = 1b), this register has 32-stage FIFO structure.

Access Size : 8, 16, 32 bits
Address : <RSCIm_base> + 0004h
Initial Value : FFFF_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	MPBT	TDAT[8:0]								
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 1	RW	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
9	MPBT	1h	RW	Multi-processor transfer bit flag Value of the multi-processor bit in the transmission frame. This bit is use in Asynchronous mode. When writing to this bit when not used, write the initial value. 0b: Data transmission cycles 1b: ID transmission cycles
8 to 0	TDAT[8:0]	1FFh	RW	Serial transmit data TDAT is a 9-bit register for setting transmit data. Transmit data is set in [6:0] when 7-bit data is selected, in [7:0] when 8-bit data is selected, and in [8:0] when 9-bit data is selected. When byte access for 9-bit data, write TDR[15:8] and then write TDR[7:0].

Note: The read access size is fixed at 32 bits.

TDAT[8:0] bits (Serial transmit data)

The TDR is a 9-bit register for storing transmit data.

When empty space is detected in the TSR register, the transmit data stored in the TDR registers is transferred to TSR, and transmitting is started.

The TSR and TDR registers have a double-buffer structure to realize continuous transmission. When the next data to be transmitted is stored in TDR after one frame of data has been transmitted, the transmitting operation is continued by transfer to the TSR register.

In Non-FIFO mode, when a transmit data empty interrupt (TXI) request is issued and CCR0.TE is 1b, write transmit data to the TDR only once.

In FIFO mode, when the SCI detects that the transmit shift register (TSR) is empty, it transmits data in the transmit FIFO (TDR) into TSR and starts serial transmission. Continuous serial transmission is executed until there is no transmit data left in the transmit FIFO (TDR).

When transmit FIFO is full of transmit data (32 frames), no more data can be written. If new data is written, the data is ignored.

When using bytes access, write TDR[15:8] and then write TDR[7:0].

MPBT bit (Multi-processor transfer bit flag)

Selects the multi-processor bit of transmit frame.

7.3.2.2.4 Transmit Shift Register (RSCIm_TSR)

TSR is a shift register that transmits serial data. TSR cannot be directly accessed by the CPU.

To perform serial data transmission, the SCI first automatically transfers transmit data from TDR to TSR, and then sends the data to the TXDn pin.

7.3.2.2.5 Common Control Register 0 (RSCIm_CCR0)

Access Size : 8, 16, 32 bits

Address : <RSCIm_base> + 0008h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	SSE	-	-	TEIE	TIE	-	-	-	RIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	IDSEL	DCME	MPIE	-	-	-	TE	-	-	-	RE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
24	SSE	0h	RW	SSn# Pin Function Enable (Valid in Simple-SPI mode.) In slave mode (CCR3.CKE[1:0] = 1xb), set 1b to this bit. 0b: SSn# pin function is disabled. 1b: SSn# pin function is enabled.
23, 22	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
21	TEIE	0h	RW	Transmit End Interrupt Enable This bit should be set to 0 in smart card interface mode. 0b: TEI interrupt request is disabled. 1b: TEI interrupt request is enabled.
20	TIE	0h	RW	Transmit Interrupt Enable 0b: TXI interrupt request is disabled. 1b: TXI interrupt request is enabled.
19 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	RIE	0h	RW	Receive Interrupt Enable 0b: RXI and ERI interrupt requests are disabled. 1b: RXI and ERI interrupt requests are enabled.
15 to 11	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10	IDSEL	0h	RW	ID frame select (Valid only in asynchronous mode with multi-processor mode) 0b: All data is to be compared. 1b: The data with RDR.MPB = 1b is to be compared.
9	DCME	0h	RW*2	Data Compare Match Enable (Valid only in asynchronous mode) 0b: Address match function is disabled. 1b: Address match function is enabled.
8	MPIE	0h	RW*2	Multi-Processor Interrupt Enable (Valid in asynchronous mode when CCR3.MP = 1b.) This bit should be set to 0 in smart card interface mode. When the data with the multiprocessor bit set to 1b is received, the MPIE bit is automatically cleared to 0b, and non-multiprocessor reception is resumed. If you want to continue receiving operation using the multiprocessor function, set this bit to 1 sufficiently earlier than receiving the STOP bit of the next received frame. (Consider the synchronization delay time.) 0b: Non-Multi-Processor reception 1b: Multi-Processor reception
7 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
4	TE*3	0h	RW*1	Transmit Enable 0b: Serial transmission is disabled. 1b: Serial transmission is enabled.
3 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	RE*3	0h	RW*1	Receive Enable In clock synchronous mode and simple-SPI mode, receive only setting is prohibited (TE = 0 and RE = 1b setting prohibited). 0b: Serial reception is disabled. 1b: Serial reception is enabled.

Note: The read access size is fixed at 32 bits.

Note 1. In clock-synchronous mode (CCR3.MOD[2:0] = 010b), simple-SPI mode (CCR3.MOD[2:0] = 011b), and simple-I2C mode (CCR3.MOD[2:0] = 100b), 1b can be written only when TE = 0b and RE = 0b. After setting TE or RE to 1b, only 0b can be written in TE and RE. In other mode, writing is enabled under any condition.

Note 2. This bit is a bit that is cleared by hardware. Note that writing to a bit other than this bit with a bit manipulation instruction may cause this bit to be unintentionally set to 1b by a read-modify-write operation.

Note 3. In clock-synchronous mode or simple-SPI mode and internal clock (master mode), the receive-only setting is prohibited (TE = 0b and RE = 1b settings are prohibited).

RE bit (Receive Enable)

Enables or disables serial receive operation.

When this bit is set to 1b, serial reception becomes possible after the synchronization delay time has elapsed in asynchronous mode or the synchronous clock input in clock synchronous mode or start bit in smart-card-interface mode.

Note that CCR3 should be set prior to setting the RE bit to 1b in order to designate the reception format.

Except smart-card-interface-mode, even if reception is halted by setting the RE bit to 0b, the CSR.RDRF, FER, PER, ORER, FRSR.DR flags are not affected and the previous values are retained. In smart-card-interface-mode, even if reception is halted by setting the RE bit to 0b, the CSR.FER, PER, ORER flags are not affected and the previous value is retained. Also, to stop the reception operation, synchronization delay time will be required from when the RE bit is set to 0b until the reception operation is stopped.

TE bit (Transmit Enable)

Enables or disables serial transmission.

When this bit is set to 1b, serial transmission becomes possible after the synchronization delay time has elapsed. After the synchronization delay time, transmission is started by writing transmit data to TDR. Note that CCR3 should be set prior to setting the TE bit to 1b in order to designate the transmission format. In addition, the synchronization delay time is required until the transmission control circuit is stopped after the TE bit is set to 0b.

MPIE bit (Multi-Processor Interrupt Enable)

When this bit is set to 1b and the data with the multi-processor bit set to 0b is received, the data is not read and setting the status flags (CSR.RDRF, ORER, FER, FRSR.DR) are not set.

When the data with the multi-processor bit set to 1b is received, the MPIE is automatically cleared to 0b, and normal reception is resumed. For details, see **7.3.4 Multi-Processor Communication Function**. If you want to continue receiving operation using the multiprocessor function, set this bit to 1b sufficiently earlier than receiving the STOP bit of the next received frame.

When the receive data includes the MPB bit set to 0b, the receive data is not transferred from the RSR to the RDR, a receive error is not detected, and setting the flags ORER and FER to 1b is disabled.

When the receive data includes the MPB bit set to 1b, the MPIE bit is automatically cleared to 0b, the RXI and ERI interrupt requests are enabled (if CCR0.RIE is set to 1b), and setting the flags ORER, FER to 1b is enabled.

MPIE should be set to 0b if multi-processor communications function is not to be used.

DCME bit (Data Compare Match Enable)

It can select whether the Address match function (data compare match function) uses or not.

When DCME is 1b, if SCI detects the match to the comparison data (CCR4.CMPD) with receive data, DCME is cleared automatically, and after that, SCI operation mode will be receive mode without data compare match function.

See **7.3.3.6 Address Match (Receive Data Match Detection) Function**.

The write value should be 0b other than asynchronous mode.

IDSEL bit (ID frame select)

It can select whether it's compared in spite of the value of MPB bit or it's compared only the data of MPB bit = 1b (ID frame) when the Address match function is valid. Please set at the same time as DCME.

RIE bit (Receive Interrupt Enable)

Enables or disables RXI and ERI interrupt requests.

RXI and ERI interrupt request is disabled by setting the RIE bit to 0b.

An ERI interrupt request can be canceled by reading 1b from the CSR.ORER, FER, or PER and then setting the flag to 0, or setting the RIE bit to 0b.

TIE bit (Transmit Interrupt Enable)

Enables or disables a TXI interrupt request.

A TXI interrupt request is disabled by setting the TIE bit to 0b. At the beginning of transmission, set 1b to CCR0.TE and CCR0.TIE simultaneously. Then the TXI interrupt request is generated.

TEIE bit (Transmit End Interrupt Enable)

Enables or disables a TEI interrupt request.

A TEI interrupt request is disabled by setting the TEIE bit to 0b.

In simple I2C mode, the TEI is allocated to the interrupt on completion of issuing a start, restart, or stop condition (STI). In this case, the TEIE bit can be used to enable or disable the STI.

SSE bit (SSn# Pin Function Enable)

Set this bit to 1b if the SSn# pin is to be used in control of transmission and reception (in simple SPI mode). Set this bit to 0b in any other mode. Do not set both the SSE and CTSE (even if this setting is made, operation is the same as that when these bits are set to 0b).

In the slave mode (CCR3.CKE[1:0] = 10b or 11b), SSE should be set to 1b.

In the master mode (CCR3.CKE[1:0] = 00b or 01b) and single-master, the SSn# pin on the master side is not required to control reception and transmission, so SSE should be set to 0b.

7.3.2.2.6 Common Control Register 1 (RSCIm_CCR1)

Access Size : 8, 16, 32 bits

Address : <RSCIm_base> + 000Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	NFEN	-	NFCS[2:0]			-	-	-	SHARPS	-	-	-	SPLP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	RINV	TINV	-	-	PM	PE	-	-	SPB2IO	SPB2DT	-	-	CTSPEN	CTSE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
28	NFEN	0h	RW	Digital Noise Filter Function Enable (Valid in asynchronous mode, simple-LIN mode and simple-I2C mode) Noise filter function is available for RXD only in asynchronous mode and simple-LIN mode, for SCL and SDA inputs in simple-I2C mode. 0b: Noise filter disabled 1b: Noise filter enabled
27	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
26 to 24	NFCS[2:0]	0h	RW	Noise Filter Clock Select (Valid in asynchronous mode, simple-LIN mode and simple-I2C mode.) Select for the noise filter's clock source. In simple-I2C mode, 000b setting is prohibited. "The baud rate generator source clock" means the clock selected by CCR2.CKS[1:0]. 000b: The base clock signal divided by 1. 001b: The baud rate generator source clock divided by 1. 010b: The baud rate generator source clock divided by 2. 011b: The baud rate generator source clock divided by 4. 100b: The baud rate generator source clock divided by 8. Others: Setting prohibited
23 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20	SHARPS	0h	RW	Half-duplex communication select In the simple I2C mode or in the simple SPI mode, this bit should be set to 0. 0b: The TXD pin and the RXD pin independent 1b: Half-duplex communication using TXD pin
19 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	SPLP	0h	RW	Loopback Control This function is available in asynchronous mode and clock synchronous mode. 0b: Normal mode 1b: Loopback mode
15, 14	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
13	RINV	0h	RW	RXD invert 0b: Received data from RXD is not inverted and input.*2 1b: Received data from RXD is inverted and input.
12	TINV	0h	RW	TXD invert 0b: Transmit data is not inverted and output to TXD.*2 1b: Transmit data is inverted and output to TXD.
11, 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
9	PM	0h	RW	Parity Mode (Valid only when PE = 1b) 0b: Selects even parity 1b: Selects odd parity
8	PE	0h	RW	Parity Enable (Valid only in asynchronous mode. In Smart Card Interface mode, set 1b to this bit.) 0b: Parity bit addition is not performed during transmission. Parity bit checking is not performed during reception. 1b: The parity bit is added during transmission. The parity bit is checked during reception.
7, 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	SPB2IO	0h	RW	Serial port break I/O This bit selects output or no output on the TXD pin when CCR0.TE = 0b.*1 0b: The value of SPB2DT bit is not output on the TXD pin. 1b: The value of SPB2DT bit is output on the TXD pin.
4	SPB2DT	0h	RW	Serial port break data select The output level of TXD pin is selected when CCR0.TE = 0b.*1 0b: Low level is output on the TXD pin when TINV = 0b. High level is output on the TXD pin when TINV = 1b. 1b: High level is output on the TXD pin when TINV = 0b. Low level is output on the TXD pin when TINV = 1b.
3, 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	CTSPEN	0h	RW	CTS external pin Enable 0b: 1-pin mode, CTS and RTS share 1 pin. 1b: 2-pin mode, CTS and RTS have dedicated pins.
0	CTSE	0h	RW	CTS Enable 0b: CTS function is disabled (RTS output function is enabled). 1b: CTS function is enabled.

Note: The read access size is fixed at 32 bits.

Note 1. Use this bit in asynchronous mode. Operation in any other mode is not guaranteed.

Note 2. RINV/TINV should be set to 0b in smart card interface mode and simple I2C mode.

CTSE bit (CTS Enable)

Set this bit to 1b if the SSn# pin is to be used for CTS input. The RTS signal is output when this bit is set to 0b. Set this bit to 0b in smart card interface mode, simple SPI mode, simple-LIN mode, and simple I2C mode. Do not set both the CTSE and SSE bits (even if this setting is made, operation is the same as that when these bits are set to 0b).

CTSPEN bit (CTS external pin Enable)

When CTSE is 1b, select the terminals usage method when using the CTS and RTS functions. Set this bit to 1b when assigning the CTS/RTS function to 2 terminals and using them at the same time. Set it to 0b except in asynchronous mode.

Table 7.3-5 shows the relationship between the CTSE bit and CTSPEN bit settings and the functions of the CTSn#/RTSn# terminals.

Table 7.3-5 CTSE Bit and CTSPEN Bit Settings and Pin Functions

CTSE Bit	CTSPEN Bit	CTSn#/RTSn# Pin	CTSn# Pin
0	0	Output RTS	Not used
1	0	Input CTS	Not used
1	1	Output RTS	Input CTS
0	1	Prohibited	Prohibited

SPB2DT bit (Serial port break data select), SPB2IO bit (Serial port break I/O)

The TXD pin status decided by combination of CCR0.TE bit, CCR1.SPB2IO bit and CCR1.SPB2DT bit is indicated in **Table 7.3-6**.

Table 7.3-6 TXD Pin Status

CCR0.TE	CCR1.SPB2IO	CCR1.SPB2DT	TXD Pin Status (when CCR1.TINV = 0b)
0	0	x	Hi-Z (initial value)
0	1	0	Low level output
0	1	1	High level output
1	x	x	Serial transmission data is output.

PE bit (Parity Enable)

When PE bit to 1b, the parity bit is added to transmit data before transmission, and the parity bit is checked in reception.

In the multiprocessor format, the parity bit is not added or checked regardless of this bit setting.

PM bit (Parity Mode)

Selects the parity mode for transmission and reception (even or odd). In multi-processor mode, this bit is invalid.

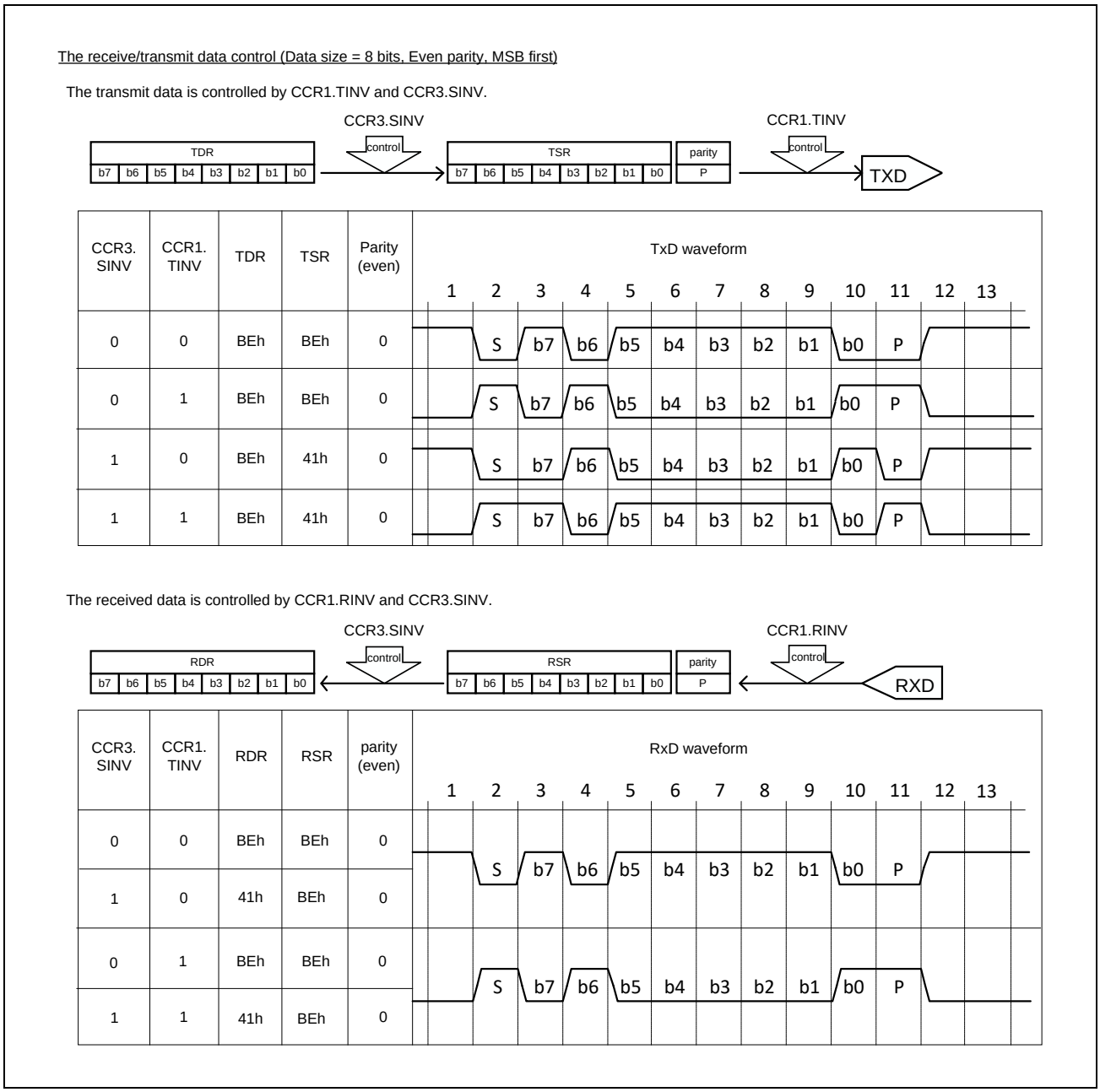
For details on the usage of this bit in smart card interface mode, see **7.3.5.2 Data Format (Except in Block Transfer Mode)**.

TINV bit (TXD invert), RINV bit (RXD invert)

The data of RDR is controlled by RINV and CCR3.SINV. And the data from TXD pin is controlled by TINV and CCR3.SINV. The control by RINV/TINV are done to communication pins (RXD/TXD), so they can control not only data- bits but also other bits (start bit, stop bit, parity bit). For details, see **Figure 7.3-2**.

During half-duplex communication and slave operation in simple SPI mode, use the TXD pin for reception, so set the inversion control of the received data with the TINV bit.

Note: Description and timing charts in this section assumes CCR1.TINV = 0b and CCR1.RINV = 0b when TINV and RINV value are not specified.



CCR3. SINV	CCR1. TINV	TDR	TSR	Parity (even)
0	0	BEh	BEh	0
0	1	BEh	BEh	0
1	0	BEh	41h	0
1	1	BEh	41h	0

CCR3. SINV	CCR1. TINV	RDR	RSR	parity (even)
0	0	BEh	BEh	0
1	0	41h	BEh	0
0	1	BEh	BEh	0
1	1	41h	BEh	0

Figure 7.3-2 Example of the Receive/Transmit Data Control

SPLP bit (Loopback Control)

When SPLP = 1b, TXD is connected to RXD internally.

Transmit data can be inverted and received by TINV = 1b.

Set to 0b in asynchronous mode with external clock, clock synchronous mode, slave, and Simple-LIN mode.

SHARPS bit (Half-duplex communication select)

Setting this bit to 1b enables half-duplex communication using the TXD pin. However, it cannot be used in Simple-SPI mode, Simple-I2C mode, and Smart Card Interface mode.

If this bit is set to 1b and CCR0.TE = 1b, CCR0.RE = 0b, the TXD pin is for communication output. If this bit is set to 1b and CCR0.TE = 0b, CCR0.RE = 1b, the TXD pin is for communication input. For details, see **7.3.12 RS-485 Driver Control Function**.

NFCS[2:0] bits (Noise Filter Clock Select)

These bits select the sampling clock for the digital noise filter.

To use the noise filter in asynchronous mode and simple-LIN mode, set these bits from 000b to 100b. In simple I2C mode, set the bits to a value in the range from 001b to 100b.

NFEN bit (Digital Noise Filter Function Enable)

This bit enables or disables the digital noise filter function. When the function is enabled, noise cancellation is applied to the RXDn input in asynchronous mode and simple-LIN mode, and noise cancellation is applied to the SDAn and SCLn input in simple I2C mode. In any mode other than above, set the NFEN bit to 0b.

7.3.2.2.7 Common Control Register 2 (RSCIm_CCR2)

Access Size : 8, 16, 32 bits

Address : <RSCIm_base> + 0010h

Initial Value : FF00_FF04h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MDDR[7:0]								-	-	CKS[1:0]		-	-	-	BRME
Initial Value	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BRR[7:0]								-	ABCSE	ABCS	BGDM	-	BCP[2:0]		
Initial Value	1	1	1	1	1	1	1	1	0	0	0	0	0	1	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	MDDR[7:0]	FFh	RW	Modulation Duty setting MDDR corrects the bit rate adjusted by the BRR register.
23, 22	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
21, 20	CKS[1:0]	0h	RW	Clock Select 00b: RSCI_m_TCLK clock (n = 0)*2 01b: RSCI_m_TCLK/4 clock (n = 1)*2 10b: RSCI_m_TCLK/16 clock (n = 2)*2 11b: RSCI_m_TCLK/64 clock (n = 3)*2
19 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	BRME	0h	RW	Bit Rate Modulation Enable 0b: Bit rate modulation function is disabled. 1b: Bit rate modulation function is enabled.
15 to 8	BRR[7:0]	FFh	RW	Bit rate setting BRR is an 8-bit register that adjusts the bit rate.
7	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6	ABCSE	0h	RW	Asynchronous Mode Extended Base Clock Select (Valid only in asynchronous mode and CCR3.CKE[1] = 0b) 0b: Clock cycles for 1-bit period are decided in combination with CCR2.BGDM and CCR2.ABCS. 1b: 6 base clock cycles for 1-bit period and doubled frequency is output from the baud rate generator.
5	ABCS	0h	RW	Asynchronous Mode Base Clock Select (Valid only in Asynchronous mode and simple-LIN mode) 0b: Selects 16 base clock cycles for 1-bit period. 1b: Selects 8 base clock cycles for 1-bit period.
4	BGDM	0h	RW	Baud Rate Generator Double-Speed Mode Select Valid in asynchronous/clock-synchronous/simple-SPI mode and CCR3.CKE[1] = 0b. 0b: Baud rate generator outputs the clock with single frequency. 1b: Baud rate generator outputs the clock with doubled frequency.
3	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	BCP[2:0]	4h	RW	Base Clock Pulse Selects the number of base clock cycles in smart card interface mode. 000b: 93 clock cycles (S = 93)*1 001b: 128 clock cycles (S = 128)*1 010b: 186 clock cycles (S = 186)*1 011b: 512 clock cycles (S = 512)*1 100b: 32 clock cycles (S = 32)*1 (Initial value) 101b: 64 clock cycles (S = 64)*1 110b: 372 clock cycles (S = 372)*1 111b: 256 clock cycles (S = 256)*1

Note: The read access size is fixed at 32 bits.

Note 1. S is the value of S in **Table 7.3-8**.

Note 2. n is the decimal notation of the value of n in **Table 7.3-8**.

BCP[2:0] bits (Base Clock Pulse)

The BCP[2:0] bits select the number of base clock cycles in a 1-bit data transfer time in smart card interface mode.

For details, see **7.3.5.4 Receive Data Sampling Timing and Reception Margin**.

BGDM bit (Baud Rate Generator Double-Speed Mode Select)

The BGDM bit is valid when the baud rate generator is selected as the clock source (CCR3.CKE[1] = 0b) in asynchronous mode, clock synchronous mode, simple-SPI mode. When external clock is selected (CCR3.CKE[1] = 1b), set it to 0b. For the clock output from the baud rate generator, either single or doubled frequency can be selected. The base clock is generated by the clock output from the baud rate generator. When the BGDM bit is set to 1b, the base clock cycle is halved and the bit rate is doubled.

Set this bit to 0b in modes other than asynchronous mode or clock synchronous mode or simple-SPI.

ABCS bit (Asynchronous Mode Base Clock Select)

Selects the clock cycles for 1-bit period.

Set it to 0b in modes other than asynchronous mode and simple-LIN mode.

ABCSE bit (Asynchronous Mode Extended Base Clock Select)

The pulse number for a base clock at 1-bit period is 6b and the clock of a double frequency is output from baud rate generator. Only when the bit rate is set to 6 dividing frequency of the bus clock, please use this bit and set CCR2.CKS[1:0] = 00b and BRR = 0b.

Set it to 0b in modes other than asynchronous mode. Even in asynchronous mode, set it to 0b when using external clock.

Table 7.3-7 Base Clock Cycle Number per 1-Bit

ABCSE Bit	ABCS Bit	BGDM Bit	The Base Clock Cycles/1 Bit	The Frequency of the Baud Rate Generator
0	0	0	16	×1
0	0	1	16	×2
0	1	0	8	×1
0	1	1	8	×2
1	x	x	6	×2

BRR[7:0] bits (Bit rate setting)

BRR is an 8-bit register that adjusts the bit rate.

SCI has independent baud rate generator control, different bit rates can be set for each. **Table 7.3-8** shows the relationship between the setting (N) in the BRR and the bit rate (B) for asynchronous mode, multiprocessor transfer, clock synchronous mode, smart card interface mode, simple SPI mode, and simple I2C mode.

Table 7.3-8 Relationship between N Setting in BRR and Bit Rate B

Mode	BGDM Bit	ABCS Bit	ABCSE Bit	BRR[7:0] Setting	Error (%)
Asynchronous multiprocessor transfer	0	0	0	$N = \frac{RSCI_{m_{TCLK}} \times 10^6}{64 \times 2^{2n-1} \times B} - 1$	$Error = \left\{ \frac{RSCI_m_TCLK \times 10^6}{B \times 64 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$
Simple-LIN*2	1	0	0		
	0	1	0	$N = \frac{RSCI_m_TCLK \times 10^6}{32 \times 2^{2n-1} \times B} - 1$	$Error = \left\{ \frac{RSCI_m_TCLK \times 10^6}{B \times 32 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$
	1	1	0	$N = \frac{RSCI_m_TCLK \times 10^6}{16 \times 2^{2n-1} \times B} - 1$	$Error = \left\{ \frac{RSCI_m_TCLK \times 10^6}{B \times 16 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$
	x	x	1	$N = \frac{RSCI_m_TCLK \times 10^6}{12 \times 2^{2n-1} \times B} - 1$	$Error = \left\{ \frac{RSCI_m_TCLK \times 10^6}{B \times 12 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$
Clock synchronous, Simple-SPI	0	0	0	$N = \frac{RSCI_m_TCLK \times 10^6}{8 \times 2^{2n-1} \times B} - 1$	—
	1	0	0	$N = \frac{RSCI_m_TCLK \times 10^6}{4 \times 2^{2n-1} \times B} - 1$	—
Smart card interface				$N = \frac{RSCI_m_TCLK \times 10^6}{S \times 2^{2n+1} \times B} - 1$	$Error = \left\{ \frac{RSCI_m_TCLK \times 10^6}{B \times S \times 2^{2n+1} \times (N+1)} - 1 \right\} \times 100$
Simple-I2C*1				$N = \frac{RSCI_m_TCLK \times 10^6}{64 \times 2^{2n-1} \times B} - 1$	—

Note: B: Bit rate (bps)
N: BRR setting for baud rate generator (0 ≤ N ≤ 255)
RSCI_m_TCLK: Operating frequency (MHz)
n and S: Determined by the settings of the CCR2 registers as listed in the table below. Please be careful about 2^{2n+1} is used in the expression for Smart card interface, 2^{2n-1} is used in other modes.

Note 1. Adjust the bit rate so that the widths at high and low level of the SCL output in simple I2C mode satisfy the I2C standard.

Note 2. In simple-LIN mode, BGDM = 0b and ABCSE = 0b can be selected.

Table 7.3-9 Calculating Widths at High and Low Level for SCL

Mode	SCL	Formula (Result in Seconds)
I^2C	Width at high level (minimum value)	$(N+1) \times 4 \times 2^{2n-1} \times 7 \times \frac{1}{RSCI_m_TCLK \times 10^6}$
	Width at low level (minimum value)	$(N+1) \times 4 \times 2^{2n-1} \times 8 \times \frac{1}{RSCI_m_TCLK \times 10^6}$

Table 7.3-10 Clock Source Settings

CCR2 Setting	Clock Source	n
CKS[1:0] = 00b	RSCI_m_TCLK clock	0
CKS[1:0] = 01b	RSCI_m_TCLK/4 clock	1
CKS[1:0] = 10b	RSCI_m_TCLK/16 clock	2
CKS[1:0] = 11b	RSCI_m_TCLK/64 clock	3

Table 7.3-11 Base Clock Settings in Smart Card Interface Mode

CCR2 Setting	Base Clock Cycles for 1-Bit Period	S
BCP[2:0] = 000b	93 clock cycles	93
BCP[2:0] = 001b	128 clock cycles	128
BCP[2:0] = 010b	186 clock cycles	186
BCP[2:0] = 011b	512 clock cycles	512
BCP[2:0] = 100b	32 clock cycles	32
BCP[2:0] = 101b	64 clock cycles	64
BCP[2:0] = 110b	372 clock cycles	372
BCP[2:0] = 111b	256 clock cycles	256

Table 7.3-12 lists examples of N settings in BRR in asynchronous mode. **Table 7.3-13** and **Table 7.3-18** list the maximum bit rate settable for each operating frequency. Examples of BRR (N) settings in clock synchronous mode and simple SPI mode are listed in **Table 7.3-15**. Examples of BRR (N) settings in smart card interface mode are listed in **Table 7.3-17**. Examples of BRR (N) settings in simple I2C mode are listed in **Table 7.3-19**. In smart card interface mode, the number of base clock cycles S in a 1-bit data transfer time can be selected. For details, see **7.3.5.4 Receive Data Sampling Timing and Reception Margin**. **Table 7.3-14** and **Table 7.3-16** list the maximum bit rates with external clock input. **Table 7.3-20** lists the minimum widths at high and low level for SCL at various bit rates.

When either the asynchronous mode base clock select bit (ABCS) or the baud rate generator double-speed mode select bit (BGDM) is set to 1b in asynchronous mode, the bit rate becomes twice that listed in **Table 7.3-12**.

When both of those registers are set to 1b, the bit rate becomes four times the listed value.

Table 7.3-12 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode)

	RSCI_m_TCLK = 100 MHz		
Bit Rate (bps)	n	N	Error (%)
9600	—	—	—
19200	0	162	−0.15
38400	0	80	0.47
57600	0	53	0.47
115200	0	26	0.47

Note: This is an example when CCR2.ABCS = 0b, CCR2.BGDM = 0b and CCR2.ABCSE = 0b.
When either ABCS bit or BGDM bit is set to 1b, the bit rate doubles.
When both ABCS = 1b and BGDM = 1b, the bit rate increases four times.

Table 7.3-13 Maximum Bit Rate for Each Operating Frequency (Asynchronous Mode)

RSCI_m_TCLK (MHz)	BGDM Bit	ABCS Bit	ABCSE Bit	n	N	MAX Baud Rate (Baud)
100	0	0	0	0	0	3,125,000
		1	0	0	0	6,250,000
	1	0	0	0	0	
		1	0	0	0	12,500,000
	x	x	1	0	0	16,666,667

Table 7.3-14 Maximum Bit Rate with External Clock Input (Asynchronous Mode)

RSCI_m_TCLK (MHz)	External Clock (MHz)	Max Baud Rate (Baud)	
		CCR2.ABCS = 0b	CCR2.ABCS = 1b
100	25	1,562,500	3,125,000

Table 7.3-15 BRR Settings for Various Bit Rates (Clock Synchronous Mode, Simple SPI Mode)

	RSCI_m_TCLK = 100 MHz		
Bit Rate (bps)	BGDM Bit	n	N
2.5 M	0	0	9
4 M	—	—	—
5 M	0	0	4
7.5 M	—	—	—
8 M	—	—	—
10 M	1	0	4
12 M	—	—	—
12.5 M	0	0	1
24 M	—	—	—
18.75 M	—	—	—
25 M	0	0	0
37.5 M	—	—	—

Note: '—' can be set, but an error over 10% will occur.

Table 7.3-16 Maximum Bit Rate with External Clock Input (Clock Synchronous Mode, Simple SPI Mode)

RSCI_m_TCLK (MHz)	External Clock (MHz)	MAX Bit Rate (Mbps)
100	25.00	25.00

Note: 37.5 MHz is max AC spec in this LSI.

Table 7.3-17 BRR Settings for Various Bit Rates (Smart Card Interface Mode, n = 0, S = 372)

	RSCI_m_TCLK = 100 MHz		
Bit Rate (bps)	n	N	Error (%)
9600	0	13	0.01

Table 7.3-18 Maximum Bit Rate for Each Operating Frequency (Smart Card Interface Mode, S = 32)

RSCI_m_TCLK (MHz)	MAX Bit Rate (bps)	n	N
100	1,562,500	0	0

Table 7.3-19 BRR Settings for Various Bit Rates (Simple I2C Mode)

Bit Rate (kbps)	RSCI_m_TCLK = 100 MHz		
	n	N	Error (%)
100	0	30	0.81
400	0	7	−2.34

Table 7.3-20 Minimum Widths at High and Low Level for SCL at Various Bit Rates (Simple I2C Mode)

	RSCI_m_TCLK = 100 MHz		
Bit Rate (kbps)	n	N	Minimum Width of High/Low Level (µs)
100	0	30	4.34/4.96
400	0	7	1.12/1.28

BRME bit (Bit Rate Modulation Enable)

Enables and disables the bit rate modulation function. The bit rate generated by baud rate generator is evenly corrected when this function is enabled.

Set to 0b in Clock-synchronous mode, Simple-SPI mode, Smart Card Interface mode and simple-LIN mode.

CKS[1:0] bits (Clock Select)

These bits select the clock source for the baud rate generator.

For the relationship between the settings of these bits and the baud rate, refer to BRR explanation.

MDDR[7:0] bits (Modulation Duty setting)

When the BRME bit is set to 1b, the bit rate generated by the baud rate generator is evenly corrected according to the settings of MDDR (M/256). The relationship between the MDDR setting (M) and the bit rate (B) is given in **Table 7.3-21**.

The initial value of MDDR is FFh. Bit 7 in this register is fixed to 1b.

Table 7.3-21 Relationship between MDDR Setting (M) and Bit Rate (B) Using Bit Rate Modulation Function

Mode*1	BGDM Bit	ABCS Bit	ABCSE Bit	BRR Setting	Error (%)
Asynchronous Multiprocessor transfer	0	0	0	$N = \frac{RSCI_m_TCLK \times 10^6}{64 \times 2^{2n-1} \times (256/M) \times B} - 1$	$Error = \left\{\frac{RSCI_m_TCLK \times 10^6}{B \times 64 \times 2^{2n-1} \times (256/M) \times (N+1)} - 1\right\} \times 100$
	1	0	0	$N = \frac{RSCI_m_TCLK \times 10^6}{32 \times 2^{2n-1} \times (256/M) \times B} - 1$	$Error = \left\{\frac{RSCI_m_TCLK \times 10^6}{B \times 32 \times 2^{2n-1} \times (256/M) \times (N+1)} - 1\right\} \times 100$
	0	1	0		
	1	1	0	$N = \frac{RSCI_m_TCLK \times 10^6}{16 \times 2^{2n-1} \times (256/M) \times B} - 1$	$Error = \left\{\frac{RSCI_m_TCLK \times 10^6}{B \times 16 \times 2^{2n-1} \times (256/M) \times (N+1)} - 1\right\} \times 100$
	x	x	1	$N = \frac{RSCI_m_TCLK \times 10^6}{12 \times 2^{2n-1} \times (256/M) \times B} - 1$	$Error = \left\{\frac{RSCI_m_TCLK \times 10^6}{B \times 12 \times 2^{2n-1} \times (256/M) \times (N+1)} - 1\right\} \times 100$
Simple-I2C*2				$N = \frac{RSCI_m_TCLK \times 10^6}{64 \times 2^{2n-1} \times (256/M) \times B} - 1$	

Note: B: Bit rate (bps)
M: MDDR setting (128 ≤ M ≤ 255)
N: BRR setting for bound rate generator (0 ≤ N ≤ 255)
RSCI_m_TCLK: Operating frequency (MHz)
n and S: Determined by the settings of the CCR2.BCP[2:0] and CKS[1:0] as listed in **Table 7.3-10** and **Table 7.3-11**.
Please be careful about 2^{2n+1} is used in the expression for Smart card interface, 2^{2n-1} is used in other modes.

Note 1. Do not use this function in Clock-synchronous mode, Simple-SPI mode, Smart card Interface mode and Simple-LIN mode.

Note 2. Adjust the bit rate so that the widths at high and low level of the SCL output in simple I2C mode satisfy the I2C standard.

Table 7.3-22 lists examples of N settings in BRR and M settings in MDDR in asynchronous mode.

Table 7.3-22 Examples of BRR and MDDR Settings for Various Bit Rates (Asynchronous Mode)

	RSCI_m_TCLK = 100 MHz				
Bit Rate (bps)	n	N	M	BGDM Bit	Error (%)
38400	0	47	151	0	0.00
57600	0	31	151	0	0.00
115200	0	15	151	0	0.00
230400	0	7	151	0	0.00
460800	0	3	151	0	0.00

Note: In this example when CCR2.ABCS = 0b and CCR2.ABCSE = 0b.
CCR2.BRME = 0b (M = 256) disables the bit rate modulation function.

7.3.2.2.8 Common Control Register 3 (RSCIm_CCR3)

Access Size : 8, 16, 32 bits

Address : <RSCIm_base> + 0014h

Initial Value : 0000_1203h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	BLK	GM	-	-	CKE[1:0]		-	-	DEN	FM	MP	MOD[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RXDESEL	STP	SINV	LSBF	-	-	CHR[1:0]		BPEN	-	-	-	-	-	CPOL	CPHA
Initial Value	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
29	BLK	0h	RW	Block Transfer Mode (Valid only in Smart card interface mode) 0b: Non-block transfer mode operation 1b: Block transfer mode operation
28	GM	0h	RW	GSM Mode (Valid only in Smart card interface mode) 0b: Non-GSM mode operation 1b: GSM mode operation
27, 26	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
25, 24	CKE[1:0]	0h	RW	Clock enable In the case of asynchronous mode When using the external clock 16 times the bit rate should be input from the SCKm pin when CCR2.ABCS bit is 0b. Input a clock signal with a frequency 8 times the bit rate when the CCR2.ABCS bit is 1b. 00b: The baud rate generator The SCKm pin is available for use as an I/O port in accord with the I/O port settings. 01b: The baud rate generator The clock with the same frequency as the bit rate is output from the SCKm pin. 1xb: External clock In the case of simple-LIN mode 00b:The baud rate generator The SCKm pin is available for use as an I/O port in accord with the I/O port settings. Others: Setting prohibited In the case of Clock synchronous mode, Simple-SPI 0xb: Internal clock (Master operation) The SCKm pin functions as the clock output pin. 1xb: External clock (Slave operation) The SCKm pin functions as the clock input pin. In the case of Smart card interface mode When CCR3.GM = 0b 00b: Output disabled (The SCKm pin can be used for other peripherals.) 01b: Clock output 1xb: Setting prohibited When CCR3.GM = 1b 00b: Output fixed low 10b: Output fixed high x1b: Clock output
23, 22	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
21	DEN	0h	RW	Driver enable (Valid only in Asynchronous mode) 0b: RS-485 Driver control function disable. 1b: RS-485 Driver control function enable.
20	FM	0h	RW	FIFO Mode select (Valid in Asynchronous mode (including multi-processor mode), Clock synchronous mode, and Simple SPI mode) 0b: Non-FIFO mode (TDR and RDR registers are non-FIFO). 1b: FIFO mode (TDR and RDR registers are FIFO). Note: When using DMAC, set to 1b
19	MP	0h	RW	Multi-Processor Mode (Valid in Asynchronous mode) 0b: Multi-processor communications function is disabled. 1b: Multi-processor communications function is enabled.
18 to 16	MOD[2:0]	0h	RW	Communication mode select Select the SCI communication mode. 000b: Asynchronous mode (Multi-processor mode) 001b: Smart card interface mode 010b: Clock synchronous mode 011b: Simple SPI mode 100b: Simple I2C mode 110b: Simple-LIN mode Others: Setting prohibited
15	RXDESEL	0h	RW	Asynchronous Start Bit Edge Detection Select (Valid only in asynchronous mode) Set this bit to 1b in simple-LIN mode. 0b: The low level on the RXDn pin is detected as the start bit. 1b: A falling edge on the RXDn pin is detected as the start bit.
14	STP	0h	RW	Stop Bit Length (Valid in asynchronous mode and simple-LIN mode) 0b: 1 stop bit / Break delimiter length = 1 bit length 1b: 2 stop bits / Break delimiter length = 2 bits length
13	SINV	0h	RW	Transmitted/Received Data Invert The level of communication pins (RXD/TXD) is controlled by the combination of this bit and CCR1.TINV/RINV. For details, see **Figure 7.3-2**. Set this bit to 0 in simple-I2C mode. 0b: TDR contents are transmitted to TSR as they are. RSR contents are stored to RDR as they are. 1b: TDR contents are inverted before being transmitted to TSR. RSR contents are inverted and stored to RDR.
12	LSBF	1h	RW	LSB First select Set this bit to 0b in simple I2C mode. Set this bit to 1b in simple-LIN mode. 0b: MSB first 1b: LSB first
11, 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9, 8	CHR[1:0]	2h	RW	Character Length (Valid in Asynchronous mode)*1 Select the data length for transmission and reception. 00b: Transmit/receive in 9-bit data length 01b: Transmit/receive in 9-bit data length 10b: Transmit/receive in 8-bit data length (initial value) 11b: Transmit/receive in 7-bit data length*2
7	BPEN	0h	RW	Synchronizer bypass enable This bit controls whether to bypass the synchronizer circuit between the bus clock and operation clock. When BPEN = 1, PCLKSPIn (operation clock) is input from RSCI_m_PCLK (Bus clock) and the synchronization circuit is bypassed. See **4.4 Clock Pulse Generator (CPG)**. 0b: Synchronizer circuit is not bypassed. 1b: Synchronizer circuit is bypassed.
6 to 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	CPOL	1h	RW	Clock Polarity Select (Valid in Clock-synchronous mode and Simple-SPI mode. Set this bit only when CCR0.TE = 0b and RE = 0b.) 0b: SCKm in idle state is 0. 1b: SCKm in idle state is 1.

Bit	Bit Name	Initial Value	R/W	Description
0	CPHA	1h	RW	Clock Phase Select (Valid in Clock-synchronous mode and Simple-SPI mode. Set this bit only when CCR0.TE = 0b and RE = 0b.) 0b: Data is sampled at an odd edge and changes at an even edge. (Clock is delayed.) 1b: Data changes at an odd edge and is sampled at an even edge. (Clock is not delayed.)

Note: The read access size is fixed at 32 bits.

Note 1. In modes other than asynchronous mode, the setting of these bits is invalid and a fixed data length of 8 bits is used. In simple-LIN mode, the data length of 8 bits can only be used, so set these bits to their initial value.

Note 2. LSB first is fixed and the MSB (bit 7) in TDR is not transmitted in transmission.

CPHA bit (Clock Phase Select)

This bit selects the phase of the clock signal output through the SCKm pin. For details, see **Figure 7.3-96**.

Set the bit to 1b in other than simple SPI mode and clock synchronous mode.

CPOL bit (Clock Polarity Select)

This bit selects the polarity of the clock signal output through the SCKm pin. For details, see **Figure 7.3-96**.

Set the bit to 1b in other than simple SPI mode and clock synchronous mode.

BPEN bit (Synchronizer bypass enable)

The synchronization circuit can be bypassed by this bit only when the same clock is input to the bus clock and the operation clock (RSCI_m_TCLK). For details, see **7.3.15 Synchronizer Bypass Function**.

CHR[1:0] bits (Character Length)

Selects the data length for transmission and reception.

Except of asynchronous mode, a fixed data length of 8 bits is used.

LSBF bit (LSB First select)

Select whether to transmit/receive data in MSB first or LSB first.

SINV bit (Transmitted/Received Data Invert)

SINV can invert the transmit data-bit from TDR to TSR, and also can invert the received data from RSR to RDR. This bit does not affect the logic level of the parity bit. To invert the parity bit, invert the CCR1.PM.

STP bit (Stop Bit Length)

Selects the stop bit length in transmission.

In reception, only the first stop bit is checked regardless of this bit setting. If the second stop bit is 0b, it is treated as the start bit of the next transmit frame.

In addition, it is used as the Break delimiter length setting when sending Start Frame in simple-LIN mode.

RXDESEL bit (Asynchronous Start Bit Edge Detection Select)

Selects the detection method of the start bit for reception in asynchronous mode. When a break occurs, data receiving operation depends on the settings of this bit. Set this bit to 1b when reception should be stopped while a break occurs or when reception should be started without retaining the RXDn pin input at high level for the period of one data frame or longer after completion of the break.

Set this bit to 1b in simple-LIN mode. Set this bit to 0b in modes other than asynchronous mode and simple-LIN mode.

MOD[2:0] bits (Communication mode select)

Selects the SCI communication mode.

Table 7.3-23 Relationship between Communication Mode Selection Bits (MOD[2:0]), Other Operation Mode Setting Bits

Communication Mode	Asynchronous								SMIF*1	Clock Synchronous		Simple SPI		Simple I2C	Simple LIN
CCR3.MOD[2:0]	000b								001b	010b		011b		100b	110b
CCR3.MP	0				1				—	—		—		—	—
CCR3.FM	0		1		0		1		—	0	1	0	1	—	—
CCR3.DEN	0	1	0	1	0	1	0	1	—	—		—		—	—
CCR0.SSE	—								—	—		0	1	—	—

Note: '—' means setting prohibited.

Note 1. Smart card interface

MP bit (Multi-Processor Mode)

Disables/enables the multi-processor communications function. The settings of the PE bit and PM bit are invalid in multi- processor mode.

FM bit (FIFO Mode select)

When FM = 1b, the TDR register/RDR register switches to FIFO mode.

DEN bit (Driver enable)

Select RS-485 Driver control function disable or enable.

CKE[1:0] bits (Clock enable)

These bits select the clock source and SCKm pin function.

In smart card interface mode, these bits control the clock output from the SCKm pin.

In GSM mode, clock output can be dynamically switched. For details, see **7.3.5.8 Clock Output Control**.

GM bit (GSM Mode)

Setting this bit to 1b allows GSM mode operation.

In GSM mode, the CSR.TEND flag set timing is put forward to 11.0 etu (elementary time unit = 1-bit transfer time) from the start and the clock output control function is appended. For details, see **7.3.5.6 Serial Data Transmission (Except in Block Transfer Mode)** and **7.3.5.8 Clock Output Control**.

BLK bit (Block Transfer Mode)

Setting this bit to 1b allows block transfer mode operation.

For details, see **7.3.5.3 Block Transfer Mode**.

7.3.2.2.9 Common Control Register 4 (RSCIm_CCR4)

Access Size : 8, 16, 32 bits

Address : <RSCIm_base> + 0018h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AET	ATT[2:0]			AJD	AST[2:0]			-	-	-	-	-	-	ATEN	ASEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CMPD[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	AET	0h	RW	Adjustment edge for transmit timing (Valid in Asynchronous mode using internal clock and simple-LIN mode using internal clock) The adjustable edge is set by this bit. This bit is valid only when ATEN = 1b. For details, see **7.3.3.11 The Function of Adjust Transmit Timing (Asynchronous Mode)**. The function of adjust transmit timing (Asynchronous Mode). 0b: Adjust the rising edge timing, when CCR1.TINV = 0b. Adjust the falling edge timing, when CCR1.TINV = 1b. 1b: Adjust the falling edge timing, when CCR1.TINV = 0b. Adjust the rising edge timing, when CCR1.TINV = 1b.
30 to 28	ATT[2:0]	0h	RW	Adjustment value for Transmit timing (Valid in Asynchronous mode using internal clock and simple-LIN mode using internal clock) This bit is valid only when ATEN is 1b. The selected edge timing of TxD is adjusted by the following formula. Adjustment edge timing = base clock × the setting value of ATT[2:0] This setting timing is limited by setting the base clock cycles. For details, see **7.3.3.11 The Function of Adjust Transmit Timing (Asynchronous Mode)**. The function of adjust transmit timing (Asynchronous Mode).
27	AJD	0h	RW	Adjustment Direction for receive sampling timing (Valid in Asynchronous mode using internal clock and simple-LIN mode using internal clock) This bit is valid only when ASEN is 1b. Adjustment direction for RxD receive sampling timing is determined by this bit. For details, see **7.3.3.10 The Function of Adjust Receive Sampling Timing (Asynchronous Mode)**. The function of adjust receive sampling timing (Asynchronous Mode). 0b: The sampling timing is adjusted backward to the middle of bit. 1b: The sampling timing is adjusted forward to the middle of bit.
26 to 24	AST[2:0]	0h	RW	Adjustment value for receive Sampling Timing This bit is valid only when ASEN is 1b. In asynchronous mode and simple-LIN mode using internal clock The sampling timing of RXDn pin is adjusted from the middle of bit by the following formula. Adjustment sampling timing = base clock × the setting value of AST[2:0]. In Clock-synchronous mode and Simple-SPI mode using internal clock The RxD sampling timing can be adjusted by delaying by 1 to 4 RSCI_m_TCLK. 000b: 1 RSCI_m_TCLK delay 001b: 2 RSCI_m_TCLK delay 010b: 3 RSCI_m_TCLK delay 011b: 4 RSCI_m_TCLK delay Others: Setting prohibited
23 to 18	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
17	ATEN	0h	RW	Adjust transmit timing enable (Valid only in asynchronous mode using internal clock) 0b: Adjust transmit timing disable. 1b: Adjust transmit timing enable.
16	ASEN	0h	RW	Adjust receive sampling timing enable (Valid in Asynchronous mode using internal clock, simple-LIN mode using internal clock, Clock-synchronous mode operating as master, and Simple-SPI mode operating as master) 0b: Adjust sampling timing disable. 1b: Adjust sampling timing enable.
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8 to 0	CMPD[8:0]	0h	RW	Compare Match Data (Valid only in Asynchronous mode) Set the compare data pattern for address match function.

Note: If the description in this document and the timing chart do not specify the ASEN / ATEN setting value, it means that the reception sampling adjustment function and transmission timing adjustment function are OFF (CCR4.ASEN = 0b, CCR4.ATEN = 0b).

Note: The read access size is fixed at 32 bits.

CMPD[8:0] bits (Compare Match Data)

Set the comparison data for receive data when the address match function is enabled (CCR0.DCME = 1b). CCR4.CMPD[8:0] should be written while CCR0.DCME is 0b.

For the comparison data, the length can be selected from 3 types: these are CMPD[6:0] for 7-bit length, CMPD[7:0] for 8-bit length, and CMPD[8:0] for 9-bit length.

ASEN bit (Adjust receive sampling timing enable)

When this bit is 1b, the receive sampling timing adjustment function is enabled. Control is different in Asynchronous mode, Simple-LIN mode, Clock-synchronous mode, and Simple-SPI mode.

For Asynchronous mode using internal clock, see **7.3.3.10 The Function of Adjust Receive Sampling Timing (Asynchronous Mode)** in details.

For Clock-synchronous mode operating as master and Simple-SPI mode operating as master, see **7.3.8.7 Reception Sampling Timing Adjustment Function in Clock Synchronous Mode with Internal Clock Used** in details. Only the digital delay of the master mode receive sampling clock (MRCLK) can be controlled by this bit.

ATEN bit (Adjust transmit timing enable)

When this bit is 1b, the transmission timing adjustment function is enabled. The transmission timing adjustment function can adjust the edge timing of the waveform output from the TXD pin. See **7.3.3.11 The Function of Adjust Transmit Timing (Asynchronous Mode)** in details.

AST[2:0] bits (Adjustment value for receive Sampling Timing)

When ASEN = 1b, the receive sampling timing can be adjusted according to this bit setting value.

In Asynchronous mode and Simple-LIN mode using internal clock

The sampling timing of RXD pin is adjusted from the middle of bit by the following formula. This setting value is limited by setting the base clock cycles. For details, see **7.3.3.10 The Function of Adjust Receive Sampling Timing (Asynchronous Mode)**.

Adjustment sampling timing = base clock × the setting value of AST[2:0].

In Clock-synchronous mode and Simple-SPI mode using internal clock

The RxD sampling timing can be adjusted by delaying 1 to 4 RSCI_m_TCLK. For details, see **7.3.8.7 Reception Sampling Timing Adjustment Function in Clock Synchronous Mode with Internal Clock Used**.

000: 1 RSCI_m_TCLK delay

001: 2 RSCI_m_TCLK delay

010: 3 RSCI_m_TCLK delay

011: 4 RSCI_m_TCLK delay 1xx: Setting prohibited

AJD bit (Adjustment Direction for receive sampling timing)

Set the RXD pin sampling timing adjustment direction from the bit center to the rear or front. See **7.3.3.10 The Function of Adjust Receive Sampling Timing (Asynchronous Mode)** for details.

ATT[2:0] bits (Adjustment value for Transmit timing)

The edge timing of the TXD pin specified by the AET bit is adjusted by the base clock × ATT[2:0] setting value. The upper limit of the adjustment time that can be set is limited by the number of base clock cycles. See **7.3.3.11 The Function of Adjust Transmit Timing (Asynchronous Mode)** for details.

AET bit (Adjustment edge for transmit timing)

Set the TXD pin edge for timing adjustment. See **7.3.3.11 The Function of Adjust Transmit Timing (Asynchronous Mode)** for details.

7.3.2.2.10 Communication Enable Status Register (RSCIm_CESR)

Access Size : 8 bits
Address : <RSCIm_base> + 001Ch
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	TIST	-	-	-	RIST
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

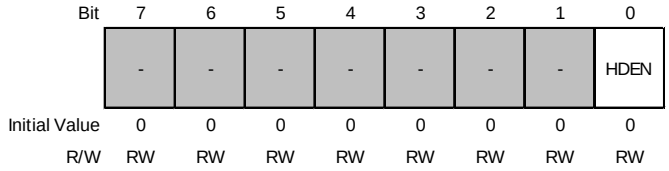
Bit	Bit Name	Initial Value	R/W	Description
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read.
4	TIST	0h	R	0b: TE signal internal state value “0” 1b: TE signal internal state value ”1”
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read.
0	RIST	0h	R	0b: RE signal internal state value “0” 1b: RE signal internal state value “1”

The operation clocks of the communication module and control register can be used asynchronously. Since some control register values are transmitted internally via the synchronization circuit so that they operate correctly even if they are asynchronous, it takes some time for the state to be reflected internally after rewriting the control register.

Communication enable CCR0.TE and CCR0.RE correspond to this register, and when these control bits are changed from 1b to 0b to rewrite the control bits for the next communication, it is necessary to rewrite these control bits after the internal state of the TE and RE signals becomes 0b. If a very slow clock is used as the clock for the communication module, the TE and RE bit values will be reflected slowly internally. At this time, you can check the internal state using this register.

7.3.2.2.11 Half Data Communication Control Register (RSCIm_HCR)

Access Size : 8 bits
Address : <RSCIm_base> + 001Eh
Initial Value : 00h



Bit	Bit Name	Initial Value	R/W	Description
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	HDEN	0h	RW	0b: HDC(Half data communication) function is disabled. 1b: HDC function is enabled.

HDEN (Half Data Communication Enable)

When this bit is 1b, data 0b is communicated as a low pulse only in the first half of the 1-bit period. This function permits setting to 1b only in asynchronous mode.

7.3.2.2.12 Simple I2C Control Register (RSCIm_ICR)

Access Size : 8, 16, 32 bits

Address : <RSCIm_base> + 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	IICSCLS[1:0]		IICSDAS[1:0]		-	IICSTPREQ	IICRSTAREQ	IICSTAREQ
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	IICACKT	-	-	-	IICCSC	IICINTM	-	-	-	IICDL[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23, 22	IICSCLS[1:0]	0h	RW	SCL Output Select 00b: Serial clock output 01b: Generate a start, restart, or stop condition. 10b: Output the low level on the SCLn pin. 11b: Place the SCLn pin in the high-impedance state.
21, 20	IICSDAS[1:0]	0h	RW	SDA Output Select 00b: Serial data output 01b: Generate a start, restart, or stop condition. 10b: Output the low level on the SDAn pin. 11b: Place the SDAn pin in the high-impedance state.
19	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18	IICSTPREQ	0h	RW	Stop Condition Generation 0b: A stop condition is not generated. 1b: A stop condition is generated. *2 *3 *4 *5
17	IICRSTAREQ	0h	RW	Restart Condition Generation 0b: A restart condition is not generated. 1b: A restart condition is generated. *2 *3 *4 *5
16	IICSTAREQ	0h	RW	Start Condition Generation 0b: A start condition is not generated. 1b: A start condition is generated. *1 *3 *4 *5
15, 14	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
13	IICACKT	0h	RW	ACK Transmission Data 0b: ACK transmission 1b: NACK transmission and reception of ACK/NACK
12 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	IICCSC	0h	RW	Clock Synchronization 0b: No synchronization with the clock signal 1b: Synchronization with the clock signal
8	IICINTM	0h	RW	I2C Interrupt Mode Select 0b: Use ACK/NACK interrupts. 1b: Use reception and transmission interrupts.
7 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
4 to 0	IICDL[4:0]	0h	RW	SDA Delay Output Select (Cycles below are of the clock signal from the baud rate generator.) 00h: No output delay 01h: 0 to 1 cycle 02h: 1 to 2 cycles 03h: 2 to 3 cycles 04h: 3 to 4 cycles 05h: 4 to 5 cycles ⋮ 1Eh: 29 to 30 cycles 1Fh: 30 to 31 cycles

Note: The read access size is fixed at 32 bits.

Note 1. In the bus free state, perform the start condition generation.

Note 2. In the bus busy state, perform restart or stop condition generation when the SCLn pin after acknowledgment described in **7.3.7 Simple I2C Mode** in **Figure 7.3-66** and **Figure 7.3-67** is low level.

Note 3. Do not set more than one from among the IICSTAREQ, IICRSTAREQ, and IICSTPREQ bits to 1b at a given time.

Note 4. Execute the generation of a condition after the value of the IICSTIF flag is 0b.

Note 5. Do not write 0b to this bit while it is 1b. Generation of a condition is suspended by writing 0b to this bit while it is 1b.

IICDL[4:0] bits (SDA Delay Output Select)

These bits are used to set a delay for output on the SDAn pin relative to the falling edge of the output on the SCLn pin. The available delay settings range from no delay to 31 cycles, with the clock signal from the baud rate generator as the base. The signal obtained by frequency-dividing RSCI_m_TCLK by the divisor set in CCR2.CKS[1:0] is supplied as the clock signal from the baud rate generator.

Set these bits to 00h unless operation is in simple I2C mode. In simple I2C mode, set these bits from 01h to 1Fh.

IICINTM bit (I2C Interrupt Mode Select)

This bit selects the sources of interrupt requests in simple I2C mode.

IICCSC bit (Clock Synchronization)

Set the IICCSC bit to 1b if the internally generated SCLn clock signal is to be synchronized when the SCLn pin has been placed at the low level in the case of a wait inserted by the other device, etc.

The SCLn clock signal is not synchronized if the IICCSC bit is 0b. The SCLn clock signal is generated in accord with the rate selected in the BRR regardless of the level being input on the SCLn pin.

Set the IICCSC bit to 1b except during debugging.

IICACKT bit (ACK Transmission Data)

Transmitted data contains ACK bits. Set this bit to 1b when ACK and NACK bits are received.

IICSTAREQ bit (Start Condition Generation)

When a start condition is to be generated, set both the IICSDAS[1:0] and IICSCLS[1:0] bits to 01b as well as setting the IICSTAREQ bit to 1b.

If you want to generate the start condition after generating the stop condition, start the generation of the start condition with a half cycle period of the bit rate from the stop condition generation interrupt request output.

[Setting condition]

- Writing 1b to the bit

[Clearing condition]

- Completion of generation of the start condition

IICRSTAREQ bit (Restart Condition Generation)

When a restart condition is to be generated, set both the IICSDAS[1:0] and IICSCLS[1:0] bits to 01b as well as setting the IICRSTAREQ bit to 1b.

[Setting condition]

- Writing 1b to the bit

[Clearing condition]

- Completion of generation of the restart condition

IICSTPREQ bit (Stop Condition Generation)

When a stop condition is to be generated, set both the IICSDAS[1:0] and IICSCLS[1:0] bits to 01b as well as setting the IICSTPREQ bit to 1b.

[Setting condition]

- Writing 1b to the bit

[Clearing condition]

- Completion of stop condition

IICSDAS[1:0] bits (SDA Output Select)

These bits control output from the SDAn pin.

IICSCLS[1:0] bits (SCL Output Select)

These bits control output from the SCLn pin.

7.3.2.2.13 FIFO Control Register (RSCIm_FCR)

Access Size : 8, 16, 32 bits

Address : <RSCIm_base> + 0024h

Initial Value : 1F1F_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	RSTRG[4:0]					RFRST	-	-	RTRG[4:0]				
Initial Value	0	0	0	1	1	1	1	1	0	0	0	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	W	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TFRST	-	-	TTRG[4:0]					-	-	-	-	-	-	-	DRES
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
28 to 24	RSTRG[4:0]	1Fh	RW	RTS# Output Active Trigger Number Select (Valid in Asynchronous mode (including multi-processor mode) and Clock-synchronous mode) These bits are valid only when CCR3.FM = 1b and CCR1.CTSE = 0b and CCR0.SSE = 0b. Trigger number must be set to 31 or less. 00h: Trigger number 0 ⋮ 1Fh: Trigger number 31
23	RFRST	0h	W	The read value is undefined Receive FIFO Data Register Reset This bit is valid only when CCR3.FM is 1b. The read value is always 0b. 0b: Invalid. This does not affect the operation. 1b: The number of data stored in receive FIFO (RDR register) is 0.
22 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20 to 16	RTRG[4:0]	1Fh	RW	Receive FIFO data trigger number (Valid in Asynchronous mode (including multi-processor mode), Clock-synchronous mode, and Simple-SPI mode) Trigger number must be set to 31 or less. When DMAC transfer is used, set RTRG[4:0] = 00h. 00h: Trigger number 0 ⋮ 1Fh: Trigger number 31
15	TFRST	0h	W	The read value is undefined Transmit FIFO Data Register Reset This bit is valid only when CCR3.FM is 1b. The read value is always 0b. 0b: Invalid. This does not affect the operation. 1b: The number of data stored in transmit FIFO (TDR register) is 0.
14 to 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12 to 8	TTRG[4:0]	0h	RW	Transmit FIFO data trigger number (Valid in Asynchronous mode (including multi-processor mode), Clock-synchronous mode, and Simple-SPI mode) Trigger number must be set to 31 or less. When DMAC transfer is used, set TTRG[4:0] = 1Fh. 00h: Trigger number 0 ⋮ 1Fh: Trigger number 31

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	DRES	0h	RW	Receive data ready error select (Valid in Asynchronous mode) This bit selects the interrupt request for receive data ready detection. 0b: Receive data full interrupt (RXI) 1b: Receive error interrupt (ERI)

Note: The read access size is fixed at 32 bits.

DRES bit (Receive data ready error select)

Select whether the detection of receive data ready (FRSR.DR = 1b) is the cause of RXI interrupt request or the cause of ERI interrupt request.

TTRG[4:0] bits (Transmit FIFO data trigger number)

The TDRE flag is set to 1b when the quantity of transmit data in the transmit FIFO (TDR register) is equal to or less than the specified transmit triggering number. If CCR0.TIE = 1b, TXI interrupt request is occurred.

TFRST bit (Transmit FIFO Data Register Reset)

When the TFRST bit is set to 1b, the number of the transmission data stored in transmit FIFO (TDR register) is made 0b.

RTRG[4:0] bits (Receive FIFO data trigger number)

The CSR.RDRF flag is set to 1b when the quantity of receive data in the receive FIFO (RDR register) is equal to or greater than the specified receive triggering number. If CCR0.RIE = 1b, RXI interrupt request is occurred. When FCR.RTRG is set to 0b, RDRF bit is set if the quantity of data in receive FIFO is greater than or equal to 1b.

RFRST bit (Receive FIFO Data Register Reset)

When the RFRST bit is set to 1b, the number of the reception data stored in receive FIFO (RDR register) is made 0b.

RSTRG[4:0] bits (RTS# Output Active Trigger Number Select)

When the quantity of receive data stored in the receive FIFO (RDR register) is equal to or greater than this number, the RTS# signal is in the High state. When FCR.RSTRG is set to 0b, RTS# is in the high state if the quantity of data in receive FIFO is greater than or equal to 1b.

7.3.2.2.14 Driver Control Register (RSCIm_DCR)

Access Size : 8, 16, 32 bits
Address : <RSCIm_base> + 0030h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	DENGT[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	DEAST[4:0]					-	-	-	-	-	-	-	DEPOL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20 to 16	DENGT[4:0]	0h	RW	Driver negation time (Valid only in Asynchronous mode) Set the driver negation time. When CCR3.DEN = 1b, the driver negation time is inserted after STOP bit transmission end. Setting DENGT[4:0] = 00h is prohibited.
15 to 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12 to 8	DEAST[4:0]	0h	RW	Driver Assertion Time (Valid only in Asynchronous mode) Set the driver assertion time. When CCR3.DEN = 1b, the driver assertion time is inserted in addition to the normal transmission waiting time. Setting DEAST[4:0] = 00h is prohibited.
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	DEPOL	0h	RW	Driver effective polarity select (Valid only in Asynchronous mode) 0b: The DE signal is active high. 1b: The DE signal is active low.

Note: The read access size is fixed at 32 bits.

DEPOL bit (Driver effective polarity select)

Select the active level of the DE signal.

DEAST[4:0] bits (Driver Assertion Time)

Set the driver assertion time (= time from the activation of the DE (Driver Enable) signal to the start of the start bit). It is expressed in base clock units (1/8 or 1/16 bit period) as follows.

Driver assertion time = value of DEAST[4:0] × base clock period + transmission waiting time

DENGT[4:0] bits (Driver negate time)

Set the driver negation time (= time from the end of the last stop bit of the transmitted message until the DE (Driver Enable) signal is disabled). It is expressed in base clock units (1/8 or 1/16 bit period) as follows.

Driver negation time = value of DENGT[4:0] × base clock period

If the transmission data is written during the driver negate time, transmit starting operation is different depends on the writing timing. (The DE signal remains valid, and transmission of the start bit may start after the transmission wait time

has elapsed. Also, the DE signal may become invalid once, and start bit transmission may start after the Driver assertion time + transmission wait time has elapsed.)

7.3.2.2.15 Simple-LIN (SCIX) Control Register 0 (RSCIm_XCR0)

Access Size : 8, 16, 32 bits

Address : <RSCIm_base> + 0034h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	BCCS[1:0]		-	AEDIE	COFIE	BFDIE	-	-	BCDIE	BFOIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIBS[2:0]			PIBE	CF1DS[1:0]		CF0RE	BFE	-	-	-	-	-	-	TCSS[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
25, 24	BCCS[1:0]	0h	RW	Select the sampling clock for the bus conflict detection circuit. When CCR2.ABCS=1b, setting BCCS[1:0]=1xb is prohibited. 00b: Base clock 01b: Base clock/2 10b: Base clock/4 11b: Prohibited (same as 10b)
23	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
22	AEDIE	0h	RW	Select whether to output an AED interrupt when a valid edge is detected. 0b: Active edge detection interrupt disabled 1b: Active edge detection interrupt enabled
21	COFIE	0h	RW	Select whether to include counter overflow as an ERI interrupt factor. 0b: Counter overflow is not included as an ERI interrupt factor 1b: Counter overflow is included as an ERI interrupt factor
20	BFDIE	0h	RW	Select whether to output a BFD interrupt when a break field is detected. 0b: Break Field detection interrupt disabled 1b: Break Field detection interrupt enabled
19, 18	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
17	BCDIE	0h	RW	Select whether to output an ERI interrupt when a bus collision is detected. 0b: Bus conflict detection is not included as an ERI interrupt factor 1b: Bus conflict detection is included as an ERI interrupt factor
16	BFOIE	0h	RW	Select whether to include Break Field output completion as a TXI interrupt factor. 0b: Break field output completion is not included as a TXI interrupt factor 1b: Break Field output completion is included as a TXI interrupt factor
15 to 13	PIBS[2:0]*2	0h	RW	Specify one of bits 0 to 7 of Control Field 1 as the priority interrupt bit. 000b: bit 0 of Control Field 1 001b: bit 1 of Control Field 1 010b: bit 2 of Control Field 1 011b: bit 3 of Control Field 1 100b: bit 4 of Control Field 1 101b: bit 5 of Control Field 1 110b: bit 6 of Control Field 1 111b: bit 7 of Control Field 1
12	PIBE	0h	RW	0b: Priority interrupt bit disable 1b: Priority interrupt bit enable

Bit	Bit Name	Initial Value	R/W	Description
11, 10	CF1DS[1:0]*2	0h	RW	Select the compare data for Control Field 1 00b: Select XCR1.PCF1D[7:0] as the compare data 01b: Select XCR1.SCF1D[7:0] as the compare data 10b: Select both XCR1.PCF1D[7:0] and XCR1.SCF1D[7:0] as the compare data 11b: Prohibited (same as 10b)
9	CF0RE*2	0h	RW	Set the presence or absence of Control Field 0 of Start Frame 0b: No Control Field 0 1b: With Control Field 0
8	BFE*2	0h	RW	Set the presence or absence of Break Field of Start Frame. 0b: No Break Field 1b: With Break Field
7 to 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1, 0	TCSS[1:0]*1	0h	RW	(Valid in Simple-LIN mode) Select the clock source of the timer in the LIN module. 00b: TCLK 01b: TCLK/4 10b: TCLK/16 11b: TCLK/64

Note: The read access size is fixed at 32 bits.

Note: Base clock: 1/16 period of 1 bit period when CCR2.ABCS = 0b, 1/8 period of 1 bit period when CCR2.ABCS = 1b.

Note 1. You can rewrite TCSS[1:0] only when the timer is stopped (TCST = 0b and SDST = 0b and BMEN = 0b)

Note 2. This bit is a setting bit required for Start Frame reception operation. Rewrite this bit when Start Frame reception or transmission is not in progress (XCR1.SDST = 0b and XCR1.TCST = 0b).

TCSS[1:0] (Timer count clock source select)

Select the clock source of the timer in Simple-LIN module.

BFE (Break Field enable)

Set the presence or absence of Break Field of Start Frame.

CF0RE (Control Field 0 enable)

Set the presence or absence of Control Field 0b of Start Frame.

CF1DS[1:0] (Control Field 1 compare data select)

Select the compare data for Control Field 1b.

PIBE (Priority interrupt bit enable)

Select whether to enable comparison of the priority interrupt bit with Control Field 1b. When this bit is 1b, regardless of the XCR1.CF1CE[7:0] setting value, the bit specified in PIBS[2:0] is compared with the primary comparison data for Control Field 1b (XCR1.PCF1D[7:0]).

PIBS[2:0] (Priority interrupt bit select)

Specify bit N (N = 0 to 7) of Control Field 1 as the priority interrupt bit.

BFOIE (Break Field output completion interrupt enable)

Select whether to include Break Field output completion as a TXI interrupt factor. Set CCR0.TIE = 1b and CCR3.MOD[1:0] = 110b, to output TXI upon completion of Break Field output.

BCDIE (Bus conflict detection interrupt enable)

Select whether to output an ERI interrupt when a bus collision is detected. In Simple-LIN mode (CCR3.MOD[1:0] = 110b), ERI output control is performed with this bit. When CCR3.MOD[1:0] = 110b and BCDIE = 1b, an ERI interrupt is issued when a bus collision is detected even if CCR0.RIE = 0b.

COFIE (Conter overflow interrupt enable)

Select whether to include counter overflow as an ERI interrupt factor. Setting CCR0.RIE = 1b and CCR3.MOD[1:0] = 110b is required to output ERI upon counter overflow.

AEDIE (Active edge detection interrupt enable)

Select whether to output an AED interrupt when a valid edge is detected. To output AED with valid edge detection, XCR1.BMEN = 1b and CCR3.MOD[1:0] = 110b must be set.

BCCS[1:0] (Bus conflict detection clock select)

Select the sampling clock for the bus conflict detection circuit.

7.3.2.2.16 Simple-LIN (SCIX) Control Register 1 (RSCIm_XCR1)

Access Size : 8, 16, 32 bits
Address : <RSCIm_base> + 0038h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CF1CE[7:0]								SCF1D[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCF1D[7:0]								-	-	BMEN	SDST	-	-	-	TCST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CF1CE[7:0]	0h	RW	Select whether to compare bit N of Control Field 1. (N = 0-7) 0b: Control Field 1 bit N compare disabled 1b: Control Field 1 bit N compare enabled
23 to 16	SCF1D[7:0]	0h	RW	The secondary compare data for Control Field 1
15 to 8	PCF1D[7:0]	0h	RW	The priority compare data for Control Field 1
7, 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	BMEN	0h	RW	0b: Bit rate measurement disabled 1b: Bit rate measurement enabled Set this bit to 1b simultaneously with the SDST bit. When this bit is set to 0b, it can be set to 0b at any timing.
4	SDST	0h	RW	0b: Start Frame/Break Field detection disabled 1b: Start Frame/Break Field detection enabled Do not set this bit and TCST bit to 1b at the same time.
3 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	TCST	0h	RW	0b: Break Field output timer count stopped 1b: Break Field output timer count start Do not set this bit and SDST bit to 1b at the same time.

Note: The read access size is fixed at 32 bits.

TCST (Break Field output start trigger)

[Clearing conditions]

- When 0b is written to TCST. Break Field output timer count is stopped and TXD output becomes idle.
- When Break Field output for the period set in XCR2.BFLW[15:0] is completed.

[Setting condition]

- When 1b is written to TCST. Start Break Field output from TXD. Holds 1b during Break Field output.

SDST (Start Frame detection start enable)

When 1b is written to this bit, Start Frame detection starts. When XCR0.BFE = 1b is set, Break Field can be detected during and after Start frame detection. When XCR0.BFE = 0b is set, Break Field is not detected. When 0b is written to this bit, Start Frame detection and Break Field detection are stopped. However, if XSR0.RXDSF = 0b when stopped, it is not possible to stop data reception of the SCI core with this bit. Write 0b to CCR0.RE to stop the reception operation or perform reception completion processing (CSR.RDRF clear or RDR read) after reception is completed.

BMEN (Bit rate measurement enable)

Set this bit to 1b simultaneously with the SDST bit. When this bit is set to 1b, the valid edge interval of Control Field 0b and Control Field 1b data is measured.

PCF1D[7:0] (Priority compare data for Control Field 1)

Set the priority compare data for Control Field 1.

SCF1D[7:0] (Secondary compare data for Control Field 1)

Set the secondary compare data for Control Field 1.

CF1CE[7:0] (Control Field 1 compare bit enable)

Select whether to compare bit N of Control Field 1. (N = 0 to 7)

When all of these bits are set to 0b (CF1CE[7:0] = 00h), it is always judged that Control Field 1 matches when reception is completed, and XSR0.CF1MF is set. This bit is a comparison enable with PCF1D[7:0] or SCF1D[7:0], and it is not a priority interrupt bit comparison enable.

7.3.2.2.17 Simple-LIN (SCIX) Control Register 2 (RSCIm_XCR2)

Access Size : 8, 16, 32 bits
Address : <RSCIm_base> + 003Ch
Initial Value : FFFE_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BFLW[15:0]															
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF0CE[7:0]								CF0D[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	BFLW[15:0]	FFFEh	RW	This register sets the Break Field length. The break field length is (BFLW [15:0] setting value + 1) x clock of the timer. The upper limit for setting this register is FFFEh. (Setting prohibited for FFFFh)
15 to 8	CF0CE[7:0]	0h	RW	Select whether to compare bit N of Control Field 0. (N = 0 - 7) 0b: Control Field 0 bit N compare disabled 1b: Control Field 0 bit N compare enabled
7 to 0	CF0D[7:0]	0h	RW	The compare data for Control Field 0

Note: The read access size is fixed at 32 bits.

CF0D[7:0] (Control Field 0 compare data)

The compare data for Control Field 0.

CF0CE[7:0] (Control Field 0 compare bit enable)

Select whether to compare bit N of Control Field 0. (N = 0 to 7)

When all of these bits are set to 0b (CF0CE[7:0] = 00h), it is always judged that Control Field 0 matches when reception is completed, and XSR0.CF0MF is set.

BFLW[15:0] (Break Field length setting)

BFLW[15:0] are 16-bit Break Field length setting bits and the initial value is FFFEh.

Set the break field length to 1 frame or more. The LIN standard stipulates that the Break Field length is 13 bits or more.

When sending the Break Field, write 1b to TCST. SCI starts output of the Break Field on TXD. Up-counting is performed with the clock of the timer selected by XCR0.TCSS[1:0]. When the count value matches the value set in this register, up-counting is stopped and Break Field output from TXD is also stopped.

When receiving the Break Field, write 1b to SDST to enable Start-Frame detection. SCI starts counting from the negative edge of RXD. The clock of the timer is selected by XCR0.TCSS[1:0]. When the count value matches the value set in this register, it is determined that a break field has been detected. Up-counting continues until the next valid edge or counter overflow.

7.3.2.2.18 Common Status Register (RSCIm_CSR)

Access Size : 32 bits

Address : <RSCIm_base> + 0048h

Initial Value : 6000_8000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDRF	TEND	TDRE	FER	PER	MFF	-	ORER	-	-	-	-	-	DFER	DPER	DCMF
Initial Value	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RXDMON	-	-	-	-	-	-	-	-	-	-	ERS	-	-	-	-
Initial Value	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RDRF	0h	R	Receive Data Full Flag 0b: No received data is in RDR register (Non-FIFO mode) The quantity of receive data written in receive FIFO falls below the specified receive triggering number (FIFO mode) 1b: Received data is in RDR register (Non-FIFO mode) The quantity of receive data written in receive FIFO is equal to or greater than the specified receive triggering number (FIFO mode)
30	TEND	1h	R	Transmit End Flag 0b: A character is being transmitted or standing by for transmission. 1b: Character transfer has been completed or Break field is being transmitted.
29	TDRE	1h	R	Transmit Data Empty Flag 0b: Transmit data is in TDR register (Non-FIFO mode) The quantity of transmit data written in transmit FIFO exceeds the specified transmit triggering number. (FIFO mode) 1b: No transmit data is in TDR register (Non-FIFO mode) The quantity of transmit data written in transmit FIFO is equal to or less than the specified transmit triggering number. (FIFO mode)
28	FER	0h	R	Framing Error Flag 0b: No framing error occurred (Non-FIFO mode) No framing error in all received data in receive FIFO (FIFO mode) 1b: A framing error has occurred (Non-FIFO mode) One or more framing errors occurred in received data in receive FIFO (FIFO mode)
27	PER	0h	R	Parity Error Flag 0b: No parity error occurred (Non-FIFO mode) No parity error in all received data in receive FIFO (FIFO mode) 1b: A parity error has occurred (Non-FIFO mode) One or more parity errors occurred in received data in receive FIFO (FIFO mode)
26	MFF	0h	R	Mode Fault Error Flag (Valid only in Simple-SPI mode) 0b: No mode fault error 1b: Mode fault error
25	-	0h	R	Reserved Whenever it is read, 0b is read.
24	ORER	0h	R	Overrun Error Flag 0b: No overrun error occurred. 1b: An overrun error has occurred.
23 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read.
18	DFER	0h	R	Data Compare Match Framing Error Flag (Valid only in Asynchronous mode) 0b: No framing error occurred at address match detection. 1b: A framing error has occurred at address match detection.

Bit	Bit Name	Initial Value	R/W	Description
17	DPER	0h	R	Data Compare Match Parity Error Flag (Valid only in Asynchronous mode) 0b: No parity error occurred at address match detection. 1b: A parity error has occurred at address match detection.
16	DCMF	0h	R	Data Compare Match Flag (Valid only in Asynchronous mode) 0b: Not matched 1b: Matched
15	RXDMON	1h	R	Serial input data monitor The state of the RXD pin without synchronizing by bus clock is shown. 0b: RXD pin is at the Low level when CCR1.RINV = 0b. RXD pin is at the High level when CCR1.RINV = 1b. 1b: RXD pin is at the High level when CCR1.RINV = 0b. RXD pin is at the Low level when CCR1.RINV = 1b.
14 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read.
4	ERS	0h	R	Error Signal Status Flag (Valid only in Smart card interface mode) 0b: Error signal Low not responded 1b: Error signal Low responded
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read.

ERS bits (Error Signal Status Flag)

[Setting condition]

- When an error signal LOW is sampled.

[Clearing condition]

- When 1b is written to CFCLR.ERSC.

DCMF bit (Data Compare Match Flag)

Indicates that SCI detects the match to the comparison data (CCR4.CMPD) with receive data.

Clearing the CCR0.RE bit to 0b does not affect the DCMF flag, which retains its previous state.

[Setting condition]

- Matching of the comparison data (CCR4.CMPD) with receive data, while CCR0.DCME = 1b.

[Clearing condition]

- When 1b is written to CFCLR.DCMFC.

DPER bit (Data Compare Match Parity Error Flag)

Indicates that a parity error occurred at Address Match detection (reception data match detection).

Clearing the CCR0.RE bit to 0b does not affect the DPER flag, which retains its previous state.

[Setting condition]

- When a parity error was detected by the frame in which an address match was detected.

[Clearing condition]

- When 1b is written to CFCLR.DPERC.

DFER bit (Data Compare Match Framing Error Flag)

Indicates that a framing error occurred at Address Match detection (reception data match detection).

Clearing the CCR0.RE bit to 0b does not affect the DFER flag, which retains its previous state.

[Setting condition]

- When a stop bit of the frame in which an Address Match was detected is 0b.
  When it is a 2-stop mode, the 1st bit of stop bit judges only whether it's 1 and does not check the 2nd bit of stop bit.

[Clearing condition]

- When 1b is written to CFCLR.DFERC.

ORER bit (Overrun Error Flag)

Indicates that an overrun error has occurred during reception and the reception ends abnormally.

Clearing the CCR0.RE bit to 0b does not affect the ORER flag, which retains its previous state. In simple-I2C mode, this bit is not use.

[Setting condition in Non-FIFO mode (CCR3.FM = 0b)]

- When the next data is received before reading out RDR with no-error reception data stored in RDR.
  In RDR, receive data prior to an overrun error occurrence is retained, but data received after the overrun error occurrence is lost. When the ORER flag is set to 1b, reception data is not forwarded to RDR register.
  Note that, in clock synchronous mode and simple-SPI mode, serial reception will be stopped.

[Setting condition in FIFO mode (CCR3.FM = 1b)]

- When the next serial reception is completed while the receive FIFO is full of NFIFO (= 32) receive data. (NFIFO: Indicates the number of available FIFO stages)

[Clearing condition]

- When 1b is written to CFCLR.ORERC.

MFF bit (Mode Fault Error Flag)

This bit indicates mode fault errors. In a multi-master mode, determine the mode fault error occurrence by reading the MFF flag.

[Setting condition]

- Input on the SSn# pin being at the low level during master operation (CCR3.CKE[1:0] = 00b or 01b) in simple SPI mode.

[Clearing condition]

- When 1b is written to CFCLR.MFFC.

PER bit (Parity Error Flag)

Indicates that a parity error has occurred during reception and the reception ends abnormally.

Clearing the CCR0.RE bit to 0b does not affect the PER flag, which retains its previous state.

In Clock-synchronous mode, Simple-SPI mode, and Simple-I2C mode, this bit is not used.

[Setting condition]

- When a parity error is detected during reception.
  In FIFO mode, when one or more parity errors are detected in receive FIFO data.
  In non-FIFO mode, although receive data where the parity error has occurred is transferred to RDR, no RXI interrupt request occurs. Note that while the PER flag is set to 1b, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 1b is written to CFCLR.PERC.

FER bit (Framing Error Flag)

Indicates that a framing error has occurred during reception and the reception ends abnormally.

Clearing the CCR0.RE bit to 0b does not affect the FER flag, which retains its previous state.

In Clock-synchronous mode, Simple-SPI mode, and Simple-I2C mode, this bit is not used.

[Setting condition]

- When 0 is sampled as the stop bit during reception.
  In FIFO mode, when one or more framing errors are detected in receive FIFO data. In simple-LIN mode, even if a condition that the bit changes to 1b occurs when XCR1.SDST = 1b, the FER set timing is delayed up to the Break Field judgment timing at the longest, since it may be a Break Field. If an edge is detected on the RXD signal before the Break Field judgment timing, FER is detected. If no edge is detected in the RXD signal before the Break Field judgment timing, Break Field is detected.
  In 2-stop-bit mode, only the first stop bit is checked whether it is 1b but the second stop bit is not checked. Note that in non-FIFO mode, although receive data where the framing error has occurred is transferred to RDR, no RXI interrupt request occurs. In addition, while the FER flag is set to 1b, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 1b is written to CFCLR.FERC.

TDRE bit (Transmit Data Empty Flag)

[Non-FIFO selected (CCR3.FM = 0b)]

Indicates the presence of transmit data in the TDR register.

The condition of CCR0.TE = 0b has priority over the condition of 0b.

If other conditions that become 1b and conditions that become 0b are satisfied at the same time, the TDRE flag is set to 0b.

[Setting condition]

- When CCR0.TE is 0b.
- When data is transmitted from the TDR register to the TSR register.

[Clearing condition]

- When 1b is written to CFCLR.TDREC.
- When the transmission data is written to the TDR register during CCR0.TE = 1b.

[FIFO selected (CCR3.FM = 1b)]

- Indicates that data has been transferred from the transmit FIFO (TDR) into the transmit shift register (TSR), the quantity of data in transmit FIFO has fallen below the specified transmit triggering number.
  When the condition which becomes 1b and the condition which becomes 0b were formed at the same time, TDRE flag will be 0b. After that, when the number of data stored in transmit FIFO is judged, and if that is equal to or less than TTRG value, TDRE is set to 1b after 1b RSCI_m_PCLK.

[Setting condition]

- When the quantity of transmit data written in transmit FIFO is equal to or less than the specified transmit triggering number.*[1]

[Clearing condition]

- When 1b is written to CFCLR.TDREC.

- When the transmission data is written to transmit FIFO by the DMAC.

Note 1. The transmit FIFO is 32-stage FIFO, the maximum number of data that can be written when the TDRE flag is 1b is “NFIFO (= 32) - FTSR.T[5:0]”. Even if data any more is written, data is discarded.

TEND bit (Transmit End Flag)

[Non-FIFO selected (CCR3.FM = 0b), and Not Smart card interface mode (CCR3.MOD[2:0] ≠ 001b)]

Indicates completion of transmission.

[Setting condition]

- When CCR0.TE is 0b.
- When the CCR0.TE bit is changed from 0b to 1b, the TEND flag is not affected and retains the value 1b.
- When the TDR register is not updated at transmission of the tail-end bit of a character.
- When the TDR register is not updated at the end of DE negate time with DE control function enable (CCR3.DEN = 1b).
- When Break Field is sent.

[Clearing condition]

- After the synchronization delay time has elapsed since the transmission data was written to the TDR register during CCR0.TE = 1b.
- When 1b is written to CFCLR.TDREC during CCR0.TE = 1b.

[Non-FIFO selected (CCR3.FM = 0b), and Smart card interface mode (CCR3.MOD[2:0] = 001b)]

With no error signal from the receiving side, this bit is set to 1b when next data is ready to be transferred to the TDR register.

[Setting condition]

- When CCR0.TE is 0b.
- When the CCR0.TE bit is changed from 0b to 1b, the TEND flag is not affected and retains the value 1.
- When ERS flag is 0b and TDR register is not updated after 1-byte transmission + fixed time, the set timing is determined by setting CCR3 register as listed below.
  - When GM = 0b and BLK = 0b, 12.5 etu after the start of transmission
  - When GM = 0b and BLK = 1b, 11.5 etu after the start of transmission
  - When GM = 1b and BLK = 0b, 11.0 etu after the start of transmission
  - When GM = 1b and BLK = 1b, 11.0 etu after the start of transmission

[Clearing condition]

- After the synchronization delay time has elapsed since the transmission data was written to the TDR register during CCR0.TE = 1b.
- When 1b is written to CFCLR.TDREC during CCR0.TE = 1b.

[FIFO selected (CCR3.FM = 1b)]

Indicates that the transmit FIFO does not contain valid data when transmitting the last bit of a serial character, so the transmission is halted.

[Setting condition]

- TEND is set to 1b when transmit FIFO does not contain transmit data when the last bit of a one-byte serial character is transmitted.
- When the TDR register is not updated at the end of DE negate time with DE control function enable (CCR3.DEN = 1b).

[Clearing condition]

- After the synchronization delay time has elapsed since the transmission data was written to the TDR register during CCR0.TE = 1b.

RDRF bit (Receive Data Full Flag)

[Non-FIFO selected (CCR3.FM = 0b)]

Indicates the presence of receive data in the RDR register.

[Setting condition]

- When the reception ends normally, and receive data is forwarded from the RSR register to the RDR register.

[Clearing condition]

- When 1b is written to CFCLR.RDRFC.
- When a data is read from the RDR register.

[FIFO selected (CCR3.FM = 1b)]

Indicates that receive data has been transferred to the receive FIFO data register (RDR), and the quantity of data in receive FIFO is equal to or greater than the specified receive triggering number. When FCR.RTRG is set to 0b, RDRF is set if receive FIFO is not empty.

[Setting condition]

RDRF is set to 1b when the quantity of receive data in receive FIFO is equal to or greater than the specified receive triggering number.*[1]

[Clearing condition]

- When 1b is written to CFCLR.RDRFC.
- When the reception data is read from receive FIFO by the DMAC.
  When setting condition and clearing condition are occurred at the same time, RDRF flag will be 0b. After that, if the data count in receive FIFO is equal to or greater than RTRG value, RDRF is set to 1b after 1 RSCI_m_PCLK.

Note 1. Since the receive FIFO is NFIFO (= 32) stage, the maximum quantity of data that can be read when RDRF is 1b is equal to FRSR.R[5:0]. Further data can be read, but the value is undefined.

Note: In Non-FIFO mode, except when interrupting communication, RDRF and TDRE should not be cleared by CFCLR register.

7.3.2.2.19 Simple I2C Status Register (RSCIm_ISR)

Access Size : 32 bits
Address : <RSCIm_base> + 004Ch
Initial Value : 0000_00xxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	IICSTIF	-	-	IICACKR
Initial Value	0	0	0	0	0	0	0	0	0	0	x	x	0	x	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read.
5 to 4	-	xh	R	Reserved Whenever it is read, undefined value is read.
3	IICSTIF	0h	R	Issuing of Start, Restart, or Stop Condition Completed Flag 0b: There are no requests for generating conditions or a condition is being generated. 1b: A start, restart, or stop condition is completely generated.
2	-	xh	R	Reserved Whenever it is read, undefined value is read.
1	-	0h	R	Reserved Whenever it is read, 0b is read.
0	IICACKR	0h	R	ACK Reception Data Flag 0b: ACK received 1b: NACK received

Note: x: Undefined

IICACKR bit (ACK Reception Data Flag)

Received ACK and NACK bits can be read from this bit.

The IICACKR flag is updated at the rising of SCLn clock for the ACK/NACK receiving bit.

IICSTIF bit (Issuing of Start, Restart, or Stop Condition Completed Flag)

After generating a condition, this bit indicates that the generation is completed. When using the IICSTAREQ, IICRSTAREQ, or IICSTPREQ, do it after clearing the IICSTIF flag.

When the IICSTIF flag is 1b while CCR0.TEIE = 1b, an STI request is output.

[Setting condition]

Completion of the generation of a start, restart, or stop condition (when setting condition and clearing condition are occurred at the same time, IICSTIF will be 0b).

[Clearing condition]

- Writing 1b to ICFCLR.IICSTIFC bit
- When operation is not in simple-I2C
- Writing 0b to CCR0.TE bit

7.3.2.2.20 FIFO Receive Status Register (RSCIm_FRSR)

Access Size : 32 bits
Address : <RSCIm_base> + 0050h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	FNUM[5:0]						-	-	PNUM[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	R[5:0]						-	-	-	-	-	-	-	DR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 30	-	All 0	R	Reserved Whenever it is read, 0b is read.
29 to 24	FNUM[5:0]	0h	R	Framing Error Count (Valid only in Asynchronous mode) Indicates the quantity of data with a framing error among the data in receive FIFO.
23 to 22	-	All 0	R	Reserved Whenever it is read, 0b is read.
21 to 16	PNUM[5:0]	0h	R	Parity Error Count (Valid only in Asynchronous mode) Indicates the quantity of data with a parity error among the data in receive FIFO.
15 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read.
13 to 8	R[5:0]	0h	R	Receive FIFO Data Count (Valid in Asynchronous mode (including multi-processor), Clock synchronous mode, Simple SPI mode, when CCR3.FM is 1b.) Indicate the quantity of data in receive FIFO.
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read.
0	DR	0h	R	Receive Data Ready flag 0b: Receiving is in progress, or no received data has remained in receive FIFO after normally completed reception. (Receive FIFO is empty.) 1b: The following receive data does not come for a fixed period after storing data below the threshold.

DR bit (Receive Data Ready flag)

Indicates that the quantity of data stored in the receive FIFO falls below the specified receive triggering number, and that no next data has been received yet after the elapse of 15 etu from the last stop bit in asynchronous mode. This bit is valid only asynchronous mode (including multi-processor mode) and FIFO selected. In other modes, this bit does not set to 1b.

[Setting conditions]

DR is set to 1b when both of the following conditions are met.

- Data in receive FIFO is less than the specified receive triggering number, and no next data is received after the elapse of 15 etu*1 from the last stop bit.
- CSR.FER, PER flags are 0b.

[Clearing conditions]

- When all receive data in the receive FIFO is read and 1b is written to FFCLR.DRC.

- When CCR3.FM bit is 0b.

Note 1. This is equivalent to one and a half (1.5) frames in the 8-bit format with one stop bit (etu: elementary time unit).

R[5:0] bits (Receive FIFO Data Count)

Indicate the quantity of receive data stored in receive FIFO.

00h means no receive data. NFIFO (= 32) means receive FIFO is full.

PNUM[5:0] bits (Parity Error Count)

Indicate the quantity of data with parity error in the received FIFO.

FNUM[5:0] bits (Framing Error Count)

Indicate the quantity of data with framing error in the received FIFO.

7.3.2.2.21 FIFO Transmit Status Register (RSCIm_FTSR)

Access Size : 32 bits

Address : <RSCIm_base> + 0054h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	T[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read.
5 to 0	T[5:0]	0h	R	Transmit FIFO Data Count (Valid in Asynchronous mode (including multi-processor), Clock synchronous mode, and Simple SPI mode, when CCR3.FM is 1b.) Indicate the quantity of non-transmitted data stored in transmit FIFO.

T[5:0] bits (Transmit FIFO Data Count)

Indicate the quantity of non-transmitted data stored in transmit FIFO.

00h means no un-transmit data. NFIFO (= 32) means transmit FIFO is full.

7.3.2.2.22 Simple-LIN (SCIX) Status Register 0 (RSCIm_XSR0)

Access Size : 32 bits

Address : <RSCIm_base> + 005Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CF1RD[7:0]								CF0RD[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AEDF	COF	PIBDF	CF1MF	CF0MF	BFDF	BCDF	BFOF	-	-	-	-	-	-	RXDSF	SFSF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CF1RD[7:0]	0h	R	Control Field 1 received data
23 to 16	CF0RD[7:0]	0h	R	Control Field 0 received data
15	AEDF	0h	R	0b: No valid edge detected 1b: Detected a valid edge
14	COF	0h	R	0b: Break Field detection counter has not overflowed 1b: Break Field detection counter
13	PIBDF	0h	R	0b: Priority interrupt bit is not detected 1b: Priority interrupt bit is detected
12	CF1MF	0h	R	0b: Control Field 1 received data does not match the setting data 1b: Control Field 1 received data matches the setting data
11	CF0MF	0h	R	0b: Control Field 0 received data does not match the setting data 1b: Control Field 0 received data matches the setting data
10	BFDF	0h	R	0b: BF is not detected 1b: BF is detected
9	BCDF	0h	R	0b: No bus collision detected 1b: Detected a bus collision
8	BFOF	0h	R	0b: Break Field is being output or is not being output 1b: Completed BF output
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read.
1	RXDSF	0h	R	0b: RXD input to RSCI core is enabled 1b: RXD input to RSCI core is disabled (RXD is not input to the RSCI core)
0	SFSF	0h	R	0b: Start Frame detection disabled or Start Frame detection complete 1b: Before Start Frame detection or during detection

Note: When RSCI_m_PCLK is faster than RSCI_m_TCLK, the flag set timing is delayed from the receive data full interrupt (RXI) output. To see this flag under these conditions, wait at least 1 RSCI_m_TCLK cycle after the receive data full interrupt (RXI) before reading this register.

SFSF (Start Frame status flag)

Indicates whether Start Frame is being detected.

[Setting conditions]

- When 1b is written to XCR1.SDST.
- When a Break Field is detected in the Control Field 0/ Control Field 1/Information Field phase and the transition to the Control Field 0 or Control Field 1 reception state occurs.

[Clearing conditions]

- When XCR1.SDST is 0b.
- When Start Frame detection is completed.

RXDSF (RXD input status flag)

Indicates the RXD input status to the SCI core. When this bit is 1b, RXD input is received only by the simple LIN module and the Break Field is detected and is not input to the SCI core.

BFOF (Break Field output completion flag)

Indicates the completion of Break Field output.

[Setting condition]

- When Break Field output is completed

[Clearing condition]

- When 1b is written to XFCLR.BFOC

BCDF (Bus conflict detection flag)

Indicates the detection of bus conflict in Simple-LIN transmit operation.

[Setting condition]

- When Bus Conflict is detected

[Clearing condition]

- When 1b is written to XFCLR.BCDC

BFDF (Break Field detection flag)

Indicates Break Field detection.

[Setting condition]

- When Break Field is detected

[Clearing condition]

- When 1b is written to XFCLR.BFDC

CF0MF (Control Field 0 compare match flag)

Indicates compare match between Control Field 0 and compare data.

[Setting condition]

- When Control-Field-0 data and the compare data match

[Clearing condition]

- When 1b is written to XFCLR.CF0MC

CF1MF (Control Field 1 compare match flag)

Indicates the detection of a compare match between Control Field 1 and compare data.

[Setting condition]

- When Control-Field-1 data and the compare data match

[Clearing condition]

- When 1b is written to XFCLR.CF1MC

PIBDF (Priority interrupt bit detection flag)

Indicates the detection of a compare match between Control Field 1 and priority interrupt bit.

[Setting condition]

- When the priority interrupt bit is detected

[Clearing condition]

- When 1b is written to XFCLR.PIBDC

COF (Counter overflow flag)

Indicates that the 16-bit counter in the simple LIN module has overflowed.

[Setting condition]

- When the counter for Break Field detection overflows

[Clearing condition]

- When 1b is written to XFCLR.COFC

AEDF (Active edge detection flag)

Indicates active edge detection.

[Setting condition]

- When Active edge is detected

[Clearing condition]

- When 1b is written to XFCLR.AEDC
- When XSR1.TCNT[15:0] are read out

CF0RD[7:0] (Control Field 0 received data)

Stores the received data with a Control Field 0 match detected.

CF1RD[7:0] (Control Field 1 received data)

Stores the received data with a Control Field 1 match detected.

7.3.2.2.23 Simple-LIN (SCIX) Status Register 1 (RSCIm_XSR1)

Access Size : 32 bits
Address : <RSCIm_base> + 0060h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read.
15 to 0	TCNT[15:0]	0h	R	Stores the 16-bit counter capture value. The initial value is 0000h.

TCNT[15:0] (Timer count capture value)

Stores the capture value of the 16-bit counter of the simple-LIN module.

When sending Start Frame, this register holds the previous value.

When receiving Start frame with Bit rate measurement disabled, if a break field is detected in the break field detection state (see **Figure 7.3-61**), the break field length is captured and held (counter value is captured at the rising edge of RXD). If a Break Field is detected in a state other than the Break Field detection state, the previous value is retained. If the counter overflows, it will not be captured.

When receiving Start frame with Bit rate measurement enabled, the count value is captured and held at the valid edge (both RXD edges). However, in the Break Field detection state, the count value is not captured even if a valid edge occurs. Counter capture value retention is canceled by reading this register. Even if a valid edge occurs before reading, the counter value is not captured.

7.3.2.2.24 Common Flag Clear Register (RSCIm_CFCLR)

Access Size : 8, 16, 32 bits

Address : <RSCIm_base> + 0068h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDRFC	-	TDREC	FERC	PERC	MFFC	-	ORERC	-	-	-	-	-	DFERC	DPERC	DCMFC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	ERSC	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31	RDRFC	0h	W	RDRF clear 0b: No effect 1b: Clear the CSR.RDRF bit
30	-	0h	W	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
29	TDREC	0h	W	TDRE clear 0b: No effect 1b: Clear the CSR.TDRE bit
28	FERC	0h	W	FER clear 0b: No effect 1b: Clear the CSR.FER bit
27	PERC	0h	W	PER clear 0b: No effect 1b: Clear the CSR.PER bit
26	MFFC	0h	W	MFF clear 0b: No effect 1b: Clear the CSR.MFF bit
25	-	0h	W	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
24	ORERC	0h	W	ORER clear 0b: No effect 1b: Clear the CSR.ORER bit
23 to 19	-	All 0	W	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18	DFERC	0h	W	DFER clear 0b: No effect 1b: Clear the CSR.DFER bit
17	DPERC	0h	W	DPER clear 0b: No effect 1b: Clear the CSR.DPER bit
16	DCMFC	0h	W	DCMF clear 0b: No effect 1b: Clear the CSR.DCMF bit
15 to 5	-	All 0	W	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	ERSC	0h	W	ERS clear 0b: No effect 1b: Clear the CSR.ERS bit
3 to 0	-	All 0	W	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Note: The read access size is fixed at 32 bits.

7.3.2.2.25 Simple I2C Flag Clear Register (RSCIm_ICFCLR)

Access Size : 8, 16, 32 bits

Address : <RSCIm_base> + 006Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	IICSTIFC	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	W	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3	IICSTIFC	0h	W	IICSTIF clear 0b: No effect 1b: Clear the ISR.IICSTIF bit
2 to 0	-	All 0	W	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Note: The read access size is fixed at 32 bits.

7.3.2.2.26 FIFO Flag Clear Register (RSCIm_FFCLR)

Access Size : 8, 16, 32 bits

Address : <RSCIm_base> + 0070h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DRC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	W	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	DRC	0h	W	DR clear 0b: No effect 1b: Clear the FRSR.DR bit

Note: The read access size is fixed at 32 bits.

7.3.2.2.27 Simple-LIN (SCIX) Flag Clear Register (RSCIm_XFCLR)

Access Size : 8, 16, 32 bits

Address : <RSCIm_base> + 0078h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AEDC	COFC	PIBDC	CF1MC	CF0MC	BFDC	BCDC	BFOC	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All0	W	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15	AEDC	0h	W	Setting this bit to 1b clears the XSR0.AEDF bit and cancels holding the XSR1 register. The read value is always 0b.
14	COFC	0h	W	Setting this bit to 1b clears the XSR0.COFbit. The read value is always 0b.
13	PIBDC	0h	W	Setting this bit to 1b clears the XSR0.PIBDFbit. The read value is always 0b.
12	CF1MC	0h	W	Setting this bit to 1b clears the XSR0.CF1MF bit. The read value is always 0b.
11	CF0MC	0h	W	Setting this bit to 1b clears the XSR0.CF0MF bit. The read value is always 0b.
10	BFDC	0h	W	Setting this bit to 1b clears the XSR0.BFDFbit. The read value is always 0b.
9	BCDC	0h	W	Setting this bit to 1b clears the XSR0.BCDFbit. The read value is always 0b.
8	BFOC	0h	W	Setting this bit to 1b clears the XSR0.BFOF bit. The read value is always 0b.
7 to 0	-	All 0	W	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Note: The read access size is fixed at 32 bits.

7.3.3 Asynchronous Mode

Figure 7.3-3 shows the general format for asynchronous serial communications.

One frame consists of a start bit (low level), transmit/receive data, a parity bit, and stop bits (high level).

In asynchronous serial communications, the communications line is held in the mark state (high level) when not communicating.

The SCI monitors the communications line. When the SCI detects a start bit, it starts serial communication. The detection condition of the start bit changes according to the CCR3.RXDESEL setting. SCI regards space (Low level) as a start bit when CCR3.RXDESEL is 0b. SCI regards a fall edge as a start bit when RXDESEL is 1b.

Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communications. Both the transmitter and the receiver also have a double-buffered structure (it has also FIFO mode), so that data can be read or written during transmission or reception, enabling continuous data transmission and reception.

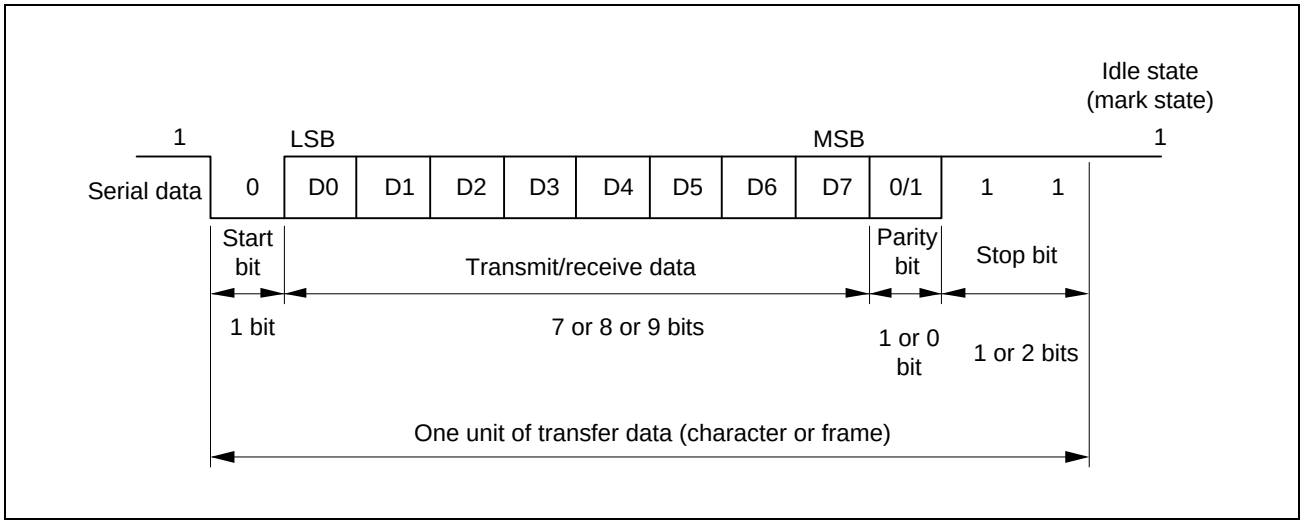


Figure 7.3-3 Data Format in Asynchronous Serial Communications (Example with 8-Bit Data, Parity, 2 Stop Bits)

7.3.3.1 Serial Data Transfer Format

Table 7.3-24 lists the serial data transfer formats that can be used in asynchronous mode. Any of 18 transfer formats can be selected according to the CCR1 and CCR3 settings.

For details of multi-processor function, see **7.3.4 Multi-Processor Communication Function**.

Table 7.3-24 Serial Transfer Formats (Asynchronous Mode)

CCR3				CCR1	Serial Transfer Format and Frame Length
CHR[1]	CHR[0]	MP	STP	PE	1 2 3 4 5 6 7 8 9 10 11 12 13
0	0	0	0	0	St 9-bit data STP
0	0	0	1	0	St 9-bit data STP STP
0	0	0	0	1	St 9-bit data P STP
0	0	0	1	1	St 9-bit data P STP STP
1	0	0	0	0	St 8-bit data STP
1	0	0	1	0	St 8-bit data STP STP
1	0	0	0	1	St 8-bit data P STP
1	0	0	1	1	St 8-bit data P STP STP
1	1	0	0	0	St 7-bit data STP
1	1	0	1	0	St 7-bit data STP STP
1	1	0	0	1	St 7-bit data P STP
1	1	0	1	1	St 7-bit data P STP STP
0	0	1	0	—	St 9-bit data MPB STP
0	0	1	1	—	St 9-bit data MPB STP STP
1	0	1	0	—	St 8-bit data MPB STP
1	0	1	1	—	St 8-bit data MPB STP STP
1	1	1	0	—	St 7-bit data MPB STP
1	1	1	1	—	St 7-bit data MPB STP STP

Note: St: Start bit
STP: Stop bit
P: Parity bit
MPB: Multi-processor bit

7.3.3.2 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCI operates on a base clock with a frequency of 16 times*1 the bit rate. In reception, the SCI samples the falling edge of the start bit using the base clock and performs internal synchronization*2. When sampling timing does not adjust (“CCR4.ASEN = 0b” or ”CCR4.ASEN = 1b and CCR4.AST[2:0] = 000b”), receive data is sampled at the rising edge of the 8th pulse*1 of the base clock, data is latched at the middle of each bit, as shown in **Figure 7.3-4**. Thus, the reception margin in asynchronous mode is determined by formula (1) below.

$$M = \left| \left(0.5 - \frac{1}{2N}\right) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100[\%] \cdots \text{Equation (1)}$$

M: Reception margin
N: Ratio of bit rate to clock

(N = 16, when CCR2.ABCSE = 0b and CCR2.ABCS = 0b, N = 8, when CCR2.ABCS = 1b, N = 6, when CCR2.ABCSE = 1b)

D: Duty cycle of clock (D = 0.5 to 1.0)
L: Frame length (L = 9 to 13)
F: Absolute value of clock frequency deviation

Assuming values of N = 16, F = 0 and D = 0.5 in formula (1), the reception margin is determined by the formula below.

$$M = \left\{\frac{0.5 - 1}{2 \times 16}\right\} \times 100(\%) = 46.875\%$$

However, this is only the computed value, and a margin of 20% to 30% should be allowed in system design.

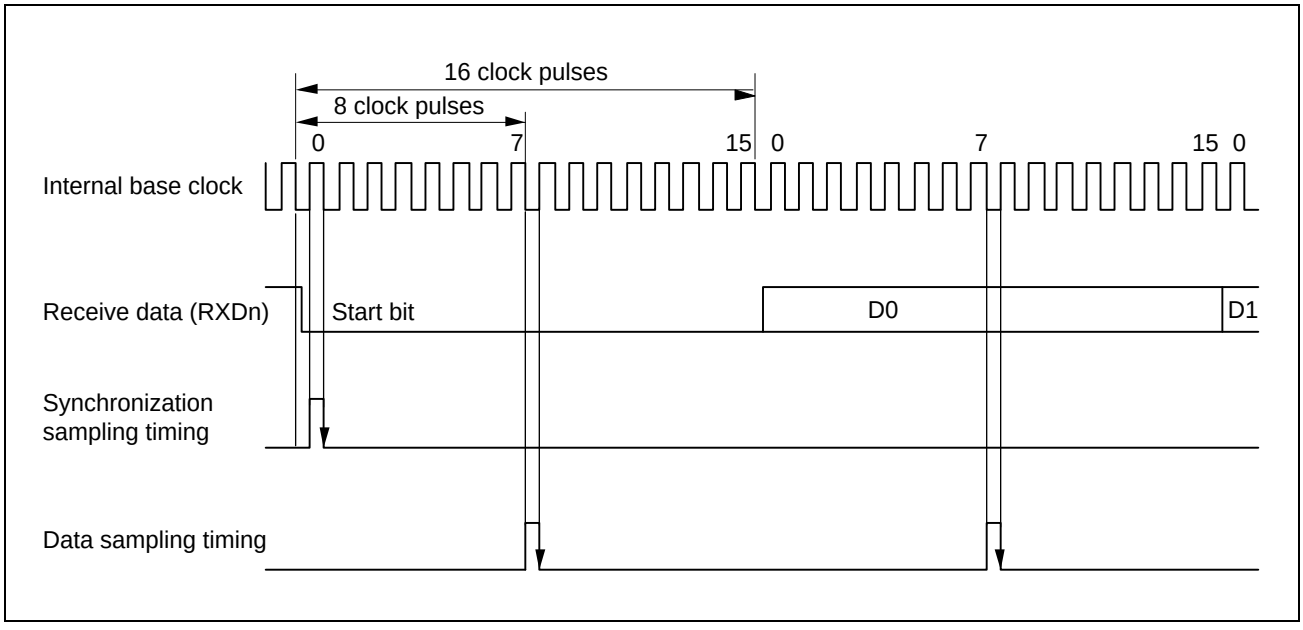


Figure 7.3-4 Receive Data Sampling Timing in Asynchronous Mode

Note 1. This is an example when CCR2.ABCS = 0b and CCR2.ABCSE = 0b. When CCR2.ABCS = 1b and CCR2.ABCS = 0b, a frequency of 8 times the bit rate is used as a base clock and receive data is sampled at

the rising edge of the 4th pulse of the base clock. When CCR2.ABCSE is 1b, a sextuple frequency of a bit rate is a base clock and receive data is sampled at the rising edge of the 3rd pulse of the base clock.

Note 2. The determination condition of the start bit is as follows.
In the case of the function of adjust sampling timing is OFF (CCR4.ASEN = 0b):
The determination condition of a start bit is that Low beyond half bit length continues. It is same as the sampling timing.
In **Figure 7.3-4**, the low period should be kept over 8 cycles to detect a start bit. If Low period does not keep over 8 cycles, the module judges this as a noise. So, the module does not start reception and wait start bit.
In the case of the function of adjust sampling timing is ON (CCR4.ASEN = 1b):
The determination condition of a start bit is that Low keeps up until the sampling timing.
Adjusting the sampling timing forward (CCR4.AJD = 1b) increases the possibility of erroneously determining a noise as the start bit.

7.3.3.3 Clock

Either an internal clock generated by the baud rate generator or an external clock input to the SCKm pin can be selected as the SCI's transfer clock, according to the setting of CCR3.CKE[1:0]. When an external clock is input to the SCKm pin, the clock frequency should be 16 times the bit rate (when CCR2.ABCS = 0b) and 8 times the bit rate (when CCR2.ABCS = 1b).

When the SCI is operated on an internal clock, the clock can be output from the SCKm pin. The frequency of the clock output in this case is equal to the bit rate, and the phase is such that the rising edge of the clock is in the middle of the transmit data, as shown in **Figure 7.3-5**.

If you selected an internal clock, the SCK pin is outputted after CCR0.TE is set to 1b or CCR0.RE is set to 1b.

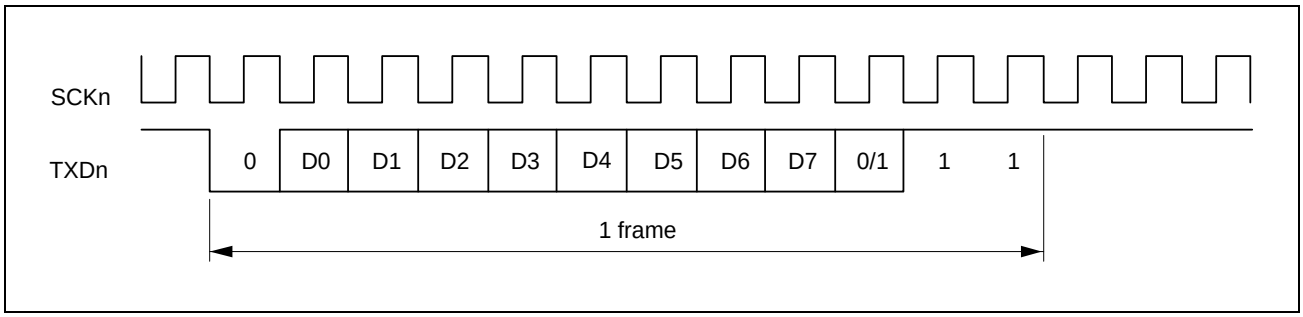


Figure 7.3-5 Phase Relationship between Output Clock and Transmit Data (Asynchronous Mode: CHR[1:0] = 10b, PE = 1b, MP = 0b, STP = 1b)

7.3.3.4 Double-Speed Operation and Frequency of 6 Times the Bit Rate

When CCR2.ABCS is set to 1b, the SCI operates on the bit rate twice that in the case where ABCS is set to 0b. And when CCR2.BGDM is set to 1b, the cycle of the base clock is halved and the bit rate is doubled from that in the case where BGDM is set to 0b. When CCR3.CKE[1] is set to 0b and the baud rate generator is selected, setting the ABCS and BGDM bits to 1b allows the SCI to operate on a bit rate four times that in the case ABCS = 0b and BGDM = 0b.

When CCR2.ABCSE is set to 1b, the number of base clock pulses are 6 during a period of 1 bit, and the base clock frequency is half. And SCI works 16/3 times of bit rate compared with a case of CCR2.ABCS = 0b, CCR2.BGDM = 0b and CCR2.ABCSE = 0b.

As shown by Formula (1) in **7.3.3.2 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode**, the reception margin decreases when CCR2.ABCS is set to 1b or CCR2.ABCSE is set to 1b.

Therefore, if the desired bit rate can be obtained with CCR2.ABCS set to 0b and/or CCR2.ABCSE set to 0b, it is recommended to use the SCI with CCR2.ABCS set to 0b and/or CCR2.ABCSE set to 0b.

7.3.3.5 CTS and RTS Function

The CTS function is the transmission control function by the CTSm# pin. Setting the CCR1.CTSE bit to 1b enables the CTS function. For the functions of CTS and RTS, you can select the alternate setting that uses either function with one terminal or the dedicated setting that uses each function independently with two terminals. This setting is done with the CCR1.CTSPEN bit.

When the CTS function is enabled, placing the low level on the CTSm# pin, causes transmission to start.

Even if the CTSm# pin goes high after transmission starts, the frame being transmitted is not affected and transmission will continue.

The RTS function is the transmission request function by the RTSn# pin. In the RTS function, the RTSm# pin output low level, when reception becomes possible. Conditions for output of the low and high levels are shown below.

[Conditions of low-level output]

(1) Non-FIFO selected when all of the following conditions are satisfied.

- CCR0.RE is 1b
- When in receivable state
- There are no received data yet to be read and not receiving.
- CSR.ORER, FER, PER flags are all 0b

(2) FIFO selected when all of the following conditions are satisfied.

- CCR0.RE is 1b
- When the quantity of receive data written in receive FIFO (RDR) are less than the setting value of FCR.RSTRG[4:0]
- CSR.ORER (RDR.ORER) is 0b

[Condition for high-level output]

When the conditions of low-level output have not been satisfied.

7.3.3.6 Address Match (Receive Data Match Detection) Function

The address match function can be used only for the asynchronous mode.

If CCR0.DCME is set to 1b,*2 when one frame of data has been received, SCI compares that receive data with the data which is set to CCR4.CMPD. If SCI detects the match to the comparison data (CCR4.CMPD*1) with receive data, SCI can issue RXI interrupt request.

If CCR3.MP bit is set to 0b, this comparative target in communication data is valid only data field in receive format. In multi- processor mode (CCR3.MP = 1b), if CCR0.IDSEL bit is set to 1b, the reception data at which MPB bit is 1b detects an address match or mismatch, and the reception data at which MPB bit is 0b detects always a mismatch. If CCR0.IDSEL bit is set to 0b, SCI detects an address match or mismatch despite the value of the MPB bit of the reception data at every reception complete.

Until SCI detects the match to the comparison data (CCR4.CMPD) with receive data, the communication data is skipped (discarded), and SCI cannot detect parity error, framing error.

When SCI detects the match, the CCR0.DCME is automatically cleared, and CSR.DCMF is set to 1b. If CCR0.IDSEL bit is set to 1b at this time, CCR0.MPIE bit is automatically cleared. If CCR0.IDSEL bit is set to 0b at this time, CCR0.MPIE bit is kept. At the same time, if CCR0.RIE is set to 1b, SCI issues RXI interrupt request.

If SCI detects framing error in comparative receive data which is detected the match, CSR.DFER is set to 1b, and if SCI detects parity error in that frame, CSR.DPER is set to 1b. That comparative receive data and MPB bit are not stored to RDR register, and CSR.RDRF is retained to 0b.

After SCI detects the match, and the CCR0.DCME is automatically cleared, it receives next data continuously in current register setting.

When CSR.DFER flag or CSR.DPER flag is set, the address much is not detected. Before making the address match function effective, please be sure to set CSR.DFER and CSR.DPER flag as 0b.

Figure 7.3-6 and **Figure 7.3-7** show the address match function example.

Note 1. This comparative target can select one length of 3 types, they are CMPD[6:0] with 7-bit length enable, CMPD[7:0] with 8-bit, and CMPD[8:0] with 9-bit length.

Note 2. Set the CCR0.DCME bit to 1b before receiving the start bit of the received frame that performs address matching.

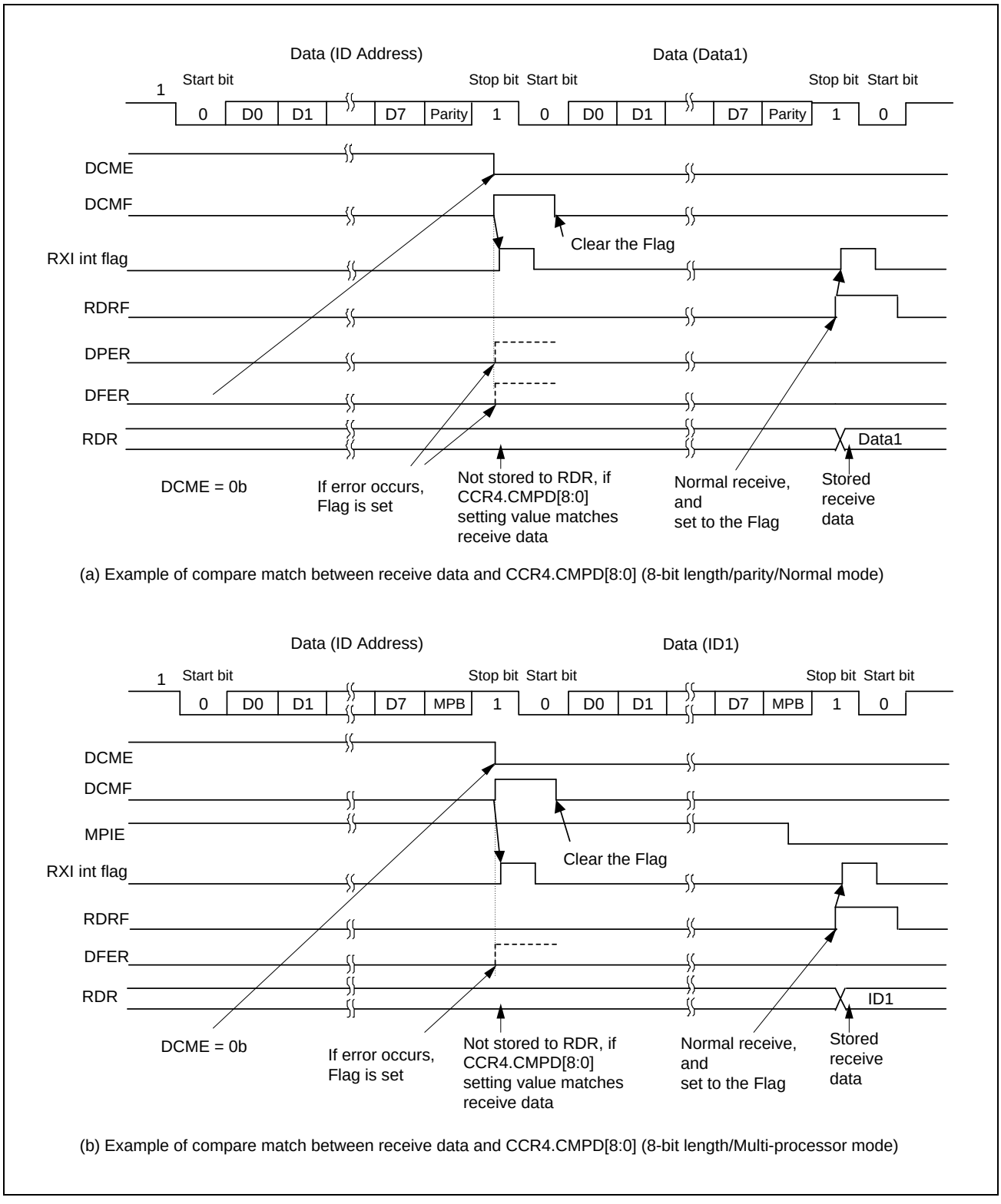
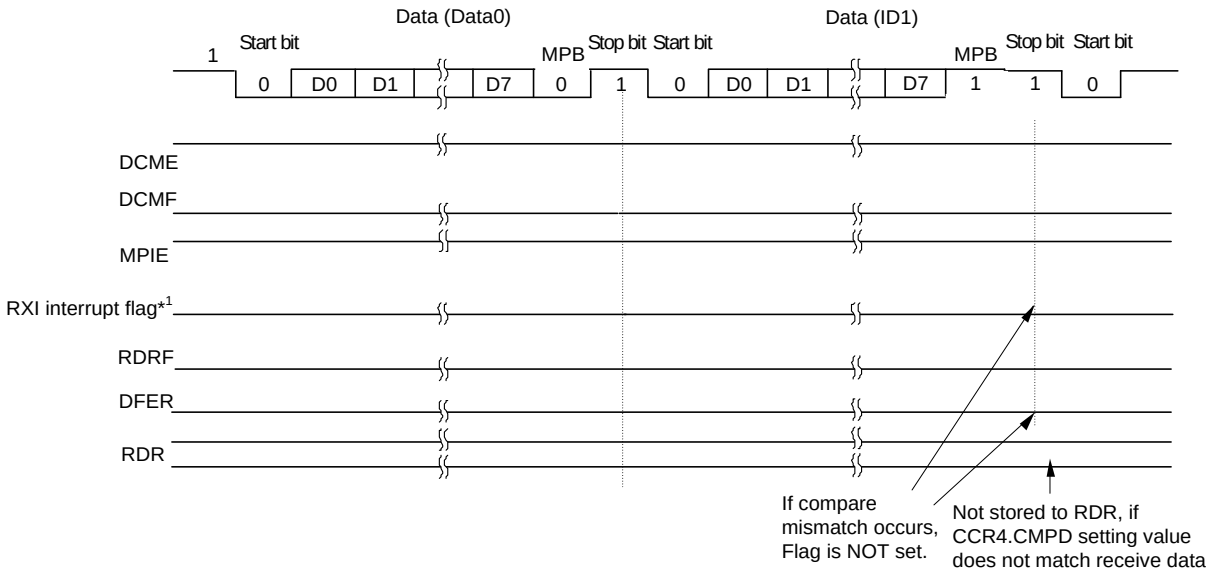
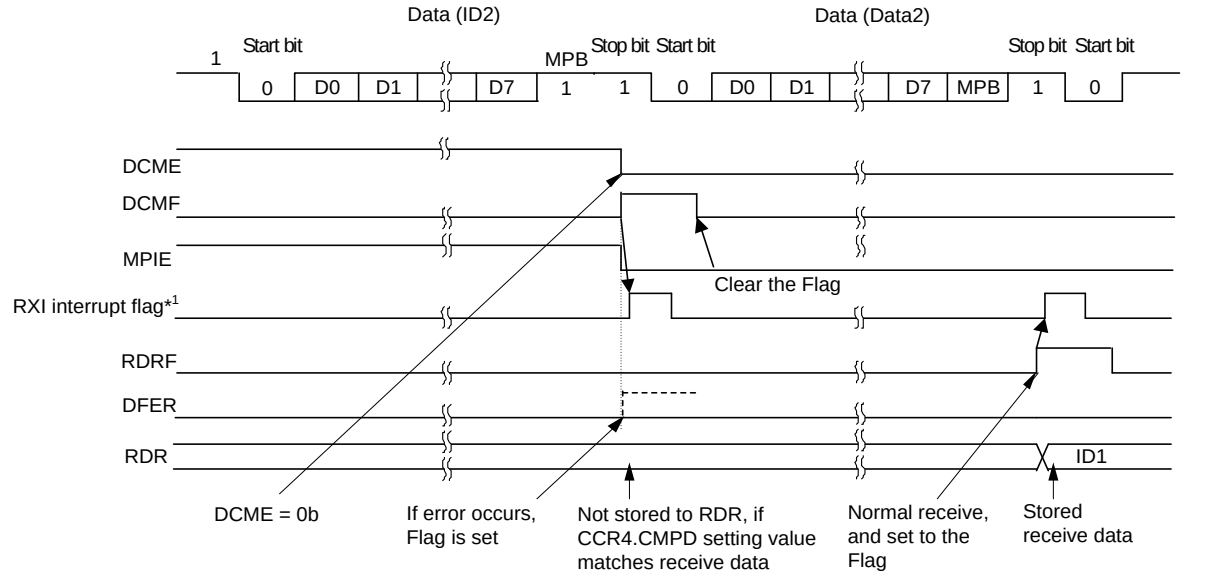


Figure 7.3-6 Example of Address Match (1) 8-Bit Data



(a) Example of compare mismatch between receive data and CCR4.CMPD (8-bit length / IDSEL = 1 / Multi-processor mode)



(b) Example of compare match between receive data and CCR4.CMPD (8-bit length / IDSEL = 1 / Multi-processor mode)

Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-7 Example of Address Match (2) Multi-Processor Mode/8-Bit Data

7.3.3.7 SCI Initialization (Asynchronous Mode)

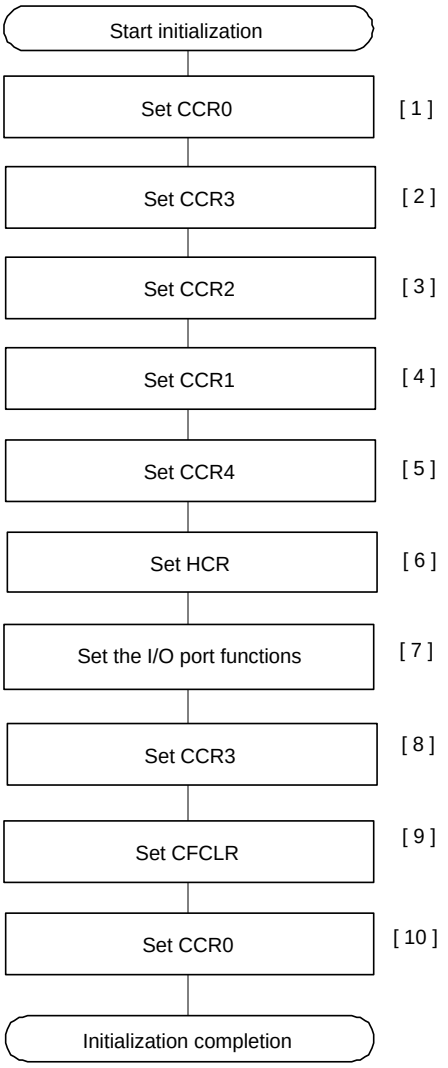
Before transmitting and receiving data, start by writing 0b to CCR0.TE and CCR0.RE (or writing the initial value to CCR0) and then continue through the Procedure (select to Non-FIFO or FIFO) for SCI given in **Figure 7.3-8** or **Figure 7.3-9**. Whenever the operating mode or transfer format is changed, CCR0.TE and CCR0.RE must be initialized before the change is made.

When the external clock is used in asynchronous mode, ensure that the clock signal is supplied even during initialization.

Note that setting the CCR0.RE bit to 0b initializes neither the ORER, FER, PER, RDRF, DR and RDAT. When FIFO selected, even if the TE bit is set as 0b, the TEND flag is not initialized, so please be careful.

Please be also careful at the time of change in the operation mode.

Moreover, note that switching the value of the CCR0.TE bit from 0b to 1b while the CCR0.TIE bit is 1b leads to the generation of a TXI interrupt request.



[1] Set the following bits of CCR0 to 0.
TEIE, TIE, RIE, TE, RE
If you have not changed from the initial settings, you can skip this step.

[2] Set up following function and communication mode.
Driver control function for RS-485
FIFO no-use
Multi-Processor mode
Communication mode (MOD[2:0] = 000b)
Transmission / reception format
Clock enable (Leave the initial value in this step when outputting the clock)
Leave unused bits at their initial values.

[3] Set up the bit-rate-modulation function*1,*2, select clock, set bit rate*2.

[4] Set up the Noise-filter function, the loop-back function, communication terminal status, the parity check function, and the CTS/RTS function.

[5] Set up the adjust sampling timing function and the adjust transmit timing function.

[6] When using the half data communication function, enable or set the function. If this function is not used, leave the default value.

[7] Make I/O port settings to enable input and output functions as required for TXDn, RXDn, and SCKn pins.

[8] Set clock enable bit (CKE[1:0]) at this step when outputting the clock. After this register setting, a clock will be output immediately.

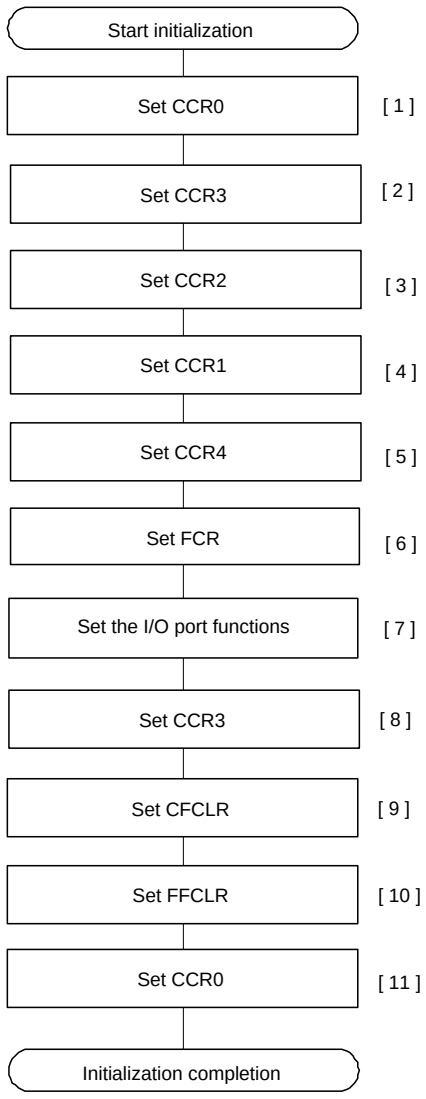
[9] Write 1 to the following bits and clear the corresponding flag.
CFCLR.RDRFC, FERC, PERC, MFFC, ORERC, DFERC, DPERC, DCMFC, ERSC
If it's initialization flow after a reset, you can skip this step.

[10] Set the TE or RE bit to 1.
To enable interrupts, set the TE bit and TIE bit, and the RE bit and RIE bit to 1 with one instruction at the same time.
Setting the TE and RE bits allows TXDn and RXDn to be used.

Note 1. If you do not use the bit rate modulation function, you do not need to set it.

Note 2. If you use an external clock, you do not need to set it.

Figure 7.3-8 Sample SCI Initialization Flowchart (Asynchronous Mode/Non-FIFO Selected)



[1] Set the following bits of CCR0 to 0.
TEIE, TIE, RIE, TE, RE
If you have not changed from the initial settings, you can skip this step.

[2] Set up following function and communication mode.
Driver control function for RS-485
FIFO use
Multi-Processor mode
Communication mode (MOD[2:0] = 000b)
Transmission / reception format
Clock enable (Leave the initial value in this step when outputting the clock)
Leave unused bits at their initial values.

[3] Set up the bit-rate-modulation function*1,*2, select clock, set bit rate*2.

[4] Set up the Noise-filter function, the loop-back function, communication terminal status, the parity check function, and the CTS/RTS function.

[5] Set up the adjust sampling timing function and the adjust transmit timing function.

[6] Set the TFRST and RFRST to 1 to empty FIFO.
Set the DRES, TTRG[4:0], RTRG[4:0], and RSTRG[4:0] bits

[7] Make I/O port settings to enable input and output functions as required for TXDn, RXDn, and SCKn pins.

[8] Set clock enable bit (CKE[1:0]) at this step when outputting the clock. After this register setting, a clock will be output immediately.

[9] Write 1 to the following bits and clear the corresponding flag.
CFCLR.RDRFC, FERC, PERC, MFFC, ORERC, DFERC, DPERC, DCMFC, ERSC
If it's initialization flow after a reset, you can skip this step.

[10] Write 1 to the FFCLR.DRC and clear the corresponding flag.
If it's initialization flow after a reset, you can skip this step.

[11] Set the TE or RE bit to 1.
To enable interrupts, set the TE bit and TIE bit, and the RE bit and RIE bit to 1 with one instruction at the same time.
Setting the TE and RE bits allows TXDn and RXDn to be used.

Note 1. If you do not use the bit rate modulation function, you do not need to set it.

Note 2. If you use an external clock, you do not need to set it.

Figure 7.3-9 Sample SCI Initialization Flowchart (Asynchronous Mode/FIFO Selected)

Figure 7.3-10 shows an example of the timing when data is transmitted after reset is released, and SCI is set to asynchronous mode according to **Figure 7.3-8** or **Figure 7.3-9**. As shown in the figure, when the pin function is set to the TXD pin, the CCR0.TE bit is 0b, so the pin is high impedance. When transmit data is written after setting the CCR0.TE bit to 1b, data transmission starts. There is a transmission wait time from writing TDR to data transmission starts. In asynchronous mode, TXD is high during this period.

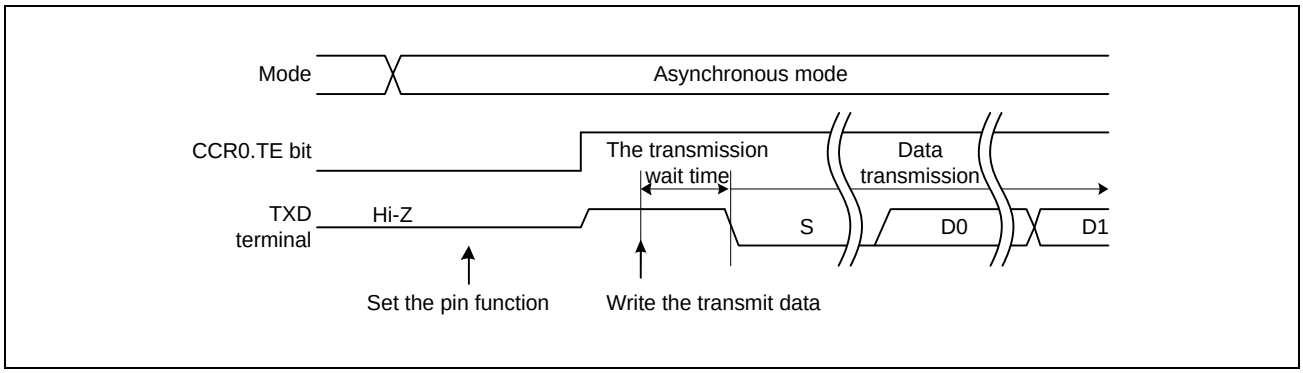


Figure 7.3-10 Data Transmission Timing Example in Asynchronous Mode

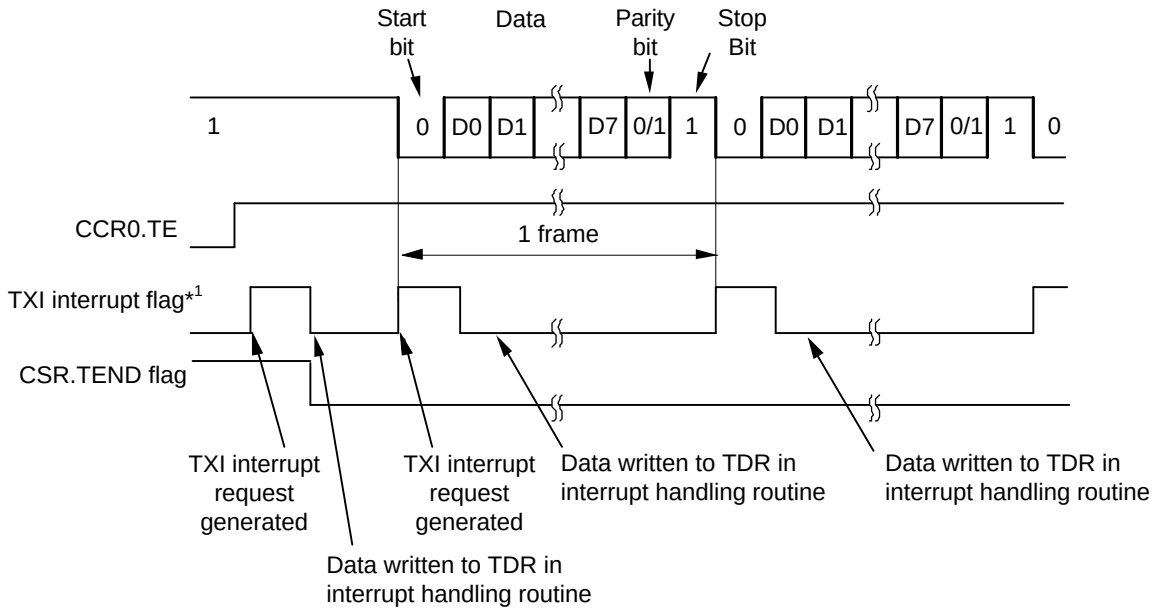
7.3.3.8 Serial Data Transmission (Asynchronous Mode)

(1) Non-FIFO selected

Figure 7.3-11, **Figure 7.3-12**, and **Figure 7.3-13** show an example of the operation for serial transmission in asynchronous mode. In serial transmission, the SCI operates as described below.

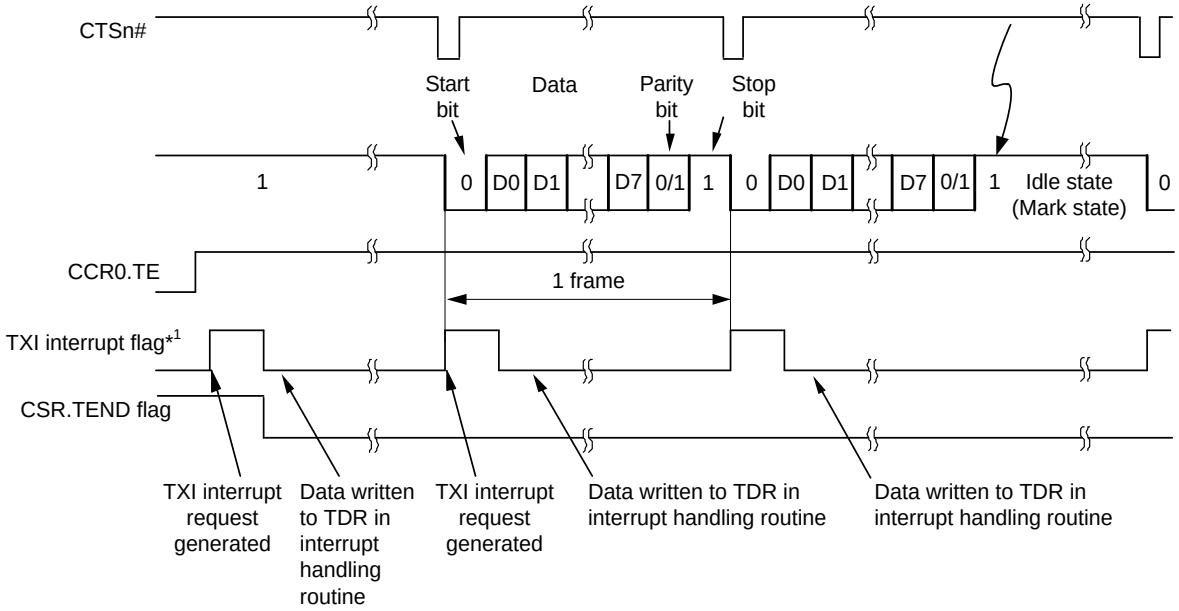
1. The SCI transfers data from TDR to TSR when data is written to TDR in the TXI interrupt handling routine. At the beginning of transmission, set 1b to CCR0.TE and CCR0.TIE simultaneously. Then the TXI interrupt request is generated.
2. Transmission starts at data transfer from TDR to TSR when CCR1.CTSE = 0b (CTS function is disabled) or when the CTS# pin level is low. If CCR0.TIE is 1b at this time, a TXI interrupt request is generated. Continuous transmission is obtainable by writing the next transmit data to TDR in the TXI interrupt handling routine before transmission of the current transmit data is completed. Write the last transmission data, then the last data transmit start, and TXI is generated. When TEI interrupt requests are in use, before the last data transmit end, set the CCR0.TIE bit to 0b (a TXI interrupt request is disabled) and the CCR0.TEIE bit to 1b (a TEI interrupt request is enabled) using the TXI handling routine.
3. Data is sent from the TXDn pin in the following order: start bit, transmit data, parity bit or multi-processor bit (may be omitted depending on the format), and stop bit.
4. The SCI checks for updating of (writing to) TDR at the time of stop bit output.
5. When TDR is updated, setting of CCR1.CTSE to 0b (CTS function is disabled) or a low level input on the CTSn# pin cause the next transfer of the next transmit data from TDR to TSR and sending of the stop bit, after which serial transmission of the next frame starts.
6. If TDR is not updated, CSR.TEND is set to 1b, the stop bit is sent, and then the mark state is entered in which 1 is output. If the CCR0.TEIE is 1b at this time, a TEI interrupt request is generated.

Figure 7.3-15 shows the example of Serial Transmission Flowchart.



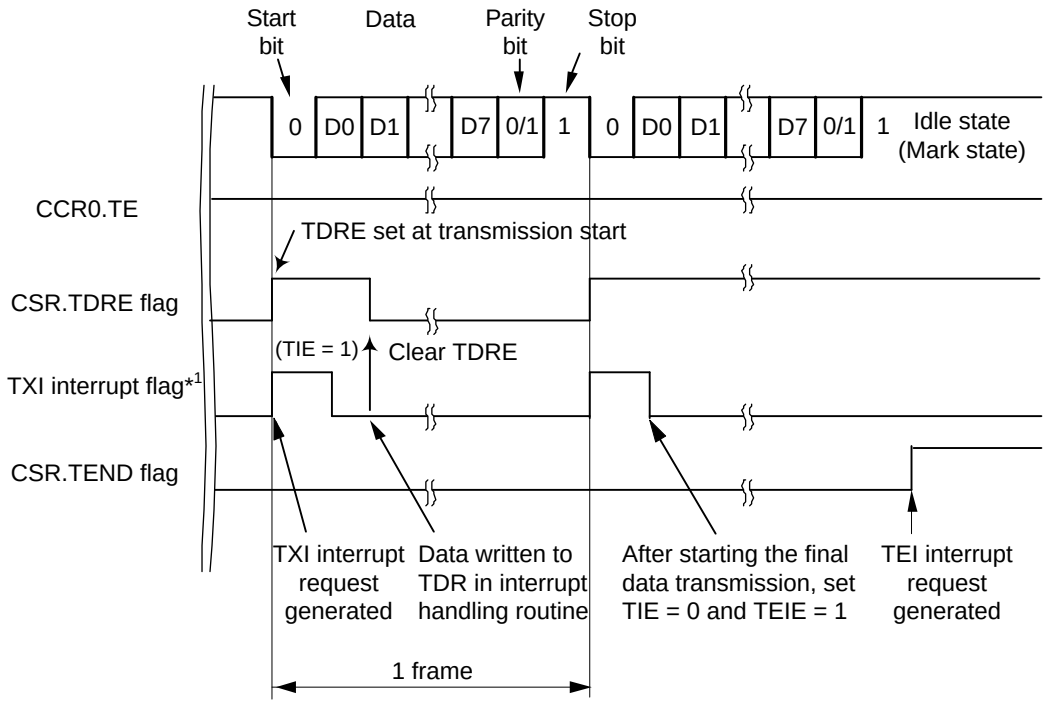
Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-11 Example of Operation for Serial Transmission in Asynchronous Mode (1) (with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Not Used, at the Beginning of Transmission)



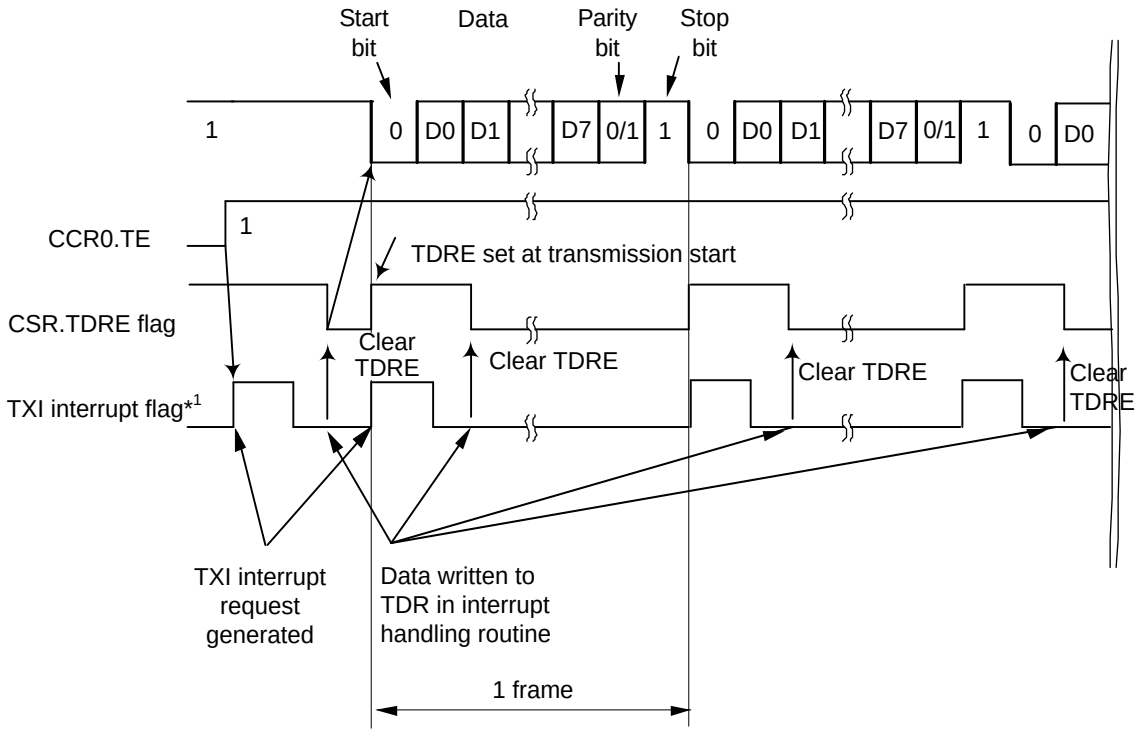
Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-12 Example of Operation for Serial Transmission in Asynchronous Mode (2) (with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Used, at the Beginning of Transmission)



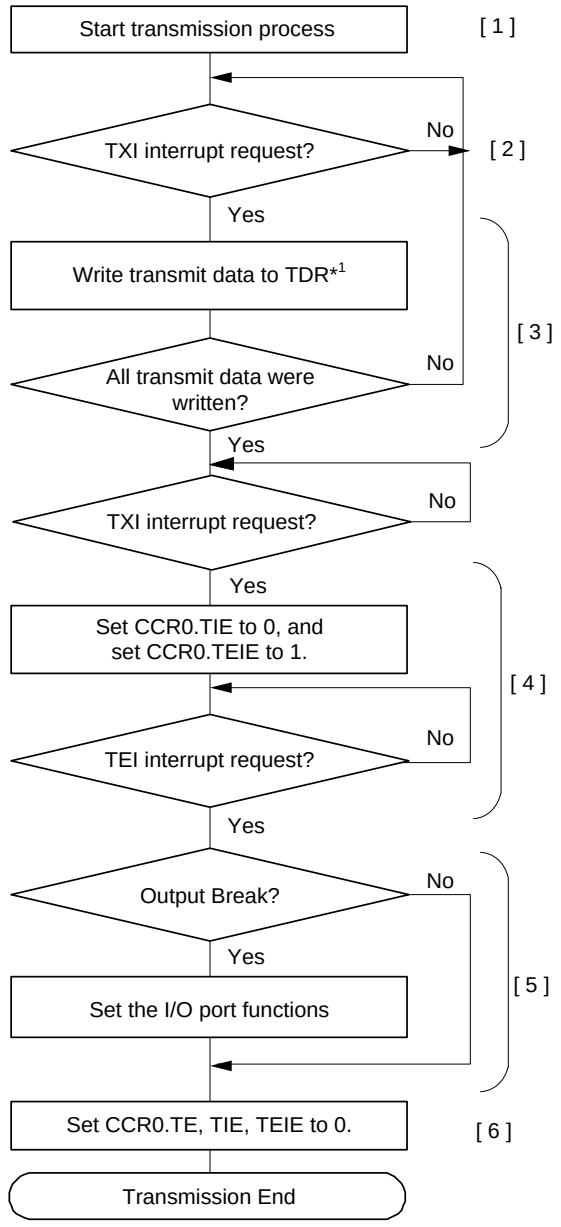
Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-13 Example of Operation for Serial Transmission in Asynchronous Mode (3) (with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Not Used, from the Middle of Transmission until Transmission Completion)



Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-14 Example of Operation for Serial Transmission in Asynchronous Mode (4) (with 8-Bit Data, Parity, 1 Stop Bit, CTS Function Not Used, in the Middle of Transmission)



[1] Start transmission process

[2] TXI interrupt request:
The transmit-data-empty-interrupt (TXI) is generated at following conditions.
When set TE and TIE to 1 at same time.
When transmission-data forwards to TSR from TDR.

[3] Serial transmission continuation procedure:
To continue serial transmission write transmit data to TDR*1 once by using a TXI.
Transmit data can also be written by activating the DMAC.

[4] If you use TEI, set TIE = 0 and TEIE = 1 after writing the last-transmit-data to TDR*1 and transmission is started.

[5] Break output at the end of serial transmission:
To output a break, it is necessary to set SPB2IO and SPB2DT, for output status setting when TE is 0.
(If you set these bits at initialization, you can skip this step.)

[6] Set CCR0.TE, TIE, TEIE to 0.

Note 1. In case of byte access with data length 9 bits, write transmit data in order of TDR[15:8] and TDR[7:0].

Figure 7.3-15 Example of Serial Transmission Flowchart in Asynchronous Mode (Non-FIFO Selected)

(2) FIFO selected

Figure 7.3-16 shows an example of data format that is written to transmit FIFO (TDR) in asynchronous mode with FIFO selected. MPBT write to transmit FIFO (TDR) bit 9. Data is set to TDR.TDAT[8:0] corresponded to data length. It should write to 0b for unused bits. It should write it in order of the TDR[15:8] register and the TDR[7:0] register at byte-access.

Data Length	Register Setting CCR3.CHR[1:0]		Transmit data in TDR[15:0]															
			b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
7 bits	1	1	—	—	—	—	—	—	MPBT	—	—	TDAT[6:0]						
8 bits	1	0	—	—	—	—	—	—	MPBT	—	TDAT[7:0]							
9 bits	0	x	—	—	—	—	—	—	MPBT	TDAT[8:0]								

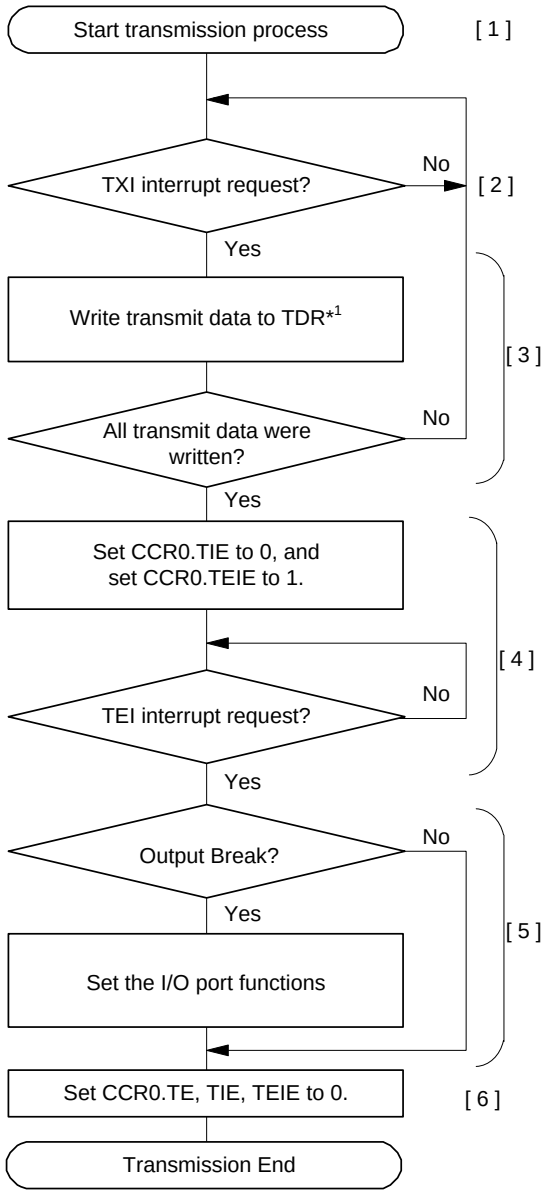
Note: —: Do not used. It should write to 0b

Figure 7.3-16 Data Format that is Written to Transmit FIFO (TDR) (FIFO Selected)

In serial transmission, the SCI operates as described below.

1. The SCI transfers data from TDR to TSR when data is written to TDR in the TXI interrupt handling routine. The writable transmit data number is until (NFIFO (= 32) − FTSR.T) bytes. At the beginning of transmission, set 1b to CCR0.TE and CCR0.TIE simultaneously. Then the TXI interrupt request is generated.
2. Transmission starts at data transfer from TDR to TSR when CCR1.CTSE = 0b (CTS function is disabled) or when the CTS# pin level is low. When the quantity of transmit data written in transmit FIFO (TDR) is equal to or less than the specified transmit triggering number, CSR.TDRE is set to 1b. If CCR0.TIE is 1b at this time, a TXI interrupt request is generated. Continuous transmission is obtainable by writing the next transmit data to TDR in the TXI interrupt handling routine before transmission of the current transmit data is completed. Write the last transmission data, then the last data transmit start, and TXI is generated. When TEI interrupt requests are in use, before the last data transmit end, set the CCR0.TIE bit to 0b (a TXI interrupt request is disabled) and the CCR0.TEIE bit to 1b (a TEI interrupt request is enabled) using the TXI handling routine.
3. Data is sent from the TXDn pin in the following order: start bit, transmit data, parity bit or multi-processor bit (may be omitted depending on the format), and stop bit.
4. The SCI checks whether non-transmitted data in transmit FIFO (TDR) or not at the time of stop bit output.
5. When data is set to transmit FIFO (TDR), setting of CCR1.CTSE to 0b (CTS function is disabled) or a low level input on the CTSn# pin cause the next transfer of the next transmit data from transmit FIFO (TDR) to TSR and sending of the stop bit, after which serial transmission of the next frame starts.
6. If data is not set to transmit FIFO (TDR), CSR.TEND is set to 1b, the stop bit is sent, and then the mark state is entered in which 1 is output. If CCR0.TEIE is 1b at this time, a TEI interrupt request is generated.

Figure 7.3-17 shows a sample flowchart for serial transmission in asynchronous mode at FIFO selected.



[1] Start transmission process:
Set transmit-data to the transmit FIFO (TDR):
The number of transmittable data that can be set is 32.

[2] TXI interrupt request:
The transmit-data-empty-interrupt (TXI) is generated when transmission-data forwards to TSR from the transmit FIFO (TDR) and the number of transmit-data written in the transmit FIFO is equal to or less than the transmit FIFO threshold.

[3] Serial transmission continuation procedure:
To continue serial transmission, write all transmit data to the transmit FIFO (TDR) by using a TXI interrupt and clear TDRE flag to 0. (Transmit data can also be written to the transmit FIFO (TDR) by activating the DMAC. In this case, the TDRE flag is cleared automatically, so please do not operate the TDRE flag.)
The number of transmission data it's possible to write in is N_{FIFO} (= 32) - (the number of stored transmit FIFO data).

[4] If you use TEI, set TIE = 0 and TEIE = 1 after writing the last-transmit-data to TDR.*1

[5] Break output at the end of serial transmission:
To output a break, it is necessary to set SPB2IO and SPB2DT, for output status setting when TE is 0.
(If you set these bits at initialization, you can skip this step.)

[6] Set CCR0.TE, TIE, TEIE to 0.

Note 1. In case of byte access with data length 9 bits, write transmit data in order of TDR[15:8] and TDR[7:0].

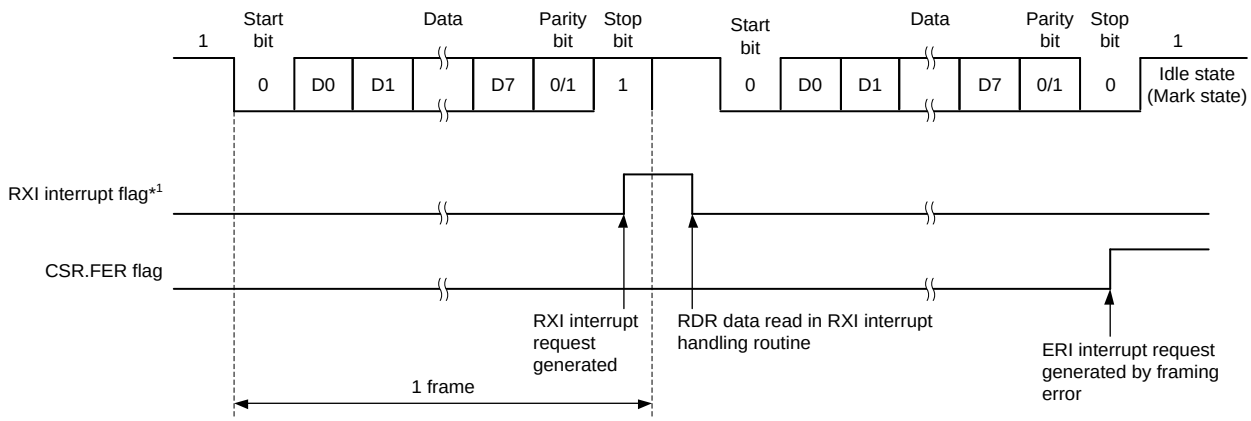
Figure 7.3-17 Example of Serial Transmission Flowchart in Asynchronous Mode (FIFO Selected)

7.3.3.9 Serial Data Reception (Asynchronous Mode)

(1) Non-FIFO selected

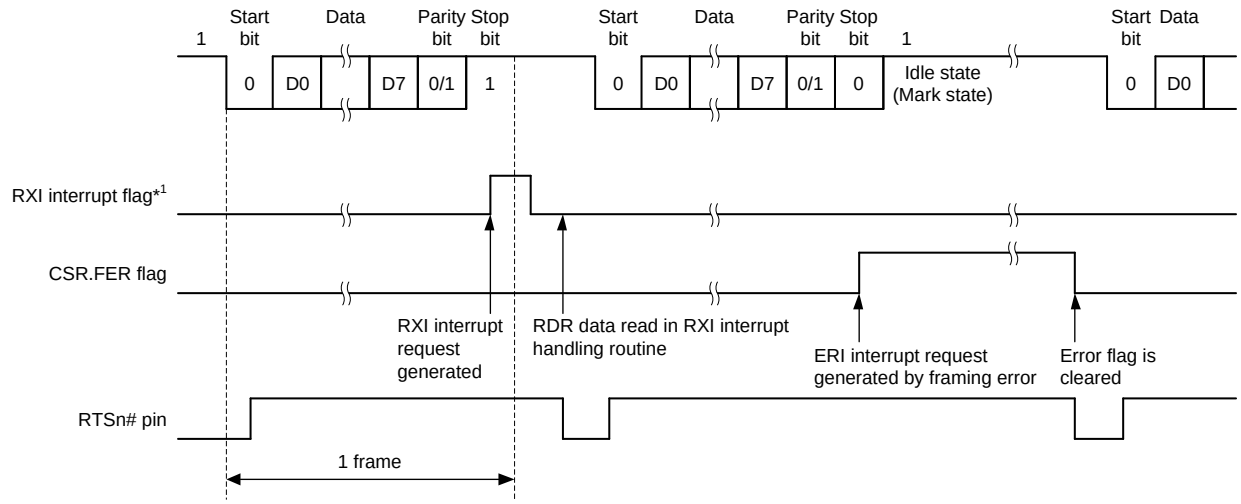
Figure 7.3-18 and **Figure 7.3-19** show an example of the operation for serial data reception in asynchronous mode. In serial data reception, the SCI operates as described below.

1. When the value of CCR0.RE becomes 1b, the output signal on the RTSn# pin goes to the low level in the case of RTS function use.
2. When the SCI monitors the communications line and detects a start bit, it performs internal synchronization, stores receive data in RSR, and checks the parity bit and stop bit.
3. If an overrun error occurs, CSR.ORER is set to 1b. If CCR0.RIE is 1b at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR.
4. If a parity error is detected, CSR.PER is set to 1b and receive data is transferred to RDR. If the CCR0.RIE is 1b at this time, an ERI interrupt request is generated.
5. If a framing error (when the stop bit is 0b) is detected, CSR.FER is set to 1b and receive data is transferred to RDR. If CCR0.RIE is 1b at this time, an ERI interrupt request is generated.
6. When reception finishes successfully, receive data is transferred to RDR. If CCR0.RIE is 1b at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to RDR in this RXI interrupt handling routine before reception of the next receive data is completed. Reading the received data that have been transferred to RDR causes the RTSn# pin to output the low level in the case of RTS function use. If you do not want to turn the RTSn# pin output low after receiving the last data, set CCR0.RE bit to 0b, before reading the RDR.



Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-18 Example of SCI Operation for Serial Reception in Asynchronous Mode (1) (Example with 8-Bit Data, Parity, 1 Stop Bit, RTS Function is Not Used)



Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-19 Example of SCI Operation for Serial Reception in Asynchronous Mode (2) (Example with 8-Bit Data, Parity, 1 Stop Bit, RTS Function is Used)

Table 7.3-25 lists the states of the flags in CSR status register and receive data handling when a receive error is detected. If a receive error is detected, an ERI interrupt request is generated but an RXI interrupt request is not generated. Data reception cannot be resumed while the receive error flag is 1. Accordingly, set the ORER, FER, and PER bits to 0b before resuming reception. Moreover, be sure to read the RDR during overrun error processing. When a reception is forcibly terminated by setting CCR0.RE bit to 0b during operation, read the RDR register because received data which has not yet been read may be left in RDR. **Figure 7.3-20** and **Figure 7.3-21** show samples of flowcharts for serial data reception.

Table 7.3-25 Flags in the CSR Status Register and Receive Data Handling

Flags in the CSR Status Register			Receive Data	Receive Error Type
ORER	FER	PER		
1	0	0	Lost	Overrun error
0	1	0	Transferred RDR	Framing error
0	0	1	Transferred RDR	Parity error
1	1	0	Lost	Overrun error + Framing error
1	0	1	Lost	Overrun error + Parity error
0	1	1	Transferred RDR	Framing error + Parity error
1	1	1	Lost	Overrun error + Framing error + Parity error

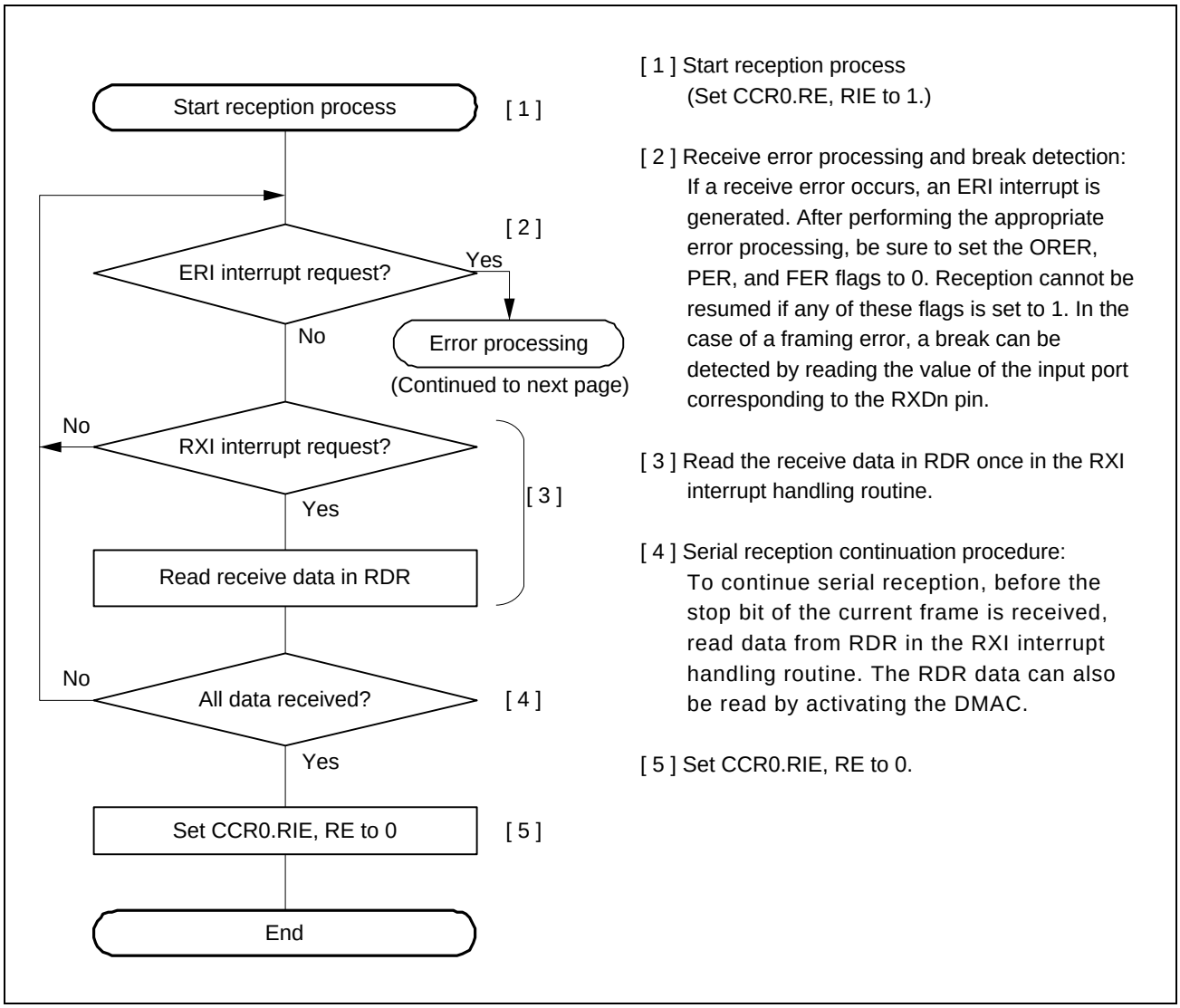


Figure 7.3-20 Example Flowchart of Serial Reception in Asynchronous Mode (Non-FIFO Selected) (1)

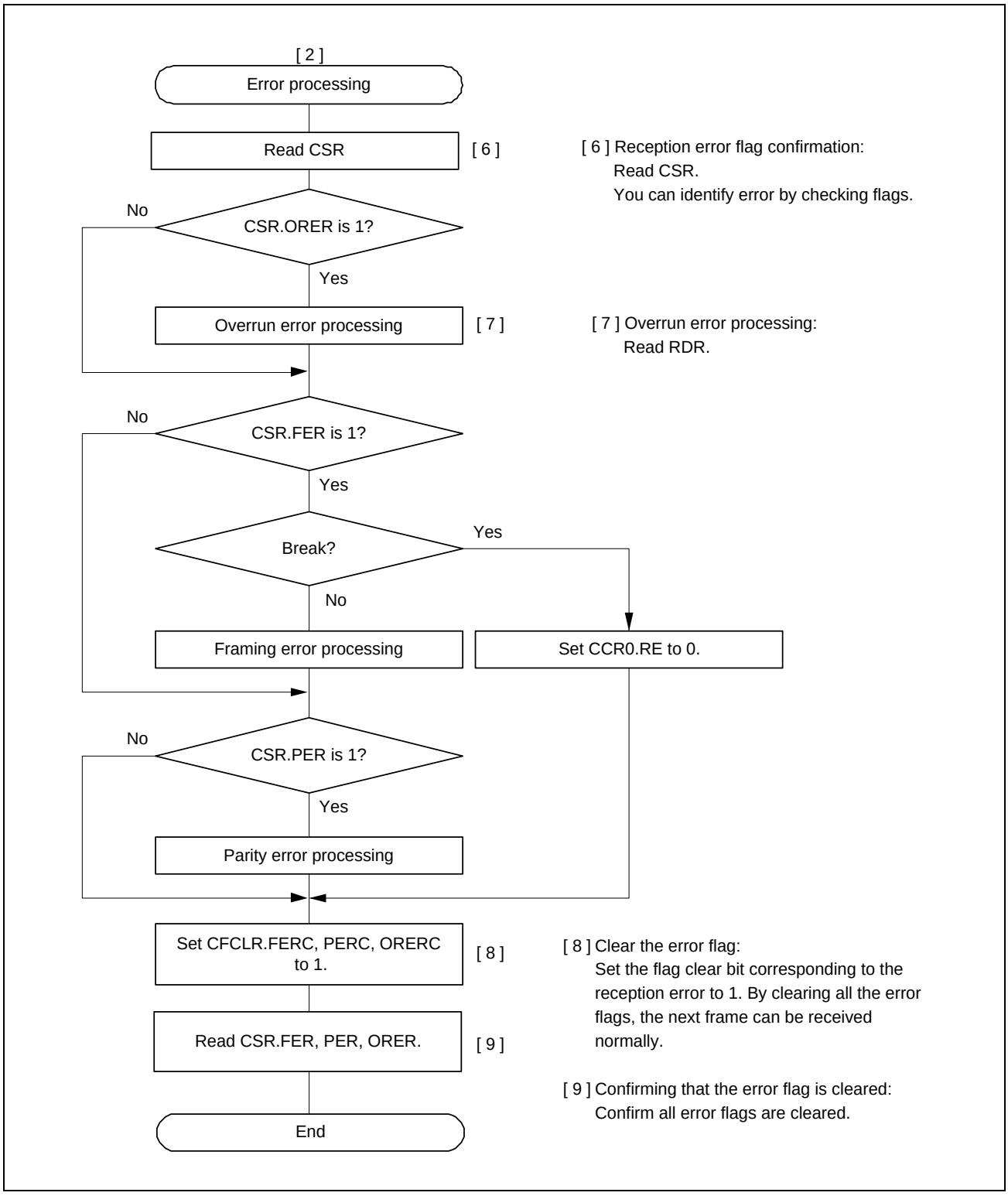


Figure 7.3-21 Example Flowchart of Serial Reception in Asynchronous Mode (Non-FIFO Selected) (2)

(2) FIFO selected

Figure 7.3-22 shows an example of data format that is stored to receive FIFO (RDR) in asynchronous mode.

MPB (Receive FIFO (RDR) bit 9) is stored 0. Data is stored to receive FIFO (RDR) corresponded to data length. It is stored 0 for unused bits. If reading receive FIFO (RDR), SCI updates to next data which are FER, PER and receive data

(RDAT[8:0]) in receive FIFO. The flags which are BRK, RDRF, ORER and DR in receive FIFO, are always indicated to the flags corresponded to CSR register.

Data Length	Register Setting CCR3.CHR[1:0]		Receive flag in RDR[31:0], MPB, RDAT[8:0] b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
7 bits	1	1	—	—	—	FFER	FPER	DR	MPB	0	0	RDAT[6:0]						
8 bits	1	0	—	—	—	FFER	FPER	DR	MPB	0	RDAT[7:0]							
9 bits	0	x	—	—	—	FFER	FPER	DR	MPB	RDAT[8:0]								
Data Length	Register Setting CCR3.CHR[1:0]		b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
7 bits	1	1	—	—	—	FER	PER	—	—	ORER	—	—	—	—	—	—	—	—
8 bits	1	0	—	—	—	FER	PER	—	—	ORER	—	—	—	—	—	—	—	—
9 bits	0	x	—	—	—	FER	PER	—	—	ORER	—	—	—	—	—	—	—	—

Note: The MPB flag, the RDR[9] bit, is always read as 0b.
When a 7-bit data length is selected, the RDAT[8:7] bits are read as 00b.
When an 8-bit data length is selected, the RDAT[8] bit is read as 0b.

Figure 7.3-22 Data Format that is Stored to Receive FIFO (RDR) (FIFO Selected)

Table 7.3-26 lists the states of the flags in CSR status register and receive data handling when a receive error is detected with FIFO selected. **Figure 7.3-23** and **Figure 7.3-24** show samples of flowcharts for serial data reception with FIFO selected.

In serial data reception, the SCI operates as described below.

1. When the value of CCR0.RE becomes 1b, the output signal on the RTSn# pin goes to the low level in the case of RTS function use.
2. When the SCI monitors the communications line and detects a start bit, it performs internal synchronization, stores receive data in RSR, and checks the parity bit and stop bit.
3. If there is no space in RDR, CSR.ORER is set to 1b. If CCR0.RIE is 1b at this time, an ERI interrupt request is generated. Receive data is not transferred to receive FIFO (RDR).
4. If a parity error is detected, CSR.PER is set to 1b and receive data is transferred to receive FIFO (RDR). If the CCR0.RIE is 1b at this time, an ERI interrupt request is generated.
5. If a framing error (when the stop bit is 0b) is detected, CSR.FER is set to 1b and receive data is transferred to RDR. If CCR0.RIE is 1b at this time, an ERI interrupt request is generated.
6. After framing error is detected, when SCI detects that continuous receive data is 0b for 1 frame, reception stops.
7. When quantity of data stored in the receive FIFO data register (RDR) falls the specified receive triggering number, and that no next data has been received yet after the elapse of 15 etu from the last stop bit in asynchronous mode, FRSR.DR is set to 1b. If RIE bit is set to 1b, SCI occurs RXI interrupt request when FCR.DRES bit is 0b and SCI occurs ERI interrupt request when FCR.DRES = 1b.
8. When reception finishes successfully, receive data is transferred to receive FIFO (RDR). RDRF is set to 1b when the quantity of receive data which is equal to or greater than the specified receive triggering number are stored in receive FIFO (RDR). If CCR0.RIE is 1b at this time, an RXI interrupt request is generated. Continuous reception

is enabled by reading the receive data transferred to receive FIFO (RDR) in this RXI interrupt handling routine before overrun error is occurred. Reading the received data that have been transferred to receive FIFO (RDR), and if it is less than RTS trigger number, causes the RTSn# pin to output the low level in the case of RTS function use.

Table 7.3-26 Flags in the CSR Status Register and Receive Data Handling (FIFO Selected)

CSR Value			Receive FIFO (RDR)	
ORER	FER*1	PER*1	RDAT[8:0]	Receive Error Type
1	0	0	Lost	Overrun error
0	1	0	Transferred RDR	Framing error
0	0	1	Transferred RDR	Parity error
1	1	0	Lost	Overrun error + Framing error
1	0	1	Lost	Overrun error + Parity error
0	1	1	Transferred RDR	Framing error + Parity error
1	1	1	Lost	Overrun error + Framing error + Parity error

Note 1. This flag indicates whether there is an error in received data when reception is completed.

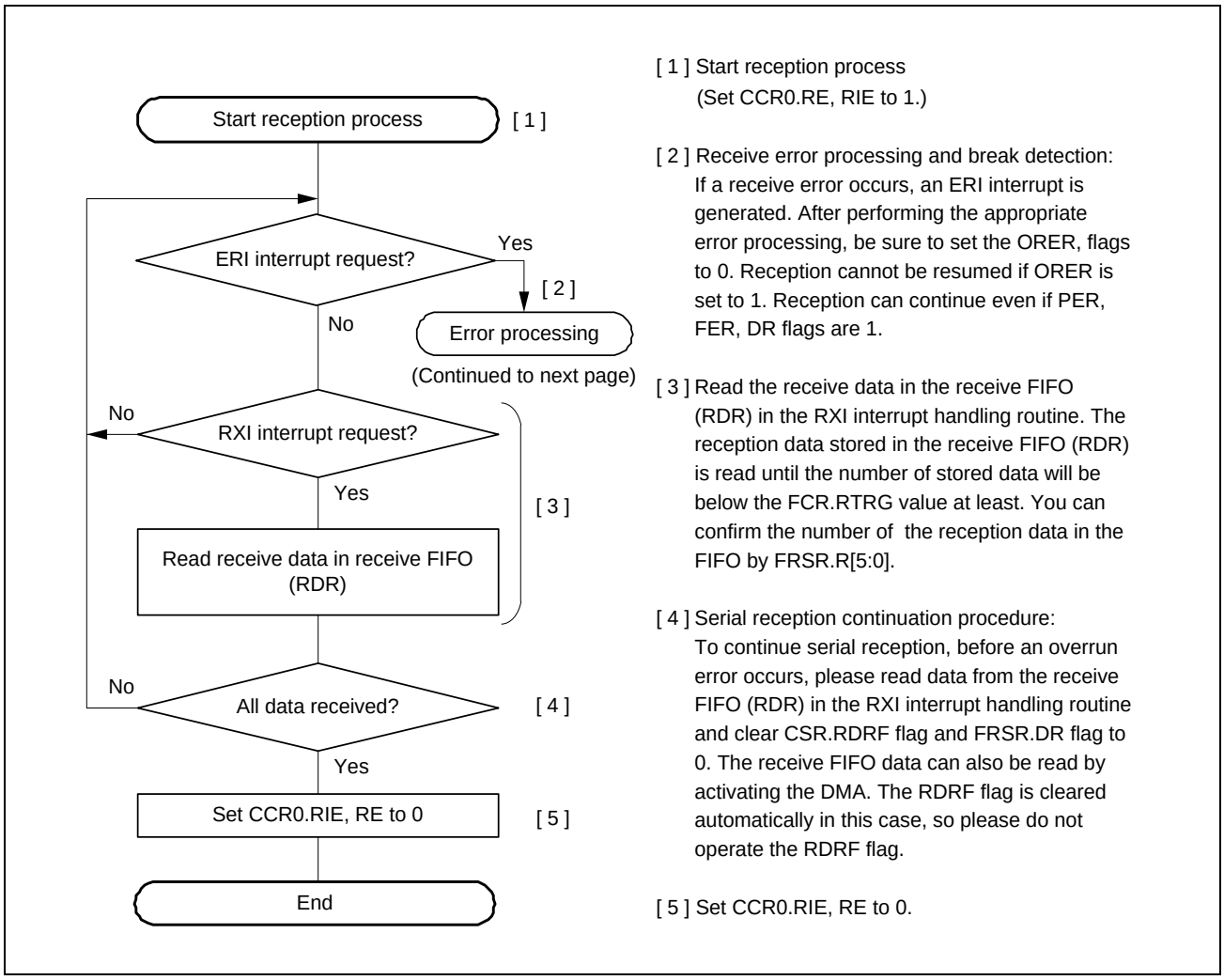
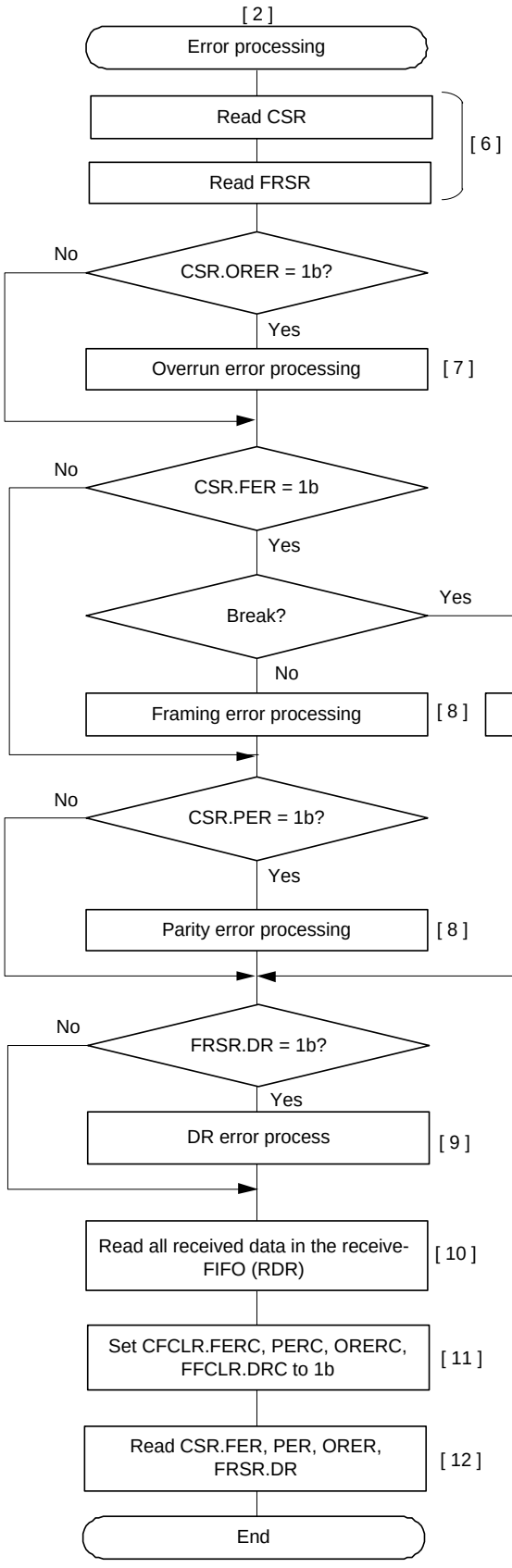


Figure 7.3-23 Example Flowchart of Serial Reception in Asynchronous Mode (FIFO Selected) (1)



[6] Reception error flag confirmation:
Read CSR and FRSR.
You can identify error by checking flags.

[7] Overrun error processing:
Read the receive FIFO (RDR) and make space in the receive FIFO (RDR).

[8] Break processing:
When CCR3.RXDESEL is 0b, the latest receive data in the receive FIFO (RDR) is the break-data which has framing error and all data-bits are 0b. So, discard the latest receive data.

[8] Framing error / Parity error processing:
Read all error occurrence data stored in the receive FIFO (RDR). (You can substitute to write 1b in FCR.RFRST bit and empty the receive FIFO (RDR).)
When [10] is handled in addition to this processing, the next frame can be normally received.

[9] DR error process when FCR.DRES is 1b.
Performs error processing required for the system (for example, setting a counter for data number management).

[10] Read all received data in the receive-FIFO (RDR).

[11] Clear the error flag:
Make clear FER, PER, ORER, DR

[12] Confirming that the error flag is cleared:
Confirm FER, PER, ORER, DR are cleared.

Figure 7.3-24 Example Flowchart of Serial Reception in Asynchronous Mode (FIFO Selected) (2)

7.3.3.10 The Function of Adjust Receive Sampling Timing (Asynchronous Mode)

When there is the difference between the rising transfer time and the falling transfer time through a photo coupler, the receive sampling timing at middle of bit affects the reception margin. In this case, the receive sampling timing can adjust from the middle of bit to the optimum timing by using this function.

The receive sampling timing is adjusted from the middle of bit by following formula. And the adjustable direction is set by CCR4.AJD. You can select forward or backward from the middle of bit. When adjusting backward (CCR4.AJD = 0b), substitute AJD = +1 and substitute AJD = -1 when adjusting forward (CCR4.AJD = 1b).

Adjusted sampling timing = the middle of bit + AJD × (base clock × the setting value of CCR4.AST[2:0]) The setting timing is limited by base clock cycles per 1 bit. Please see **Table 7.3-27** in detail.

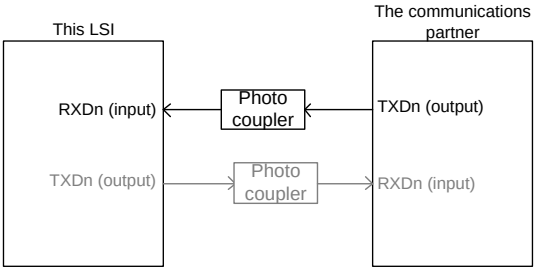
A reception movement image figure of the communication through a photo coupler with this function is shown in **Figure 7.3-25** and **Figure 7.3-26**, the movement explanation of this function is shown in **Figure 7.3-27**.

Do not use this function when there is no difference between the rising transfer time and the falling transfer time, because there is a possibility of deteriorating the reception margin.

Table 7.3-27 The Acceptable Value of Setting Register (Asynchronous Mode Using Internal Clock)

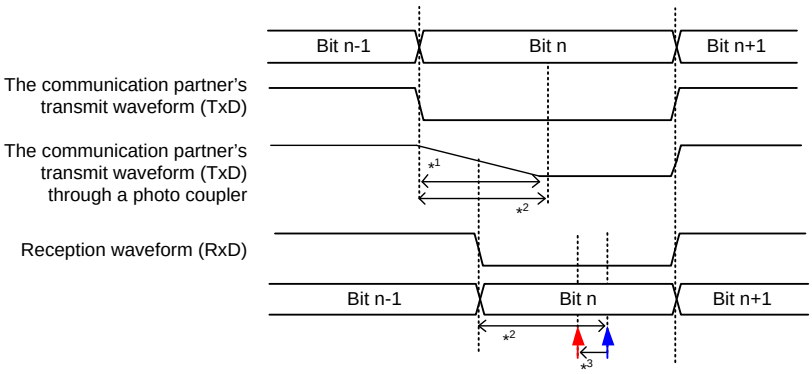
CCR2.ABCSE	CCR2.ABCS	The Number of Base Cycles/1 Bit	The Acceptable Value	
			CCR4.AJD	CCR4.AST
1	x	6	0	000b to 010b*1
			1	
0	1	8	0	000b to 011b*1
			1	
0	0	16	0	000b to 111b
			1	

Note 1. When the value of CCR4.AST is over the acceptable value, sampling is done at default timing. (Adjustment of sampling is not done.)



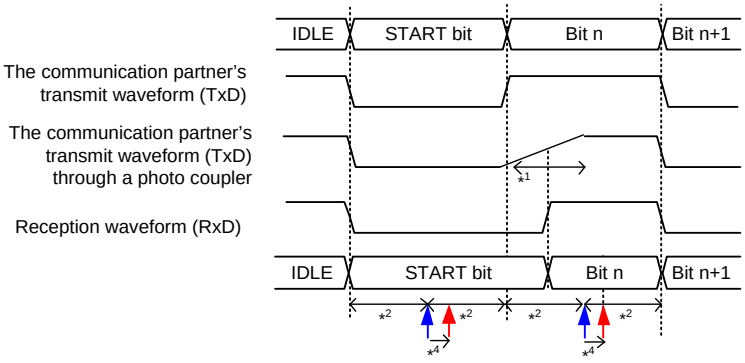
(a) In the case of the falling transfer time >> rising transfer time

The falling edge of reception waveform is made dull like following chart.
In this case, you can sampling at the middle of bit if you adjust the receive sampling timing to forward (AJD = 1b).



(b) In the case of the falling transfer time << rising transfer time

The rising edge of reception waveform is made dull like following chart.
So, the reception margin of communications partner will be bad. In this case, you can improve the reception margin if you adjust the receive sampling timing to back.



The receive sampling timing when unadjusted (middle of bit)

The adjusted receive sampling timing

Note: This waveform shows the operation image of adjustment receive sampling timing.

Note 1. The dull period by a photo coupler

Note 2. Bit center timing at set communication rate

Note 3. When CCR4.AJD is 1b, the receive sampling timing is shifted to forward by the setting value of CCR4.AST[2:0].

Note 4. When CCR4.AJD is 0b, the Receive sampling timing is shifted to backward by the setting value of CCR4.AST[2:0].

Figure 7.3-25 The Reception Movement Image Figure of the Communication through a Photo Coupler

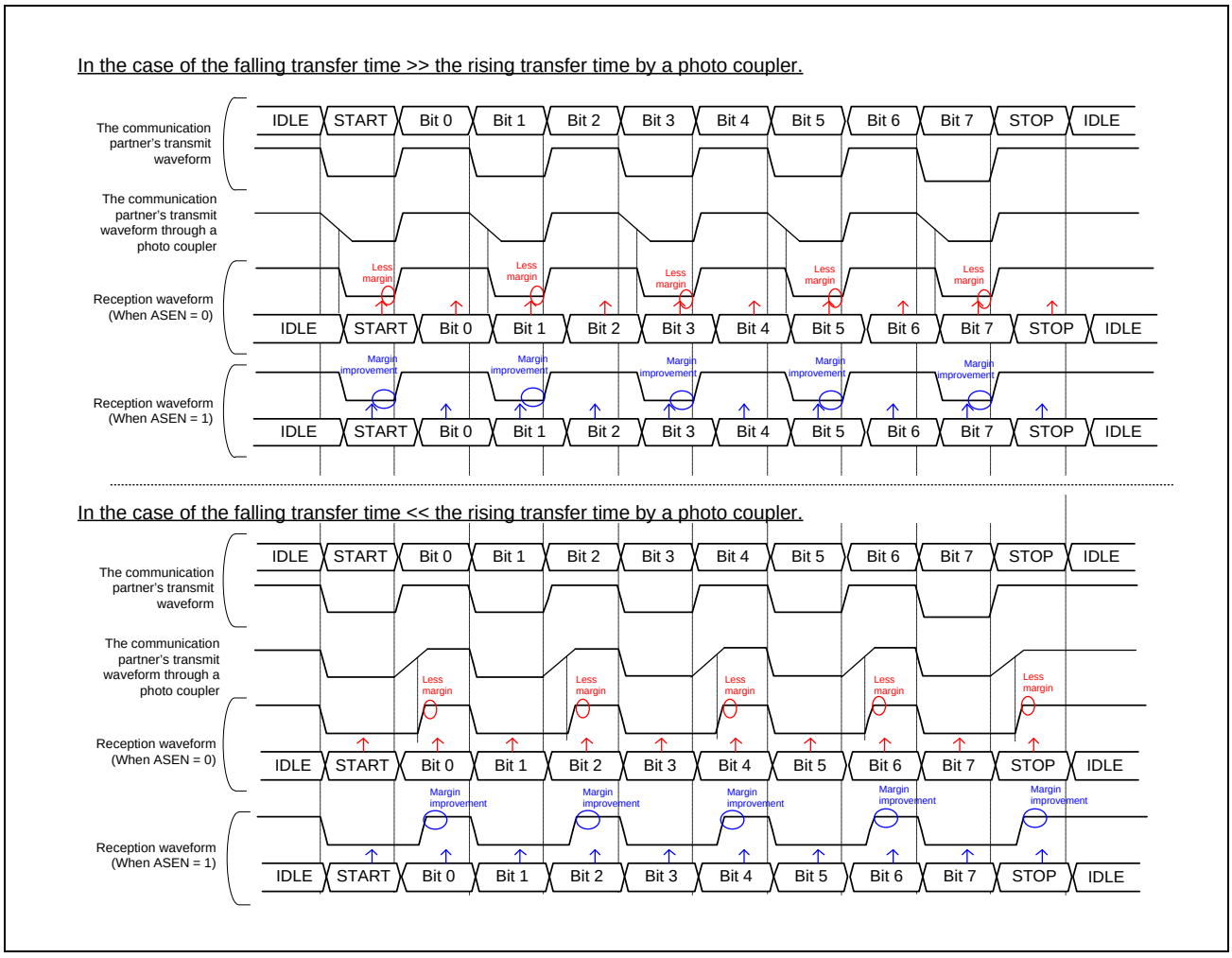


Figure 7.3-26 Improvement of Reception Margin when Receive Sampling Timing Adjustment Function Image

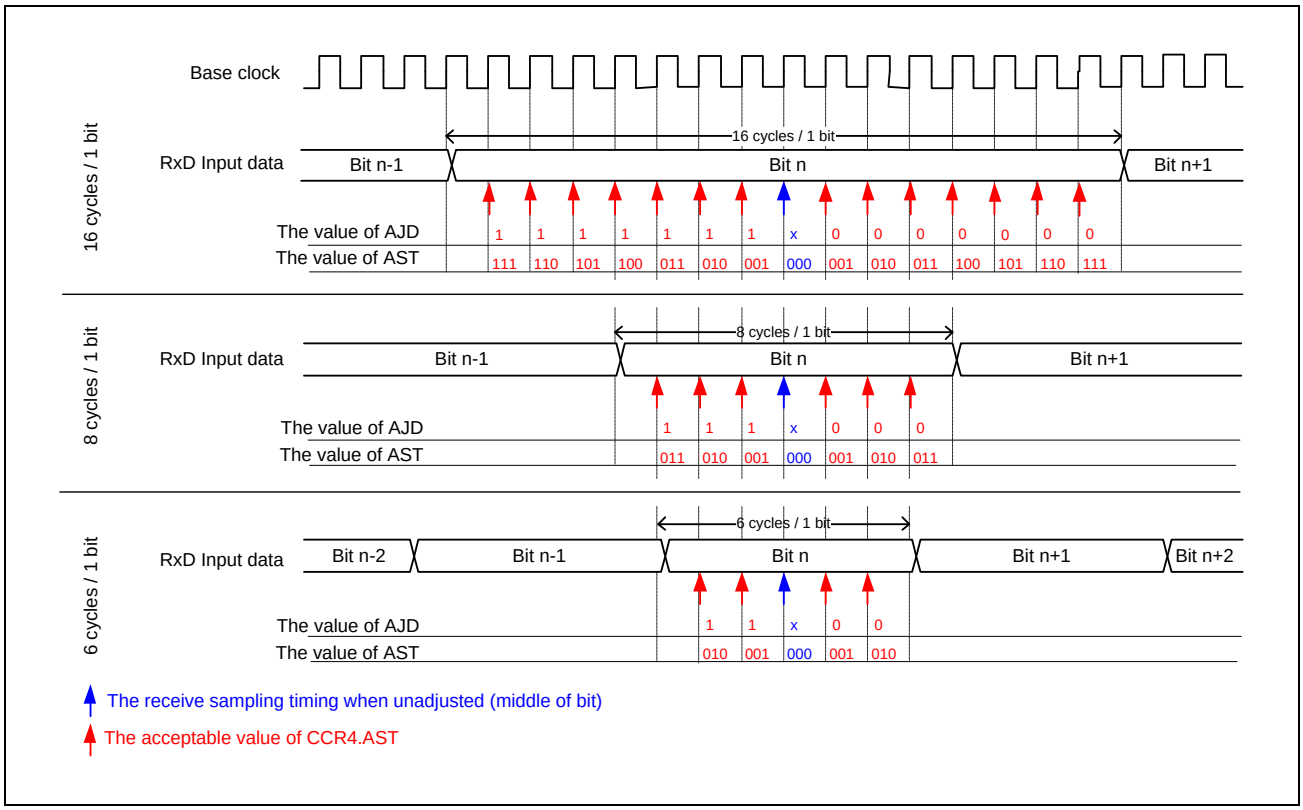


Figure 7.3-27 The Adjustment Operation Explanation for the Receive Sampling Timing (Asynchronous Mode Using Internal Clock)

7.3.3.11 The Function of Adjust Transmit Timing (Asynchronous Mode)

In communication via a photo coupler or the like, when either the rising or falling transition time of the TxD output signal is long, then a communication partner receive dulled waveform. In this case, the reception margin may be affected.

In these cases, make a communication partner to be sampling at middle of bit using the function of adjust transmit timing. When CCR4.ATEN is 1b, this function can adjust the edge timing at the timing calculated by the following formula for the edge set with CCR4.AET.

$$\text{The adjustment edge timing} = \text{base clock} \times \text{CCR4.ATT}[2:0]$$

In addition, the upper limit of the adjustment edge timing is limited by setting the base clock cycles. For details, see **Table 7.3-28**.

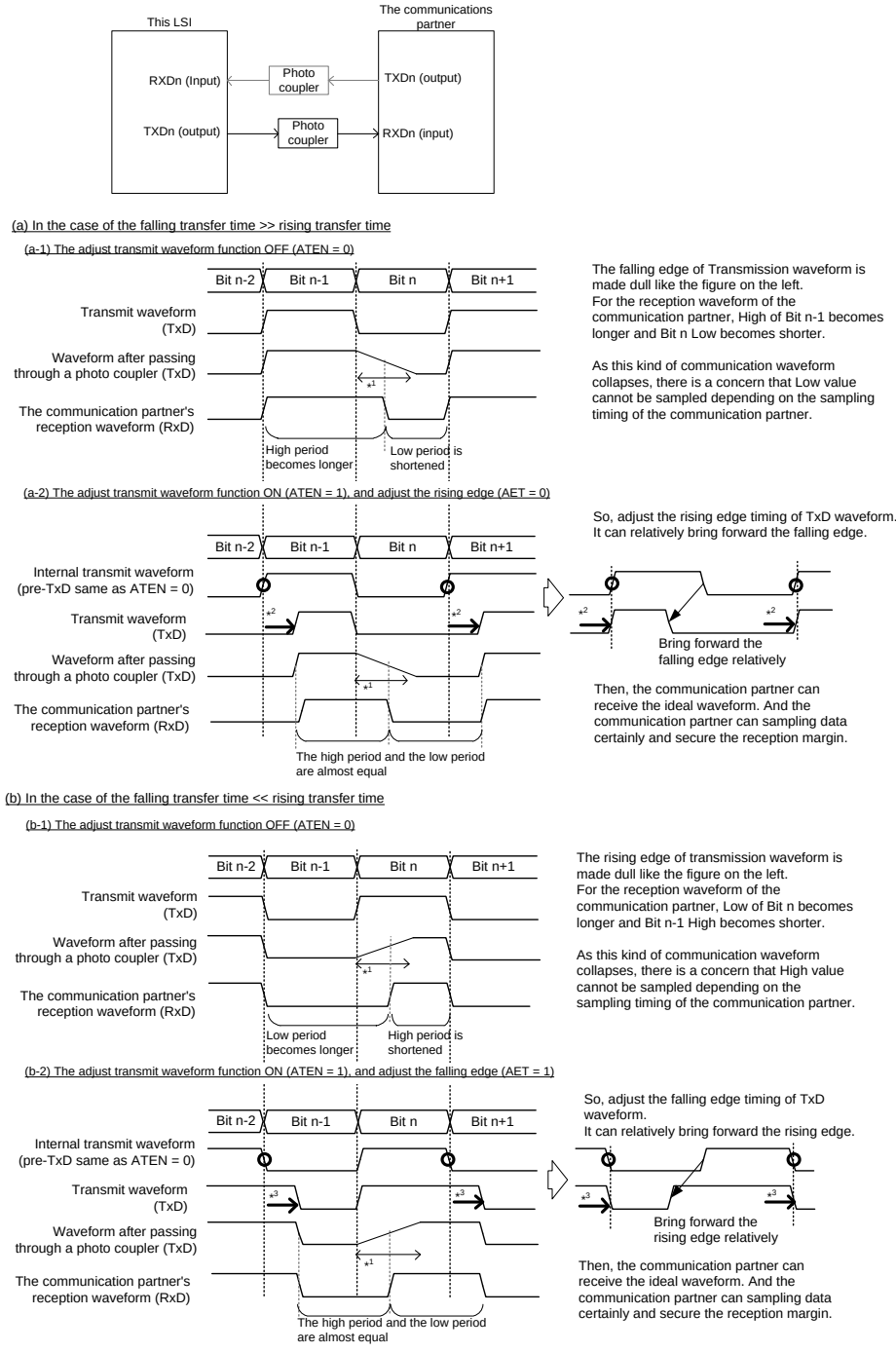
A transmission movement image figure of the communication through a photo coupler with this function is shown in **Figure 7.3-28** and **Figure 7.3-29**, the movement explanation of this function is shown in **Figure 7.3-30** and **Figure 7.3-31**.

Do not use this function when there is not the difference between the rising transfer time and the falling transfer time, there is a possibility of deteriorating the reception margin of a communication partner.

Table 7.3-28 The Acceptable Value of CCR4.AET and CCR4.ATT (Asynchronous Mode Using Internal Clock)

ABCSE	ABCS	The Number of Base Clock Cycles/1 Bit	The Acceptable Value	
			AET	ATT[2:0]
1	x	6	0	000b to 101b
			1	
0	1	8	0	000b to 111b
			1	
0	0	16	0	000b to 111b
			1	

Note: When the value of CCR4.AET/ATT is out of the acceptable value, this module does not adjust transmit timing.



Note 1. The dull period by a photo coupler

Note 2. When CCR4.AET is 0b, the rising edge of transmit waveform timing is shifted to back by the setting value of CCR4.ATT[2:0]. It can transmit waveform bringing forward the falling edge relatively.

Note 3. When CCR4.AET is 1b, the falling edge of transmit waveform timing is shifted to back by the setting value of CCR4.ATT[2:0]. It can transmit waveform bringing forward the rising edge relatively.

Note: This waveform shows the operation image of adjustment transmit waveform.

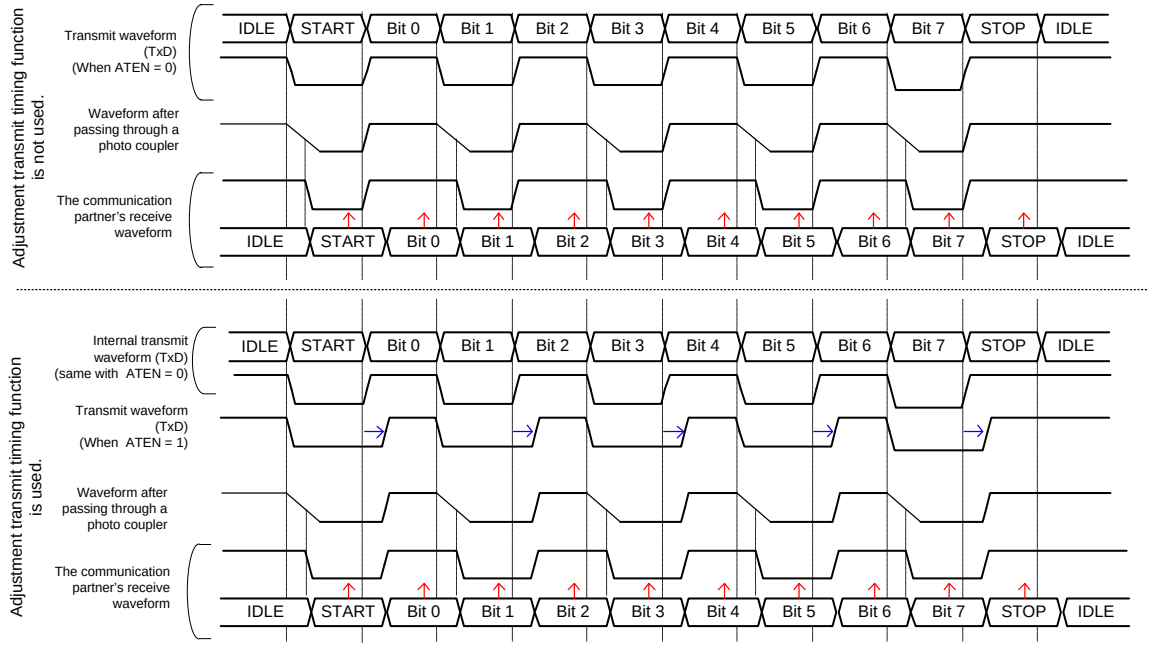
Figure 7.3-28 The Transmission Movement Image Figure of the Communication through a Photo Coupler

The explanation of transmit waveforms of the communication through a photo coupler using adjust transmit timing function

When using the transmission timing adjustment function, adjust the edge timing of the transmission waveform and correct the reception waveform of the communication partner

The following example is 8 bit long data.

(a) In the case of the falling edge transfer time >> the rising transfer time



(b) In the case of the falling edge transfer time << the rising transfer time

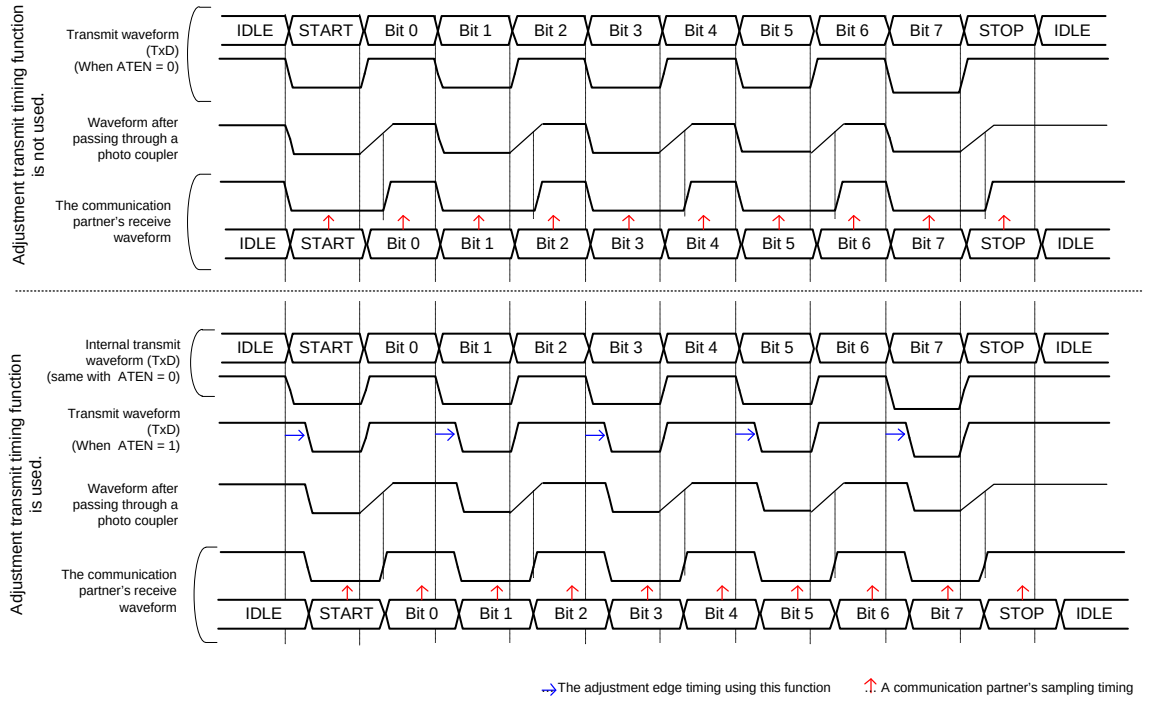


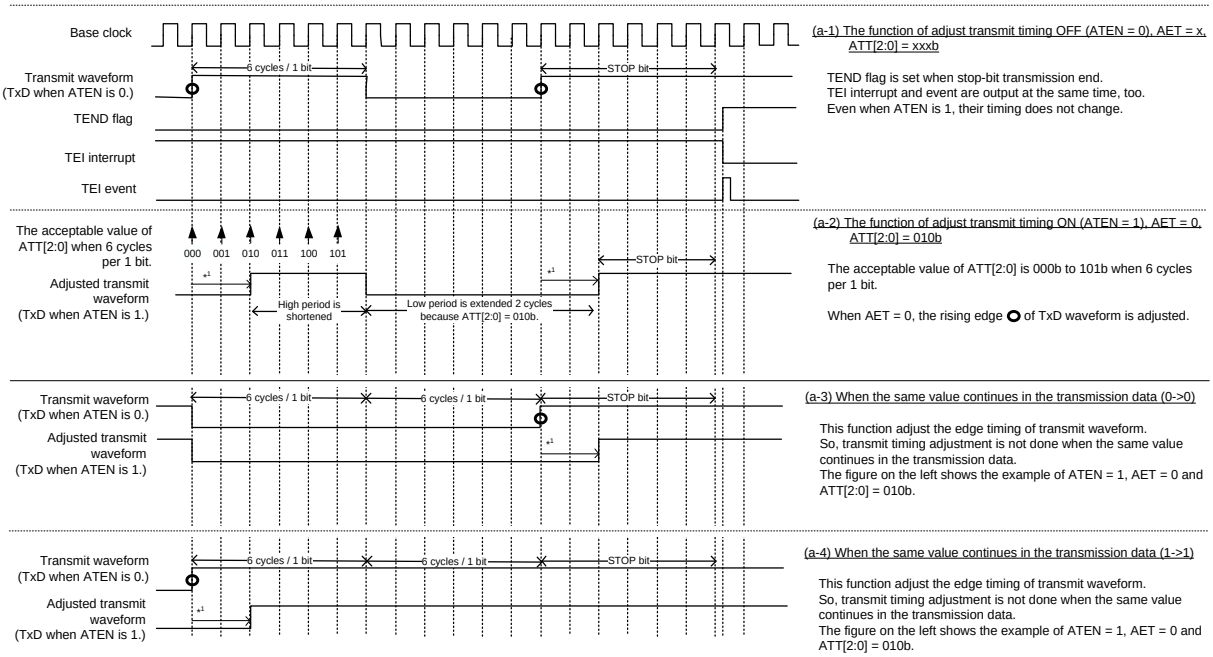
Figure 7.3-29 The Explanation for the Transmit Waveform through a Photo Coupler

The operation explanation of adjustment the transmit timing

(a) In the case of the falling transfer time >> rising transfer time

In this case, the high period of a communication partner's reception waveform is made long, and the low period is made short.
Therefore, the LSI transmits the waveform with the falling edge relatively brought forward by adjusting the rising edge timing.
Then adjust value (ATT[2:0]) should be set to make equal the low-period/1 bit and high-period/1 bit for a communication partner.

This function's operation is explained as an example of 6 cycles/1 bit.



Note 1. The rising edge of transmit timing is shifted to back by the setting value of CCR4.ATT[2:0].

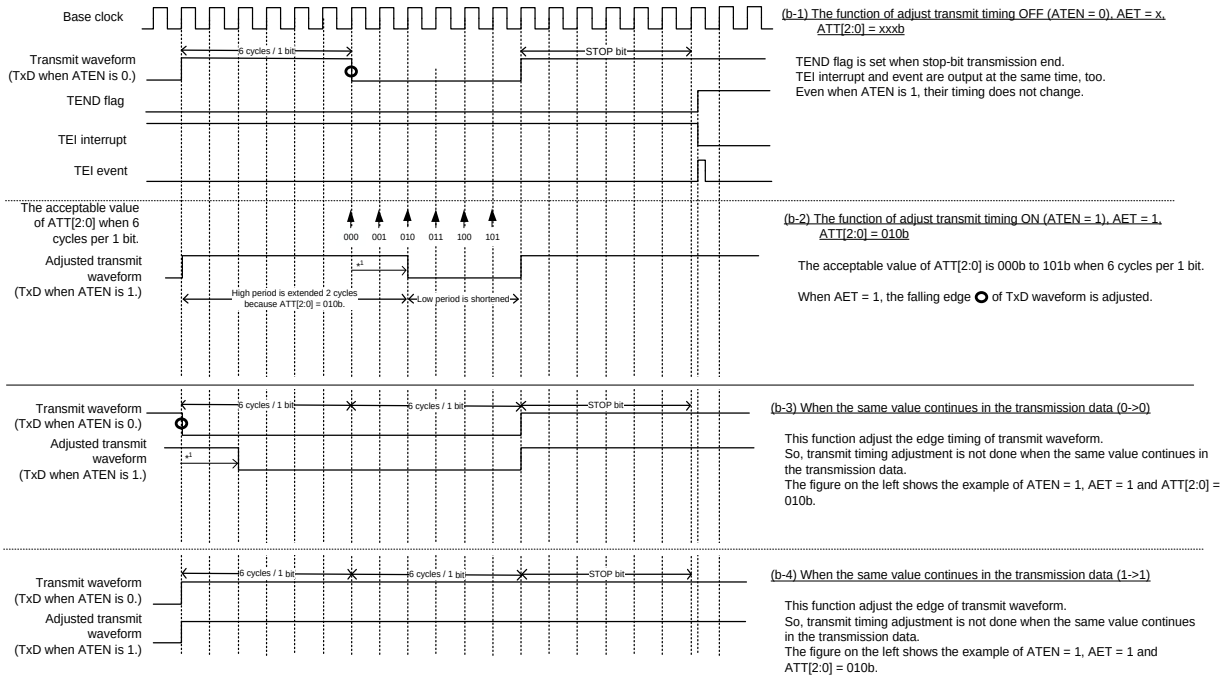
Figure 7.3-30 The Adjustment Operation Explanation for the Transmit Timing when AET is 0b

The operation explanation of adjustment the transmit timing

(b) In the case of the falling transfer time << rising transfer time

In this case, the low period of a communication partner's reception waveform is made long, and the high period is made short.
Therefore, the LSI transmits the waveform with the rising edge relatively brought forward by adjusting the falling edge timing.
The adjust value (ATT[2:0]) should be set to make equal the low-period/1 bit and high-period/1 bit for a communication partner.

This function's operation is explained as an example of 6 cycles/1 bit.



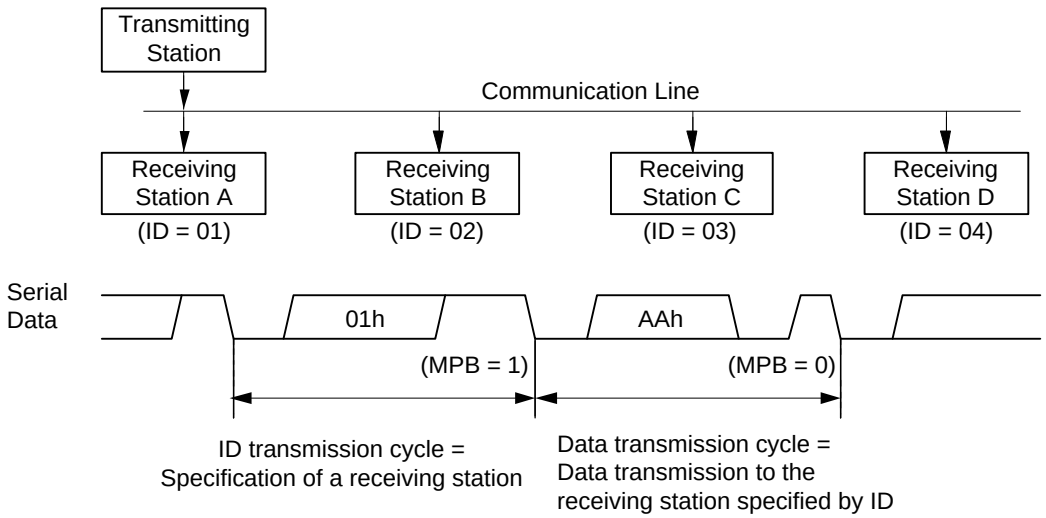
Note 1. The falling edge of transmit timing is shifted to back by the setting value of CCR4.ATT[2:0].

Figure 7.3-31 The Adjustment Operation Explanation for the Transmit Timing when AET is 1b

7.3.4 Multi-Processor Communication Function

Using the multi-processor communication functions enables to transmit and receive data by sharing a communication line between multiple processors by using asynchronous serial communication in which the multi-processor bit is added. In multi-processor communication, a unique ID code is allocated to each receiving station. Serial communication cycles consist of an ID transmission cycle to specify the receiving station and a data transmission cycle to transmit data to the specified receiving station. The multi-processor bit is used to distinguish between the ID transmission cycle and the data transmission cycle. When the multi-processor bit is set to 1b, it indicates the ID transmission cycle and when the multiprocessor bit is set to 0b, it indicates the data transmission cycle. **Figure 7.3-32** shows an example of communication between processors by using a multi-processor format. First, a transmitting station transmits communication data in which the multi-processor bit set to 1b is added to the ID code of the receiving station. Next, the transmitting station transmits the communication data in which the multi-processor bit set to 0b is added to the transmit data. On receiving the communication data in which the multi-processor bit is set to 1b, the receiving station compares the received ID with the ID of the receiving station itself and if the two matches, receives the communication data that is subsequently transmitted. If the received ID does not match with the ID of the receiving station, the receiving station skips the communication data until again receiving the communication data in which the multi-processor bit is set to 1b.

RTS control cannot be used at the time of multi-processor communication function use, because this is a function corresponding to one-to-many communications.



Note: MPB: Multi-processor bit

Figure 7.3-32 An Example of Communication Using the Multi-Processor Format (Example of Transmission of Data AAh to Receiving Station A)

(1) Non-FIFO selected

For supporting this function, the SCI provides the MPIE bit in CCR0. When the MPIE bit is set to 1b, transfer of receive data from the RSR to the RDR, detection of a receive error, and setting the respective status flags RDRF, ORER, and FER in CSR are disabled until reception of data in which the multi-processor bit is set to 1b.

On receiving a reception character in which the multi-processor bit is set to 1b, the MPB bit in RDR is set to 1b and the MPIE bit in CCR0 is automatically cleared, thus returning to a non-multi-processor reception operation. At this time, an RXI interrupt is generated if the RIE bit in CCR0 is set.

When the multi-processor format is specified, specification of the parity bit is disabled.

Apart from this, there is no difference from the operation in the non-multi-processor asynchronous mode. A clock which is used for the multi-processor communication is also the same as the clock used in the non-multi-processor asynchronous mode.

(2) FIFO selected

For transmission, it should write to TDR.MPBT (Multi-Processor Bit Transfer) which corresponds to transmit data in TDR.TDAT. For reception, Multi-processor bit that is a part of receive data, is stored to RDR.MPB, and receive data is stored to RDR.RDAT. When the MPIE bit is set to 1b, transfer of receive data from the RSR to the RDR.RDAT, detection of a receive error, and detection of DR, and setting the respective status flags RDRF, ORER, and FER in CSR are disabled until reception of data in which the multi-processor bit is set to 1b.

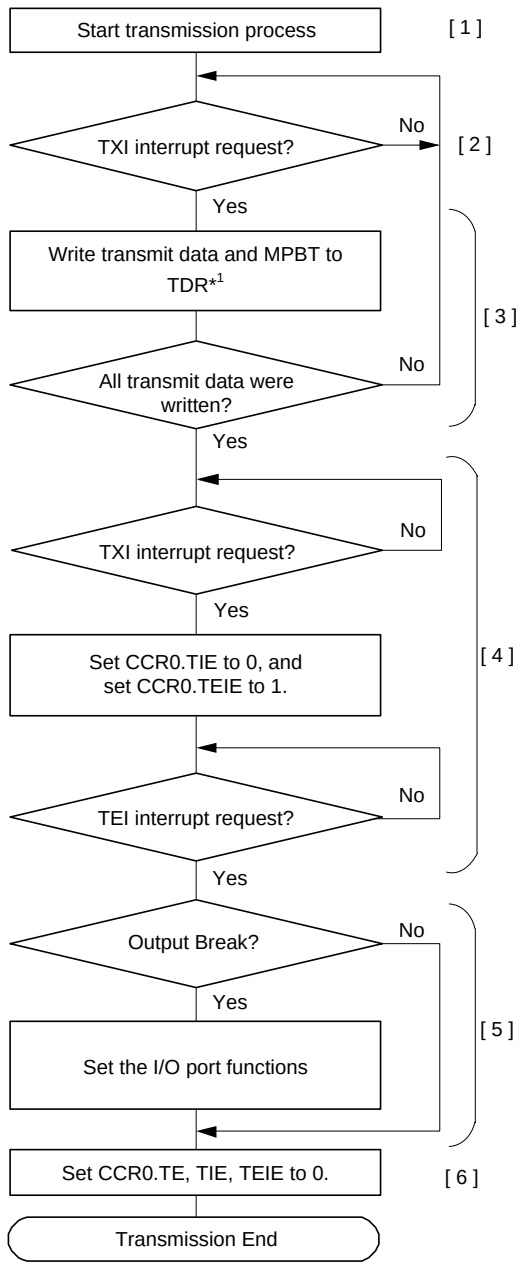
On receiving a reception character in which the multi-processor bit is set to 1b, the MPB bit in RDR is set to 1b, and receive data is stored to receive FIFO (RDR.RDAT), and the MPIE bit in CCR0 is automatically cleared, thus returning to a non- multi-processor reception operation. At this time, an RXI interrupt is generated if the RIE bit in CCR0 is set.

When the multi-processor format is specified, specification of the parity bit is disabled. Apart from this, there is no difference from the operation in the non-multi-processor asynchronous mode and non-FIFO selected.

7.3.4.1 Multi-Processor Serial Data Transmission

(1) Non-FIFO selected

Figure 7.3-33 shows a sample flowchart of multi-processor data transmission. In the ID transmission cycle, the ID should be transmitted with the MPBT bit in TDR set to 1b. In the data transmission cycle, the data should be transmitted with the MPBT bit set to 0b. The other operations are the same as the operations in asynchronous mode.



[1] Start transmission process:
To enable TXI interrupt at the start of transmission, set TE and TIE to 1 with one instruction at the same time. And set CCR3.MP to 1.

[2] TXI interrupt request:
The transmit-data-empty-interrupt (TXI) is generated at following conditions:
When set TE and TIE to 1 at same time.
When transmission-data forwards to TSR from TDR.

[3] Serial transmission continuation procedure:
To continue serial transmission write transmit data and MPBT setting value to TDR*1 once by using a TXI. Transmit data and MPBT setting value can also be written by activating the DMAC.

[4] If you use TEI, set TIE = 0 and TEIE = 1 after writing the last-transmit-data to TDR*1 and transmission is started.

[5] Break output at the end of serial transmission:
To output a break, it is necessary to set SPB2IO and SPB2DT, for output status setting when TE is 0. (If you set these bits at initialization, you can skip this step.)

[6] Set CCR0.TE, TIE, TEIE to 0.

Note 1. In case of byte access with data length 9 bits, write transmit data in order of TDR[15:8] and TDR[7:0].

Figure 7.3-33 Example of Multi-Processor Serial Transmission Flowchart (Non-FIFO Selected)

(2) FIFO selected

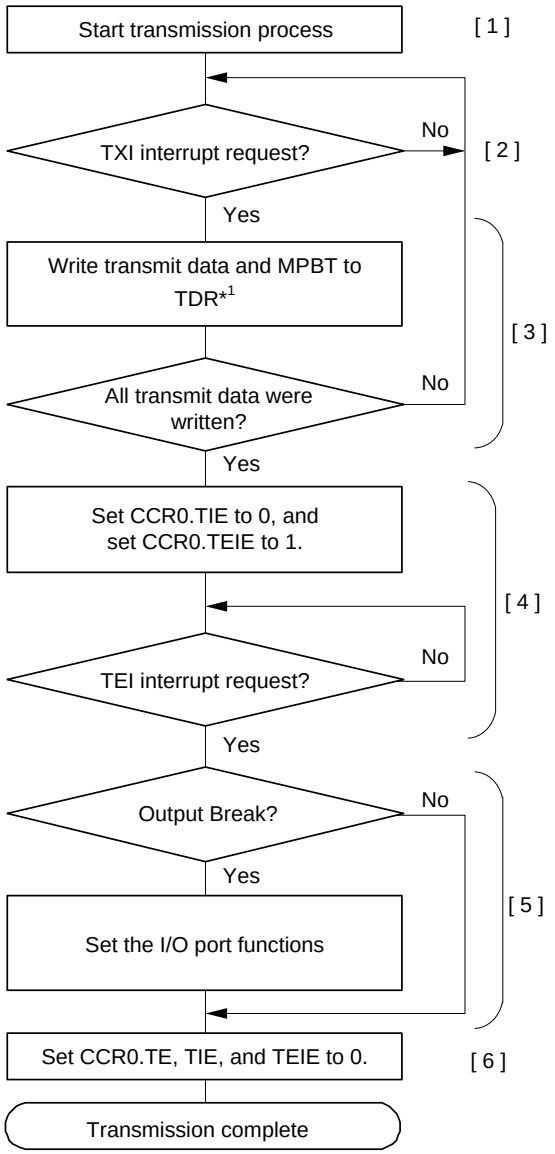
Figure 7.3-34 shows an example of data format that is written to transmit FIFO (TDR.TDAT) in multi-processor mode. Write MPBT in bit 9 of TDR. And write data to transmit FIFO (TDR.TDAT) corresponded to data length. It should write to 0b for unused bits.

Data Length	Register Setting CCR3.CHR[1:0]		Transmit data in TDR[15:0]															
			b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
7 bits	1	1	—	—	—	—	—	—	MPBT	—	—	TDAT[6:0]						
8 bits	1	0	—	—	—	—	—	—	MPBT	—	TDAT[7:0]							
9 bits	0	x	—	—	—	—	—	—	MPBT	TDAT[8:0]								

Note: The bits Indicated by "—" are not used. Therefore, those bits should be set to 0b.

Figure 7.3-34 Data Format in Multi-Processor Mode that is Written to Transmit FIFO (TDR.TDAT)

Figure 7.3-35 shows a sample flowchart for multi-processor data transmission with FIFO selected. In the ID transmission cycle, the ID should be transmitted with the MPBT bit in TDR set to 1b. In the data transmission cycle, the data should be transmitted with the MPBT bit set to 0b. The other operations are the same as the operations in asynchronous mode with FIFO selected.



[1] Start transmission process.
To enable TXI interrupt at he start of transmission, set TE and TIE to 1 with one instructions at the same time. And set CCR3.FM and MP to 1.

[2] TXI interrupt request:
The transmit-data-empty-interrupt (TXI) is generated when transmission-data forwards to TSR from the transmit FIFO (TDR) and the number of transmit-data written in the transmit FIFO is equal to or less than the transmit FIFO threshold.

[3] Serial transmission continuation procedure:
To continue serial transmission, write all transmit data and MPBT setting value to the transmit FIFO (TDR) by using a TXI interrupt. Transmit data and MPBT setting value can also be written to the transmit FIFO (TDR) by activating the DMAC.

[4] If you use TEI, set TIE = 0 and TEIE = 1 after writing the last-transmit-data to TDR*1.

[5] Break output at the end of serial transmission:
To output a break, it is necessary to set SPB2IO and SPB2DT, for output status setting when TE is 0.
(If you set these bits at initialization, you can skip this step.)

[6] Set CCR0.TE, TIE, and TEIE to 0.

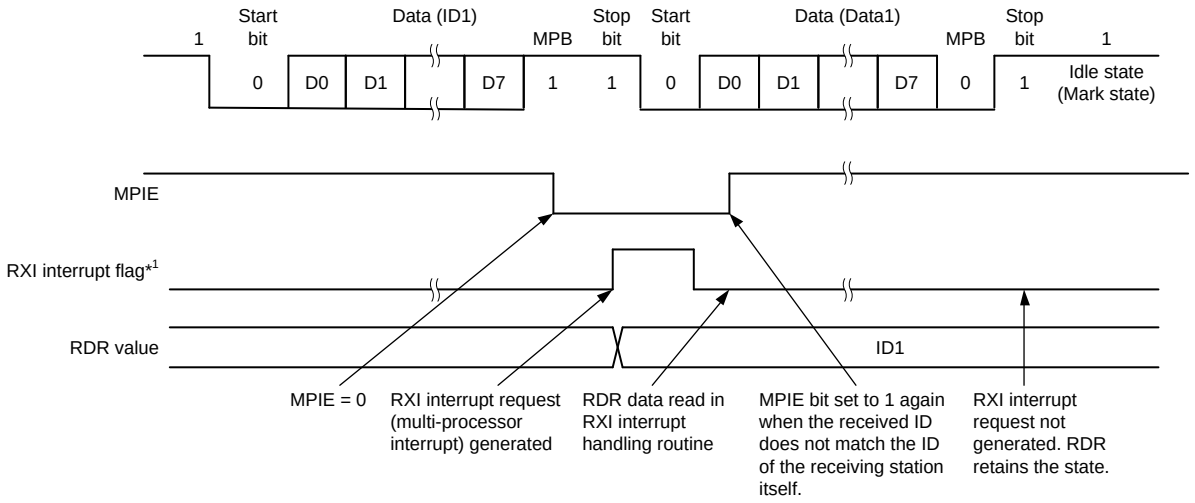
Note 1. In case of byte access with data length 9 bits, write transmit data in order of TDR[15:8] and TDR[7:0].

Figure 7.3-35 Example of Serial Transmission Flowchart in Multi-Processor Mode (FIFO Selected)

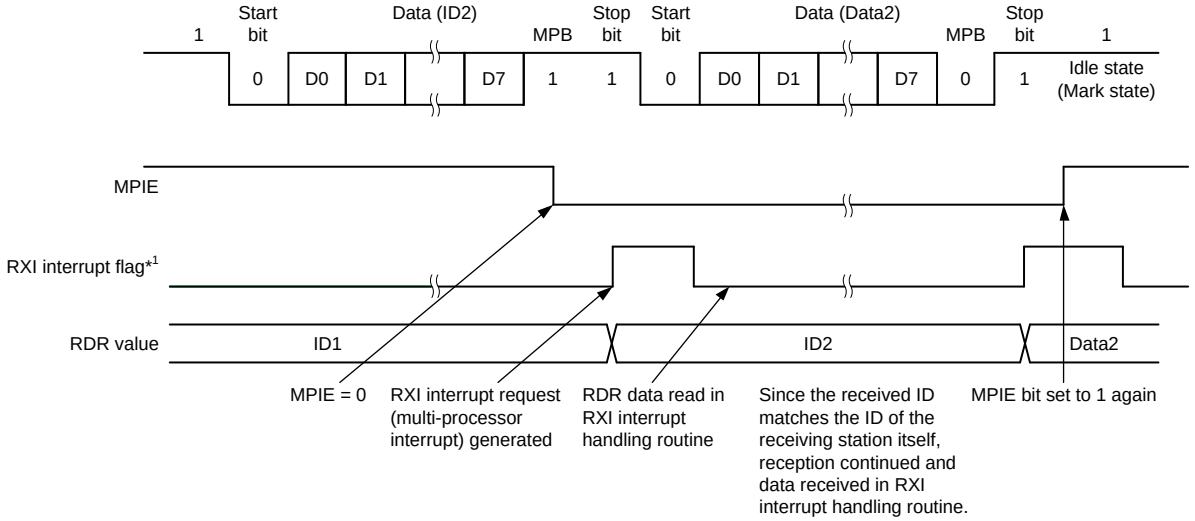
7.3.4.2 Multi-Processor Serial Data Reception

(1) Non-FIFO selected

Figure 7.3-37 and **Figure 7.3-38** are sample flowcharts of multi-processor data reception. When the MPIE bit in CCR0 is set to 1b, reading the communication data is skipped until reception of the communication data in which the multi-processor bit is set to 1b. When the communication data in which the multi-processor bit is set to 1b is received, the received data is transferred to RDR. At this time, the RXI interrupt request is generated. The other operations are the same as the operations in asynchronous mode. **Figure 7.3-36** is the example of operation for reception.



(a) When the received ID does not match the ID of the receiving station itself



(b) When the received ID matches the ID of the receiving station itself

Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-36 Example of SCI Reception (8-Bit Data/Multi-Processor Bit/1 Stop Bit)

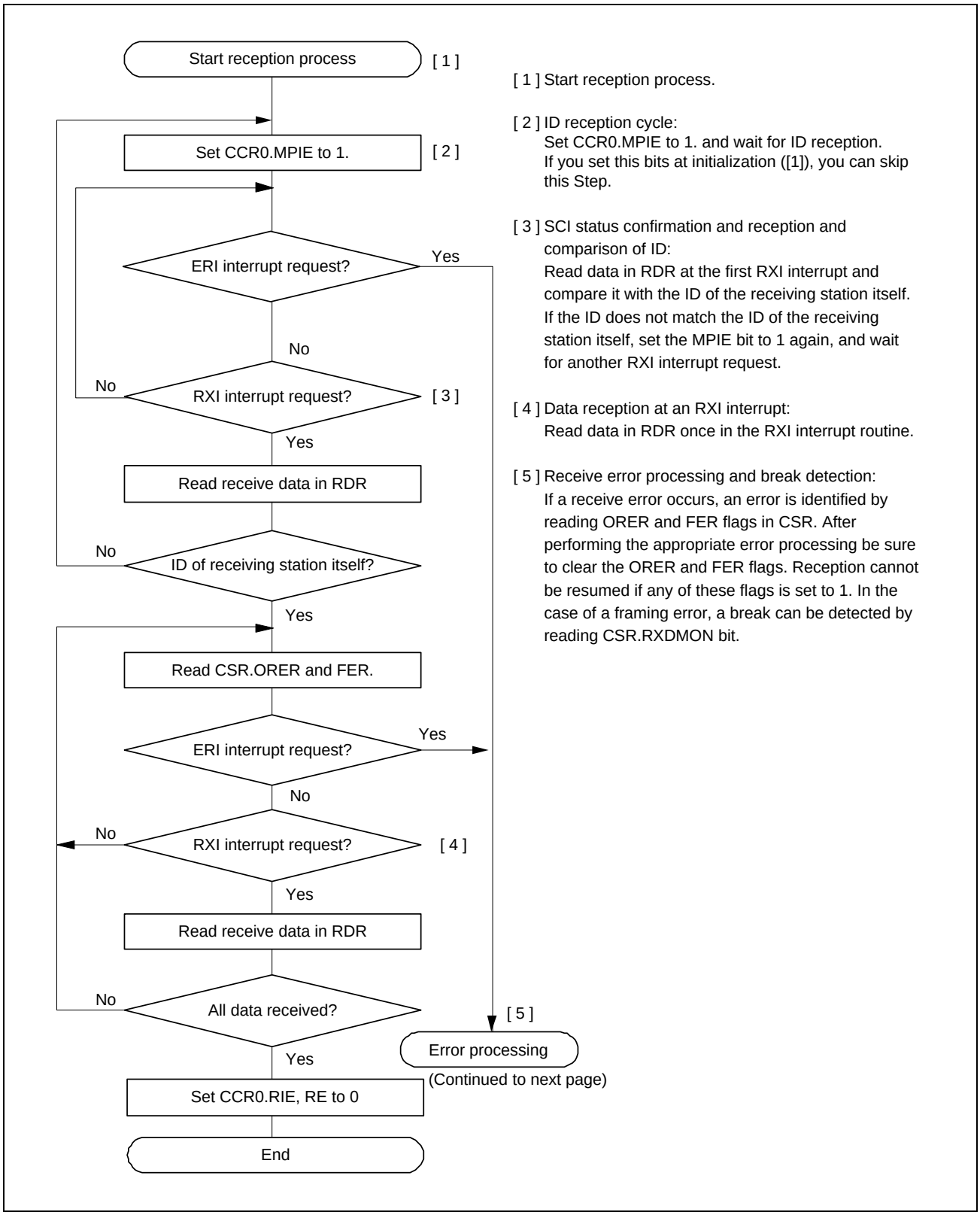


Figure 7.3-37 Example of Multi-Processor Serial Reception Flowchart (1) (Non-FIFO Selected)

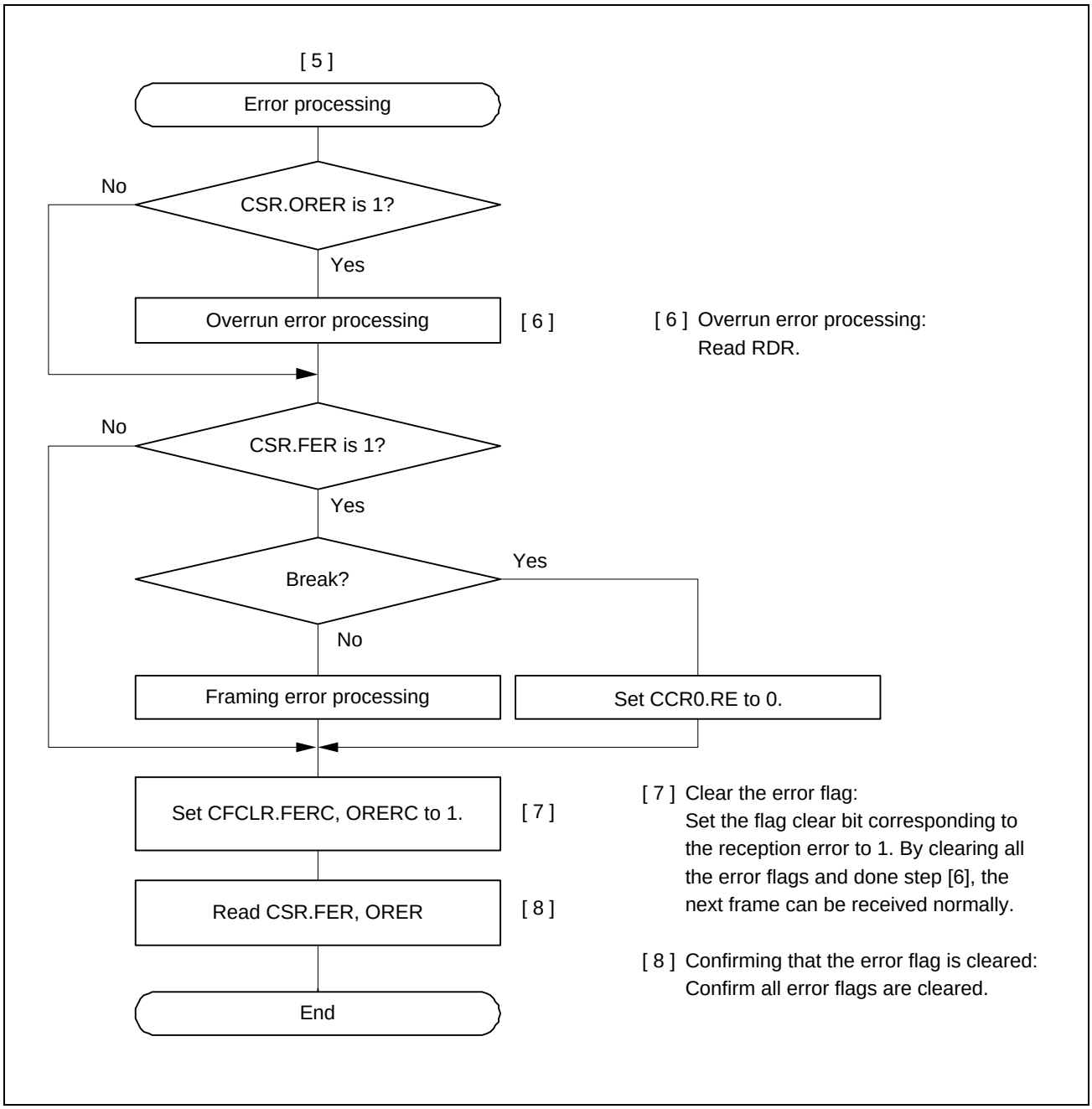


Figure 7.3-38 Example of Multi-Processor Serial Reception Flowchart (2) (Non-FIFO Selected)

(2) FIFO selected

Figure 7.3-39 shows an example of data format that is stored to receive FIFO (RDR) in multi-processor mode. MPB is stored in bit 9 of RDR. 0b is stored to FPER and PER flag bit. Data is stored to receive FIFO (RDR) corresponded to data length. It is stored to 0b for unused bits. Reading the receive FIFO (RDR register) updates the FFER, FPER, MPB flags and receive data (RDAT[8:0]) in the receive FIFO (RDR register) with the next received data. The FER, PER, and ORER flags in the receive FIFO (RDR register) always reflect the status of the corresponding flags in the CSR and FRSR registers.

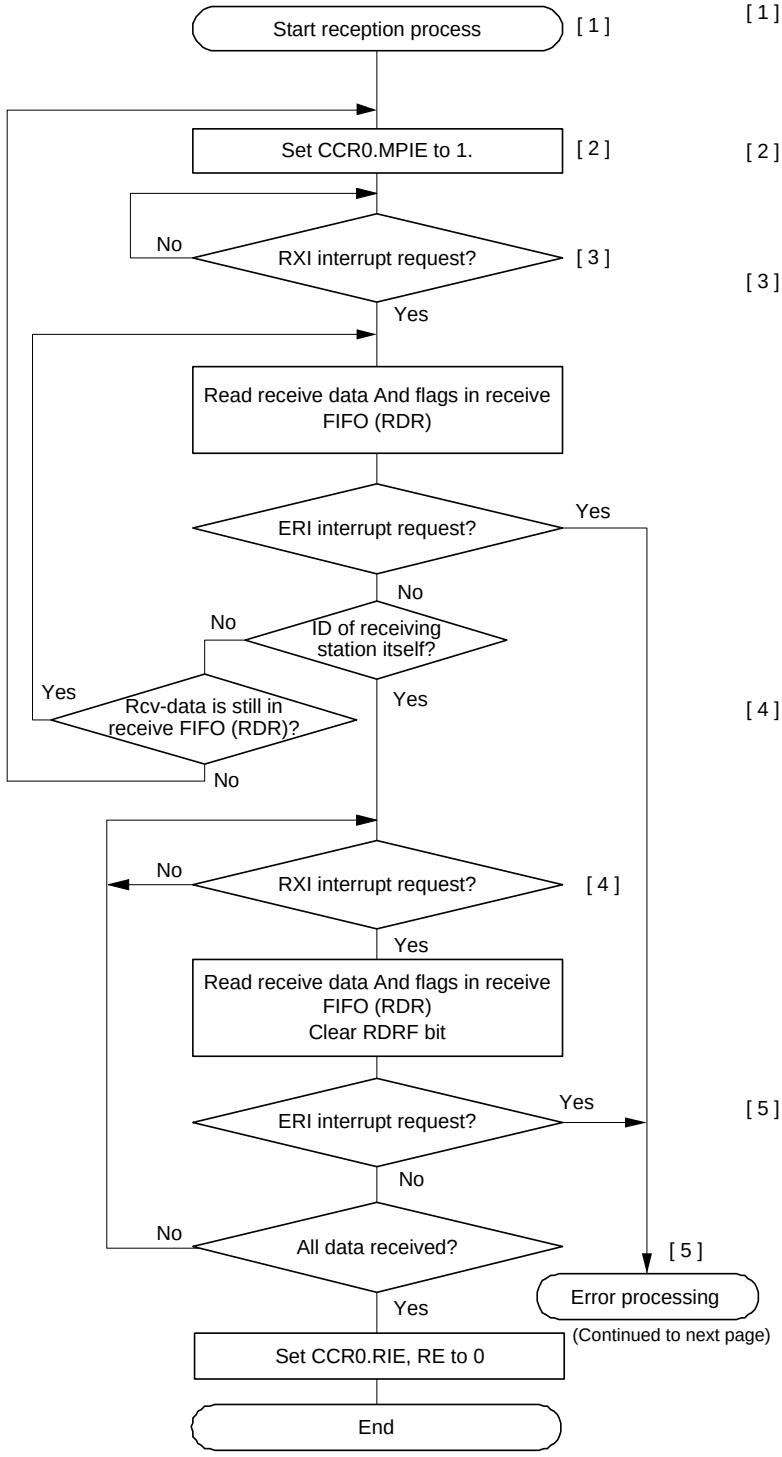
Data Length	Register Setting CCR3.CHR[1:0]		Receive data in RDR[31:0] b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
7 bits	1	1	0	0	0	FFER	FPER	DR	MPB	0	0	RDAT[6:0]						
8 bits	1	0	0	0	0	FFER	FPER	DR	MPB	0	RDAT[7:0]							
9 bits	0	x	0	0	0	FFER	FPER	DR	MPB	RDAT[8:0]								
Data Length	Register Setting CCR3.CHR[1:0]		b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
7 bits	1	1	0	0	0	FER	PER	0	0	ORER	0	0	0	0	0	0	0	0
8 bits	1	0	0	0	0	FER	PER	0	0	ORER	0	0	0	0	0	0	0	0
9 bits	0	x	0	0	0	FER	PER	0	0	ORER	0	0	0	0	0	0	0	0

Note: When a 7-bit data length is selected, the RDAT[8:7] bits are always read as 00b.
When an 8-bit data length is selected, the RDAT[8] bit is always read as 0b.

Figure 7.3-39 Data Format in Multi-Processor Mode that is Stored to Receive FIFO (FIFO Selected)

Figure 7.3-40 shows a sample flowchart for multi-processor data reception at FIFO selected. When the MPIE bit in CCR0 is set to 1b, reading the communication data is skipped until reception of the communication data in which the multiprocessor bit is set to 1b. When the communication data in which the multi-processor bit is set to 1b is received, the received data, MPB, and each errors are transferred to receive FIFO (RDR), and the MPIE bit in CCR0 is automatically cleared, thus returning to a normal reception operation. After a framing error occurred and CSR.FER flag is set to 1b, but SCI continues data reception.

The other operations are the same as the operations in asynchronous mode at non-FIFO selected.



[1] Start reception process
To generate an RXI interrupt request with MP detection, please set FCR.RTRG=1 and FCR.DRES=0.

[2] ID reception cycle:
Set CCR0.MPIE to 1. and wait for ID reception. If you set this bits at Initialization ([1]), you can skip this Step.

[3] SCI status confirmation and reception and comparison of ID:
Skip the communication data until the MPB bit receives data "1".
When the MPB bit receives data "1", an RXI interrupt request is generated.
When an RXI interrupt request is received, read the data in the receiving FIFO (RDR register) and compare it to the ID of my station.
If it is not my station's ID, set the MPIE bit to "1" again and wait for the RXI interrupt request. For my station's ID, set the received FIFO threshold to the necessary value.

[4] Data reception at an RXI interrupt:
When the number of received data stored in the received FIFO (RDR register) exceeds the received FIFO threshold, the CSR.RDRF bit becomes "1" and an RXI interrupt request is generated. In addition, if the number of received data stored in the received FIFO (RDR register) is "1" or more and less than the received FIFO threshold, the FRSR.DR bit becomes "1" after 15 ETU, and an RXI interrupt request is generated. The RXI interrupt routine reads the data from the received FIFO (RDR register).

[5] Receive error processing and break detection:
If a receive error occurs, an error is identified by reading ORER and FER flags in CSR. After performing the appropriate error processing be sure to clear the ORER and FER flags. In the case of a framing error, a break can be detected by reading the CSR.RXDMON bit.

Figure 7.3-40 Example Flowchart of Serial Reception in Multi-Processor Mode (1) (FIFO Selected)

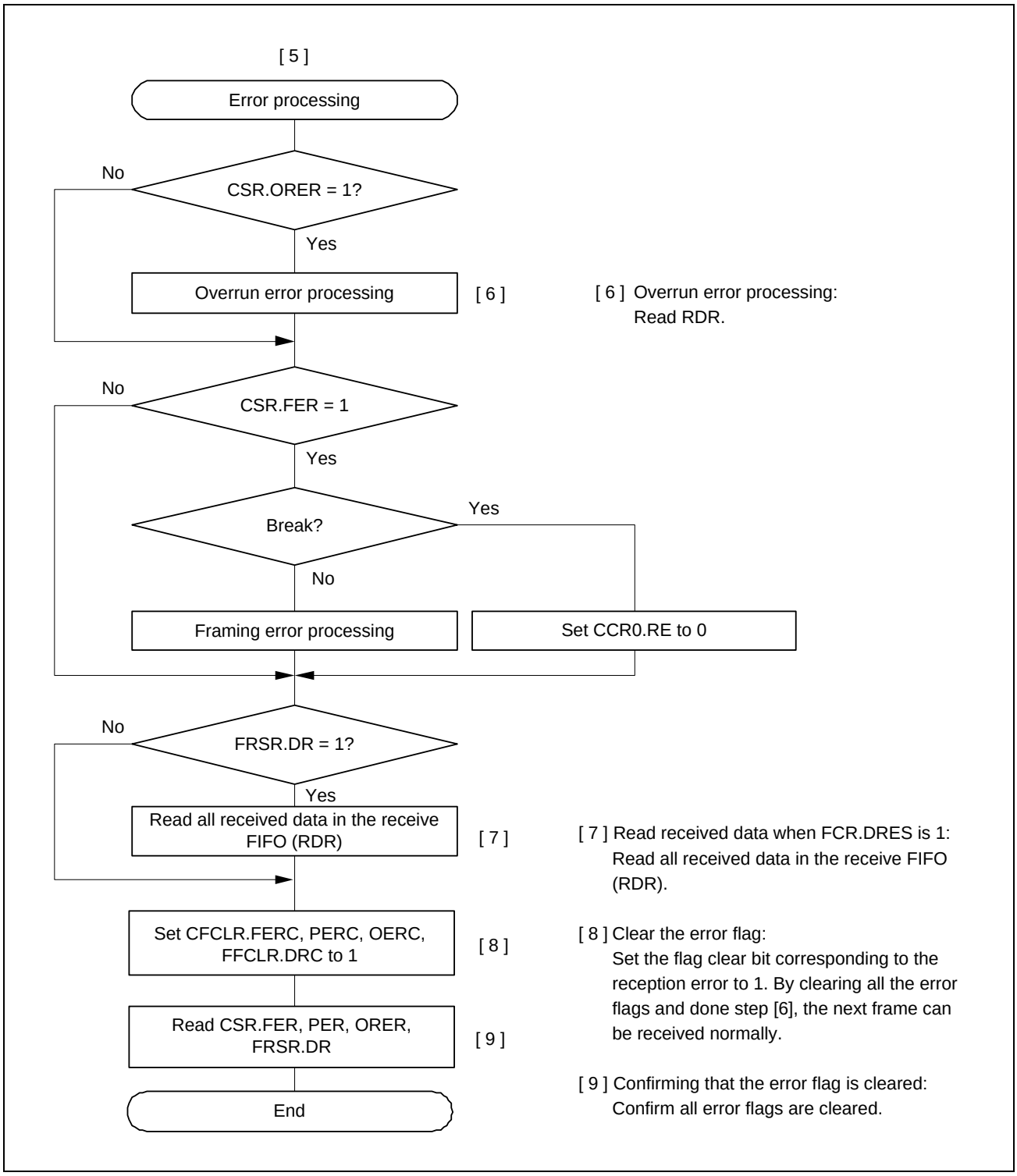


Figure 7.3-41 Example Flowchart of Serial Reception in Multi-Processor Mode (2) (FIFO Selected)

7.3.5 Smart Card Interface Mode

The SCI supports smart card (IC card) interfaces conforming to ISO/IEC 7816-3 (standard for Identification Cards), as an extended function.

Smart card interface mode can be selected using the appropriate register.

7.3.5.1 Sample Connection

Figure 7.3-42 shows a sample connection between a smart card (IC card) and this LSI.

As in the figure, since this LSI communicates with an IC card using a single transmission line, interconnect the TXDn and RXDn pins and pull up the data transmission line to VCC using a resistor. Setting the TE and RE bits in CCR0 to 1b with an IC card disconnected enables closed transmission/reception allowing self-diagnosis.

To supply an IC card with the clock pulses generated by the SCI, input the SCKm pin output to the CLK pin of an IC card.

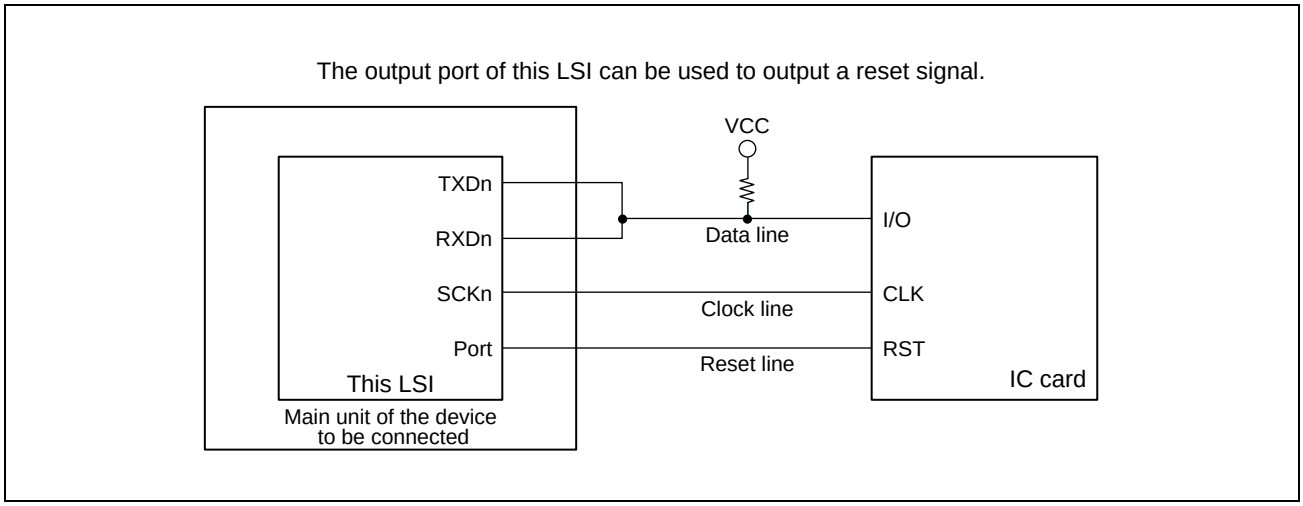


Figure 7.3-42 Sample Connection with a Smart Card (IC Card)

7.3.5.2 Data Format (Except in Block Transfer Mode)

Figure 7.3-43 shows the data transfer formats in smart card interface mode.

- One frame consists of 8-bit data and a parity bit in asynchronous mode.
- During transmission, at least 2 etu (elementary time unit: time required for transferring 1 bit) is secured as a guard time from the end of the parity bit until the start of the next frame.
- If a parity error is detected during reception, a low-level error signal is output for 1 etu after 10.5 etu has passed from the start bit.
- If an error signal is sampled during transmission, the same data is automatically retransmitted after at least 2 etu.

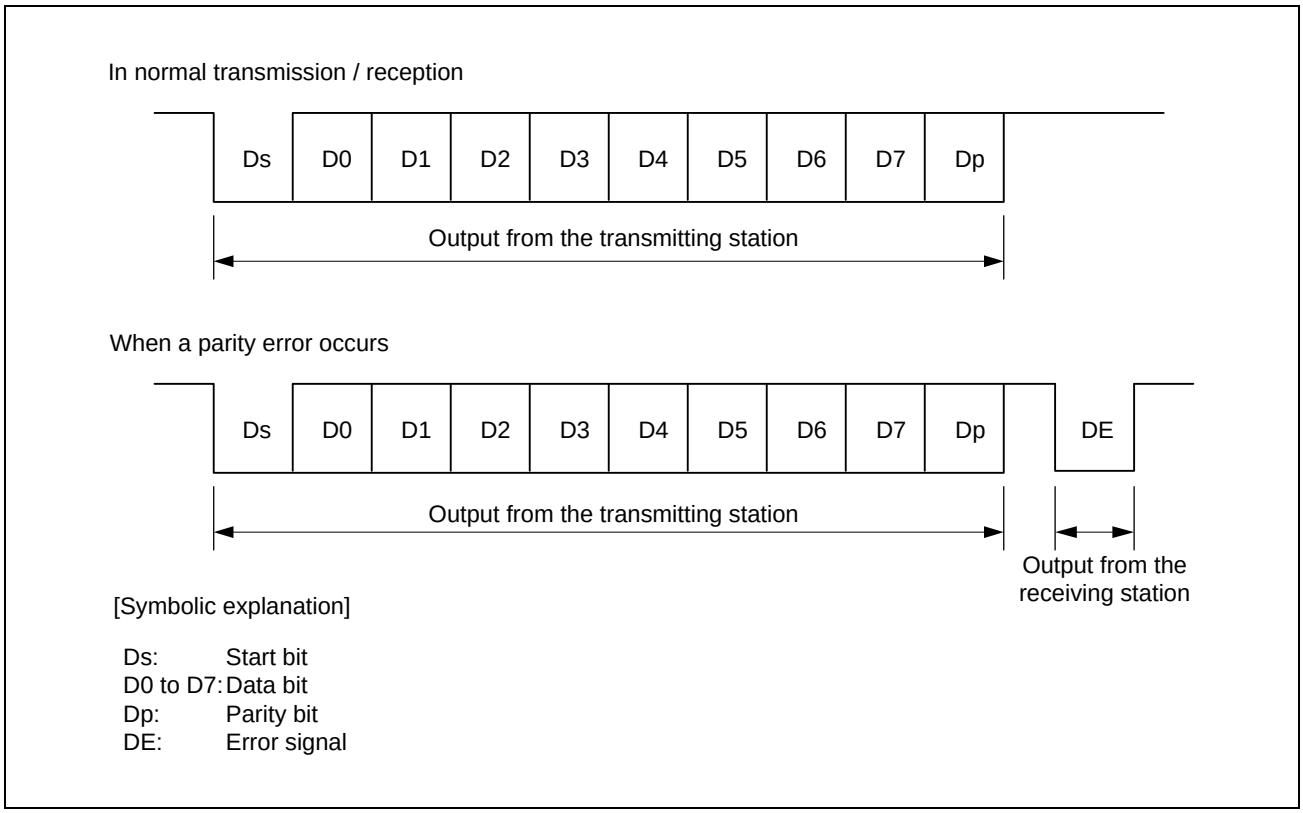


Figure 7.3-43 Data Formats in Smart Card Interface Mode

For communications with IC cards of the direct convention type and inverse convention type, follow the procedure below.

(1) Direct Convention Type

For the direct convention type, logic levels 1 and 0 correspond to states Z and A, respectively, and data is transferred with LSB first as the start character, as shown in **Figure 7.3-44**. Therefore, data in the start character in the figure is 3Bh.

When using the direct convention type, write 0b to the CCR3.SINV bit and write 1b to the CCR3.LSBF bit. Write 0b to CCR1.PM in order to use even parity, which is prescribed by the smart card standard.

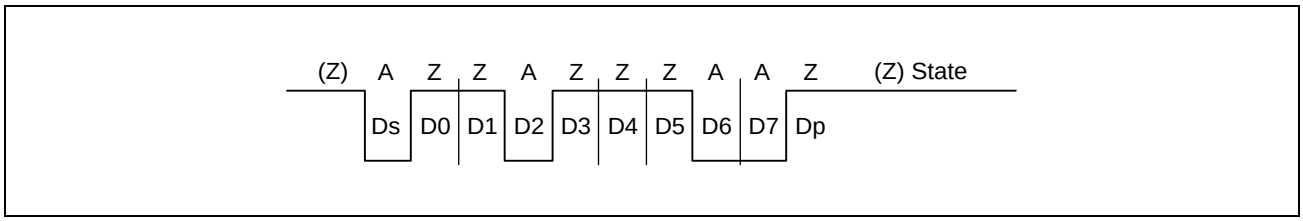


Figure 7.3-44 Direct Convention Type (CCR3.LSBF = 1b, CCR3.SINV = 0b, CCR1.PM = 0b)

(2) Inverse Convention Type

For the inverse convention type, logic levels 1 and 0 correspond to states A and Z, respectively and data is transferred with MSB first as the start character, as shown in **Figure 7.3-45**. Therefore, data in the start character in the figure is 3Fh.

When using the inverse convention type, write 1b to the CCR3.SINV bit and write 0b to the CCR3.LSBF bit. The parity bit is logic level 0 to produce even parity, which is prescribed by the smart card standard, and corresponds to state Z. Since the SINV bit of this LSI only inverts data bits D7 to D0, write 1b to CCR1.PM to invert the parity bit for both transmission and reception.

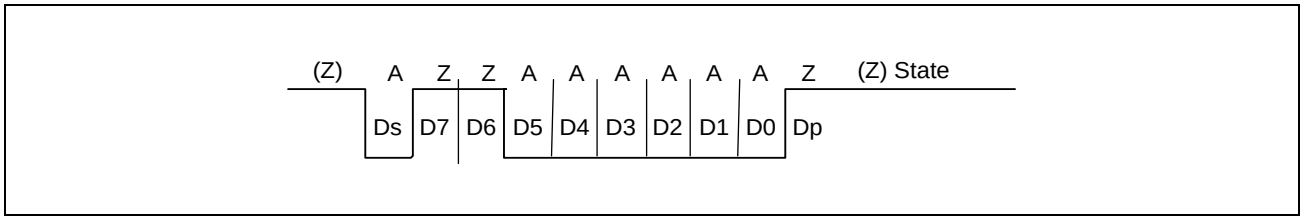


Figure 7.3-45 Inverse Convention Type (CCR3.LSBF = 0b, CCR3.SINV = 1b, CCR1.PM = 1b)

7.3.5.3 Block Transfer Mode

Block transfer mode is different from non-block-transfer-mode of smart card interface mode in the following respects.

- Even if a parity error is detected during reception, no error signal is output. Since CSR.PER is set by error detection, clear the PER bit before receiving the parity bit of the next frame.
- During transmission, at least 1 etu is secured as a guard time from the end of the parity bit until the start of the next frame.
- Since the same data is not retransmitted during transmission, CSR.TEND is set for 11.5 etu after transmission start.
- In block transfer mode, CSR.ERS flag indicates the error signal status as in non-block-transfer-mode of smart card interface mode, but the flag is always read as 0b because no error signal is transferred.

7.3.5.4 Receive Data Sampling Timing and Reception Margin

Only the internal clock generated by the baud rate generator can be used as a transfer clock in smart card interface mode. In this mode, the SCI can operate on a base clock with a frequency of 32, 64, 372, 256, 93, 128, 186, or 512 times the bit rate according to the settings of CCR2.BCP[2:0]. For data reception, the falling edge of the start bit is sampled with the base clock to perform synchronization. Receive data is sampled on the 16th, 32nd, 186th, 128th, 46th, 64th, 93rd, and 256th rising edges of the base clock so that it can be latched at the middle of each bit as shown in **Figure 7.3-46**. The reception margin here is determined by the following formula.

$$M = \left| \left(0.5 - \frac{1}{2N}\right) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\%$$

M: Reception margin (%)
N: Ratio of bit rate to clock (N = 32, 64, 372, 256)
D: Duty cycle of clock (D = 0 to 1.0)

L: Frame length (L = 10)
F: Absolute value of clock frequency deviation

Assuming values of F = 0, D = 0.5, and N = 372 in the above formula, the reception margin is determined by the formula below.

$$M = \left\{\frac{0.5 - 1}{2 \times 372}\right\} \times 100\% = 49.866\%$$

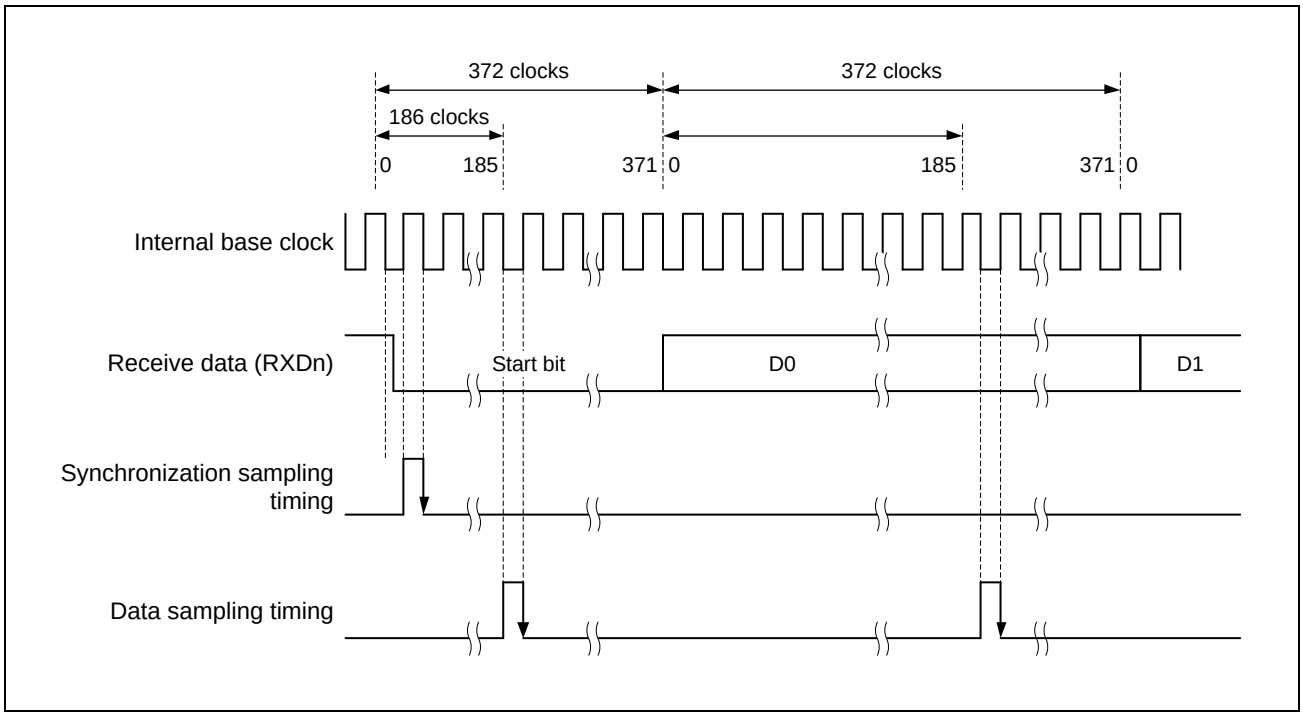


Figure 7.3-46 Receive Data Sampling Timing in Smart Card Interface Mode (when Clock Frequency is 372 Times the Bit Rate)

7.3.5.5 Initialization of SCI (Smart Card Interface Mode)

Before transmitting and receiving data, write 0b to CCR0.TE and CCR0.RE (Or write the initial value to CCR0). And initialize SCI following example flowchart in **Figure 7.3-47**.

Be sure to set the initial value in the TIE, RIE, TE, RE, and TEIE bits in CCR0 register before switching from transmission mode to reception mode and vice versa. Even if the RE bit is set to 0b, the RDR register is not initialized. In transmission mode, set 1b to the TE bit and TIE bit simultaneously, then the TXI interrupt request is generated. To change reception mode to transmission mode, first check that reception has completed, and then initialize SCI. At the end of initialization, set TE = 1b and RE = 0b. Reception completion can be verified by RXI request, CSR.ORER, or CSR.PER. To change transmission mode to reception mode, first check that transmission has completed, and then initialize SCI. At the end of initialization, set TE = 0b and RE = 1b. Transmission completion can be verified by reading CSR.TEND.

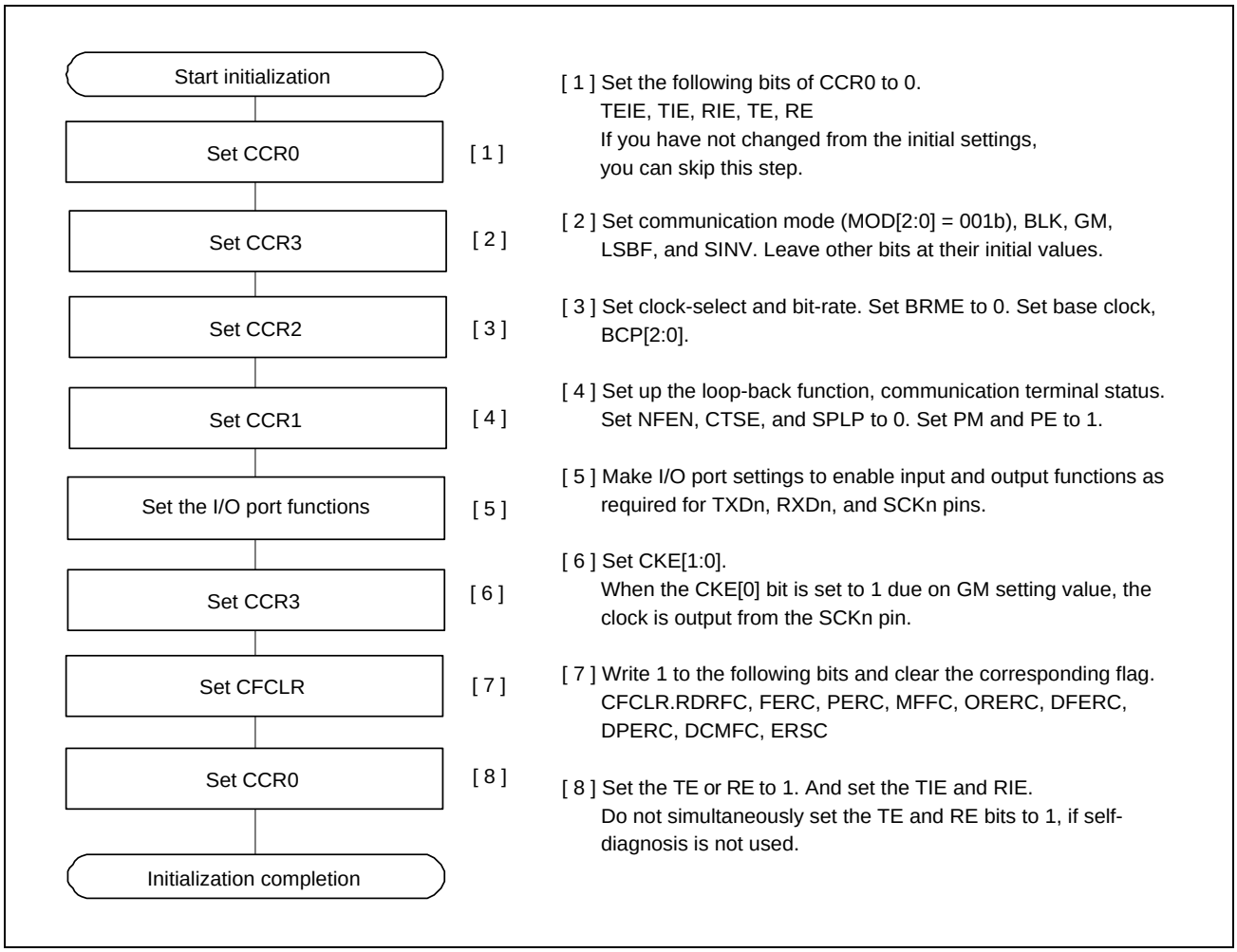


Figure 7.3-47 Example of SCI Initialization Flowchart (Smart Card Interface Mode)

Figure 7.3-48 is a timing chart when data transmission is performed by making transition to the Smart Card Interface mode according to the above flow chart. The figure shows the case when CCR3.GM bit is 0b. As shown in the figure, when the pin function is set to the SCK pin, the SCK pin is high impedance because the CCR3.CKE[0] bit is 0b. When the TXD pin is set, the TXD pin is high impedance because the CCR0.TE bit is 0b. Start clock output to the SCK pin with the clock output setting CCR3.CKE[0] to 1b, start data transmission by writing transmit data after setting CCR0.TE to 1b.

In the smart card interface mode, even if not communicating at CCR0.TE = 0b and CCR0.RE = 0b, the clock is continuously output if the clock output setting is used.

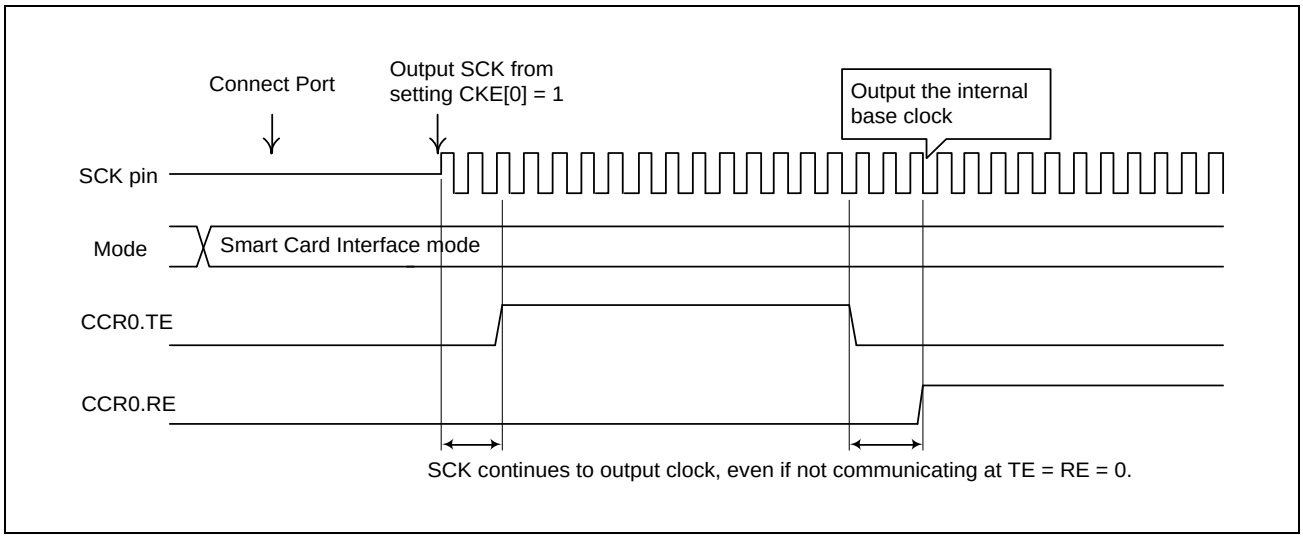
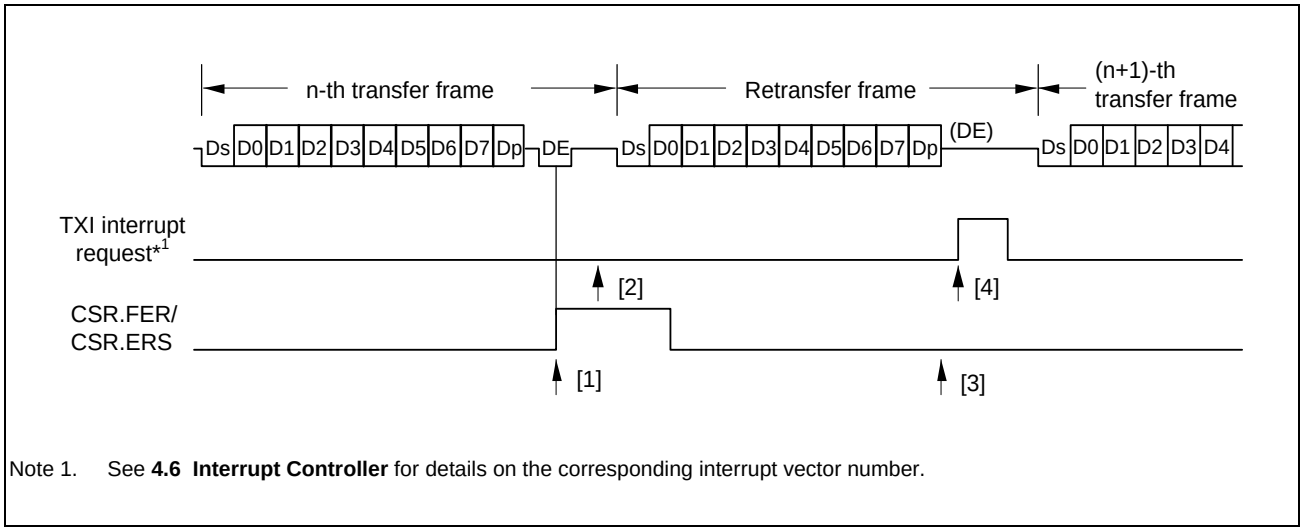


Figure 7.3-48 Example of Timing Chart of Data Transmission (Smart Card Interface Mode)

7.3.5.6 Serial Data Transmission (Except in Block Transfer Mode)

Serial data transmission in smart card interface mode (except in block transfer mode), in that an error signal is sampled and data can be retransmitted, is different from that in non-smart card interface mode. **Figure 7.3-49** shows the data retransfer operation during transmission.

1. When an error signal from the receiver end is sampled after one-frame data has been transmitted, CSR.ERS is set to 1b. If CCR0.RIE is 1b at this time, an ERI interrupt request is generated. Clear the ERS flag to 0b before the next parity bit is sampled.
2. For a frame in which an error signal is received, CSR.TEND is not set. Data is retransferred from TDR to TSR allowing data retransmission automatically.
3. If no error signal is returned from the receiver, the ERS flag is not set to 1b.
4. In this case, the SCI judges that transmission of one-frame data (including retransfer) has been completed, and the TEND flag is set. If CCR0.TIE is 1b at this time, a TXI interrupt request is generated. Writing transmit data to TDR starts transmission of the next data.



Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-49 Data Retransfer Operation in SCI Transmission Mode

Figure 7.3-51 shows a sample flowchart of serial transmission. All the processing steps are automatically performed by using a TXI interrupt request to activate the DMAC. When CSR.TEND is set to 1b in transmission, if the CCR0.TIE is 1b, a TXI interrupt request is generated. The DMAC is activated by a TXI interrupt request if the TXI interrupt request is specified as a source of DMAC activation beforehand, allowing transfer of transmit data. The TEND flag is automatically set to 0b when the DMAC transfers the data. If an error occurs, the SCI automatically retransmits the same data. During this retransmission, the TEND flag is kept to 0b and the DMAC is not activated. Therefore, the SCI and DMAC automatically transmit the specified number of bytes, including retransmission in the case of error occurrence. However, since the ERS flag is not automatically cleared, set the RIE bit to 1b beforehand to enable an ERI interrupt request to be generated at error occurrence, and clear the ERS flag to 0b.

When transmitting/receiving data using the DMAC, be sure to make settings to enable the DMAC before making SCI settings. For DMAC settings, see **4.7 DMA Controller (DMAC)**.

Note that CSR.TEND flag is set in different timings depending on the CCR3.GM bit setting. **Figure 7.3-50** shows the TEND flag generation timing.

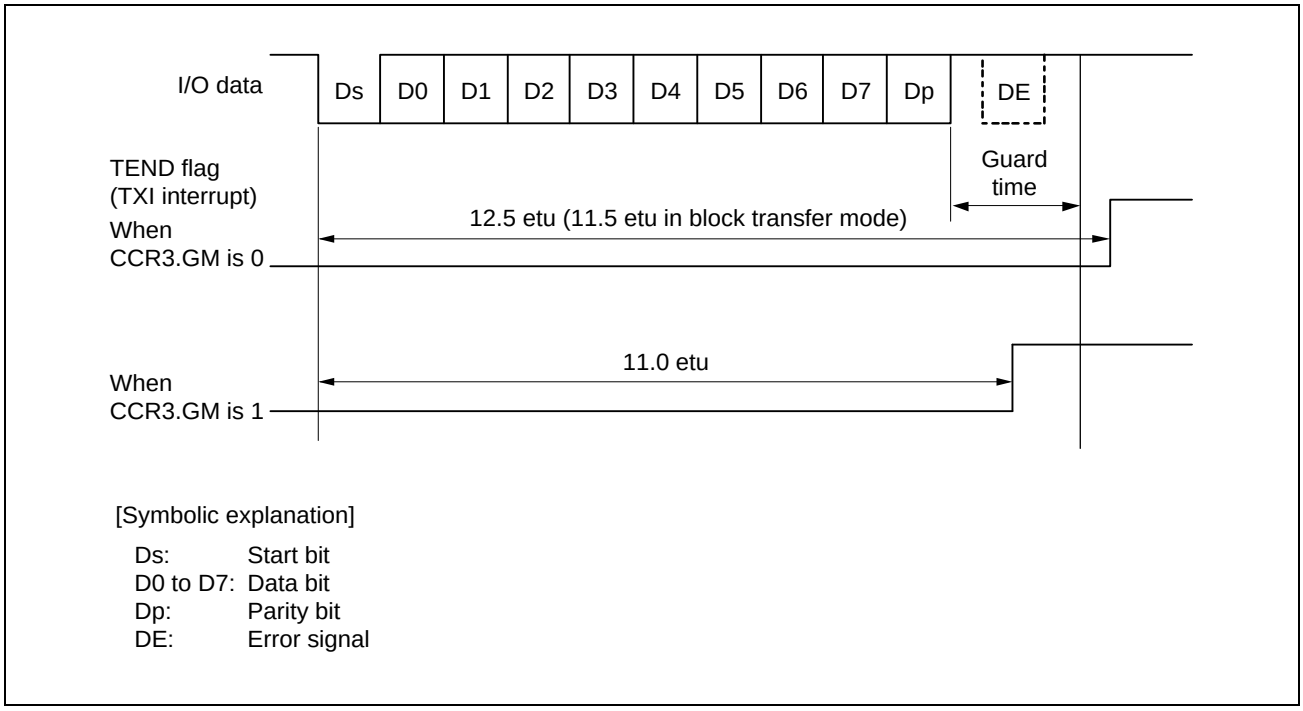


Figure 7.3-50 CSR.TEND Flag Generation Timing during Transmission

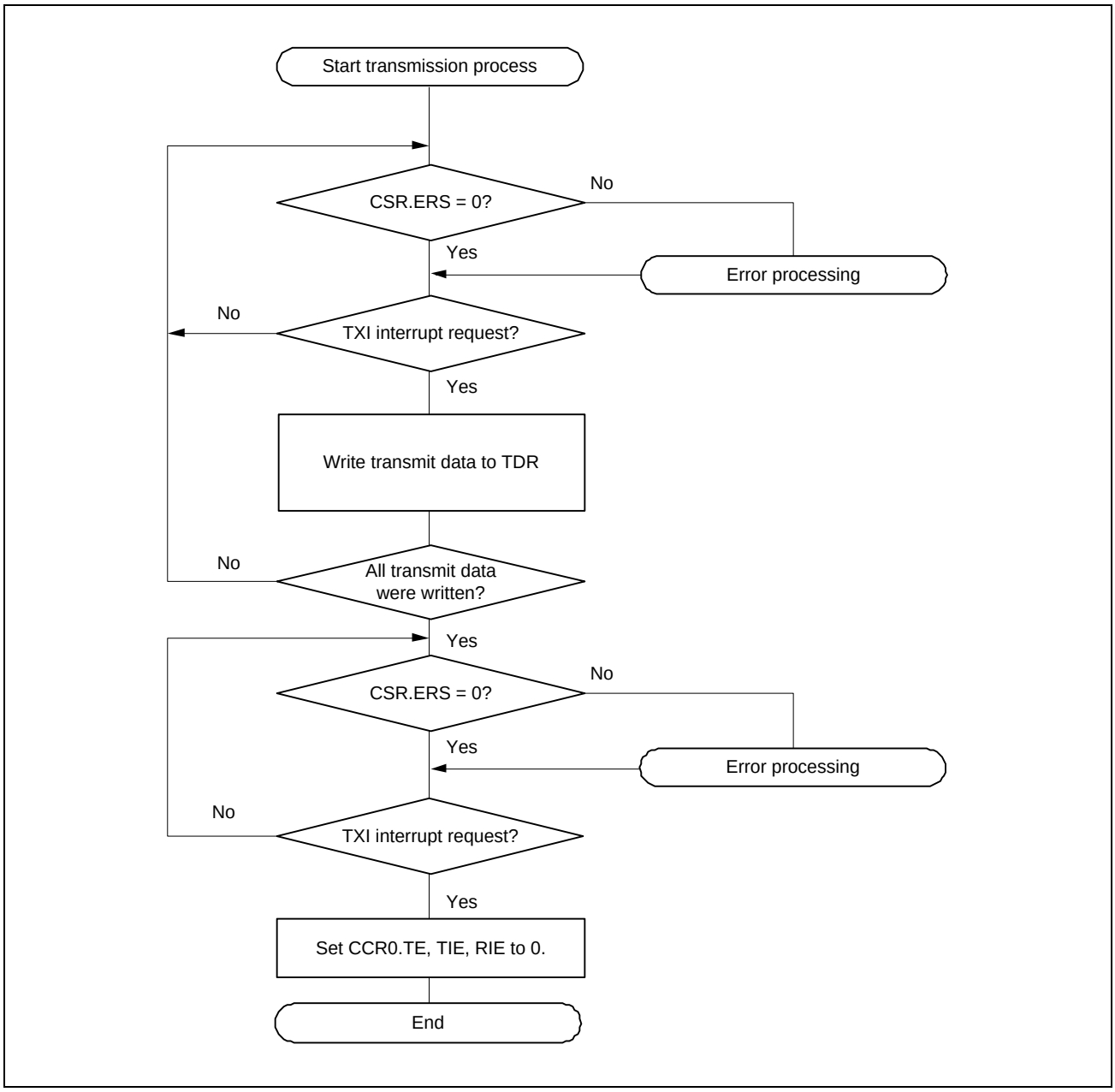
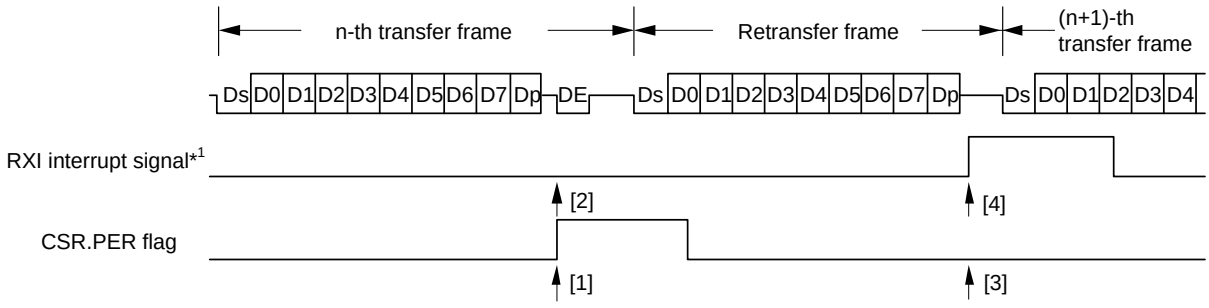


Figure 7.3-51 Sample Smart Card Interface Transmission Flowchart

7.3.5.7 Serial Data Reception (Except in Block Transfer Mode)

Serial data reception in smart card interface mode is like that in non-smart card interface mode. **Figure 7.3-52** shows the data retransfer operation in reception mode.

1. If a parity error is detected in receive data, the CSR.PER flag is set to 1b. When the CCR0.RIE is 1b at this time, an ERI interrupt request is generated. Clear the PER flag to 0b before the next parity bit is sampled.
2. For a frame in which a parity error is detected, no RXI interrupt is generated.
3. When no parity error is detected, the CSR.PER flag is not set to 1b.
4. In this case, data is determined to have been received successfully. When the CCR0.RIE is 1b, an RXI interrupt request is generated.



Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-52 Data Retransfer Operation in SCI Reception Mode (Data Retransfer Operation during Reception)

Figure 7.3-53 shows a sample flowchart for serial data reception. All the processing steps are automatically performed using an RXI interrupt request to activate the DMAC. In reception, setting the RIE bit to 1b allows an RXI interrupt request to be generated. The DMAC is activated by an RXI interrupt request if the RXI interrupt request is specified as a source of DMAC activation beforehand, allowing transfer of receive data. If an error occurs during reception and either the CSR.ORER or PER flag is set to 1b, a receive error interrupt (ERI) request is generated. Clear the error flag after the error occurrence. If an error occurs, the DMAC is not activated and receive data is skipped. Therefore, the number of bytes of receive data specified in the DMAC is transferred.

Even if a parity error occurs and the PER flag is set to 1b during reception, receive data is transferred to RDR, thus allowing the data to be read. When a reception is forcibly terminated by setting the CCR0.RE to 0b during operation, read the RDR register because the received data which has not yet been read may be left in RDR.

NOTE

For operations in block transfer mode, see **7.3.3 Asynchronous Mode**.

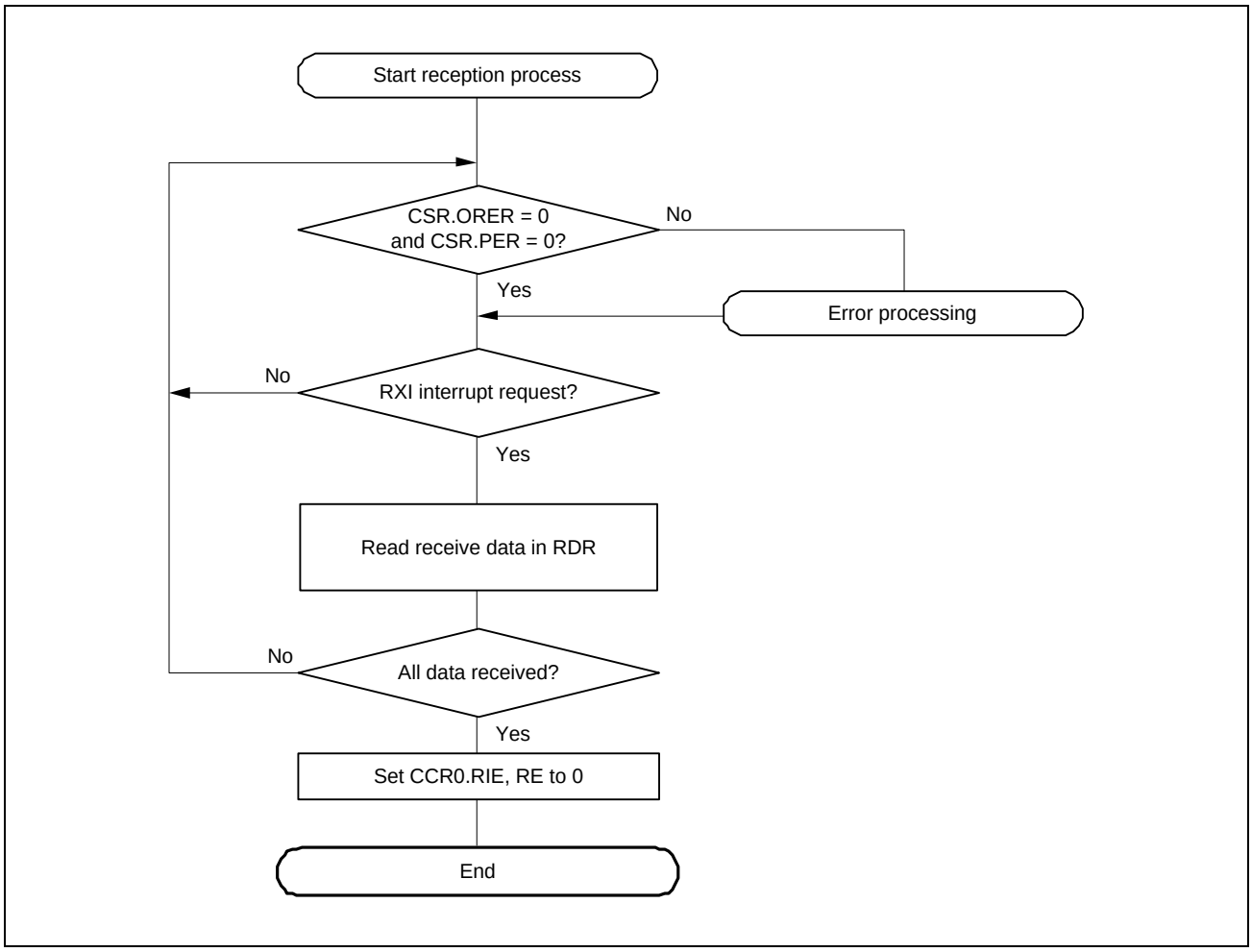


Figure 7.3-53 Sample Smart Card Interface Reception Flowchart

7.3.5.8 Clock Output Control

When the CCR3.GM is set to 1b, the clock output can be controlled by the CCR3.CKE[1:0]. Refer to the description of the CKE in **7.3.2.2.8 Common Control Register 3 (RSCIm_CCR3)**. When setting the clock output, the base clock described in **7.3.3.3 Clock** is output, so the width of the clock pulse can be kept to the width specified by setting the bit rate. It is described in **7.3.2.2.7 Common Control Register 2 (RSCIm_CCR2)**, the bit rate is set by CCR2.CKS[1:0], CCR2.BCP[2:0], and BRR[7:0].

Figure 7.3-54 shows the timing chart for explaining clock output control. This is an example when the CCR3.CKE[1] is set to 0b and the CCR3.CKE[0] is controlled.

When the CCR3.GM is 0b, output control by the CCR3.CKE[0] is immediately reflected on the SCK pin, so there is a possibility that pulses with an unintended width may be output from the SCK pin.

When the CCR3.GM is 1b, the output pulse control by the CCR3.CKE[0] controls the pulse width set to be based on the state of the base clock.

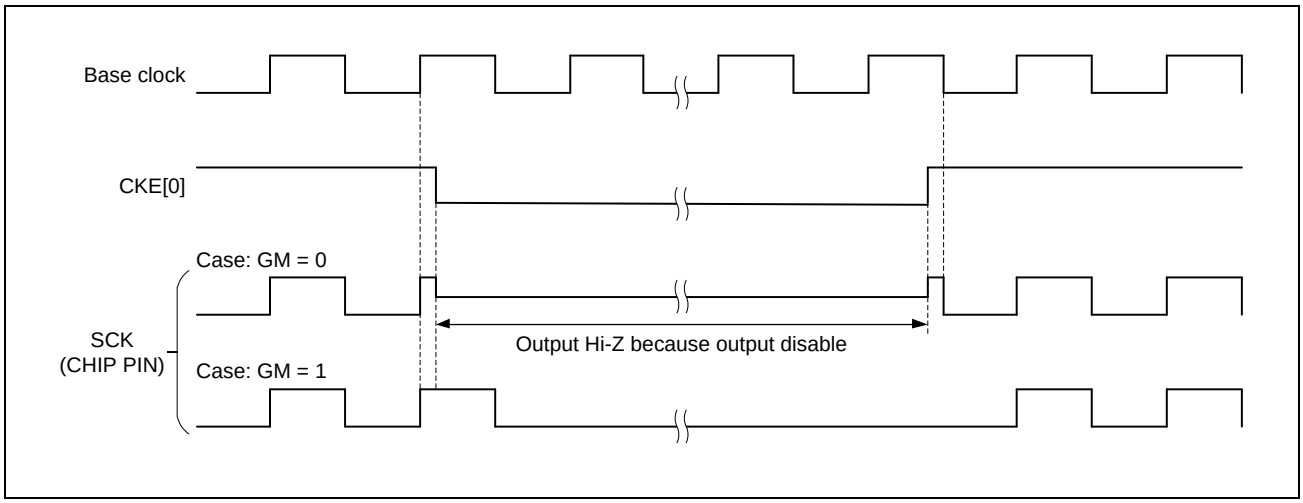


Figure 7.3-54 Clock Control Timing Chart by CCR3.GM

7.3.6 Simple-LIN Mode

As an extended function of the RSCI, the RSCI supports the serial communication protocol (**Figure 7.3-55**) consisting of a Start Frame and an Information Frame as Simple-LIN. Simple-LIN mode is enabled by setting the CCR3.MOD[2:0] = 110b. Since the simple-LIN mode uses the same circuit as the asynchronous mode for transmission/reception control other than Break Field, the basic communication settings required for the asynchronous mode are also required for simple-LIN mode. (For the setting value when using Simple-LIN mode, see the explanation of **7.3.2 SCI Registers**. In particular, note that CCR3.RXDESEL needs to be changed from the initial value and set to 1b.) The Start Frame consists of a Break Field, Control Field 0, and Control Field 1. The Information Frame can be configured with some Data Fields, a CRC16 Upper Field, and a CRC16 Lower Field.

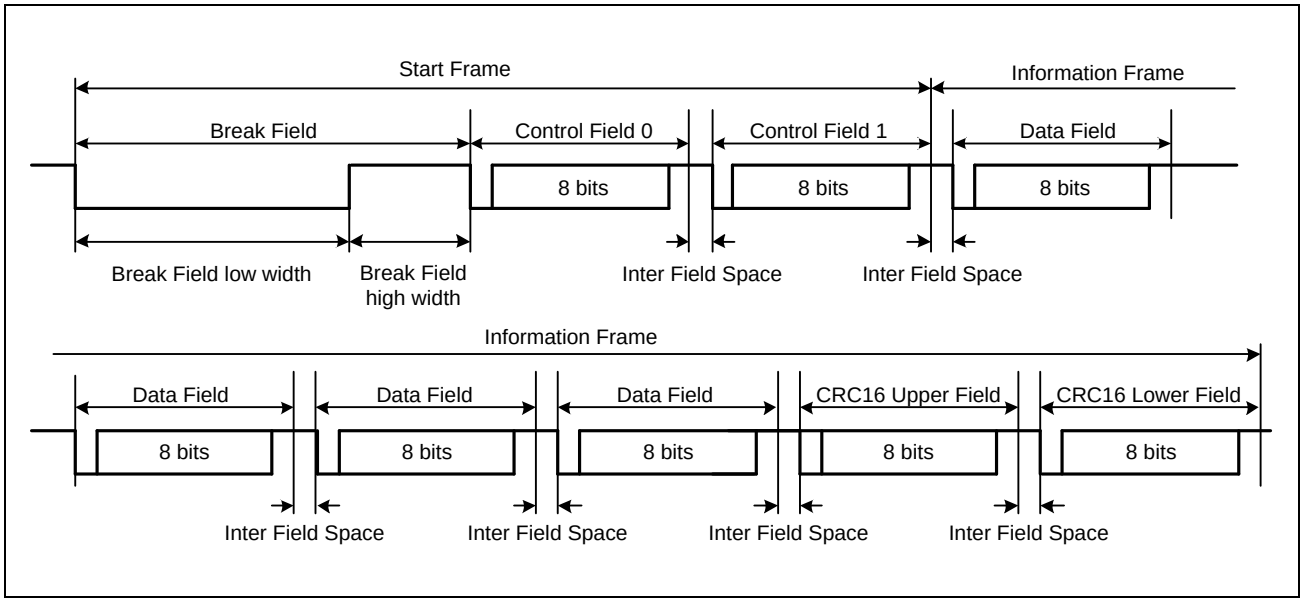


Figure 7.3-55 Clock Control Timing Chart by CCR3.GM

The following describes operations when the simple-LIN is used. In this section, operations are described with the following conditions: Communication pin (RXD/TXD) level inversion function: OFF (RINV = TINV = 0b). When using the simple LIN with the communication pin (RXD/TXD) level inversion function enabled, replace the RXD and TXD signal levels with their inverted levels.

7.3.6.1 Simple-LIN Start Frame Transmission

Figure 7.3-56 shows an example of transmission of the Start Frame consisting of a Break Field, Control Field 0, and Control Filed 1. (Omit Break Field and Control Field 0 according to the Start Frame configuration.)

Figure 7.3-57 shows a flowchart for Start Frame transmission.

The RSCI operates as follows during Start Frame transmission.

1. Make the initial settings for the RSCI according to the RSCI initialization flow (**Figure 7.3-8**) in asynchronous mode. In simple-LIN mode, do not set CCR0.TE and TIE to 1b at the same time to avoid TXI output before the Break Field. Therefore, perform the following two steps sequentially to set the SCI initialization flow (asynchronous mode) procedure [9].
   - Set the bits except CCR0.TIE. (CCR0.TIE = 0b, CCR0.TE = 1b, and CCR0.RE = 0b)
   - Set CCR0.TIE to 1b.
2. When 1b is written to TCST, the SCIX timer starts counting and outputs a low level (Break Field) from the TXD pin for the period set in XCR2.BFLW[15:0]. A timer count clock source can be selected by XCR0.TCSS[1:0]. Writing 0b to XCR1.TCST suspends output of the Break Field. After the suspension, set CCR0.TE = 0b and turn off the transmission.
3. When the simple-LIN module timer count value matches the set XCR2.BFLW[15:0] value, the timer stops counting and inverts the TXD pin output level, and the XSR0.BFOF flag is set to 1b*[1]. Furthermore, if XCR0.BFOIE has been set to 1b at this time, a TXI interrupt is generated.
4. After confirming XSR0.BFOF=1b after TXI interrupt, write transmission data and transmit Control Field 0 data.

Remark LIN communication requires a Break delimiter (IDLE period) of 1 bit or more from the end of Break Field output until the next data transmission starts. For this reason, the Break delimiter length is counted upon completion of Break Field output. If transmit data is written while the Break delimiter length is being counted, transmission does not start until the Break delimiter length counting is completed. When transmit data is written after the Break delimiter length has been counted, transmission starts at the same timing as normal data transmission (bit boundary).
Break delimiter length count time after Break Field output:
- 1-bit to 2-bit length (CCR3.STP = 0b)
- 2-bit to 3-bit length (CCR3.STP = 1b)

5. After the Control Field 0 data has been transmitted, write the Control Field 1 data to TDR. And it is transmitted.
6. After the Control Field 1 data has been transmitted, the Information Frame data is transmitted.

Note 1. After XSR0.BFOF is set to 1b, if 1b is written to XCR1.TCST without clearing it, no TXI interrupt is output at the end of Break Field transmission. Clear XSR0.BFOF before writing 1b to XCR1.TCST.

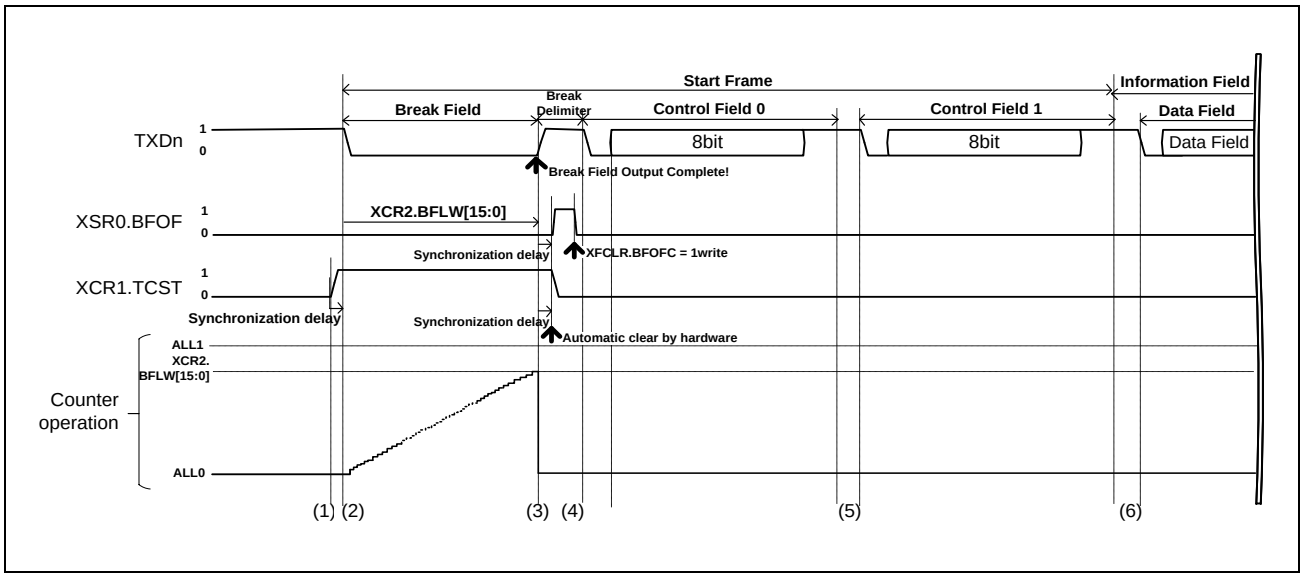


Figure 7.3-56 Start Frame Transmission Example

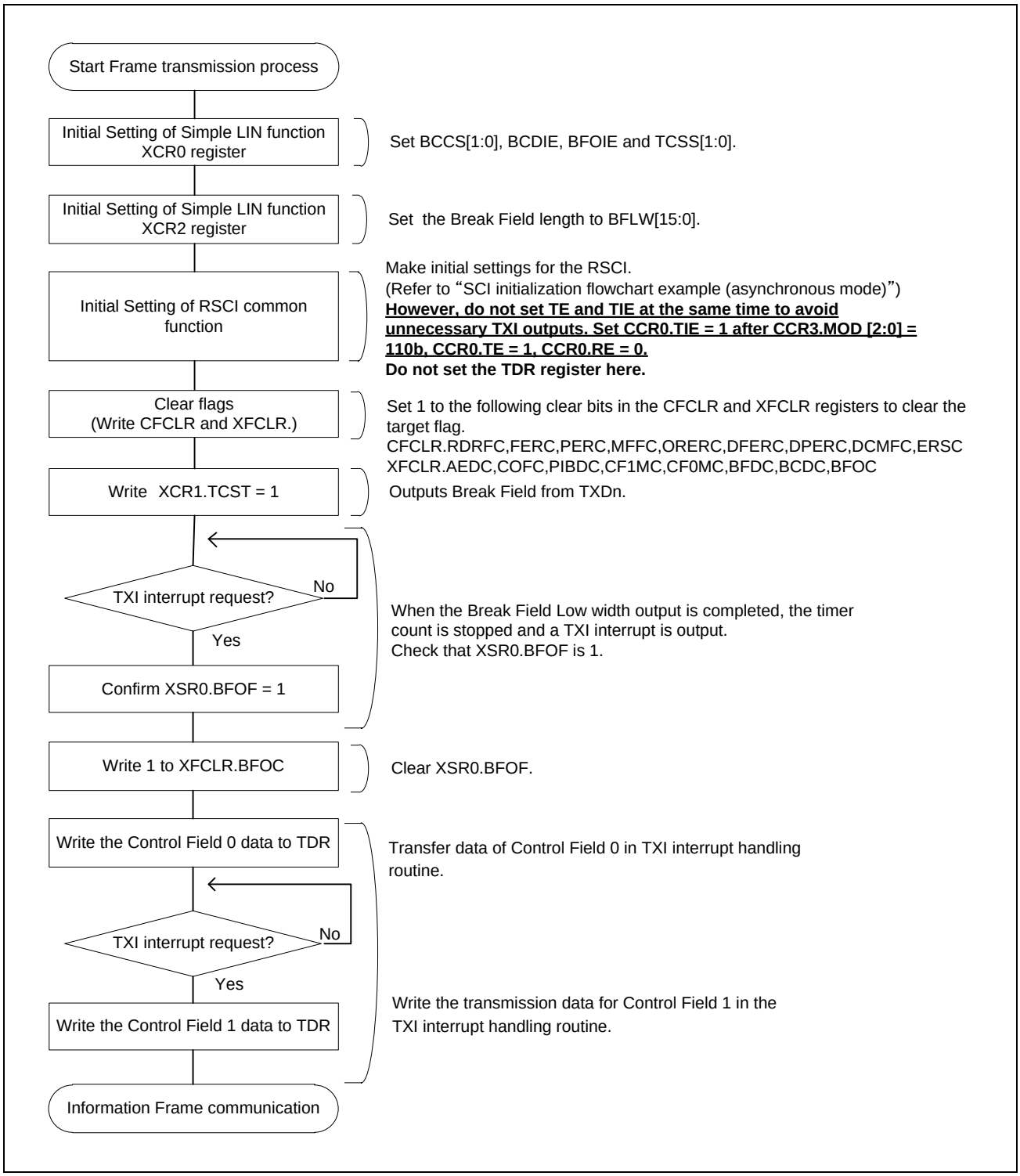
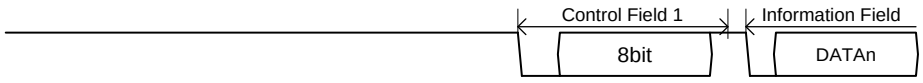
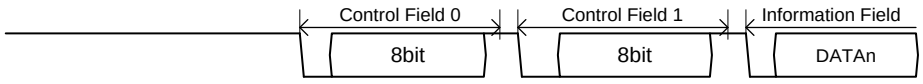
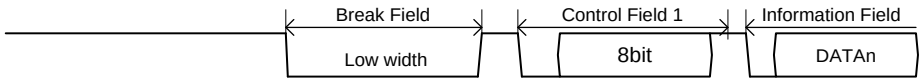
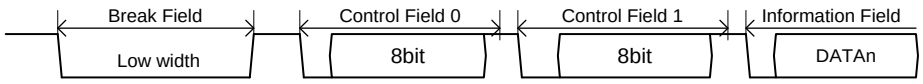


Figure 7.3-57 Start Frame Transmission Flowchart Example

7.3.6.2 Simple-LIN Start Frame Reception

The RSCI can detect Start Frames configured as shown in **Table 7.3-29**.

Table 7.3-29 Start Frame Configuration

XCR0		Start Frame Configuration
BFE	CF0RE	
0	0	
0	1	
1	0	
1	1	

7.3.6.2.1 Simple-LIN Normal Reception of Start Frame (PIB Not Used)

Figure 7.3-58 shows an example of normal reception of the Start Frame consisting of a Break Field, Control Field 0, and Control Field 1. **Figure 7.3-59** shows an example of reception to detect the Break Field during Control Field 1. **Figure 7.3-60** shows a flowchart to receive the Start Frame, and **Figure 7.3-61** shows a state transition diagram.

When receiving the Start Frame, the RSCI operates as follows. Omit the processing of Break Field and Control Field 0 according to the Start Frame configuration.

(1) Writing 1b to XCR1.SDST makes it possible to detect the Start Frame. When XCR0.BFE = 1b, RXD input to the SCI core is disabled until the Break Field is detected (because XSR0.RXDSF is set to 1b). Once the Break Field is detected, RXD input can be received by the SCI core (XSR0.RXDSF = 0b).

(2) When a low level is input from the RXD pin, the Break Field detection count starts. A timer count clock source can be selected by XCR0.TCSS[1:0].

(3) When a low level that is equal to or longer than the period set in XCR2.BFLW[15:0] is input from the RXD pin, it is determined as Break Field. At this time, XSR0.BFDF is set to 1b. If XCR0.BFDIE has been set to 1b at this time, a BFD interrupt is generated.
The timer continues counting until the RXD rising edge or counter overflow.

(4) After the Break Field is detected, when the input level from the RXD pin becomes high, the count value is captured to XSR1.TCNT[15:0] when BMEN = 0b. At this time, XSR0.RXDSF is cleared to 0b and the SCI core starts receiving the RXD input.

(5) The SCI core starts receiving Control Field 0. Because the simple LIN continuously counts the edge interval, it determines a low level that is equal to or longer than the period set in XCR2.BFLW[15:0] as detection of the Break Field. When the Break Field is detected in the Control Field 0 phase, the SCI core waits for reception of Control Field 0 again (**Figure 7.3-59**).

(6) When Control Field 0 has been received, an RXI interrupt is generated and the Control Field 0 data is stored in XSR0.CF0RD[7:0]. When the received data matches the set XCR2.CF0D[7:0] value, XSR0.CF0MF is set to 1b. If

the received data differs from the set XCR2.CF0D[7:0] value, the SCI transitions to the state before the Break Field is detected.

(7) The SCI core starts receiving Control Field 1. When BFE = 1b, the Break Field detection function is continuously enabled while SDST = 1b as in the case of Control Field 0. When the Break Field is detected in the Control Field 1 phase, the SCI core waits for reception of Control Field 0 again.

(8) When Control Field 1 has been received, an RXI interrupt is generated and the Control Field 1 data is stored in XSR0.CF1RD[7:0]. When the received data matches the set XCR1.PCF1D[7:0] value or the set XCR1.SCF1D[7:0] value, XSR0.CF1MF is set to 1b. If the received Control Field 1 data matches neither the set XCR1.PCF1D[7:0] value nor the set XCR1.SCF1D[7:0] value, the SCI transitions to the state before the Break Field is detected.

(9) The RSCI core performs Information Frame communication.

(10) When communication is completed, write 0b to XCR1.SDST and 0b to CCR0.RE to stop reception.

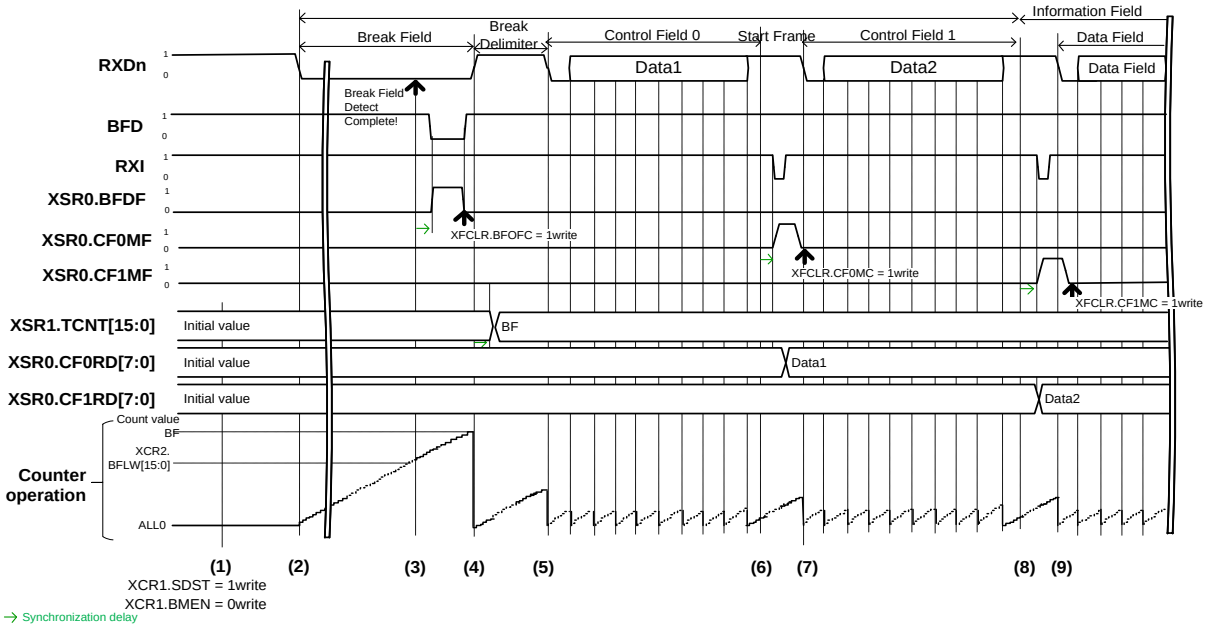


Figure 7.3-58 Normal Reception Example of Start Frame (PIB Not Used)

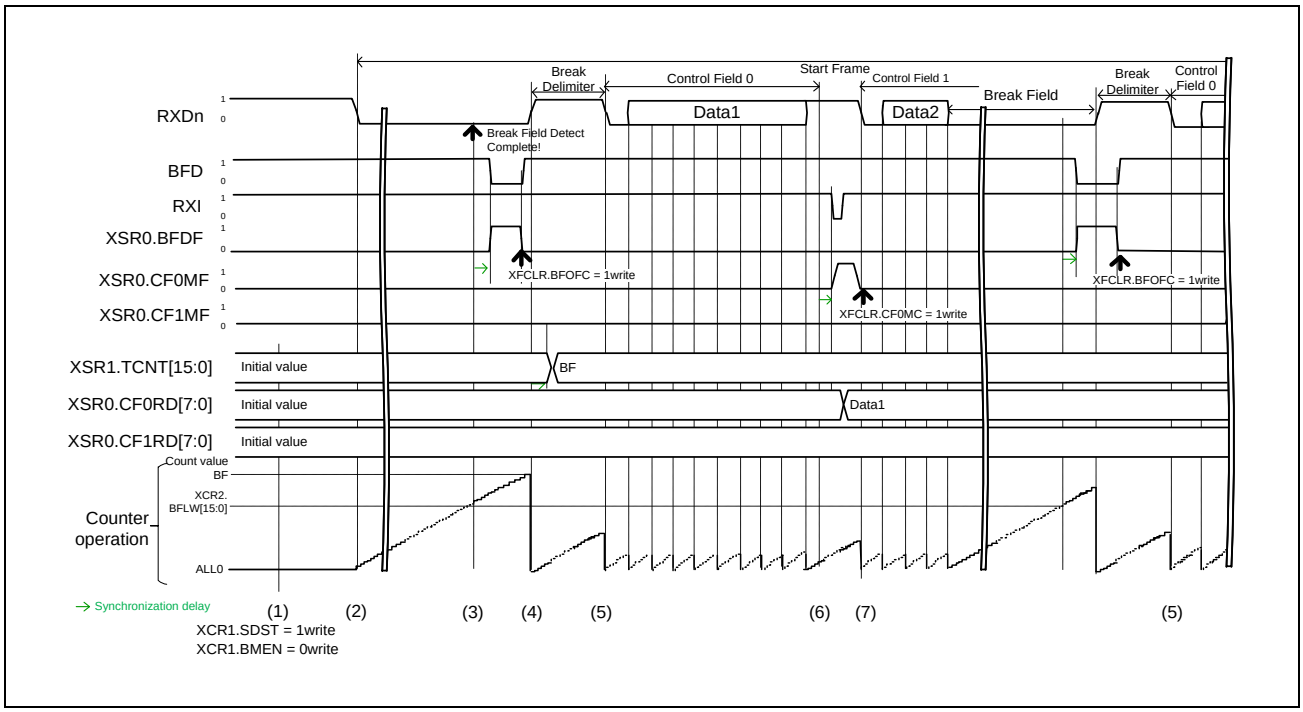


Figure 7.3-59 Start Frame Reception Example (PIB Not Used) Break Field Detected during Control Field 1

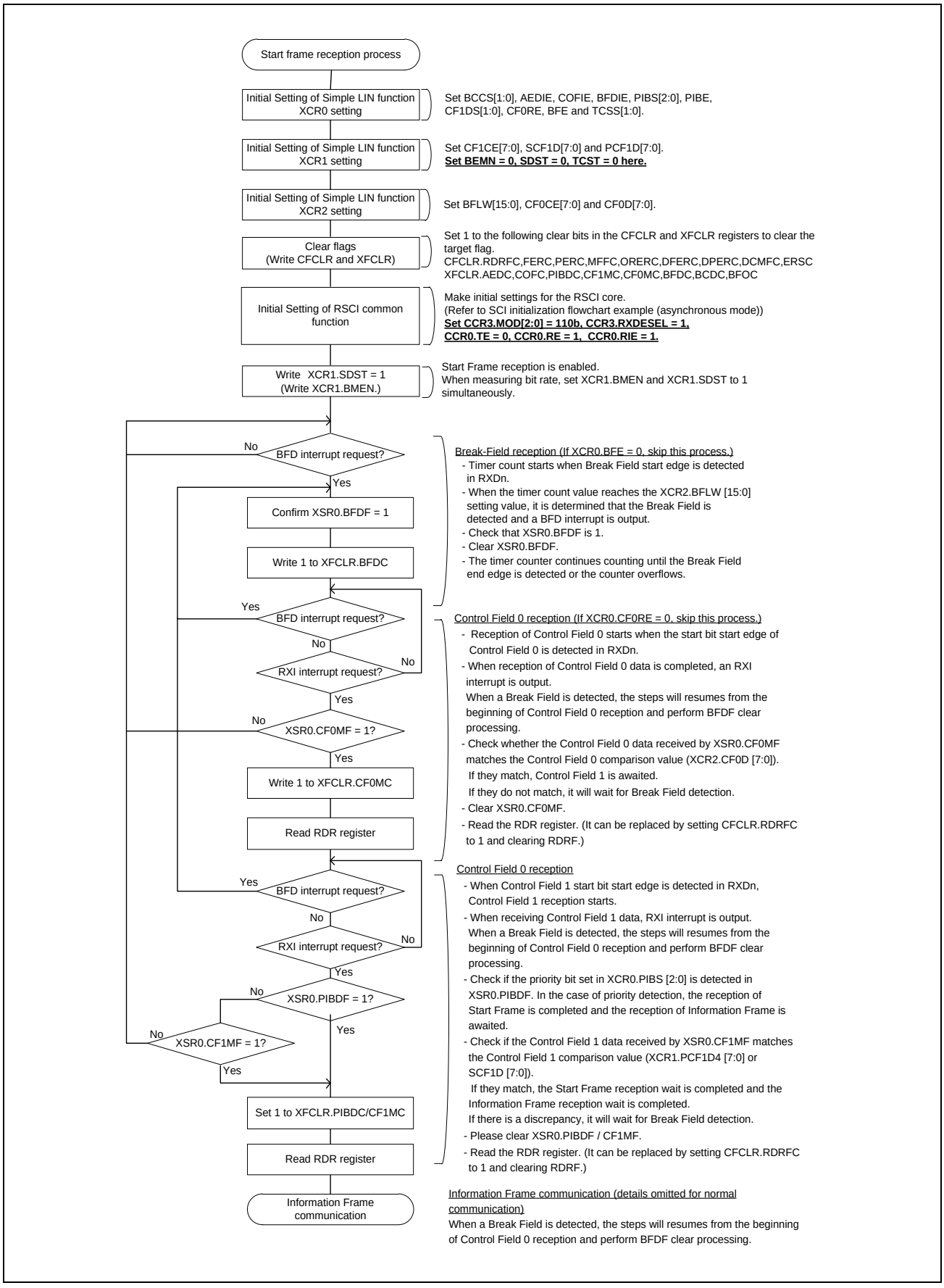


Figure 7.3-60 Example of Start Frame Reception Flowchart

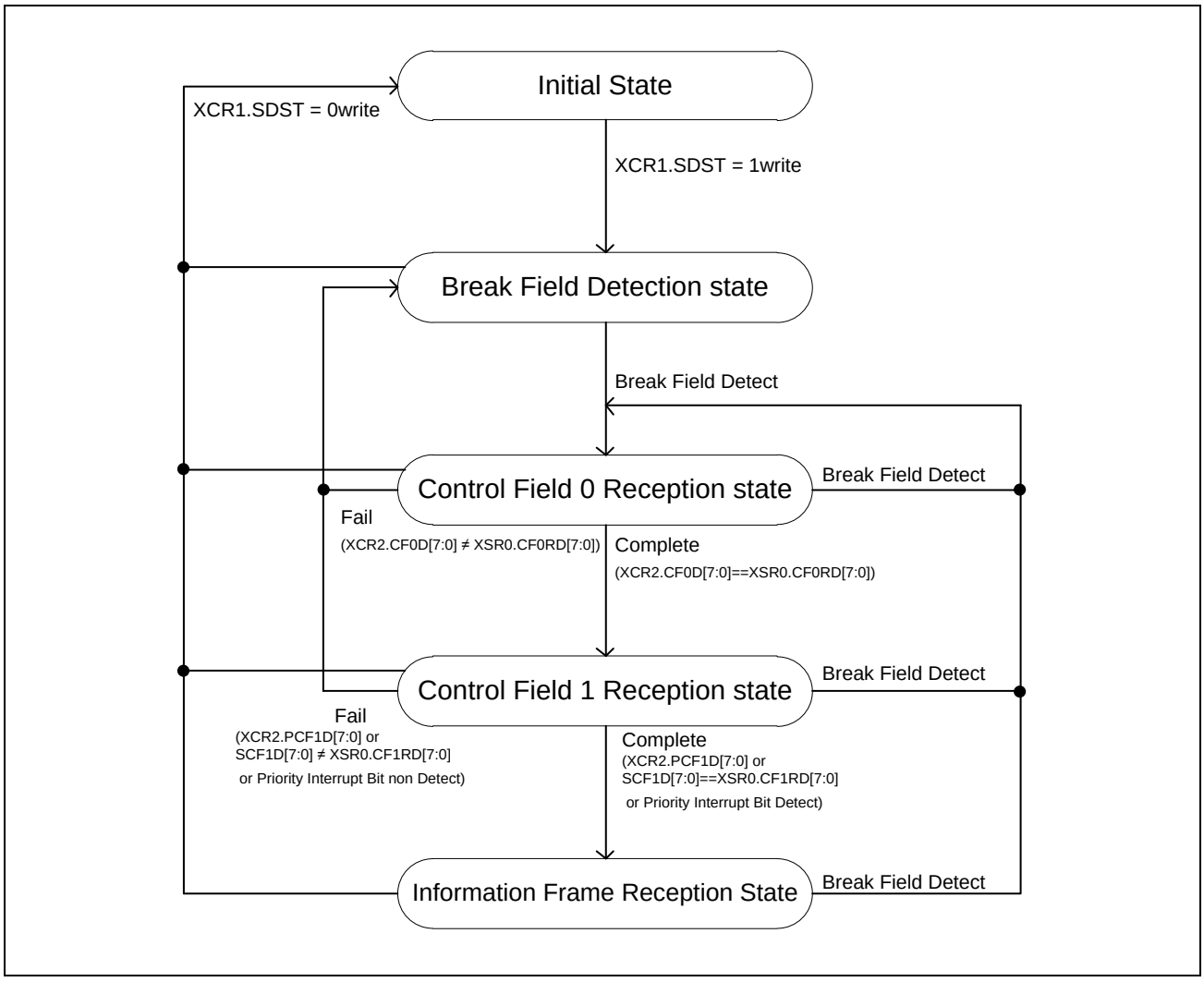


Figure 7.3-61 State Transition Diagram of Start Frame Reception

7.3.6.2.2 Simple-LIN Start Frame reception (using the priority interrupt bit)

Figure 7.3-62 shows an example of Start Frame reception using the priority interrupt bit. The priority interrupt bit is enabled by setting XCR0.PIBE to 1b.

The RSCI operates as follows during Start Frame reception using the priority interrupt bit.

Steps (1) to (7) are the same as steps (1) to (7) in the Start Frame reception example in **Figure 7.3-59**.

(8) When the value specified in the XCR0.PIBS[2:0] bits matches the set XCR1.PCF1D[7:0] value, XSR0.PIBDF is set to 1b and the RSCI core performs communication of the Information Frame. If the data received in Control Field 1 matches neither the set XCR1.PCF1D[7:0] value nor the set XCR1.SCF1D[7:0] value and the priority interrupt bit is not detected, the RSCI transitions to the state before the Break Field is detected.

(9) Communicate information frame at the RSCI core.

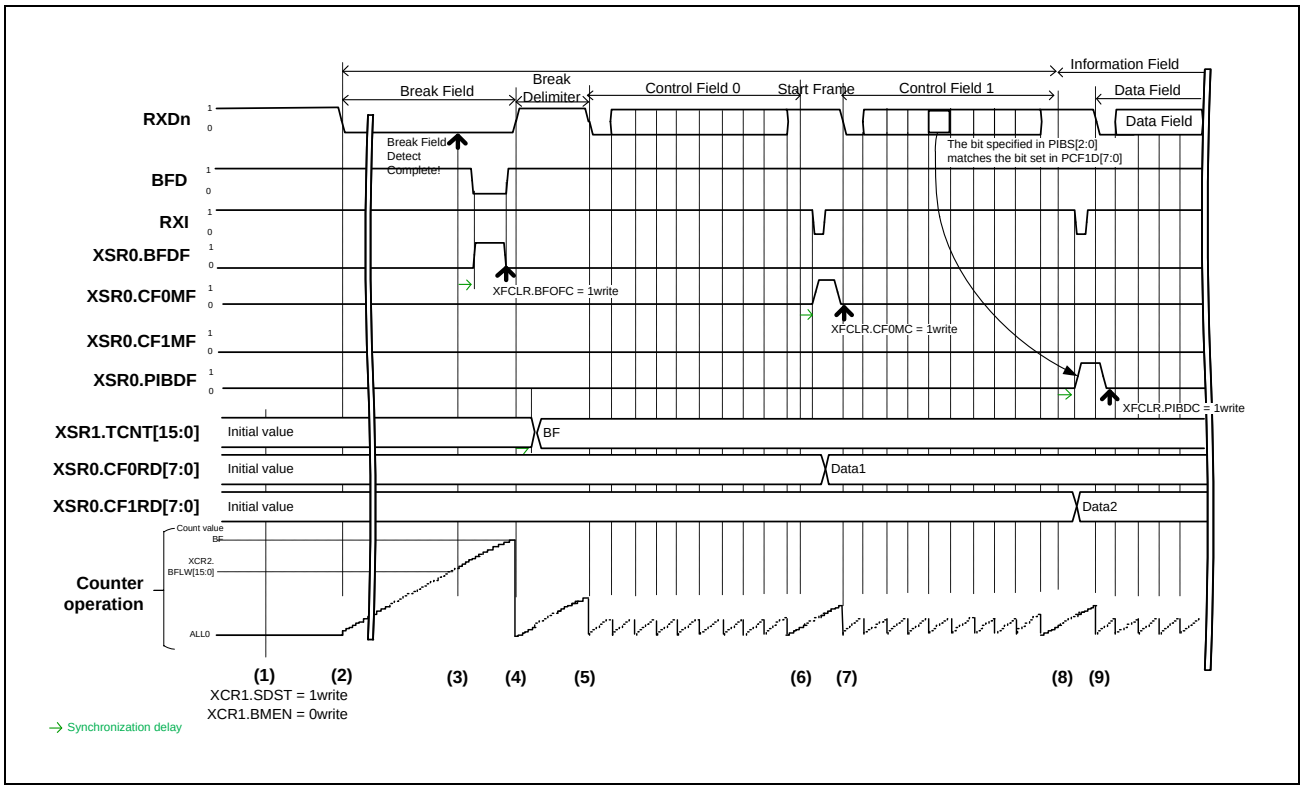


Figure 7.3-62 Start Frame Reception Example (Priority Interrupt Bit Used)

7.3.6.3 Simple-LIN Bus Conflict Detection Function

In simple-LIN mode (CCR3MOD[2:0] = 110b) when TE = 1b, the bus conflict detection function works during Break Field output and during data transmission.

Figure 7.3-63 shows an operation example of the bus conflict detection function. The TXD pin output and the RXD pin input are sampled by the bus conflict detection clock set in XCR0.BCCS[1:0]. When a mismatch occurs three times in a row, XSR0.BCDF is set to 1b, and if XCR0.BCDIE has been set to 1b at this time, an ERI interrupt is generated.

When an ERI interrupt is generated, stop transmission according to **Figure 7.3-64**. Check the bus state to decide whether to resume transmission.

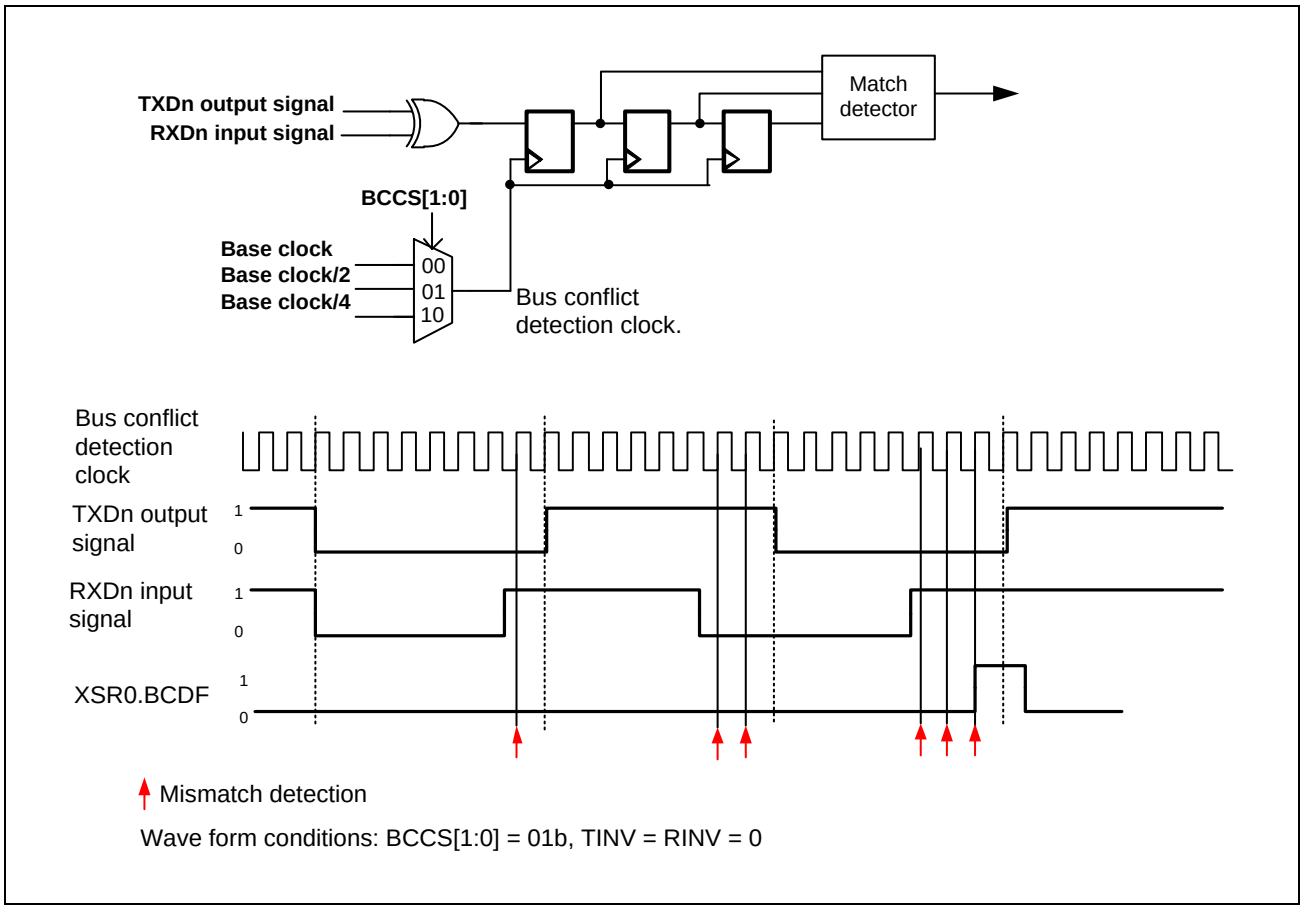


Figure 7.3-63 Operation Example of the Bus Conflict Detection Function

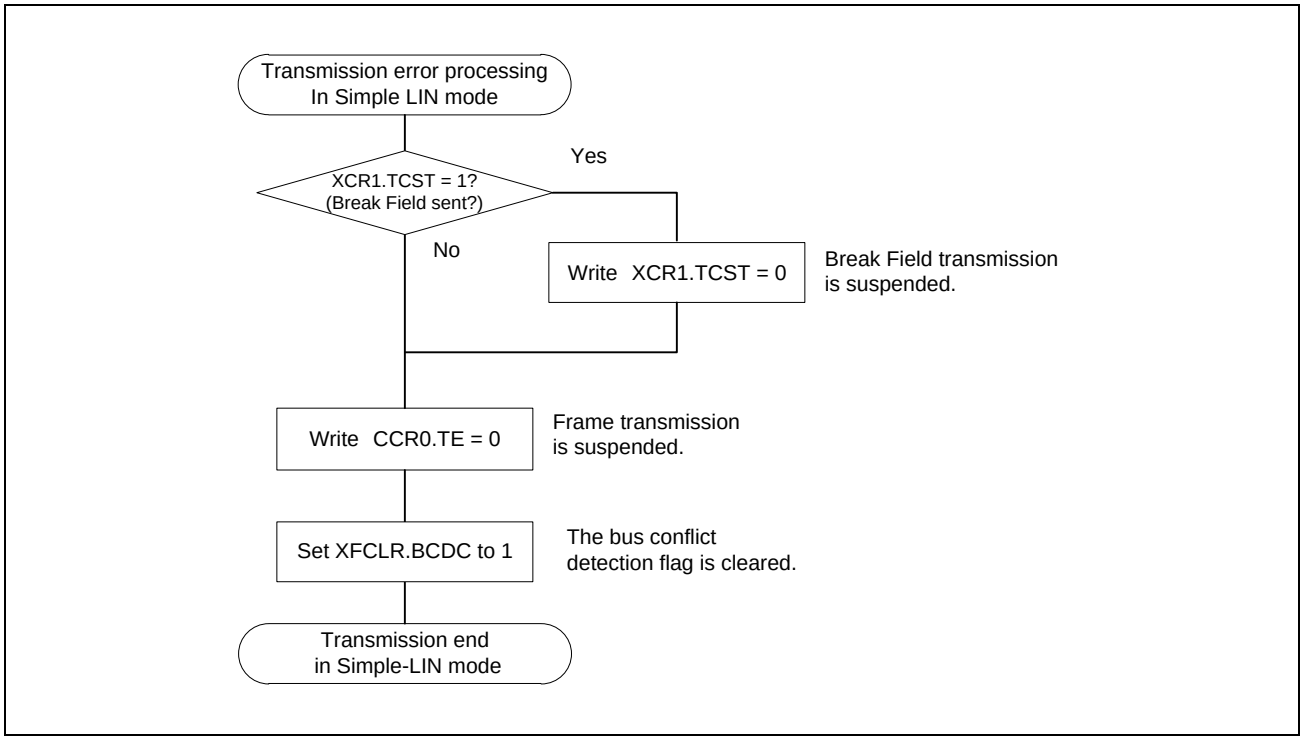


Figure 7.3-64 ERI Interrupt Handling Flow at Transmission in Simple-LIN Mode

7.3.6.4 Simple-LIN Bit Rate Measurement Function

This function measures a bit rate between the effective edges of the input signal from the RXD pin. **Figure 7.3-65** shows an operation example of the bit rate measurement function.

(1) Writing 1b to XCR1.SDST and XCR1.BMEN enables bit rate measurement. When this bit is set to 1b, the valid edge interval of Control Field 0 and Control Field 1 data is measured. However, the bit rate is not measured between the Break Field and the Break Delimiter. Set XCR1.BMEN and XCR1.SDST to 1b simultaneously only when measuring the bit rate.

(2) Because the bit rate is not measured in the Break Field, the effective edge detection flag is not set to 1b at the rising edge at the end of the Break Field, and the counter capture value is not stored in XSR1.TCNT[15:0].

(3) The counter starts counting from the falling edge of the start bit in Control Field 0. The Break Delimiter count value is not captured in XSR1.TCNT[15:0].

(4) The rising edge of the start bit is detected as an effective edge, and then the XSR0.AEDF flag is set to 1b. If XCR0.AEDIE has been set to 1b at this time, an AED interrupt is output. The start bit count value is stored in XSR1.TCNT[15:0]. The XSR1.TCNT[15:0] value is retained until the effective capture value is read.

(5) Even if an effective edge is input from the RXD input pin, the count value of this effective edge timing is not captured because the XRS1.TCNT[15:0] value has not been read and retention has not been released. In this case, an AED interrupt is not output.

(6) The XSR1.TCNT[15:0] value is read. Then the retention of XSR1.TCNT[15:0] is released and the XSR0.AEDF flag is cleared by hardware.

(7) Because the retention of XSR1.TCNT[15:0] has been released, the count value is captured at the effective edge and is retained. At the same time, the XSR0.AEDF flag is set to 1b, and if XCR0.AEDIE has been set to 1b, an AED interrupt is output. The bit rate can be adjusted by calculating it from the count value between effective edges by software and by changing the RSCI settings.

(8) To disable bit rate measurement, write 0b to XCR1.BMEN.

(9) The XSR0.AEDF value and the XSR1.TCNT[15:0] value remain unchanged at the effective edge timing because the bit rate measurement function is disabled.

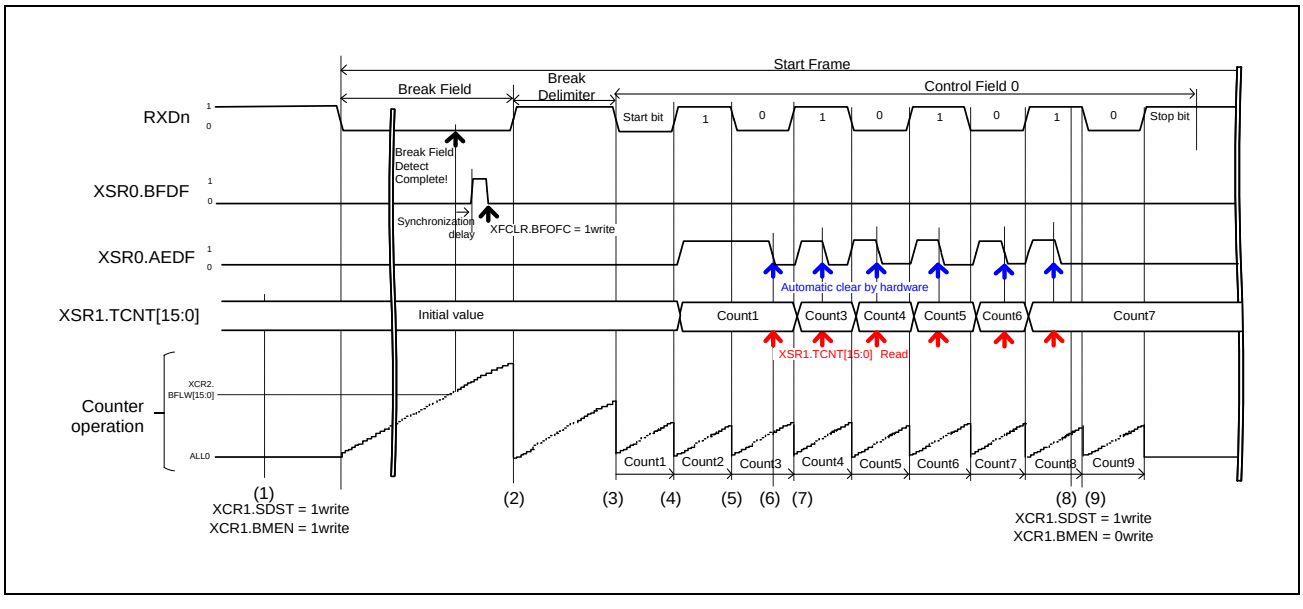


Figure 7.3-65 Operation Example of the Bit Rate Measurement Function

7.3.7 Simple I2C Mode

Simple I2C mode supports master operation only. Simple I2C bus format is composed of 8 data bits and an acknowledge bit. A slave-address frame follows start condition or restart condition. And the slave-address frame is used to specify a slave device as the partner by master device. The currently specified slave device remains valid until a new slave device is specified or a stop condition is satisfied. The 8-bit data of each frame is transmitted from MSB.

Figure 7.3-66 shows I2C bus format, and **Figure 7.3-67** shows the timing of I2C.

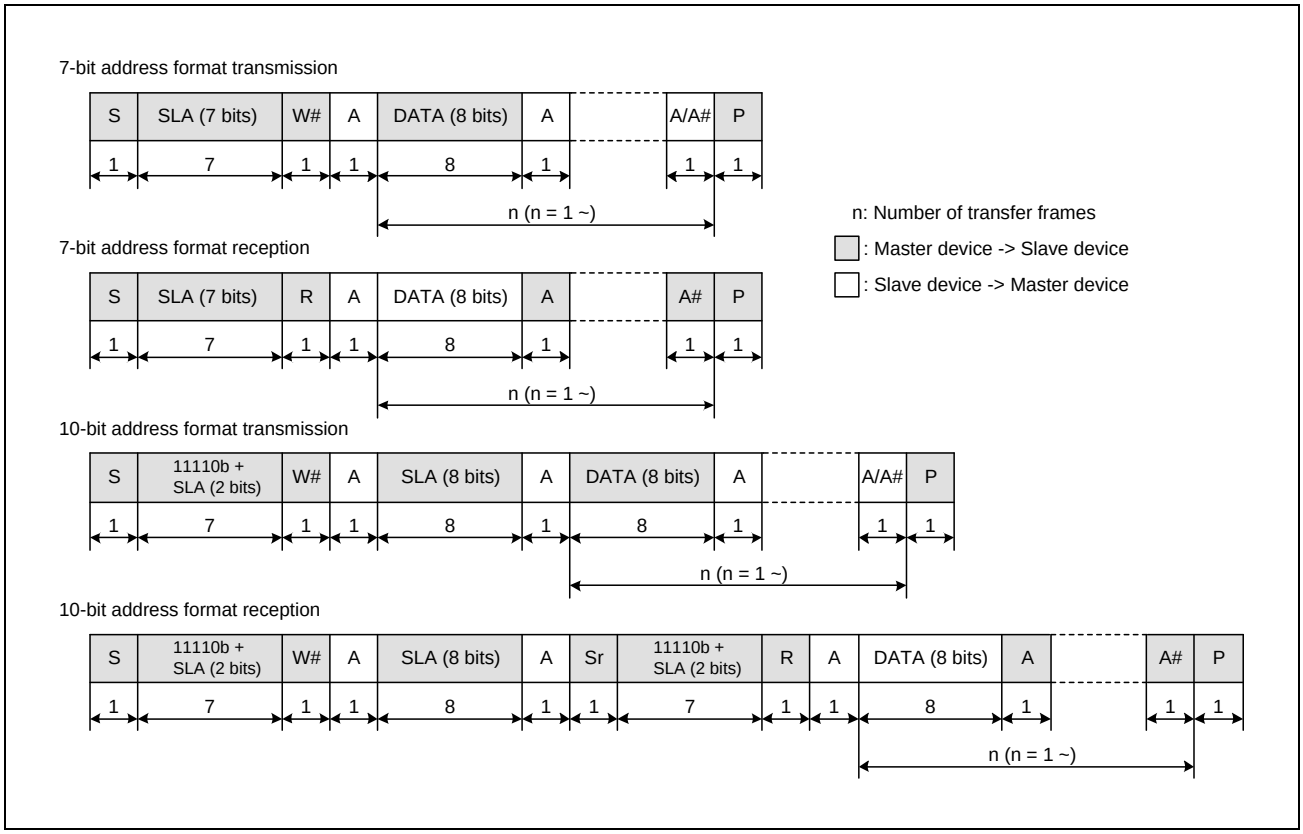
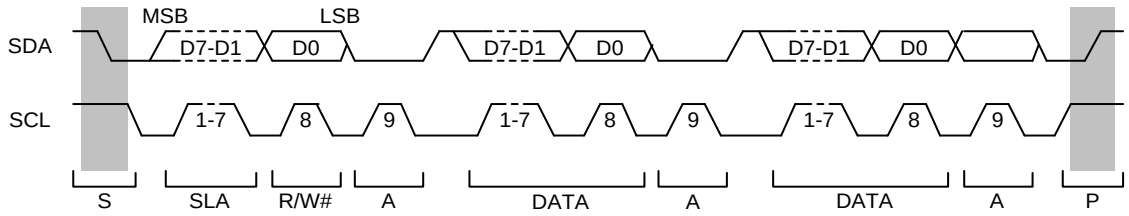


Figure 7.3-66 I2C Bus Format



[Symbol explanation]

S: Indicates a start condition. The master device changing the level on the SDAn line from the high to the low level when the SCLn line is at the high level.

SLA: Indicates a slave address. The master device selects a slave device.

R/W#: Indicates the direction of transfer (reception or transmission). When R/W# is high, the transfer direction is from the slave device to the master device. When R/W# is low, the transfer direction is from the master device to the slave device.

A/A#: Indicates an acknowledge. This is returned by the slave device for master transmission and by the master device for master reception. The low level indicates ACK and the high level indicates NACK.

Sr: Indicates a restart condition. The master device changing the level on the SDAn line from the high to the low level when the SCLn line is at the high level.

DATA: Indicates the received or transmitted data.

P: Indicates a stop condition. The master device changing the level on the SDAn line from the low to the high level when the SCLn line is at the high level.

Figure 7.3-67 I2C Bus Timing (SLA = 7 Bit)

7.3.7.1 Generation of Start, Restart, and Stop Conditions

Writing 1b to ICR.IICSTAREQ causes the generation of a start condition. The following operations are done at the generation of a start condition.

- The level on the SDAn line falls (from the high level to the low level) and the SCLn line is kept in the released state.
- The hold time for the start condition is secured as half of a bit period at the bit rate determined by the setting of the CCR2.BRR.
- The level on the SCLn line falls (from the high level to the low level), the ICR.IICSTAREQ is set (to 0b), and a start condition generated STI interrupt (TEI) is output.

Writing 1b to ICR.IICRSTAREQ causes the generation of a restart condition. The following operations are done at the generation of a restart condition.

- The SDAn line is released and the SCLn line is kept at the low level.
- The period at low level for the SCLn line is secured as half of a bit period at the bit rate determined by the setting of the CCR2.BRR.
- The SCLn line is released (transition from the low to the high level).
- Once the high level on the SCLn line is detected, the setup time for the restart condition is secured as half of a bit period at the bit rate determined by the setting of the CCR2.BRR.
- The level on the SDAn line falls (from the high level to the low level).
- The hold time for the restart condition is secured as half of a bit period at the bit rate determined by the setting of the CCR2.BRR.

- The level on the SCLn line falls (from the high level to the low level), the ICR.IICRSTAREQ is set (to 0b), and a restart condition generated STI interrupt (TEI) is output.

Writing 1b to ICR.IICSTPREQ causes the generation of a stop condition. The following operations are done at the generation of a stop condition.

- The level on the SDAn line falls (from the high level to the low level) and the SCLn line is kept at the low level.
- The period at low level for the SCLn line is secured as half of a bit period at the bit rate determined by the setting of the CCR2.BRR.
- The SCLn line is released (transition from the low to the high level).
- Once the high level on the SCLn line is detected, the setup time for the stop condition is secured as half of a bit period at the bit rate determined by the setting of the CCR2.BRR.
- The SDAn is released (transition from the low to the high level), the ICR.IICSTPREQ is set (to 0b), and a stop condition generated STI interrupt (TEI) is output.

Figure 7.3-68 shows the timing of operations in the generation of start, restart and stop conditions.

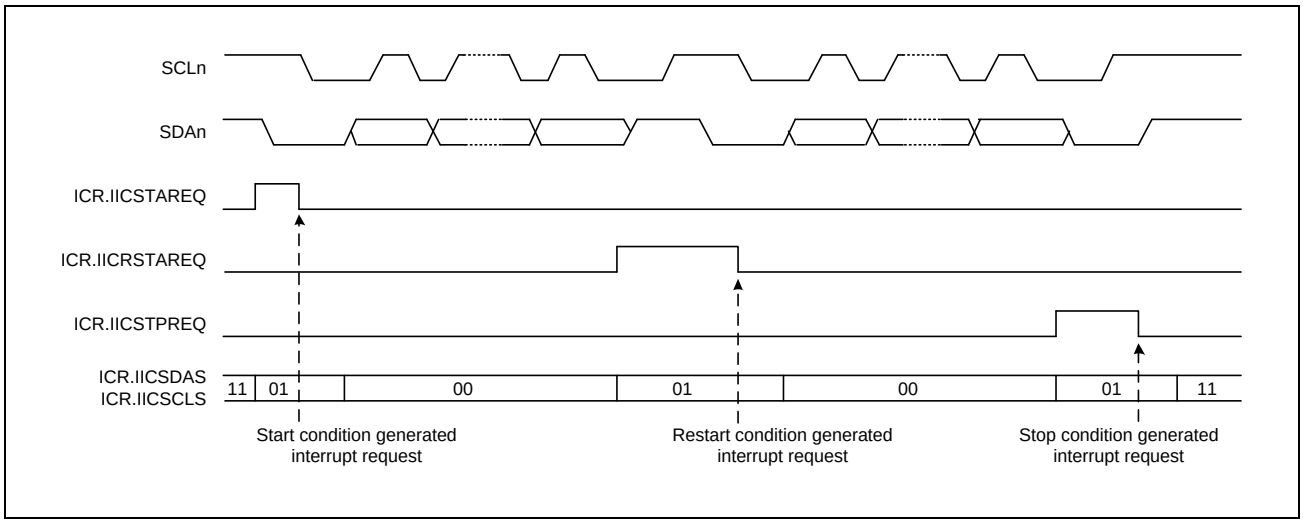


Figure 7.3-68 The Timing of Operations in the Generation of Start, Restart and Stop Conditions

7.3.7.2 Clock Synchronization

The slave device of the communication partner may make SCLn line Low-level with a view to insert a wait. Setting the ICR.IICCSC to 1b, applies control to obtain synchronization when the levels of the internal SCLn clock signal and the level being input on the SCLn pin differ.

When the ICR.IICCSC is set to 1b, the level of the internal SCLn clock signal changes from low to high, counting to determine the period at high level is stopped while the low level is being input on the SCLn pin, and counting to determine the period at high level starts after the transition of the input on the SCLn pin to the high level. The interval until counting to determine the period at high level starts on the transition of the SCLn pin to the high level is the total time which contains the SCLn input delay, the noise filtering delay of the SCLn pin (2 or 3 cycles of the filtering clock), and the internal processing delay (1 or 2 cycles of RSCI_m_TCLK). The period at high level of the internal SCLn clock is extended even if other devices are not placing the low level on the SCLn line.

If the ICR.IICCSC is 1b, synchronization is obtained for the transmission and reception of data by taking the logical AND of the input on the SCLn pin and the internal SCLn clock. If the ICR.IICCSC is 0b, synchronization with the internal SCLn clock is obtained for the transmission and reception of data.

If a slave device inserts a period of waiting into the interval until the transition of the internal SCLn clock signal from the low to the high level after a request for the generation of a start, restart, or stop condition is issued, the time until generation is prolonged by that period.

If a slave device inserts a period of waiting after the transition of the internal SCLn clock signal from the low to the high level, although the generation-completed interrupt is issued without stopping the period of waiting, generation of the condition itself is not guaranteed.

Figure 7.3-69 shows an example of operations to synchronize the clocks.

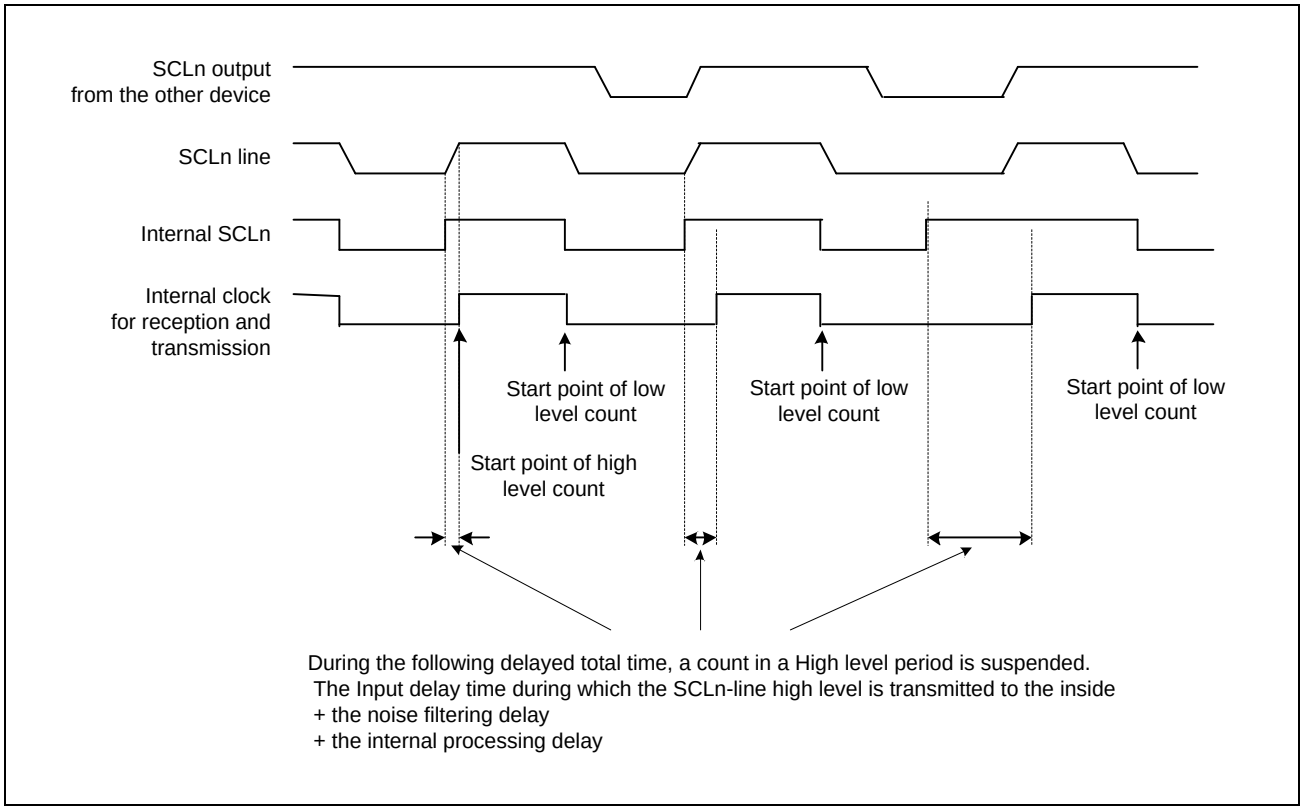


Figure 7.3-69 Example of Operations for Clock Synchronization

7.3.7.3 SDA Output Delay

The ICR.IICDL[4:0] can be used to set a delay for output on the SDAn pin relative to falling edges of output on the SCLn pin. The delay-time settings are selectable from 0 to 31 cycles of the clock signal from the baud rate generator (The base is RSCI_m_TCLK and selected the divided clock by the CCR2.CKS[1:0]). About Start/Restart/Stop conditions, 8-bit transmission data and acknowledge, the SDAn pin output can be delayed.

If the SDA output delay is shorter than the falling time of the SCLn output pin, the change of the SDAn output pin will start while the SCLn output pin level is falling, then there is a possibility of erroneous operation for slave devices. Ensure setting the SDA output delay greater than the SCLn maximum falling time (300 ns for I2C normal/fast mode).

Figure 7.3-70 shows the timing of SDA output delay.

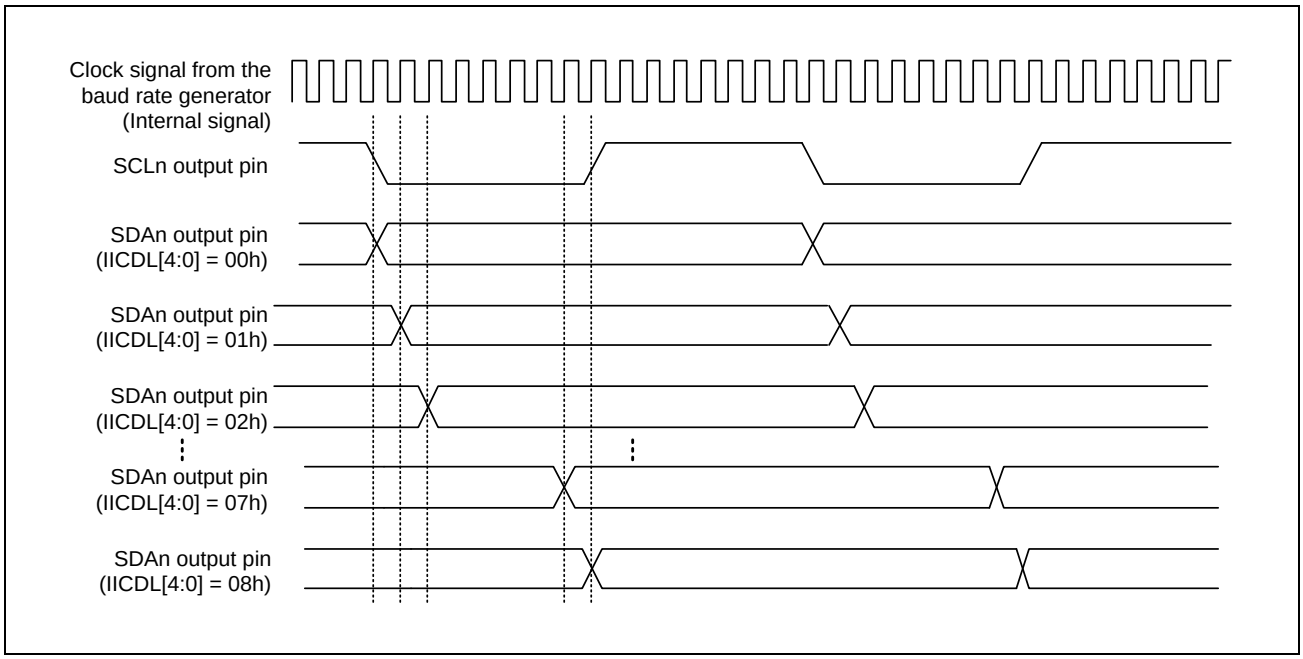
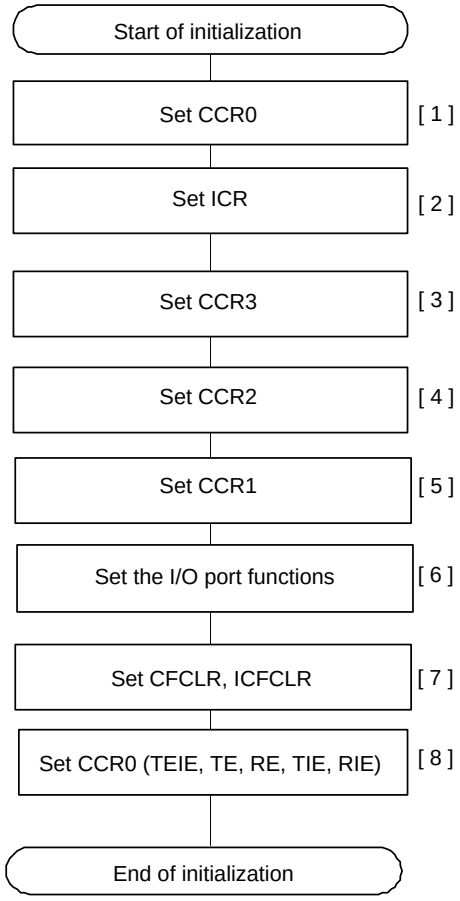


Figure 7.3-70 Timing of SDA Output Delay

7.3.7.4 SCI Initialization (Simple I2C Mode)

Write initial value (all 0) to CCR0, then initialize SCI according to **Figure 7.3-71**.

When changing the operating mode, transfer format, and so on, be sure to set 0b to CCR0.TE and CCR0.RE before proceeding with the changes. (Or write initial value to CCR0 again.)



[1] Set the following bits of CCR0 to 0.
TEIE, TIE, RIE, TE, RE
If you have not changed from the initial settings, you can skip this step.

[2] Set the IICSDAS[1:0] and IICSCLS[1:0] to 11b.
Set the IICDL[4:0] and IICINTM as required.
Set the IICACKT and the IICCSC bits to 1.

[3] Set the transmission/reception format as the communication mode (MOD[2:0] = 100b) and CCR3.LSBF = 0.

[4] Set the bit rate modulation function*1, the clock selection, and the bit rate.

[5] Set up the Noise-filter function.

[6] Set I/O port settings that allow use of the SCLn and SDAn pin functions.

[7] Write 1 to the following bits and clear the corresponding flag.
CFCLR.RDRFC, FERC, PERC, MFFC, ORERC, DFERC, DPERC, DCMFC, ERSC
ICFCLR.IICSTIFC

[8] Set the TE and RE to 1.
To enable interrupts, set the TE bit, TIE bit, TEIE bit, RE bit, and RIE bit to 1 with one instruction at the same time.
(for transmission and when the IICINTM bit is 1, clear the RIE bit.)
Setting the TE and RE bits to 1 makes the SCLn and SDAn pin functions available.

Note: Set the CCR0.TE and RE bits to 0b or 1b at the same time.

Note 1. If you do not use the bit rate modulation function, you do not need to set it.

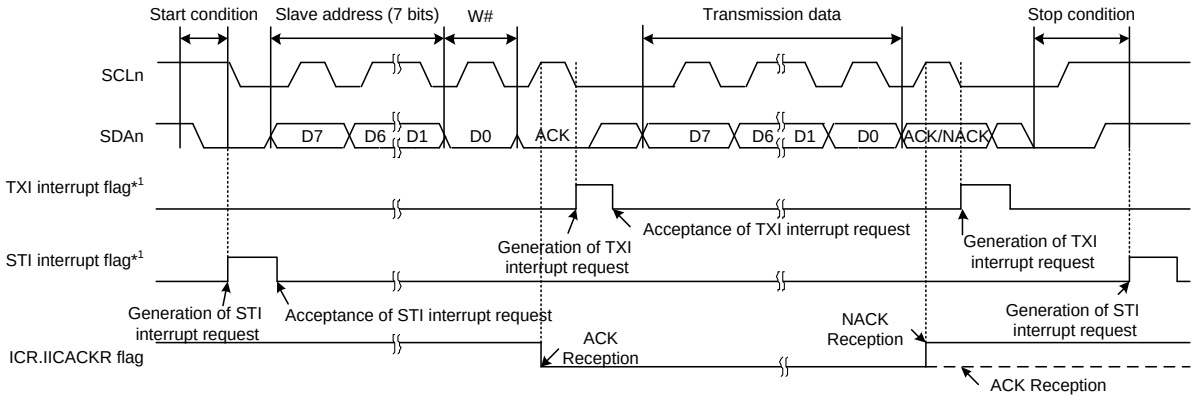
Figure 7.3-71 Example of the Flow of SCI Initialization (for Simple I2C Mode)

7.3.7.5 Operation in Master Transmission (Simple I2C Mode)

Figure 7.3-72 and **Figure 7.3-73** show examples of operations in master transmission, **Figure 7.3-74**, **Figure 7.3-75**, and **Figure 7.3-76** show the example flowcharts. See **Table 7.3-37** about the STI interrupt.

Figure 7.3-72 shows the operation example when ICR.IICINTM is 1b (reception/transmission interrupts are in use). In this case, you can start the DMAC by using the TXI interrupt. However, if the DMAC is used, ACK/NACK cannot be confirmed. If you need to confirm ACK/NACK, use the CPU to prepare transmit data. In simple I2C mode, the TXI interrupt is generated when communication of one frame is completed. In master transmission, the RXI interrupt is not used, so set the CCR0.RIE bit to 0b.

Figure 7.3-74 shows a flow chart in the case of ICR.IICINTM is 1b and address transmission by CPU and data transmission by DMAC. **Figure 7.3-75** shows a flow chart of address and data transmission by CPU. When 10-bit slave addresses are in use, steps [3] and [4] are repeated twice.



Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

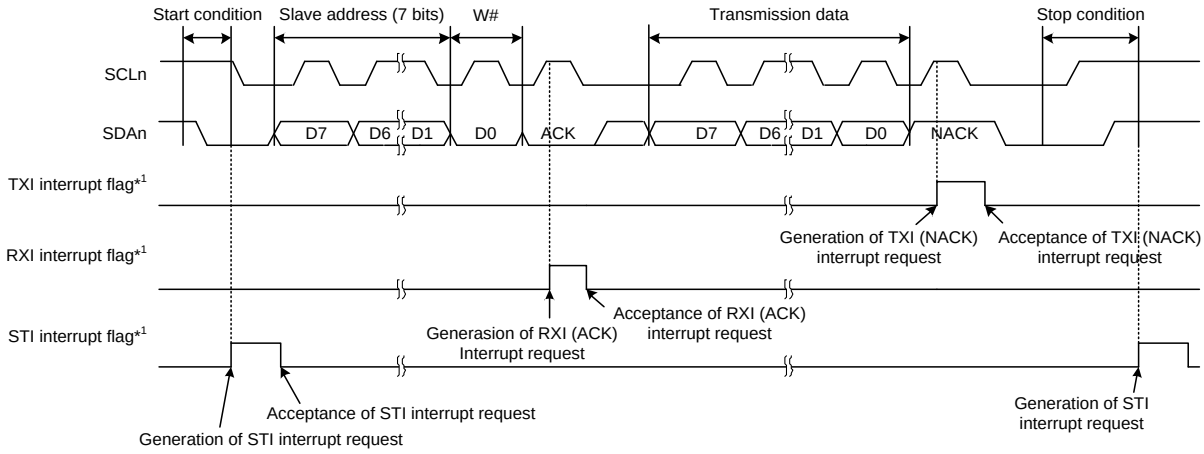
Figure 7.3-72 Example 1 of Operations for Master Transmission in Simple I2C Bus Mode (7-Bit Slave Address, Transmission and Reception Interrupt in Use (ICR.IICINTM = 1b))

Figure 7.3-73 shows an example of operations when ICR.IICINTM is 0b (ACK and NACK interrupt in use). In this case, DMAC is activated by the ACK (RXI) interrupt, and necessary number of data bytes are transmitted. When the NACK is received, error processing, such as transmission stop and retransmission, is performed by the NACK (TXI) interrupt as the trigger.

Figure 7.3-76 shows a flow chart of ICR.IICINTM is 0b.

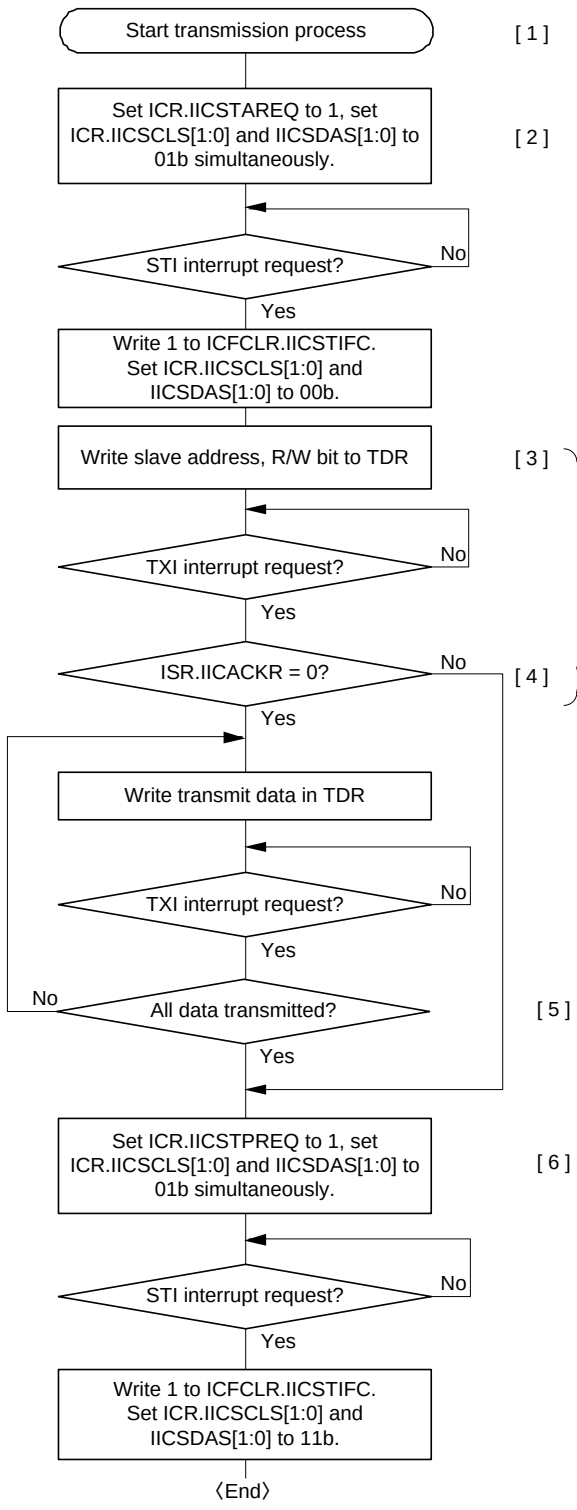
To resume communication after interrupting communication for some reason after writing transmit data to TDR, follow the procedure below.

1. Set the CCR0.TE and CCR0.RE bits to 0b to stop communication.
2. Set ICR.IICSCLS[1:0] and ICR.IICSDAS[1:0] to 11b, release the I2C bus, and clear various condition generation requests.
3. When CSR.RDRF = 1b, the RDR register is read by dummy and the RDRF bit is set to 0b.
4. Set the CCR0.TE and CCR0.RE bits to 1b and restart communication.



Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-73 Example 2 of Operations for Master Transmission in Simple I2C Bus Mode (7-Bit Slave Address, ACK and NACK Interrupt in Use (ICR.IICINTM = 0b))



[1] Start transmission process
For transmission, set the CCR0.RIE to 0.
(RXI and ERI interrupts requests are disabled)

[2] Generate a start condition.

[3] Writing to TDR:
Writing the slave address and value for the R/W bit to TDR.

[4] Confirm ACK response from the slave device:
Check the ISR.IICACKR bit. If its value is 0, it is indicated that the slave device responded with ACK and operations proceed. If its value is 1, it is indicated that there was no response from a slave device so the next transition is to generation of the stop condition.

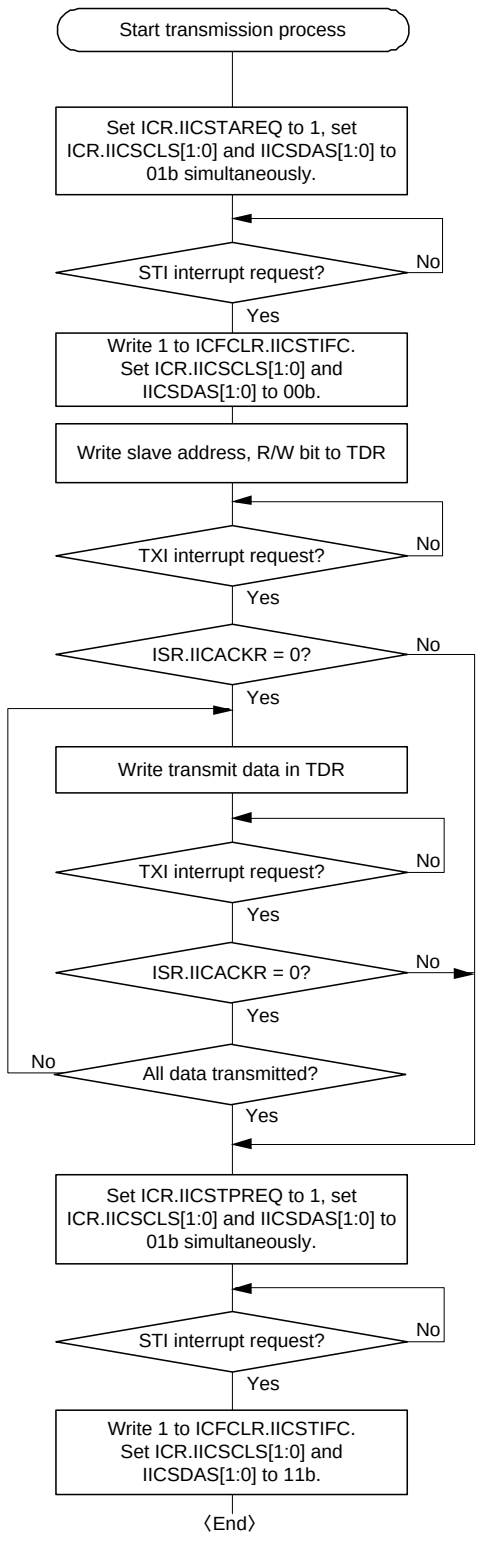
[5] Steps for continuing with serial transmission:
When transmission is to continue, write further transmit data to TDR. Except for the first data to be transmitted, a TXI interrupt request can activate the DMAC to handle writing of data to TDR.
But it cannot be confirmed status of ACK or NACK in this case.

[6] Generation of a stop condition.

If 10-bits slave address are in use, processing from [3] to [4] is repeated twice.

Note: In simple I2C mode, the TXI interrupt request is generated when communication is completed.

Figure 7.3-74 Example of the Procedure for Master Transmission Operations in Simple I2C Mode (when ICR.IICINTM is 1b, and when Confirming ACK/NACK by Address Transmission Only.)



[1] [1] Start transmission process
For transmission, set the CCR0.RIE to 0.
(RXI and ERI interrupts requests are disabled)

[2] [2] Generate a start condition.

[3] [3] Writing to TDR:
Writing the slave address and value for the R/W bit to TDR.

[4] Confirm ACK response from the slave device:
Check the ISR.IICACKR bit. If its value is 0, it is indicated that the slave device responded with ACK and operations proceed. If its value is 1, it is indicated that there was no response from a slave device so the next transition is to generation of the stop condition.

[5] Steps for continuing with serial transmission:
When transmission is to continue, write further transmit data to TDR. If ACK or NACK is checked in every transmission, DMAC cannot be used.

[6] Generation of a stop condition.

[3] to [4] If 10-bits slave address are in use, processing from [3] to [4] is repeated twice.

Note: In simple I2C mode, the TXI interrupt request is generated when communication is completed.

Figure 7.3-75 Example of the Procedure for Master Transmission Operations in Simple I2C Mode (when ICR.IICINTM is 1b, and when Confirming ACK/NACK in All Transmissions.)

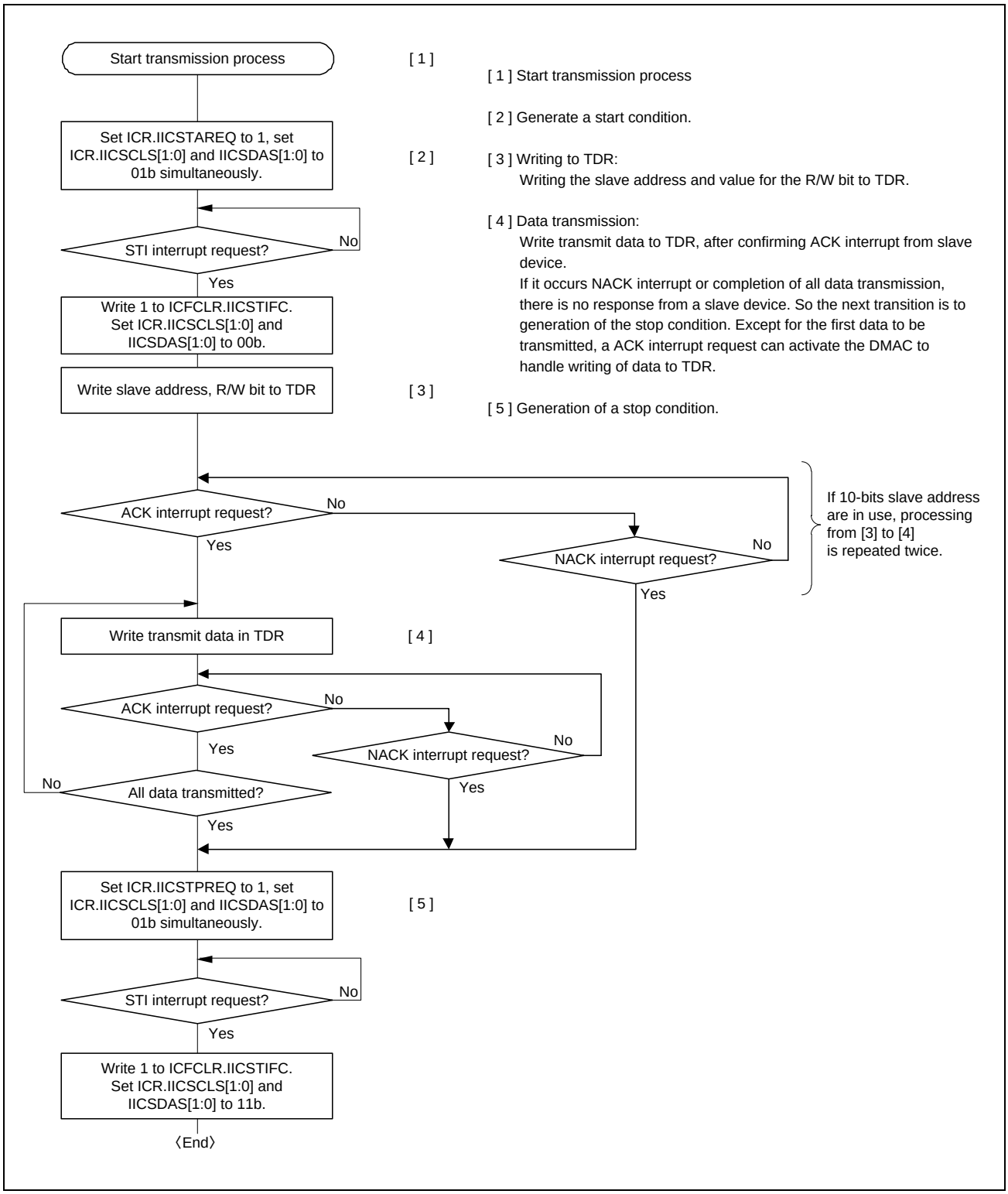
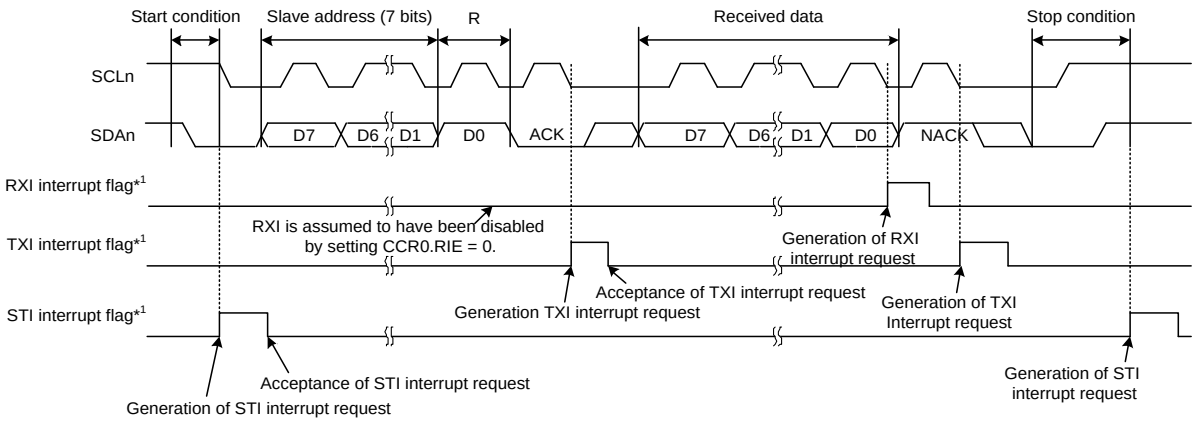


Figure 7.3-76 Example of the Procedure for Master Transmission Operations in Simple I2C Mode (when ICR.IICINTM is 0b.)

7.3.7.6 Master Reception (Simple I2C Mode)

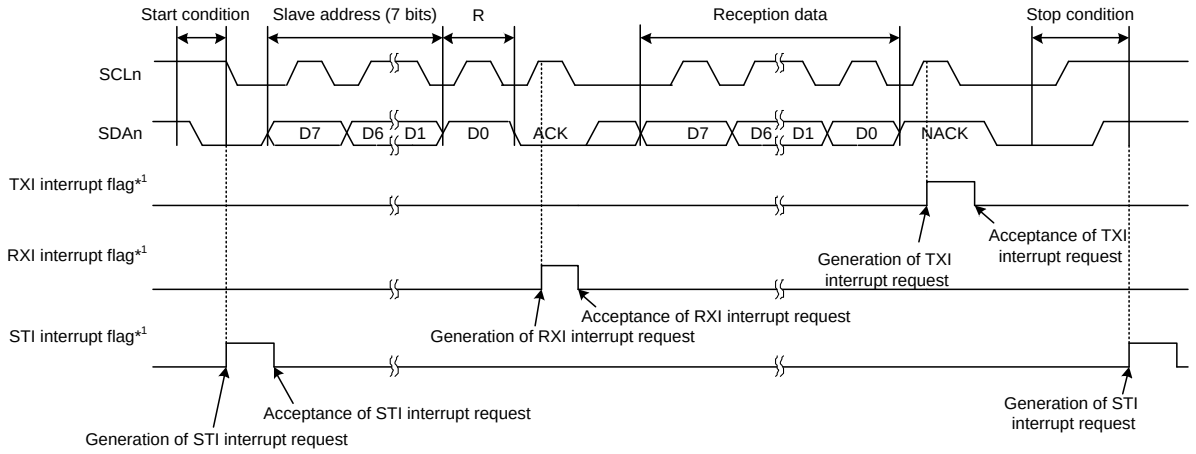
Figure 7.3-77 and **Figure 7.3-78** show example of operations in simple I2C mode master reception. **Figure 7.3-79** and **Figure 7.3-80** show flowchart of the master reception. The value of the ICR.IICINTM bit is assumed to be 1b (use reception and transmission interrupts) and 0b (use ACK and NACK interrupts).

In simple I2C mode, the transmit data empty interrupt (TXI) is generated when communication of one frame is completed.



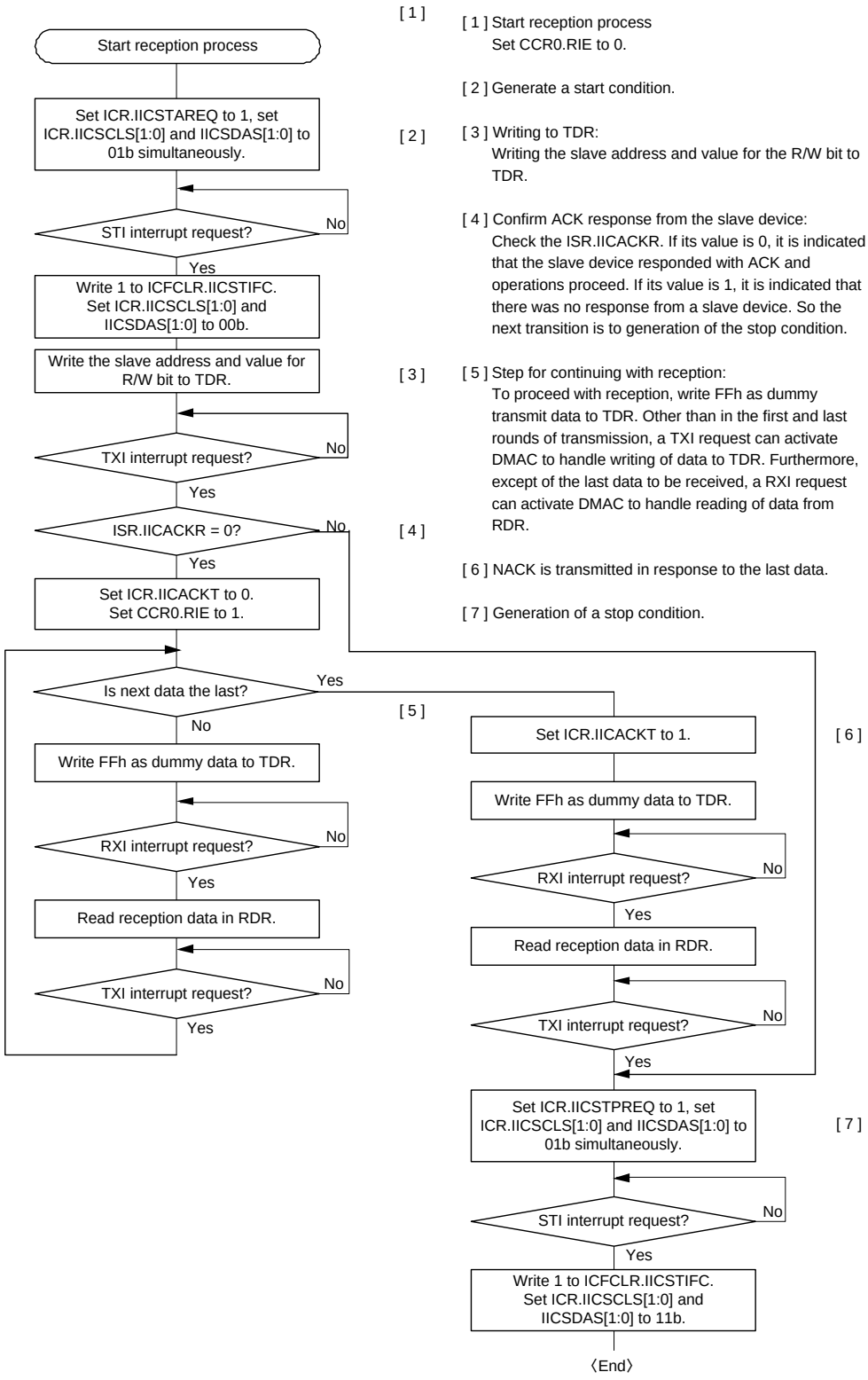
Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-77 Example of Operations for Master Reception in Simple I2C Bus Mode (7-Bit Slave Address, Transmission and Reception Interrupt in Use (ICR.IICINTM = 1b))



Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-78 Example of Operations for Master Reception in Simple I2C Bus Mode (7-Bit Slave Address, ACK and NACK Interrupt in Use (ICR.IICINTM = 0b))



Note: In simple I2C mode, the TXI interrupt request is generated when communication is completed.

Figure 7.3-79 Example of the Procedure for Master Reception Operations in Simple I2C Mode (when ICR.IICINTM is 1b, and Transmission Interrupts and Reception Interrupts are in Use.)

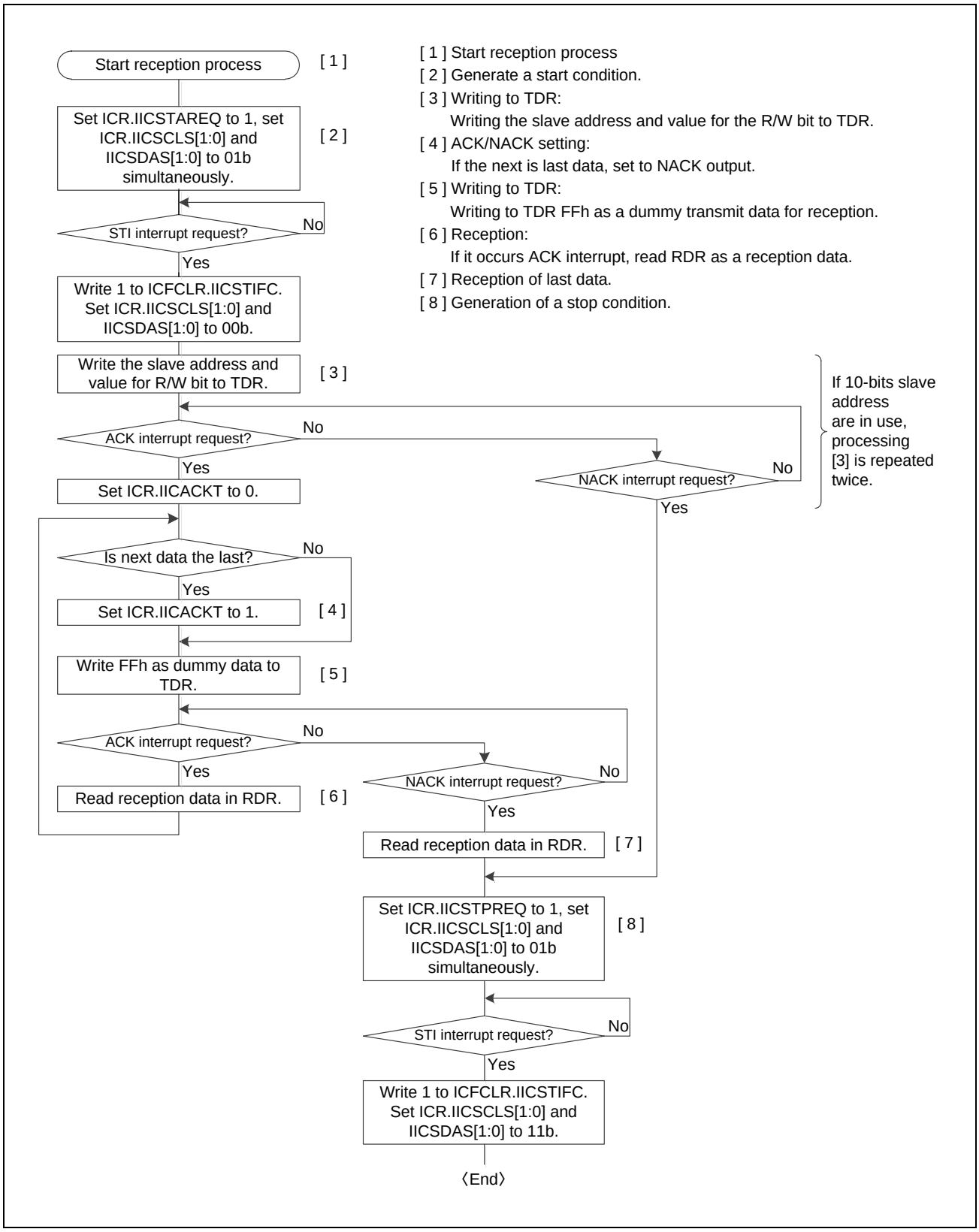


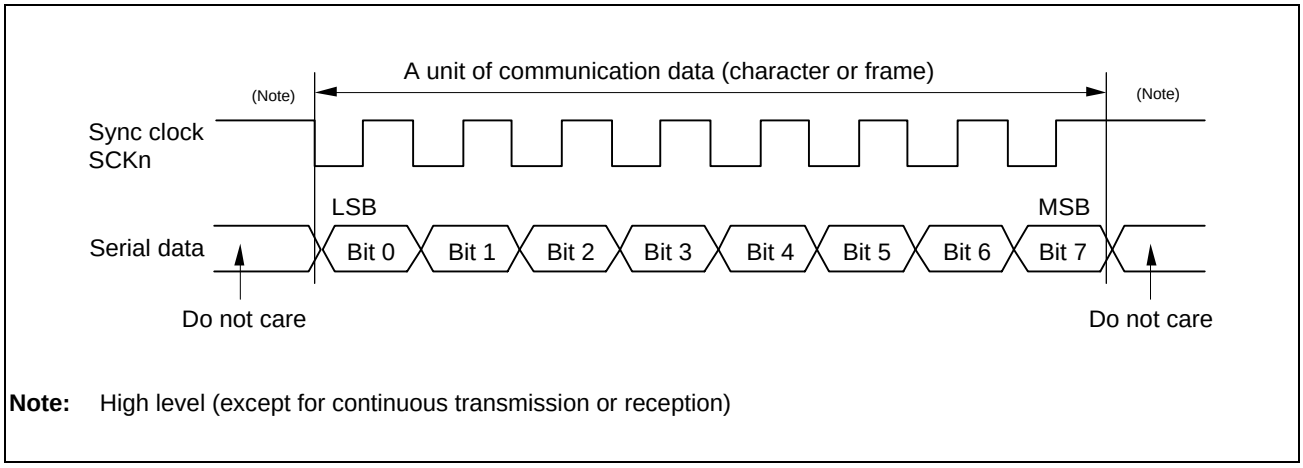
Figure 7.3-80 Example of the Procedure for Master Reception Operations in Simple I2C Mode (when ICR.IICINTM is 0b, and ACK Interrupts and NACK Interrupts are in Use.)

7.3.8 Clock Synchronous Mode

Figure 7.3-81 shows the communication data format of clock synchronous serial communication.

In clock synchronous mode, data is transmitted and received in synchronization with clock pulses. A communication data character consists of 8-bit data. In clock synchronous mode, no parity bit can be added. In data transmission when CPHA = 1b and CPOL = 1b, the SCI outputs data from the falling edge of the sync clock until the next falling edge. In data reception, data is read at the rising edges of the sync clock. After 8-bit data is output, the communication line holds the final-bit output state. In slave communication when CPHA = 0b, however, the communication line holds the first-bit output state.

Because the SCI has an internal transmitter and a receiver independently, SCI enable full-duplex communication by sharing a communication clock of the transmitter and the receiver. Furthermore, because both the transmitter and the receiver have a double- buffer structure, continuous transmission and reception are possible by writing the next transmit data during transmission and reading the previous receive data during reception.



Note: High level (except for continuous transmission or reception)

Figure 7.3-81 Data Format of Clock Synchronous Serial Communication (LSB First, CPHA = 1b, CPOL = 1b)

7.3.8.1 Clock

(1) When the internal clock is selected

When the CCR3.CKE[1:0] bits are set to 00b or 01b (master mode), the internal clock generated by the baud rate generator can be selected and the sync clock is output from the SCKm pin. Eight pulses of the sync clock are output during single- character transmission/reception. The sync clock remains at a high level*[1] while no transmission or reception is performed. In transmission-only or transmission/reception, the sync clock is not output unless transmit data is prepared.

When the internal clock is selected, the clock with a delay from the SCKm signal is used for the master reception sampling clock. This ensures the data setup time and hold time for high-speed communication.

Note 1. When CCR3.CPOL = 1b, the sync clock stops at a high level. When CCR3.CPOL = 0b, the sync clock stops at a low level.

(2) When the external clock is selected

When the CCR3.CKE[1:0] bits are set to 10b or 11b (slave mode), data is transmitted and received using the external clock that is input from the SCKm pin.

7.3.8.2 CTS and RTS Functions

The CTS function performs transmission/reception and controls transmission start using the CTSn# pin input when the internal clock is selected. Setting the CCR1.CTSE bit to 1b enables the CTS function. Two settings are possible for the CTS/RTS pin: dual setting for using either function with a single pin, and dedicated setting for using each function simultaneously with two pins. The CCR1.CTSPEN register is used for these settings.

When the CTS function is enabled, transmission/reception and transmission start only when the CTSn# pin input level is low. Even if the CTSn# pin input becomes high level during transmission/reception or transmission operation, frames that are being transmitted/received or being transmitted are not affected and transmission/reception or transmission operation continues.

The RTS function makes a serial communication start request using the RTSn# pin output when the external sync clock is selected. When serial communication is enabled, the RTSn# pin outputs a low level. A low level and a high level are output under the following conditions.

[Conditions for low level]

1. When a non-FIFO buffer is selected, all the following conditions are met.
   - CCR0.RE bit or TE bit = 1b
   - Serial communication is enabled.
   - No receive data is present before reading. (when CCR0.RE bit = 1b)
   - Data before transmission start is remaining in the TSR register. (when CCR0.TE bit = 1b)
   - CSR.ORER flag = 0b
2. When a FIFO buffer is selected, all the following conditions are met.
   - CCR0.RE bit or TE bit = 1b
   - Serial communication is enabled
   - The number of receive data stored in the receive FIFO (RDR register) is less than the value set in FCR.RSTRG[4:0]. (when CCR0.RE bit = 1b)
   - Unsent data is remaining in the transmit FIFO (TDR register). (when CCR0.TE bit = 1b and CCR3.CKE[1] bit = 0b)
   - Data before transmission start is remaining in the TSR register. (when CCR0.TE bit = 1b and CCR3.CKE[1] bit = 1b)
   - ORER (RDR.ORER) flag = 0b

[Conditions for high level]

1. When a non-FIFO buffer is selected If conditions for low level are not met:
   When CCR0.RE is set to 0b without reading the RDR register to terminate reception after reception is complete, the RTSn# pin output level remains high. At this time, write 0b to CCR0.RE.
2. When a FIFO buffer is selected
   If conditions for low level are not met:

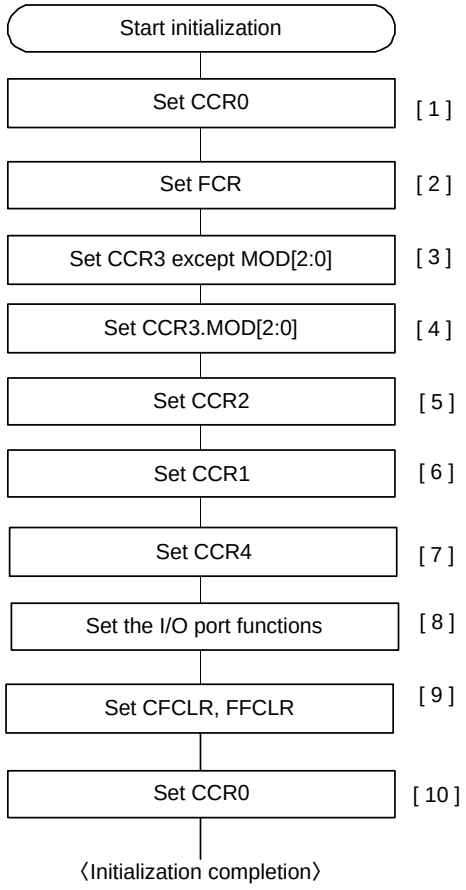
7.3.8.3 Initializing the SCI (Clock Synchronous Mode)

Before starting data transmission/reception, write 0b to CCR0.TE and CCR0.RE (or write initial values to the CCR0 register) and initialize the SCI according to the flowchart example in **Figure 7.3-82**.

Before changing operating mode or communication format, also be sure to write 0b to CCR0.TE and CCR0.RE.

Note that writing 0b to the RE bit does not initialize the ORER, FER, PER, and RDRF flags in CSR and the RDR register. Also note that writing 0b to the TE bit does not initialize the TEND flag when a FIFO buffer is selected. Attention is also needed for changing operating mode.

When CCR0.TIE = 1b, note that setting the TE bit to 1b from 0b generates a TXI interrupt.



[1] Set the following bits of CCR0 to 0*1.
TEIE, TIE, RIE, TE, RE
If you have not changed from the initial settings, you can skip this step.

[2] Set the TFRST and RFRST to 1 to empty FIFO.
Set the TTRG[4:0], RTRG[4:0], and RSTRG[4:0] bits.

[3] Set CCR3 except of communication mode.
FIFO use / no-use
Transmission / reception format
Clock setting
Leave unused bits (CHR[1:0], STP, RXDESEL, MP, DE, ACS0,GM,BLK) at their initial values.

[4] Set communication mode (MOD[2:0] = 010b)*2.

[5] Select clock, set bit rate*3. Leave unused bits (BCP[2:0], ABCS, ABCSE) at their initial values.

[6] Set up the loop-back function, communication terminal status, and the CTS/RTS function.

[7] Set up the adjust sampling timing function.
Leave unused bits (CMPD[8:0], ATEN, ATT[2:0], AET) at their initial values.

[8] Make I/O port settings to enable input and output functions as required for TXDn, RXDn, SCKn, CTS#, and RTSn# pins.

[9] Write 1 to the following bits and clear the corresponding flag.
CFCLR.RDRFC, FERC, PERC, MFFC, ORERC, DFERC, DPERC, DCMFC, ERSC
FFCLR.DRC

[10] Set the TE or RE bit to 1.*1,*4 To enable interrupts, set the TE bit and TIE bit, and the RE bit and RIE bit to 1 with one instruction at the same time.
Setting the TE and RE bits allows TXDn and RXDn to be used.

Note 1. In simultaneous transmit and receive operations, the TE and RE bits in CCR0 should both be to 0b or set to 1b simultaneously.

Note 2. Set CPOL and CPHA before setting the Communication mode.

Note 3. If you use an external clock, you do not need to set it.

Note 4. When using the internal clock (master), the setting of reception only is prohibited.

Figure 7.3-82 SCI Initialization Flowchart Example (Clock Synchronous Mode)

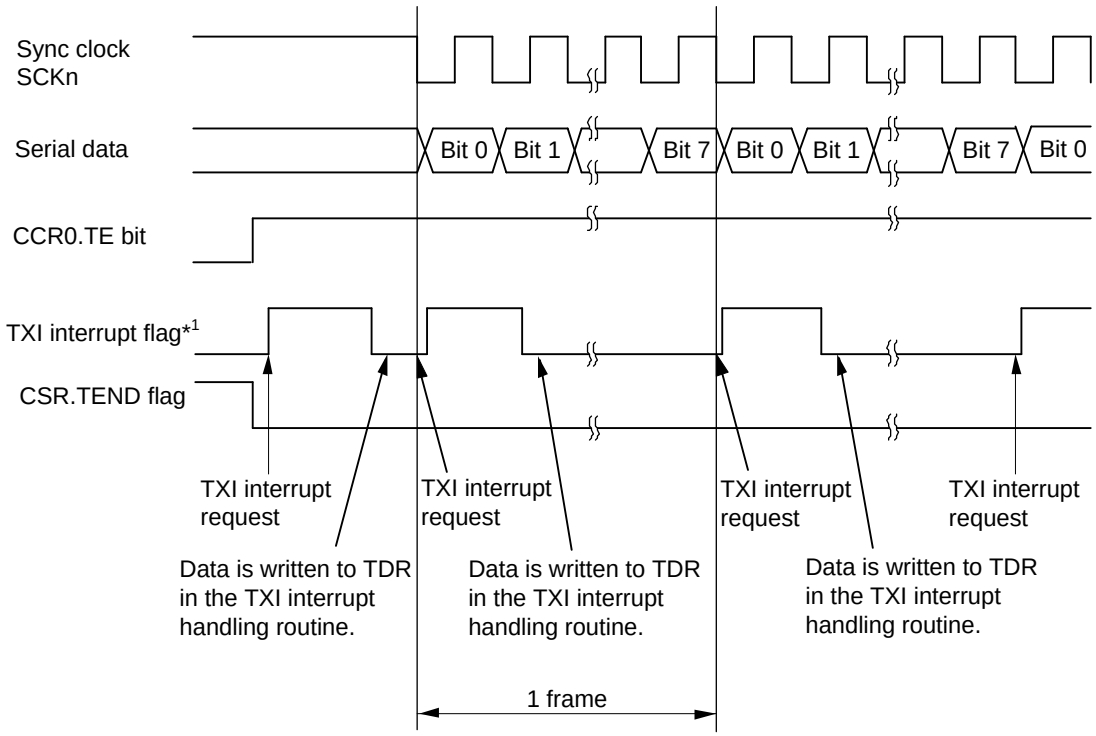
7.3.8.4 Serial Data Transmission (Clock Synchronous Mode)

(1) Non-FIFO selected

Figure 7.3-83, **Figure 7.3-84**, and **Figure 7.3-85** show operation examples of serial transmission in clock synchronous mode. The SCI operates as follows during serial data transmission.

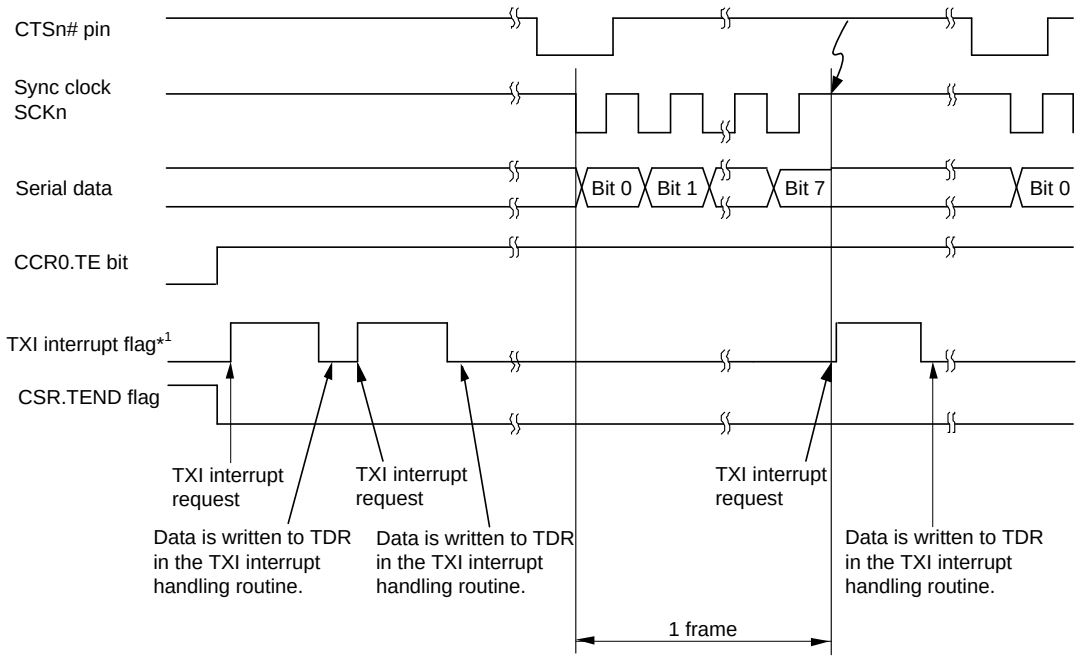
1. When data is written to the TDR register in the TXI interrupt routine, the SCI transfers the data from the TDR register to the TSR register. When starting data transmission, set the CCR0.TIE bit and the CCR0.TE bit to 1b simultaneously by a single instruction. Then a TXI interrupt request is generated.
2. The written data is transferred from the TDR register to the TSR register, which starts transmission. When the CCR0.TIE bit is set to 1b at this time, a TXI interrupt request is generated. Writing the next transmit data to the TDR register before transmission of data transferred previously in the TXI interrupt routine is complete enables continuous transmission. When a TEI interrupt request is used, after the final transmit data is written to the TDR register in the TXI interrupt request processing routine and the final data's transmission is started, set 0b to the CCR0.TIE bit and set 1b to the TEIE bit.
3. The TXDn pin outputs 8-bit data in synchronization with the output clock (in clock output mode) or with the input clock (when external clock is selected). When the CCR1.CTSE bit = 1b (CTS function enabled), the output clock starts after the CTS signal input becomes low level.
4. Update (data write) of the TDR register is checked at the final-bit transmission timing.
5. When the TDR register has been updated, data is transferred from the TDR register to the TSR register to start sending the next frame.
6. If the TDR register has not been updated, the CSR.TEND flag is set to 1b and the final-bit output state is retained. When the CCR0.TEIE bit is set to 1b at this time, a TEI interrupt request is generated. The SCKm pin is held high.

Figure 7.3-86 shows an example of data transmission flowchart.



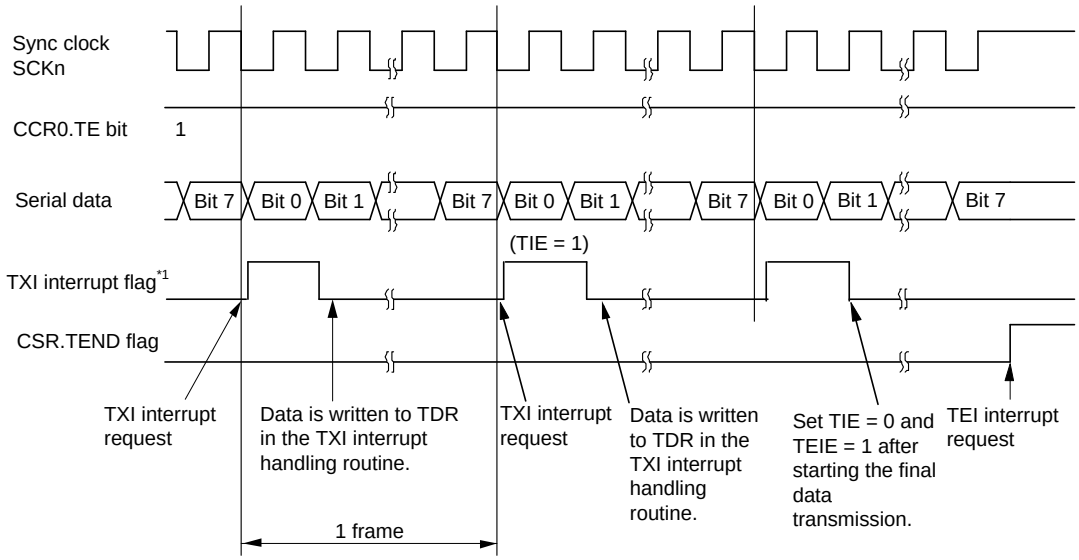
Note 1. For corresponding interrupt source numbers, see **4.6 Interrupt Controller**.

Figure 7.3-83 Serial Transmission Example in Clock Synchronous Mode (1) (CTS Function Not Used/Transmission Start/CPHA = 1b, CPOL = 1b)



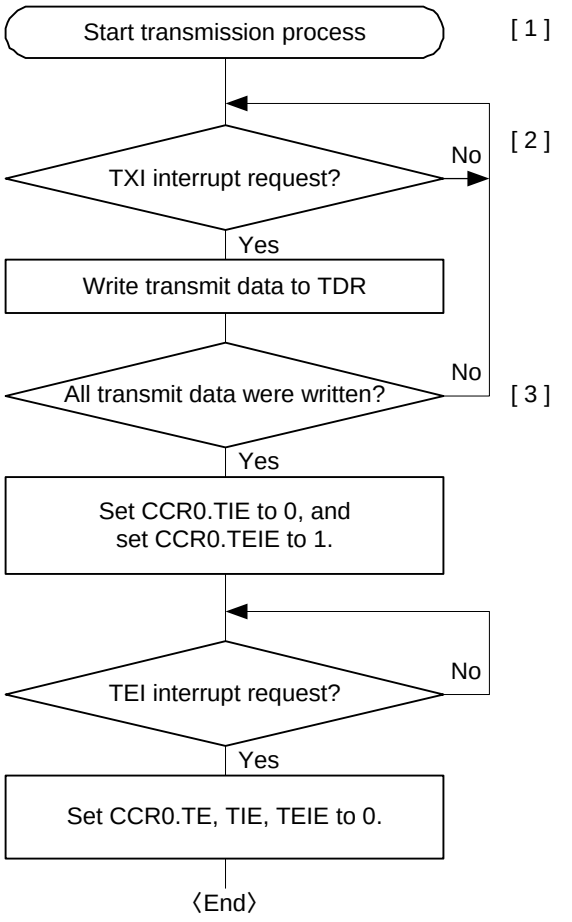
Note 1. For corresponding interrupt source numbers, see **4.6 Interrupt Controller**.

Figure 7.3-84 Serial Transmission Example in Clock Synchronous Mode (2) (CTS Function Used/Transmission Start/CPHA = 1b, CPOL = 1b)



Note 1. For corresponding interrupt source numbers, see **4.6 Interrupt Controller**.

Figure 7.3-85 Serial Transmission Example in Clock Synchronous Mode (3) (during Transmission to Transmission End/CPHA = 1b, CPOL = 1b)



[1] Start transmission process

[2] Write transmit data to TDR by using TXI:
The transmit-data-empty-interrupt (TXI) is generated when transmission-data forwards to TSR from TDR.
Write transmit data to TDR once by using a TXI.

[3] Serial transmission continuation procedure:
To continue serial transmission write transmit data to TDR by using a TXI.
Transmit data can also be written by activating the DMAC.
If you use TEI, set TIE = 0 and TEIE = 1 after writing the last-transmit-data to TDR.

Note: When the external clock is in use (the value of the CKE[1:0] is 10b or 11b), the rising edge on the SCK pin for the last bit sets the TEND flag to 1b. Setting TE to 0b immediately after this, then it may lead to insufficient received data hold time on the receiver side.

Figure 7.3-86 Serial Transmission Flowchart Example in Clock Synchronous Mode (Non-FIFO Selected)

(2) FIFO selected

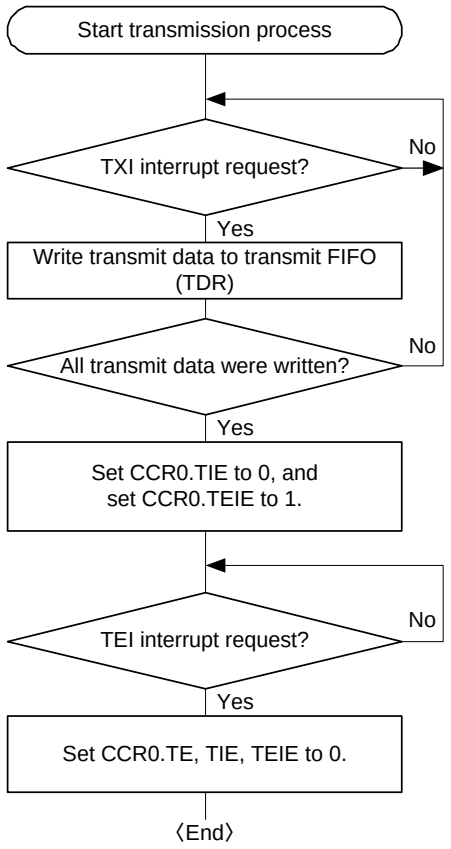
Figure 7.3-87 shows an example of flowchart of serial transmission (a FIFO buffer selected) in clock synchronous mode. The SCI operates as follows when serial data transmission.

1. When data is written to the transmit FIFO (TDR register) in the TXI interrupt routine, the SCI transfers the data from the transmit FIFO (TDR register) to the TSR register. The number of writable transmit data is [NFIFO (= 32) - number of unsent transmit data stored in the transmit FIFO (TDR register)]. In addition, when starting data transmission, set the CCR0.TIE bit and the CCR0.TE bit to 1b simultaneously by a single instruction. Then a TXI interrupt request is generated.
2. Data is transferred from the transmit FIFO (TDR register) to the TSR register and transmission starts. When the number of data stored in the transmit FIFO (TDR register) is equal to or less than the threshold value of the transmit FIFO, the CSR.TDRE flag is set to 1b. When the CCR0.TIE bit is set to 1b at this time, a TXI interrupt request is generated. Writing the next transmit data to the transmit FIFO (TDR register) in the TXI interrupt routine before transmission of data written to the transmit FIFO (TDR register) is complete enables continuous

transmission. When a TEI interrupt request is used, after the final transmit data is written to the transmit FIFO (TDR register) in the TXI interrupt request processing routine, set 0b to the CCR0.TIE bit and set 1b to the TEIE bit.

3. The TXDn pin outputs 8-bit data in synchronization with the output clock (in clock output mode) or with the input clock (when external clock is selected). When CCR1.CTSE = 1b (CTS function enabled), the output clock starts after the CTS signal input becomes low level.
4. The SCI checks whether unsent transmit data is remaining in the transmit FIFO (TDR register)*1 at the final-bit transmission timing.
5. When data is remaining in the transmit FIFO (TDR register), the data is transferred from the transmit FIFO (TDR register) to the TSR register to start sending the next frame.
6. If no data is remaining in the transmit FIFO (TDR register), the CSR.TEND flag is set to 1b and the final-bit output state is retained. When the CCR0.TEIE bit is set to 1b at this time, a TEI interrupt request is generated. The SCKm pin is held high.

Note 1. The number of unsent transmit data stored in the TDR register (transmit FIFO) can be monitored by reading the FTSR.T[5:0] bits.



[1] Start transmission process

[2] Write transmit data to transmit FIFO (TDR) by using TXI:
The transmit-data-empty-interrupt (TXI) is generated when transmission-data forwards to TSR from the transmit FIFO (TDR) and the number of transmit-data written in the transmit FIFO is equal to or less than the transmit FIFO threshold.

[3] Serial transmission continuation procedure:
To continue serial transmission, write transmit data to the transmit FIFO (TDR) by using a TXI interrupt and clear TDRE flag to 0.
The number of transmission data it's possible to write in is N_{FIFO} (= 32) - (the number of stored transmit FIFO data).
Transmit data can also be written to the transmit FIFO (TDR) by activating the DMAC. In this case, the TDRE flag is cleared automatically, so please do not operate the TDRE flag.
If you use TEI, set TIE = 0 and TEIE = 1 after writing the last-transmit-data to TDR.

Note: When the external clock is in use (the value of the CKE[1:0] is 10b or 11b), the rising edge on the SCK pin for the last bit sets the TEND flag to 1b. Setting TE to 0b immediately after this, then it may lead to insufficient received data hold time on the receiver side.

Figure 7.3-87 Serial Transmission Flowchart Example in Clock Synchronous Mode (FIFO Buffer Selected)

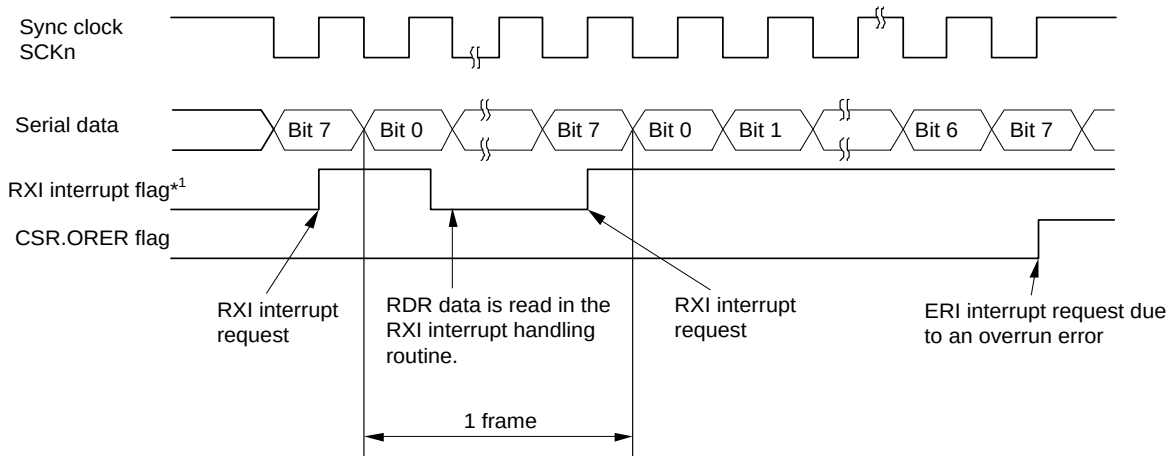
7.3.8.5 Serial Data Reception (Clock Synchronous Mode)

(1) Non-FIFO selected

Figure 7.3-88 and **Figure 7.3-89** show operation examples of serial data reception in clock synchronous mode.

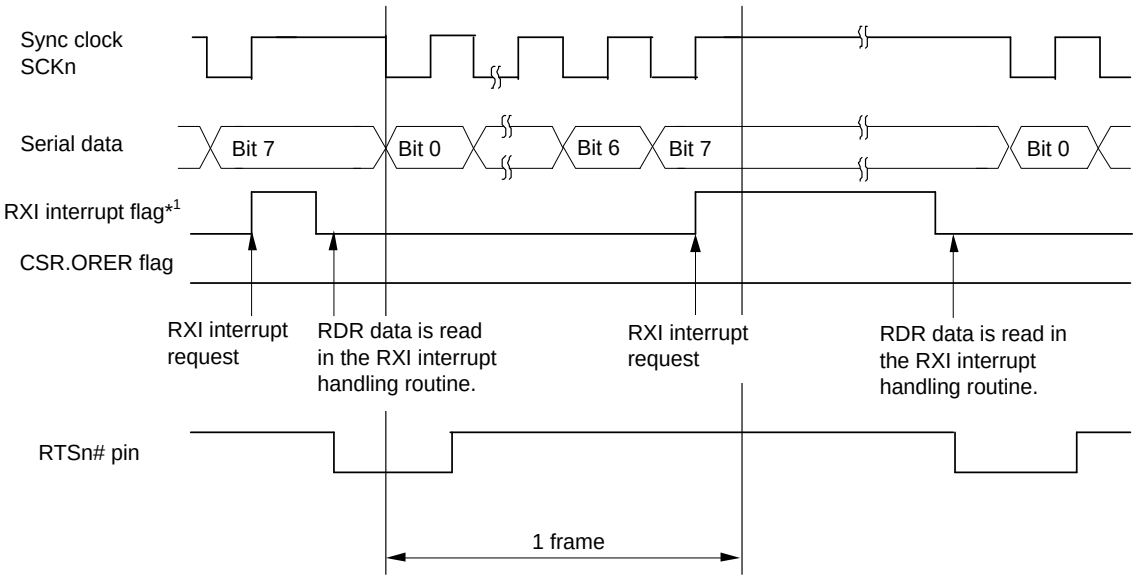
The SCI operates as follows during serial data reception. Reception-only operation is possible only in slave mode. (In master mode, reception-only operation is prohibited.)

1. When the CCR0.RE bit is set to 1b, the RTSn# pin output becomes low level (when the RTS function is used).
2. The SCI starts data reception in synchronization with input or output of the sync clock, and transfers receive data to the RSR register.
3. When an overrun error occurs, the CSR.ORER flag is set to 1b. When the CCR0.RIE bit = 1b at this time, an ERI interrupt request is generated and the received data is not transferred to the RDR register.
4. When data is normally received, the received data is transferred to the RDR register. When the RIE bit = 1b at this time, an RXI interrupt request is generated. Reading the received data transferred to the RDR register in the RXI interrupt handling routine before the next data is completely received enables continuous reception. When the received data transferred to the RDR register is read, the RTSn# pin output becomes low level (when the RTS function is used).
   If you want to prevent the RTSn# pin output from turning low level after the final data is received, clear the CCR0.RE bit to 0b and then read the RDR register.



Note 1. For corresponding interrupt source numbers, see **4.6 Interrupt Controller**.

Figure 7.3-88 Serial Data Reception Example in Clock Synchronous Mode (1) (RTS Function Not Used/CPHA = 1b, CPOL = 1b)

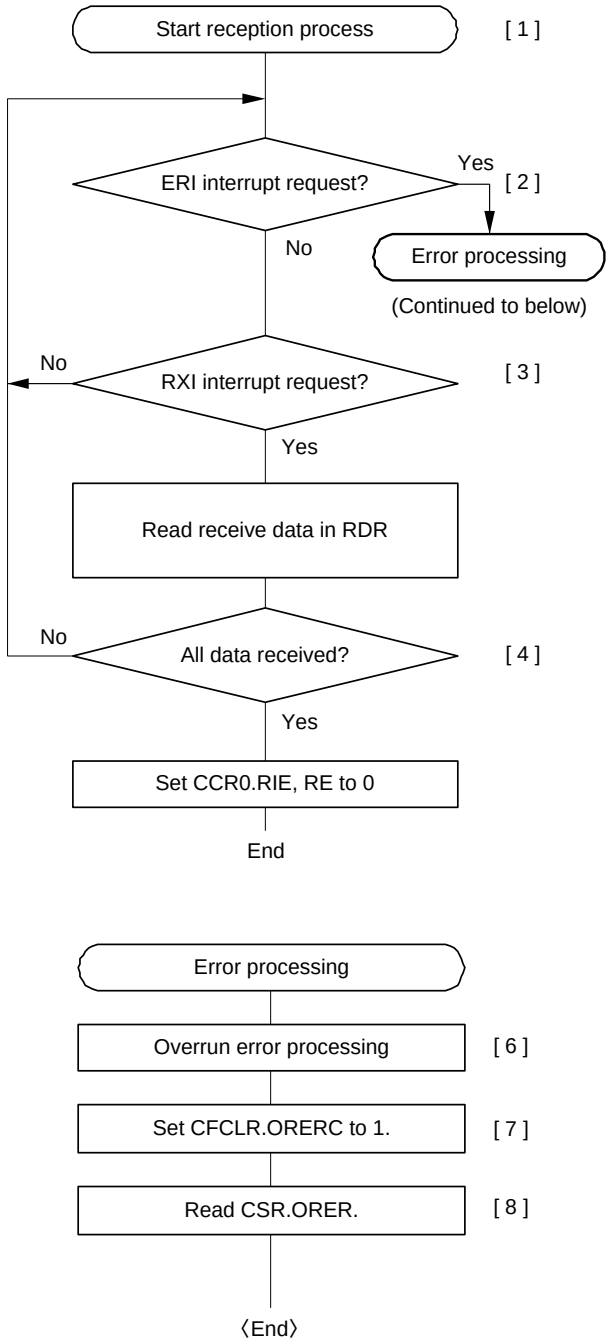


Note 1. For corresponding interrupt source numbers, see **4.6 Interrupt Controller**.

Figure 7.3-89 Serial Data Reception Example in Clock Synchronous Mode (2) (RTS Function Used/CPHA = 1b, CPOL = 1b)

While the reception error flag is set to 1b, subsequent reception is disabled. Therefore, before continuing reception, be sure to clear the ORER flag in CSR to 0b. Also be sure to read the RDR register in the overrun error processing. If the CCR0.RE bit is set to 0b during reception to forcibly terminate the reception operation, unread receive data may be remaining in the RDR register. In this case, read the RDR register.

Figure 7.3-90 shows an example of serial data reception flowchart.



[1] Start reception process

[2] Receive error processing:
If a receive error occurs, after performing the appropriate error processing, be sure to ORER is 0. Reception cannot be resumed if ORER is 1.

[3] Read the receive data in RDR once in the RXI interrupt handling routine.

[4] Serial reception continuation procedure:
To continue serial reception, before the MSB bit (bit 7) of the current frame is received, read data from RDR in the RXI interrupt handling routine. The RDR data can also be read by activating the DMAC.

[6] Overrun error processing:
Read RDR.

[7] Clear the error flag:
Set the flag clear bit corresponding to the reception error to 1. Done this step and step [6], the next frame can be received normally.

[8] Confirming that the error flag is cleared:
Confirm all error flags are cleared.

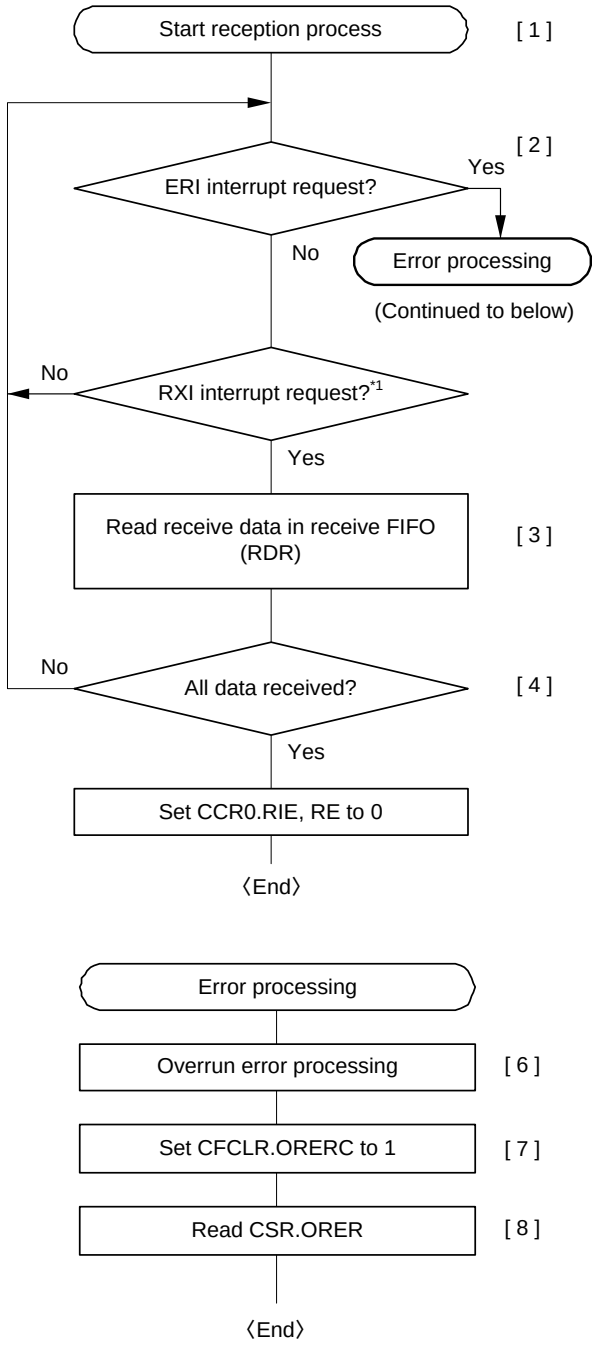
Figure 7.3-90 Sample Flowchart for Serial Reception in Clock Synchronous Mode (Non-FIFO Selected)

(2) FIFO selected

Figure 7.3-91 shows an example of serial data reception flowchart (a FIFO buffer selected) in clock synchronous mode. The SCI operates as follows during serial data reception. Reception-only operation is possible only in slave mode. (In master mode, reception-only operation is prohibited.)

1. When the CCR0.RE bit is set to 1b, the RTSn# pin output turns low level (when the RTS function is used).
2. The SCI starts receiving data in synchronization with input or output of the sync clock and transfers the received data to the receive FIFO (RDR register).
3. When an overrun error occurs, the CSR.ORER flag is set to 1b. When the CCR0.RIE bit = 1b at this time, an ERI interrupt request is generated and the received data is not transferred to the receive FIFO (RDR register)*1.
4. When data is normally received, the received data is transferred to the receive FIFO (RDR register)*1. When the number of receive data stored in the receive FIFO (RDR register) is equal to or more than the threshold value of the receive FIFO, the CSR.RDRF flag is set to 1b. When the RIE bit = 1b at this time, an RXI interrupt request is generated. Reading the received data transferred to the receive FIFO (RDR register) in the RXI interrupt handling routine before an overrun error occurs enables continuous reception. When the received data transferred to the receive FIFO (RDR register) is read and the number of data becomes lower than the RTS# output threshold value, the RTSn# pin output becomes low level (when the RTS function is used).

Note 1. The RDR.RDAT[8] bit is not used.



[1] Start reception process

[2] Receive error processing:
If a receive error occurs, after performing the appropriate error processing, be sure to ORER is 0. Reception cannot be resumed if ORER is 1.

[3] Read the receive data in the receive FIFO (RDR) in the RXI interrupt handling routine. The reception data stored in the receive FIFO (RDR) is read until the number of stored data will be below the FCR.RTRG value at least. You can confirm the number of the reception data in the FIFO by FRSR.R[5:0].

[4] Serial reception continuation procedure:
To continue serial reception, before the overrun error occurs, read data from RDR and clear RDRF flag to 0. The RDR data can also be read by activating the DMAC. In this case, the RDRF flag is cleared automatically, so please do not operate the RDRF flag.

Note 1. Set all received data to an integer multiple of the receive FIFO threshold. If it is not an integral multiple, return to step [2] and it will become an infinite loop, so set a timeout time separately using a timer etc.

[6] Overrun error processing:
Read RDR.

[7] Clear the error flag:
Set the flag clear bit corresponding to the reception error to 1. Done this step and step [6], the next frame can be received normally.

[8] Confirming that the error flag is cleared:
Confirm all error flags are cleared.

Figure 7.3-91 Example of Serial Reception in Clock Synchronous Mode (FIFO Selected)

7.3.8.6 Serial Data Transmission and Reception (Clock Synchronous Mode)

(1) Non-FIFO selected

Figure 7.3-92 shows an example of serial data concurrent transmission/reception flowchart in clock synchronous mode. After the SCI is initialized, perform the following procedure for serial data concurrent transmission/reception. When switching mode from transmission to concurrent transmission/reception, check that the CSR.TEND flag is set to 1b to ensure that the SCI is in the transmission complete state. Then set CCR0.TE = 0b and RE = 0b and then set the TE, RE, TIE, and RIE bits in CCR0 to 1b simultaneously by a single instruction.

When switching mode from reception to concurrent transmission/reception, check that the SCI is in the reception complete state, and then set CCR0.TE = 0b and RE = 0b. After that, check that the error flags (ORER, FER, and PER) in CSR are cleared to 0b, and then set the TE, RE, TIE, and RIE bits in CCR0 to 1b simultaneously by a single instruction.

When the RTS function is used in the concurrent transmission/reception operation, if you want to prevent the RTSn# pin output from turning to low after the final data is received as in the reception operation, clear the RE and TE bits in CCR0 to 0b simultaneously, and then read the RDR register.

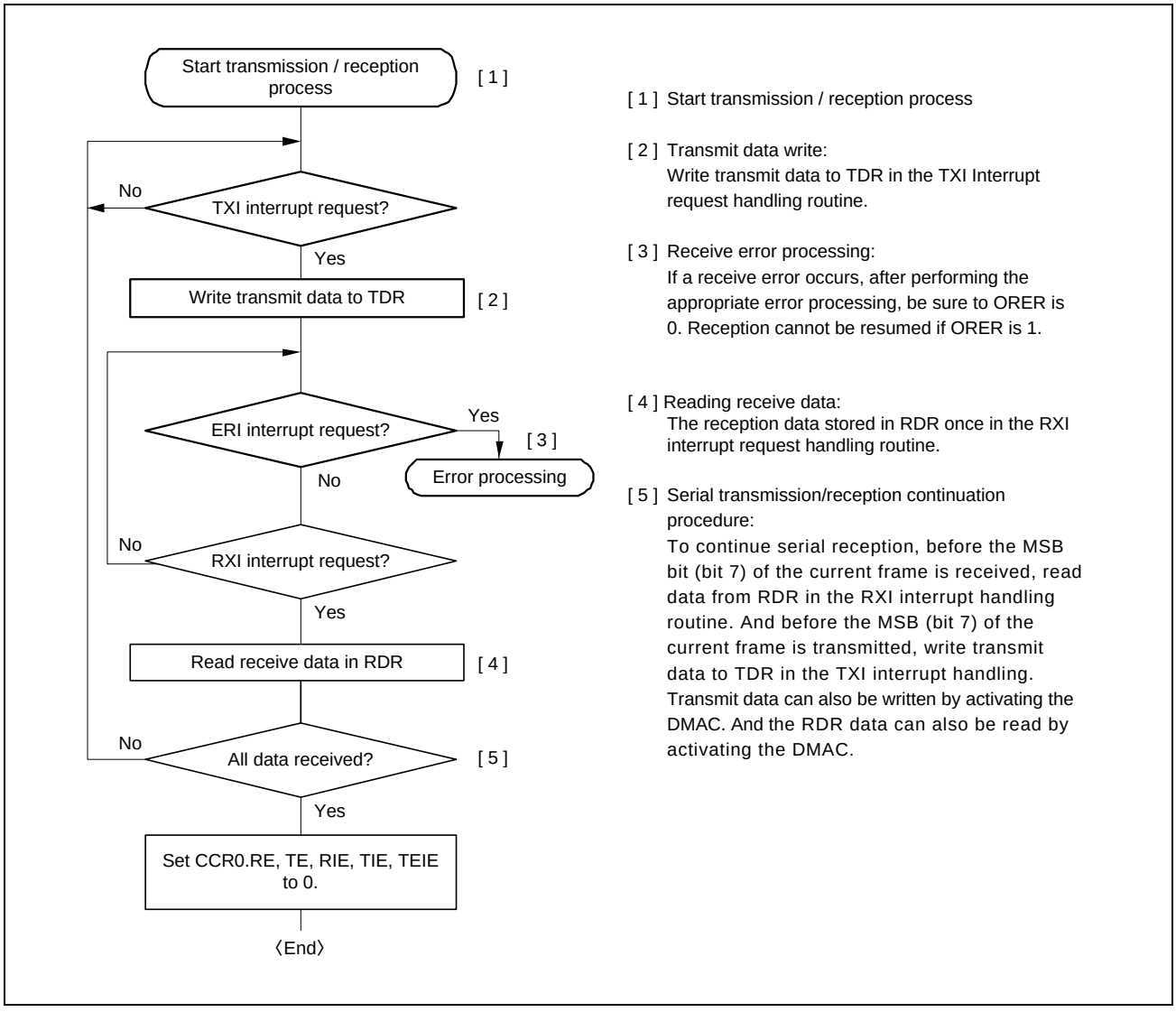


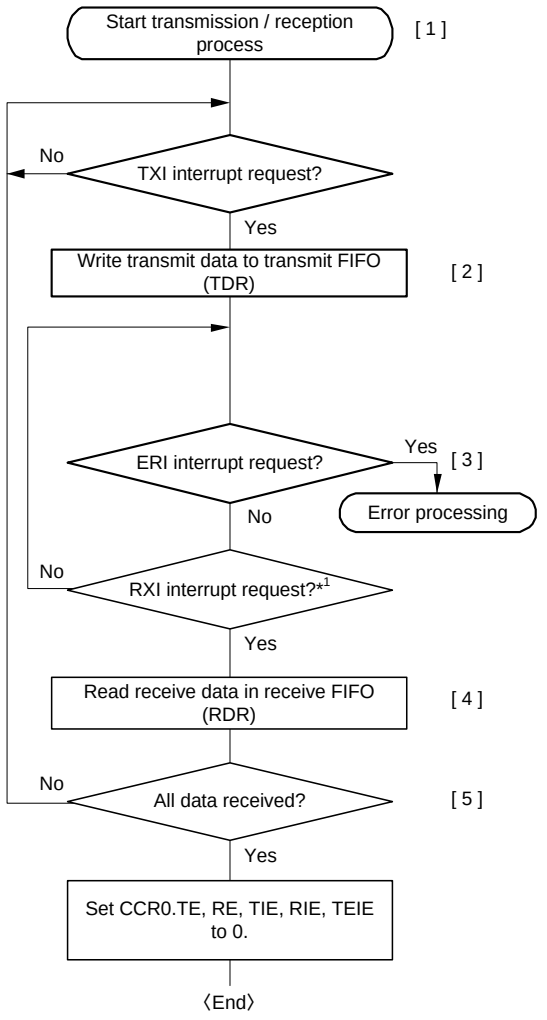
Figure 7.3-92 Serial Data Concurrent Transmission/Reception Flowchart Example in Clock Synchronous Mode (Non-FIFO Selected)

(2) FIFO selected

Figure 7.3-93 shows an example of serial data concurrent transmission/reception flowchart (a FIFO buffer selected) in clock synchronous mode.

After the SCI is initialized, perform the following procedure for serial data concurrent transmission/reception. When switching mode from transmission to concurrent transmission/reception, check that the CSR.TEND flag is set to 1b to ensure that the SCI is in the transmission complete state. Then set CCR0.TE = 0b and RE = 0b and then set the TE, RE, TIE, and RIE bits in CCR0 to 1b simultaneously by a single instruction.

When switching mode from reception to concurrent transmission/reception, check that the SCI is in the reception complete state, and then set CCR0.TE = 0b and RE = 0b. After that, check that the error flags (ORER, FER, and PER) in CSR are cleared to 0b, and then set the TE, RE, TIE, and RIE bits in CCR0 to 1b simultaneously by a single instruction.



[1] Start transmission / reception process

[2] Transmit data write:
Write transmit data to transmit FIFO (TDR) in the TXI Interrupt request handling routine.
The number of transmission data it's possible to write in is N_{FIFO} (= 32) - (the number of stored transmit FIFO data).

[3] Receive error processing:
If a receive error occurs, after performing the appropriate error processing, be sure to ORER is 0. Reception cannot be resumed if ORER is 1.

[4] Read receive data:
Read the receive data in the receive FIFO (RDR) in the RXI interrupt handling routine. The reception data stored in the receive FIFO (RDR) is read until the number of stored data will be below the FCR.RTRG value at least. You can confirm the number of the reception data in the FIFO by FRSR.R[5:0].

[5] Serial transmission/reception continuation procedure:
To continue serial reception, before the overrun error occurs, read data from RDR and clear CSR.RDRF.
To continue serial transmission, write transmit data to the transmit FIFO (TDR) by using a TXI interrupt and clear TDRE flag to 0.
Transmit data can also be written to the transmit FIFO (TDR) by activating the DMAC. And The RDR data can also be read by activating the DMAC. In this case, the TDRE and RDRF are cleared automatically, so please do not operate the TDRE and RDRF.

Note 1. Set all received data to an integer multiple of the receive FIFO threshold. If it is not an integral multiple, return to step [2] and it will become an infinite loop, so set a timeout time separately using a timer etc.

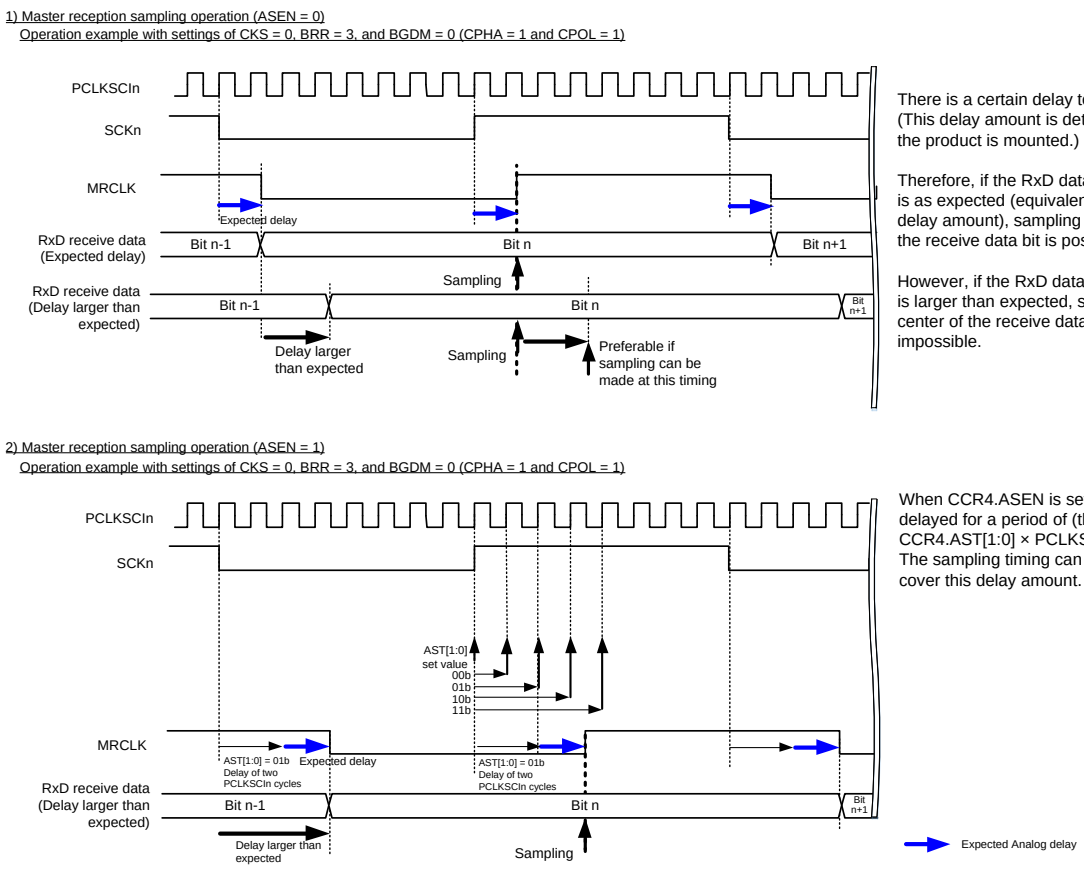
Figure 7.3-93 Serial Data Concurrent Transmission/Reception Flowchart Example in Clock Synchronous Mode (FIFO Selected)

7.3.8.7 Reception Sampling Timing Adjustment Function in Clock Synchronous Mode with Internal Clock Used

When the clock synchronous internal clock is used (master mode), MRCLK is used as a reception sampling clock.

This function adjusts the reception sampling timing by delaying MRCLK by 1 to 4 RSCI_m_TCLK and adding a digital delay.

Setting the CCR4.ASEN bit to 1b enables this function. The delay value is set in CCR4.AST[1:0].



Note: This timing chart illustrates the reception sampling timing adjustment function.

Figure 7.3-94 Reception Sampling Timing Adjustment Operation in Clock Synchronous Mode (Master)

7.3.9 Simple-SPI Mode

As an extended function of the SCI, the SCI supports Simple-SPI (4-wire serial bus) mode that allows communication from one or more master devices to multiple slave devices.

Simple-SPI mode is enabled by the Simple-SPI mode setting (CCR3.MOD[2:0] = 011b) and by setting the CCR0.SSE bit to 1b. When using the 4-wire serial bus in master mode with a single master, the SS pin function on the master side is not required. Therefore, the CCR0.SSE bit is set to 0b.

Figure 7.3-95 shows an example of Simple-SPI connection.

In simple-SPI mode, data is transmitted and received in synchronization with clock pulses in the same way as clock synchronous mode. A character of communication data consists of 8-bit data. No parity bit can be attached.

Because the SCI has an internal transmitter and a receiver independently, the transmitter and the receiver can share a clock to enable full-duplex communication. Furthermore, because both the transmitter and the receiver have a double-buffer structure, continuous transmission and reception are possible by writing the next transmit data during transmission and reading the previous receive data during reception.

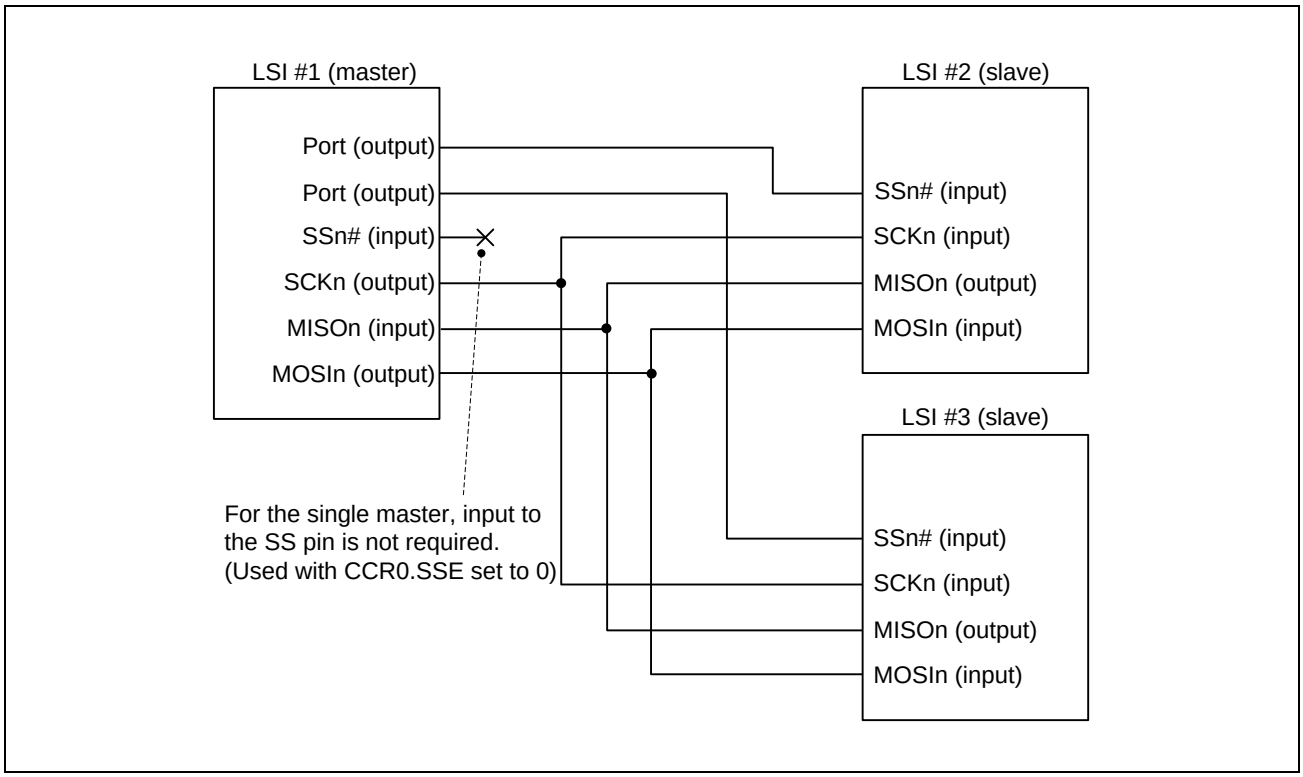


Figure 7.3-95 Simple-SPI Connection Example

7.3.9.1 Pin Status in Master and Slave Mode

In simple-SPI mode, input and output directions of each pin vary depending on master mode (CCR3.CKE[1:0] = 00b or 01b) and slave mode (CCR3.CKE[1:0] = 10b or 11b).

Table 7.3-30 shows the relationship among mode, SSn# pin input, and each pin state.

Table 7.3-30 Relationship among Mode, SSn# Pin Input, and Each Pin State

Mode	SSn# Pin Input	TXDn (MOSIn) Pin State	RXDn (MISOn) Pin State	SCKm Pin State
Master mode*1	High level (Communication enabled)	Transmit data output*2	Receive data input	Clock output*3
	Low level (Communication disabled)	High impedance	Receive data input (disabled)	High impedance
Slave mode	High level (Communication disabled)	Receive data input (disabled)	High impedance	Clock input (disabled)
	Low level (Communication enabled)	Receive data input	Transmit data output*2	Clock input

Note 1. When single master is selected (SSE bit = 0b), communication is enabled (equivalent to the state when the SSn# pin input is high level) regardless of the SSn# pin input level. The SSn# pin is not used and is available for other purposes.

Note 2. High impedance when transmission is disabled (CCR0.TE bit = 0b)

Note 3. High impedance when multi-master (SSE bit = 1b) is selected and transmission/reception is disabled (CCR0.TE, RE bits = 00b)

7.3.9.2 SS Function in Master Mode

Setting the CCR3.CKE[1:0] bits to 00b or 01b enables master mode.

In single master mode (SSE bit = 0b), the SSn# pin is not used and data transmission and reception are enabled regardless of the SSn# pin input level. The SSn# pin is available for other purposes.

When in multi-master mode (SSE bit = 1b) and the SSn# pin input is high, the master outputs a clock from the SCKm pin and performs transmission and reception operations. And outputting clock indicates “There are no other masters” or “Another master is not performing reception or transmission”. When the SSn# pin input level is low in multi-master mode (SSE bit = 1b), this means that another master exists and it is performing data transmission/reception. At this time, the SCI makes the TXDn pin output and the SCKm pin output high impedance and does not start data transmission/reception. In addition, the CSR.MFF bit is set to 1b as a mode fault error. In multi-master mode, read this flag bit to perform the error processing. If a mode fault error occurs during the transmission/reception operation, the SCKm pin and the TXDn pin output are made high impedance while the SSn# pin input level is low. When the SSn# pin input becomes high level, the SCKm pin outputs a clock signal and the TXDn pin outputs data. Even if the SCKm pin and the TXDn pin are in the high impedance state, internal transmission/reception operation continues, but it stops after transmission/reception of a single character is complete. In this case, any of TXI, RXI, and TEI interrupts occurs.

Control the SSn# pin output in master mode with a general-purpose port.

7.3.9.3 SS Function in Slave Mode

Setting the CCR3.CKE[1:0] bits to 10b or 11b enables slave mode.

When the SSn# pin input level is high, the RXDn pin output becomes high impedance and the clock input from the SCKm pin is ignored. When the SSn# pin input level is low, the clock input from the SCKm pin becomes effective, enabling data transmission and reception.

When the SSn# pin input changes from low to high level during the transmission/reception operation, the RXDn pin output is made high impedance and the transmission/reception operation is immediately suspended. If the transmission is in progress, the CSR.TEND flag will not be set, a transmit end interrupt will not be output, and an abnormal stop status will occur. So, do not negate the SSn# pin during slave transmission/reception. If an abnormal stop occurs, set CCR0.RE and CCR0.TE to 0b to stop transmission / reception. To resume transmission/reception, set CCR0.RE and CCR0.TE to 1b after at least RSCI_m_TCLK × 3 cycles + RSCI_m_PCLK × 3 cycles.

7.3.9.4 Relationship between Clock and Transmit/Receive Data

The clock to be used for data transmission/reception is selectable from four types using the CCR3.CPOL and CPHA bits. **Figure 7.3-96** shows the relationship between clock and transmit data/receive data. The same relationship applies to master mode and slave mode.

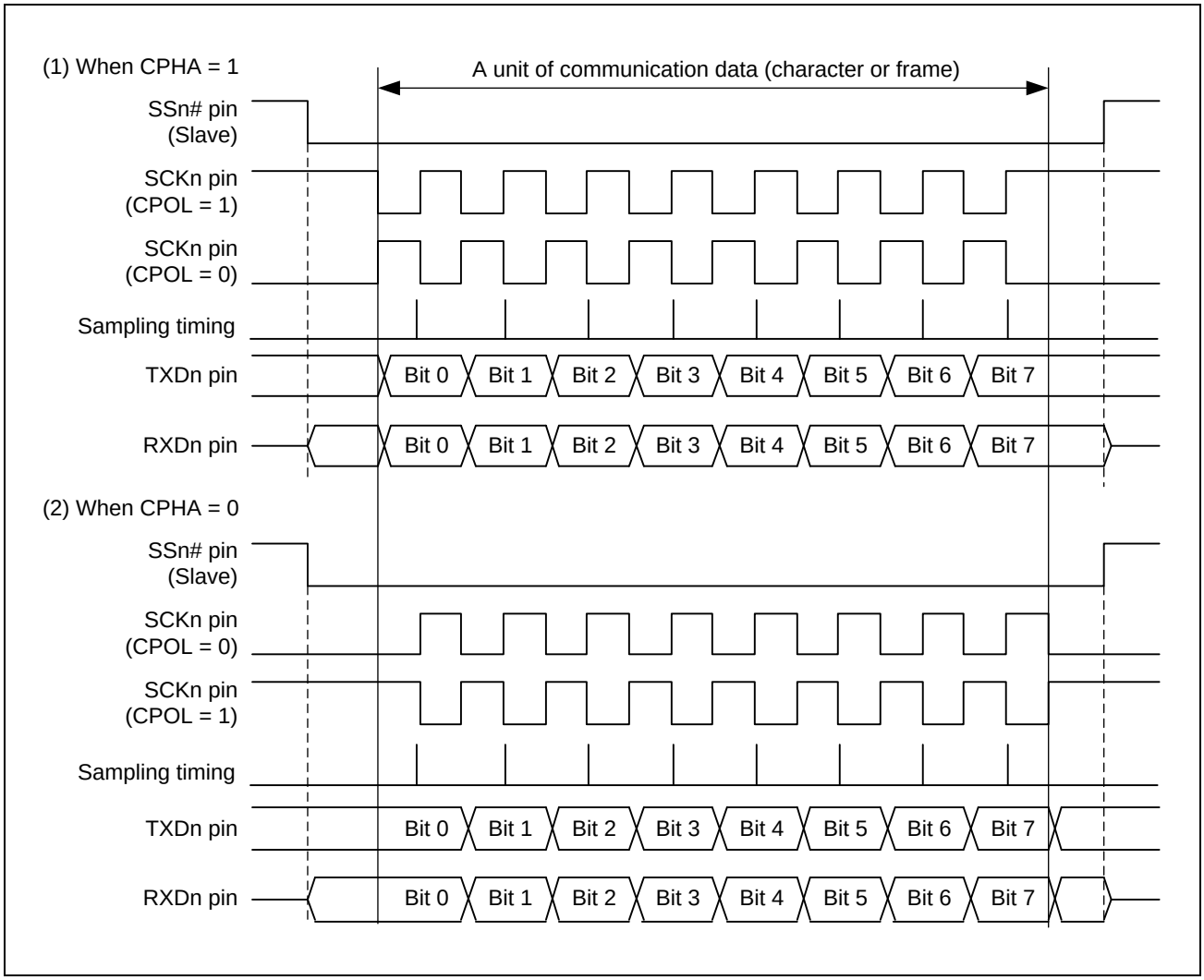


Figure 7.3-96 Relationship between Clock and Transmit Data/Receive Data in Simple-SPI Mode

7.3.9.5 SCI Initialization in Simple-SPI Mode

The SCI can be initialized using the same initialization procedure as for clock synchronous mode (**Figure 7.3-82**). The master devices and slave devices use the same clock type selected by the CCR3.CPOL and CPHA bits.

Before performing initialization or changing operating mode or communication format, be sure to stop communication (CCR0.RE = 0b and CCR0.TE = 0b).

Note that setting the RE bit to 0b does not initialize the ORER, FER, and PER flags in CSR and the RDR register.

Note that, when CCR0.TIE = 1b, setting the TE bit to 1b from 0b generates a TXI interrupt.

7.3.9.6 Serial Data Transmission and Reception in Simple-SPI Mode

In master mode, set the SSn# pin of the destination slave device to low level before starting data transmission/reception and set to high level after the end of data transmission/reception. In multiple master operation with CCR0.SSE = 1b even in master mode, a mode fault error will occur if the SSn# pin goes low. Therefore, make sure that no mode fault error has occurred before starting communication, and start communication, and make sure that no mode fault error has occurred even after communication ends. If a mode fault error has occurred, communication may be incomplete, so measures such as retransmission are required. The other procedures are the same as in clock synchronous mode.

In slave mode, it operates according to the SSn# pin input level. Other steps are the same as those of clock synchronous mode.

7.3.9.7 Reception Sampling Timing Adjustment Function in Simple-SPI Mode with Internal Clock Used

The reception sampling timing adjustment function in simple SPI mode is the same as the reception sampling timing adjustment function in clock synchronous mode. For the description of operation, see **7.3.8.7 Reception Sampling Timing Adjustment Function in Clock Synchronous Mode with Internal Clock Used**.

7.3.10 Bit Rate Modulation Function

Using the bit rate modulation function, the bit rate can be corrected by evenly enabling the clocks of the number specified in the CCR2.MDDR[7:0] bits among 256 clock cycles of internal clocks which is selected by the CKS[1:0] bits in CCR2.

Figure 7.3-97 shows an example where the RSCI_m_TCLK is selected by the CKS[1:0] bits in CCR2 and the BRR and MDDR are set to 0 and 160 respectively in asynchronous mode. In this example, the cycle of the base clock is evenly corrected (256/160) and the bit rate is also corrected (160/256). The enable of internal clock has non-balanced. So, notice the pulse width of the internal base clock is caused bias.

Do not use this function in Clock Synchronous mode, Simple-SPI mode, Smart Card Interface mode and Simple-LIN mode.

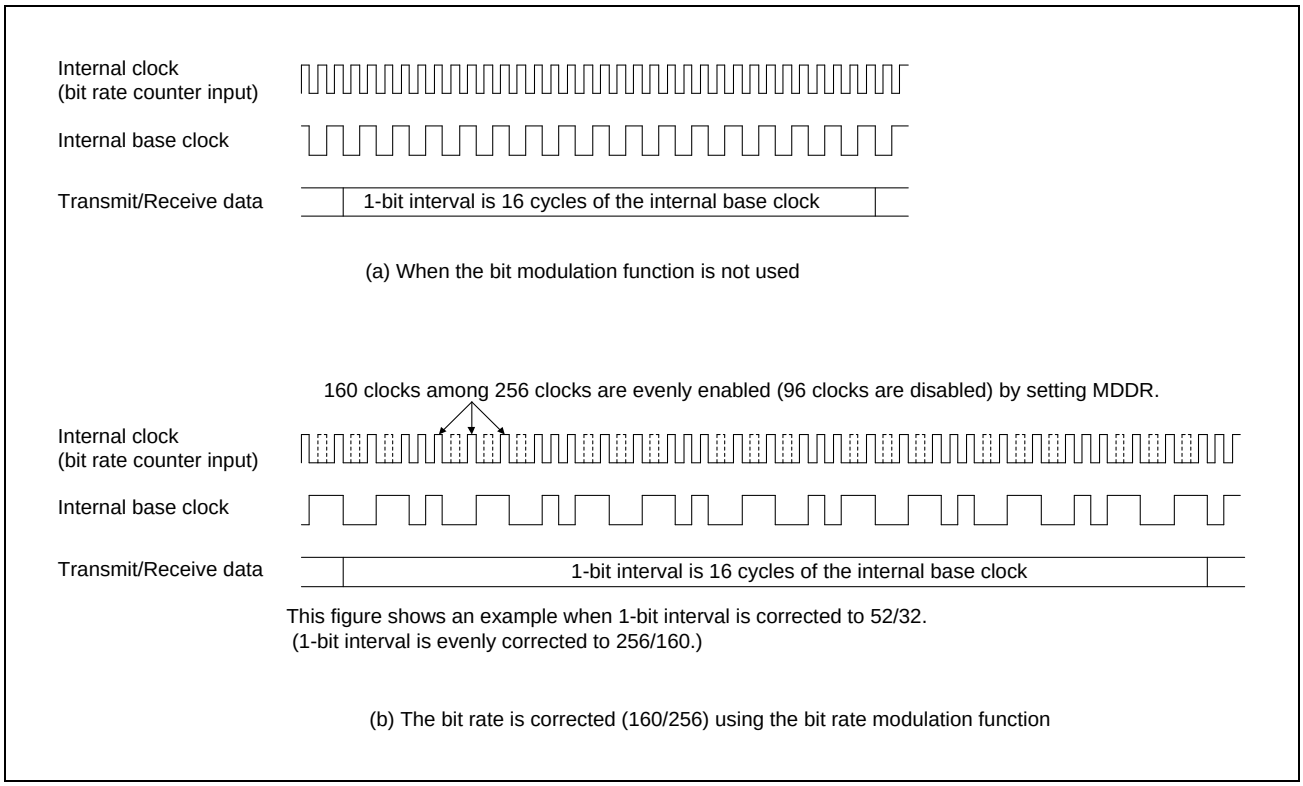


Figure 7.3-97 Example of Internal Base Clock when Bit Rate Modulation Function is Used

7.3.11 Noise Cancellation Function

Figure 7.3-98 shows the configuration of the noise filter used for noise cancellation. The noise filter consists of a two-stage flip-flop circuits and a match detection circuit. When the input signals of the noise filter and the output signals of the two- stage flip-flop circuits completely match, the matched level is conveyed as an internal signal. Unless otherwise matched, the previous value is retained. (When the same level is retained for three cycles or longer on the sampling clock of the noise filter, it is considered as a valid receive signal. A change in pulse for three cycles or shorter is considered as a noise, not as a receive signal.)

In asynchronous mode and simple-LIN mode, the noise cancellation function can be applied to the receive signal input to the RXDn pin. The sampling period of the noise filter can be selected from the base clock period and the divided clock of the baud rate generator clock source by CCR1.NFCS[2:0].

(When CCR1.NFCS[2:0] = 000b, CCR2.ABCS = 0b and CCR2.ABCSE = 0b, the cycle is 1/16 of a period 1 transfer bit.
When CCR1.NFCS[2:0] = 000b, CCR2.ABCS = 1b and CCR2.ABCSE = 0b, the cycle is 1/8 of a period 1 transfer bit.
When CCR1.NFCS[2:0] = 000b, CCR2.ABCSE = 1b, the cycle is 1/6 of a period 1 transfer bit.)

In simple I2C mode, the noise elimination function can be used for the input pins of SDAn and SCLn. The sampling period of the noise filter can be selected from the divided clock of the baud rate generator clock source by CCR1.NFCS[2:0].

If the base clock is stopped once with the noise filter enabled and then the base clock input is restarted again, the noise filter operation resumes from the state where the clock was stopped. When CCR0.TE and CCR0.RE are set to 0b during base clock input, all of the noise filter flip-flop values are initialized to 1. Accordingly, if the input data is 1 when reception operation resumes, it is determined that a level match is detected and is conveyed as an internal signal. When the level being input corresponds to 0b, the initial output of the noise filter is retained until the level matches in three consecutive sampling cycles.

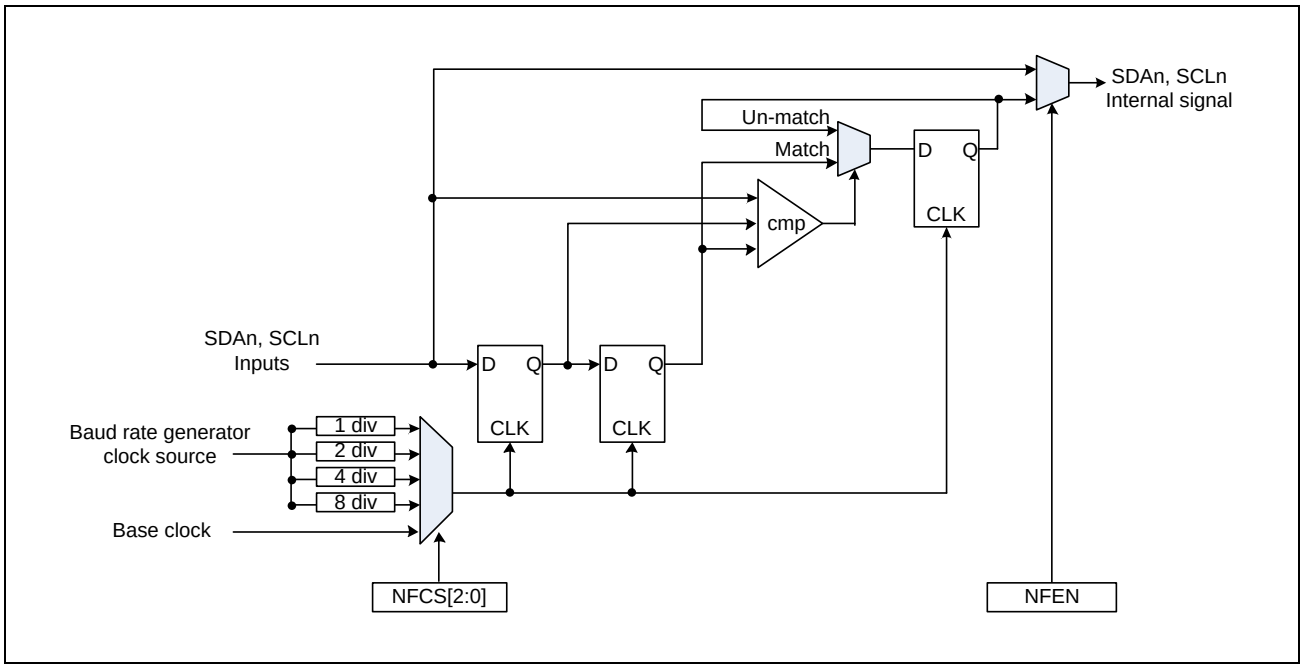


Figure 7.3-98 Block Diagram of Digital Noise Filter Circuit

7.3.12 RS-485 Driver Control Function

Setting the DEN bit in the SCI Common Control Register 3 (CCR3) to 1b enables the RS-485 driver control function and generates a DE (Driver Enable) signal that enables the external transceiver transmission mode. The DE signal outputs a valid level for the period with Driver assertion time and Driver negation time added before and after data transmission. The DE signal valid level is set by the DEPOL bit in the driver control register (DCR).

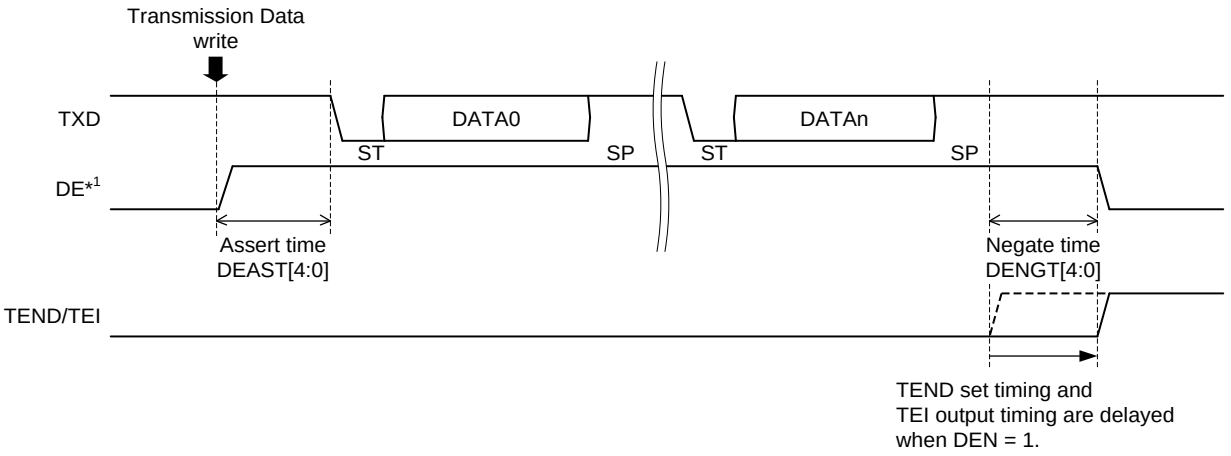
The Driver assertion time is the time from when the DE signal is valid until the start bit starts. Set by DEAST[4:0] of driver control register (DCR).

The Driver negation time is the time from the end of the last stop bit of the transmitted message to the invalidation of the DE signal. Set with DENGT[4:0] of the driver control register (DCR).

DEAST and DENGT are expressed in base clock units. For details, see **7.3.2.2.14 Driver Control Register (RSCIm_DCR)**.

When this function is used (DEN = 1b), the TEND set timing and TEI interrupt output timing are at the end of the driver negation time.

When transmission is completed and the next transmission data is not written before the DE signal is negated, the DE signal is negated once. If the timing for writing the next transmit data is not in time, assert the DE signal after negating it again, insert the Driver assertion time, and transmit the next data. If you want to perform the next transmission with the DE signal asserted, write the next transmission data to the TDR quickly enough in consideration of the synchronization delay time of the register.



Note: ST = Start bit
SP = Stop bit

Note 1. When DCR.DEPOL = 0b

Figure 7.3-99 The Image Waveform for RS-485 Driver Control DE Signal Output

7.3.13 Loopback Function

The loopback function can be used in asynchronous mode with the internal clock, and clock synchronous mode with the internal clock.

When 1b is written to the SPLP bit in the CCR1 register, SCI blocks the external input (RXD) path and connects the output path of the transmit data register and the input path of the receive data register.

When this function is used with TINV = 1b, inversion of transmission data becomes reception data. However, this function can be used with TINV = 1b only when operating in clock synchronous mode internal clock.

Table 7.3-31 shows the relationship between the TINV and SPLP bit settings and the received data.

Table 7.3-31 TINV and SPLP Bit Settings and Received Data

			Communication Mode	
TINV	SPLP	Receive Data	Async Internal Clock	Clock Sync Internal Clock
x	0	Receive Data from RXD pin	Possible	Possible
0	1	Transmit Data	Possible	Possible
1	1	Inverted transmit data	Impossible	Possible

Figure 7.3-100 shows the configuration of the shift register input/output path in loopback mode.

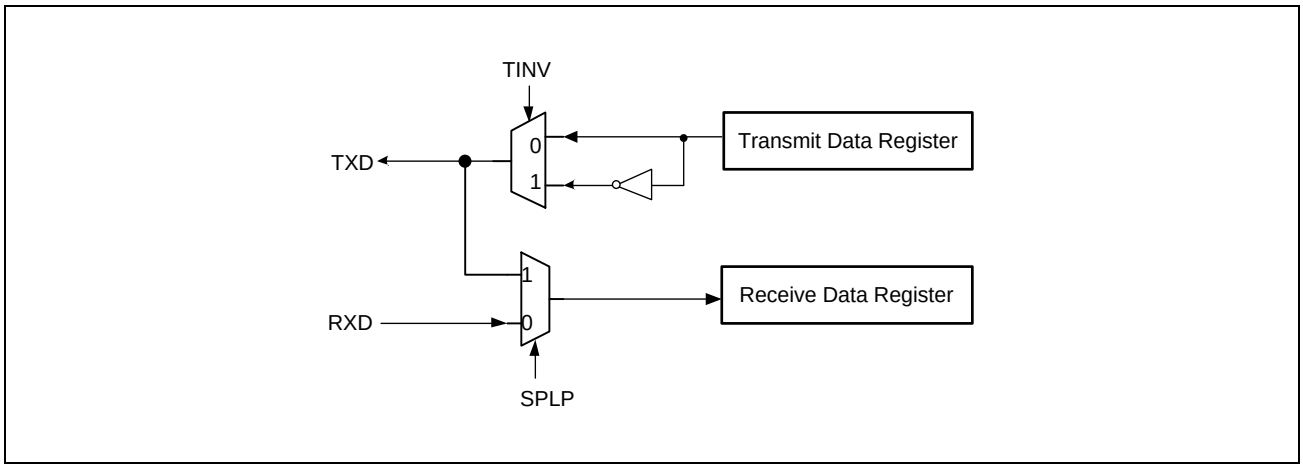


Figure 7.3-100 Shift Register Input/Output Configuration Image in Loopback Mode

7.3.14 Half-Duplex Communication Function

Do not use the half-duplex communication function in Simple-I2C mode, Simple-SPI mode, and Smart Card Interface mode.

In other communication modes, if the CCR1.SHARPS bit is set to 1b, half-duplex communication using the TXD pin is possible. When half-duplex communication is used, transmission and reception must be performed exclusively. Transmission and reception settings (CCR0.TE = 1b and CCR0.RE = 1b) is prohibited.

However, if half-duplex communication is performed as the master in clock synchronous mode, perform transmission/reception settings (CCR0.TE = 1b and CCR0.RE = 1b) and perform dummy transmission. By dummy transmission (arbitrary transmission data is written to TDR), SCKm is output and reception is enabled. The dummy transmission data is discarded inside the module and is not actually transmitted.

During half-duplex communication, the only communication port terminal used is the TXD pin. Output when CCR0.TE = 1b, input when CCR0.TE = 0b.

7.3.15 Synchronizer Bypass Function

This module has a bus clock (RSCI_m_PCLK) and the operation clock (RSCI_m_TCLK). And these have each operating circuit. Therefore, there is a synchronization circuit for signal transfer between different clocks, and synchronization delay time is required for signal propagation between different clocks.

However, the synchronization circuit can be bypassed by the CCR3.BPEN bit only when the same clock is input to the bus clock and the operation clock. In this case, eliminates synchronization delay time and improves responsiveness. **Figure 7.3-101** shows the image waveform of the bypass function.

This module also has a synchronization circuit between the communication clock (SCK) and the operation clock (RSCI_m_TCLK), but this synchronization circuit cannot be bypassed.

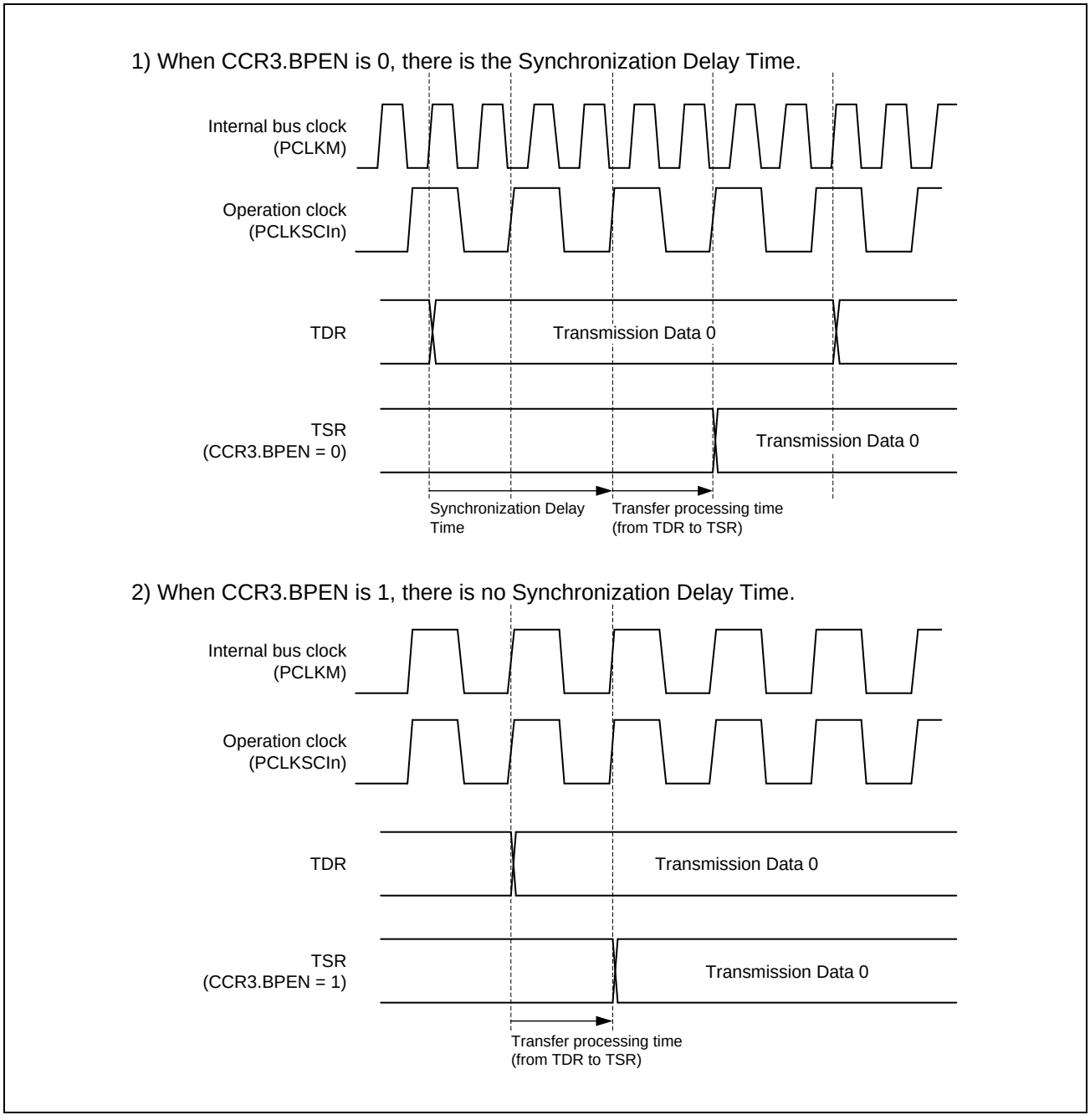


Figure 7.3-101 Image Waveform of Synchronizer Bypass Function

7.3.16 Half Data Communication Function

Setting the HCR.HDEN bit to 1b supports half data communication in which data 0 is pulsed low for the first half of a 1-bit period. Since this function operates only in the asynchronous mode, see the asynchronous mode for the setting, transmission, and reception flow.

7.3.16.1 Reception in Half Data Communication

When receiving during half data communication, the falling edge of the input from the RXDm_MISOm_SCLm pin is detected and the signal after the start bit is recognized is received. One frame is sampled according to the set bit rate, and if the stop bit is correctly received without error, the data value is stored in the receive data register RDR.

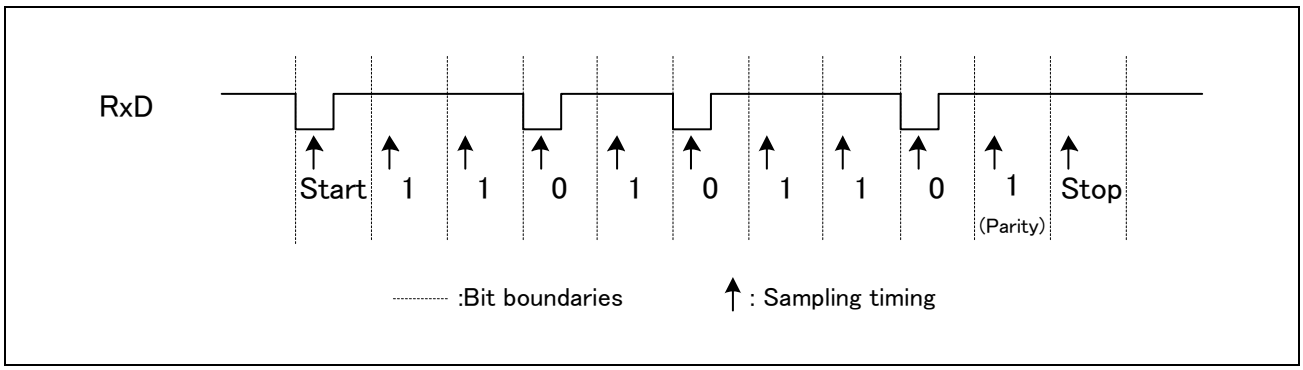


Figure 7.3-102 Half Data Communication Reception Timing Chart

For reception in half data communication, it is necessary to sample at the timing of 1/4 of 1 bit in order to capture the pulse in the first half of the 1 bit period. Sampling operates with a frequency 16 times the bit rate*1 as the base clock as in the asynchronous mode. The start bit is detected by detecting the Low level from the falling edge of RxD four times continuously with the base clock. When High is detected during communication, it is regarded as noise and waits for the next falling edge.

To set the sampling timing to 1/4 of the 1-bit period, enable the reception sampling timing adjustment function with the CCR4.ASEN bit, set the CCR4.AJD bit to 1b and the CCR4.AST bit to 100b, and adjust from the center of the bit, which is the previous sampling timing, four clocks ahead of the base clock.

Since the sampling timing can be adjusted backward and forward using the reception sampling timing adjustment function, this timing can be adjusted according to the reception status. Increasing the CCR4.AST bit value from 100b will move the sampling timing forward, and decreasing it will move it backward. See **7.3.3.10 The Function of Adjust Receive Sampling Timing (Asynchronous Mode)** for the details of adjustment.

After recognizing the start bit, sampling is performed at the timing according to the set bit rate, but the low width and high width of the waveform are not checked. Therefore, it is possible to receive even a normal asynchronous waveform.

Note 1. Half data communication function is supported only when CCR2.ABCS = 0b and CCR2.ABCSE = 0b.

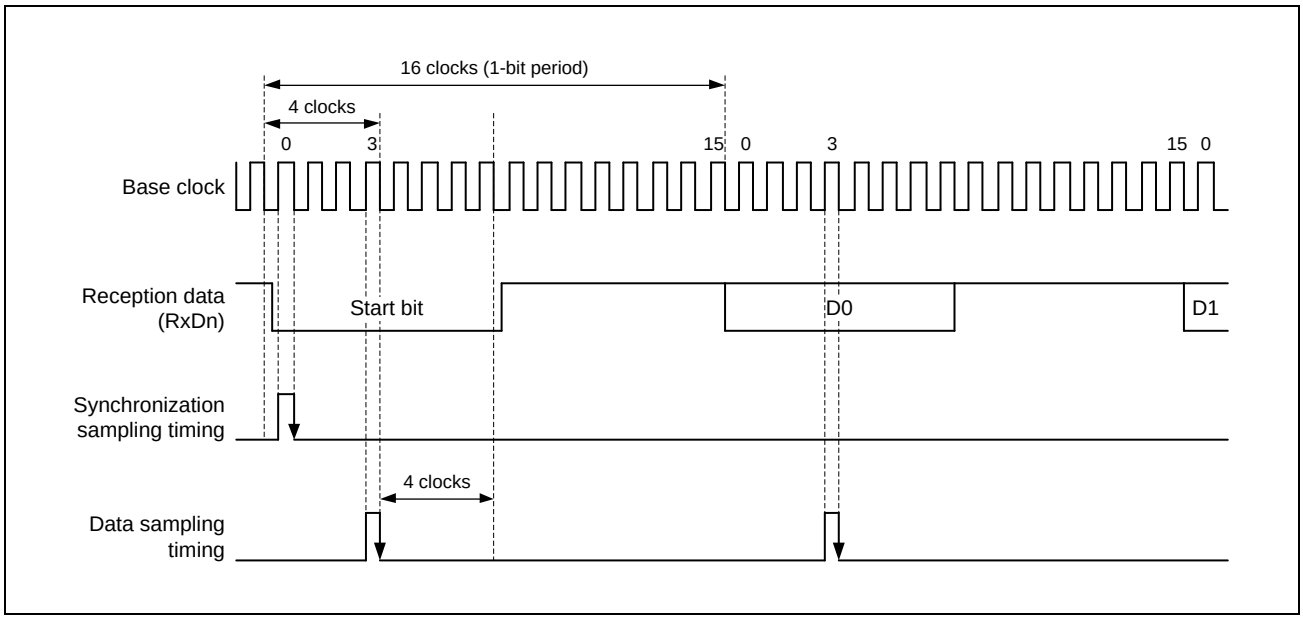


Figure 7.3-103 Details of Reception Sampling Timing during Half Data Communication

7.3.16.2 Transmission in Half Data Communication

In transmission during half data communication, data 0 is output as a low pulse for the first half of the 1-bit period.

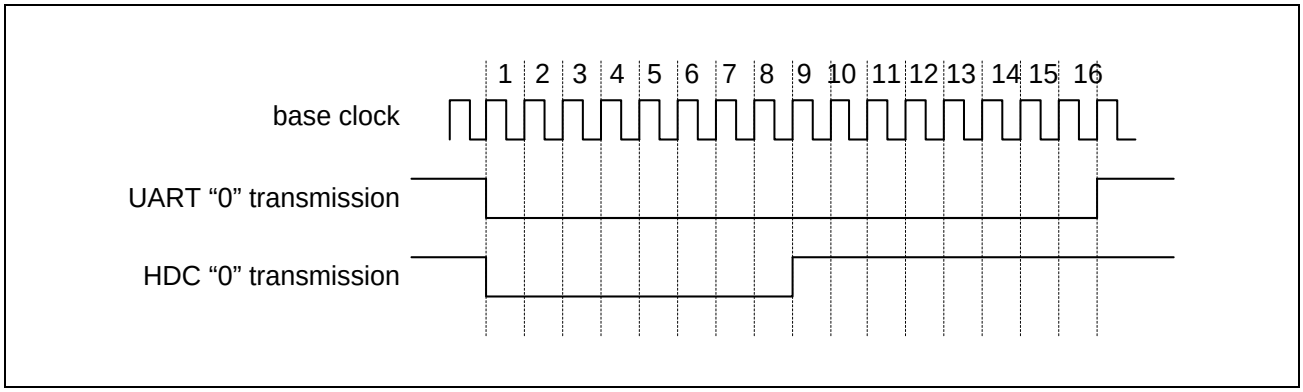


Figure 7.3-104 Transmission Waveform in Half Data Communication

Figure 7.3-105 shows an example of the transmission waveform.

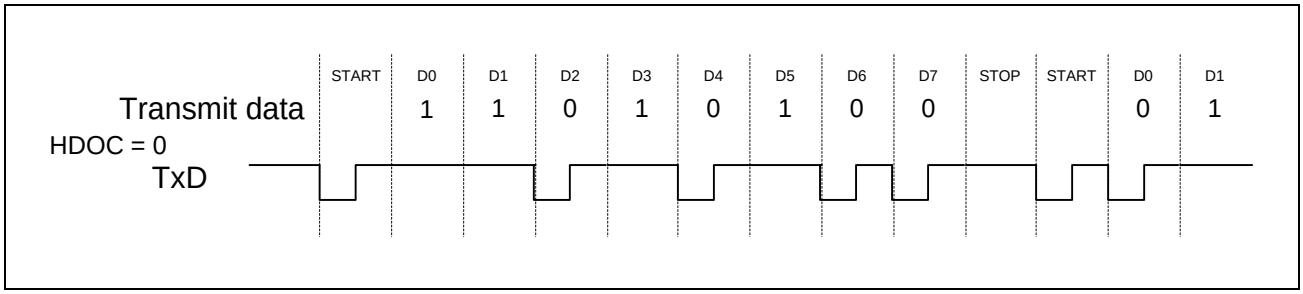


Figure 7.3-105 Difference in Half Data Transmission Waveform (when HDST = 0b, HDOC = 0b)

When CCR0.TE bit = 0b, TxD pins become Hi-Z, but can be controlled by the CCR1.SPB2IO bit and CCR1.SPB2DT bit. At this time, output is applied to the TxD pins.

7.3.16.3 Register Settings for Half Data Communication

The half data communication function is a part of asynchronous mode function, but some settings are not supported when using this function. Set each bit of the common control register as shown in **Table 7.3-32** before use. Register bits not listed can be set in the same way as in the asynchronous mode.

Table 7.3-32 Common Control Register Setting Values for Half Data Communication

Register.Bit Name	Value	Remarks
CCR0.DCME	0b	Use this setting when the address match function is to be disabled.
CCR1.NFCS[2:0]	000b	Use this setting when using the noise filter.
CCR1.SHARPS	0b	Half-duplex communication with the TxD terminal cannot be used.
CCR1.CTSE	0b	Use this setting to disable the CTS function.
CCR2.BRME	0b	Bit rate modulation function cannot be used.
CCR2.ABCSE	0b	Only 1 bit can be set for 16 cycles of base clock.
CCR2.ABCS	0b	Only 1 bit can be set for 16 cycles of base clock.
CCR3.CKE[1:0]	00b	Use this setting to enable internal clock and disable clock output.
CCR3.DEN	0b	Use this setting to disable the RS-485 driver function.
CCR3.FM	0b	Use this setting to disable the FIFO function.
CCR3.MOD[2:0]	000b	Set to asynchronous mode.
CCR3.RxDESEL	1b	Detect the start bit at the falling edge of the RxD pin input.
CCR3.STP	0b	Use this setting to select 1 stop bit.
CCR3.SINV	0b	Use this setting to disable data inversion.
CCR3.LSBF	1b	Use this setting for LSB first.
CCR3.CHR[1:0]	10b	Use this setting for the 8 bit length.
CCR4.AJD	1b	Use this setting when receiving half data communication.
CCR4.AST[2:0]	100b	Use this setting when receiving half data communication.*1
CCR4.ATEN	0b	Use this setting to disable the adjust transmit timing function.
CCR4.ASEN	1b	Use this setting when receiving half data communication.

Note 1. This is the central timing setting for half data. It can be adjusted if needed.

7.3.17 Interrupt Signal

Table 7.3-33 lists SCI interrupt signals.

When performing transmission and reception using DMAC, be sure to set DMAC first, and then set SCI. See **4.7 DMA Controller (DMAC)** for how to set DMAC.

Table 7.3-33 SCI Interrupt Sources

Name	ICU Input Name	Interrupt Sources	CA55 GIC Request	CM33 GIC Request	DMAC Activation
ERI	RSCI_CHm_INT_sc_eri_n	Error interrupt	Possible	Possible	Not possible
RXI	RSCI_CHm_INT_sc_rxi_n	Simple-I2C: Reception end interrupt Other mode: Receive data full interrupt	Possible	Possible	Possible*1
TXI	RSCI_CHm_INT_sc_txi_n	Simple-I2C and Smart Card Interface: Transmit end interrupt Other mode: Transmit data empty interrupt	Possible	Possible	Possible*1
TEI	RSCI_CHm_INT_sc_tei_n	Simple-I2C: Completion of generation of a start, restart, or stop condition (STI) Other mode: Transmit end interrupt	Possible	Possible	Not possible
AED	RSCI_CHm_INT_sc_aed_n	Active edge detection interrupt	Possible	Possible	Possible
BFD	RSCI_CHm_INT_sc_bfd_n	Break Field detection interrupt	Possible	Possible	Not possible

Note: m = 0 to 9

Note 1. Available interrupt sources change dependent on operation mode.

7.3.17.1 Buffer Operations for TXI and RXI Interrupts

The TXI and RXI interrupts have an interrupt buffer function. When the first interrupt request is generated during interrupt handling and the next interrupt request is generated (when the status flag of the Interrupt Controller Unit (ICU) is 1), the module does not output the interrupt request, and the SCI holds it internally. The interrupt that can be held is up to one.

7.3.17.2 Interrupt in Asynchronous Mode, Clock Synchronous Mode, and Simple-SPI Mode

A different interrupt vector is assigned to each interrupt source, and individual interrupt sources can be enabled or disabled with the enable bits in CCR0 register.

(1) Non-FIFO selected

Table 7.3-34 lists interrupt sources in Asynchronous Mode, Clock Synchronous Mode, and Simple-SPI Mode with Non-FIFO selected.

If the CCR0.TIE bit is 1b, a TXI interrupt request is generated when transmit data is transferred from the TDR to the TSR. At the start of transmission, a TXI interrupt request can also be generated by using a single instruction to set the CCR0.TE and CCR0.TIE bit to 1b at the same time. A TXI interrupt request can activate the DMAC to handle data transfer.

A TXI interrupt request is not generated by setting the CCR0.TIE bit to 1b while the setting of the CCR0.TE bit is 1b.*1

When the CCR0.TEIE bit is 1b, the CSR.TEND flag becomes 1b and the TEI interrupt request is generated if the next data is not written to the TDR register by the timing to transmit the last bit of transmission data. In addition, the TEND flag holds 1b during the period from setting the CCR0.TE bit to 1b until writing transmit data to the TDR register, and if the TEIE bit is set to 1b, a TEI interrupt request is generated.

Writing data to the TDR register clears the TEND flag and cancels the TEI interrupt request, but it takes time to cancel it.

If the CCR0.RIE bit is 1b, an RXI interrupt request is generated when received data is stored in the RDR. An RXI interrupt request can activate the DMAC to handle data transfer.

Setting of any from among the ORER, FER, PER flags in the CSR to 1b while the CCR0.RIE bit is 1b leads to the generation of an ERI interrupt request.

An RXI interrupt request is not generated at this time. Clearing all three flags (ORER, FER, PER) leads to discarding of the ERI interrupt request.

Note 1. To temporarily prohibit TXI interrupts at the time of transmission of the last of the data and so on when you wish a new round of transmission to start after handling of the transmit end interrupt, control prohibiting and permitting of the interrupt by using the interrupt request enable bit in the interrupt controller rather than using the TIE bit. This can prevent the suppression of TXI interrupt requests in the transfer of new data.

Table 7.3-34 SCI Interrupt Sources with Non-FIFO Selected

Name	Interrupt Source	Interrupt Flag	Level/Pulse Output	Interrupt Enable	DMAC Activation
ERI	Receive error	ORER, FER, PER, DFER, DPER	Level	RIE	Not possible
RXI	Receive data full	RDRF	Pulse	RIE	Possible
	Address match	DCMF			
TXI	Transmit data empty	TDRE	Pulse	TIE	Possible
	TE = 0 -> 1 detection				
TEI	Transmit end	TEND	Level	TEIE	Not possible

(2) FIFO selected

Table 7.3-35 lists interrupt sources in Asynchronous Mode, Clock Synchronous Mode, and Simple-SPI Mode with FIFO selected.

If the CCR0.TIE bit is 1b, a TXI interrupt request is generated when the stored number of data in transmit FIFO becomes the threshold value indicated in FCR.TTRG or below. A TXI interrupt request can also be generated by using a single instruction to set the CCR0.TE and CCR0.TIE bit to 1b at the same time. A TXI interrupt request is not generated by setting the CCR0.TIE bit to 1b while the setting of the CCR0.TE bit is 1b. If CCR0.TEIE bit is 1b, when the next data is not being written in transmit FIFO by the timing to which the last bit of the transmission data is sent, CSR.TEND flag will be 1b and TEI interrupt request is generated.

If the CCR0.RIE bit is 1b, RXI interrupt request is generated when the stored number of data in receive FIFO becomes the threshold value indicated in FCR.RTRG or above. When FCR.RTRG is set to 0b, RXI interrupt request is occurred if the quantity of data in receive FIFO is greater than or equal to 1.

If the CCR0.RIE bit is 1b, when CSR.ORER flag is set to 1b or the data a framing error or a parity error generated is stored in receive FIFO, ERI interrupt request is generated.

When the number of data stored in a receive FIFO at this time is a threshold value or above, RXI interrupt request is also generated. The ERI interrupt request can be canceled by clearing all flags (CSR.ORER, FER, and PER).

Table 7.3-35 SCI Interrupt Sources with FIFO Selected

Name	Interrupt Source	Interrupt Flag	Level/Pulse Output	Interrupt Enable	DMAC Activation
ERI	Receive error	ORER, FER, PER, DFER, DPER	Level	RIE	Not possible
		DR (FCR.DRES = 1b)			
RXI	Receive FIFO data full	RDRF	Pulse	RIE	Possible
	Receive data ready	DR (FCR.DRES = 0b)			
	Address match	DCMF			
TXI	Transmit FIFO data empty	TDRE	Pulse	TIE	Possible
	TE = 0 -> 1 detection				
TEI	Transmit end	TEND	Level	TEIE	Not possible

7.3.17.3 Interrupt in Smart Card Interface Mode

Table 7.3-36 lists interrupt sources in smart card interface mode. A transmit end interrupt (TEI) request cannot be used in this mode.

Table 7.3-36 SCI Interrupt Sources in Smart Card Interface Mode

Name	Interrupt Source	Interrupt Flag	Level/Pulse Output	Interrupt Enable	DMAC Activation
ERI	Receive error or error signal detection	ORER, PER, ERS	Level	RIE	Not possible
RXI	Receive data full	RDRF	Pulse	RIE	Possible
TXI	Transmit end	TEND	Pulse	TIE	Possible
	When set TE = 0 -> 1				

Data transmission/reception using the DMAC is also possible in smart card interface mode. In transmission operation, when the TEND flag in CSR is set to 1b, a TXI interrupt request is generated. This TXI interrupt request activates the DMAC allowing transfer of transmit data if the TXI request is specified beforehand as a source of DMAC activation. The TEND flag is automatically set to 0b when the DMAC transfers the data.

If an error occurs, the SCI automatically retransmits the same data. During the retransmission, the TEND flag is kept to 0b and the DMAC is not activated. Therefore, the SCI and DMAC automatically transmit the specified number of bytes, including retransmission in the case of error occurrence. However, the ERS flag in CSR is not automatically cleared to 0b at error occurrence. Therefore, the ERS flag must be cleared by previously setting the RIE bit in CCR0 to 1b to enable an ERI interrupt request to be generated at error occurrence.

In reception operation, an RXI interrupt request is generated when receive data is set to RDR. This RXI interrupt request activates the DMAC allowing transfer of receive data if the RXI request is specified beforehand as a source of DMAC activation. If an error occurs, the error flag is set. Therefore, the DMAC is not activated and an ERI interrupt request is issued to the CPU instead; the error flag must be cleared.

When transmitting/receiving data using the DMAC, be sure to make settings to enable the DMAC before making SCI settings. For DMAC settings, see **4.7 DMA Controller (DMAC)**.

7.3.17.4 Interrupts in Simple-I2C Mode

Table 7.3-37 lists SCI interrupts in simple-I2C mode.

The STI interrupt is allocated to the transmit end interrupt (TEI) request. The receive error interrupt (ERI) request cannot be used.

The DMAC can also be used to handle transfer in simple I2C mode.

When the value of the IICINTM bit in ICR is 1b, a RXI request will be generated on the falling edge of the SCLn signal for the eighth bit. If the RXI has been set up as an activating request for the DMAC beforehand, the RXI request will activate the DMAC to handle transfer of the received data. Furthermore, a TXI request is generated on the falling edge of the SCLn signal for the ninth bit (acknowledge bit). If the TXI has been set up as an activating request for the DMAC beforehand, the TXI request will activate the DMAC to handle transfer of the transmit data. (In this case, ACK/NACK judging are impossible.)

When the value of the IICINTM bit in ICR is 0b, SCI operates as follows.

RXI request (ACK detection) is generated if the input on the SDAn pin is at the low level on the rising edge of the SCLn signal for the ninth bit (acknowledge bit).

TXI request (NACK detection) is generated if the input on the SDAn pin is at the high level on the rising edge of the SCLn signal for the ninth bit (acknowledge bit).

If the RXI has been set up as an activating request for the DMAC beforehand, the RXI request will activate the DMAC to handle transfer of the received data. Also, if the DMAC is used for data transfer in reception or transmission, be sure to set up and enable the DMAC before setting up the SCI.

When the IICSTAREQ, IICRSTAREQ, and IICSTPREQ bits in ICR are used to generate a start condition, restart condition, or stop condition, the STI request is issued when generation is complete.

Table 7.3-37 SCI Interrupt Sources in Simple-I2C Mode

Name	Interrupt Source		Interrupt Flag	Level/Pulse Output	Interrupt Enable	DMAC Activation
	IICINTM = 1	IICINTM = 0				
RXI	Reception end	—	—	Pulse	RIE	Possible*1
	—	ACK detection	—			Possible
TXI	Transmit end	—	—	Pulse	TIE	Possible*1
	—	NACK detection	—			Possible
STI	Completion of generation of a start, restart, or stop condition		IICSTIF	Level	TEIE	Not Possible

Note 1. If the DMAC is being used, you cannot confirm whether ACK or NACK.

7.3.17.5 Interrupts in Simple-LIN Mode

Table 7.3-38 lists RSCI interrupts in simple-LIN mode.

Table 7.3-38 RSCI Interrupt Sources in Simple-LIN Mode

Name	Interrupt Sources	Interrupt Flag	Flag that Needs to be Confirmed	Level/Pulse Output	Interrupt Enable	DMAC Activation
ERR	Error	ORER,FER,PER	—	Level	RIE	Not Possible
		BCDF,COF	—		BCDIE,RIE, COFIE	
RXI	Reception Data full flag	RDRF	CF0MF, CF1MF, PIBDF	Pulse	RIE	XSR0.SFSF = 0b :Possible XSR0.SFSF = 1b: Not Possible
AED	Active edge detection	AEDF	—	Pulse	AEDIE	Possible
TXI	Transmit data empty interrupt	TDRE	—	Pulse	TIE	Possible
	When set TE = 0->1					
	Break Field output completion	BFOF	—		TIE, BFOIE	
TEI	Transmit end	TEND	—	Level	TEIE	Not Possible
BFD	Break Field Detection	BFDF	—	Level	BFDIE	Not Possible

In Simple-LIN mode, in addition to reception errors (ORER, FER, PER), an ERI interrupt request is output when a bus conflict is detected during transmission, or when a counter overflow of the Simple-LIN module occurs. At this time, a RXI interrupt request is not output. The ERI interrupt request can be canceled by clearing all the flags.

When transmitting Start Frame, if CCR0.TIE = 1b and XCR0.BFOIE = 1b, a TXI interrupt request is output when Break Field transmission is completed. When Control Field 0 data is written to the TDR register, data transmission starts. Therefore, transmission using DMAC is possible.

Set CCR0.TEIE = 1b after writing the last transmit data to the TDR register and transmission starts.

During Start Frame reception (XSR0.SFSF = 1b), reception using DMAC by RXI interrupt is not possible. Check the CSR register and XSR0 register, check the reception status (see **Figure 7.3-61**), and then clear the flag. Also read the RDR register (if you do not need to check the received data value, clear the RDRF flag without reading the RDR register). When reception of Control Field 1 is completed (XSR0.CF1MF = 1b), Start Frame detection is disabled (XSR0.SFSF = 0b) and reception using DMAC is possible. Be sure to read the RDR register.

When Start Frame / Break Field detection is enabled (XCR1.SDST = 1b), if a Break Field longer than the period set in XCR2.BFLW [15: 0] is received, the BFDF flag is set and a BFD interrupt request is output. Then RSCI becomes the Start Frame reception state. Clear the BFDF flag.

When Start Frame / Break Field detection is enabled (XCR1.SDST = 1b) and the bit rate measurement function is enabled (XCR1.BMEN = 1b), an AED interrupt factor is output when an active edge is detected. Read the timer count capture value (XSR1.TCNT [15: 0]).

7.3.18 Usage Notes

7.3.18.1 Notes on Module Standby Mode

(1) Transmission

Before using the power consumption reduction function to reduce SCI's power consumption, please do the following to confirming transmission end (CSR.TEND = 1b):

- Set the output pin state after transmission operation is stopped by CCR1.SPB2DT, SPB2IO.
- Stop the transmission (CCR0.TIE = 0b, TE = 0b, TEIE = 0b).

When transitions to these states are made during transmission, the data being transmitted become indeterminate.

To transmit data in the same transmission mode after cancellation of the low power consumption state, set the TE bit to 1b, read CSR, and write data to TDR sequentially to start data transmission. To transmit data with a different transmission mode, initialize the SCI first.

To start transmission using the DMAC after cancellation of the low power consumption state, set the TE and TIE bit to 1b. The TXI interrupt flag is set to 1b and transmission starts using the DMAC.

Figure 7.3-106 shows a sample flowchart for transition to the low power consumption state during transmission. **Figure 7.3-107** and **Figure 7.3-108** show the port pin states during transition to the low power consumption state.

(2) Reception

(a) When Address match function is non used as condition of resumption (wake-up)

Before specifying the module stop state or making a transition to the low power consumption state, stop the receive operations (CCR0.RE = 0b). If transition is made during data reception, the data being received will be invalid.

Figure 7.3-109 shows a sample flowchart for reception to software standby mode during reception.

(b) When Address match function is used as condition of resumption (wake-up)

When using the power consumption reduction function to reduce SCI's power consumption, please do the following:

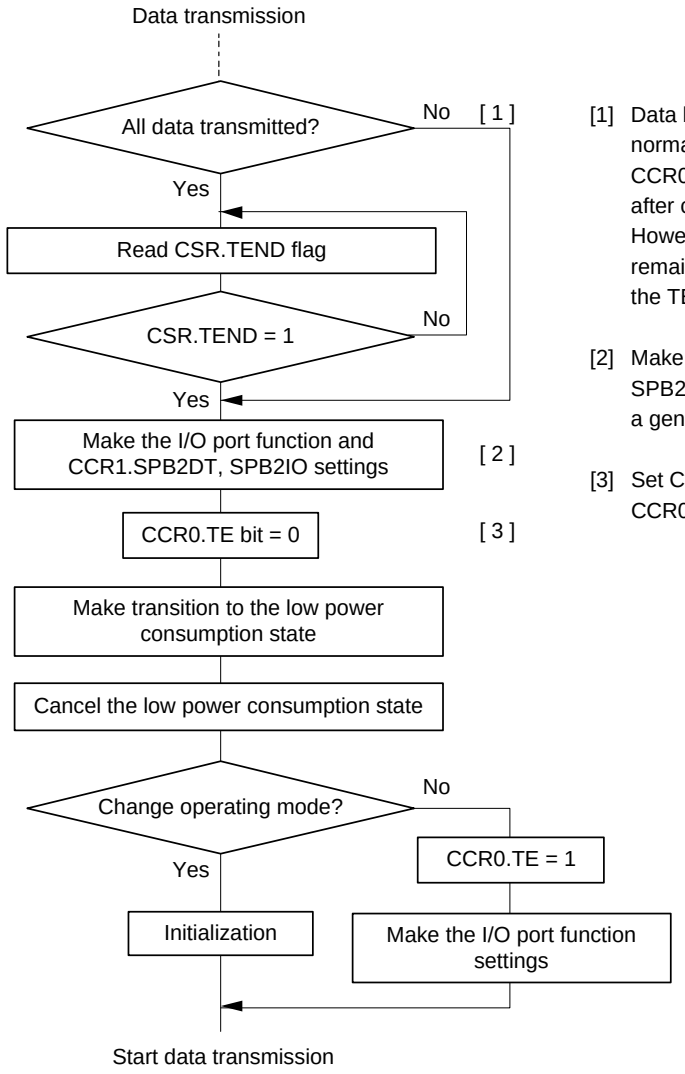
- Set the operation mode of after released.
- Set the compare data to CCR4.CMPD and set 1b to CCR0.DCME.
- Set the receive operation (CCR0.RE = 1b).

Please set CCR3.RXDESEL= 0b. Because there is a possibility that a start bit (falling edge of RXDn pin) cannot be detected at the time of low power consumption mode release.

After detecting the falling edge of the RxD terminal and transitioning from standby mode to snooze mode, the operating clock is supplied to SCI and data is received by RSCI. The address match function determines the received data, and if it matches, it transitions from snooze mode to normal mode, and if it does not match, it transitions to standby mode again. This operation behaves as if the standby mode is continued until the communication data matching the CMPD is received, and after the standby mode is released, the normal reception operation is continued.

When using this function, the reception speed must be slow enough because it is necessary to perform the transition from the falling-edge detection of the RxD terminal to the snooze mode, and from the clock supply to the RSCI to the reception.

Figure 7.3-110 shows a sample flowchart for reception to the low power consumption state during reception.



[1] Data being transmitted is lost halfway. Data can be normally transmitted from the CPU by setting the CCR0.TE to 1, reading CSR, and writing data to TDR after canceling the low power consumption state. However, if the DMAC has been activated, the data remaining in the DMAC will be transmitted when both the TE and TIE bits in CCR0 are set to 1.

[2] Make the I/O port function and CCR1.SPB2DT, SPB2IO settings to switch the TXDn pin to operate as a general I/O port.

[3] Set CCR0.TE bit to 0. Set also 0 to CCR0.TIE and CCR0.TEIE if these bits were 1.

Figure 7.3-106 Example of Flowchart for Transition to Module Standby Mode during Transmission

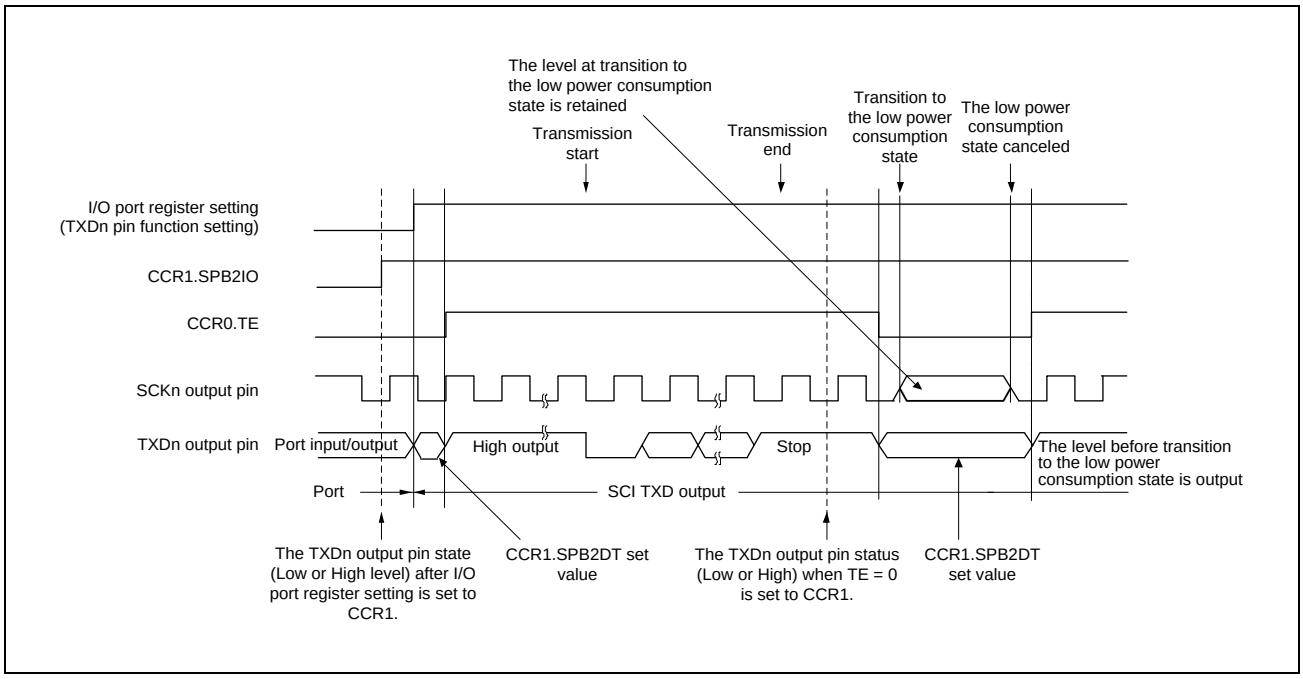


Figure 7.3-107 Port Pin States during Transition to Module Standby Mode (Internal Clock, Asynchronous Transmission)

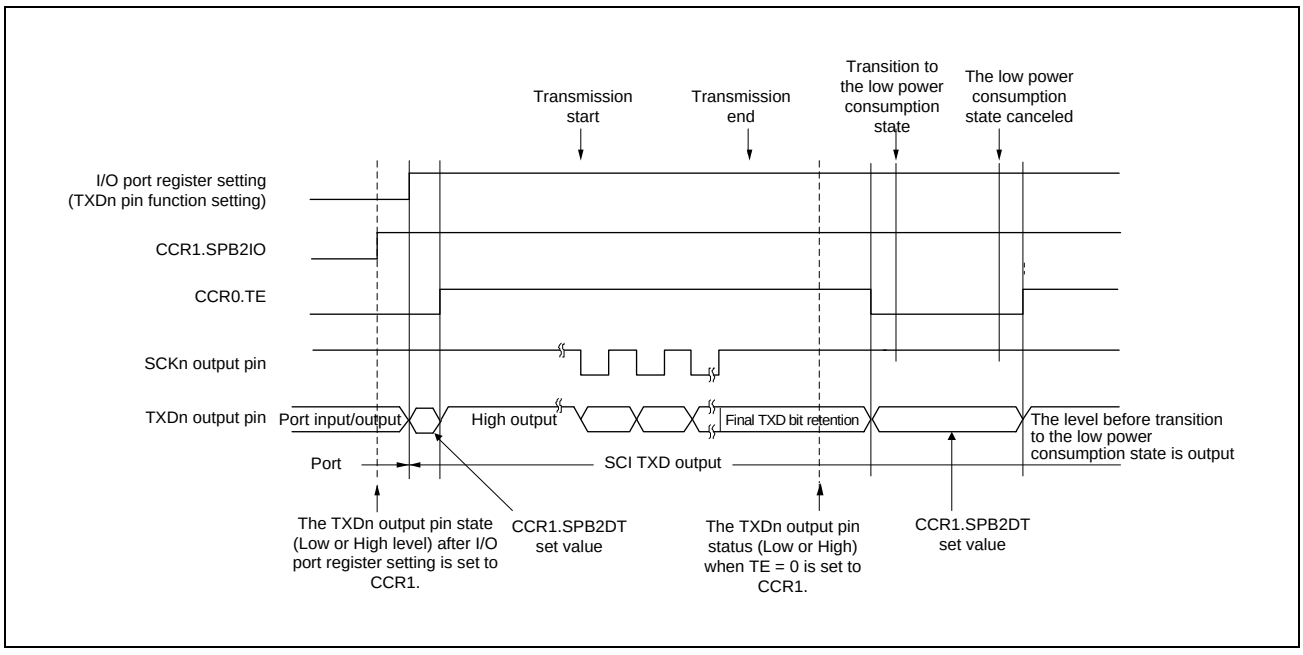


Figure 7.3-108 Port Pin States during Transition to Module Standby Mode (Internal Clock, Clock Synchronous Transmission)

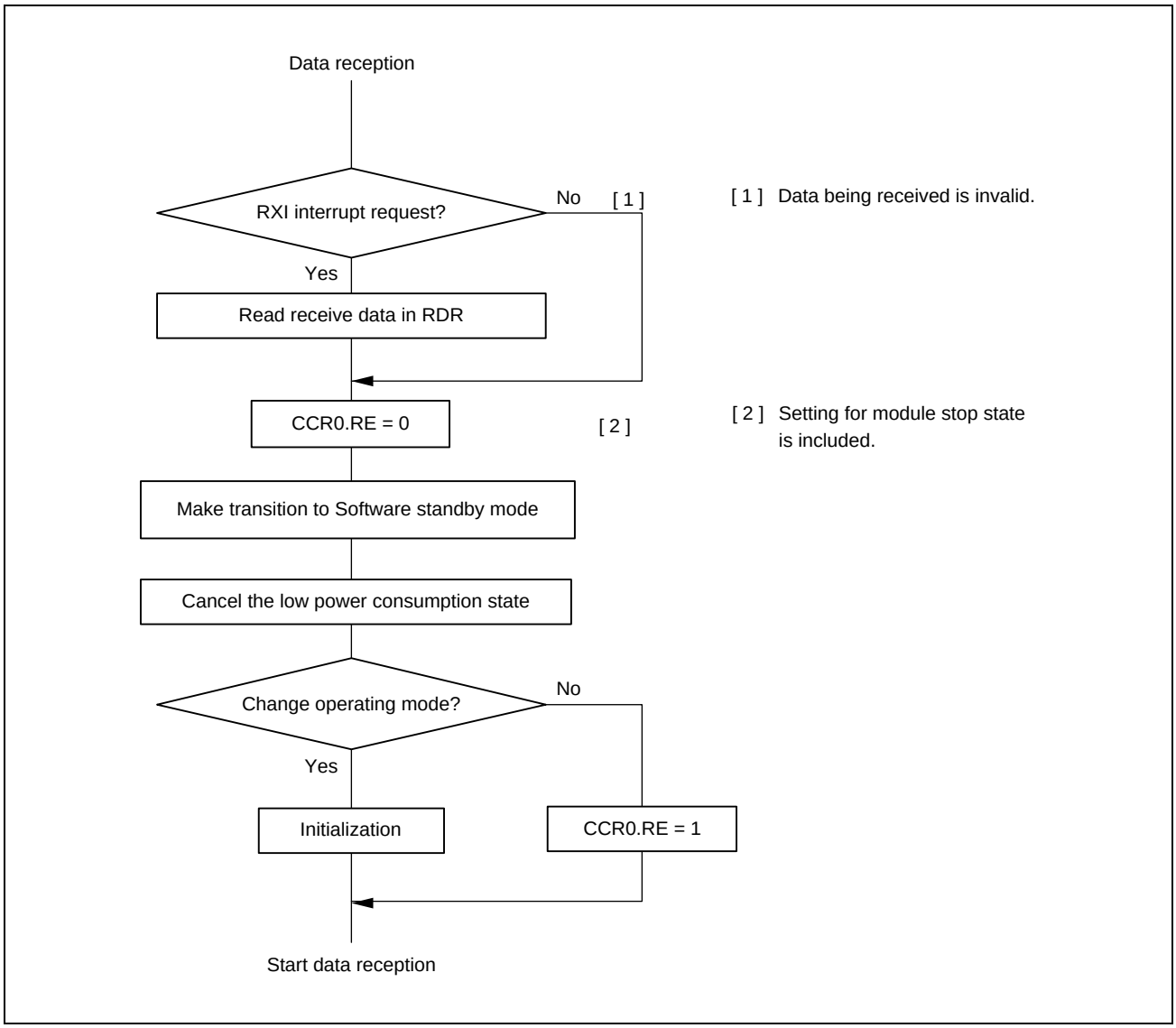


Figure 7.3-109 Example of Flowchart for Reception to Module Standby Mode during Reception

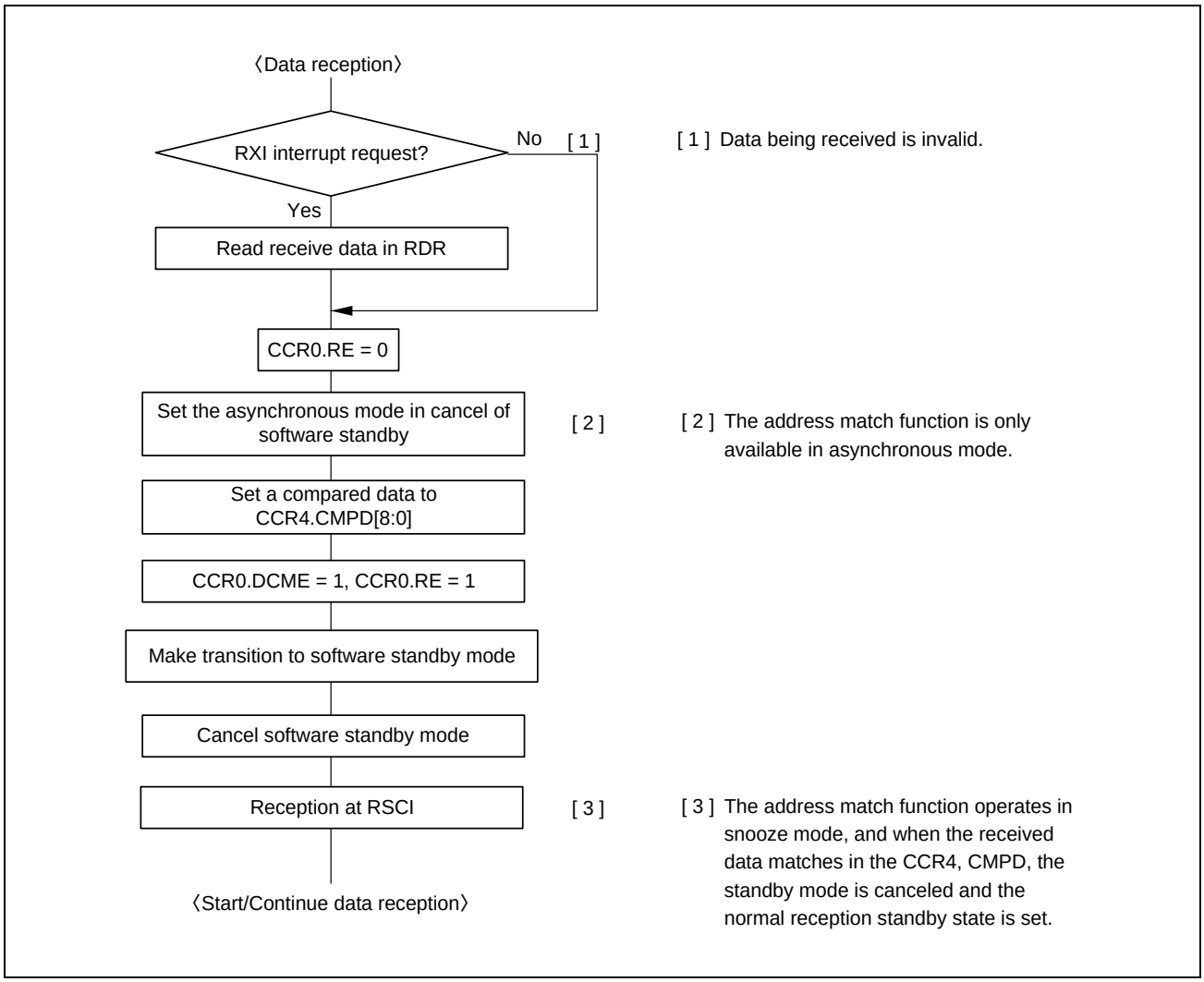


Figure 7.3-110 Example of Flowchart for Reception to Module Standby Mode during Reception With Address Match

7.3.18.2 Break Detection and Processing

(1) Non-FIFO selected

When a framing error is detected, a break can be detected by reading CSR.RXDMON bit value. In a break, the input from the RXDn pin becomes all 0s, and so the CSR.FER flag is set to 1b (framing error has occurred), and the CSR.PER flag may also be set to 1b (parity error has occurred). When the CCR3.RXDESEL bit is 0b, the SCI continues the receive operation even after a break is received. Therefore, note that even if the FER flag is set to 0b (no framing error occurred), it will be set to 1b again. When the CCR3.RXDESEL bit is 1b, the SCI sets the CSR.FER flag to 1b and stops receiving operation until a start bit of the next data frame is detected. If the CSR.FER flag is set to 0b at this time, the CSR.FER flag retains 0b during the break. When the RXDn pin is set to 1b and the break ends, detecting the beginning of the start bit at the first falling edge of the RXDn pin allows the SCI to start the receiving operation.

(2) FIFO selected

After a framing error is detected, when SCI detects that continuous receive data is 0b for 1 frame, the data stored into the receive FIFO (RDR) and reception stops. When a framing error is detected, a break can be detected by reading CSR.RXDMON bit value. After the RxD signal is in the mark state and it has finished the break, a stock of reception data to the receive FIFO (RDR) resumes.

7.3.18.3 Mark State and Production of Breaks

When CCR0.TE = 0b (serial transmission is disabled), the state of the TXDn pin can be set by CCR1.SPB2IO bit and CCR1.SPB2DT bit. Using this, it's possible to do a TXDn pin in the mark state and send a breakout. When you want to make a communication-line in the mark state (the state of 1) until the CCR0.TE bit is set to 1b (serial transmission is enabled), execute the following operations. First, set output level High by the CCR1.SPB2IO and CCR1.SPB2DT bits. Next, it's changed to a TXDn pin by I/O port function. On the other hand, if you want to send a break when sending data, set the CCR0.TE bit to 0b after setting the SPB2IO and SPB2DT bits in the CCR1 register to low level output. Setting the TE bit to 0b initializes the transmitter regardless of the current transmission status.

7.3.18.4 Receive Error Flags and Transmit Operations (Clock Synchronous Mode and Simple SPI Mode)

Transmission can be started by writing transmit-data to TDR even if CSR.ORER is 1b. However, reception cannot be started. Note also that the receive error flags cannot be set to 0 even if the CCR0.RE is set to 0b (serial reception is disabled).

7.3.18.5 Writing Data to TDR

(1) Non-FIFO selected

Data can be written to TDR anytime when CCR0.TE = 1b. However, if new data is written to TDR when transmit data is remaining in TDR, the previous data in TDR is lost because it has not been transferred to TSR yet. If you use DMAC, be sure to write transmit data to TDR in the TXI interrupt request handling routine.

(2) FIFO selected

Data can be written to transmit FIFO (TDR) when TE is 1b. Check the number of writable data with the FTSR.T[5:0] bits.

7.3.18.6 Restrictions on Clock Synchronous Transmission (Clock Synchronous Mode and Simple SPI Mode)

When the external clock source is used as a synchronization clock, the following restrictions apply.

(1) Start of transmission

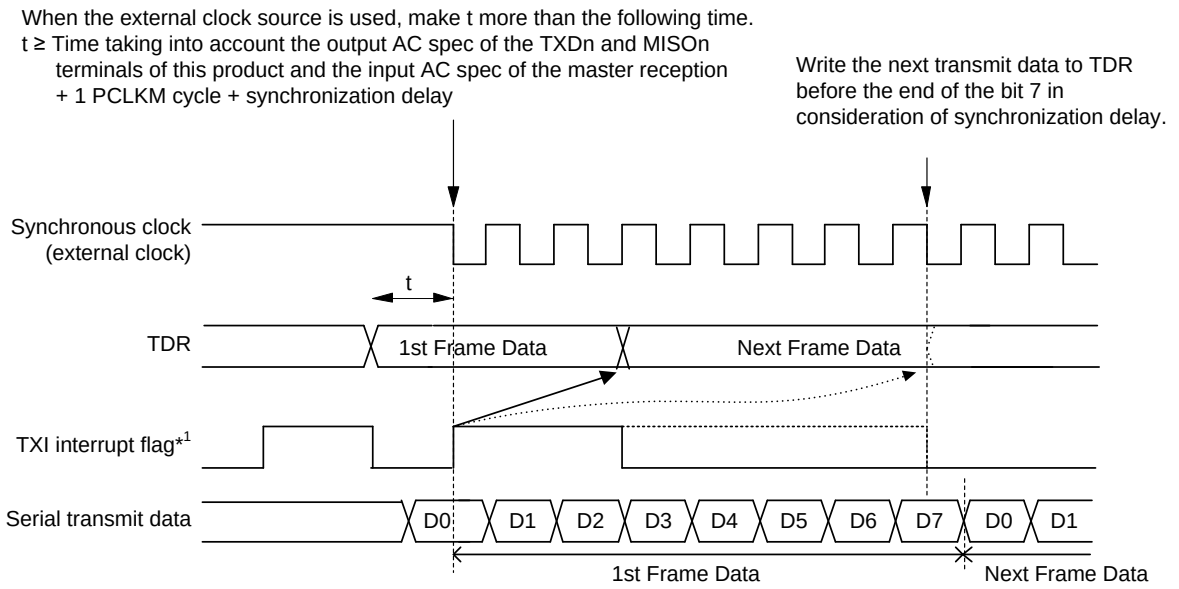
Update TDR by the CPU or DMAC and wait at least the following time until the start of the external clock input: (See **Figure 7.3-111**.)

Time taking into account the output AC specification of the TXDn and MISOn pins of this product and the input AC specification of the master reception + 1 RSCI_m_PCLK cycle + synchronization delay (2 to 3 RSCI_m_TCLK).

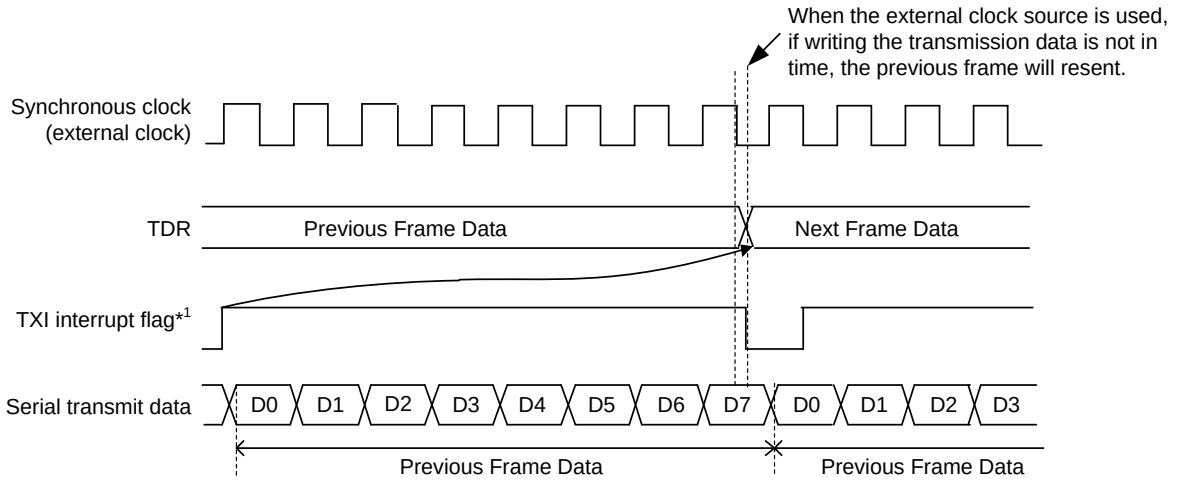
(2) Continuous transmission

Write the next transmit data to TDR before the falling edge*[1] of the transmit clock for bit 7. Please write the transmit data to TDR in consideration of synchronization delay [2 to 3 RSCI_m_TCLK]. If the transmit data cannot be written in time, the previous frame data is resent. (See **Figure 7.3-111**.)

Note 1. When CCR3.CPOL = 1b and CCR3.CPHA = 0b, or CCR3.CPOL = 0b and CCR3.CPHA = 1b. In the case of CCR3.CPOL = 0b and CCR3.CPHA = 0b, or CCR3.CPOL = 1b and CCR3.CPHA = 1b, it is the rising edge.



(1) Start of transmission and (2) Continuous transmission (When transmission data is in time)



(2) Continuous transmission (When transmission data is not in time)

Note 1. See **4.6 Interrupt Controller** for details on the corresponding interrupt vector number.

Figure 7.3-111 Restrictions on Use of External Clock in Clock Synchronous Transmission

7.3.18.7 Notes on Starting Transfer

At the point where transfer starts, when the interrupt status flag in the interrupt controller is 1, follow the procedure below to clear interrupt requests before permitting operations (by setting the CCR0.TE or CCR0.RE bit to 1b).

For details on the interrupt status flag, see section XX, Interrupt Controller (ICU).

- Confirm that transfer has stopped (the setting of the CCR0.TE or CCR0.RE bits is 0b).
- Set the corresponding interrupt enable bit (CCR0.TIE, CCR0.TEIE, or CCR0.RIE) to 0b.
- Read the corresponding interrupt enable bit (CCR0.TIE, CCR0.TEIE, or CCR0.RIE bit) to check that it has actually become 0b.
- Set the interrupt status flag in the interrupt controller to 0b.

7.3.18.8 Limitations on Simple SPI Mode

(1) Master Mode

- Use a resistor to pull up or pull down the clock line matching the initial settings for the transfer clock set by the CCR3.CPHA and CPOL bits when CCR0.SSE = 1b. This prevents the clock line from being placed in the high-impedance state when the CCR0.TE bit is set to 0b or unexpected edges from being generated on the clock line when the CCR0.TE bit is changed from 0b to 1b.
  In a single-master mode, pull up or pull down the clock line is not necessary because the clock line does not become high-impedance state when CCR0.SSE = 0b and CCR0.TE = 0b.
- In the case of the setting for clock delay (CCR3.CPHA = 0b), the receive data full interrupt (RXI) is generated before the final clock edge on the SCKm pin as indicated in **Figure 7.3-112**. If the TE and RE bits become 0b at this time before the final edge of the clock signal on the SCKm pin, the SCKm pin is placed in the high-impedance state, so the width of the last clock pulse of the transfer clock is shortened. Furthermore, an RXI interrupt may lead to the input signal on the SSn# pin of a connected slave going to the high level before the final edge of the clock signal on the SCKm pin, leading to incorrect operation of the slave.
- In a multi-master mode, take care because the SCKm pin output becomes high-impedance while the input on the SSn# pin is at the low level if a mode fault error occurs as the current character is being transferred. And stopping supply of the clock signal to the connected slave. Remake the settings for the connected slave to avoid misaligned bits when transfer is restarted.

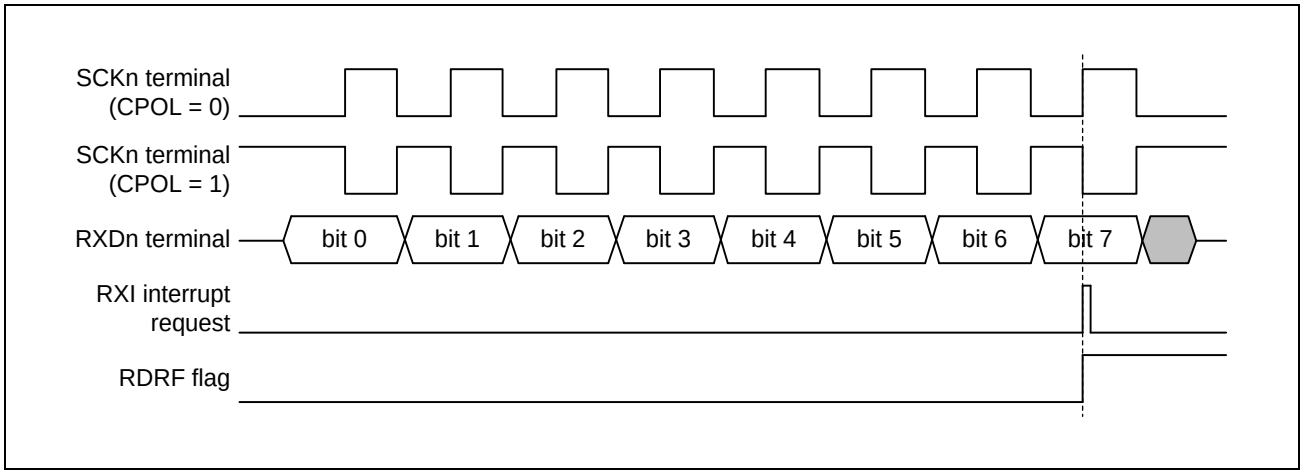


Figure 7.3-112 Timing of the RXI Interrupt in Simple SPI Mode (with Clock Delay)

(2) Slave Mode

- It takes “1 RSCI_m_PCLK + synchronization delay time + data output delay time (AC spec)” from writing the transmit data to the TDR register until the data is output to the MSIOn pin. Take these into account when starting external clock input.
- Provide an external clock signal to the master the same as the data length for transfer.
- Secure the SS input setup time (AC spec) from the SS# low-level input to the start of external clock input.
- Control the input on the SSn# pin before the start and after the end of data transfer.
- When the level being input on the SSn# pin is to be changed from low to high while the current character is being transferred, the transmission and reception are stopped immediately. Set the TE and RE bits in the CCR0 to 0b and, after remaking the settings, restart transfer of the first byte.

7.3.18.9 Notes on Transmit Enable Bit (TE)

In initial register value, when CCR0.TE = 0b, and the terminal function is TXDn, the pin outputs high impedance. So please make sure that the TXDn line will not be high impedance by the following one of ways.

1. The pull-up resistance is connected to the TXDn line.
2. Before CCR0.TE bit is 0b, the function of the terminal is changed to general-purpose input port or output port. And after CCR0.TE bit is 1b, the function of the pin is changed to TXDn.
3. In asynchronous mode, you can set CCR1 and decided level of TXDn pin during TE is 0b.

In the simple SPI mode slave operation, the MSIOn pin operates in the same way as the above TXDn pin, so please deal with 1 or 2 in the same way. (3 cannot be used.)

7.3.18.10 Notes on Simple-LIN mode

In Simple-LIN mode (CCR3.MOD[2:0] = 110b), the following functions cannot be used.

- Multi-processor communication function
- Bit Rate Modulation function
- Loopback function
- FIFO buffer

7.3.18.11 Notes on RS-485 Driver Control Function

- RS-485 Driver control function is valid only in asynchronous mode.
- When RS-485 Driver control function is active (CCR3.DEN = 1b), the TEND set timing/TEI output timing changes as follows. Wait for the TEI interrupt and set the TE bit in CCR0 to 0b.
  When RS-485 Driver control function is inactive: When STOP bit output is completed. When RS-485 Driver control function is active: At the end of DE negation time.

7.3.18.12 Notes on Loopback Function

The Loopback function is valid in asynchronous mode with internal clock and clock synchronous mode with internal clock.

7.3.18.13 Notes Regarding Register Access when Operation Clock (RSCI_m_TCLK) is Slower than Bus Clock (RSCI_m_PCLK)

If the operating clock (RSCI_m_TCLK) is slower than the bus clock (RSCI_m_PCLK), the interval for writing access to the same register should be at least RSCI_m_TCLK × 3 cycles + RSCI_m_PCLK × 3 cycles. Synchronization delay time and signal processing time are required to ensure that the setting value of the first write access is propagated to the subsequent logic circuit and the circuit is controlled correctly.

7.3.18.14 Notes on Interrupting Operation

If 0b is written to CCR0.RE during data reception and the reception operation is interrupted, there is a possibility of an invalid state, so please do not use the received data (RDR register stored value) and the flag value of each status register. To interrupt the reception operation, stop the interrupt or event link reception side and then write 0b to the CCR0.RE bit.

In addition, when 0b is written to each operation enable bit (CCR0.RE, CCR0.TE) and the operation is interrupted, the following time is required to safely stop the internal circuit. During this time, access to this register is prohibited.

RSCI_m_TCLK × 3 cycles + RSCI_m_PCLK × 3 cycles

7.3.18.15 Notes on Communication Mode Transition

When changing the communication mode setting bits CCR3.MOD[2:0] and switching the communication mode, set CCR0.RE = 0b and CCR0.TE = 0b to stop the communication operation. In order to transition the internal circuit state safely, the register access interval should be at least RSCI_m_TCLK × 3 cycles + RSCI_m_PCLK × 3 cycles. During this time, write access to this register is prohibited. As an example, the following describes the transition from Asynchronous mode to Clock- synchronous mode.

1. Confirm stop of reception and transmission at asynchronous mode (CCR3.MOD[2:0] = 000b).
2. Write 0b to CCR0.TE and CCR0.RE, stop communication.
3. Wait “RSCI_m_TCLK × 3 cycles + RSCI_m_PCLK × 3 cycles”. During this time, access to this register is prohibited.
4. Write 010b to CCR3.MOD[2:0].
5. Wait “RSCI_m_TCLK × 3 cycles + RSCI_m_PCLK × 3 cycles”. During this time, access to this register is prohibited.

SECTION 7 LOW-SPEED INTERFACE

7.4 Serial Communications Interface with FIFO (SCIF)

This LSI has one channel of serial communication interface (SCIF) with FIFO that supports asynchronous serial communication. The SCIF has 16-stage FIFO buffers for transmission and reception that enable this LSI to perform efficient high-speed continuous communication.

7.4.1 Overview

Table 7.4-1 lists the specifications of the SCIF.

Table 7.4-1 Specifications of SCIF

Item		Description
Channel		1 channel
Serial communication method		Asynchronous communication
Transfer speed		Selectable bit rate with an on-chip baud rate generator
Full duplex communication		Transmitting section: realizes continuous data transmission using 16-stage FIFO buffer Receiving section: realizes continuous data reception using 16-stage FIFO buffer
Data transmission		Selectable either LSB-first or MSB-first transfer
Interrupt source		The following six sources: • Transmit-end (TEIF) • Transmit-FIFO-data-empty (TXIF) • Receive-FIFO-data-full (RXIF) • Receive-data-ready (DRIF) • Receive-error (ERIF) • Break detection or overrun (BRIF)
Asynchronous communication	Character length	7 or 8 bits
	Transmission stop bit length	1 or 2 bits
	Parity	Even, odd, or none
	Receive error detection	Parity, overrun, and framing errors
	Break detection	Break signal detection function by hardware.
	Noise cancellation	Incorporates a digital noise filter in the RXD pin input path.
Bit rate modulation		Enables errors to be decreased by correcting the output of the on-chip baud rate generator.

Figure 7.4-1 shows a block diagram of the SCIF.

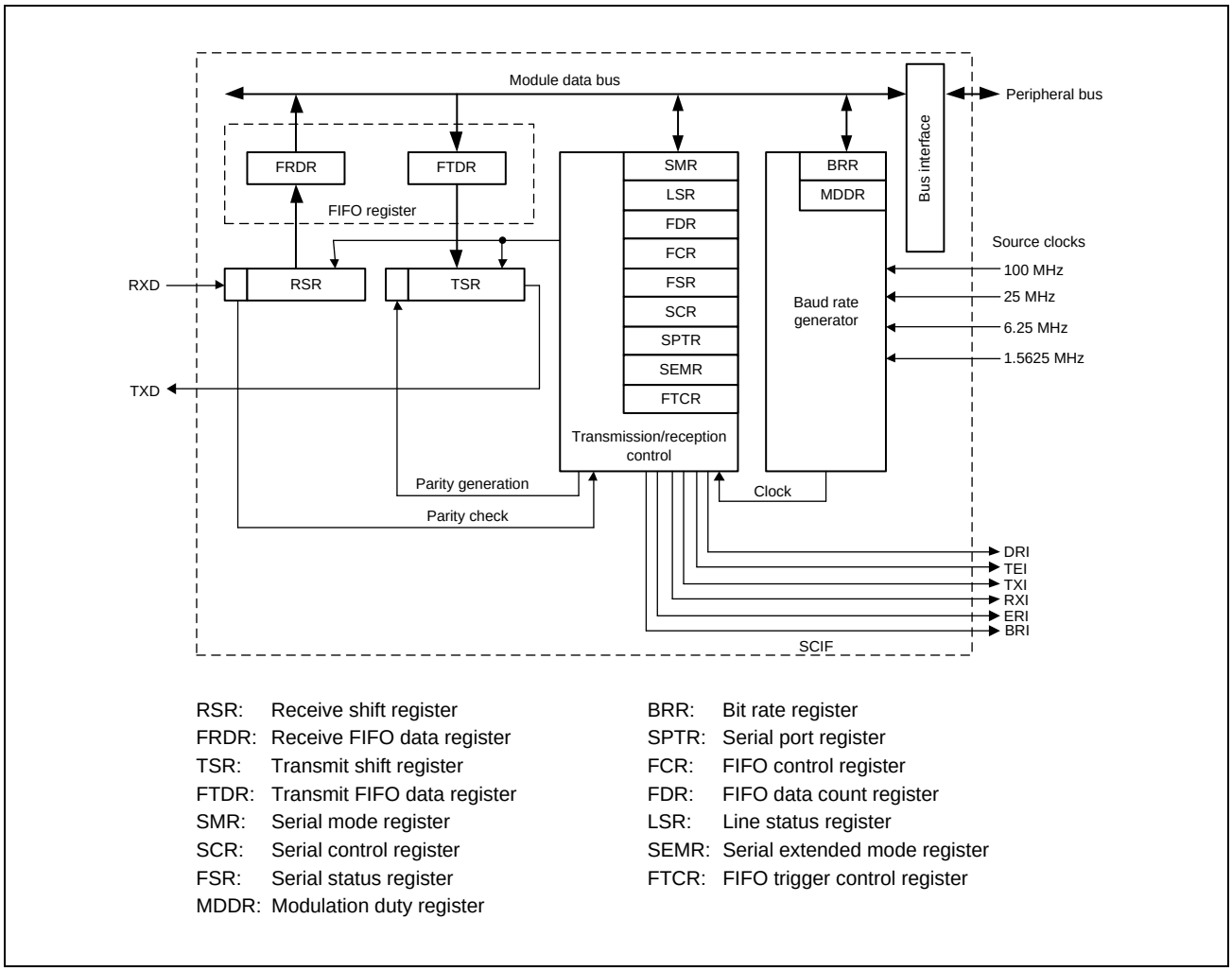


Figure 7.4-1 Block Diagram of SCIF

Table 7.4-2 lists the input/output pins of the SCIF.

Table 7.4-2 Pin Configuration of the SCIF

Channel	Item	Pin Name	Input/ Output	Function
SCIFn (n = 0)	Receive data pin	SCIF_RXD	Input	Receive data input
	Transmit data pin	SCIF_TXD	Output	Transmit data output

Note: In the notation of the pins, the channel is omitted, and pin names are represented as RXD and TXD.

7.4.2 Registers

The base address for the SCIF is as follows.

Table 7.4-3 Register Base Address

Base Register Name	Unit Name	Base Address
<SCIF0_base>	SCIF0	0_11C0_1400h (0_51C0_1400h*1, 0_41C0_1400h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

7.4.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Receive Shift Register	SCIFm_RSR	xxh	-	-
Transmit Shift Register	SCIFm_TSR	xxh	-	-
Serial Mode Register	SCIFm_SMR	0000h	0000h	16
Bit Rate Register	SCIFm_BRR	FFh	0002h	8
Modulation Duty Register	SCIFm_MDDR	FFh	0002h	8
Serial Control Register	SCIFm_SCR	0000h	0004h	16
Transmit FIFO Data Register	SCIFm_FTDR	FFh	0006h	8
Reserve	-	-	0007h	-
Serial Status Register	SCIFm_FSR	0020h	0008h	16
Receive FIFO Data Register	SCIFm_FRDR	00h	000Ah	8
Reserve	-	-	000Bh	-
FIFO Control Register	SCIFm_FCR	0000h	000Ch	16
FIFO Data Count Register	SCIFm_FDR	0000h	000Eh	16
Serial Port Register	SCIFm_SPTR	000xh	0010h	16
Line Status Register	SCIFm_LSR	0000h	0012h	16
Serial Extended Mode Register	SCIFm_SEMR	00h	0014h	8
Reserve	-	-	0015h	-
FIFO Trigger Control Register	SCIFm_FTCR	1F1Fh	0016h	16

Note: SCIFm_BRR and SCIFm_MDDR are located at the same address. Setting the MDDRS bit of the SEMR register switches these registers.

7.4.2.2 Register Description

The prefix (SCIFm_) of the register names is omitted in this and subsequent sections.

7.4.2.2.1 Receive Shift Register (SCIFm_RSR)

The RSR register receives serial data and temporally stores the data. The SCIF stores the serial data input via the RXD pin into the RSR register and converts the data to the parallel form. When one byte of data has been received, it is automatically transferred to the receive FIFO data register (FRDR).

The CPU cannot read from or write to the RSR register directly.

Access Size : 8 bits
Address : Undefined
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	-							
Initial Value	x	x	x	x	x	x	x	x
R/W	-	-	-	-	-	-	-	-

7.4.2.2.2 Transmit Shift Register (SCIFm_TSR)

The SCIF transfers the transmit data from the transmit FIFO data register (FTDR) to the TSR register, and then transmits the data serially to the TXD pin. After transmitting one byte of data, the SCIF automatically transfers the next transmit data from the FTDR register into the TSR register and starts transmission.

The CPU cannot read from or write to the TSR register directly.

Access Size : 8 bits
Address : Undefined
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	[7:0]							
Initial Value	x	x	x	x	x	x	x	x
R/W	-	-	-	-	-	-	-	-

x = Undefined

7.4.2.2.3 Serial Mode Register (SCIFm_SMR)

The SMR register specifies the SCIF serial communication format and selects the clock source for the baud rate generator.

The CPU can always read and write to the SMR register.

Access Size : 16 bits
Address : <SCIF0_base> + 0000h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	CHR	PE	PM	STOP	-	CKS[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6	CHR	0h	RW	Character Length 0b: 8-bit data 1b: 7-bit data*2
5	PE	0h	RW	Parity Enable 0b: Parity bit addition or check is disabled. 1b: Parity bit addition or check is enabled.
4	PM	0h	RW	Parity Mode 0b: Even parity 1b: Odd parity
3	STOP	0h	RW	Stop Bit Length 0b: One stop bit 1b: Two stop bits
2	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1 to 0	CKS[1:0]	0h	RW	Clock Select 00b: 1 × P0ϕ*1 01b: 1/4 × P0ϕ*1 10b: 1/16 × P0ϕ*1 11b: 1/64 × P0ϕ*1

Note 1. P0ϕ: Peripheral clock (SCIF_0_clk_pck)

Note 2. When 7-bit data is selected, the MSB (bit 7) of the transmit FIFO data register is not transmitted.

CKS[1:0] Bits (Clock Select)

Select an internal clock source for the on-chip baud rate generator. For further information on the clock source, bit rate register settings, and baud rates, see **7.4.2.2.9 Bit Rate Register (SCIFm_BRR)**.

STOP Bit (Stop Bit Length)

Selects one bit or two bits as the stop bit length. When receiving, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1b, it is treated as a stop bit, but if the second stop bit is 0b, it is treated as the start bit of the next incoming character.

Note: When transmitting with one stop bit, a single 1 bit (stop bit) is added at the end of each transmission character.

Note: When transmitting with two stop bits, two 1 bits (stop bits) are added at the end of each transmission character.

PM Bit (Parity Mode)

Selects either the even or odd parity check. The setting of this bit is effective only when the parity enable (PE) bit of this register is set to 1b. The setting of this bit is ignored when parity addition/check is disabled.

Note: If even parity is selected, the parity bit is added to data to be transmitted to make the total number of 1s even in the transmission character and parity bit combined. When receiving, the SCIF verifies that the total number of 1s in the received character and parity bit combined is even.

Note: If odd parity is selected, the parity bit is added to data to be transmitted to make the total number of 1s odd in the transmission character and parity bit combined. When receiving, the SCIF verifies that the total number of 1s in the received character and parity bit combined is odd.

PE Bit (Parity Enable)

Selects whether to add a parity bit on data transmission and whether to enable/disable the parity check on data reception.

Note: When this bit is set to 1b, an even or odd parity bit specified in the PM bit is added to data to be transmitted. The SCIF verifies whether the parity bit of the received data is even or odd as specified in the PM bit when receiving.

CHR Bit (Character Length)

Selects 7- or 8-bit data length.

7.4.2.2.4 Bit Rate Register (SCIFm_BRR)

The BRR register is an 8-bit register that, together with the baud rate generator clock source selected by the CKS[1:0] bits in the serial mode register (SMR), determines the serial transmit/receive bit rate.

This register is located at the same address as that of the MDDR register and selected when the MDDRS bit in SEMR is 0b. The CPU can read and write to BRR. Writing to BRR should be executed when TE = RE = 0b in the SCR register.

Access Size : 8 bits
Address : <SCIF0_base> + 0002h
Initial Value : FFh

Bit	7	6	5	4	3	2	1	0
	-							
Initial Value	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW

The BRR setting is calculated using the following formulae.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0b):

$$N = \frac{100}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0b))

$$N = \frac{100}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1b))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1b):

$$N = \frac{100}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0b))

$$N = \frac{100}{16 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1b))

B: Bit rate (bit/s)
N: Setting of the BRR register ($0 \leq N \leq 255$) (The setting must satisfy the electrical characteristics).
n: Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see **Table 7.4-4**).

NOTE

The MDDR register is used to adjust the bit rate. For details, see **7.4.2.2.6 Modulation Duty Register (SCIFm**_MDDR).

Table 7.4-4 SMR Register Setting

n	Frequency of Baud Rate Generator Clock Source (MHz)	SMR Register Settings	
		CKS1	CKS0
0	100	0	0
1	25	0	1
2	6.25	1	0
3	1.5625	1	1

The bit rate error is calculated using the following formulae.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0b):

$$\text{Error}\ (\%) = \left\{\frac{100 \times 10^6}{(\mathrm{N}+1) \times \mathrm{B} \times 64 \times 2^{2\mathrm{n}-1}} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0b))

$$\text{Error}\ (\%) = \left\{\frac{100 \times 10^6}{(\mathrm{N}+1) \times \mathrm{B} \times 32 \times 2^{2\mathrm{n}-1}} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1b))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1b):

$$\text{Error}\ (\%) = \left\{\frac{100 \times 10^6}{(\mathrm{N}+1) \times \mathrm{B} \times 32 \times 2^{2\mathrm{n}-1}} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SMER.ABCS0 = 0b))

$$\text{Error}\ (\%) = \left\{\frac{100 \times 10^6}{(\mathrm{N}+1) \times \mathrm{B} \times 16 \times 2^{2\mathrm{n}-1}} - 1\right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SMER.ABCS0 = 1b))

Table 7.4-5 lists the examples of the BRR register setting.

Table 7.4-5 Bit Rates and BRR Register Settings

Bit Rate (bps)	n	N	Error (%)
150	—	—	—
300	3	162	−0.15
600	2	80	0.47
1200	2	162	−0.15
2400	2	80	0.47
4800	1	162	−0.15
9600	1	80	0.47
14400	0	216	0.01
19200	0	162	−0.15
28800	0	108	−0.45
31250	0	99	0.00
38400	0	80	0.47
115200	0	26	0.47
500000	0	5	*1

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 0b.
When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1b, the bit rate is doubled.
When bits SEMR.ABCS0 and SEMR.BGDM are both 1b, the bit rate is quadrupled.
Configure settings so the range of error is no greater than 1%.

Note 1. Values for the blank cells in the table can be set using the MDDR register.
For details, see **7.4.2.2.6 Modulation Duty Register (SCIFm**_MDDR) and the **Table 7.4-7**.

Table 7.4-6 lists the maximum bit rates for various frequencies when the baud rate generator is used.

Table 7.4-6 Maximum Bit Rates for Various Frequencies with Baud Rate Generator

Frequency of Baud Rate Generator Clock Source (MHz)	Maximum Bit Rate (bit/s)	Settings	
		n	N
100	12500000	0	0

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 1b. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1b, the bit rate is 1/2. When bits SEMR.ABCS0 and SEMR.BGDM are both 1b, the bit rate is 1/4.

7.4.2.2.5 Serial Control Register (SCIFm_SCR)

The SCR register enables or disables the SCIF transmission/reception and interrupt requests, and selects the transmit/receive clock source. The CPU can always read from and write to the SCR register.

Access Size : 16 bits
Address : <SCIF0_base> + 0004h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TIE	RIE	TE	RE	REIE	TEIE	CKE[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7	TIE	0h	RW	Transmit Interrupt Enable 0b: Transmit-FIFO-data-empty interrupt request (TXI) is disabled. 1b: Transmit-FIFO-data-empty interrupt request (TXI) is enabled.
6	RIE	0h	RW	Receive Interrupt Enable 0b: Receive-FIFO-data-full interrupt (RXI), receive-data ready interrupt (DRI), receive-error interrupt (ERI), and break interrupt (BRI) requests are disabled. 1b: Receive-FIFO-data-full interrupt (RXI), receive-data ready interrupt (DRI), receive-error interrupt (ERI), and break interrupt (BRI) requests are enabled.
5	TE	0h	RW	Transmit Enable 0b: Data transmission is disabled. 1b: Data transmission is enabled.
4	RE	0h	RW	Receive Enable 0b: Data reception is disabled. 1b: Data reception is enabled.
3	REIE	0h	RW	Receive Error Interrupt Enable 0b: Receive-error interrupt (ERI) and break interrupt (BRI) requests are disabled. 1b: Receive-error interrupt (ERI) and break interrupt (BRI) requests are enabled.
2	TEIE*1	0h	RW	Transmit End Interrupt Enable 0b: Transmit end interrupt (TEI) request is disabled. 1b: Transmit end interrupt (TEI) request is enabled.
1 to 0	CKE[1:0]	0h	RW	Clock Enable In asynchronous mode: 0xb: Internal clock is used for clock source. 1xb: Setting prohibited

Note 1. TEI interrupt requests can be cleared by reading 1b from the TEND flag, and then clearing the setting to 0b, or by setting the TEIE bit to 0b.

REIE Bit (Receive Error Interrupt Enable)

Specifies whether to enable or disable a receive-error interrupt (ERI) request and a break interrupt (BRI) request. The setting of this bit is only valid when the RIE bit is set to 0b.

Note: ERI interrupt requests can be cleared by reading 1b from the ER bit in the FSR register, and then clearing the setting to 0b, or by clearing both the RIE and REIE bits in this register to 0b. BRI interrupt requests can be cleared by reading 1b from the BRK bit in the FSR register, or from the ORER flag in the LSR register, and then clearing the setting to 0b, or by clearing both the RIE and REIE bits in this register to 0b.

RE Bit (Receive Enable)

Specifies whether to enable or disable the serial data reception.

Note: Setting this bit to 0b does not affect the receive flags (DR, ER, BRK, RDF, FER, and PER in the FSR register, and ORER in the LSR register). These flags retain their previous values.

Note: Serial reception starts when a start bit is detected. Before setting this bit to 1b, be sure to set the serial mode register (SMR) and the FIFO control register (FCR) to select the receive format and reset the receive FIFO.

TE Bit (Transmit Enable)

Specifies whether to enable or disable the serial data transmission.

Note: Serial transmission starts after writing of data to be transmitted into the FTDR register under this condition. Before setting this bit to 1b, be sure to set the serial mode register (SMR) and the FIFO control register (FCR) to select the transmit format and reset the transmit FIFO.

RIE Bit (Receive Interrupt Enable)

Specifies whether to enable or disable a receive-FIFO-data-full (RXI) interrupt request when the RDF flag in the serial status register (FSR) is set to 1b, a receive-data ready (DRI) interrupt request when the DR flag in the FSR register is set to 1b, a receive-error (ERI) interrupt request when the ER flag in the FSR register is set to 1b, and a break (BRI) interrupt request when the BRK flag in the FSR register or the ORER flag in the line status register (LSR) is set to 1b.

Note: RXI interrupt requests can be cleared by reading 1b from the DR or RDF flag in the FSR register, then clearing the flag to 0b, or by clearing the RIE bit to 0b. DRI interrupt requests can be cleared by reading 1b from the DR flag in the FSR register, and then clearing the setting to 0b, or by clearing the RIE bit in this register to 0b. Receive error interrupt (ERI) requests and break interrupt (BRI) requests can be cleared by clearing both the RIE and REIE bits in this register to 0b.

TIE Bit (Transmit Interrupt Enable)

Specifies whether to enable or disable a transmit-FIFO-data-empty interrupt (TXI) request when the serial transmit data is transferred from the transmit FIFO data register (FTDR) into the transmit shift register (TSR), the quantity of data in the transmit FIFO data register falls below the specified trigger number for transmission, and the TDFE flag in the serial status register (FSR) is set to 1b.

Note: TXI interrupt requests can be cleared either by writing a greater quantity of transmit data than the specified transmission trigger number into the FTDR register, reading 1b from the TDFE flag, and then clearing the TDFE flag to 0b, or by clearing this bit to 0b.

7.4.2.2.6 Modulation Duty Register (SCIFm_MDDR)

The MDDR register corrects the bit rate adjusted by the BRR register. The value after reset of this register is FFh. When the BRME bit in SEMR is set to 1b, the bit rate generated by the on-chip baud rate generator is evenly corrected according to the settings of MDDR (MDDR/256). The relationship between the MDDR register setting and the bit rate (B) is given by the following formula. The MDDR register is located at the same address as that of the BRR register and is selected when the MDDRS bit in SEMR is 1b. This register is only writable when TE = RE = 0b in the SCR register. b7 in this register is fixed to 1b.

Access Size : 8 bits
Address : <SCIF0_base> + 0002h
Initial Value : FFh

Bit	7	6	5	4	3	2	1	0
	-							
Initial Value	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW

The formulae below show the relationships between the MDDR setting and the bit rate (B) when the bit rate modulation function is used.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0b):

$$B = \frac{100 \times 10^6}{64 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0b))

$$B = \frac{100 \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1b))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1b):

$$B = \frac{100 \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0b))

$$B = \frac{100 \times 10^6}{64 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1b))

B: Bit rate (bits/s)
N: BRR register setting ($0 \leq N \leq 255$)
(The setting must satisfy the electrical characteristics).
MDDR: MDDR setting ($128 \leq MDDR \leq 256$)
n: Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see **Table 7.4-7**).

Table 7.4-7 Bit Rates and BRR and MDDR Registers Settings

Bit Rate (bps)	n	N	MDDR	Error (%)
150	3	252	199	0.02
300	3	161	255	0.08
600	3	80	255	0.08
1200	2	161	255	0.08
2400	2	80	255	0.08
4800	1	161	255	0.08
9600	1	80	255	0.08
14400	—	—	—	—
19200	0	161	255	0.08
28800	0	107	254	0.08
31250	—	—	—	—
38400	0	80	253	0.08
115200	0	26	255	0.08
500000	0	5	246	0.10

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 0b. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1b, the bit rate is doubled. When bits SEMR.ABCS0 and SEMR.BGDM are both 1b, the bit rate is quadrupled. Configure settings so the range of error is no greater than 1%.

7.4.2.2.7 Transmit FIFO Data Register (SCIFm_FTDR)

The FTDR register is an 8-bit, 16-stage FIFO register that stores serial transmission data. When the SCIF detects that the transmit shift register (TSR) is empty, it transmits data written in the FTDR register to the TSR register and starts serial transmission. Continuous serial transmission is executed until there is no transmit data left in the FTDR register. Writing the transmit data to the FTDR register should be done when a transmit data empty interrupt (TXI) request is generated. When the FTDR register becomes full of transmit data (16 bytes), no more data can be written. Even if new data is written, the data is ignored.

The CPU can read from the FTDR register but cannot write to it.

Access Size : 8 bits
Address : <SCIF0_base> + 0006h
Initial Value : FFh

Bit	7	6	5	4	3	2	1	0
	[7:0]							
Initial Value	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R

7.4.2.2.8 Serial Status Register (SCIFm_FSR)

The FSR register is a 16-bit register. The 8 lower-order bits indicate the status flag representing the SCIF operating state.

The CPU can always read and write to the FSR register, but cannot write 1b to the status flags (ER, TEND, TDFE, BRK, RDF, and DR bits) in this register. These flags can be only cleared to 0 when they have first been read (after being set to 1). b3 (FER) and b2 (PER) are read-only bits that cannot be written.

Access Size : 16 bits
Address : <SCIF0_base> + 0008h
Initial Value : 0020h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ER	TEND	TDFE	BRK	FER	PER	RDF	DR
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7	ER	0h	RW*1	Receive Error Flag 0b: Reception is in progress or has normally completed. 1b: A framing error or parity error has occurred during reception.
6	TEND	0h	RW*1	Transmit End Flag 0b: Transmission is in the waiting state or in progress. 1b: Transmission is completed.
5	TDFE	1h	RW*1	Transmit FIFO Data Empty Flag 0b: The quantity of transmit data written in the FTDR register exceeds the specified transmission trigger number. 1b: The quantity of transmit data written in the FTDR register is equal to or less than the specified transmission trigger number.*3
4	BRK	0h	RW*1	Break Detect Flag 0b: No break signal is received. 1b: A break signal is received.*2
3	FER	0h	R	Framing Error Flag*4 0b: No receive framing error occurred in the next data read from the FRDR register. 1b: A receive framing error occurred in the next data read from the FRDR register.
2	PER	0h	R	Parity Error Flag*4 0b: No receive parity error occurred in the next receive data read from the FRDR register. 1b: A receive parity error occurred in the next receive data read from the FRDR register.
1	RDF	0h	RW*1	Receive FIFO Data Full Flag 0b: The quantity of receive data in the FRDR register falls below the specified reception trigger number. 1b: The quantity of receive data written in the FRDR register is equal to or greater than the specified reception trigger number.
0	DR	0h	RW*1	Receive Data Ready Flag 0b: Reception is in progress, or no received data has remained in the FRDR register after normally completed receiving. 1b: Next receive data has not been received.

Note 1. To clear the flag, 0b can only be written after 1b is read.

Note 2. When a break signal is detected, transfer of the receive data (0h) to the FRDR register stops after the detection. When the break ends and the receive signal becomes mark state (high level), the transfer of receive data resumes.

Note 3. Since the FTDR register is a 16-byte FIFO register, the maximum quantity of data that can be written when TDFE is 1 is “16 minus the number of non-transmitted data units”. If additional data is written, the data is ignored. The quantity of data in the FTDR register is indicated by the 8 higher-order bits of the FDR register.

Note 4. When the DMAC is used to read data, the generation of errors cannot be checked by reading this flag.

DR Bit (Receive Data Ready Flag)

Indicates that the quantity of data stored in the receive FIFO data register (FRDR) falls below the specified reception trigger number, and that no next data has been received yet after the elapse of 15 ETUs from the last stop bit.

[Setting condition]

- DR is set to 1b when the FRDR register contains less data than the specified reception trigger number, and no next data has been received yet after the elapse of 15 ETUs*1 from the last stop bit.

[Clearing conditions]

When either of the following is satisfied:

- DR is cleared to 0b when DR = 1b is read and then 0b is written to the DR flag.
- DR is cleared to 0b when all received data in the FRDR register are read.

Note 1. This is equivalent to one and a half (1.5) frames in the 8-bit format with one stop bit (ETU: elementary time unit).

Note: When the RE bit in SCR is cleared to 0, the DR bit is not affected and retains its previous value.

RDF Bit (Receive FIFO Data Full Flag)

Indicates that receive data has been transferred to the receive FIFO data register (FRDR), and the quantity of data in FRDR becomes equal to or greater than the specified reception trigger number.

[Setting condition]

- RDF is set to 1b when the quantity of receive data which is equal to or greater than the specified reception trigger number are stored in the FRDR register*1.

[Clearing conditions]

- RDF is cleared to 0b when RDF = 1b is read and then 0b is written to this bit.
- RDF is cleared to 0b when the FRDR register is read.

Note 1. Since the FRDR register is a 16-byte FIFO register, the maximum quantity of data that can be read when this bit is 1b is equivalent to the specified reception trigger number. If an attempt is made to read after all the data in the FRDR register has been read, the read data is undefined. The quantity of receive data in the FRDR register is indicated by the 8 lower-order bits of the FDR register.

PER Bit (Parity Error Flag)

Indicates whether there is a parity error in the data read from the receive FIFO data register (FRDR).

[Setting condition]

- PER is set to 1b when a parity error is present in the next data read from the FRDR register.

[Clearing condition]

- PER is cleared to 0b when no parity error is present in the next data read from the FRDR register.

FER Bit (Framing Error Flag)

Indicates whether there is a framing error in the data read from the receive FIFO data register (FRDR).

[Setting condition]

- FER is set to 1b when a framing error is present in the next data read from the FRDR register.

[Clearing condition]

- FER is cleared to 0b when no framing error is present in the next data read from the FRDR register.

BRK Bit (Break Detect Flag)

Indicates that a break signal has been detected in receive data.

[Setting condition]

- BRK is set to 1b when data including a framing error is received, and the framing error is followed by at least one frame of data received at the space 0 level (low level).

[Clearing condition]

- BRK is cleared to 0b when software reads BRK after it has been set to 1b and then writes 0b to BRK.

TDFE Bit (Transmit FIFO Data Empty Flag)

Indicates that data has been transferred from the transmit FIFO data register (FTDR) into the transmit shift register (TSR), the quantity of data in the FRDR register becomes equal to or less than the specified transmission trigger number, and writing of transmit data to the FRDR register is enabled.

[Setting conditions]

When either of the following is satisfied:

- TDFE is set to 1b when the TE bit in SCR is 0b.
- TDFE is set to 1b when the quantity of transmit data written in the FRDR register is equal to or less than the specified transmission trigger number.

[Clearing conditions]

- TDFE is cleared to 0b when 0b is written in the TDFE bit after reading TDFE = 1b.
- When transmit data is written to the FTDR register

TEND Bit (Transmit End Flag)

Indicates that the FRDR register contains no more valid data and transmission is completed when transmitting the last bit of the transmit data.

[Setting condition]

When the following is satisfied:

- TEND is set to 1b when the FTDR register does not contain transmit data when the last bit of the serial transmission data is transmitted.

[Clearing conditions]

- When transmit data is written to the FTDR register
- When 0b is written to TEND after it has been read as 1b

ER Bit (Receive Error Flag)

Indicates the occurrence of a framing error, or of a parity error when receiving the parity-added data*[1].

[Setting conditions]

When either of the following is satisfied:

- ER is set to 1b when the stop bit is found to be 0b after checking whether the stop bit of the received data is 1b at the end of one data receive operation*[1].

- ER is set to 1b when the total number of 1s in the received data and parity bit combined does not match the even or odd parity setting specified by the PM bit in the SMR register.

[Clearing condition]

- When 0b is written to ER after it has been read as 1b

Note 1. Clearing the RE bit to 0b in the SCR register does not affect this bit, which retains its previous value. Even if a receive error occurs, the receive data is transferred to the FRDR register and the receive operation is continued. Whether the data read from the FRDR register includes a receive error can be detected by the FER and PER bits in the FSR register.

Note: In two stop bits mode, only the first stop bit is checked; the second stop bit is not checked.

7.4.2.2.9 Receive FIFO Data Register (SCIFm_FRDR)

The FRDR register is an 8-bit, 16-stage FIFO register that stores the received serial data. When the SCIF receives one byte of serial data, it transfers the received data from the receive shift register (RSR) to the FRDR register and completes the receive operation. Continuous reception is possible until the received 16 bytes of data are stored. If the FRDR register is read when there is no received data in the FRDR register, an undefined value is read.

When the FRDR register is full of received data, subsequently received serial data is lost.

Access Size : 8 bits
Address : <SCIF0_base> + 000Ah
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-							
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

7.4.2.2.10 FIFO Control Register (SCIFm_FCR)

The FCR register resets the quantity of data in the transmit FIFO data register (FTDR) and the receive FIFO data register (FRDR) and specifies the number of triggers. This register also specifies whether to enable the loop-back test.

The CPU can always read and write to the FCR register.

Access Size : 16 bits
Address : <SCIF0_base> + 000Ch
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RTRG[1:0]		TTRG[1:0]		-	TFRST	RFRST	LOOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 6	RTRG[1:0]	0h	RW	Receive FIFO Data Trigger Number Select In asynchronous mode: 00b: 1 01b: 4 10b: 8 11b: 14
5 to 4	TTRG[1:0]	0h	RW	Transmit FIFO Data Trigger Number Select 00b: 8 (8)*1 01b: 4 (12)*1 10b: 2 (14)*1 11b: 0 (16)*1
3	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	TFRST	0h	RW	Transmit FIFO Data Register Reset 0b: Normal operation 1b: Resets the FTDR register.
1	RFRST	0h	RW	Receive FIFO Data Register Reset 0b: Normal operation 1b: Resets the FRDR register.
0	LOOP	0h	RW	Loop-Back Test 0b: Loop back test is disabled. 1b: Loop back test is enabled

Note 1. Values in parentheses are the numbers of empty bytes in the FTDR register when the TDFE flag is set to 1b and a transmit FIFO data empty interrupt (TXI) request is generated.

LOOP Bit (Loop-Back Test)

Internally connects between the transmit output pin (TXD) and the receive input pin (RXD) to perform loop-back testing.

RFRST Bit (Receive FIFO Data Register Reset)

Disables the receive data in the receive FIFO data register (FRDR) and makes the data to the empty state. If you set this bit to 1b, be sure to clear it to 0b afterward.

TFRST Bit (Transmit FIFO Data Register Reset)

Disables the transmit data in the transmit FIFO data register (FTDR) and makes the data to the empty state. If you set this bit to 1b, be sure to clear it to 0b afterward.

TTRG[1:0] Bits (Transmit FIFO Data Trigger Number Select)

Specify the reference quantity of data for transmission (i.e., the threshold number of entries to trigger the writing of further data for transmission) for setting of the TDFE flag in the serial status register (FSR). When the number of entries for transmission in the transmission FIFO, i.e. the number of entries written to the transmit FIFO data register (FTDR) that are yet to be transmitted, falls to or below the specified trigger number for transmission, the TDFE flag is set to 1b and a transmit FIFO data empty interrupt (TXI) request is generated.

The setting in these bits is valid when the TTRGS bit in the FTCR register is 0b. When the TTRGS bit in the FTCR register is 1b, the setting of the TFTC[4:0] bits in the FTCR register is valid.

RTRG[1:0] Bits (Receive FIFO Data Trigger Number Select)

Specify the reference quantity of receive data (i.e., the threshold number of entries to trigger the reading of received data) for setting of the RDF flag in the serial status register (FSR). When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number for reception, the RDF flag is set to 1b and a receive FIFO data full interrupt (RXI) request is generated.

The setting in these bits is valid when the RTRGS bit in the FTCR register is 0b. When the RTRGS bit in the FTCR register is 1b, the setting of the RFTC[4:0] bits in the FTCR register is valid.

7.4.2.2.11 FIFO Data Count Register (SCIFm_FDR)

The FDR register indicates the quantity of data stored in the transmit FIFO data register (FTDR) and the receive FIFO data register (FRDR).

This register indicates the quantity of transmit data in the FTDR register with the 8 higher-order bits, and the quantity of receive data in the FRDR register with the 8 lower-order bits. The CPU can always read the FDR register.

Access Size : 16 bits
Address : <SCIF0_base> + 000Eh
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	T[4:0]					-	-	-	R[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 8	T[4:0]	0h	R	Non-Transmitted Data Quantity in FTDR Indicate the quantity of non-transmitted data stored in the FTDR register.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4 to 0	R[4:0]	0h	R	Receive Data Quantity in FRDR Indicate the quantity of receive data stored in the FRDR register.

Note: To clear the flag, 0b can only be written after 1b is read.

R[4:0] Bits

Indicate the quantity of receive data stored in the FRDR register.

00h means no received data, and 10h means that all of the received data is stored in the FRDR register.

T[4:0] Bits

Indicate the quantity of non-transmitted data stored in the FTDR register.

00h means no transmit data, and 10h means that all of the data for transmission is stored in the FTDR register.

7.4.2.2.12 Serial Port Register (SCIFm_SPTR)

The SPTR register controls input/output and data of the pins multiplexed to SCIF function. The CPU can always read and write to the SPTR register.

NOTE

b0 of this register indicates the input status of the corresponding pin. See the description of the bit for details.

Access Size : 16 bits
Address : <SCIF0_base> + 0010h
Initial Value : 000xh

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SPB2IO	SPB2DT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	SPB2IO	0h	RW	Serial Port Break Input/Output Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2DT bit.
0	SPB2DT	x	RW	Serial Port Break Data Select Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2IO bit. See **Table 7.4-8**.

x = Undefined

SPB2DT Bit (Serial Port Break Data Select)

This bit specifies the output level of the TXD pin when the setting of the SCR.TE bit is 0b. The RXD pin input status can be read from this bit regardless of the SPB2IO bit setting. However, the RXD pin function must have been selected with the general I/O port.

SPB2IO Bit (Serial Port Break Input/Output)

Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2DT bit.

Table 7.4-8 TXD Pin Status

SCR.TE Bit Setting	SPB2IO Bit Setting	SPB2DT Bit Setting	TXD Pin Status
0	0	×	Setting prohibited
0	1	0	Low output
0	1	1	High output
1	×	×	Transmit data output

Remarks: ×: Don’t care

7.4.2.2.13 Line Status Register (SCIFm_LSR)

The LSR register is a 16-bit register. The PER and FER bits indicate the number of receive errors in the receive FIFO data register. 1b cannot be written to the OREER status flag. The flag should be read as 1b prior to clearing it to 0b.

Access Size : 16 bits
Address : <SCIF0_base> + 0012h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	PER[3:0]				-	-	FER[3:0]				-	ORER
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW*1

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	PER[3:0]	0h	R	Parity Error Count Indicates the quantity of data with a parity error among the receive data stored in the receive FIFO data register (FRDR).
7 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5 to 2	FER[3:0]	0h	R	Framing Error Count Indicates the quantity of data with a framing error among the receive data stored in the receive FIFO data register (FRDR).
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	ORER	0h	RW*1	Overrun Error Flag 0b: Reception is in progress or has normally completed. 1b: An overrun error has occurred during reception.

Note 1. To clear the flag, 0 can only be written after 1b is read.

ORER Bit (Overrun Error Flag)

Indicates that receive operation abnormally stops due to occurrence of an overrun error. This flag is not affected and retains its previous state if the RE bit in the serial control register (SCR) is cleared to 0b. The receive FIFO data register (FRDR) retains the data before an overrun error occurred, and newly received data is lost. When the ORER bit is set to 1b, the SCIF cannot continue subsequent serial reception.

[Setting condition]

- When the next serial reception is completed with the receive FIFO in full state (16-byte data is received)

[Clearing condition]

- When 0b is written to ORER after being read as 1b.

Note: When the internal clock is selected while the SCIF is in clock synchronous mode, the amount of receive data can be controlled, so no overrun occurs.

FER[3:0] Bits (Framing Error Count)

The values of bits 5 to 2 indicate the quantity of data with a framing error after the ER bit in the FSR register is set. Reading 0000b from the FER[3:0] bits means all 16-byte receive data in the FRDR register have a framing error.

PER[3:0] Bits (Parity Error Count)

The values of bits 11 to 8 indicate the quantity of data with a parity error after the ER bit in the FSR register is set. Reading 0000b from the PER[3:0] bits means all 16-byte receive data in the FRDR register have a parity error.

7.4.2.2.14 Serial Extended Mode Register (SCIFm_SEMR)

The SEMR register specifies either LSB or MSB first, enables the noise cancellation, operation in normal or double-speed mode of the baud rate generator, and bit rate modulation, and selects the modulation register and the sampling count (either 8 or 16 times).

Access Size : 8 bits
Address : <SCIF0_base> + 0014h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	BGDM	-	BRME	MDDRS	DIR	NFEN	-	ABCS0
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	BGDM	0h	RW	Baud Rate Generator Double- Speed Mode Select 0b: Baud rate generator normal mode: Baud rate generator operates on the clock signal produced by dividing the clock source by two. 1b: Baud rate generator double-speed mode: Baud rate generator operates on the clock signal produced by the clock source (no frequency division).
6	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	BRME	0h	RW	Bit Rate Modulation Enable 0b: Bit rate modulation is disabled. 1b: Bit rate modulation is enabled.
4	MDDRS	0h	RW	Modulation Duty Register Select 0b: BRR register is accessible. 1b: MDDR register is accessible.
3	DIR	0h	RW	Data Transfer Direction Select 0b: Transmits the data in the FTDR register by the LSB-first method. The received data is stored in the FRDR register by the LSB-first method. 1b: Transmits the data in the FTDR register by the MSB-first method. The received data is stored in the FRDR register by the MSB-first method.
2	NFEN	0h	RW	Noise Cancellation Enable 0b: Noise cancellation for the RxD pin is disabled. 1b: Noise cancellation for the RxD pin is enabled.
1	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	ABCS0	0h	RW	Asynchronous Base Clock Select 0b: Operates on a frequency 16 times the transfer rate as the base clock. 1b: Operates on a frequency 8 times the transfer rate as the base clock.

ABCS0 Bit (Asynchronous Base Clock Select)

Selects the base clock for 1-bit period.

NFEN Bit (Noise Cancellation Enable)

Reduces noise of the input to the RxD pin. For details, see **7.4.8 Noise Cancellation**.

DIR Bit (Data Transfer Direction Select)

Selects the serial communication format. This bit is valid only when the transmit/receive data length is 8 bits.

MDDRS Bit (Modulation Duty Register Select)

Selects the register to be enabled access to it.

BRME Bit (Bit Rate Modulation Enable)

Specifies whether to enable or disable the bit rate modulation.

BGDM Bit (Baud Rate Generator Double-Speed Mode Select)

Selects operating mode of the baud rate generator. When setting 1b in this bit, the baud rate generator included in the SCIF operates in double-speed mode.

7.4.2.2.15 FIFO Trigger Control Register (SCIFm_FTCR)

The FTCR register is a 16-bit register that specifies FIFO trigger conditions. The CPU can always read from and write to the FTCR register.

Access Size : 16 bits
Address : <SCIF0_base> + 0016h
Initial Value : 1F1Fh

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RTRGS	-	-	RFTC[4:0]					TTRGS	-	-	TFTC[4:0]				
Initial Value	0	0	0	1	1	1	1	1	0	0	0	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15	RTRGS	0h	RW	Receive Trigger Select 0b: RTRG[1:0] bits in FCR are valid. 1b: RFTC[4:0] bits in FTCR are valid.
14 to 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12 to 8	RFTC[4:0]	1Fh	RW	Receive FIFO Data Trigger Number 01h: Receive data trigger number is 1. 10h: Receive data trigger number is 16. Do not set 00h and 11h to 1Fh in these bits.
7	TTRGS	0h	RW	Transmit Trigger Select 0b: TTRG[1:0] bits in FCR are valid. 1b: TFTC[4:0] bits in FTCR are valid.
6 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4 to 0	TFTC[4:0]	1Fh	RW	Transmit FIFO Data Trigger Number 00h: Transmit data trigger number is 0. 0Fh: Transmit data trigger number is 15. Do not set 10h to 1Fh in these bits.

TFTC[4:0] Bits (Transmit FIFO Data Trigger Number)

Specify the reference quantity of data for transmission (i.e., the threshold number of entries to trigger the writing of further data for transmission) for setting of the TDFE flag in the serial status register (FSR).

When the number of entries for transmission in the transmission FIFO, i.e. the number of entries written to the transmit FIFO data register (FTDR) that are yet to be transmitted, falls to or below the specified trigger number for transmission, the TDFE flag is set to 1b and a transmit FIFO data empty interrupt (TXI) request is generated.

RFTC[4:0] Bits (Receive FIFO Data Trigger Number)

Specify the reference quantity of receive data (i.e., the threshold number of entries to trigger the reading of received data) for setting of the RDF flag in the serial status register (FSR).

When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number for reception, the RDF flag is set to 1b and a receive FIFO data full interrupt (RXI) request is generated.

7.4.3 Operation

7.4.3.1 Overview

The SCIF supports asynchronous serial communication in which characters are synchronized individually.

The SCIF has a 16-stage FIFO buffer for both transmission and reception, reducing the overhead of the CPU and enabling continuous high-speed communication. Selection of a transmission/reception format is enabled with the serial mode register (SMR). **Table 7.4-9** shows the transmission format which can be selected in the serial mode register (SMR).

- Data length is selectable as either 7 or 8 bits
- Parity addition and 1- or 2-bit stop bit addition are selectable.
  (The combination of the preceding selections determines the transmission/reception format and character length).
- In reception, it is possible to detect framing errors, parity errors, receive FIFO data full, overrun errors, receive data ready, and breaks.
- The stored data quantities are indicated in the FIFO data count register (FDR), respectively for transmit and receive FIFO data.
- The SCIF operates using the clock of on-chip baud rate generator.

Table 7.4-9 SMR Register Settings and SCIF Communication Formats

SMR Register			SCIF Transmission/Reception Format		
b6	b5	b3			
CHR	PE	STOP	Data Length	Parity Bit	Stop Bit Length
0	0	0	8 bits	Not set	1 bit
		1			2 bits
	1	0		Set	1 bit
		1			2 bits
1	0	0	7 bits	Not set	1 bit
		1			2 bits
	1	0		Set	1 bit
		1			2 bits

7.4.3.2 Asynchronous Serial Communication

In the asynchronous serial communication performed by the SCIF, each transmitted or received character begins with a start bit and ends with a stop bit. Serial communication is synchronized one character at a time.

The transmitting and receiving sections of the SCIF are independent, so full duplex communication is possible. The transmitter and receiver are 16-stage FIFO buffered, so data can be written and read while transmitting and receiving are in progress, enabling continuous transmission and reception.

Figure 7.4-2 shows the general format of asynchronous serial communication.

In asynchronous serial communication, the communication line is normally held in the mark (high) state. The SCIF monitors the line and starts serial communication when the line goes to the space (low) state, considered as a start bit. One serial character consists of a start bit (low), data (LSB first when LSB-first transfer is selected), parity bit (high or low), and stop bit (high), in that order.

When receiving, the SCIF synchronizes at the falling edge of the start bit. The SCIF samples each data bit on the eighth pulse of a clock with a frequency 16 or 8 times the bit rate*[1]. Receive data is latched at the center of each bit.

Note 1. When the SEMR.ABCS0 bit = 0b, data is sampled on the eighth pulse of a clock with a frequency 16 times the bit rate.
When the SEMR.ABCS0 bit = 1b, data is sampled on the fourth pulse of a clock with a frequency 8 times the bit rate.

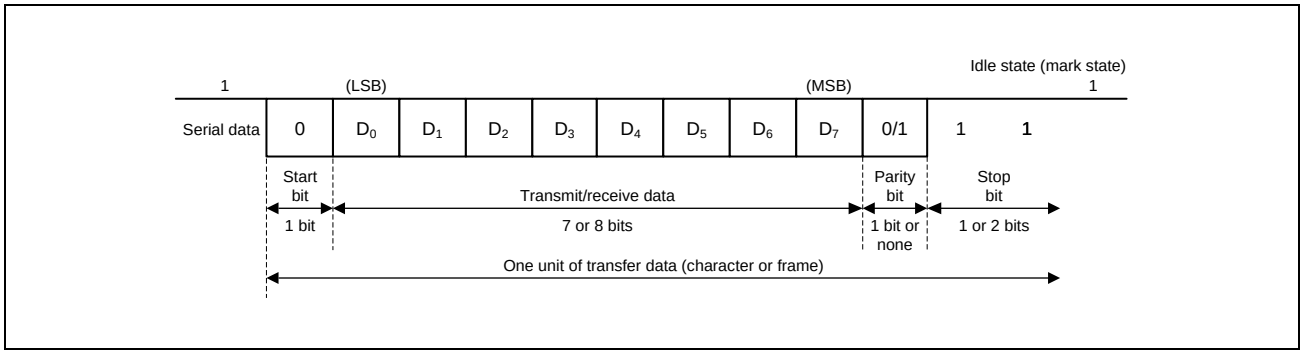


Figure 7.4-2 Data Format in Asynchronous Communication
(8-bit Data with Parity and Two Stop Bits when LSB-First Transfer is Selected)

7.4.3.3 Transmit/Receive Formats

Table 7.4-10 lists the eight communications formats that can be selected. The format is selected by setting in the serial mode register (SMR).

Table 7.4-10 Serial Communications Formats

SMR Setting			Serial Transmit/Receive Format and Frame Length											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	START	8-bit data								STOP		
		1	START	8-bit data								STOP	STOP	
	1	0	START	8-bit data								P	STOP	
		1	START	8-bit data								P	STOP	STOP
1	0	0	START	7-bit data							STOP			
		1	START	7-bit data							STOP	STOP		
	1	0	START	7-bit data							P	STOP		
		1	START	7-bit data							P	STOP	STOP	

Note: START: Start bit
STOP: Stop bit
P: Parity bit

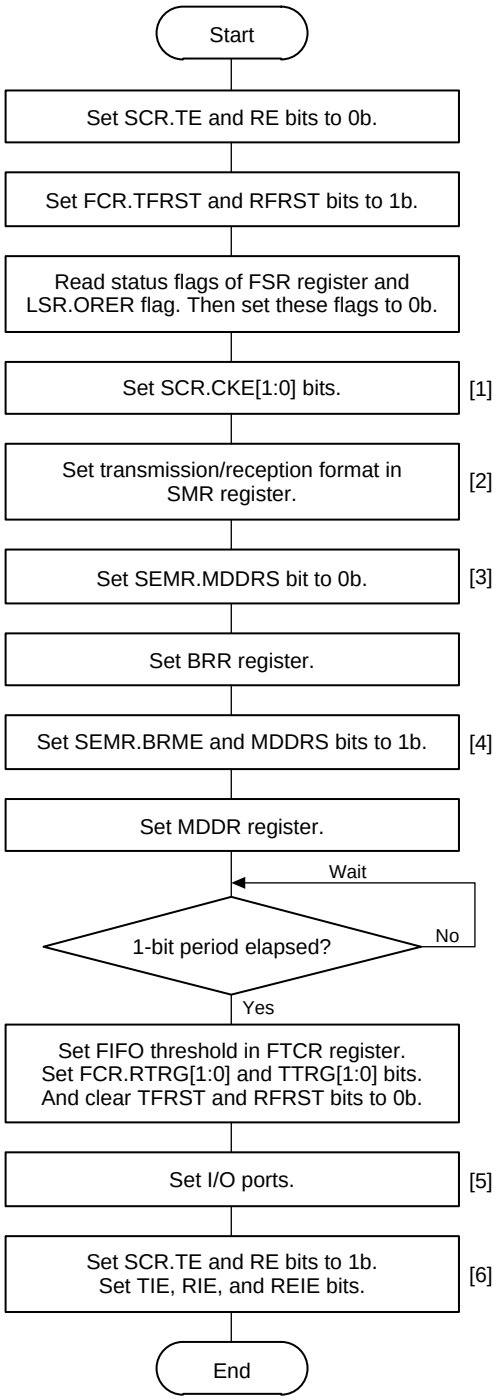
7.4.3.4 Transmitting and Receiving Data

7.4.3.4.1 SCIF initialization

Before transmitting or receiving data, clear the TE and RE bits to 0b in the serial control register (SCR), and then initialize the SCIF as follows.

When changing operating mode or communication format, always clear the TE and RE bits in the SCR register to 0b before following the procedure given below. Clearing TE to 0b initializes the transmit shift register (TSR). Clearing TE and RE to 0b, however, does not initialize the serial status register (FSR), transmit FIFO data register (FTDR), or receive FIFO data register (FRDR), which retain their previous contents. Clear TE to 0b after all transmit data has been transmitted and the TEND flag in the FSR register is set. The TE bit can be cleared to 0b during transmission, but the transmit data (the TXD pin output level) after the TE bit is cleared to 0b depends on the settings of the SPB2IO and SPB2DT bits in the SPTR register. Set the TFRST bit in the FCR register to 1b and reset the FTDR register before TE is set to 1b again to start transmission.

When an external clock is used, the clock should not be stopped during initialization or subsequent operation. SCIF operation becomes unreliable if the clock is stopped. **Figure 7.4-3** shows a sample flowchart for initializing the SCIF.



[1] Set the TIE, RIE, TE, and RE bits in the SCR register to 0b before setting the CKE[1:0] bits.

[2] Set the transmission/reception format in the SMR register.

[3] Clear the SEMR.MDDRS bit to 0b and write a value corresponding to the bit rate in the BRR register. Not necessary if an external clock is used.

[4] Set the SEMR.BRME and MDDRS bits to 1b and write a value corresponding to the bit rate in the MDDR register. Not necessary if an external clock is used. Wait for at least 1-bit period before proceeding to the next step.

[5] Set I/O ports and the MPC register so that they can be used as the SCK, TXD, and RXD pins.

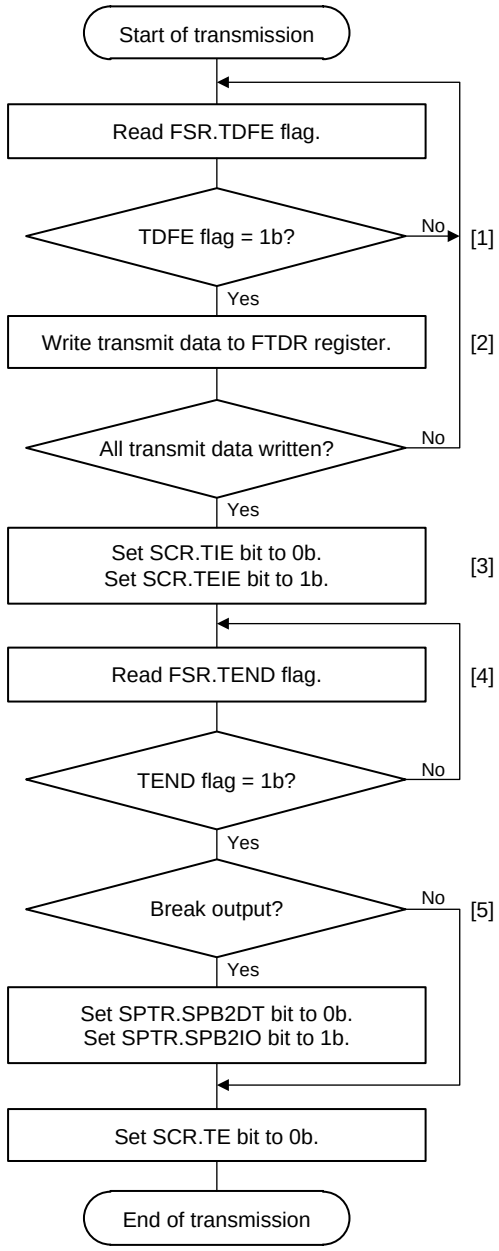
[6] After setting the MDDR register, wait for at least 1-bit period and then write 1b to the SCR.TE or RE bit. To permit interrupt requests, set TIE, RIE and REIE bits to 1b. Setting the TE and RE bits to 1b makes the TXD and RXD pins available to be used. The TXD pin enters the mark state and the RXD pin enters the idle state waiting for a start bit.

Figure 7.4-3 Sample Flowchart for SCIF Initialization

7.4.3.4.2 Transmitting serial data

Figure 7.4-4 shows a sample flowchart for serial transmission.

Follow the procedure given below for serial data transmission after enabling the SCIF for transmission.



[1] Check the SCIFA status and write the transmit data:
Read the FSR register, confirm that the TDFE flag is set to 1b, and then write the transmit data to the FTDR register. The transmit data quantity that can be written is “16 - FDR.T[4:0]”. A change of the TDFE flag from 0b to 1b can be known by a TXIF interrupt request.

[2] Serial transmission continuation procedure:
When transmit data is written to the FTDR register, the TDFE flag is automatically cleared to 0b.

[3] Serial transmission end procedure (when TXIF or TEIF interrupt is used):
After the last transmission data is written to the FTDR register, set the TIE bit to 0b and the TEIE bit to 1b in the SCR register. If the DMAC is activated to write the transmit data to the FTDR register, enable the transmit end interrupt. This will generate a CPU interrupt request when the last transmit data is written in the FTDR register. The TIE and TEIE bit settings will be changed by the CPU interrupt process.
This procedure is unnecessary if the TXIF or TEIF interrupt is not used.

[4] Serial transmission end confirmation:
Read the FSR register and confirm that the TEND flag has changed to 1b.
A change of the TEND flag from 0b to 1b can be known by a TEIF interrupt request.

[5] Break output during serial transmission:
To cause a break to be output during serial transmission, set the SPB2DT bit to 0b and the SPB2IO bit to 1b in the SPTR register. Then set the SCR.TE bit to 0b.*1

In Step [1], the number of data bytes that can be written can be checked by the FDR.T[4:0] flag (non-transmitted data quantity in the FTDR register).

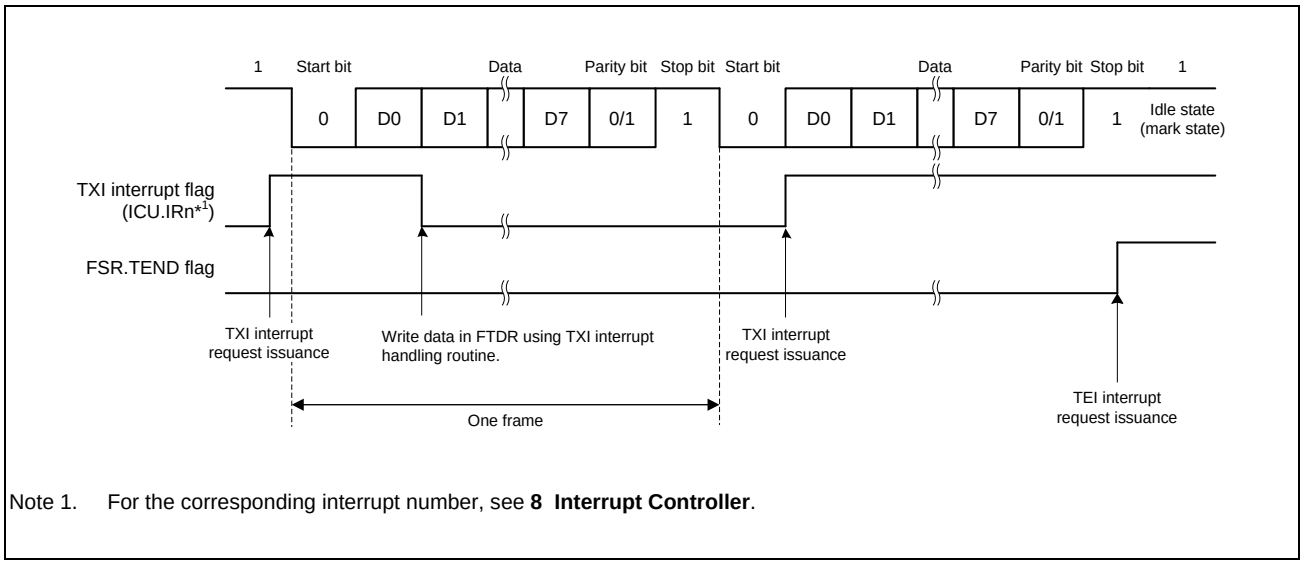
Note 1. The SPB2DT bit of the SPTR register indicates the input status of the RXD pin. Writings to the SPB2DT and SPB2IO bits of this register in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written to the SPB2DT bit. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

Figure 7.4-4 Sample Flowchart for Transmitting Serial Data

The SCIF performs serial transmission as described below.

1. When data is written into the transmit FIFO data register (FTDR) by the TXI interrupt processing routine, the SCIF transfers the data from the FTDR register to the transmit shift register (TSR) and starts transmission. Confirm that the TDFE flag in the serial status register (FSR) is set to 1b before writing transmit data to the FTDR register. The number of data bytes that can be written is “16 minus the number of non-transmitted data units”.
2. When data is transferred from the FTDR register to the TSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the FTDR register. When the number of transmit data bytes in the FTDR register becomes equal to or less than the transmission trigger number specified in the FIFO control register (FCR) or FIFO trigger control register (FTCR), the TDFE flag is set. If the TIE bit in the serial control register (SR) is set to 1b at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated. The serial transmit data is output from the TXD pin in the following order.
   a) Start bit: One-bit 0 is output.
   b) Transmit data: 8- or 7-bit data is output in LSB-first order (when LSB-first transfer is selected).
   c) Parity bit: One parity bit (even or odd parity) is output. (A format in which a parity bit is not output can also be selected.)
   d) Stop bit(s): One or two 1 bits (stop bits) are output.
   e) Mark state: 1 is output continuously until the start bit that starts the next transmission is output.
3. The SCIF checks the transmit data of the FTDR register at the timing for sending the stop bit. If data is present, the data is transferred from the FTDR register to the TSR register, the stop bit is output, and then serial transmission of the next frame is started. If there is no data to be transmitted, the TEND flag in the FSR register is set to 1b, the stop bit is output, and then the SCIF enters the mark state (high level) in which 1 is output continuously.

Figure 7.4-5 shows an example of the operation for transmission.

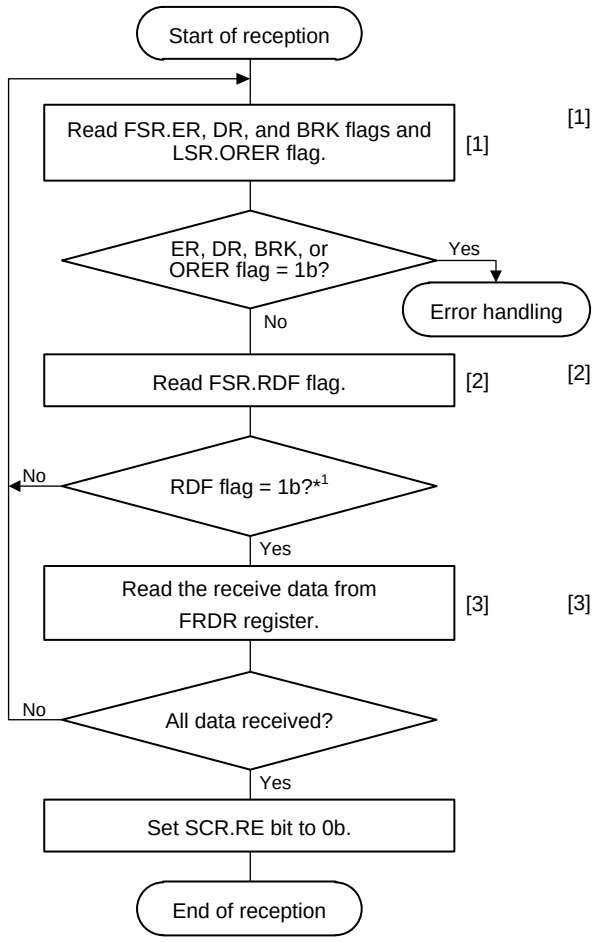


Note 1. For the corresponding interrupt number, see **8 Interrupt Controller**.

Figure 7.4-5 Example of Transmit Operation
(8-bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

7.4.3.4.3 Receiving serial data

Figure 7.4-6 and **Figure 7.4-7** show sample flowcharts for serial reception. Follow the procedure given below for serial data reception after enabling the SCIF for reception.



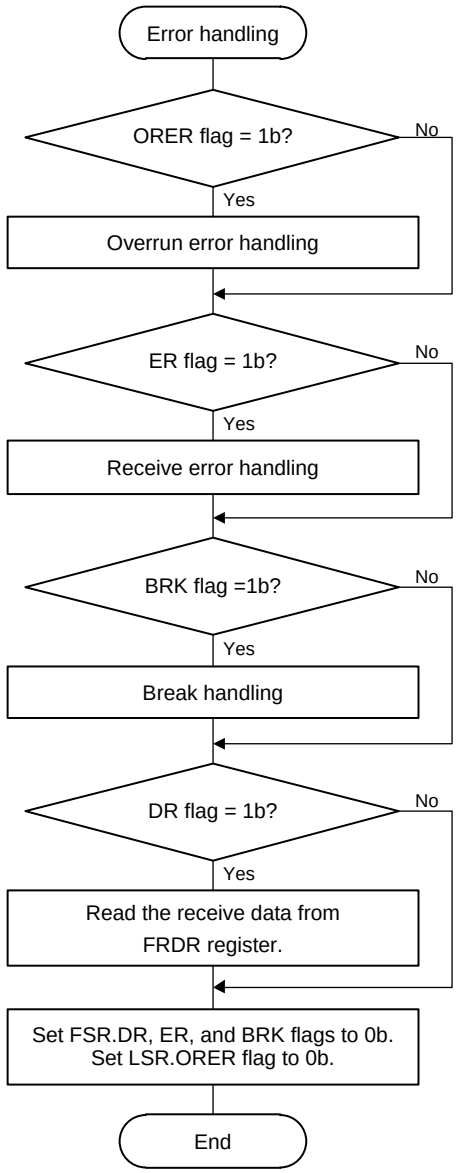
[1] Receive error handling and break detection:
When a receive error occurs, read the ER, DR, and BRK flags in the FSR register and the LSR.ORER flag to identify the error. After handling the error as specified, clear the ER, DR, BRK, and ORER flags to 0b.
In the case where a framing error occurs, a break can be detected by reading the value of the RXD pin.
Error occurrence can be known by a BRIF, ERIF, or DRIF interrupt request.

[2] Check the SCIFA status and read the receive data:
Read the FSR register, confirm that the RDF flag is set to 1b, and then read the receive data at least as much as the amount specified by the receive FIFO threshold from those stored in the FEDR register. The receive data quantity that can be read can be checked by the FDR.R[4:0] flag.
A change of the RDF flag from 0b to 1b can be known by an RXIF interrupt request.

[3] Serial reception continuation procedure:
When receive data is read from the FRDR register, the RDF flag is automatically cleared to 0b.

Note 1. The number of all the received data units should be an integral multiple of the receive FIFO threshold. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 7.4-6 Sample Flowchart for Receiving Serial Data (1)



- Whether a framing error or parity error has occurred in the received data that is read from the FRDR register can be checked with the FER and PER flags in the FSR register.
- When a break signal is received, the received data is not transferred to the FRDR register while the BRK flag is set to 1b. However, note that the last data in the FRDR register is 00h and the break data in which a framing error occurred is stored.

Figure 7.4-7 Sample Flowchart for Receiving Serial Data (2)

The SCIF performs serial reception as described below.

1. The SCIF monitors the communication line, and if a 0 start bit is detected, it performs internal synchronization to start reception.
2. The received data is stored into the RSR register in LSB-to-MSB order (when LSB-first transfer is selected).
3. The parity bit and stop bit are received.
   After receiving these bits, the SCIF carries out the following checks.
   a) Stop bit check: The SCIF checks whether the stop bit is 1. If there are two stop bits, only the first is checked.
   b) The SCIF checks whether receive data can be transferred from the receive shift register (RSR) to the receive FIFO data register (FRDR).
   c) Parity bit check: The SCIF checks whether the parity bit is an expected value.
   d) Overrun error check: The SCIF checks whether the ORER flag is 0b, indicating that the overrun error has not occurred.
   e) Break check: The SCIF checks whether the BRK flag is 0b, indicating that the break state is not set.

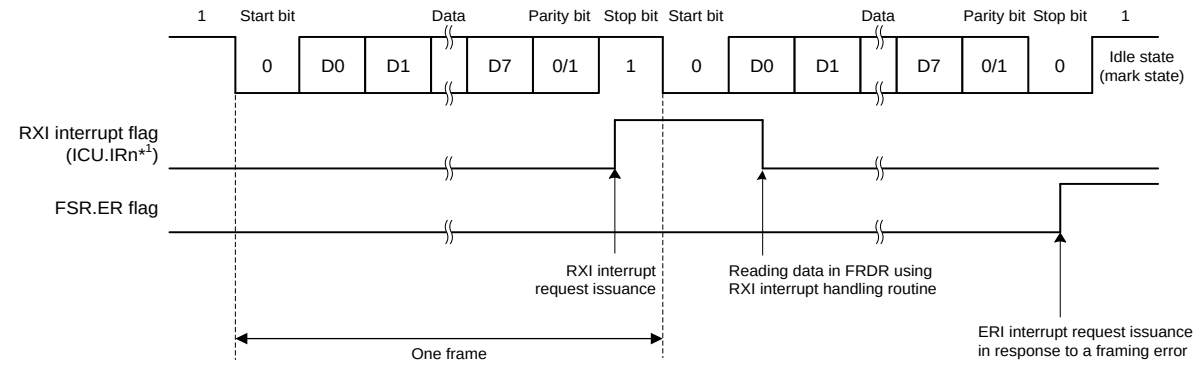
   If all the above checks are passed, the receive data is stored in the FRDR register.

   Note: When a parity error or a framing error occurs, reception is not suspended.

4. When receive data units equaling or exceeding the specified reception trigger number are stored in the receive FIFO data register (FRDR) and the RDF flag is changed to 1b, a receive FIFO data full interrupt (RXI) request is generated while the RIE bit in the SCR register is set to 1b. When the quantity of data in the FRDR register falls below the specified reception trigger number and the RIE bit in the SCR register is set to 1b, a receive data ready interrupt (DRI) request is generated if no next data is received after the elapse of 15 ETUs*[1] from the last stop bit (the DR flag in the FSR register is 1b). When the ER flag in the FSR register is changed to 1b, a receive error interrupt (ERI) request is generated while the RIE or REIE bit in the SCR register is set to 1b. When the BRK or ORER flag is changed to 1b in the FSR register, a break reception interrupt (BRI) request is generated while the RIE or REIE bit in the SCR register is set to 1b.

Note 1. It is equivalent to 1 and half frames of 8-bit format with one stop bit (ETU: Element Time Unit).

Figure 7.4-8 shows an example of the operation for reception.



Note 1. For the corresponding interrupt number, see **4.6 Interrupt control unit (ICU)**.

Figure 7.4-8 Example of SCIF Receive Operation
(8-bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

7.4.4 Bit Rate Modulation

Using the bit rate modulation, the bit rate can be corrected by skipping the specified number of clock pulses input to the baud rate generator. To correct the bit rate, only the number of clock pulses specified in the MDDR register are enabled among 256 internal clock pulses specified by the CKS1 and CKS0 bits in the SMR register in a way that forms average intervals.

Figure 7.4-9 shows an example where the baud rate generator clock source is selected by the CKS[1:0] bits in SMR and BRR and MDDR are set to 0 and 160, respectively. In this example, the cycle of the base clock is evenly corrected (256/160) and the bit rate is also corrected (160/256). Note that skipping an internal clock causes bias and expansion or contraction is generated in the pulse width of the base clock.

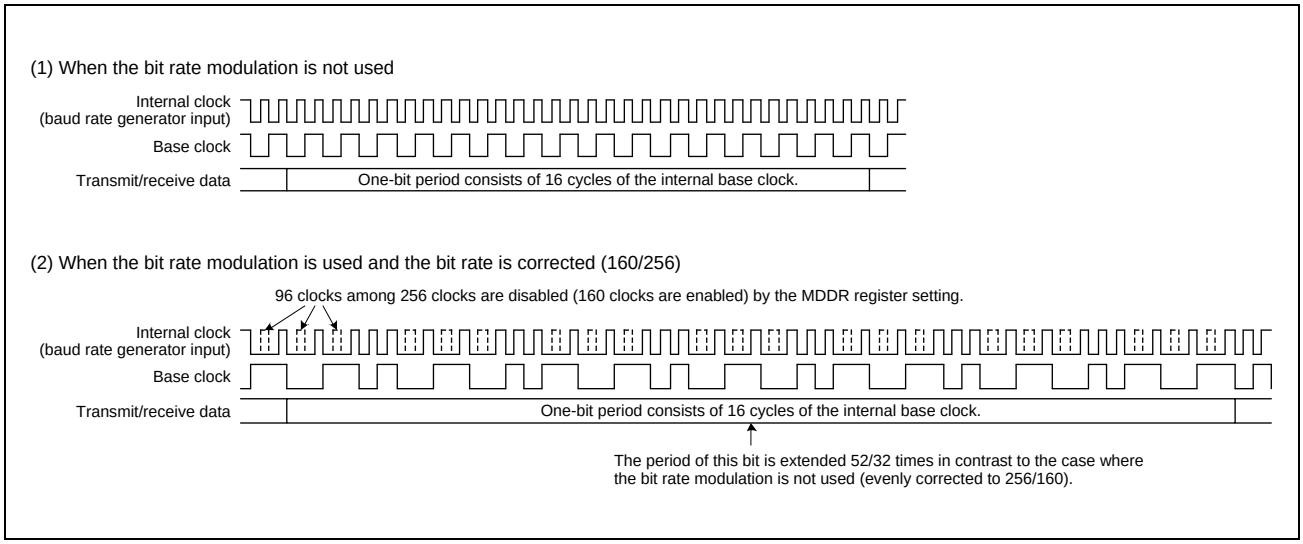


Figure 7.4-9 Example of Internal Base Clock when Bit Modulation is Used

7.4.5 Interrupt Sources

The SCIF has six interrupt sources: transmit-FIFO-data-empty (TXI), receive-error (ERI), receive-FIFO-data-full (RXI), break (BRI), transmit-end (TEI), receive-data-ready (DRI). The TEI, DRI, ERI, and BRI interrupts share the same vector number.

Table 7.4-11 shows the interrupt sources and priority. The interrupt sources can be enabled or disabled using the TIE, RIE, REIE, TEIE bits in the SCR register and are separately input to the interrupt controller.

When the quantity of transmit data written in the FTDR register as a result of transmission is equal to or less than the specified transmission trigger number, the TDFE flag in the serial status register (FSR) is set to 1b and a TXI interrupt request is generated.

When the data units equaling or exceeding the specified transmission trigger number are stored in the receive FIFO register (FRDR) and the RDF flag in the FSR register is set to 1b, a receive data full interrupt (RXI) request is generated. When the quantity of received data in the FRDR register is below the specified reception trigger number and no next data has been received yet even after the period of 15 ETUs elapsed*[1] from the last stop bit, the DR flag in the FSR register is set to 1b and a receive data ready interrupt (DRI) request is generated. In clock synchronous mode, a DRI interrupt request is not generated.

When the BRK flag in the FSR register or the ORER flag in the LSR register is set to 1b, a BRI interrupt request is issued. When the ER flag in the FSR register is set to 1b, an ERI interrupt request is issued.

When the TEND flag in the FSR register is set to 1b, a TEI interrupt request is issued.

When the RIE bit is cleared to 0b and the REIE bit in the SCR register is set to 1b, an ERI and a BRI interrupt requests are issued but an RXI interrupt request is not.

An TXI interrupt indicates that transmit data can be written and an RXI interrupt indicates that receive data is stored in the FRDR register.

Note 1. It is equivalent to 1 and half frames of 8-bit format with one stop bit (ETU: Element Time Unit).

Table 7.4-11 SCIF Interrupt Sources

Name	Level/ Edge	Interrupt Source	Interrupt Enable Bit	DMAC Activation
SCIF_ub1_brk_n	Level	Interrupt caused by break (BRK) or overrun (ORER).	RIE or REIE	Impossible
SCIF_ub1_rerr_n	Level	Interrupt caused by framing or parity (ER).	RIE or REIE	Impossible
SCIF_ub1_rxi_n	Level	Interrupt caused by receive FIFO data full (RDF).	RIE	Possible
SCIF_ub1_txi_n	Level	Interrupt caused by transmit FIFO data empty (TDFE).	TIE	Possible
SCIF_ub1_tei_nI	Level	Interrupt caused by transmit end (TEND).	TEIE	Impossible
SCIF_ub1_dri_n	Level	Interrupt caused by receive data ready (DR).	RIE	Impossible
SCIF_ub1_tei_dri_n	Level	Interrupt caused by transmit end (TEND) and receive data ready (DR)	TEIE and RIE	Impossible

Note: The TEI and DRI interrupts share the same vector number.
The ERI and BRI interrupts share the same vector number.
If CPU processing is used, clear the flag after the block transfer. If the DMAC is activated, access to the flags is prohibited.

7.4.6 Event Link Output

The SCIF handles event output for the event link controller (ELC) corresponding to the following sources.

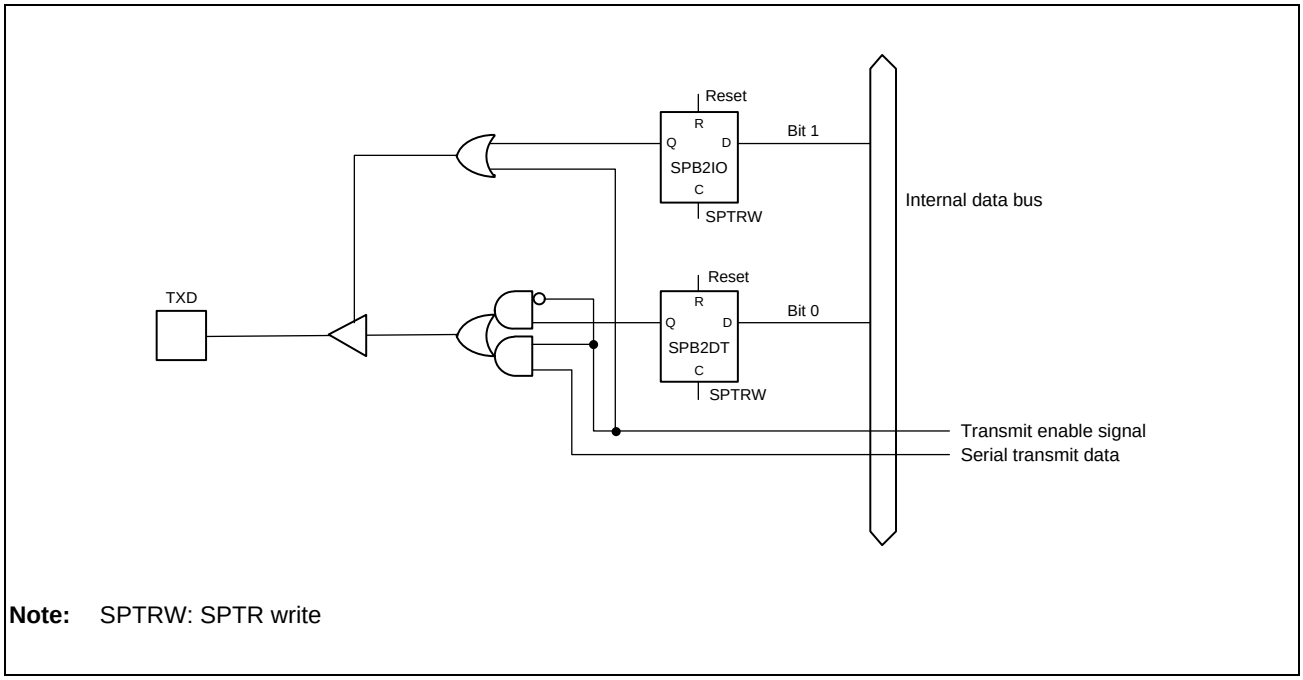
■ **Transmit FIFO data empty**

When a transmit data FIFO becomes empty, the corresponding event signal can be output for another module via the ELC.

■ **Receive FIFO data full**

When a receive data FIFO becomes full, the corresponding event signal can be output for another module via the ELC.

7.4.7 Serial Port Register (SPTR) and SCIF-Related Pins



Note: SPTRW: SPTR write

Figure 7.4-10 SPB2IO Bit and SPB2DT Bit in the SPTR Register, and TXD Pin

7.4.8 Noise Cancellation

Figure 7.4-11 shows the configuration of the noise filter used for noise cancellation. The noise filter consists of a two-stage flip-flop circuits and a match detection circuit. When the input signals of the noise filter and the output signals of the two-stage flip-flop circuits completely match, the matched level is conveyed as an internal signal. Unless otherwise matched, the previous value is retained. (When the levels sampled on three consecutive cycles of the sampling clock of the noise filter match, the signal is considered valid. If three consecutive sampled values do not match, the signal is considered to be noise rather than a received signal).

The noise cancellation can be applied to the receive signal input to the RXDn pin. The receive level of the RXDn pin is taken in the flip-flop circuit of the noise filter on the base clock (the clock with a frequency 16 or 8 times the transfer rate*[1]).

If the base clock is stopped once with the noise filter enabled and then the base clock input is restarted again, the noise filter operation resumes from the state where the clock was stopped. When SCR.RE is set to 0b during input of the base clock, the noise filter outputs 0 as the internal RxDn signal. The internal match detector continues operating even while operations for reception are stopped, and the result from the last time previous consecutive samples matched is output at the same time as operations for reception are resumed.

Note 1. A frequency 16 times bit rate when the SEMR.ABCS0 bit and the SEMR.BGDM bit are both 0b, and a frequency 8 times bit rate when either the SEMR.ABCS0 bit or the SEMR.BGDM bit is 1b.

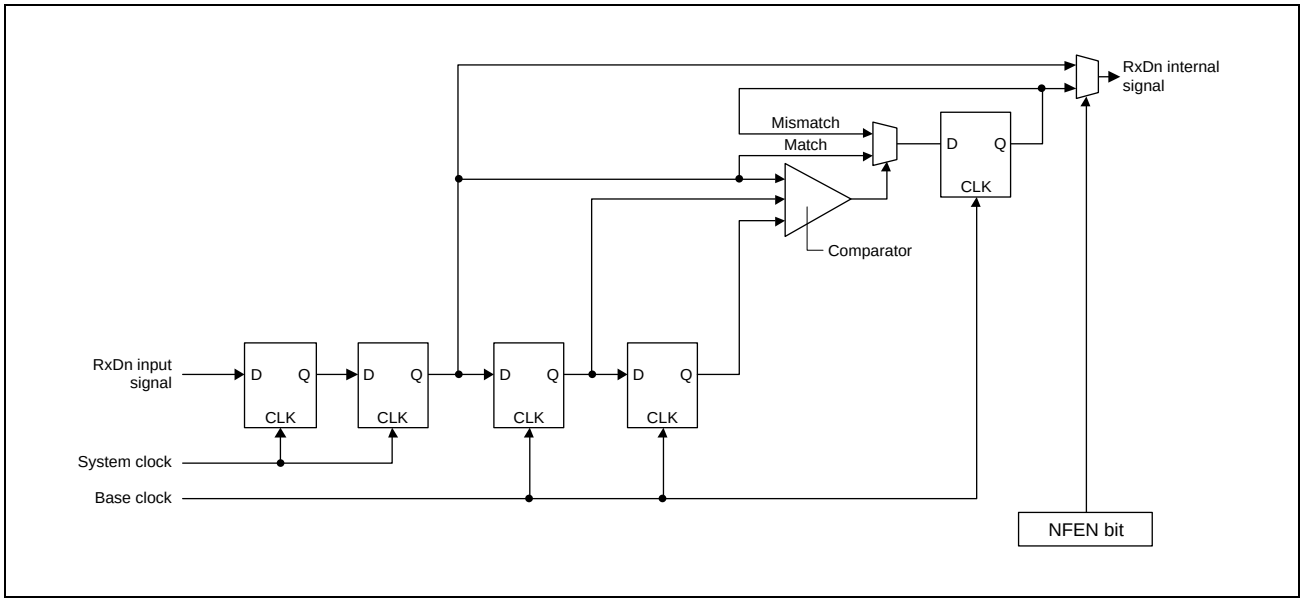


Figure 7.4-11 Block Diagram of Digital Noise Filter Circuit

7.4.9 Usage Notes

The following is the notes on using the SCIF.

7.4.9.1 FTDR Register Writing and TDFE Flag

The TDFE flag in the serial status register (FSR) is set when the number of transmit data bytes written in the transmit FIFO data register (FTDR) has fallen below the transmission trigger number set by bits TTRG[1:0] in the FIFO control register (FCR) or bits TFTC[4:0] in the FIFO trigger control register (FTCR). After the TDFE flag is set, transmit data up to the number of empty bytes in the FTDR register can be written, allowing efficient continuous transmission.

However, if the number of data bytes written in the FTDR register is equal to or less than the specified transmission trigger number, the TDFE flag will be set to 1b again even after being read as 1b and cleared to 0b.

The number of transmit data bytes in the FTDR register can be checked by the 8 higher-order bits of the FIFO data count register (FDR).

7.4.9.2 FRDR Register Reading and RDF Flag

The RDF flag in the serial status register (FSR) is set when the number of receive data bytes in the receive FIFO data register (FRDR) has become equal to or greater than the reception trigger number set by bits RTRG[1:0] in the FIFO control register (FCR) or bits RFTC[4:0] in the FIFO trigger control register (FTCR). After the RDF flag is set, receive data equivalent to the trigger number can be read from the FRDR register, allowing efficient continuous reception.

However, if the number of data bytes in the FRDR register exceeds the reception trigger number, the RDF flag will be set to 1b again even after being read as 1b and cleared to 0b.

The number of receive data bytes in the FRDR register can be checked by the 8 lower-order bits of the FIFO data count register (FDR).

7.4.9.3 Break Detection and Processing

When a framing error (FER) is detected, a break signal can be detected by reading the RXD pin value directly. In a break, the input from the RXD pin becomes all low. Therefore, the FER flag in the serial status register (FSR) is set to 1b and the parity error flag (PER) may also be set to 1b.

Upon detection of a break signal, the SCIF stops the received data transfer to the FRDR register but continues the receive operation.

7.4.9.4 Writing to the SPTR Register

b0 of the SPTR register indicates the input status of the corresponding pin. (See the description of the bit in **7.4.2.2.12 Serial Port Register (SCIFm**_SPTR) for details.)

7.4.9.5 Break Signal Transmission

The output signal from the TXD pin is determined by the SPB2IO bit and the SPB2DT bit in the serial port register (SPTR). The break signal can be sent by using these bits.

The TXD pin does not function as a transmit data output pin during the period from when the SCIF is initialized to when the TE bit in the SCR register is set to 1b (transmission possible). The TXD pin status during this period is replaced by the SPB2DT bit value. Therefore, the SPB2IO and SPB2DT bits in the SPTR register must have been set to 1b (high output) at first (mark (high) status).

To transmit the break signal during serial transmission, set the SPB2IO bit in the SPTR register to 1b, clear the SPB2DT bit to 0b (specify a low level), and then clear the TE bit in the SCR register to 0b (transmission stop). Clearing the TE bit to 0b initializes the transmitter regardless of the current transmission status, and outputs a low level from the TXD pin.

7.4.9.6 Receive Data Sampling Timing and Receive Margin

The SCIF operates on a base clock with a frequency 16 times the transfer rate*[1]. In reception, the SCIF internally latches the received data at the rising edge of the eighth base clock pulse*[1]. The timing is shown in **Figure 7.4-12**.

Note 1. This is an example when the SEMR.ABCS0 bit is 0b. When the ABCS0 bit is 1b, a frequency of 8 times the bit rate is used as a base clock and receive data is sampled at the rising edge of the 4th pulse of the base clock. The receive margin can therefore be expressed as shown in equation 1.

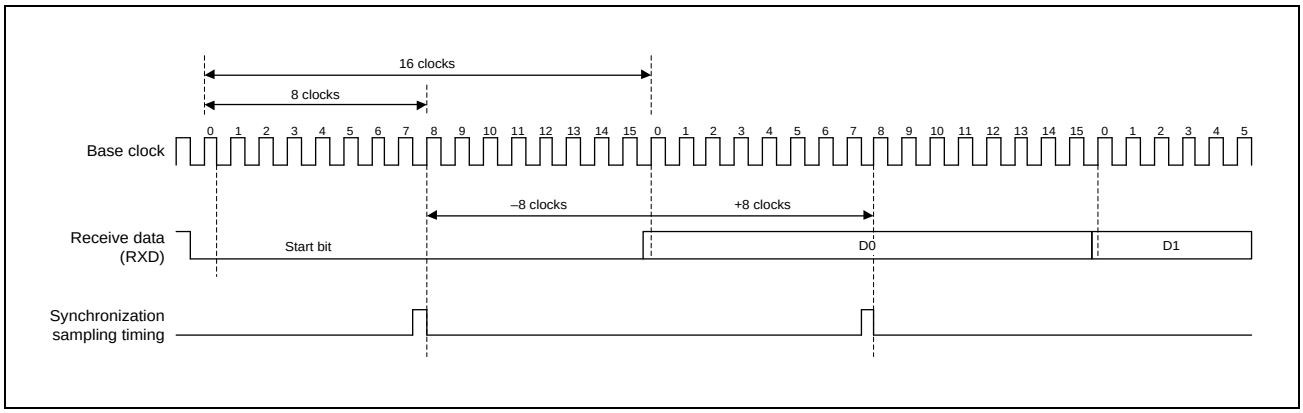


Figure 7.4-12 Receive Data Sampling Timing

The receive margin can therefore be expressed as shown in equation 1.

[Equation 1]

$$M = \{(0.5 - \frac{1}{2N} - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F)\} \times 100[\%]$$

Where: M: Receive margin (%)
N: Ratio of clock frequency to bit rate (N = 16)
D: Clock duty (D = 0 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute deviation of clock frequency

From equation 1, if F = 0 and D = 0.5, the receive margin is 46.875%, as given by equation 2.

[Equation 2]

When D = 0.5 and F = 0:

$$M = (\frac{0.5 - 1}{(2 \times 16)}) \times 100\% = 46.875\%$$

This is a theoretical value. A reasonable margin to allow in system designs is 20% to 30%.

7.4.9.7 Note on FER Flag and PER Flag in Serial Status Register (FSR)

The FER flag and PER flag in the serial status register (FSR) are status flags that apply to next entry to be read from the receive FIFO data register (FRDR). After the CPU or DMAC reads the receive FIFO data register, the flags of framing errors and parity errors in the receive data will be cleared. To check the received data for the states of framing errors and parity errors, only read the receive FIFO data register after reading the serial status register.

SECTION 7 LOW-SPEED INTERFACE

7.5 Serial Peripheral Interface (RSPI)

7.5.1 Overview

The RSPI (hereafter SPI) supports serial communications for transmit-receive, transmit only, and receive only. The SPI is provided with a function for high-speed serial communication with multiple processors and peripheral devices.

7.5.1.1 Features

Table 7.5-1 lists the specifications of the SPI, and **Figure 7.5-1** shows a block diagram of the SPI.

Table 7.5-1 SPI Specifications (1/3)

Item	Description
Number of channels	3 channels
Transfer functions	• SPI serial communication (4-wire) and clock synchronous (3-wire) serial communication are possible by using the MOSI (Master Out Slave In), MISO (Master In Slave Out), SSL (Slave Select), and RSPCK (SPI Clock) signals. • Transmit-only operation is available • Receive-only operation is available • Serial communication is possible in master mode and slave mode • RSPCK polarity switching • RSPCK phase switching
Data format	• MSB first or LSB first selectable. • Transfer bit length selectable from 4 to 32 bits • 32 bit × 16 stage FIFO transmit and receive buffers • Up to four frames transferable in one round of transmission or reception (each frame consisting of up to 32 bits) • Byte swap operating function • Transmit/receive data inversion
Bit rate	• In master mode, the on-chip baud rate generator divides the RSPI_n_TCLK to generate RSPCK. A division ratio of 2 to 4096 is settable. • In slave mode, an external input clock is used as a serial clock. The maximum frequency = RSPI_n_TCLK/2 (High width: 1 RSPI_n_TCLK cycle, low width: 1 RSPI_n_TCLK cycle)
Buffer configuration	• Transmit buffer and receive buffer are configured independently.
Error detection	• Mode fault error detection • Underrun error detection • Overrun error detection • Parity error detection

Table 7.5-1 SPI Specifications (2/3)

Item	Description
SSL control function	**[Motorola SPI mode]** • Four SSL pins (SSLn0 to SSLn3) each channel • In master mode, SSLn0 to SSLn3 pins are output. • In slave mode, SSLn0 pin for input, and SSLn1 to SSLn3 pins unused • Controllable delay from SSL output assertion to RSPCK operation (RSPCK delay) Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units) • Controllable delay from RSPCK stop to SSL output negation (SSL negation delay) Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units) • Controllable wait for next-access SSL output assertion (next-access delay) Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units) • Configurable delay between frames in burst transfer • Function for changing SSL polarity **[TI SSP mode]** • Four SSL pins (SSL0 to SSL3) each channel • In master mode, SSLn0 to SSLn3 pins are output. • In slave mode, SSLn0 pin for input, and SSLn1 to SSLn3 pins unused • Controllable delay from SSL output assertion to SSL output negation (RSPCK delay) Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units) • Controllable delay from RSPCK stop to Output disable (SSL negation delay) Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units) • Controllable wait for next-access SSL output assertion (next-access delay) Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units) • Configurable delay between frames in burst transfer • Function for changing SSL polarity
Communication Protocol	• Motorola SPI • TI SSP (Synchronous Serial Protocol)
Synchronization bypass function	• Synchronization circuit can be bypassed using bus clock (RSCI_n_PCLK) as operation clock (RSCI_n_TCLK)
Control in master transfer	**[Motorola SPI mode]** • Transfers of up to eight commands each can be executed sequentially in looped execution • For each command, the following can be set: SSL signal value, bit rate, RSPCK polarity and phase, transfer data length, LSB first/MSB first, burst, RSPCK delay, SSL negation delay, next-access delay • Transfers can be initiated by writing to the transmit buffer • MOSI signal value specifiable in SSL negation • RSPCK auto-stop function **[TI SSP mode]** • Transfers of up to eight commands each can be executed sequentially in looped execution • For each command, the following can be set: SSL signal value, bit rate, RSPCK polarity and phase, transfer data length, LSB first/MSB first, burst, RSPCK delay, SSL negation delay (Output disable delay), next- access delay • Transfers can be initiated by writing to the transmit buffer • MOSI signal value specifiable in SSL negation • RSPCK auto-stop function
Interrupt sources	• Maskable Interrupt sources SPI receive buffer full / Receive data ready interrupt SPI transmit buffer empty interrupt SPI communication end interrupt SPI error interrupt (mode fault, overrun, parity error, receive data ready) SPI idle interrupt (SPI idle)

Table 7.5-1 SPI Specifications (3/3)

Item	Description
Others	• SPI disable (initialization) function • Loopback mode function
Module-stop function	Module-stop state can be set to reduce power consumption*1

Note 1. Module stop condition: It depends on the setting higher than this module. This module is not reset, and the clock stops on module stop condition. It contributes to low power consumption, because the clock stops.

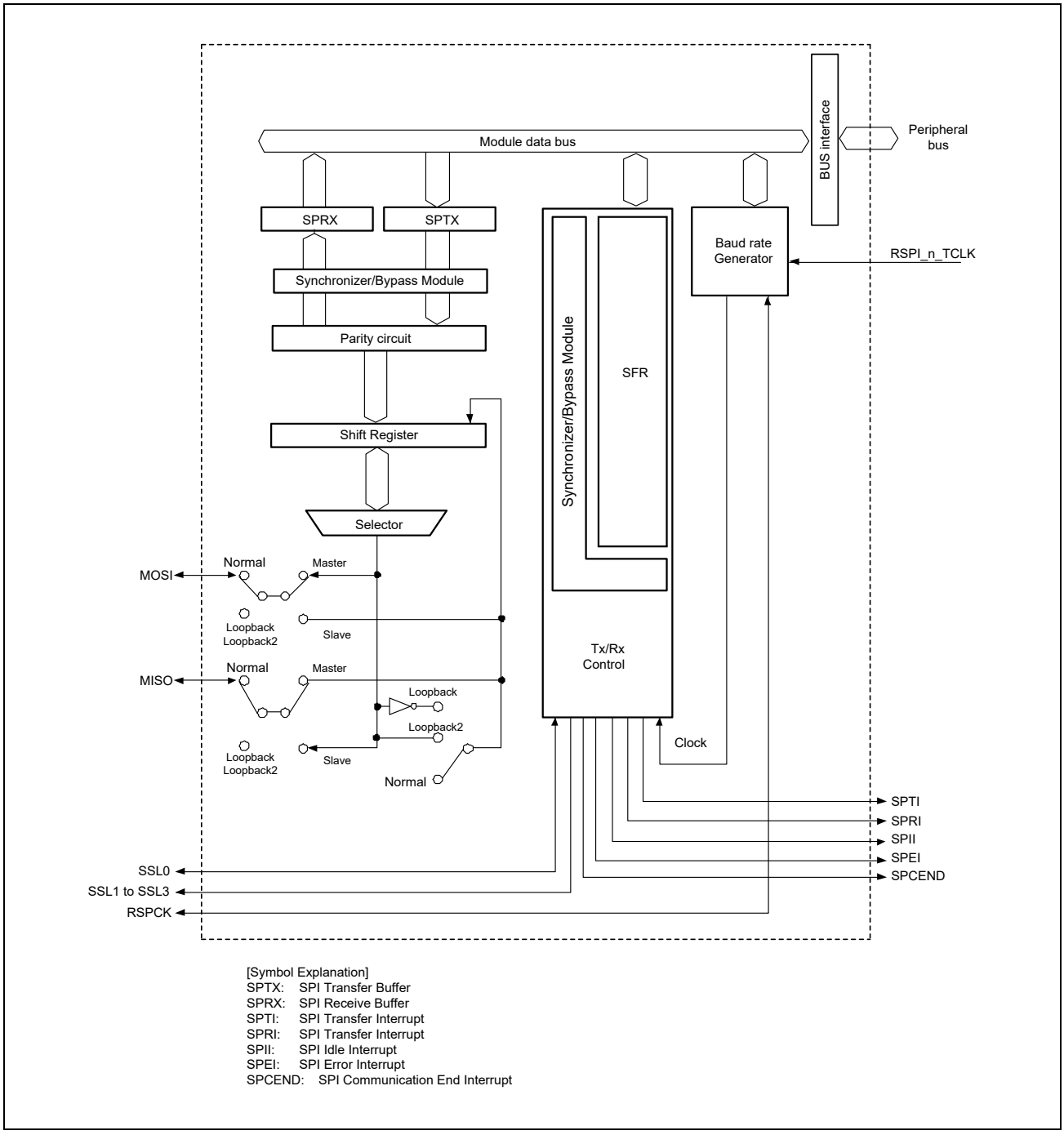


Figure 7.5-1 SPI Block Diagram

The SPI automatically switches the SSL0 pin input and output directions. The SSL0 pin is switched to output when master is set, and it is switched to input when slave is set. Furthermore, the SPI automatically switches the input and output directions of pins RSPCK, MOSI, MISO, SSL1 to SSL3 according to the master/slave setting and SPI operation (4-wire)/clock synchronous operation (3-wire) SSL0 input level. (See **7.5.3.2 SPI Pin Control**.)

Table 7.5-2 lists the I/O pins used in the SPI.

Table 7.5-2 SPI Pin Configuration

Channel	Pin Name	Signal Name	Input/Output	Function
RSPI0	RSPCKA	RSPCK	I/O	Clock I/O
	MOSIA	MOSI	I/O	Master transmit data I/O
	MISOA	MISO	I/O	Slave transmit data I/O
	SSLA0	SSL0	I/O	Slave selection signal I/O
	SSLA1	SSL1	Output	Slave selection signal output
	SSLA2	SSL2	Output	Slave selection signal output
	SSLA3	SSL3	Output	Slave selection signal output
RSPI1	RSPCKB	RSPCK	I/O	Clock I/O
	MOSIB	MOSI	I/O	Master transmit data I/O
	MISOB	MISO	I/O	Slave transmit data I/O
	SSLB0	SSL0	I/O	Slave selection signal I/O
	SSLB1	SSL1	Output	Slave selection signal output
	SSLB2	SSL2	Output	Slave selection signal output
	SSLB3	SSL3	Output	Slave selection signal output
RSPI2	RSPCKC	RSPCK	I/O	Clock I/O
	MOSIC	MOSI	I/O	Master transmit data I/O
	MISOC	MISO	I/O	Slave transmit data I/O
	SSLC0	SSL0	I/O	Slave selection signal I/O
	SSLC1	SSL1	Output	Slave selection signal output
	SSLC2	SSL2	Output	Slave selection signal output
	SSLC3	SSL3	Output	Slave selection signal output

7.5.2 Registers

The base addresses for the respective channels are as follows.

Table 7.5-3 Register Base Addresses

Base Register Name	Channel	Base Address
<RSPI0_base>	RSPI0	0_1280_0000h (5280_0000h*1, 4280_0000h*2)
<RSPI1_base>	RSPI1	0_1280_0400h (5280_0400h*1, 4280_0400h*2)
<RSPI2_base>	RSPI2	0_1280_0800h (5280_0800h*1, 4280_0800h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

7.5.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
SPI Data Register	RSPIm_SPDR/SPDR_HA/SPDR_BY	0000_0000h	0000h	8, 16, 32
SPI Clock Delay Register	RSPIm_SPCKD	00h	0004h	8
SPI Slave Select Negation Delay Register	RSPIm_SSLND	00h	0005h	8
SPI Next-Access Delay Register	RSPIm_SPND	00h	0006h	8
SPI Clock Digital Control Register for Master Receive	RSPIm_MRCKD	00h	0007h	8
SPI Control Register	RSPIm_SPCR	0000_0000h	0008h	8, 16, 32
SPI Control Register for Master Receive only	RSPIm_SPCRRM	00h	000Ch	8
SPI Control Register for Received Data Ready Detection	RSPIm_SPDRCR	00h	000Dh	8
SPI Pin Control Register	RSPIm_SPPCR	00h	000Eh	8
SPI Control Register 2	RSPIm_SPCR2	00h	000Fh	8
SPI Slave Select Polarity Register	RSPIm_SSLP	00h	0010h	8
SPI Bit Rate Register	RSPIm_SPBR	FFh	0011h	8
Reserve	-	-	0012h to 0012h	-
SPI Sequence Control Register	RSPIm_SPSCR	00h	0013h	8
SPI Command Register 0	RSPIm_SPCMD0	0007_0000h	0014h	8, 16, 32
SPI Command Register 1	RSPIm_SPCMD1	0007_0000h	0018h	8, 16, 32
SPI Command Register 2	RSPIm_SPCMD2	0007_0000h	001Ch	8, 16, 32
SPI Command Register 3	RSPIm_SPCMD3	0007_0000h	0020h	8, 16, 32
SPI Command Register 4	RSPIm_SPCMD4	0007_0000h	0024h	8, 16, 32
SPI Command Register 5	RSPIm_SPCMD5	0007_0000h	0028h	8, 16, 32
SPI Command Register 6	RSPIm_SPCMD6	0007_0000h	002Ch	8, 16, 32
SPI Command Register 7	RSPIm_SPCMD7	0007_0000h	0030h	8, 16, 32
Reserve	-	-	0034h to 003Fh	-
SPI Data Control Register	RSPIm_SPDCR	0000h	0040h	8, 16
Reserve	-	-	0042h to 0043h	-
SPI Data Control Register 2	RSPIm_SPDCR2	0000h	0044h	8, 16
Reserve	-	-	0046h to 0050h	-
SPI Sequence Status Register	RSPIm_SPSSR	00h	0051h	8
SPI Status Register	RSPIm_SPSR	2000h	0052h	8, 16
Reserve	-	-	0054h to 0057h	-
SPI Transfer FIFO Status Register	RSPIm_SPTFSR	10h	0058h	8
Reserve	-	-	0059h to 005Bh	-
SPI Receive FIFO Status Register	RSPIm_SPRFSR	00h	005Ch	8
Reserve	-	-	005Dh to 005Fh	-
SPI Polling Register	RSPIm_SPPSR	0000_0000h	0060h	8, 16, 32
Reserve	-	-	0064h to 0069h	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
SPI Status Clear Register	RSPIm_SPSRC	0000h	006Ah	8, 16
SPI FIFO Clear Register	RSPIm_SPFCR	00h	006Ch	8

Note 1. The read access size is fixed at 32 bits.

7.5.2.2 Register Description

The prefix (RSPIm_) of the register names is omitted in this and subsequent sections.

7.5.2.2.1 SPI Data Register (RSPIm_SPDR/SPDR_HA/SPDR_BY)

Access Size : 8, 16, 32 bits
Address : <RSPIm_base> + 0000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPD[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

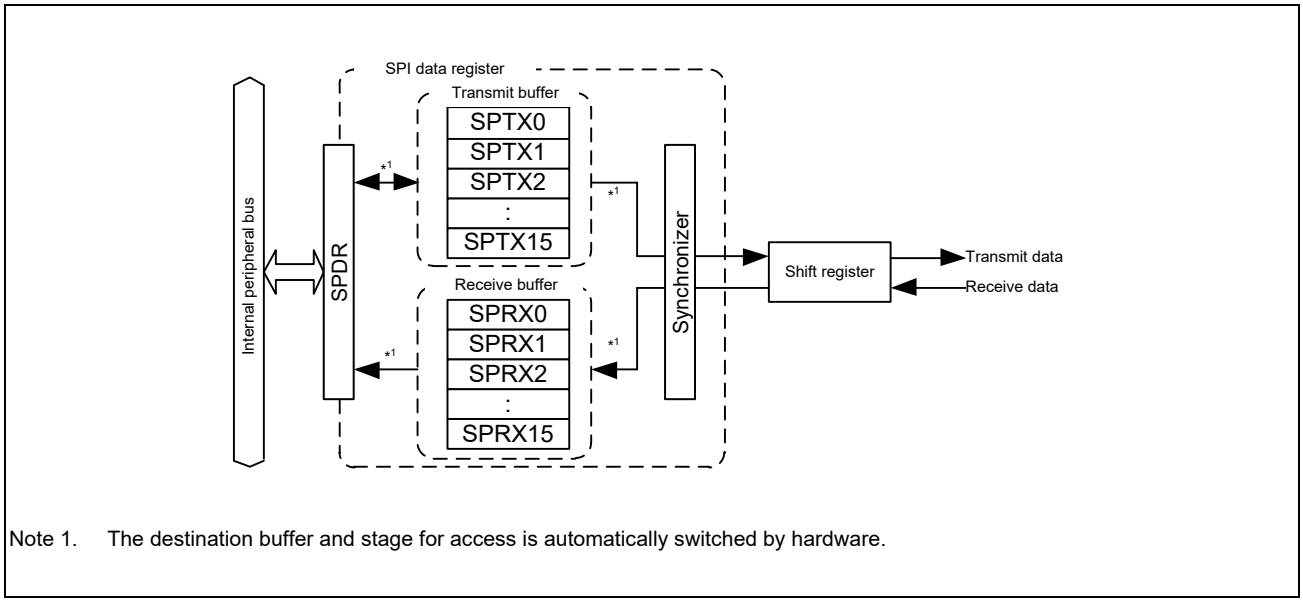
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SPD[31:0]	0h	RW	The SPI data register (SPDR) is used to store SPI's transmit data and receive data. Transmit buffers and receive buffers independently function.

Note: The read access size is fixed at 32 bits.

SPDR is the interface with the buffers that hold data for transmission and reception by the SPI.

Figure 7.5-2 shows the structure of SPDR.



Note 1. The destination buffer and stage for access is automatically switched by hardware.

Figure 7.5-2 Structure of SPDR

32 bit × 16 stage transmit FIFO and 32 bit × 16 receive FIFO are provided. These 32 stage FIFO are mapped to one address in the SPDR. Transmit buffers (SPTXn, n = 0 to 15) can be written by writing data to SPDR to transmit written data.

Upon completion of receiving data, receive buffers store received data. When an overrun error occurs, data in the receive buffer is not updated.

Bus Interface

The SPI data register has 32 bit × 16 stage transmit FIFO and 32 bit × 16 receive FIFO (32 bytes in total). These 128 bytes are mapped to the 4-byte space of SPDR. Write transmit data from the LSB. Received data is stored from the LSB.

SPDR register write operation and read operation are described below.

a) Write

A transmit buffer write pointer is provided for transmit buffers. When data is written to SPDR, the pointer automatically switches to the next buffer. The following illustrates the structure of the transmit buffer bus interface (write).

Figure 7.5-3 shows the configuration of the bus interface with the transmission buffer in the case of writing to SPDR.

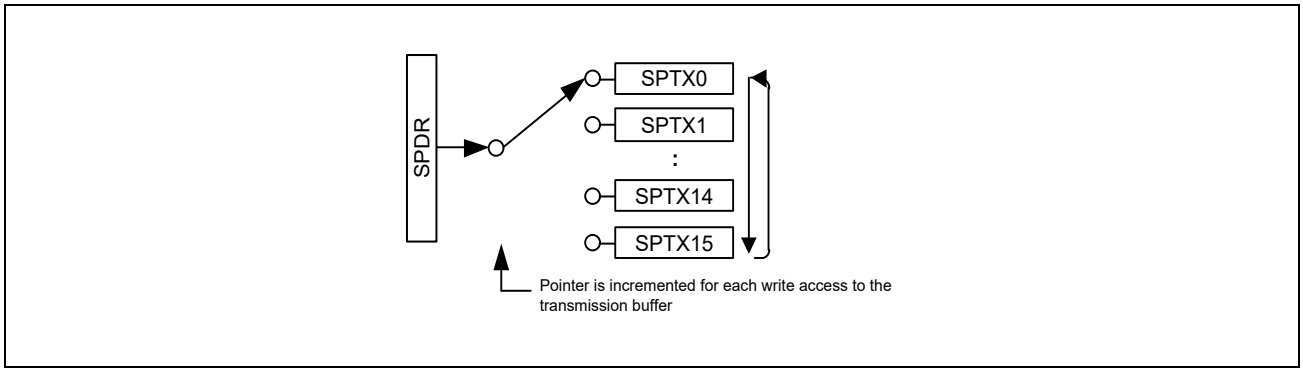


Figure 7.5-3 Structure of SPDR (Write)

The transmit buffer (SPTX0 to SPTX3) switching order:

SPTX0→SPTX1→SPTX2→SPTX3→…→SPTX15→SPTX0→SPTX1→…

When writing transmit data to transmit buffers (SPTXn), write transmit data of frames +1 specified by the Transmission FIFO threshold setting bits of SPI data control register 2 (SPDCR2.TTRG[1:0]) while an SPI transmit buffer empty interrupt is present (SPSR.SPTEF flag = 1b). Writing to the transmit buffer (SPTXn, n = 0 to 15) in the state where there is no empty stage in the transmit FIFO does not update the buffer value.

b) Read

Values can be read from receive buffers (SPRXn, n = 0 to 15) or transmit buffers (SPTXn, n = 0 to 15) by reading the SPDR register. Reading a receive buffer or reading a transmit buffer can be selected by the SPI receive data or transmit data select bit (SPDCR.SPRDTD).

The SPDR register is read according to the independent receive buffer read pointer and the transmit buffer read pointer.

The following illustrates the structure of the receive buffer and transmit buffer bus interface (read).

Figure 7.5-4 shows the configuration of the bus interface with the receive and transmission buffers in the case of reading from SPDR.

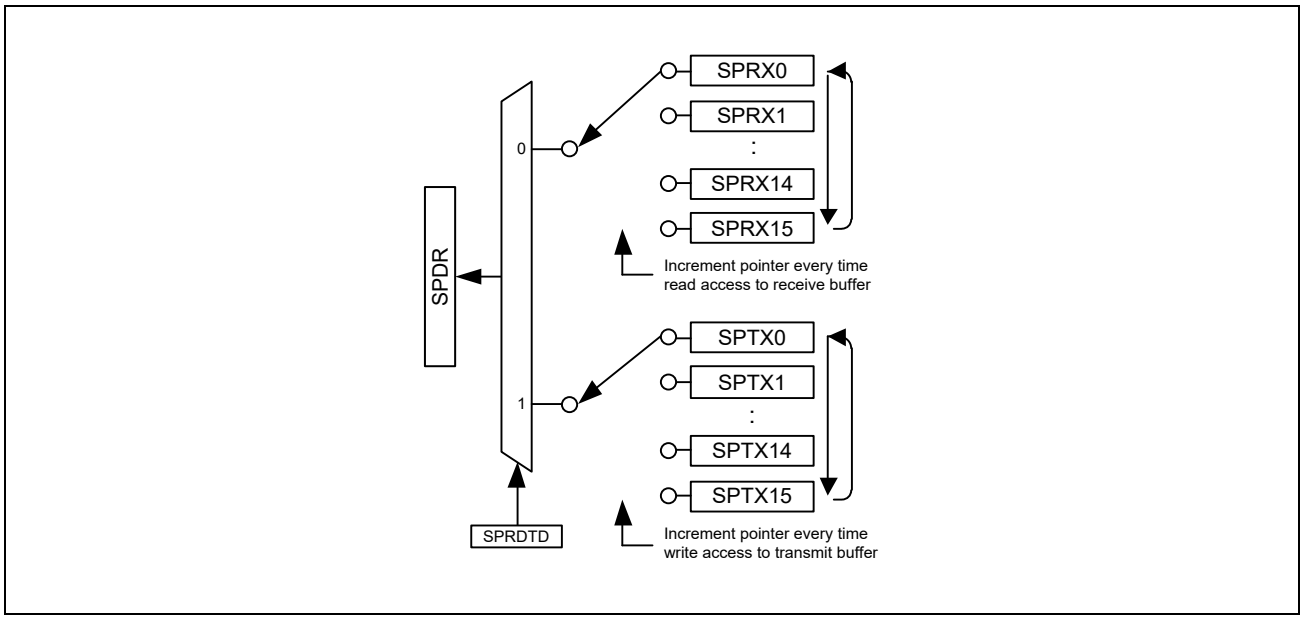


Figure 7.5-4 Structure of SPDR (Read)

When a receive buffer is read, the receive buffer read pointer automatically switches to the next buffer. The receive buffer read pointer switches in the same order as the transmit buffer write pointer.

The transmit buffer read pointer is updated during the SPDR write access, but it is not updated during the transmit buffer read access. When a transmit buffer is read, the value written to SPDR last can be read.

7.5.2.2.2 SPI Clock Delay Register (RSPIm_SPCKD)

Access Size : 8 bits
Address : <RSPIm_base> + 0004h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	SCKDL[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	SCKDL[2:0]	0h	RW	RSPCK Delay Setting 000b: 1 RSPCK 001b: 2 RSPCK 010b: 3 RSPCK 011b: 4 RSPCK 100b: 5 RSPCK 101b: 6 RSPCK 110b: 7 RSPCK 111b: 8 RSPCK

[In the Motorola-SPI case]

The SPCKD register is used to set the period (RSPCK delay) from the start of assertion of the SSL signal to the RSPCK oscillation while SPCMD.SCKDEN = 1b. If SPCKD is modified while SPCR.MSTR = 1b and SPCR.SPE = 1b, subsequent operation is not guaranteed.

[In the TI-SSP case]

The SPCKD register is used to set the period (RSPCK delay) from the start of assertion of the SSL signal to the RSPCK oscillation while SPCMD.SCKDEN = 1b. Also, it is used to set the period until the SSL signal is negated. If SPCKD is modified while SPCR.MSTR = 1b and SPCR.SPE = 1b, subsequent operation is not guaranteed.

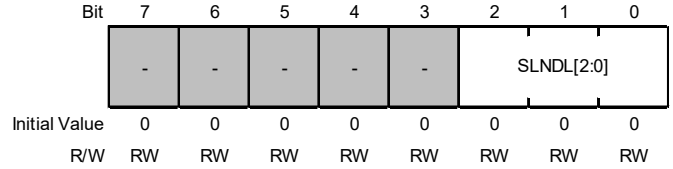
SCKDL[2:0] bits (RSPCK Delay Setting)

These bits are used to set the RSPCK delay value when SPCMD.SCKDEN = 1b.

To use the SPI in slave mode, set SCKDL[2:0] bits to 0b.

7.5.2.2.3 SPI Slave Select Negation Delay Register (RSPIm_SSLND)

Access Size : 8 bits
Address : <RSPIm_base> + 0005h
Initial Value : 00h



Bit	Bit Name	Initial Value	R/W	Description
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	SLNDL[2:0]	0h	RW	SSL Negation Delay Bits This is the synchronization delay for replacing the RSPCK input with RSPI_n_TCLK. RSPCK Delay Setting 000b: 1 RSPCK [Master mode] 1 RSPI_n_TCLK + (2~3 RSPI_n_TCLK (Synchronization delay)) [TI-SSP case in Slave Mode] 001b: 2 RSPCK [Master mode] 2 RSPI_n_TCLK + (2~3 RSPI_n_TCLK (Synchronization delay)) [TI-SSP case in Slave Mode] 010b: 3 RSPCK [Master mode] 3 RSPI_n_TCLK + (2~3 RSPI_n_TCLK (Synchronization delay)) [TI-SSP case in Slave Mode] 011b: 4 RSPCK [Master mode] 4 RSPI_n_TCLK + (2~3 RSPI_n_TCLK (Synchronization delay)) [TI-SSP case in Slave Mode] 100b: 5 RSPCK [Master mode] 5 RSPI_n_TCLK + (2~3 RSPI_n_TCLK (Synchronization delay)) [TI-SSP case in Slave Mode] 101b: 6 RSPCK [Master mode] 6 RSPI_n_TCLK + (2~3 RSPI_n_TCLK (Synchronization delay)) [TI-SSP case in Slave Mode] 110b: 7 RSPCK [Master mode] 7 RSPI_n_TCLK + (2~3 RSPI_n_TCLK (Synchronization delay)) [TI-SSP case in Slave Mode] 111b: 8 RSPCK [Master mode] 8 RSPI_n_TCLK + (2~3 RSPI_n_TCLK (Synchronization delay)) [TI-SSP case in Slave Mode]

[In the Motorola-SPI case]

The SSLND register is used to set the period (SSL negation delay) after the SPI in master mode sends the final RSPCK edge during serial transfer until it negates the SSL signal while SPCMD.SCKDEN = 1b. If SSLND is modified while SPCR.MSTR = 1b and SPCR.SPE = 1b, subsequent operation is not guaranteed.

[In the TI-SSP case]

The SSLND register is used to set the period (Output disable delay) after the SPI in master mode sends the final RSPCK edge during serial transfer until it negates the Output enable signal while SPCMD.SCKDEN = 1b. Also, that is used to set the period from when the SPI in slave mode detects the last RSPCK edge of serial transfer to when the Output enable signal is negated. If SSLND is modified while the SPE bit in SPCR.SPE = 1b, subsequent operation is not guaranteed.

SLNDL[2:0] bits (SSL Negation Delay Bits)

[In the Motorola-SPI case]

These bits are used to set the SSL negation delay value when SPCMD.SLNDEN = 1b.
To use the SPI in slave mode except TI-SSP, set SLNDL[2:0] bits to 0b.

[In the TI-SSP case]

These bits are used to set the Output disable delay value when SPCMD.SLNDEN = 1b.

7.5.2.2.4 SPI Next-Access Delay Register (RSPIm_SPND)

The SPND register is used to set the SSL signal inactive period (next-access delay) after completion of serial transfer while SPCMD.SCKDEN = 1b. If SPND is modified while SPCR.MSTR = 1b and SPCR.SPE = 1b, subsequent operation is not guaranteed.

Access Size : 8 bits
Address : <RSPIm_base> + 0006h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	SPNDL[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	SPNDL[2:0]	0h	RW	SPI Next-Access Delay Bits RSPCK Delay Setting 000b: 1 RSPCK + 5 RSPI_n_TCLK 001b: 2 RSPCK + 5 RSPI_n_TCLK 010b: 3 RSPCK + 5 RSPI_n_TCLK 011b: 4 RSPCK + 5 RSPI_n_TCLK 100b: 5 RSPCK + 5 RSPI_n_TCLK 101b: 6 RSPCK+ 5 RSPI_n_TCLK 110b: 7 RSPCK + 5 RSPI_n_TCLK 111b: 8 RSPCK + 5 RSPI_n_TCLK

SPNDL[2:0] bits (SPI Next-Access Delay Bits)

These bits are used to set the next-access delay value when SPCMD.SCKDEN = 1b. To use the SPI in slave mode, set SPNDL[2:0] bits to 0b.

7.5.2.2.5 SPI Clock Digital Control Register for Master Receive (RSPIm_MRCKD)

The MRCKD register is a register for adjusting the reception sampling timing in master mode. If the ARST[2:0] bits are rewritten while SPCR.SPE = 1b, subsequent operations are not guaranteed.

Access Size : 8 bits
Address : <RSPIm_base> + 0007h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	ARST[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	ARST[2:0]	0h	RW	Receive Sampling Timing Adjustment Bits The sampling of the serial data (MISO) input in master mode can be adjusted by delaying 0 to 7 RSPI_n_TCLK from the center of the bit. 000b: 0 RSPI_n_TCLK delay 001b: 1 RSPI_n_TCLK delay 010b: 2 RSPI_n_TCLK delay 011b: 3 RSPI_n_TCLK delay 100b: 4 RSPI_n_TCLK delay 101b: 5 RSPI_n_TCLK delay 110b: 6 RSPI_n_TCLK delay 111b: 7 RSPI_n_TCLK delay

ARST[2:0] bits (Receive Sampling Timing Adjustment Bits)

These bits are used to select the amount of reception sampling timing adjustment. The sampling timing can be adjusted by delaying 0 to 7 RSPI_n_TCLK from the center of the bit.

Delay setting exceeding the dividing ratio is prohibited.

This bit setting is valid only when the loopback function is not used (when SPPCR.SPLP = 0b, when MRCLK is selected (SPCR.SPSCKSEL = 1b), and SPPCR.SPLP2 = 0b) and in master mode. For details, see SPSCKSEL bit of **7.5.2.2.6 SPI Control Register (RSPIm_SPCR)**.

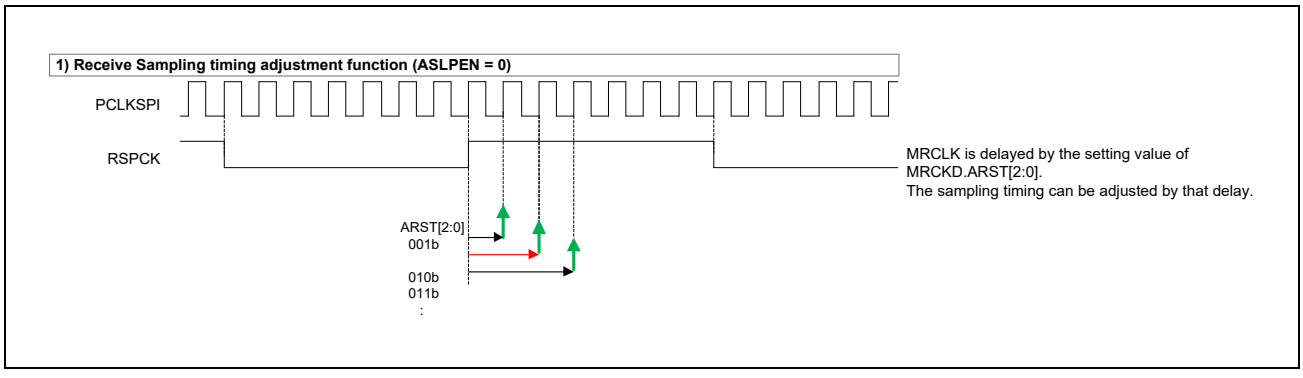


Figure 7.5-5 Description about Receive Sampling Timing Adjustment Function

7.5.2.2.6 SPI Control Register (RSPIm_SPCR)

The SPCR register is used to set operating mode of the SPI. If the set MSTR, TXMD[1:0], SPFRF, SPMS, MODFEN, BFDS, SCKASE, PTE, SPOE, SPPE, SPSCKSEL, and BPEN bit value is modified while SPCR.SPE = 1b, subsequent operation is not guaranteed.

Access Size : 8, 16, 32 bits
Address : <RSPIm_base> + 0008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BPEN	MSTR	TXMD[1:0]		-	-	SPFRF	SPMS	-	-	CENDIE	SPTIE	SPDRES	SPIIE	SPRIE	SPEIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	MODFEN	BFDS	SCKASE	PTE	-	SPOE	SPPE	SPSCKSEL	-	-	-	-	-	-	SPE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	BPEN	0h	RW	Synchronization Circuit Bypass Enable 0b: Non-Bypass 1b: Bypass
30	MSTR	0h	RW	SPI Master/Slave Mode Select 0b: Slave mode 1b: Master mode
29, 28	TXMD[1:0]	0h	RW	Communication Mode Select 00b: Transmit-Receive 01b: Transmit only 1xb: Receive only
27, 26	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
25	SPFRF	0h	RW	SPI Frame Format Select When SPMS = 1 (clock synchronous operation (3-wire)), this bit setting is invalid. 0b: Motorola-SPI 1b: TI-SSP
24	SPMS	0h	RW	SPI Function Enable 0b: SPI operation (4-wire) 1b: Clock synchronous operation (3-wire)
23, 22	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
21	CENDIE	0h	RW	SPI Communication End Interrupt Enable 0b: Communication end interrupt request is disabled. 1b: Communication end interrupt request is enabled.
20	SPTIE	0h	RW	SPI Transmit Buffer Empty Interrupt Enable 0b: SPI transmit buffer empty interrupt request is disabled. 1b: SPI transmit buffer empty interrupt request is enabled.
19	SPDRES	0h	RW	SPI Receive Data Ready Error Select 0b: Receive data full interrupt 1b: Error interrupt
18	SPIIE	0h	RW	SPI Idle Interrupt Enable 0b: Idle interrupt request is disabled. 1b: Idle interrupt request is enabled.
17	SPRIE	0h	RW	SPI Receive Buffer Full Interrupt Enable 0b: SPI receive buffer full interrupt request is disabled. 1b: SPI receive buffer full interrupt request is enabled.
16	SPEIE	0h	RW	SPI Error Interrupt Enable 0b: SPI error interrupt request is disabled. 1b: SPI error interrupt request is enabled.

Bit	Bit Name	Initial Value	R/W	Description
15	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
14	MODFEN	0h	RW	Mode Fault Error Detection Enable 0b: Mode fault error detection is disabled. 1b: Mode fault error detection is enabled.
13	BFDS	0h	RW	Between Burst Transfer Frames Delay Select 0b: Delay (RSPCK delay, SSL negation delay and next-access delay) between frames is inserted in burst transfer. 1b: Delay between frames is not inserted in burst transfer.
12	SCKASE	0h	RW	RSPCK Auto-Stop Function Enable 0b: RSPCK auto-stop function is disabled. 1b: RSPCK auto-stop function is enabled.
11	PTE	0h	RW	Parity Self-Diagnosis Enable 0b: Parity circuit self-diagnosis function is disabled. 1b: Parity circuit self-diagnosis function is enabled.
10	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	SPOE	0h	RW	Parity Mode 0b: Even parity is used for transmission and reception. 1b: Odd parity is used for transmission and reception.
8	SPPE	0h	RW	Parity Enable 0b: A parity bit is not added to transmit data. Received-data parity check is not performed. 1b: A parity bit is added to transmit data. Received-data parity check is performed.
7	SPSCKSEL	0h	RW	SPI Master Receive Clock Select 0b: MRIOCLK (Adjust with Analog Delay (SPCR2.SPSCKDL)) 1b: MRCLK (Adjust with Digital Delay (MRCKD.ARST))
6 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	SPE	0h	RW	SPI Function Enable 0b: SPI function is disabled. 1b: SPI function is enabled.

Note: The read access size is fixed at 32 bits.

SPE bit (SPI Function Enable)

The SPE bit is used to enable or disable SPI functions. Setting this bit to 1b enables SPI functions. When SPSR.MODF = 1b, the SPCR.SPE bit is cleared to 0b and the SPCR.SPE bit cannot be set to 1b until the MODF flag is cleared to 0b. (See **7.5.3.10 Error Detection**.) Setting the SPE bit to 0b disables SPI functions and initializes a part of module functions. (See **7.5.3.12 SPI Initialization**.)

SPSCKSEL bit (SPI Master Receive Clock Select)

The SPSCKSEL bit selects the clock used to sample the received data in master mode. By setting this bit, the sampling timing of the MISO can be adjusted. This bit setting is valid only when the loopback function is not used (when SPPCR.SPLP = 0b and SPPCR.SPLP2 = 0b) and in master mode.

Figure 7.5-6 shows a Master Receive Clock Delay configuration. By selecting this bit, you can select MRIOCLK (adjust with analog delay (SPCR2.SPSCKDL)) or MRCLK (adjust with digital delay (MRCKD.ARST)).

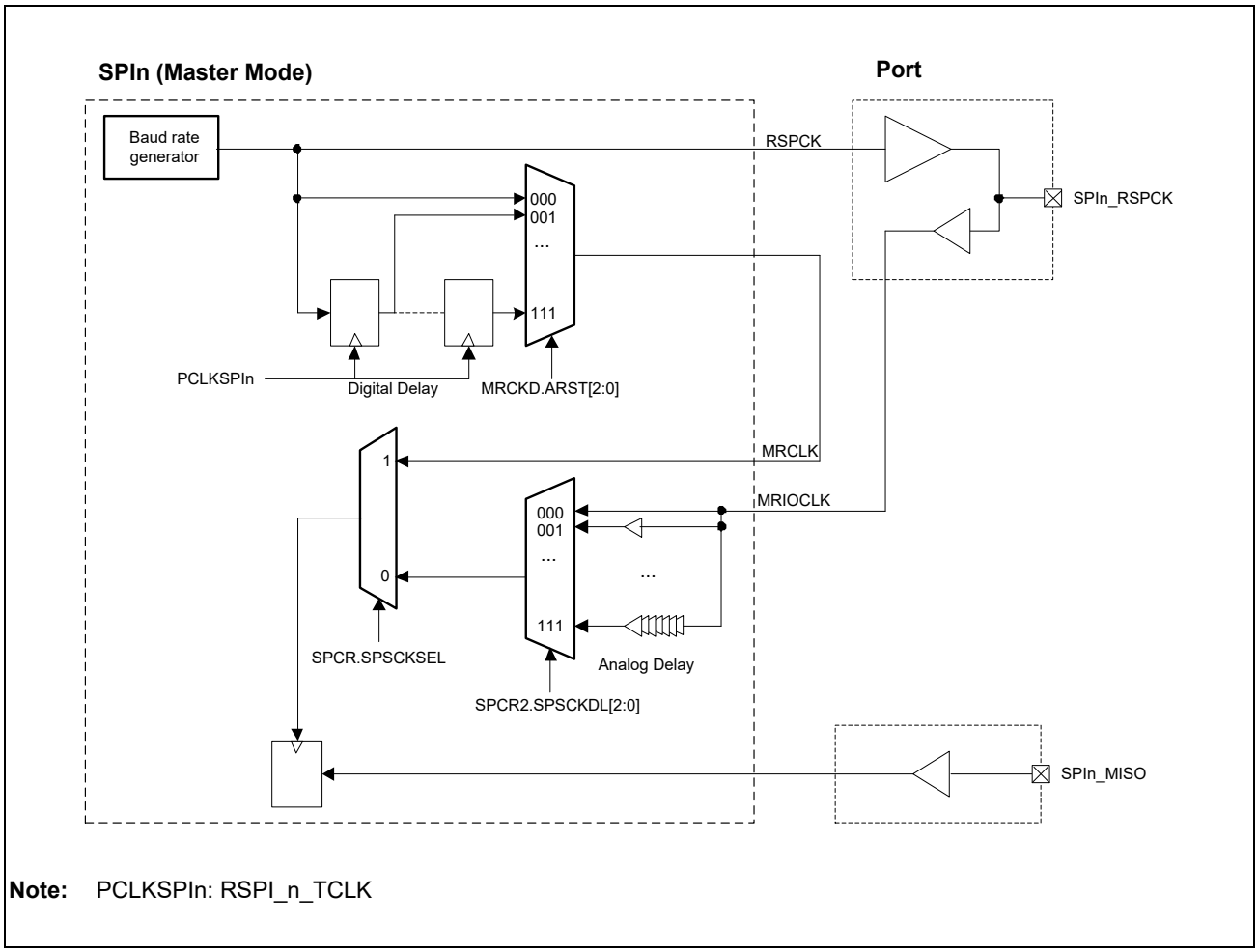


Figure 7.5-6 Master Receive Clock Delay Configuration (Master Mode)

SPPE bit (Parity Enable)

The SPPE bit is used to enable or disable the parity function.

SPOE bit (Parity Mode)

The SPOE bit is used to specify even parity or odd parity.

In even parity mode, the parity bit is determined so that the sum of 1 (parity bit + transmit characters or receive characters) becomes an even number. In the same way, in odd parity mode, a parity bit is determined so that the sum of 1 (parity bit + transmit characters or receive characters) becomes an odd number. The SPOE bit is valid only when the SPCR.SPPE is set to 1b.

PTE bit (Parity Self-Diagnosis Enable)

The PTE bit is used to enable or disable self-diagnosis of the parity circuit to confirm that the parity function is normal.

SCKASE bit (RSPCK Auto-Stop Function Enable)

The SCKASE bit is used to enable or disable the RSPCK auto-stop function. When this function is enabled, the RSPCK clock stops immediately before an overrun error occurs during data reception in master mode. For details, see **7.5.3.10.1 Overrun error**.

BFDS bit (Between Burst Transfer Frames Delay Select)

The BFDS bit controls whether insert the delay time between the burst transfer frames.

Valid in the master mode (SPCR.MSTR = 1b) for frames with the SPCMDm.SSLKP bit set to 1b.

This bit should be set to 0b in slave mode. The usage of SSL delay control between transfer frames is shown as below.

For details, see **(4) Burst transfer** in **7.5.3.13.2 Slave mode operation**.

(1) Non-burst transmits

(2) Burst transmit with delay between frames
(2-1) From the 1st frame to the last previous frame
(2-2) The last frame

(3) Burst transmit with no delay between frames
(3-1) From the 1st frame to the last previous frame
(3-2) The last frame

Table 7.5-4 Usage of SSL Delay Control between Transfer Frames (Master Mode)

No.	SPCMDm SSLKP Bit	SPCR BFDS Bit	SSL Delay Control Register*1 (RSPCK clock delay, SSL negation delay, next access delay)
(1)	0b	0b	Any given value. You can control each delay value according to setting for RSPCK clock delay, SSL negation delay, and next access delay.
(2-1)	1b	0b	
(2-2)	0b	0b	
(3-1)	1b	1b	Any given value. But delay is inserted only below. • RSPCK clock delay of the 1st frame • SSL negation delay and next access delay of the last frame
(3-2)	0b	1b	

Note 1. Whether the setting value of following bits are valid or not depends on the setting value of the SPCMD.SPNDEN bit. (See **7.5.2.2.14 SPI Command Register n (RSPIm_SPCMDn) (n = 0 to 7)**.)
The SPCKD.SCKDL[2:0] bits: RSPCK delay
The SSLND.SLNDL[2:0] bits: SSL negate delay
The SPND.SPNDL[2:0] bits: Next access delay

[Setting/Operation Example] (Motorola SPI, BFDS = 1 Case)

SPCMD0.SSLKP = 1b → Burst transfer/no interframe delay between 0 and 1 (SSL keep active)
SPCMD1.SSLKP = 1b → Burst transfer/no interframe delay between 1 and 2 (SSL keep active)
SPCMD2.SSLKP = 1b → Burst transfer/no interframe delay between 2 and 3 (SSL keep active)
SPCMD3.SSLKP = 1b → Burst transfer/no interframe delay between 3 and 4 (SSL keep active)
SPCMD4.SSLKP = 0b → do not Burst Transfer, and once in-activate SSL.

(BFDS setting is invalid because it does not Burst Transfer.)

SPCMD5.SSLKP = 1b → Burst transfer/no interframe delay between 5 and 6 (SSL keep active)
SPCMD6.SSLKP = 1b → Burst transfer/no interframe delay between 6 and 7 (SSL keep active)
SPCMD7.SSLKP = 0b → do not Burst Transfer, and once in-activate SSL.

(BFDS setting is invalid because it does not Burst Transfer.)

MODFEN bit (Mode Fault Error Detection Enable)

The MODFEN bit is used to enable or disable detection of a mode fault error. (See **7.5.3.10 Error Detection**.) The SPI determines SSL0 pin input or output direction according to the combination of the MODFEN and MSTR bits. (See **7.5.3.2 SPI Pin Control**.)

SPEIE bit (SPI Error Interrupt Enable)

The SPEIE bit is used to enable or disable an SPI error interrupt request when the SPI detects a mode fault error or an underrun error and sets the MODF flag in the SPI status register (SPSR) to 1b, when the SPI detects an overrun error and sets the OVRF flag in SPSR to 1b, or when the SPI detects a parity error and sets the PERF flag in SPSR to 1b. (See **7.5.3.10 Error Detection**.)

SPRIE bit (SPI Receive Buffer Full Interrupt Enable)

The SPRIE bit is used to enable or disable a receive buffer full interrupt request of the SPI.

SPIIE bit (SPI Idle Interrupt Enable)

The SPIIE bit is used to enable or disable an idle interrupt request of the SPI after the SPI detects the idle state and sets the IDLNF flag in the SPI status register (SPSR) to 0b.

SPDRES bit (SPI Receive Data Ready Error Select)

When a receive data ready is detected (SPSR.SPDRF = 1b), select whether to use SPRI interrupt request or SPEI interrupt request.

SPTIE bit (SPI Transmit Buffer Empty Interrupt Enable)

The SPTIE bit is used to enable or disable a transmit buffer empty interrupt request of the SPI.

A transmit buffer empty interrupt request at the beginning of transmission is generated by setting the SPE bit to 1b simultaneously when or after the SPTIE bit is set to 1b. Note that a transmit buffer empty interrupt is generated while the SPTIE bit is 1b even though SPI functions are disabled (SPE bit = 0b).

CENDIE bit (SPI Communication End Interrupt Enable)

The CENDIE bit controls generation of a communication end interrupt request.

SPMS bit (SPI Function Enable)

The SPMS bit is used to select SPI operation (4-wire) or clock synchronous operation (3-wire).

For clock synchronous operation, the SSL pin is not used but three pins RSPCK, MOSI, and MISO are used for communication. When SPMS = 1b (clock synchronous operation (3-wire)), the setting of the SPFRF bit is invalid.

To perform clock synchronous operation in master mode (SPCR.MSTR = 1b), the SPCMD.CPHA bit can be set to 0b or 1b. To perform clock synchronous operation in slave mode (SPCR.MSTR = 0b), set the CPHA bit to 1b. If this bit is set to 0b for clock synchronous operation in slave mode (SPCR.MSTR = 0b), subsequent operation is not guaranteed.

The communication status according to the settings of SPCR.MSTR, SPCR.TXMD[1:0], SPCR.SPFRF, and SPCR.SPMS as follows.

Table 7.5-5 SPI Communication Status (1/2)

SPCR MSTR	SPCR TXMD[1]	SPCR TXMD[0]	SPCR SPFRF	SPCR SPMS	Communication Status	Communication Status No.
1	0	0	0	0	Transmit-Receive Master / Motorola SPI / SPI operation (4-wire)	1-(1)
1	0	0	1	0	Transmit-Receive Master / TI-SSP / SPI operation (4-wire)	1-(2)
1	0	0	—	1	Transmit-Receive Master / Clock synchronous operation (3-wire)	1-(3)
1	0	1	0	0	Transmit only Master / Motorola SPI / SPI operation (4-wire)	1-(4)
1	0	1	1	0	Transmit only Master / TI-SSP / SPI operation (4-wire)	1-(5)

Table 7.5-5 SPI Communication Status (2/2)

SPCR MSTR	SPCR TXMD[1]	SPCR TXMD[0]	SPCR SPFRF	SPCR SPMS	Communication Status	Communication Status No.
1	0	1	—	1	Transmit only Master / Clock synchronous operation (3-wire)	1-(6)
1	1	—	0	0	Receive only Master / Motorola SPI / SPI operation (4-wire)	1-(7)
1	1	—	1	0	Receive only Master / TI-SSP / SPI operation (4-wire)	1-(8)
1	1	—	—	1	Receive only Master / Clock synchronous operation (3-wire)	1-(9)
0	0	0	0	0	Transmit-Receive Slave / Motorola SPI / SPI operation (4-wire) (default)	0-(1)
0	0	0	1	0	Transmit-Receive Slave / TI-SSP / SPI operation (4-wire)	0-(2)
0	0	0	—	1	Transmit-Receive Slave / Clock synchronous operation (3-wire)	0-(3)
0	0	1	0	0	Transmit only Slave / Motorola SPI / SPI operation (4-wire)	0-(4)
0	0	1	1	0	Transmit only Slave / TI-SSP / SPI operation (4-wire)	0-(5)
0	0	1	—	1	Transmit only Slave / Clock synchronous operation (3-wire)	0-(6)
0	1	—	0	0	Receive only Slave / Motorola SPI / SPI operation (4-wire)	0-(7)
0	1	—	1	0	Receive only Slave / TI-SSP / SPI operation (4-wire)	0-(8)
0	1	—	—	1	Receive only Slave / Clock synchronous operation (3-wire)	0-(9)

SPFRF bit (SPI Frame Format Select)

The SPFRF bit selects the communication protocol.

The format of the SPI terminal (RSPCK, SSL0 to 3) can be set according to the set communication protocol.

When SPMS = 1b (clock synchronous operation (3-wire)), this bit is invalid because SSL is not used.

TXMD[1:0] bits (Communication Mode Select)

The TXMD[1:0] bits are used to select the transmit-receive, transmit-only, and receive-only serial communication.

TXMD[1:0] = 01b

→ Transmit-only is performed without reception.
→ Receive buffer full interrupt request cannot be used.

TXMD[1] = 1b

→ Receive-only is performed without transmission.
→ Transmit buffer empty interrupt request cannot be used.

For details, see **7.5.3.6 Communications Operating Mode**.

MSTR bit (SPI Master/Slave Mode Select)

The MSTR bit is used to select master mode or slave mode of the SPI. The SPI determines input/output directions of pins RSPCK, MOSI, MISO, and SSL1 to SSL3 according to the MSTR bit setting.

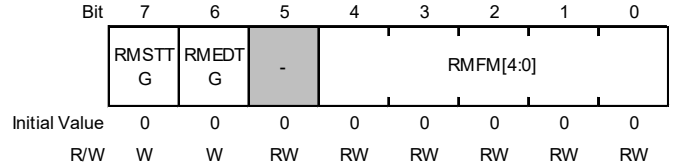
BPEN bit (Synchronization Circuit Bypass Enable)

When BPEN = 1b, bus clock (RSPI_n_PCLK) is used as operation clock (RSPI_n_TCLK) and synchronization circuit is bypassed.

7.5.2.2.7 SPI Control Register for Master Receive Only (RSPIm_SPCRRM)

The SPCRRM register controls the start and completion of communication for master mode reception only operation. If the RMFM[4:0] bits are rewritten while SPCR.SPE = 1b, subsequent operations are not guaranteed.

Access Size : 8 bits
Address : <RSPIm_base> + 000Ch
Initial Value : 00h



Bit	Bit Name	Initial Value	R/W	Description
7	RMSTTG	0h	W	1b: Receive Start (writable only when in Master Receive) Reading value is always 0b.
6	RMEDTG	0h	W	1b: Receive End (writable only when in Master Receive) Reading value is always 0b.
5	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4 to 0	RMFM[4:0]	0h	RW	Frame processing count setting in Master Receive only 00h: This function is not used.*1 01h: Automatically stop communication after processing 1 received frame. : 1Fh: Automatically stop communication after processing 31 received frames.

Note 1. See **(9) Software processing flow**.

RMFM[4:0] bits (Frame processing count setting in Master Receive only)

The number of received frames can be adjusted when operating in master receive only. Valid only when the master mode (SPCR.MSTR = 1b) and the communication operation mode select bits (SPCR.TXMD[1:0]) are 10b.

Only the start bit in master mode reception automatically stops communication after “starts frame processing according to the value set in this bit” after reception starts.

RMEDTG bit (1b: Receive End (writable only when in Master Receive))

This bit is used to end reception when master receive only. Valid only when the master mode (SPCR.MSTR = 1b) and the communication mode select bits (SPCR.TXMD[1:0]) are 10b.

RMSTTG bit (1b: Receive Start (writable only when in Master Receive))

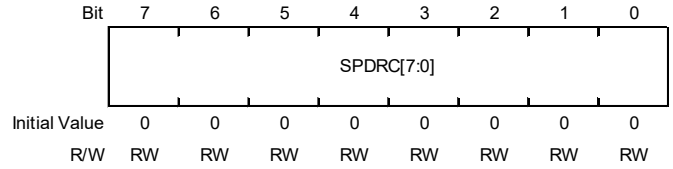
This bit is used to start reception when master receive only. Valid only when the master mode (SPCR.MSTR = 1b) and the communication mode select bits (SPCR.TXMD[1:0]) are 10b.

Writing 1b to this bit during reception is not accepted. Write again after reception is completed.

7.5.2.2.8 SPI Control Register for Received Data Ready Detection (RSPIm_SPDRCR)

The SPDRCR register is used to set the SPI receive data ready detection function. If the set value is changed while SPCR.SPE = 1b, subsequent operations are not guaranteed.

Access Size : 8 bits
Address : <RSPIm_base> + 000Dh
Initial Value : 00h



Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SPDRC[7:0]	0h	RW	SPI Receive Data Ready Detect Adjustment 00h: Disable receive data ready detection function 01h: Performs reception data ready judgment after 1 RSPI_n_TCLK : FFh: Performs reception data ready judgment after 255 RSPI_n_TCLK

SPDRC[7:0] bits (SPI Receive Data Ready Detect Adjustment)

The receive data ready detection function can be disabled or, if used, the period until detection can be set from 1 to 255 RSPI_n_TCLK.

The value set in the SPDRC[7:0] bits is used to 1b set the SPDRF flag. For details, see the description of SPDRF in **7.5.2.2.18 SPI Status Register (RSPIm_SPSR)**.

7.5.2.2.9 SPI Pin Control Register (RSPIm_SPPCR)

The SPPCR register is used to set pin mode of the SPI. If SPPCR is modified with the SPCR.SPE bit = 1b, subsequent operation is not guaranteed.

Access Size : 8 bits
Address : <RSPIm_base> + 000Eh
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	MOIFE	MOIFV	-	-	SPLP2	SPLP
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7, 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	MOIFE	0h	RW	MOSI Idle Value Fixing Enable 0b: The MOSI output value is the last data of previous transfer. 1b: The MOSI output value is the set MOIFV bit value.
4	MOIFV	0h	RW	MOSI Idle Fixed Value 0b: The fixed value of MOSI idle = 0. 1b: The fixed value of MOSI idle = 1.
3, 2	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	SPLP2	0h	RW	SPI Loopback 2 0b: Normal mode 1b: Loopback mode (data is not inverted for transmission)
0	SPLP	0h	RW	SPI Loopback 0b: Normal mode 1b: Loopback mode (data is inverted for transmission)

SPLP bit (SPI Loopback)

When the SPLP bit is set to 1b, the SPI shuts down the route between the MISO pin and the shift register (when SPCR.MSTR = 1b) or shuts down the route between the MOSI pin and the shift register, inverts the input route value in the shift register, and then connects the route to the output route (when SPCR.MSTR = 0b) (loopback mode).

SPLP2 bit (SPI Loopback 2)

When the SPLP2 bit is set to 1b, the SPI shuts down the route between the MISO pin and the shift register (when SPCR.MSTR = 1b) or shuts down the route between MOSI pin and the shift register and then connects the route to the output route without inverting the input route value in the shift register (when SPCR.MSTR = 0b) (loopback mode). If this bit is set to 1b together with the SPLP bit, setting this bit takes precedence.

MOIFV bit (MOSI Idle Fixed Value)

This bit is used to select the MOSI pin output value during the SSL negation period (including SSL retention period in burst transfer) when SPCR.MOIFE = 1b in master mode.

MOIFE bit (MOSI Idle Value Fixing Enable)

This bit is used for the SPI in master mode to fix the MOSI output value during the SSL negation period (including SSL retention period in burst transfer). When SPCR.MOIFE = 0b, the SPI outputs the last data of the previous serial transfer to MOSI during the SSL negation period. When SPCR.MOIFE = 1b, the SPI outputs the fixed MOIFV bit value to MOSI.

7.5.2.2.10 SPI Control Register 2 (RSPIm_SPCR2)

The SPCR2 register is used to set the analog delay of the master receive clock. If the value set in this register is changed while SPCR.SPE = 1b, subsequent operations are not guaranteed.

Access Size : 8 bits
Address : <RSPIm_base> + 000Fh
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	SPSCKDL[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	SPSCKDL[2:0]	0h	RW	SPI Master Receive Clock Analog Delay Valid only in master mode 000b: No delay 001b: 1.1 ns (max) 010b: 2.2 ns (max) 011b: 3.3 ns (max) 100b: 4.4 ns (max) 101b: 5.5 ns (max) 110b: 6.6 ns (max) 111b: 7.7 ns (max)

SPSCKDL[2:0] bits (SPI Master Receive Clock Analog Delay)

These bits set the analog delay for MRIOCLK input to SPI in master mode. By setting these bits, the sampling timing of the received data can be adjusted. For details, see the SPSCKSEL bit of **7.5.2.2.6 SPI Control Register (RSPIm_SPCR)**.

7.5.2.2.11 SPI Slave Select Polarity Register (RSPIm_SSLP)

The SSLP register is used to set the polarity of SSL0 to SSL3 signals of the SPI. If any of these SSLP bits is modified with the SPCR.SPE bit = 1b, subsequent operation is not guaranteed.

Access Size : 8 bits
Address : <RSPIm_base> + 0010h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	SSL3P	SSL2P	SSL1P	SSL0P
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3	SSL3P	0h	RW	SSL3 Signal Polarity Setting 0b: In the Motorola-SPI case, the SSL3 signal is active low (0). In the TI-SSP case, the SSL3 signal is active high (1). 1b: In the Motorola-SPI case, the SSL3 signal is active high (1). In the TI-SSP case, the SSL3 signal is active low (0).
2	SSL2P	0h	RW	SSL2 Signal Polarity Setting 0b: In the Motorola-SPI case, the SSL2 signal is active low (0). In the TI-SSP case, the SSL2 signal is active high (1). 1b: In the Motorola-SPI case, the SSL2 signal is active high (1). In the TI-SSP case, the SSL2 signal is active low (0).
1	SSL1P	0h	RW	SSL1 Signal Polarity Setting 0b: In the Motorola-SPI case, the SSL1 signal is active low (0). In the TI-SSP case, the SSL1 signal is active high (1). 1b: In the Motorola-SPI case, the SSL1 signal is active high (1). In the TI-SSP case, the SSL1 signal is active low (0).
0	SSL0P	0h	RW	SSL0 Signal Polarity Setting 0b: In the Motorola-SPI case, the SSL0 signal is active low (0). In the TI-SSP case, the SSL0 signal is active high (1). 1b: In the Motorola-SPI case, the SSL0 signal is active high (1). In the TI-SSP case, the SSL0 signal is active low (0).

SSLnP bits (SSLn Signal Polarity Setting)

These bits are used to specify the polarity of SSL signals. The set SSLnP bit (n = 3 to 0) values indicate the active polarity of SSLn signals.

SSL0 is different from SSL1, SSL2, and SSL3. In slave mode, it functions as an input.

For details, see **7.5.3.3.2 Master/single slave (this LSI = slave)**.

7.5.2.2.12 SPI Bit Rate Register (RSPIm_SPBR)

Access Size : 8 bits
Address : <RSPIm_base> + 0011h
Initial Value : FFh

Bit	7	6	5	4	3	2	1	0
	SPR[7:0]							
Initial Value	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SPR[7:0]	FFh	RW	The SPBR register is used to set the bit rate in master mode. If SPBR is modified while SPCR.MSTR = 1b and SPCR.SPE = 1b, subsequent operation is not guaranteed.

When the SPI is used in slave mode, the bit rate depends on the input clock bit rate regardless of the SPBR.BRDV setting. (Specify a bit rate that meets electrical characteristics.)

The bit rate is determined by a combination of the set SPBR and BRDV[1:0] in the SPCMDm register (SPCMD0 to SPCMD7).

The bit rate is calculated by the following expression, where “n” is the set SPBR value (0 to 255) and “N” is the set BRDV[1:0] bits value (0 to 3).

$$\text{Bit rate} = \frac{\text{f(RSPI_n_TCLK)}}{2 \times (n + 1) \times 2^{N}}$$

Table 7.5-6 shows an example of correspondence between bit rates and set values of SPBR and BRDV[1:0].

Table 7.5-6 Corresponding between Bit Rates and Set Values (Examples)

			Bit Rate
SPBR (n)	BRDV[1:0] Bits (N)	Division Ratio	RSPI_n_TCLK = 200 MHz
1	0	4	50.0 Mbps
2	0	6	33.4 Mbps
3	0	8	25.0 Mbps
4	0	10	20.0 Mbps
5	0	12	16.66 Mbps
5	1	24	8.34 Mbps
5	2	48	4.16 Mbps
5	3	96	2.08 Mbps
255	3	4096	48.8 kbps

7.5.2.2.13 SPI Sequence Control Register (RSPIm_SPSCR)

The SPSCR register is used to set the sequence length for the SPI to perform master operation.

Before modifying the SPSLN[2:0] bits in SPSCR while SPCR.MSTR = 1b and SPCR.SPE = 1b, confirm that SPSR.IDLNF = 0b.

Access Size : 8 bits
Address : <RSPIm_base> + 0013h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	SPSLN[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	SPSLN[2:0]	0h	RW	SPI Sequence Length Specification The order in which the SPCMD0 to SPCMD07 registers are to be referenced is changed in accordance with the sequence length that is set in these bits. The relationship among the setting of these bits, the sequence length, and the SPCMD0 to SPCMD7 register numbers referenced by SPI is as follows. However, the SPI in slave mode always refers to SPCMD0. 000b: Sequence Length = 1 (0→0→···) 001b: Sequence Length = 2 (0→1→0→···) 010b: Sequence Length = 3 (0→1→2→0→···) 011b: Sequence Length = 4 (0→1→2→3→0→···) 100b: Sequence Length = 5 (0→1→2→3→4→0→···) 101b: Sequence Length = 6 (0→1→2→3→4→5→0→···) 110b: Sequence Length = 7 (0→1→2→3→4→5→6→0→···) 111b: Sequence Length = 8 (0→1→2→3→4→5→6→7→0→···)

Note: The read access size is fixed at 32 bits.

SPSLN[2:0] bits (SPI Sequence Length Specification)

These bits are used to set the sequence length for the SPI in master mode to perform sequence operation.

According to the sequence length specified by SPSLN[2:0] bits, the SPI in master mode changes the SPCMDm registers (SPCMD0 to SPCMD7) to be referenced and the reference sequence. For details, see **7.5.3.14.1(3) Sequence control**.

The SPI in slave mode always references SPCMD0.

7.5.2.2.14 SPI Command Register n (RSPIm_SPCMDn) (n = 0 to 7)

The SPI has eight SPI command registers (SPCMD0 to SPCMD7) that are used to set the transfer format of the SPI in master mode. Furthermore, some bits in SPCMD0 are used to set the transfer format of the SPI in slave mode.

The SPI in master mode sequentially references SPCMD0 to SPCMD7 according to the setting of SPSCR.SPSLN[2:0], and then performs serial transfer specified in the referenced SPCMD.

SPCMD should be set while the transmit buffer is empty (next-transfer data is not set) and data which refers to SPCMD has not been written.

The SPCMD referenced by the SPI in master mode is indicated by SPCP[2:0] in the SPSSR register. If SPCMD0 is modified while the SPI in slave mode is enabled (SPCR.SPE = 1b), subsequent operation is not guaranteed.

Access Size : 8, 16, 32 bits

Address : <RSPIm_base> + 0014h + n x 0004h

Initial Value : 0007_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	SSLA[1:0]		-	-	-	SPB[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCKDEN	SLNDEN	SPNDEN	LSBF	-	-	-	-	SSLKP	-	-	-	BRDV[1:0]		CPOL	CPHA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
25, 24	SSLA[1:0]	0h	RW	SSL Signal Assertion 00b: SSL0 01b: SSL1 10b: SSL2 11b: SSL3
23 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20 to 16	SPB[4:0]	7h	RW	SPI Data Length 00h: Setting prohibited 01h: Setting prohibited 02h: Setting prohibited 03h: 4 bits 04h: 5 bits 05h: 6 bits : 1Eh: 31 bits 1Fh: 32 bits
15	SCKDEN	0h	RW	RSPCK Delay Setting Enable 0b: RSPCK delay is 1 RSPCK. [In the Motorola-SPI case] RSPCK delay is 0 RSPCK. [In the TI-SSP case] 1b: RSPCK delay is the set value of the RSPCK delay register (SPCKD).
14	SLNDEN	0h	RW	SSL Negation Delay Setting Enable 0b: [Master] SSL negation delay is 1 RSPCK. [Slave in the TI-SSP] SSL negation delay is 1 RSPI_n_TCLK + 2~3 RSPI_n_TCLK (Synchronization delay) 1b: SSL negation delay is the set value of the slave select negation delay register (SSLND).

Bit	Bit Name	Initial Value	R/W	Description
13	SPNDEN	0h	RW	SPI Next-Access Delay Enable 0b: Next-access delay is 1 RSPCK + 5 RSPI_n_TCLK 1b: Next-access delay is the set value of the SPI next-access delay register (SPND).
12	LSBF	0h	RW	SPI LSB First 0b: MSB first 1b: LSB first
11 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7	SSLKP	0h	RW	SSL Signal Level Hold 0b: All SSL signals are negated at the end of transfer. 1b: SSL signal level is held after the transfer ends until the next access starts.
6 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3, 2	BRDV[1:0]	0h	RW	Bit Rate Division 00b: The base bit rate is selected. 01b: Two-divided base bit rate is selected. 10b: Four-divided base bit rate is selected. 11b: Eight-divided base bit rate is selected.
1	CPOL	0h	RW	RSPCK Polarity 0b: RSPCK in idle state is 0. 1b: RSPCK in idle state is 1.
0	CPHA	0h	RW	RSPCK Phase This register is valid only for Motorola-SPI cases. (In the TI-SSP case, it is fixed at 1). 0b: Data is sampled at an odd edge and changes at an even edge. 1b: Data changes at an odd edge and is sampled at an even edge.

Note: The read access size is fixed at 32 bits.

CPHA bit (RSPCK Phase)

This bit is used to set the RSPCK phase of the SPI in master mode or slave mode. To perform data communication between SPI modules, the same RSPCK phase must be set for both modules.

When SPCR.SPMS = 0b and SPCR.SPFRF = 1b (in TI SSP mode), setting CPHA = 0b is invalid.

This register is valid only for Motorola-SPI cases.

CPOL bit (RSPCK Polarity)

This bit is used to set the RSPCK polarity of the SPI in master mode or slave mode. To perform data communication between SPI modules, the same RSPCK polarity must be set for both modules.

BRDV[1:0] bits (Bit Rate Division)

This register is used to determine the bit rate with a combination of the set values of the BRDV[1:0] bits and the SPBR register. The set SPBR value determines the base bit rate. The set BRDV[1:0] bits value is used to select undivided, 2-divided, 4-divided, or 8-divided base bit rate. SPCMD0 to SPCMD7 enable setting of different BRDV[1:0] values. This makes it possible to perform serial transfer with a different bit rate for each command.

SSLKP bit (SSL Signal Level Hold)

This bit is used to set whether to hold or negate the SSL signal level of the current command during a period from the SSL negation timing for the current command to the SSL assertion timing for the next command when the SPI in master mode performs serial transfer. Setting this bit to 1b enables burst transfer in SPI operation master mode. For details, see **(4) Burst transfer** in **7.5.3.13.1 Master mode operation**.

To use the SPI in slave mode, set SSLKP bit to 0b.

LSBF bit (SPI LSB First)

This bit is used to set the data format of the SPI in master mode or slave mode to MSB first or LSB first.

SPNDEN bit (SPI Next-Access Delay Enable)

This bit is used to set the period from SSL negation to SSL assertion in SPI master mode. When SPNDEN = 0b, the SPI sets the next-access delay to 1 RSPCK + 5 RSPI_n_TCLK. When SPNDEN = 1b, the SPI inserts the next-access delay in accordance with the SPND register setting.

To use the SPI in slave mode, set SPNDEN bit to 0b.

SLNDEN bit (SSL Negation Delay Setting Enable)

[In the Motorola-SPI case]

This bit is used to set the period from RSPCK oscillation stop to SSL negation in SPI master mode. When SLNDEN = 0b, the SPI sets the SSL negation delay to 1 RSPCK. When SLNDEN = 1b, the SPI negates the SSL signal with the RSPCK delay in accordance with the SSLND register setting.
To use the SPI in slave mode, set SLNDEN bit to 0b.

[In the TI-SSP case]

This bit is used to set the period from RSPCK oscillation stop to output disable in SPI master mode, or last edge of RSPCK to output disable in slave mode. When the SLNDEN bit is 0b, the SSL negate delay is 1 RSPCK in master mode and 1 RSPI_n_TCLK + 2 to 3 RSPI_n_TCLK (Synchronization delay) in slave mode. When SLNDEN = 1b, the SPI negates the SSL signal with the RSPCK delay in accordance with the SSLND register setting.
When using SPI in slave mode except TI SSP setting, set the SLNDEN bit to 0b.

SCKDEN bit (RSPCK Delay Setting Enable)

[In the Motorola-SPI case]

This bit is used to set the period (RSPCK delay) after the SPI in master mode activates the SSL signal until it oscillates RSPCK. When SCKDEN = 0b, the SPI sets the RSPCK delay to 1 RSPCK. When SCKDEN = 1b, the SPI starts RSPCK oscillation with the RSPCK delay in accordance with the SPCKD register setting.
To use the SPI in slave mode, set SCKDEN bit to 0b.

[In the TI-SSP case]

This bit is used to set the period from the start of assertion of the SSL signal to the RSPCK oscillation (RSPCK delay) and the period of the SSL signal to negation by the SPI in master mode. When SCKDEN = 0b, the SPI does not set the RSPCK delay. When SCKDEN = 1b, the SPI starts RSPCK oscillation with the RSPCK delay in accordance with the SPCKD register setting.
To use the SPI in slave mode, set SCKDEN bit to 0b.

SPB[4:0] bits (SPI Data Length)

These bits are used to set the transfer data length of the SPI in master mode or slave mode.

SSLA[1:0] bits (SSL Signal Assertion)

These bits are used to control SSL signal assertion for the SPI in master mode to perform serial transfer. The set SSLA[1:0] bits value controls assertion of the SSL3 to SSL0 signals. The signal polarity when the SSL signal is asserted depends on the set value of the SSLP register.

To use the SPI in slave mode, set SSLA[1:0] bits to 0b.

7.5.2.2.15 SPI Data Control Register (RSPIm_SPDCR)

The SPDCR register controls the data format.

If the value set in this register is changed while SPCR.SPE = 1b, subsequent operations are not guaranteed.

Access Size : 8, 16 bits
Address : <RSPIm_base> + 0040h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	SPFC[3:0]				-	-	-	SINV	SPRDTD	-	-	BYSW
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
11 to 8	SPFC[3:0]	0h	RW	Frame Count Number of Frames Specification 0000b: 1 frame 0001b: 2 frames 0010b: 3 frames 0011b: 4 frames : 1110b: 15 frames 1111b: 16 frames
7 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	SINV	0h	RW	Serial data invert 0b: Not invert serial data 1b: Invert serial data
3	SPRDTD	0h	RW	SPI Receive Data or Transmit Data Selection 0b: The SPDR reads the receive buffer 1b: The SPDR reads the transmit buffer
2, 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	BYSW	0h	RW	Byte Swap Operating Mode Select 0b: Byte swap is disabled 1b: Byte swap is enabled

BYSW bit (Byte Swap Operating Mode Select)

It is a setting bit, that is to swap a transmit/receive data in byte units. A data after byte swap is different by a data length (setting of SPCMD.SPB[4:0]).

When byte swap, A data length (setting of SPB[4:0]) must be set to 32 bits or 16 bits. Other case of data length (for example, 4 to 15, 17 to 31 bit length), byte swap is not guaranteed. For the arrangement of data before and after swapping data lengths of 32 bits and 16 bits, see **7.5.3.4 Data Format** and **7.5.3.4.5 Byte swap reception**.

When the parity function set to valid, the behavior is not guaranteed.

SPRDTD bit (SPI Receive Data or Transmit Data Selection)

The SPRDTD bit is used to select receive buffer or transmit buffer from which the SPI data register (SPDR) value is read.

When the transmit buffer is read, the value written to SPDR the last time is read.

SINV bit (Serial data invert)

The SINV bit is used to invert transmit data and receive data.

When the SINV bit is set to 1b, transmit buffer (SPTX) data is inverted to invert transmit data and receive data, and then the inverted data is stored in the receive buffer (SPRX). The parity bit is the value corresponding to the inverted transmission/ reception data.

SPFC[1:0] bits (Frame Count)

The SPFC[1:0] bits are used for the condition to set the CENDF flag in slave receive only mode. For details on the CENDF flag setting conditions, see **7.5.2.2.18 SPI Status Register (RSPIm_SPSR)**.

The SPFC[1:0] bits are invalid except in the slave receive only mode.

7.5.2.2.16 SPI Data Control Register 2 (RSPIm_SPDCR2)

The SPDCR2 register controls the FIFO threshold. If the value set in this register is changed while SPCR.SPE = 1b, subsequent operations are not guaranteed.

Access Size : 8, 16 bits
Address : <RSPIm_base> + 0044h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	TTRG[3:0]				-	-	-	-	RTRG[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
11 to 8	TTRG[3:0]	0h	RW	Transmission FIFO threshold setting When using DMA, set the threshold to 0 for RTRG and TTRG bits. 0000b: Threshold 0 0001b: Threshold 1 0010b: Threshold 2 0011b: Threshold 3 : 1111b: Threshold 15
7 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3 to 0	RTRG[3:0]	0h	RW	Receive FIFO threshold setting When using DMA, set the threshold to 0 for RTRG and TTRG bits. 0000b: Threshold 0 0001b: Threshold 1 0010b: Threshold 2 0011b: Threshold 3 : 1111b: Threshold 15

Note: The read access size is fixed at 32 bits.

RTRG[3:0] bits (Receive FIFO threshold setting)

Set the receive FIFO threshold.

When the number of data stored in the receive FIFO > the number of frames set by RTRG[3:0], the receive buffer full flag is set.

TTRG[3:0] bits (Transmission FIFO threshold setting)

Set the transmit FIFO threshold.

When the number of empty stages in the transmit FIFO > the number of frames set in TTRG[3:0], the transmit buffer empty flag is set.

7.5.2.2.17 SPI Sequence Status Register (RSPIm_SPSSR)

The SPSSR register indicates the sequence control status when the SPI performs master operation. Write access to SPSSR is ignored.

Access Size : 8 bits
Address : <RSPIm_base> + 0051h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	SPECM[2:0]			-	SPCP[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	-	0h	R	Reserved Whenever it is read, 0b is read.
6 to 4	SPECM[2:0]	0h	R	SPI Error Command 000b: SPCMD0 001b: SPCMD1 010b: SPCMD2 011b: SPCMD3 100b: SPCMD4 101b: SPCMD5 110b: SPCMD6 111b: SPCMD7
3	-	0h	R	Reserved Whenever it is read, 0b is read.
2 to 0	SPCP[2:0]	0h	R	SPI Command Pointer 000b: SPCMD0 001b: SPCMD1 010b: SPCMD2 011b: SPCMD3 100b: SPCMD4 101b: SPCMD5 110b: SPCMD6 111b: SPCMD7

SPCP[2:0] bits (SPI Command Pointer)

These bits show the SPCMDm registers 0 to 7 (SPCMD0 to SPCMD7) indicated by the current pointer in the SPI sequence control. For details about the SPI sequence control, see **7.5.3.13.1 Master mode operation**.

SPECM[2:0] bits (SPI Error Command)

These bits show the SPCMDm registers 0 to 7 (SPCMD0 to SPCMD7) indicated by the command pointer (SPCP[2:0] bits) when an error was detected in the SPI sequence control. The SPI updates the SPECM[2:0] bits value only when an error is detected. When no error is present (SPSR.OVRF = 0b, SPSR.MODF = 0b, SPSR.PERF = 0b), the SPECM[2:0] bits value has no meaning. For the SPI's error detection function, see **7.5.3.10 Error Detection**. For the SPI's sequence control, see **7.5.3.13.1 Master mode operation**.

7.5.2.2.18 SPI Status Register (RSPIm_SPSR)

Access Size : 8, 16 bits
Address : <RSPIm_base> + 0052h
Initial Value : 2000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPRF	CENDF	SPTEF	UDRF	PERF	MODF	IDLNF	OVRF	SPDRF	-	-	-	-	-	-	-
Initial Value	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	SPRF	0h	R	SPI Receive Buffer Full Flag 0b: The number of data stored in the receive FIFO ≤ number of frames set by the SPDCR2.RTRG bit. 1b: The number of data stored in the receive FIFO > number of frames set by the SPDCR2.RTRG bit.
14	CENDF	0h	R	Communication End Flag 0b: The SPI is not communicating or communicating. 1b: The SPI communication completed.
13	SPTEF	1h	R	SPI Transmit Buffer Empty Flag 0b: The number of empty stages in the transmit FIFO ≤ the value set in SPDCR2.TTRG 1b: The number of empty stages in the transmit FIFO > the value set in SPDCR2.TTRG
12	UDRF	0h	R	Underrun Error Flag This bit indicates error status in combination with the MODF flag. The UDRF bit is valid when MODF flag is 1. 0b: Mode fault error occurred (MODF = 1) 1b: Underrun error occurred (MODF = 1)
11	PERF	0h	R	Parity Error Flag 0b: No parity error is present. 1b: A parity error is present.
10	MODF	0h	R	Mode Fault Error Flag 0b: Neither mode fault error nor underrun error is present. 1b: A mode fault error or underrun error is present.
9	IDLNF	0h	R	SPI Idle Flag 0b: The SPI is in the idle state. 1b: The SPI is in the transfer state.
8	OVRF	0h	R	Overrun Error Flag 0b: No overrun error is present. 1b: An overrun error is present.
7	SPDRF	0h	R	SPI Receive Data Ready Flag 0b: Receive data ready not detected 1b: Receive data ready detected
6 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read.

SPDRF bit (SPI Receive Data Ready Flag)

During communication (SPCR.SPE = 1b), a certain period of time has elapsed while the number of data stored in the reception FIFO ≤ the reception FIFO threshold.

This bit is set to 0b when the reception operation is not performed (SPCR.TXMD[1:0] = 01b).

[Setting (to 1b) condition]

All the following two conditions are met:

- SPDRCR.SPDRC[7:0] ≠ 00h
- After the receive FIFO has been written, when the number of data stored in the receive FIFO ≤ the receive FIFO threshold and the value set by SPDRC[7:0] has elapsed

[Clearing (to 0b) condition]

- When 1b is written to the SPSRC.SPDRFC bit

OVRF bit (Overrun Error Flag)

This flag indicates whether an overrun error is present. When the RSPCK clock auto-stop function is enabled (SPCR.SCKASE bit = 1b) in master mode (SPCR.MSTR bit = 1b), no overrun error occurs and, therefore, this flag is not set to 1b. For details, see **7.5.3.10.1 Overrun error**.

[Setting (to 1b) condition]

When serial transfer is completed in one of the following two conditions with data stored in the receive FIFO for the number of FIFO stages.

- The SPCR.TXMD[1:0] bits = 00b. (transmit-receive mode)
- The SPCR.TXMD[1:0] bits = 10b. (receive only)

[Clearing (to 0b) condition]

- When 1b is written to the SPSRC.OVRFC bit

IDLNF bit (SPI Idle Flag)

This flag indicates transfer status of the SPI.

[Setting (to 1b) condition]

[Transmit-Receive, Transmit-only Master mode]

- None of the [Transmit-Receive, Transmit-only Master mode] in [Clearing (to 0b) conditions]

[Receive-only Master mode]

- When 1b is written to SPCRRM.RMSTTG bit

[Slave mode]

- SPCR.SPE = 1b (SPI function enabled)

[Clearing (to 0b) conditions]

Communication status: 1-(1) to (6) For details of communication status, see **Table 7.5-5**.

[Transmit-Receive, Transmit-only Master mode]

Any of the following two conditions is met:

- SPCR.SPE = 0b (SPI initialization)
- All the following three conditions are met:
  - The next transfer data is not set in the transmission buffer (SPTXn, n = 0 to 15)
  - SPSSR.SPCP[2:0] = 000b (at the beginning of sequence control)
  - The operation completed until the next access delay (the master main state machine has transitioned to the idle state)

[Receive-only Master mode]

Communication status: 1-(7) to (9)

Any of the following two conditions is met:

- SPCR.SPE = 0b (SPI initialization)
- Any of the following three conditions is met:
  - When RMFM[4:0] = 00h, after writing 1b to RMEDTG, operation completed until the next access delay (the master main state machine has transitioned to the idle state)

- When RMFM[4:0] ≠ 00h, after writing 1b to RMEDTG, operation completed until the next access delay (the master main state machine has transitioned to the idle state)
- When RMFM[4:0] ≠ 00h, the operation is completed up to the next access delay after processing is completed for the number of received frames set in RMFM[4:0] (the master main state machine has transitioned to the idle state)

[Slave mode]

Communication status: 0-(1) to (9)

- SPCR.SPE = 0b (SPI initialization)

MODF bit (Mode Fault Error Flag)

This flag indicates whether a mode fault error or an underrun error is present. The UDRF flag allows you to see which error (mode fault error or underrun error) has occurred.

[Setting (to 1b) condition]

[Slave, Motorola-SPI mode]

Any of the following two conditions is met:

- The SSL0 pin is negated before the RSPCK cycles necessary for data transfer end while the SPCR.MSTR bit = 0b (slave mode), SPCR.SPFRF bit = 0b (Motorola-SPI) and the SPCR.MODFEN bit = 1b (mode fault error detection enabled), and then the SPI has detected a mode fault error.
- Serial transfer is started before transmit data output becomes ready while the SPCR.SPE bit = 1b (SPI function enabled), and then the SPI has detected an underrun error.

[Slave, TI-SSP mode]

Any of the following two conditions is met:

- The SSL0 pin is asserted before the RSPCK cycles necessary for data transfer end while the SPCR.MSTR bit = 0b (slave mode), SPCR.SPFRF bit = 1b (TI-SSP) and the SPCR.MODFEN bit = 1b (mode fault error detection enabled), and then the SPI has detected a mode fault error.
- Serial transfer is started before transmit data output becomes ready while the SPCR.SPE bit = 1b (SPI function enabled), and then the SPI has detected an underrun error.

The SSL signal active level depends on the SSLP.SSLiP bits (SSL signal polarity bits).

[Clearing (to 0b) condition]

- When 1b is written to the SPSRC.MODFC bit

PERF bit (Parity Error Flag)

This flag indicates whether a parity error is present.

[Setting (to 1b) condition]

When the serial transfer ends and a parity error is detected with the SPCR.SPPE bit set to 1b under any of the following two conditions:

- The SPCR.TXMD[1:0] bits = 00b (transmit-receive)
- The SPCR.TXMD[1:0] bits = 10b (receive-only)

[Clearing (to 0b) condition]

- When 1 is written to the SPSRC.PERFC bit

UDRF bit (Underrun Error Flag)

This flag indicates that a mode fault error or an underrun error is present.

[Setting (to 1) condition]

- Serial transfer is started before transmit data output becomes ready while the SPCR.MSTR bit = 0b (slave mode) and the SPCR.TXMD[1:0] bits = 00b or 01b (transmit-receive or transmit-only) and the SPCR.SPE bit = 1b (SPI function enabled), and then the SPI has detected an underrun error.

[Clearing (to 0b) condition]

- When 1b is written to the SPSRC.UDRFC bit

SPTEF bit (SPI Transmit Buffer Empty Flag)

This flag indicates the transmit buffer (SPTX) status in the SPI data register (SPDR).

[Setting (to 1) condition]

Any of the following three conditions is met:

- The SPCR.SPE bit is set to 0b (SPI initialization)
- When the number of empty transmission FIFO stages > SPDCR2.TTRG[3:0]
- When 1b is written to SPFCR.SPFRST

[Clearing (to 0b) condition]

Any of the following two conditions is met:

- At the time of final access when transmission data is written to SPDR (SPTXn, n = 0 to 15) in one processing routine using DMAC
- When 1b is written to the SPSRC.SPTEFC bit

Writing a value to the SPDR register is enabled only while the SPTEF flag = 1b. If a value is written to the SPDR register while the SPTEF flag = 0b, transmit buffer data is not updated.

CENDF bit (Communication End Flag)

This flag indicates communication end status of SPI. It turns 1 at communication end and turns 0 at starting next communication.

[Setting (to 1b) condition]

- Transmit-Receive/Transmit-only Master mode

  Communication status: 1-(1) to (6). For details of communication status, see **Table 7.5-5**.
  The following three conditions are met:

  - The next transfer data is not set in the transmission buffer (SPTXn, n = 0 to 15)
  - The SPSSR.SPCP[2:0] are 000b (it means the head of the sequential control)
  - Operation completed until the next access delay (the master main state machine has transitioned to the idle state)

- Receive-only Master mode

  Communication status: 1-(7) to (9)
  Any of the following three conditions is met:

  - When RMFM[4:0] = 00h, after writing 1b to RMEDTG, operation completed until the next access delay (the master main state machine has transitioned to the idle state)

- When RMFM[4:0] ≠ 00h, after writing 1b to RMEDTG, operation completed until the next access delay (the master main state machine has transitioned to the idle state)
- When RMFM[4:0] ≠ 00h, the operation is completed up to the next access delay after processing is completed for the number of received frames set in RMFM[4:0] (the master main state machine has transitioned to the idle state)

- Transmit-receive/transmit-only slave, Motorola-SPI mode at SPI serial communication (4-wire: the SPCR.SPMS bit is 0b).

  Communication status: 0-(1), (4)
  The following three conditions are met:
  - The next transfer data is not set in the transmission buffer
  - The transmission shift register is empty (it means SPI does not do serial transfer)
  - SSL0 was negated

- Transmit-receive/transmit-only slave, TI-SSP mode at SPI serial communication (4-wire: the SPCR.SPMS bit is 0b).

  Communication status: 0-(2), (5)
  The following three conditions are met:
  - The next transfer data is not set in the transmission buffer
  - The transmission shift register is empty (it means SPI does not do serial transfer)
  - When the SSL negate delay is completed

- Transmit-receive/transmit only slave mode at clock synchronous (3-wire: the SPCR.SPMS bit is 1b)

  Communication status: 0-(3), (6)
  The following three conditions are met:
  - The next transfer data is not set in the transmission buffer
  - The transmission shift register is empty (it means SPI does not do serial transfer)
  - The last even edge of RSPCK of the frame was detected (when the SPCMD.CPHA bit is 1b)

- Receive only slave, Motorola-SPI mode at SPI serial communication (4-wire: the SPCR.SPMS bit is 0b).

  Communication status: 0-(7)
  The following condition is met:
  - SSL0 input was negated after getting frames for SPDCR.SPFC set value in the receive buffer

- Receive only slave, TI-SSP mode at SPI serial communication (4-wire: the SPCR.SPMS bit is 0b).

  Communication status: 0-(8)
  The following condition is met:
  - SSL0 negate delay is completed after getting frames for SPDCR.SPFC set value in the receive buffer

- Receive only slave mode at clock synchronous (3-wire: the SPCR.SPMS bit is 1b).

  Communication status: 0-(9)
  The following condition is met:
  - The last even edge of RSPCK of the last frame received for SPFC set value was detected (when the SPCMD.CPHA bit is 1b)

[Clearing (to 0b) condition]

- Transmit-Receive/transmit-only Master mode

  Communication status: 1-(1) to (6)
  Any of the following two conditions is met:

  - The next transmit data was written to the transmit buffer (SPTX)
  - When 1b is written to the SPSRC.CENDFC bit

- Receive-only Master mode

  Communication status: 1-(7) to (9)
  Any of the following two conditions is met:

  - When 1b is written to the SPCRRM.RMSTTG bit with SPCR.SPE = 1b
  - When 1b is written to the SPSRC.CENDFC bit

- Transmit-receive/transmit only slave mode

  Communication status: 0-(1) to (6)
  Satisfy one of following conditions:

  - The next transmit data is written to the transmit buffer (SPTX)
  - When 1b is written to the SPSRC.CENDFC bit

- Receive-only slave mode at SPI serial communication (4-wire: the SPCR.SPMS bit is 0b).

  Communication status: 0-(7) to (8)
  Satisfy one of following conditions:

  - SSL0 assertion of next data is detected
  - When 1b is written to the SPSRC.CENDFC bit

- Receive-only slave mode at clock synchronous (3-wire: the SPCR.SPMS bit is 1b).

  Communication status: 0-(9)
  Satisfy one of following conditions:

  - The first edge of RSPCK of the next data is detected
  - When 1b is written to the SPSRC.CENDFC bit

SPRF bit (SPI Receive Buffer Full Flag)

This flag indicates the receive buffer (SPRX) status in the SPDR register.

[Setting (to 1b) condition]

When the number of data stored in the receive FIFO > the number of frames set in the SPDCR2.RTRG[3:0] bits in Transmit-Receive, Receive-only modes. However, the SPRF flag does not change from 0b to 1b while the OVRF flag = 1b. (See **7.5.3.10 Error Detection**.)

[Clearing (to 0b) condition]

Any of the following three conditions is met:

- At the last access when read data is read from SPDR (SPRXn, n = 0 to 15) in one processing routine using DMAC
- When 1b is written to the SPSRC.SPRFC bit
- When 1b is written to the SPFCR.SPFRST bit

7.5.2.2.19 SPI Transfer FIFO Status Register (RSPIm_SPTFSR)

Access Size : 8 bits
Address : <RSPIm_base> + 0058h
Initial Value : 10h

Bit	7	6	5	4	3	2	1	0
	-	-	-	TFDN[4:0]				
Initial Value	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read.
4 to 0	TFDN[4:0]	10h	R	Transmit FIFO data empty stage number 0_0000b: Number of empty stages 0 0_0001b: Number of empty stages 1 0_0010b: Number of empty stages 2 0_0011b: Number of empty stages 3 0_0100b: Number of empty stages 4 : 1_0000b: Number of empty stages 16

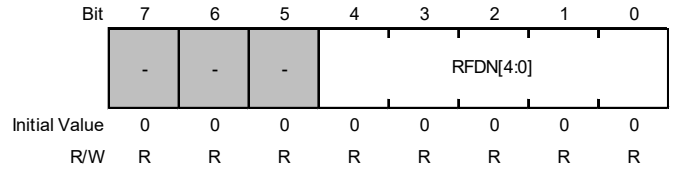
Note: TFDN[4:0] is initialized by SPCR.SPE = 0b.

7.5.2.2.20 SPI Receive FIFO Status Register (RSPIm_SPRFSR)

Access Size : 8 bits

Address : <RSPIm_base> + 005Ch

Initial Value : 00h



Bit	Bit Name	Initial Value	R/W	Description
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read.
4 to 0	RFDN[4:0]	0h	R	Receive FIFO data store stage number 0_0000b: Number of store stages 0 0_0001b: Number of store stages 1 0_0010b: Number of store stages 2 0_0011b: Number of store stages 3 0_0100b: Number of store stages 4 : 1_0000b: Number of empty stages 16

Note: RFDN[4:0] is initialized by SPCR.SPE = 0b.

7.5.2.2.21 SPI Polling Register (RSPIm_SPPSR)

Access Size : 8, 16, 32 bits

Address : <RSPIm_base> + 0060h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SPEPS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read.
0	SPEPS	0h	R	SPI Polling Status 0b: SPCR.SPE is 0 1b: SPCR SPE is 1

Note: This bit shows a result of SPCR.SPE after synchronization by the operation clock (RSPI_n_TCLK).

Note: The read access size is fixed at 32 bits.

7.5.2.2.22 SPI Status Clear Register (RSPIm_SPSRC)

The SPSRC register clears the status flag (SPSR) that indicates the operating status of SPI.

Access Size : 8, 16 bits

Address : <RSPIm_base> + 006Ah

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPRFC	CENDFC	SPTEFC	UDRFC	PERFC	MODFC	-	OVRFC	SPDRFC	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	R	W	W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	SPRFC	0h	W	SPI Receive Buffer Full Flag Clear By writing 1b, the SPI Receive Buffer Full Flag can be cleared. The read value is always 0b.
14	CENDFC	0h	W	Communication End Flag Clear By writing 1b, the Communication End Flag can be cleared. The read value is always 0b.
13	SPTEFC	0h	W	SPI Transmit Buffer Empty Flag Clear By writing 1b, the SPI Transmit Buffer Empty Flag can be cleared. The read value is always 0b.
12	UDRFC	0h	W	Underrun Error Flag Clear By writing 1b, the Underrun Error Flag can be cleared. The read value is always 0b. When clearing the UDRF flag, clear the MODF flag at the same time (MODFC = 1).
11	PERFC	0h	W	Parity Error Flag Clear By writing 1b, the Parity Error Flag can be cleared. The read value is always 0b.
10	MODFC	0h	W	Mode Fault Error Flag Clear By writing 1b, the Mode Fault Error Flag can be cleared. The read value is always 0b. Before setting MODFC and UDRFC, make sure that SPSR.MODF and UDRF are set to 1b.
9	-	0h	R	Reserved Whenever it is read, 0b is read.
8	OVRFC	0h	W	Overrun Error Flag Clear By writing 1b, the Overrun Error Flag can be cleared. The read value is always 0b.
7	SPDRFC	0h	W	SPI Receive Data Ready Flag Clear The SPI receive data ready flag can be cleared by writing 1b. When read, 0b is read.
6 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read.

7.5.2.2.23 SPI FIFO Clear Register (RSPIm_SPFCR)

The SPFCR register is used to clear the FIFO.

If SPFCR is rewritten while SPCR.SPE = 1b, subsequent operations are not guaranteed.

Access Size : 8 bits
Address : <RSPIm_base> + 006Ch
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	SPFRST
Initial Value	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	-	All 0	W	Reserved The written value should always be 0b.
0	SPFRST	0h	W	SPI FIFO clear By writing 1b, the pointer in the FIFO and the stored data are initialized. Reading value is always 0b.

7.5.3 Operation

In this section, a word “serial transfer period” is used as a word that means a period from the start of valid data drive to latching of the final valid data.

7.5.3.1 Overview of SPI Operations

The SPI can transfer data in the following five modes.

- Slave mode (SPI operation)
- Master mode (SPI operation)
- Slave mode (clock synchronous operation)
- Master mode (clock synchronous operation)

SPI modes can be set by the MSTR, MODFEN, SPMS and SPFRF bits in the SPCR register.

Table 7.5-7 lists the relationship between SPI modes and SPCR settings, and outline of each mode.

Table 7.5-7 Relationship between SPI Modes and SPCR Settings and Description of Each Mode (1/2)

Mode	Slave (SPI Operation)	Master (SPI Operation)	Slave (Clock Synchronous Operation)	Master (Clock Synchronous Operation)
MSTR bit setting	0	1	0	1
MODFEN bit setting	0 or 1	0	0	0
SPMS bit setting	0	0	1	1
SPFRF bit setting	valid	valid	In-valid	In-valid
RSPCK signal	Input	Output	Input	Output
MOSI signal	Input	Output	Input	Output
MISO signal	Output/Hi-Z	Input	Output	Input
SSL0 signal	Input	Output	Hi-Z (not used)	Hi-Z (not used)
SSL1 to SSL3 signals	Hi-Z (not used)	Output	Hi-Z (not used)	Hi-Z (not used)
SSL polarity modification function	Supported	Supported	—	—
Transfer rate	Up to RSPI_n_TCLK/2	Up to RSPI_n_TCLK/2	Up to RSPI_n_TCLK/2	Up to RSPI_n_TCLK/2
Clock source	RSPCK input	On-chip baud rate generator	RSPCK input	On-chip baud rate generator
Clock polarity	2 types			
Clock phase	2 types*1	2 types*5	One (CPHA = 1b)	2 types
First transfer bit	MSB/LSB			
Transfer data length	4 to 32 bits			
Burst transfer	Enabled (CPHA = 1b)	Enabled (CPHA = 0b, 1b)	—	—
RSPCK delay control	Not supported	Supported	Not supported	Supported
SSL negation delay control	Not supported*6	Supported	Not supported	Supported
Next-access delay control	Not supported	Supported	Not supported	Supported
Transfer activation method	SSL input active or RSPCK oscillation	Transmit buffer empty interrupt request or transmit buffer write when SPTEF = 1b	RSPCK oscillation	Transmit buffer empty interrupt request or transmit buffer write when SPTEF = 1b
Sequence control	Not supported	Supported	Not supported	Supported

Table 7.5-7 Relationship between SPI Modes and SPCR Settings and Description of Each Mode (2/2)

Mode	Slave (SPI Operation)	Master (SPI Operation)	Slave (Clock Synchronous Operation)	Master (Clock Synchronous Operation)
Transmission buffer empty detection	Supported*4	Supported	Supported*4	Supported
Reception buffer full detection	Supported*1			
Overrun error detection	Supported*1	Supported*1,*3	Supported*1	Supported*1,*3
Parity error detection	Supported*1,*2			
Mode fault error detection	Supported (MODFEN = 1b)	Not supported	Not supported	Not supported
Underrun error detection	Supported*4	Not supported	Supported*4	Not supported

Note 1. When SPI is transmit-master mode or transmit-slave mode (see **Table 7.5-5**), none of receive buffer full error, overrun error, and parity error is detected.

Note 2. When the SPCR.SPPE bit is 0b, parity error is not detected.

Note 3. When the SPCR.SCKASE bit is 1b, overrun error is not detected.

Note 4. When SPI is receive only slave mode (see **Table 7.5-5**), none of transmit buffer empty and underrun error is detected.

Note 5. CPHA = 0b is invalid in TI SSP mode. (Even if it is set, the operation is the same as when CPHA = 1b.)

Note 6. Available only in TI SSP mode.

7.5.3.2 SPI Pin Control

The SPI automatically switches pin directions and output modes according to the settings of the SPCR.MSTR, SPCR.MODFEN and SPCR.SPMS bits.

Table 7.5-8 shows the relationship between pin state and set values of each bit.

For details of OE function, see **7.5.3.5 Transfer Format (Frame Format)**.

Table 7.5-8 Relationship between Pin States and Set Control Bit Values

Mode	Pin	Pin State*1
Master mode (SPI operation) (MSTR = 1b, MODFEN = 0b, SPMS = 0b)	RSPCK	Output
	SSL0 to SSL3	Output
	MOSI	Output
	MISO	Input
Slave mode (SPI operation) (MSTR = 0b, SPMS = 0b)	RSPCK	Input
	SSL0	Input
	SSL1 to SSL3*3	Not used
	MOSI	Input
	MISO*2	Output/Hi-Z
Master mode (Clock synchronous operation) (MSTR = 1b, MODFEN = 0b, SPMS = 1b)	RSPCK	Output
	SSL0 to SSL3*3	Not used
	MOSI	Output
	MISO	Input
Slave mode (Clock synchronous operation) (MSTR = 0b, SPMS = 1b)	RSPCK	Input
	SSL0 to SSL3*3	Not used
	MOSI	Input
	MISO	Output

Note 1. The set SPI value is not applied to multi-function pins for which the SPI function is not selected.

Note 2. Motorola-SPI: When SSL0 is inactive level or when SPCR.SPE = 0b, the pin state becomes Hi-Z.
TI-SSP: When SSL0 is except the communication period or when SPCR.SPE = 0b (assertion after SPE = 1b and communication completed), the pin status changes to Hi-Z.

Note 3. These pins are available for use as I/O port pins.

The SPI in master mode (SPI operation) determines the MOSI signal value in the SSL negation period (including the SSL hold period in burst transfer) according to the settings of the SPPCR.MOIFE and SPPCR.MOIFV bits, as listed in **Table 7.5-9**.

Table 7.5-9 Determining the MOSI Signal Value in the SSL Negation Period

MOIFE Bit	MOIFV Bit	MOSI Signal Value during SSL Negation Period
0	0, 1	Final data from previous transfer
1	0	Always low
1	1	Always high

7.5.3.3 SPI System Configuration Examples

This configuration example describes that 0 level of SSL signals is active level. When connecting and using in a multi-slave mode, the transfer format of the connected device should be unified to either Motorola-SPI or TISSP.

7.5.3.3.1 Master/single slave (this LSI = master)

Figure 7.5-7 shows an example of master/single slave SPI system configuration where this LSI is used as a master. In the master/single slave configuration, the SSL0 to SSL3 output signals of this LSI (master) are not used.

SSL input signals of the SPI slave are fixed to 0 level to always select the SPI slave. When SPCMD.CPHA = 0b, some slave devices cannot fix SSL signals to active level in the relevant transfer format. If the SSL signal level cannot be fixed, connect the SSL output of this LSI to the SSL input of the slave device.

This LSI (master) always drives the RSPCK and MOSI pins. The SPI slave always drives the MISO pin.

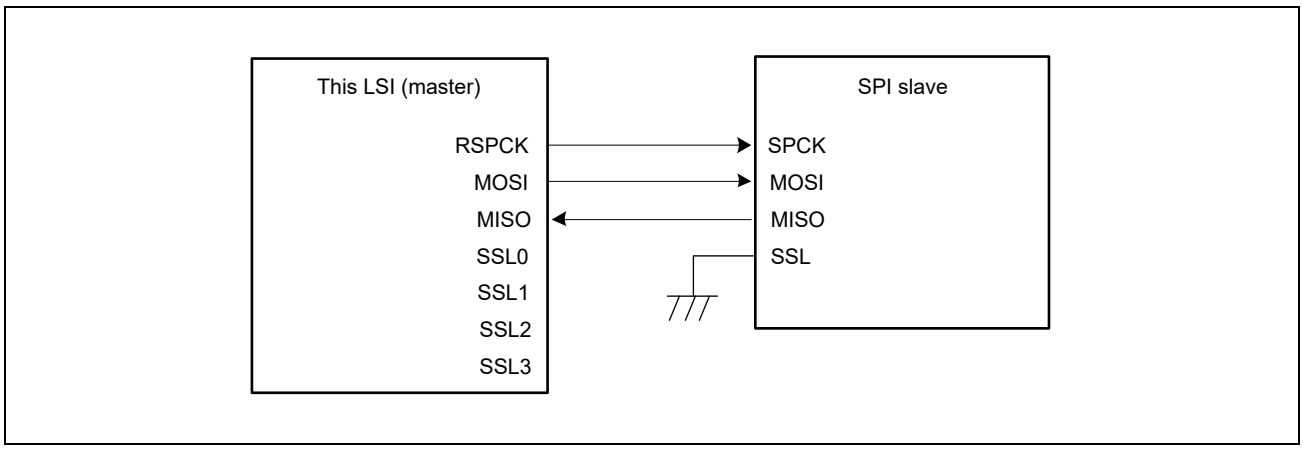


Figure 7.5-7 Master/Single-Slave Configuration Example (This LSI = Master)

7.5.3.3.2 Master/single slave (this LSI = slave)

Figure 7.5-8 shows an example of master/single slave SPI system configuration where this LSI is used as a slave. When this LSI is used as a slave, the SSL0 pin is used as SSL input. The SPI master always drives the RSPCK and MOSI pins. This LSI (slave) always drives the MISO pin. When the SSL0 level is inactive, the pin state becomes Hi-Z.

In the single slave configuration with the SPCMD.CPHA bit set to 1b, set the SPCR.SPFRF bit to 0b, and set the SPMS bit to 0b. There is the SSL0 input level of this LSI (slave) is fixed to 0 so that this LSI (slave) can be always selected and serial transfer can also be performed (**Figure 7.5-9**). However, the communication end interrupt does not output when SSL0 input was fixed as **Figure 7.5-9**.

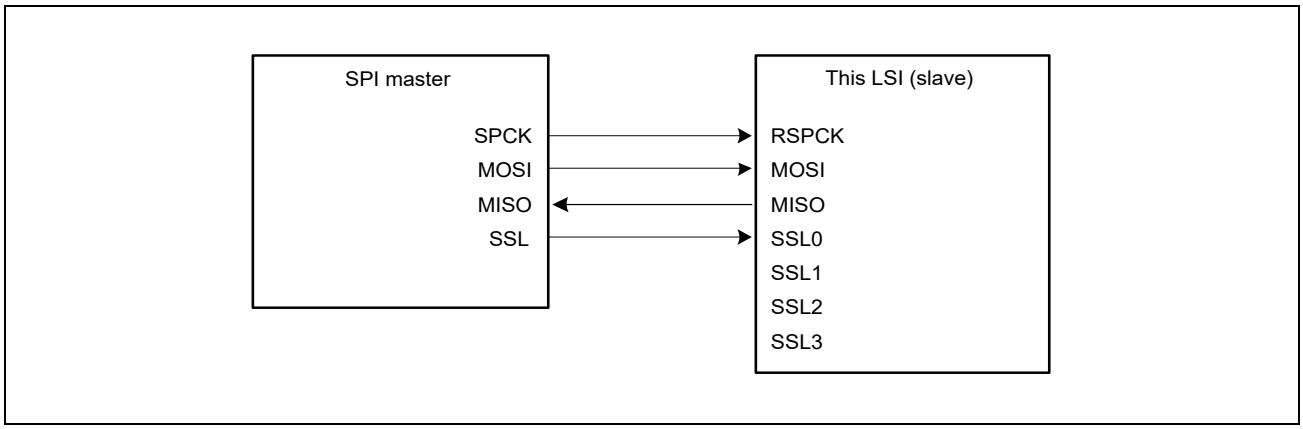


Figure 7.5-8 Master/Single-Slave Configuration Example (This LSI = Slave, CPHA = 0b)

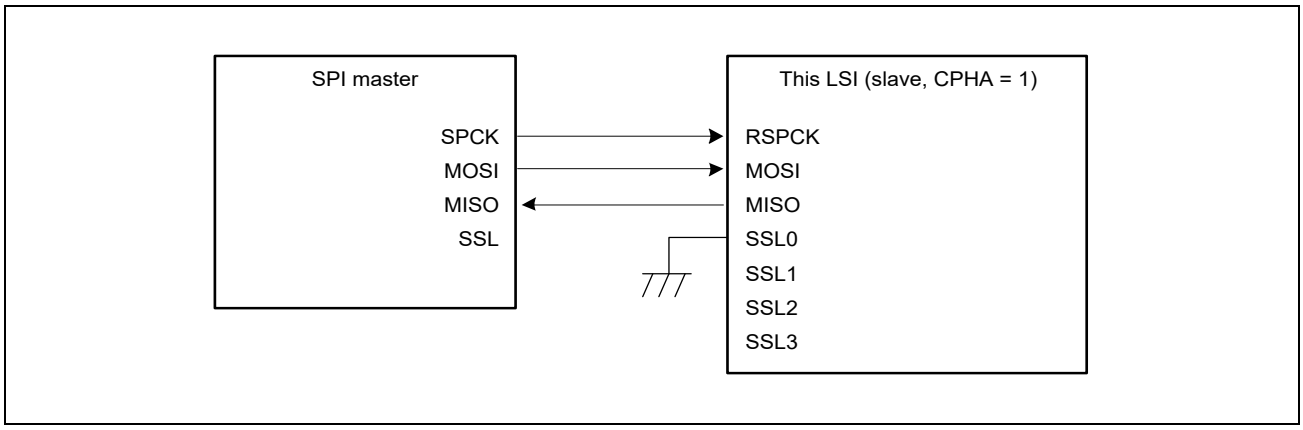


Figure 7.5-9 Master/Single-Slave Configuration Example (This LSI = Slave, CPHA = 1b)

7.5.3.3.3 Master/multi-slave (this LSI = master)

Figure 7.5-10 shows an example of master/multi-slave SPI system configuration where this LSI is used as a master. In the example in **Figure 7.5-10**, the SPI system is comprised of this LSI (master) and four slaves (SPI slave 0 to SPI slave 3).

The RSPCK output and MOSI output pins of the master are connected to the RSPCK input and MOSI input pins of SPI slave 0 to SPI slave 3. All of the MISO output pins of SPI slave 0 to SPI slave 3 are connected to the MISO input pin of the master. The SSL0 to SSL3 output pins of this LSI (master) are connected to the SSL input pin of SPI slave 0 to SPI slave 3.

The master always drives RSPCK, MOSI, and SSL0 to SSL3 pins. The slave (out of SPI slave 0 to SPI slave 3) where 0 level is input to SSL input pin drives the MISO pin.

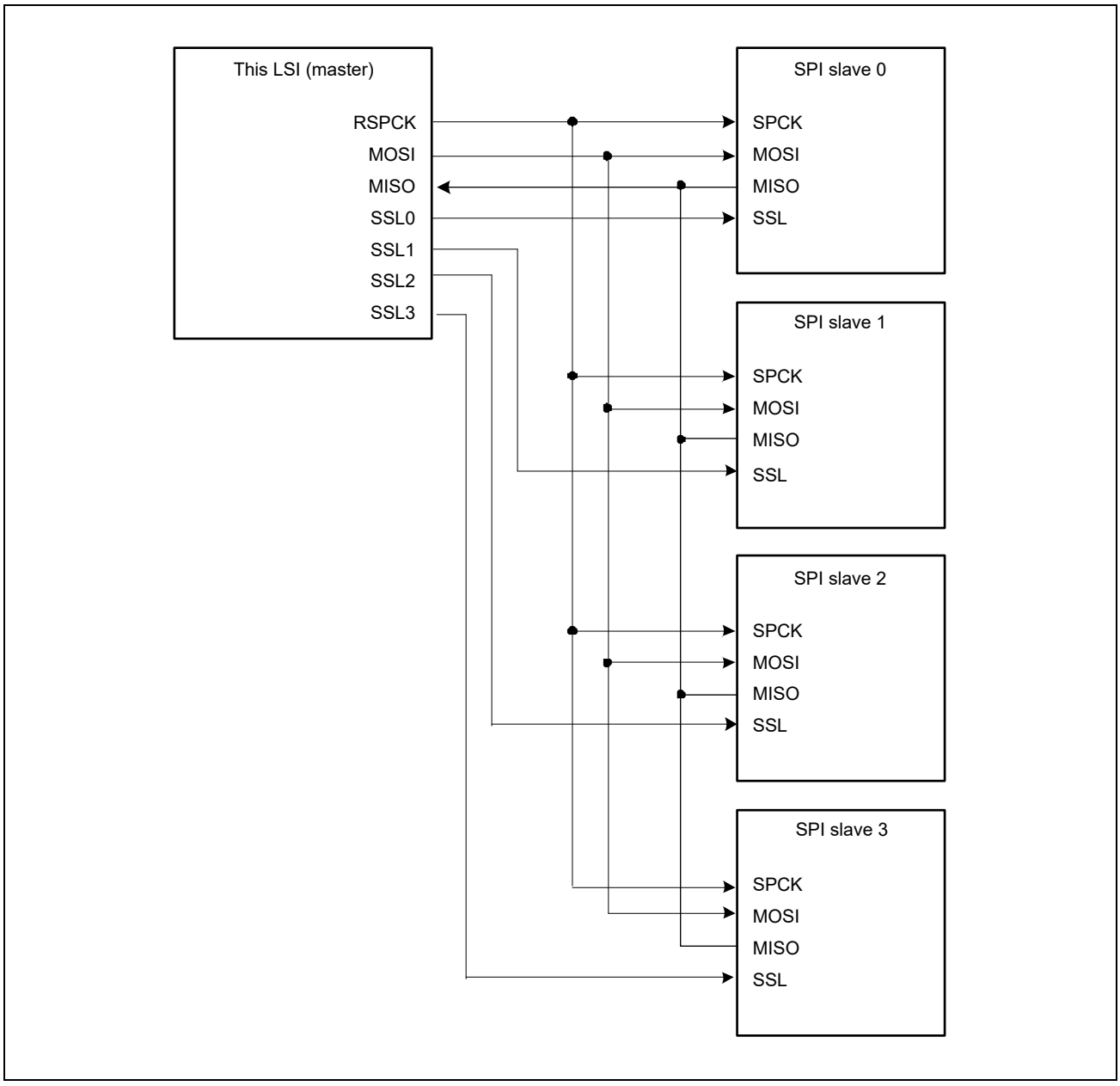


Figure 7.5-10 Master/Multi-Slave Configuration Example (This LSI = Master)

7.5.3.3.4 Master/multi-slave (with this LSI acting as slave)

Figure 7.5-11 shows an example of master/multi-slave SPI system configuration where this LSI is used as a slave. In the example in **Figure 7.5-11**, the SPI system consists of the SPI master and two these LSIs (slave X, slave Y).

The RSPCK output and MOSI output pins of the SPI master are connected to the RSPCK input and MOSI input pins of the LSIs (slave X, slave Y). The MISO output pin of the LSIs (slave X, slave Y) is connected to the MISO input pin of the SPI master. The SSLX output and SSLY output pins of the SPI master are connected to the SSL0 input pin of the LSIs (slave X, slave Y).

The SPI master always drives the RSPCK, MOSI, SSLX, and SSLY pins. The slave X or slave Y (this LSI) where 0 level is input to the SSL0 input pin drives the MISO pin.

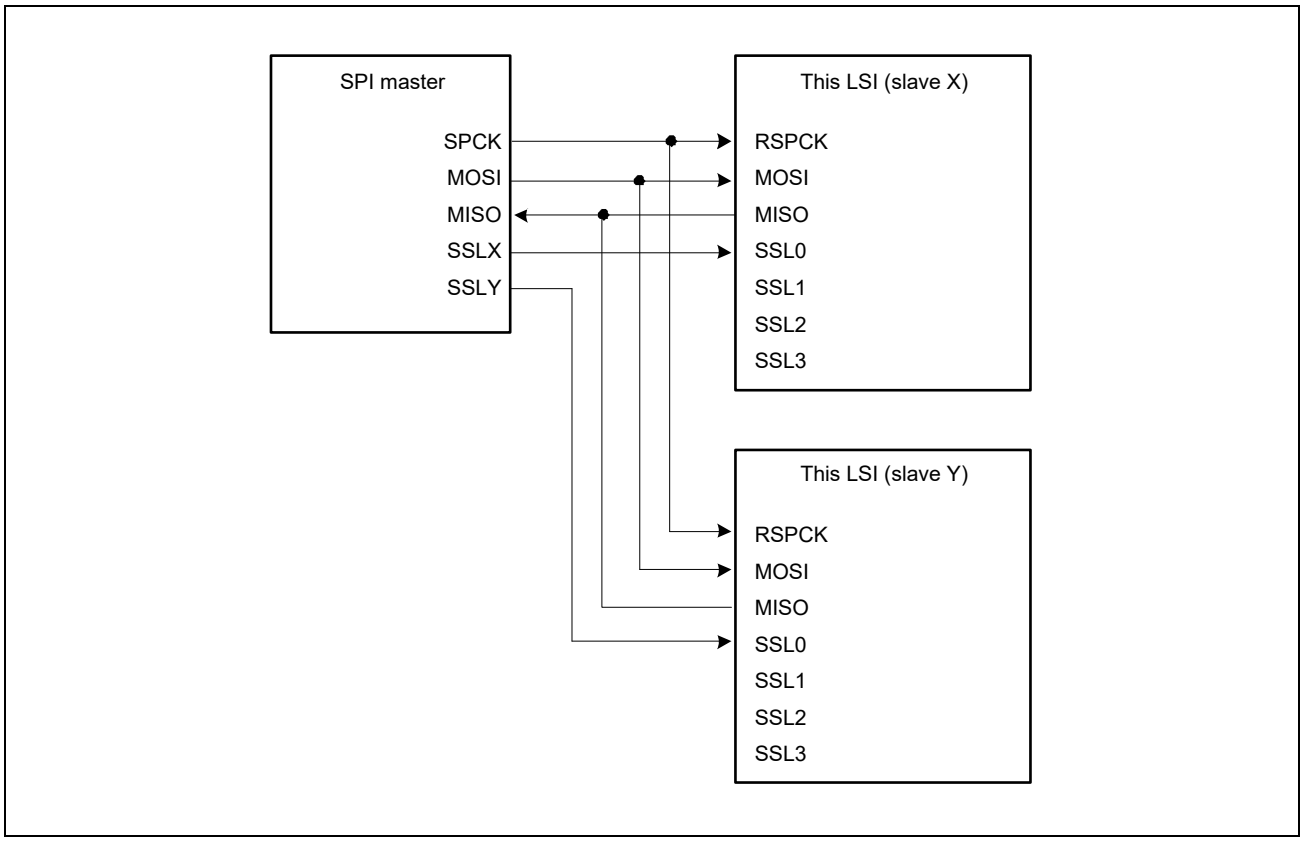


Figure 7.5-11 Master/Multi-Slave Configuration Example (This LSI = Slave)

7.5.3.3.5 Master (clock synchronous operation)/slave (clock synchronous operation) (this LSI = master)

Figure 7.5-12 shows an example of master (clock synchronous operation)/slave (clock synchronous operation) SPI system configuration where this LSI is used as a master. In this configuration, the SSL0 to SSL3 output pins of this LSI (master) are not used.

This LSI (master) always drives the RSPCK and MOSI pins. The SPI slave always drives the MISO pin.

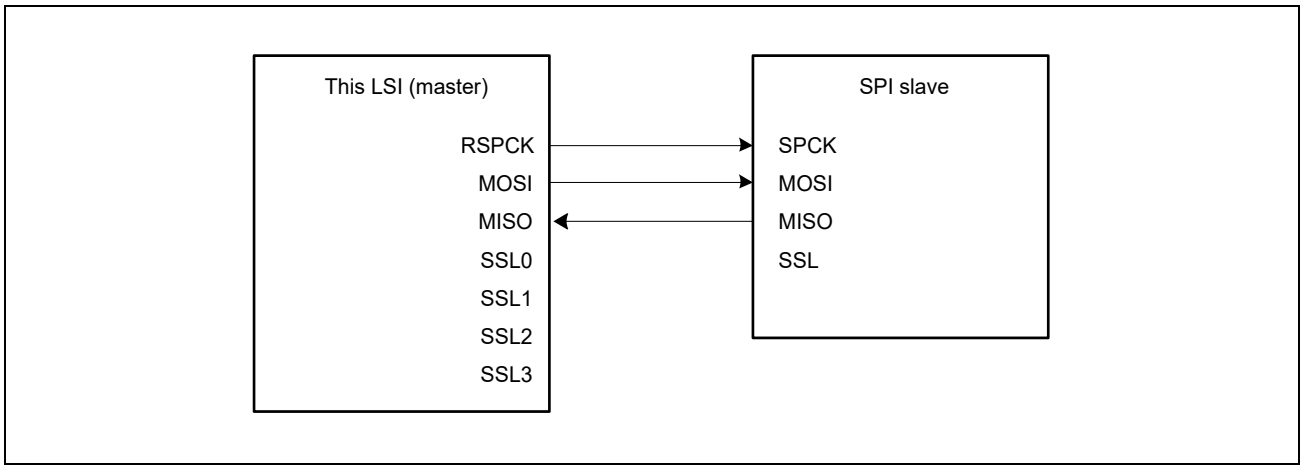


Figure 7.5-12 Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) Configuration Example (This LSI = Master)

7.5.3.3.6 Master (clock synchronous operation)/slave (clock synchronous operation) (this LSI = slave)

Figure 7.5-13 shows an example of master (clock synchronous operation)/slave (clock synchronous operation) SPI system configuration where this LSI is used as a slave. When this LSI is used as a slave (clock synchronous operation), this LSI (slave) always drives the MISO pin and the SPI master always drives the RSPCK and MOSI pins.

Only in the single slave configuration with the SPCMD.CPHA bit set to 1b, this LSI (slave) can perform serial transfer.

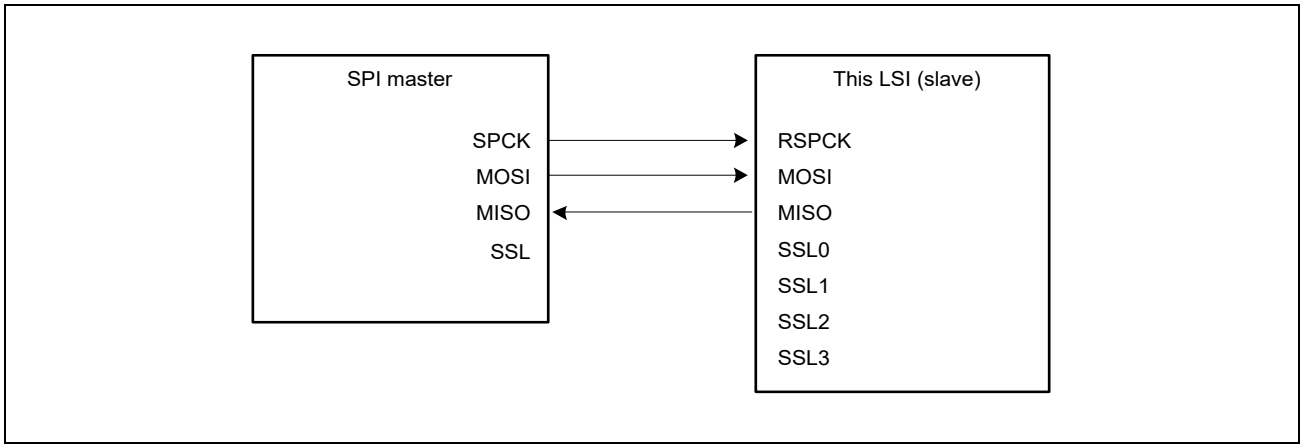


Figure 7.5-13 Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) Configuration Example (This LSI = Slave, CPHA = 1b)

7.5.3.4 Data Format

Data format of the SPI depends on the set values of the SPCMD register and the SPCR.SPPE bit. The SPI handles data of the set data length from the LSB in the SPDR register as transfer data regardless of MSB first or LSB first.

7.5.3.4.1 Data format of one frame

The data format of one frame for transmission and reception is shown below.

(1) With parity disabled (SPPE = 0b)

When the parity function is disabled, data of the bit length specified by SPCMDm.SPB[4:0] is transmitted and received.

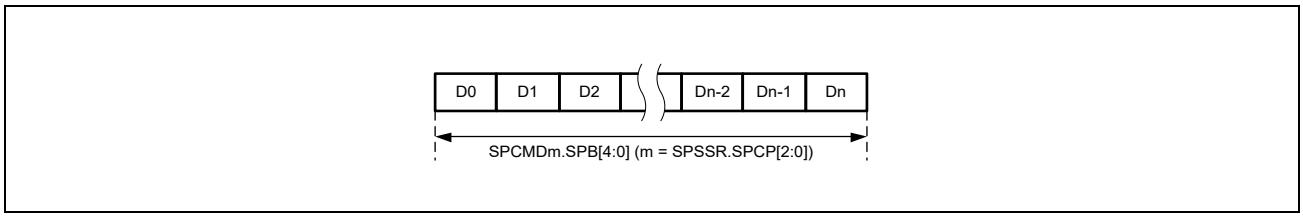


Figure 7.5-14 Outline of Data Format (when Parity Function is Disabled)

(2) With parity enabled (SPPE = 1b)

When the parity function is enabled, data of the bit length specified by SPCMDm.SPB[4:0] is transmitted and received. However, the final bit is used as a parity bit.

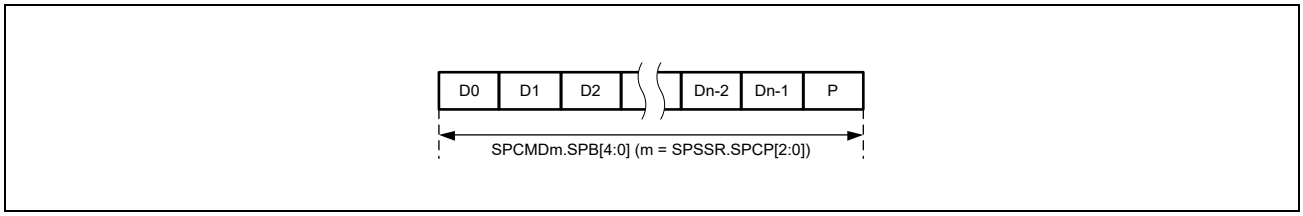


Figure 7.5-15 Outline of Data Format (when Parity Function is Enabled)

7.5.3.4.2 When parity is disabled (SPPE = 0b)

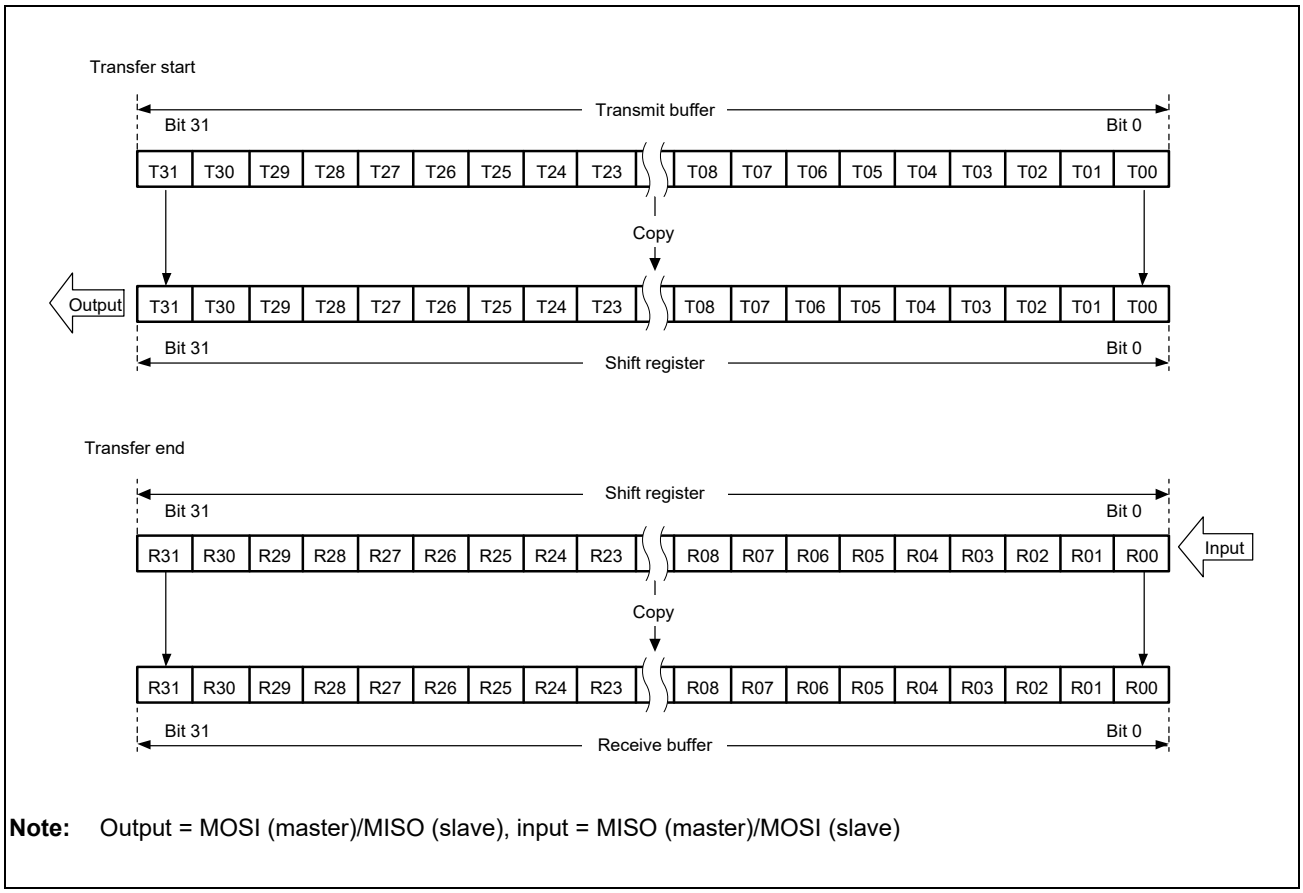
When the parity function is disabled, transmit buffer data is copied to the shift register without processing it. The following describes the relationship between SPDR, and shift register by using a combination of MSB first/LSB first and bit length.

(1) MSB first transfer (32-bit data)

Figure 7.5-16 shows operations of the SPDR register and the shift register when the SPI performs 32-bit MSB-first transfer with the parity function disabled.

In transmission, T31 to T00 in the transmit buffer are copied to the shift register. Bit values in the shift register are shifted and transmitted in the order of T31 → T30 → … → T00 as transmit data.

In reception, received data is stored in bit 0 of the shift register, and received data is shifted in units of data. When necessary RSPCK cycles are input and data is stored from R31 to R00, the shift register value is copied to the receive buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

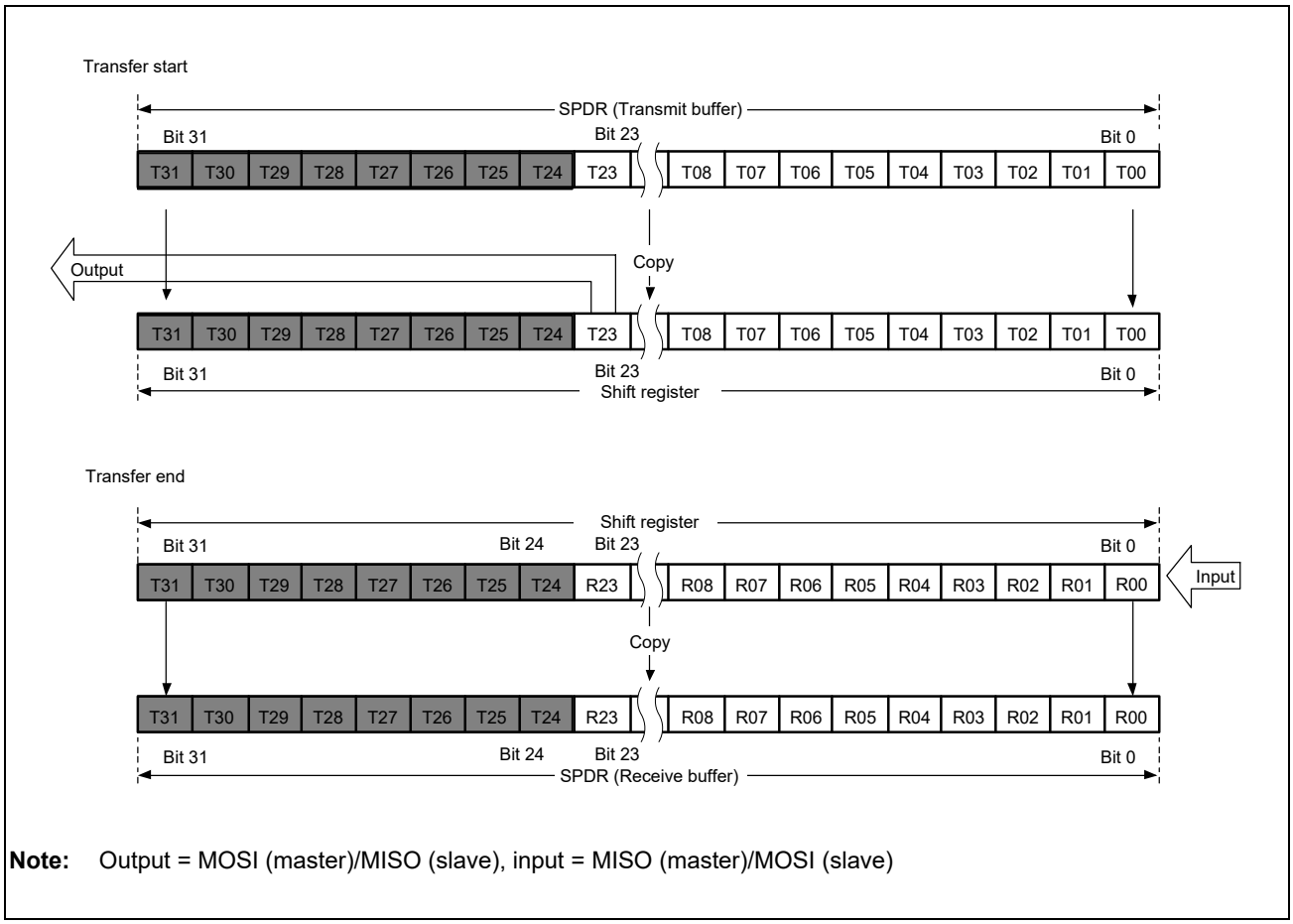
Figure 7.5-16 MSB First Transfer (32-Bit Data, Parity Disabled)

(2) MSB first transfer (24-bit data)

As an example of MSB-first transfer of data other than 32 bits with the parity function disabled, **Figure 7.5-17** shows operations of the SPDR register and the shift register when the SPI performs 24-bit data transfer.

In transmission, lower 24 bits (T23 to T00) in the transmit buffer are copied to the shift register. Bit values in the shift register are shifted and transmitted in the order of T23 → T22 → … → T00 as transmit data.

In reception, received data is stored in bit 0 of the shift register, and received data is shifted in units of data. When necessary RSPCK cycles are input and data is stored from R23 to R00, the shift register value is copied to the receive buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 7.5-17 MSB First Transfer (24-Bit Data, Parity Disabled)

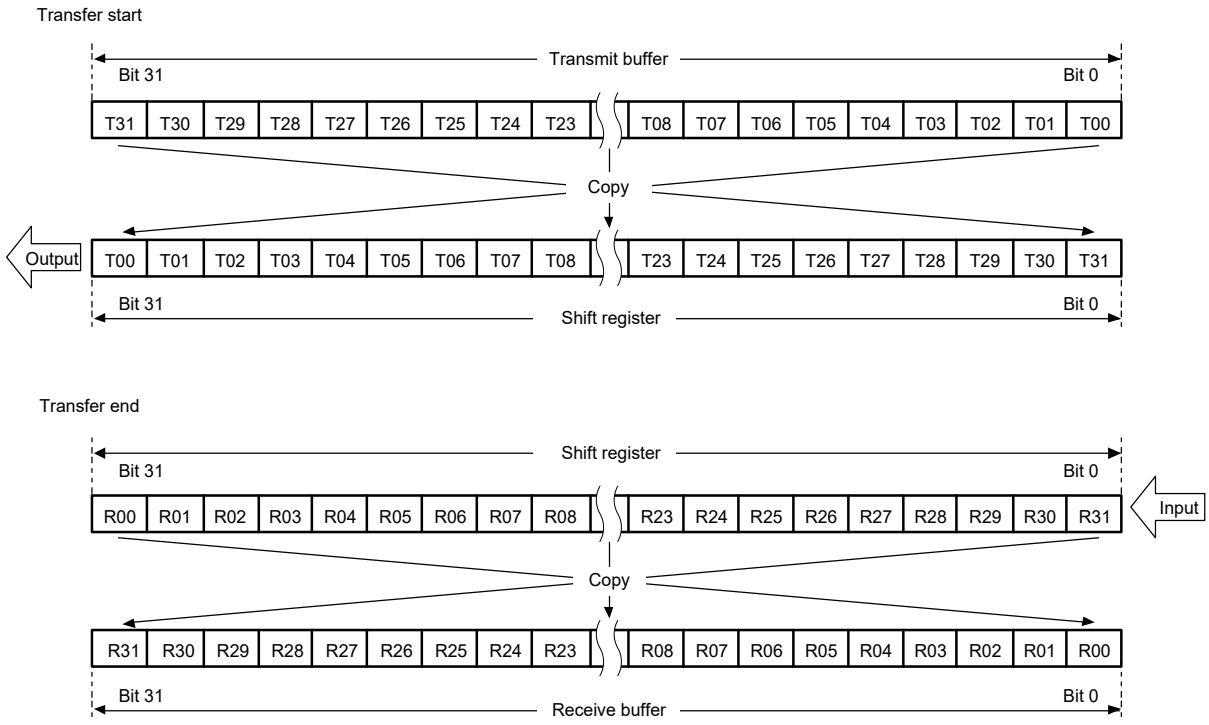
(3) LSB first transfer (32-bit data)

Figure 7.5-18 shows operations of the SPDR register and the shift register when the SPI performs 32-bit LSB-first transfer with the parity function disabled.

In transmission, bit values of the transmit buffer (T31 to T00) are reversed in bit units and are copied to the shift register in the order of T00 to T31. Bit values in the shift register are shifted and transmitted in the order of T00 → T01 → … → T31 as transmit data.

In reception, the first received data is stored in bit 0 of the shift register, and received data is shifted in units of data.

When necessary RSPCK cycles are input and data is stored from R00 to R31, bit values in the shift register are reversed in bit units and are copied to the receive buffer in the order of R31 to R00.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

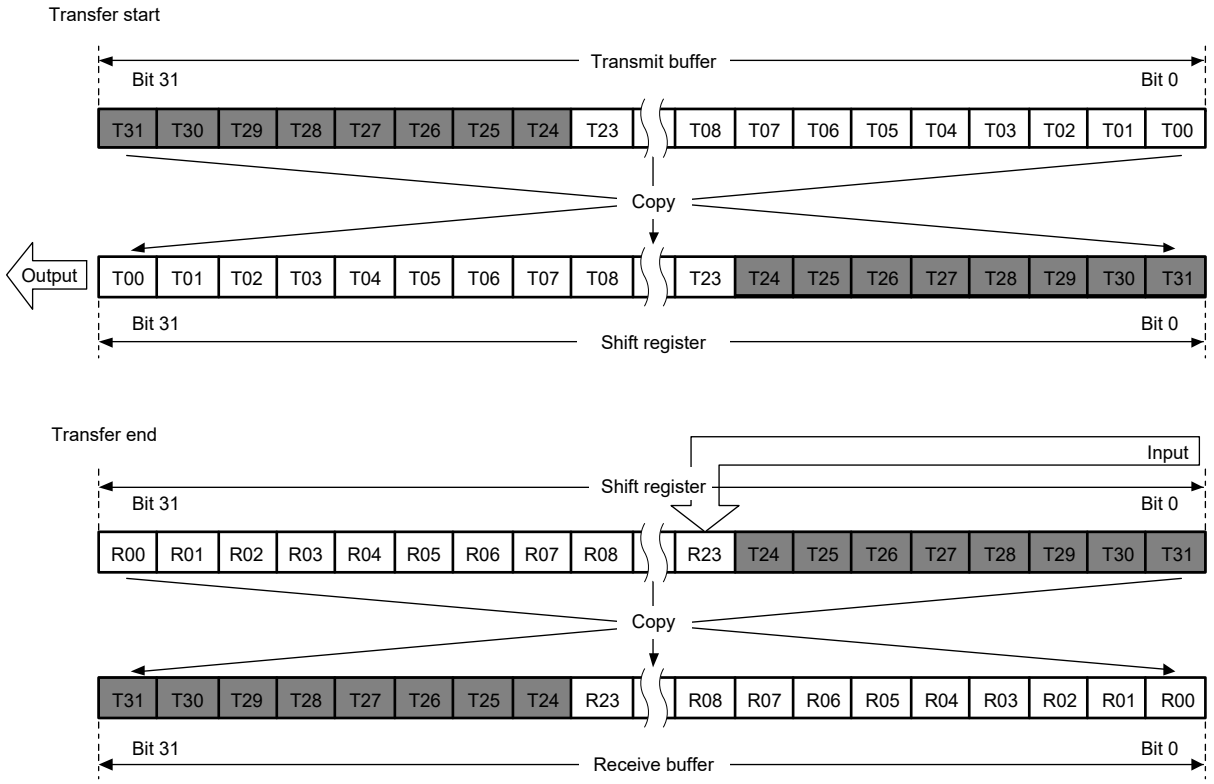
Figure 7.5-18 LSB First Transfer (32-Bit Data, Parity Disabled)

(4) LSB first transfer (24-bit data)

As an example of LSB-first transfer of data other than 32 bits with the parity function disabled, **Figure 7.5-19** shows operations of the SPDR register and the shift register when the SPI performs 24-bit data transfer.

In transmission, lower 24-bit values of the transmit buffer (T23 to T00) are reversed to the order from T00 to T23 in bit units and are copied to the shift register. Bit values in the shift register are shifted and transmitted in the order of T00 → T01 → … → T23 as transmit data.

In reception, received data is stored in bit 8 of the shift register, and received data is shifted in units of data. When necessary RSPCK cycles are input and data is stored from R00 to R23, bit values in the shift register are reversed in bit units and are copied to the receive buffer in the order of R23 (bit 23) to R00 (bit 0).



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 7.5-19 LSB First Transfer (24-Bit Data, Parity Disabled)

7.5.3.4.3 When parity is enabled (SPPE = 1b)

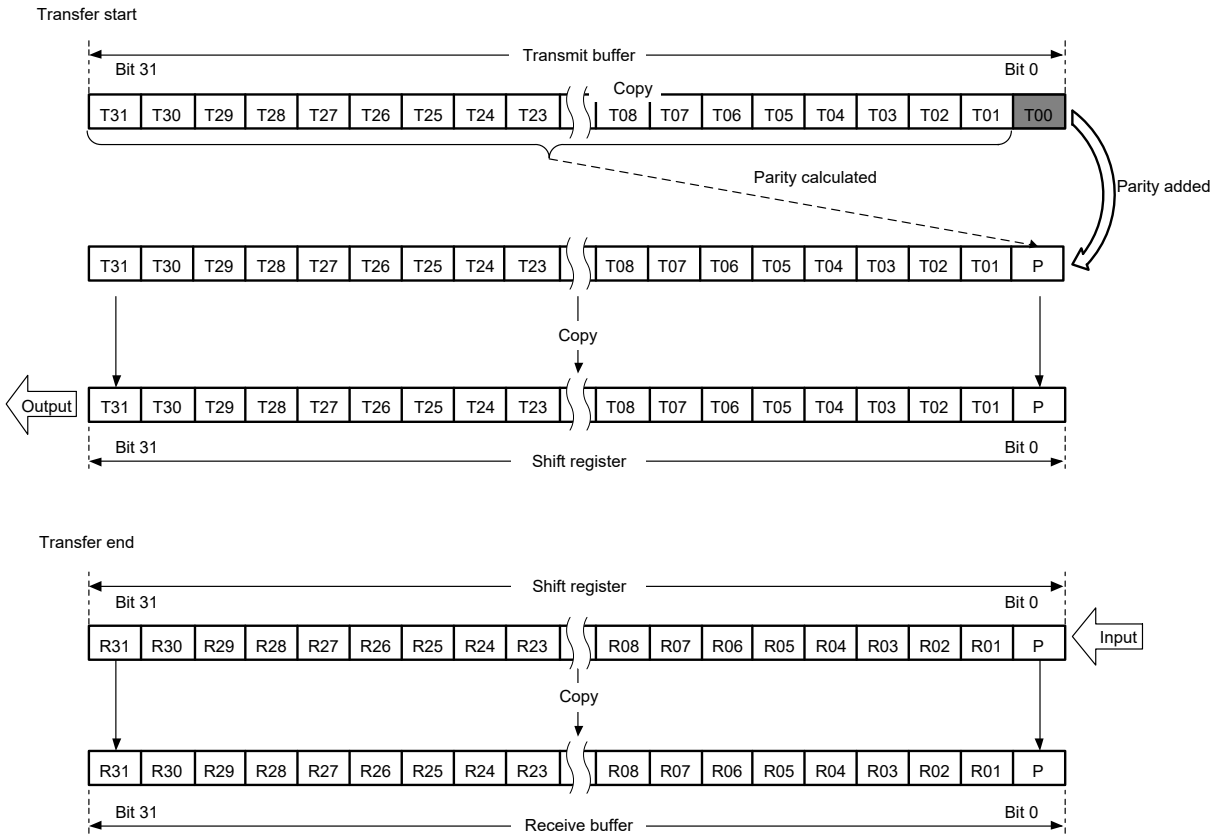
When the parity function is enabled, the LSB of transmit data or received data is converted to a parity bit. The parity bit value is calculated and converted by the hardware.

(1) MSB first transfer (32-bit data)

Figure 7.5-20 shows operations of the SPDR register and the shift register when the SPI performs 32-bit MSB-first transfer with the parity function enabled.

In transmission, the parity bit (P) value is calculated first from the data value T31 to T01, then the final bit T00 is replaced with the calculated P value, and then transmit data is copied to the shift register. Transmit data is transmitted in the order of T31 → T30 → … → T01 → P.

In reception, received data is stored in bit 0 of the shift register, and is then shifted in units of data. When necessary RSPCK cycles are input and received data is stored from R31 to P, the shift register value is copied to the receive buffer, and at the same time, R31 to P data is checked to determine whether a parity error is present.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

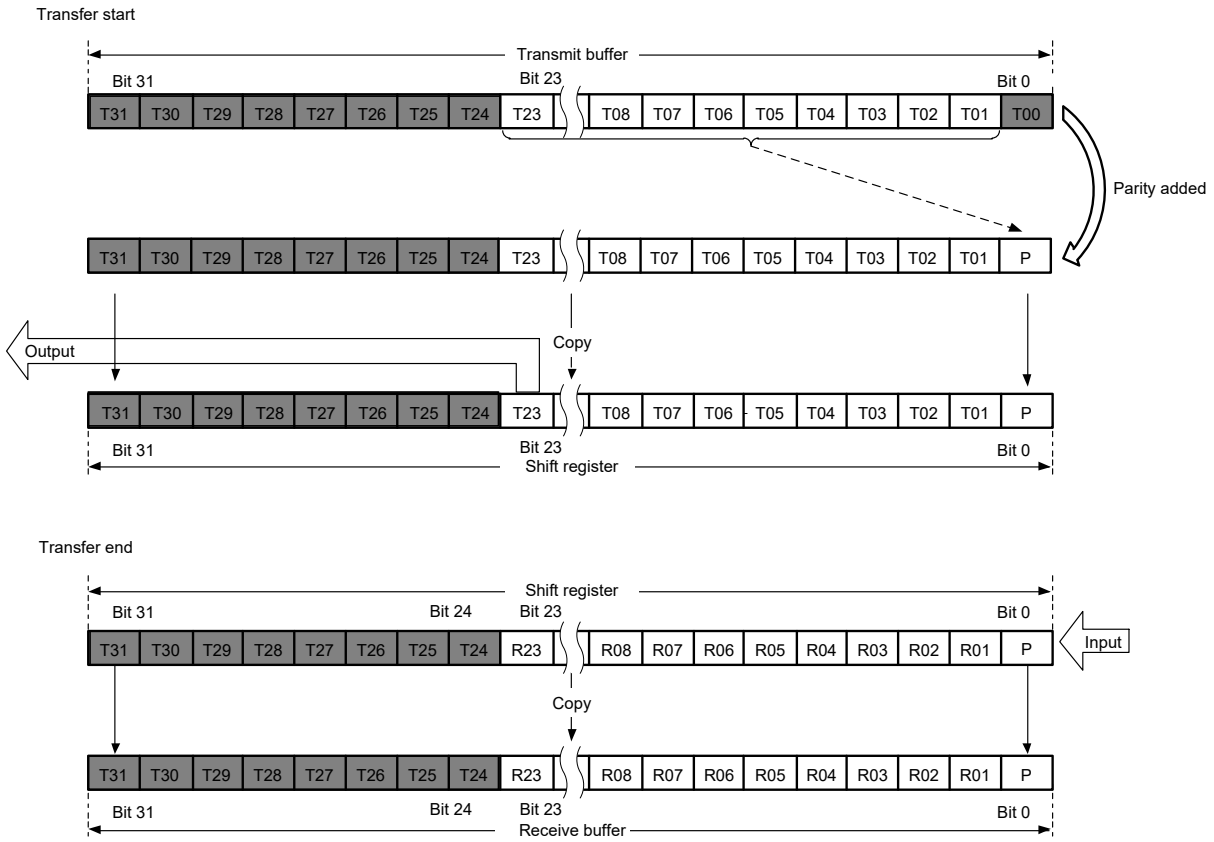
Figure 7.5-20 MSB First Transfer (32-Bit Data, Parity Enabled)

(2) MSB first transfer (24-bit data)

As an example of MSB-first transfer of data other than 32 bits with the parity function enabled, **Figure 7.5-21** shows operations of the SPDR register and the shift register when the SPI performs 24-bit data transfer.

In transmission, the parity bit (P) value is calculated first from the data value T23 to T01, then the final bit T00 is replaced with the calculated P value, and then transmit data is copied to the shift register. Transmit data is transmitted in the order of T23 → T22 → … → T01 → P.

In reception, received data is stored in bit 0 of the shift register, and is then shifted in units of data. When necessary RSPCK cycles are input and received data is stored from R23 to P, the shift register value is copied to the receive buffer, and at the same time, R23 to P data is checked to determine whether a parity error is present.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

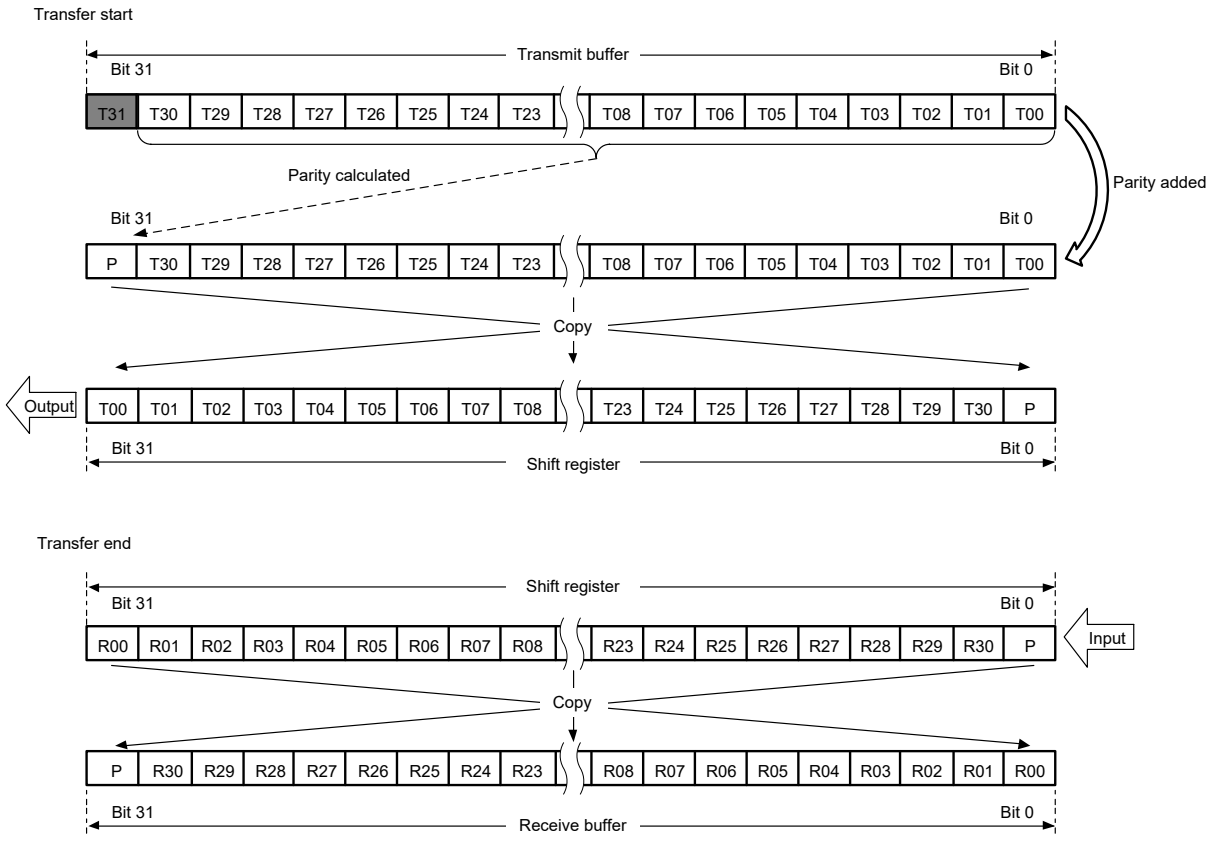
Figure 7.5-21 MSB First Transfer (24-Bit Data, Parity Enabled)

(3) LSB first transfer (32-bit data)

Figure 7.5-22 shows operations of the SPDR register and the shift register when the SPI performs 32-bit LSB-first transfer with the parity function enabled.

In transmission, the parity bit (P) value is calculated first from the data value T30 to T00, then the final bit T31 is replaced with the calculated P value, and then transmit data is copied to the shift register. Transmit data is transmitted in the order of T00 → T01 → … → T30 → P.

In reception, received data is stored in bit 0 of the shift register, and is then shifted in units of data. When necessary RSPCK cycles are input and received data is stored from R00 to P, bit values in the shift register are reversed in bit units and are copied to the receive buffer in the order of P to R00, and at the same time, R00 to P data is checked to determine whether a parity error is present.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

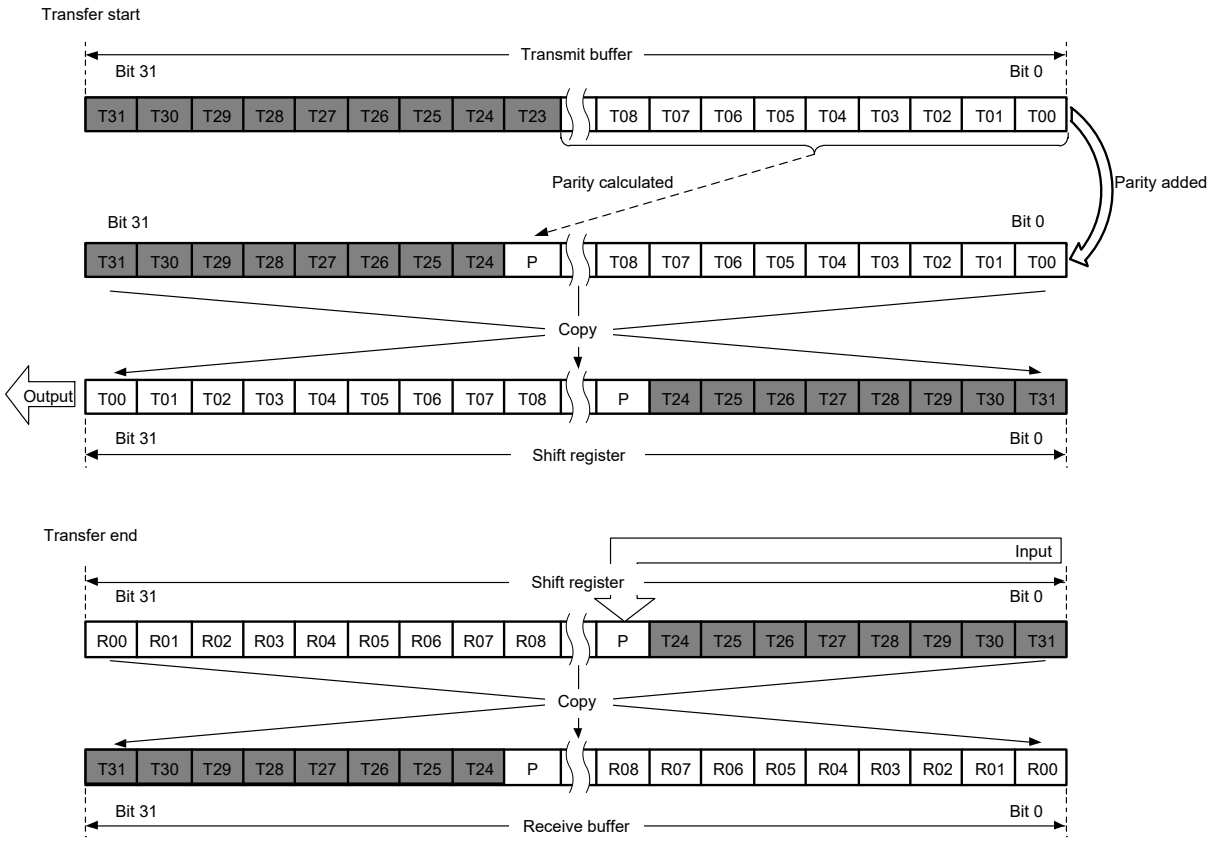
Figure 7.5-22 LSB First Transfer (32-Bit Data, Parity Enabled)

(4) LSB first transfer (24-bit data)

As an example of LSB-first transfer of data other than 32 bits with the parity function enabled, **Figure 7.5-23** shows operations of the SPDR register and the shift register when the SPI performs 24-bit data transfer.

In transmission, the parity bit (P) value is calculated first from the data value T22 to T00, then the final bit T23 is replaced with the calculated P value, and then transmit data is copied to the shift register. Transmit data is transmitted in the order of T00 → T01 → … → T22 → P.

In reception, received data is stored in bit 8 of the shift register, and is then shifted in units of data. When necessary RSPCK cycles are input and received data is stored from R00 to P, bit values in the shift register are reversed in bit units and are copied to the receive buffer in the order of P (bit 23) to R00 (bit 0), and at the same time, R00 to P data is checked to determine whether a parity error is present.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 7.5-23 LSB First Transfer (24-Bit Data, Parity Enabled)

7.5.3.4.4 Byte swap transmission

When byte swapping is enabled, the data in the transmission buffer, swapped in 8-bit units, is copied to the shift register.

(1) 32-bit data length

Figure 7.5-24 shows the relationship between the SPDR (transmit buffer) and the shift register when transferring data with a 32-bit data length, using a combination of MSB/LSB first and with/without byte swap.

1. MSB First Transfer (When the byte swap is disabled.)
   Data (Byte3 [T31 to T24] to Byte0 [T07 to T00]) in the transmit buffer are copied to the shift register.
   Bit values in the shift register are shifted and transmitted in the order of T31 → T30 → … → T00 as transmit data.
2. MSB First Transfer (When the byte swap is enabled.)
   Byte values of the transmit buffer (Byte3 [T31 to T24] to Byte0 [T07 to T00]) are reversed in byte units and are copied to the shift register in the order of Byte3 [T07 to T00] to Byte0 [T31 to T24].
   Bit values in the shift register are shifted and transmitted in the order of T07 → T06 → … → T00 → T15 → T14→ … → T08 → T23 → T22 → … → T16 → T31 → T30 → … → T24 as transmit data.
3. LSB-first transfer (When the byte swap is disabled.)
   Bit values of the transmit buffer (Byte3 [T31 to T24] to Byte0 [T07 to T00]) are reversed in bit units and are

copied to the shift register in the order of Byte0 [T00 to T07] to Byte3 [T24 to T31].
Bit values in the shift register are shifted and transmitted in the order of T00 → T01 → … → T31 as transmit data.

4. LSB-first transfer (When the byte swap is enabled.)
Bit values of each byte of the transmit buffer (Byte3 [T31 to T24] to Byte0 [T07 to T00]) are reversed in bit units and are copied to the shift register in the order of Byte3 [T24 to T31] to Byte0 [T00 to T07].
Bit values in the shift register are shifted and transmitted in the order of T24 → T25 → … → T31 → T16 → T17 → … → T23 → T08 → T09 → … → T15 → T00 → T01 → … → T07 as transmit data.

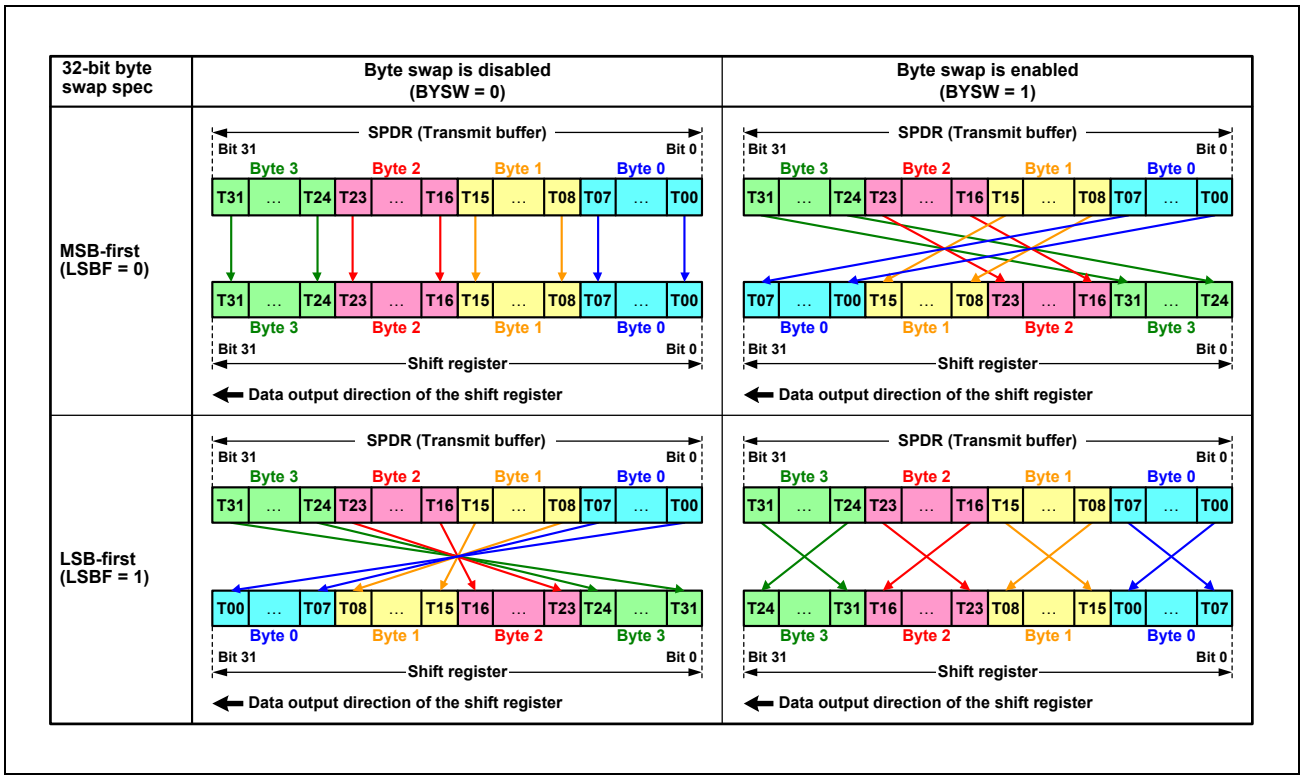


Figure 7.5-24 Byte Swap with MSB/LSB Transfer (32 Bits)

(2) 16-bit data length

Figure 7.5-25 shows the relationship between the SPDR (transmit buffer) and the shift register when transferring data with a 16-bit data length, using a combination of MSB/LSB first and with/without byte swap.

1. MSB First Transfer (When the byte swap is disabled.)
Data (Byte1 [T15 to T08] to Byte0 [T07 to T00]) in the transmit buffer are copied to the shift register.
Bit values in the shift register are shifted and transmitted in the order of T15 → T14 → … → T00 as transmit data.

2. MSB First Transfer (When the byte swap is enabled.)
Byte values of the transmit buffer (Byte1 [T15 to T08] to Byte0 [T07 to T00]) are reversed in byte units and are copied to the shift register in the order of Byte1 [T07 to T00] to Byte0 [T15 to T08].
Bit values in the shift register are shifted and transmitted in the order of T07 → T06 → … → T00 → T15 → T14→ … → T08 as transmit data.

3. LSB-first transfer (When the byte swap is disabled.)
Bit values of the transmit buffer (Byte1 [T15 to T08] to Byte0 [T07 to T00]) are reversed in bit units and are copied to the shift register in the order of Byte0 [T00 to T07] to Byte1 [T08 to T15].
Bit values in the shift register are shifted and transmitted in the order of T00 → T01 → … → T15 as transmit data.

4. LSB-first transfer (When the byte swap is enabled.)
   Bit values of each byte of the transmit buffer (Byte1 [T15 to T08] to Byte0 [T07 to T00]) are reversed in bit units and are copied to the shift register in the order of Byte1 [T08 to T15] to Byte0 [T00 to T07].
   Bit values in the shift register are shifted and transmitted in the order of T08 → T09 → … → T15 → T00 → T01 → … → T07 as transmit data.

Figure 7.5-25 Byte Swap with MSB/LSB Transfer (16 Bits)

NOTE

- When using the byte swap, set 16 bits or 32 bits to the data length (SPCMD.SPB[4:0] setting).
  If setting the other length, the behavior is not guaranteed.
- When the byte swap is valid, set the parity function as invalid (SPCR.SPPE bit = 0b).
  If setting the parity function as valid (SPPE bit = 1b), the behavior is not guaranteed.
- Set SPDCR.BYSW bit, when SPCR.SPE bit is 0b. If rewriting BYSW bit, when SPE bit is 1b, the behavior after it is not guaranteed.

7.5.3.4.5 Byte swap reception

When byte swap is enabled, the data in the shift register, swapped in 8-bit units, is copied to the receive buffer.

(1) 32-bit data length

Figure 7.5-26 shows the relationship between the shift register and SPDR (reception buffer) when transferring data with a 32- bit data length, using a combination of MSB/LSB first and with/without byte swap.

1. MSB First Transfer (When the byte swap is disabled.)
   The first received data (R31) is stored in bit 0 of the shift register, and received data is shifted in the order of R31 → R30 → … → R00.
   When necessary RSPCK cycles are input and data is stored from Byte3 [R31 to R24] to Byte0 [R07 to R00], the shift register value is copied to the receive buffer.
2. MSB First Transfer (When the byte swap is enabled.)
   The first received data (R07) is stored in bit 0 of the shift register, and received data is shifted in the order of R07 → R06 → … → R00 → R15 → R14 → … → R08 → R23 → R22 → … → R16 → R31 → R30 → … → R24.
   When necessary RSPCK cycles are input and data is stored from Byte0 [R07 to R00] to Byte3 [R31 to R24], byte values in the shift register are reversed in byte units and are copied to the receive buffer in the order of Byte3 [R31 to R24] to Byte0 [R07 to R00].
3. LSB-first transfer (When the byte swap is disabled.)
   The first received data (R00) is stored in bit 0 of the shift register, and received data is shifted in the order of R00 → R01 → … → R31.
   When necessary RSPCK cycles are input and data is stored from Byte0 [R00 to R07] to Byte3 [R24 to R31], bit values in the shift register are reversed in bit units and are copied to the receive buffer in the order of Byte3 [R31 to R24] to Byte0 [R07 to R00].
4. LSB-first transfer (When the byte swap is enabled.)
   The first received data (R24) is stored in bit 0 of the shift register, and received data is shifted in the order of R24 → R25 → … → R31 → R16 → R17 → … → R23 → R08 → R09 → … → R15 → R00 → R01 → … → R07.
   When necessary RSPCK cycles are input and data is stored from Byte3 [R24 to R31] to Byte0 [R00 to R07], bit values of each byte in the shift register are reversed in bit units and are copied to the receive buffer in the order of Byte3 [R31 to R24] to Byte0 [R07 to R00].

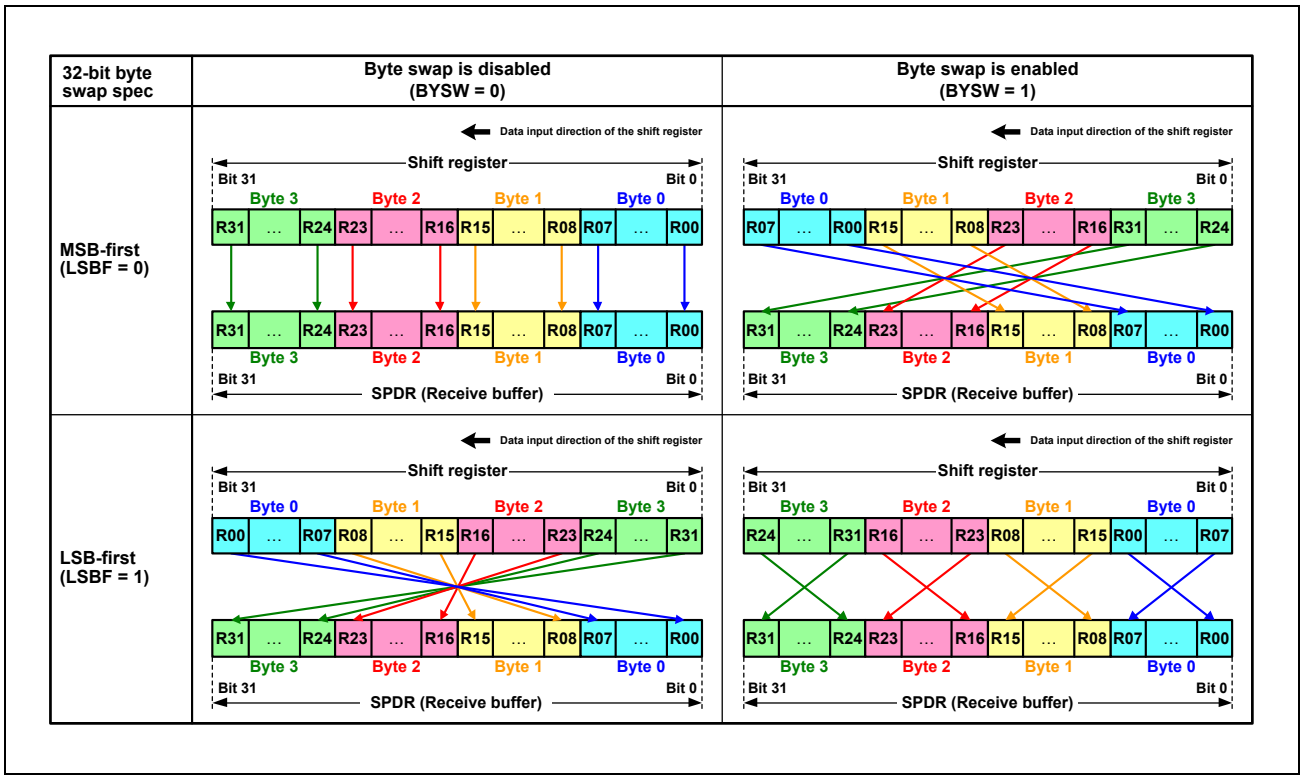


Figure 7.5-26 Byte Swap with MSB/LSB Reception (32 Bits)

(2) 16-bit data length

Figure 7.5-27 shows the relationship between the shift register and SPDR (reception buffer) when transferring data with a 16- bit data length, using a combination of MSB/LSB first and with/without byte swap.

1. MSB First Transfer (When the byte swap is disabled.)
   The first received data (R15) is stored in bit 0 of the shift register, and received data is shifted in the order of R15 → R14 → … → R00.
   When necessary RSPCK cycles are input and data is stored from Byte1 [R15 to R08] to Byte0 [R07 to R00], the shift register value is copied to the receive buffer.

2. MSB First Transfer (When the byte swap is enabled.)
   The first received data (R07) is stored in bit 0 of the shift register, and received data is shifted in the order of R07 → R06 → … → R00 → R15 → R14 → … → R08.
   When necessary RSPCK cycles are input and data is stored from Byte0 [R07 to R00] to Byte1 [R15 to R08], byte values in the shift register are reversed in byte units and are copied to the receive buffer in the order of Byte1 [R15 to R08] to Byte0 [R07 to R00].

3. LSB-first transfer (When the byte swap is disabled.)
   The first received data (R00) is stored in bit 15 of the shift register, and received data is shifted in the order of R00 → R01 → … → R15.
   When necessary RSPCK cycles are input and data is stored from Byte0 [R00 to R07] to Byte1 [R08 to R15], bit values in the shift register are reversed in bit units and are copied to the receive buffer in the order of Byte1 [R15 to R08] to Byte0 [R07 to R00].

4. LSB-first transfer (When the byte swap is enabled.)
   The first received data (R08) is stored in bit 0 of the shift register, and received data is shifted in the order of R08 → R09 → … → R15 → R00 → R01 → … → R07.

When necessary RSPCK cycles are input and data is stored from Byte1 [R08 to R15] to Byte0 [R00 to R07], bit values of each byte in the shift register are reversed in bit units and are copied to the receive buffer in the order of Byte1 [R15 to R08] to Byte0 [R07 to R00].

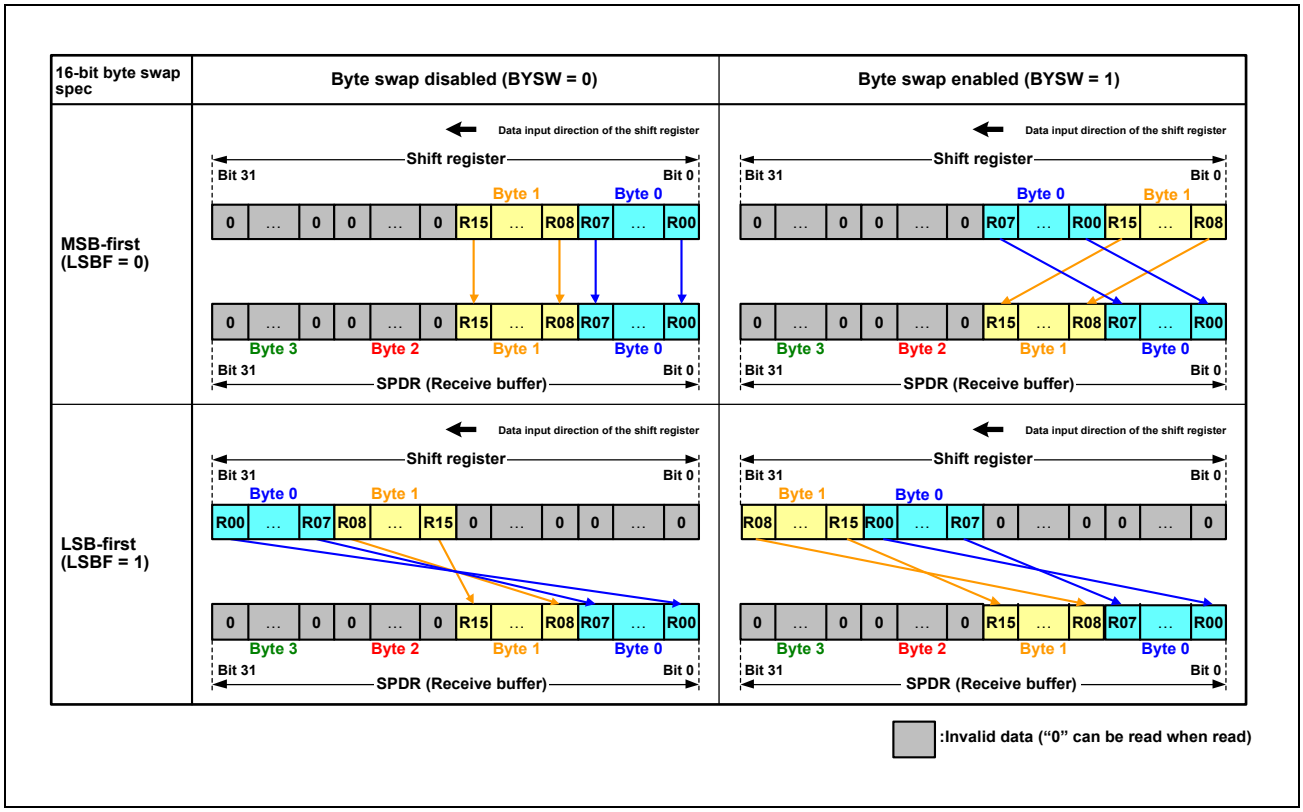


Figure 7.5-27 Byte Swap with MSB/LSB Reception (16 Bits)

NOTE

- When using the byte swap, set 16 bits or 32 bits to the data length (SPCMD.SPB[4:0] setting) .
  If setting the other length, the behavior is not guaranteed.
- When the byte swap is valid, set the parity function as invalid (SPCR.SPPE bit = 0b).
  If setting the parity function as valid (SPPE bit = 1b), the behavior is not guaranteed.
- Set SPDCR2.BYSW bit, when SPCR.SPE bit is 0b. If rewriting BYSW bit, when SPE bit is 1b, the behavior after it is not guaranteed.

7.5.3.5 Transfer Format (Frame Format)

7.5.3.5.1 CPHA = 0b

Figure 7.5-28 shows an example of transfer format of 8-bit data serial transfer when SPCMD.CPHA = 0b. However, when the SPI is in slave mode (SPCR.MSTR = 0b) with the CPHA bit set to 0b, clock synchronous operation (SPCR.SPMS = 1b) is not guaranteed. In **Figure 7.5-28**, RSPCK (CPOL = 0b) is the RSPCK signal waveform when SPCMD.CPOL = 0b, and RSPCK (SPCMD.CPOL = 1b) is the RSPCK signal waveform when SPCMD.CPOL = 1b. The sampling timing shows the time when the SPI latches serial transfer data in the shift register. Input and output directions of each signal depend on SPI settings. (See **7.5.3.2 SPI Pin Control**.)

When SPCMD.CPHA = 0b, output of valid data to the MOSI signal and valid data drive to the MISO signal start at the SSL signal assertion timing. The first RSPCK signal change timing after the SSL signal is asserted is the first transfer data latch timing. After this timing, data is sampled in each RSPCK cycle. The MOSI signal and the MISO signal change always after the half RSPCK cycles of the transfer data latch timing. The set CPOL bit value does not affect the RSPCK signal operation timing but affects only the signal polarity.

Period t1 is the period (RSPCK delay) from SSL signal assertion to RSPCK oscillation. Period t2 is the period (SSL negation delay) from RSPCK oscillation stop to SSL signal negation. Period t3 is the period (next-access delay) to inhibit SSL signal assertion for the next transfer after serial transfer is completed. Periods t1, t2, and t3 are controlled by the master device in the SPI system. For t1, t2, and t3 when this LSI (SPI) is in master mode, see **7.5.3.13.1 Master mode operation**.

[In the Motorola-SPI case]

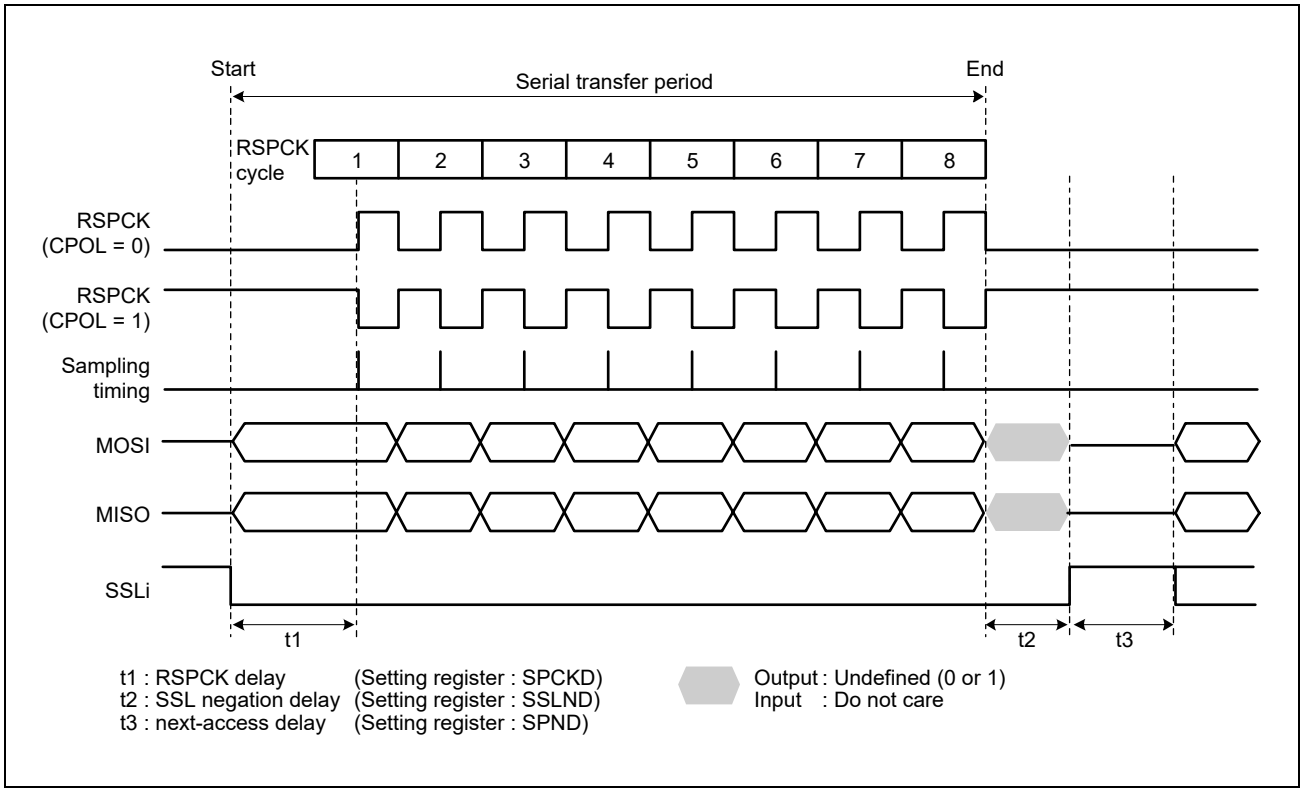


Figure 7.5-28 SPI Transfer Format (CPHA = 0b, SPFRF = 0b)

7.5.3.5.2 CPHA = 1b

Figure 7.5-29 shows an example of transfer format of 8-bit data serial transfer when SPCMD.CPHA = 1b. However, when SPCR.SPMS = 1b, communication is performed by using only RSPCK, MOSI, and MISO signals without using the SSL signal. In **Figure 7.5-29**, RSPCK (CPOL = 0b) is the RSPCK signal waveform when SPCMD.CPOL = 0b, and RSPCK (SPCMD.CPOL = 0b) is the RSPCK signal waveform when SPCMD.CPOL = 0b. The sampling timing shows the time when the SPI latches serial transfer data in the shift register. Input and output directions of each signal depend on SPI mode (master or slave mode). (See **7.5.3.2 SPI Pin Control**.)

When SPCMD.CPHA = 1b, invalid data drive to the MISO signal starts at the SSL signal assertion timing. Output of valid data to the MOSI and MISO signals starts at the first RSPCK signal change timing after the SSL signal is asserted.

After this timing, data is updated in each RSPCK cycle. Transfer data is always latched after 1/2 RSPCK cycle of this timing. The set CPOL bit value does not affect the RSPCK signal operation timing but affects only the signal polarity.

Periods t1, t2, and t3 are the same as when CPHA = 0b. For t1, t2, and t3 when this LSI (SPI) is in master mode, see **7.5.3.13.1 Master mode operation**.

[In the Motorola-SPI case]

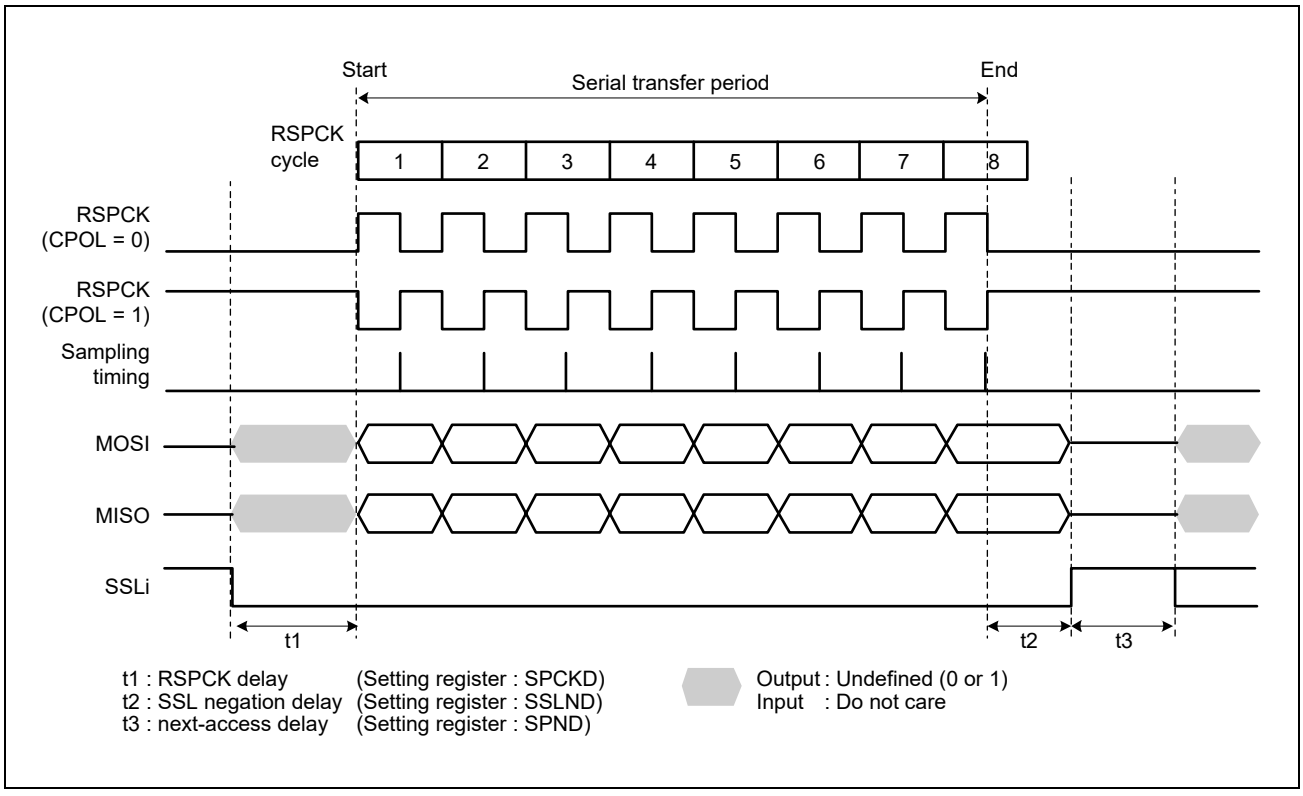


Figure 7.5-29 SPI Transfer Format (CPHA = 1b, SPFRF = 0b)

7.5.3.6 Communications Operating Mode

Transmit-Receive serial communication, transmit-only operation, and Receive-only operation are selected by setting the Communication Mode Select bits (TXMD[1:0]) of the SPCR register.

SPDR access described in **Figure 7.5-30**, **Figure 7.5-31**, and **Figure 7.5-32** shows an access to the SPDR register. W shows a write cycle.

7.5.3.6.1 Transmit-receive serial communication (TXMD[1:0] = 00b)

Figure 7.5-30 shows an example of operation when the communication mode select bit (TXMD[1:0]) in the SPI control register (SPCR) is set to 00b. In the example in **Figure 7.5-30**, the SPI performs 8-bit data serial transfer with the settings of RTRG = FIFO stage - 1, SPDCR2.TTRG = 0b, SPCMD.CPHA = 1b, and SPCMD.CPOL = 0b. Numbers under the RSPCK waveform show the number of RSPCK cycles (number of transfer bits).

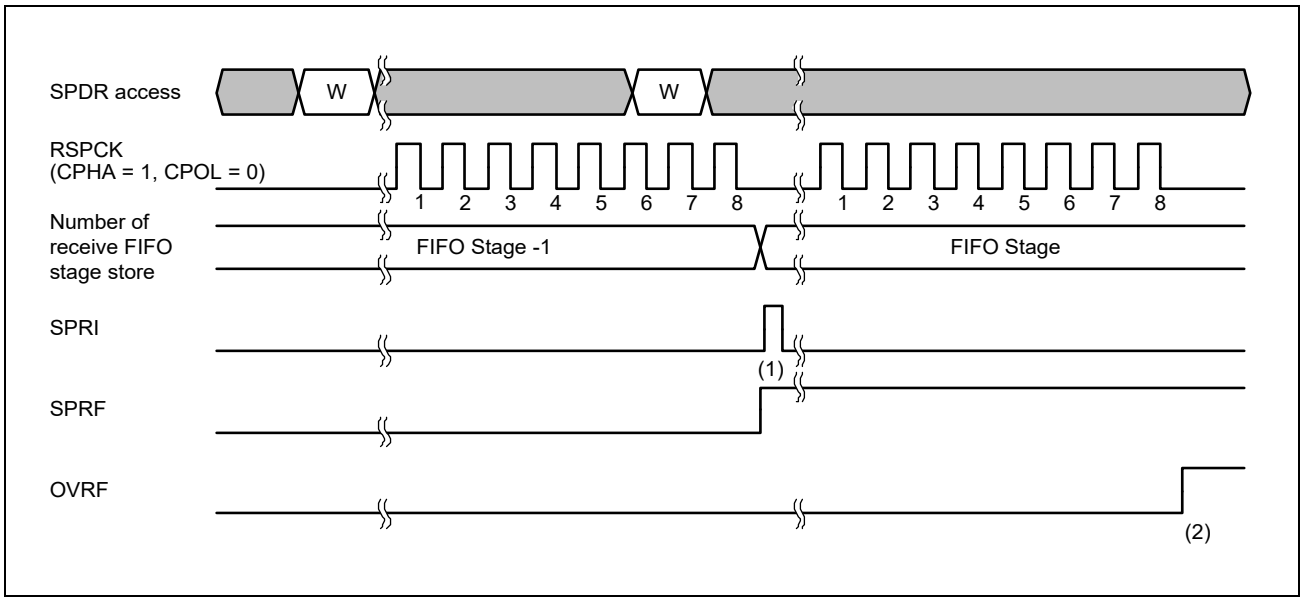


Figure 7.5-30 Operation Example of TXMD[1:0] = 00b

The following describes operation of flags at timings (1) and (2) in the figure above.

(1) When serial transfer ends while the number of SPDR receive buffer store matches the number of frames set in SPDCR2.RTRG, the SPI generates a receive buffer full interrupt request SPRI (setting the SPSR.SPRF flag to 1b) and copies the received data in the shift register to the receive buffer.

(2) When serial transfer ends with data for the number of FIFO stages stored in the SPDR receive buffer, the SPI sets the OVRF flag in the SPSR register to 1 and discards the received data in the shift register.

In Transmit-Receive serial communication (TXMD[1:0] = 00b), transmit data is transmitted and receive data is received. Therefore, the SPRF flag and the OVRF flag are set to 1b at timings (1) and (2) respectively.

7.5.3.6.2 Transmit-only serial communication (TXMD[1:0] = 01b)

Figure 7.5-31 shows an example of operation when the communication mode select bits (TXMD[1:0]) in the SPCR register are set to 01b. In the example in **Figure 7.5-31**, the SPI performs 8-bit data serial transfer with the settings of SPDCR2.TTRG[1:0] = 00b, SPDCR2.RTRG[1:0] = 00b, SPCMD.CPHA = 1b, and SPCMD.CPOL = 0b. Numbers under the RSPCK waveform show the number of RSPCK cycles (number of transfer bits).

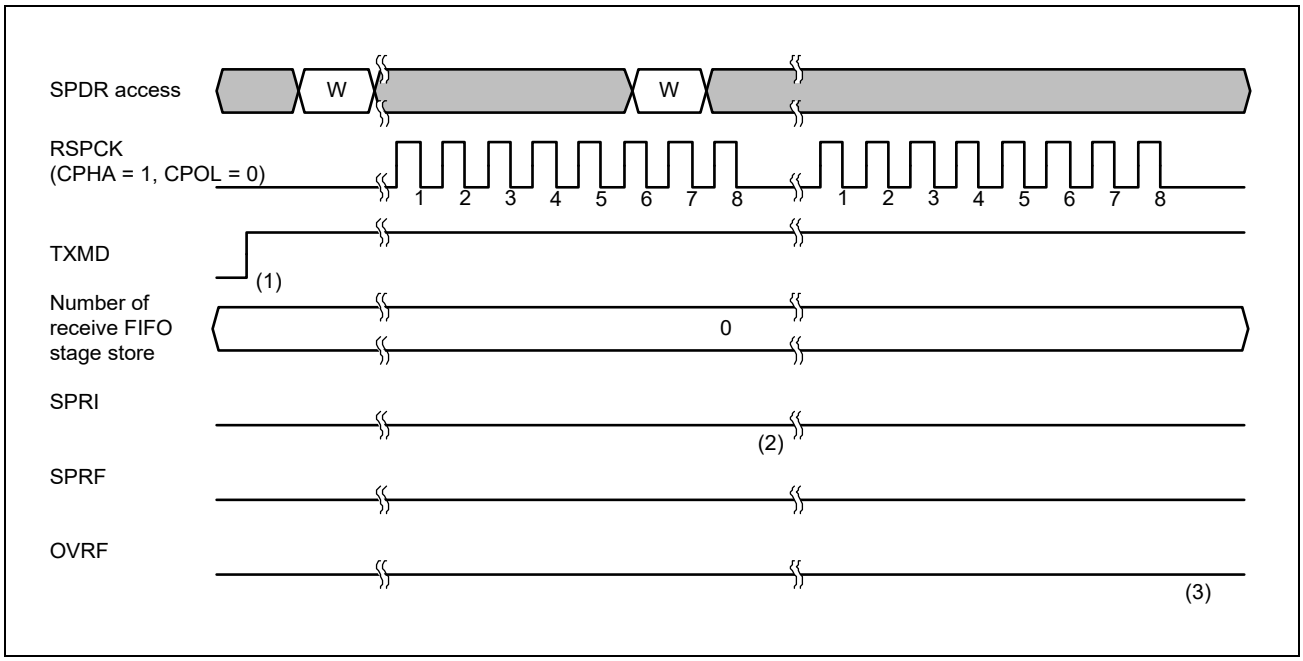


Figure 7.5-31 Operation Example of TXMD[1:0] = 01b

The following describes operation of flags at timings (1) to (3) in the figure above.

(1) Before setting TXMD[1:0] to 01b (transmit-only serial communication), confirm that the receive buffer is empty and that the SPRF and OVRF flags in the SPSR register are 0b.

(2) In transmit-only serial communication (TXMD[1:0] = 01b), when serial transfer ends without receiving data in the receiving FIFO of SPDR, the SPRF flag remains 0 and the shift register data is not copied to the receive buffer.

(3) The OVRF flag remains 0 even after serial transfer ends because previously received data is not remaining in the SPDR's receive buffer, and the shift register data is not copied to the receive buffer.

In transmit-only serial communication (TXMD[1:0] = 01b), transmit data is transmitted but receive data is not received. Therefore, the SPRF flag and the OVRF flag remain 0 at each timing of (1) to (3).

7.5.3.6.3 Receive-only serial communication (TXMD[1:0] = 10b)

Figure 7.5-32 shows an example of operation when the communication mode select bit (TXMD[1]) in the SPI control register (SPCR) is set to 1b. In the example in **Figure 7.5-32**, the SPI performs 8-bit data serial transfer with the settings of SPDCR2.TTRG[1:0] = 0b, SPDCR2.RTRG[1:0] = FIFO stage - 1, SPCMD.CPHA = 1b, and CPOL in SPCMD = 0b. Numbers under the RSPCK waveform show the number of RSPCK cycles (number of transfer bits).

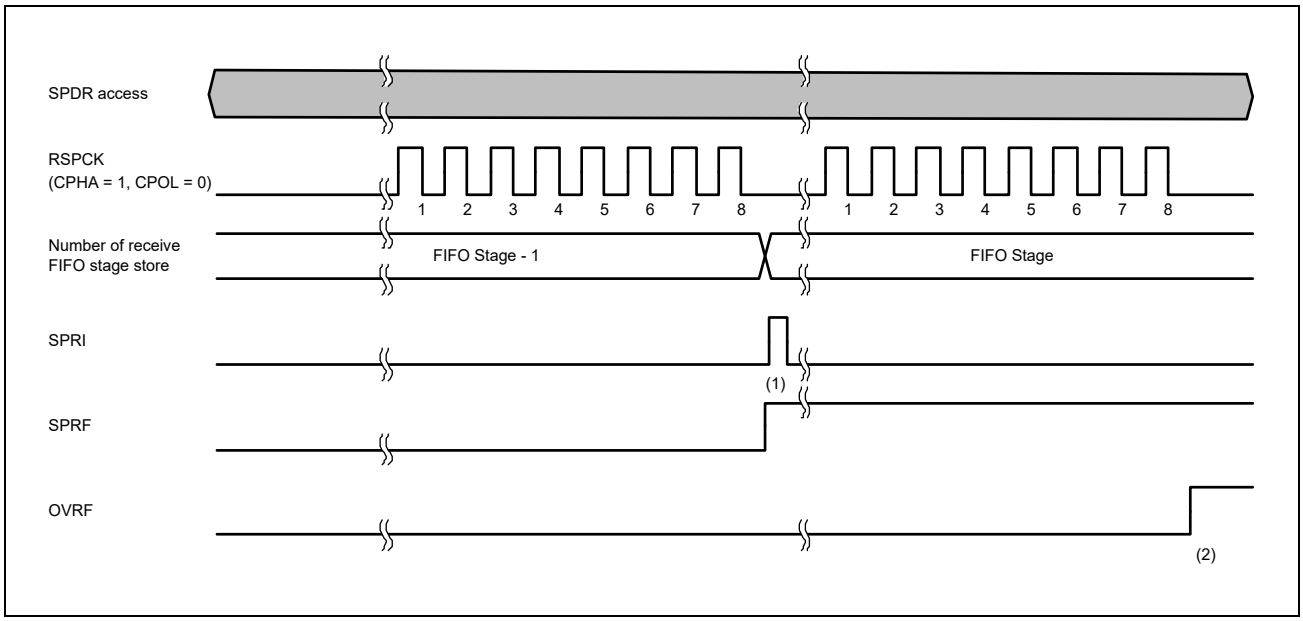


Figure 7.5-32 Operation Example of TXMD[1:0] = 10b

The following describes operation of flags at timings (1) and (2) in the figure above.

(1) When serial transfer ends while the number of SPDR receive buffer store matches the number of frames set in SPDCR2.RTRG, the SPI generates a receive buffer full interrupt request SPRI (setting the SPSR.SPRF flag to 1) and copies the received data in the shift register to the receive buffer.

(2) When serial transfer ends with data for the number of FIFO stages stored in the SPDR receive buffer, the SPI sets the OVRF flag in the SPSR.SPI register to 1b and discards the received data in the shift register.

7.5.3.7 Transmission buffer empty/receive buffer full interrupt

Figure 7.5-33 show an example of operation of SPI transmit buffer empty interrupt (SPTI) and SPI receive buffer full interrupt (SPRI) respectively. “SPDR access” described in **Figure 7.5-33** and shows an access to the SPDR register. W shows a write cycle and R shows a read cycle. In each example of operation, the SPI performs 8-bit data serial transfer with the settings of SPCR.TXMD[1:0] = 00b, SPDCR2.TTRG = 0b, SPDCR2.RTRG = 0b, SPCMD.CPHA = 0b (**Figure 7.5-33**), and SPCMD.CPOL = 0b. Numbers under the RSPCK waveform show the number of RSPCK cycles (number of transfer bits).

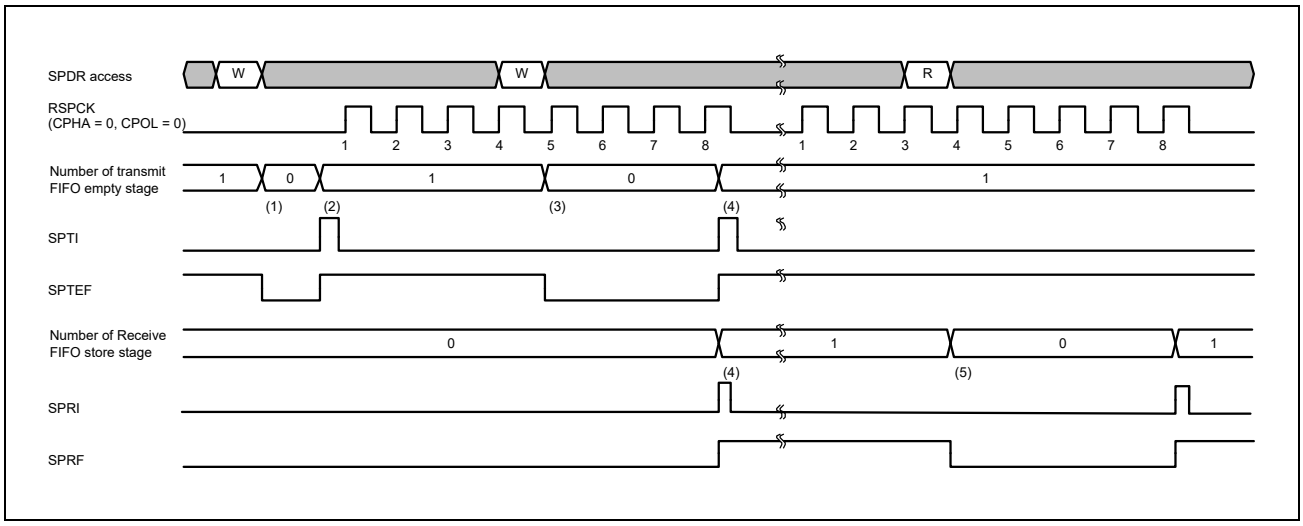


Figure 7.5-33 Example of SPTI Interrupt and SPRI Interrupt Operations

The following describes operation at timings (1) to (5) in the figure above.

(1) When transmit data is written to SPDR before the data for the next transfer is not set to the FIFO of the SPDR, the SPI writes data to the transmission buffer. When transmit data is written to SPDR in one processing routine using DMAC, the SPSR.SPTEF flag is cleared to 0b at the last access.

(2) When the shift register is empty, the SPI copies transmit buffer data to the shift register. At this time, if transmit FIFO empty stage number > TTRG value, then the SPI generates a transmit buffer empty interrupt request (SPTI) and sets the SPTEF flag to 1. How to start serial transfer depends on SPI mode. (See **7.5.3.13 SPI Operation** and **7.5.3.14 Clock Synchronous Operation**.)

(3) When transmit data is written to SPDR upon a transmit buffer empty interrupt request (SPTI) or in the transmit buffer empty determination processing due to the SPTEF flag, the SPI writes data to the transmit buffer. When the transmit data is written to SPDR in one processing routine using DMAC, the SPTEF flag is cleared to 0b at the last access. The SPI does not copy transmit buffer data to the shift register because the shift register contains serial transfer data.

(4) When serial transfer ends (data sampling clock edge in the final bit is detected) while the receive buffer of SPDR > FIFO stage number, the SPI copies the received data in the shift register to the receive buffer. At this time, a receive buffer full interrupt request is generated and the SPRF flag is set to 1b. When serial transfer ends, the shift register becomes empty. If the next transfer data is set in the transmit FIFO before serial transfer ends, the SPI set the SPTEF flag to 1b and copies transmit buffer data to the shift register. Even if received data is not copied from the shift register to the receive buffer while an overrun error is present, the SPI determines that the shift register is empty at the end of serial transfer, enabling data transfer from the transmit buffer to the shift register.

(5) When SPDR is read upon a receive buffer full interrupt request (SPRI) or in the receive buffer full determination processing due to the SPRF flag, received data can be read. If the received data is read from SPDR in one processing routine using DMAC, the SPRF flag is cleared to 0b at the last access.

When transmit data is written to the SPDR register while no empty stages in the transmit FIFO, the SPI does not update transmit buffer data. Write transmit data to the SPDR register upon a transmit buffer empty interrupt request or check the empty or not the transmission buffer by the SPTEF flag.

While SPCR.SPTIE = 1b, when the SPI is disabled (SPCR.SPE = 0b), an SPI transmit buffer empty interrupt is generated. However, SPI transmit buffer empty interrupt is inhibited by disabling SPI transmit buffer empty interrupt (SPCR.SPTIE = 0b) at the same time as SPE bit setting.

When serial transfer ends while data is stored in the receive FIFO for the number of FIFO stages, the SPI detects an overrun error without copying data from the shift register to the receive buffer (as described in **7.5.3.10 Error Detection**). To prevent overrun of received data, read received data before the next serial transfer ends.

Transmit buffer state and receive buffer state can be monitored by an SPI transmit buffer empty interrupt and an SPI receive buffer full interrupt or a corresponding interrupt flag or by the SPTEF and SPRF flags.

7.5.3.8 Idle Interrupt

The idle interrupt during master mode operation is when the SPCP[2:0] of the SPSSR register becomes 000b (start of sequence control), the IDLNF flag in the SPSR register is set to 1b and an idle interrupt request is made during master mode operation. An interrupt request is also made by clearing the SPCR.SPE bit to 0b.

7.5.3.8.1 In the Motorola-SPI case

Figure 7.5-34 shows an example of idle interrupt operation during normal operation.

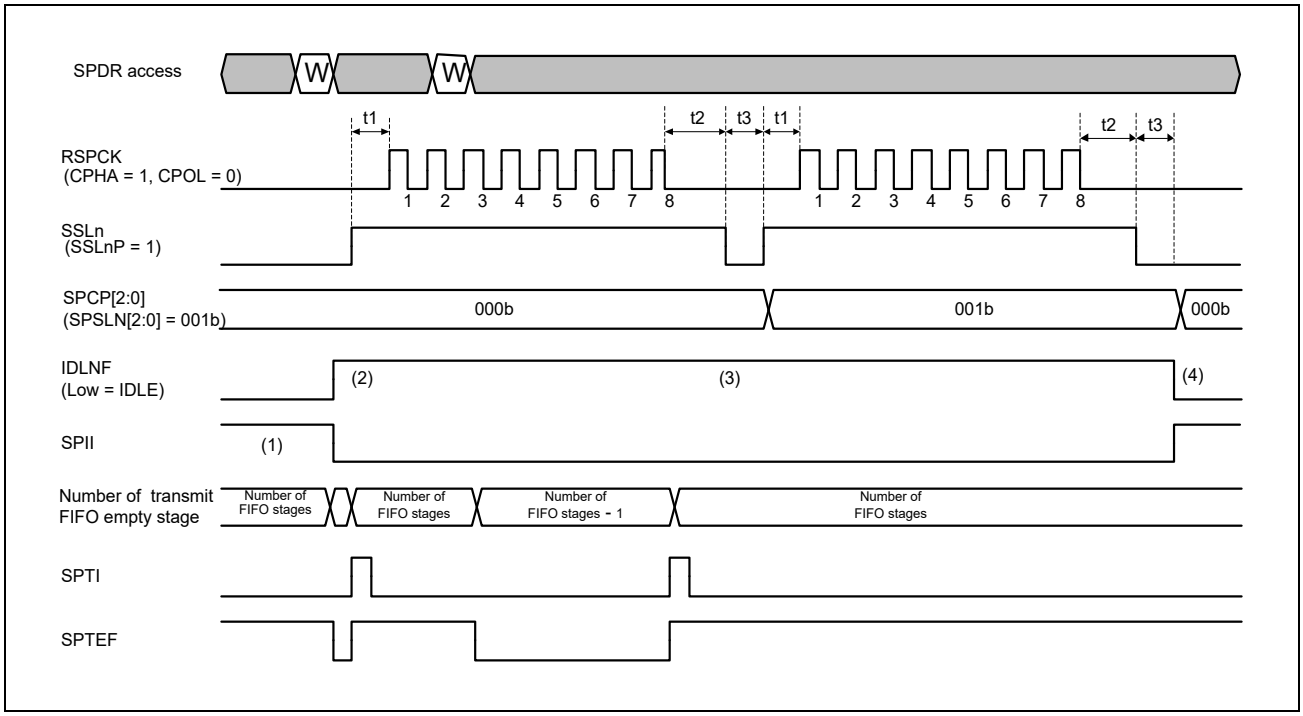


Figure 7.5-34 Example of Idle Interrupt Operation (Master Mode/Motorola-SPI)

(1) At the start of transmission, if the next transfer data is not set in the transmission buffer, the IDLNF flag is 0 (IDLE). Writing transmit data makes sets the IDLNF flag to 1 (BUSY). When the SPIIE bit in the SPI control register (SPCR) is set to 1b before transmit data is written, interrupt processing is required before transmission start. For this reason, set the SPIIE bit to 0b before starting transmission.

(2) After transmission has started, the IDLNF flag remains 1b (BUSY) regardless of the transmit buffer state.

(3) The SPCP[2:0] bits change the command to the next command the end of t3 cycle. When the next command is not 000b, the IDLNF flag remains unchanged even when the next transmit data has not been written.

(4) The IDLNF flag is cleared to 0 (IDLE) the end of t3 cycle because the next command is 000b and the next transmit data is not present. When the SPIIE bit is 1b currently, an SPII interrupt is output.

7.5.3.8.2 In the TI-SSP case

Figure 7.5-35 shows an example of idle interrupt operation during normal operation.

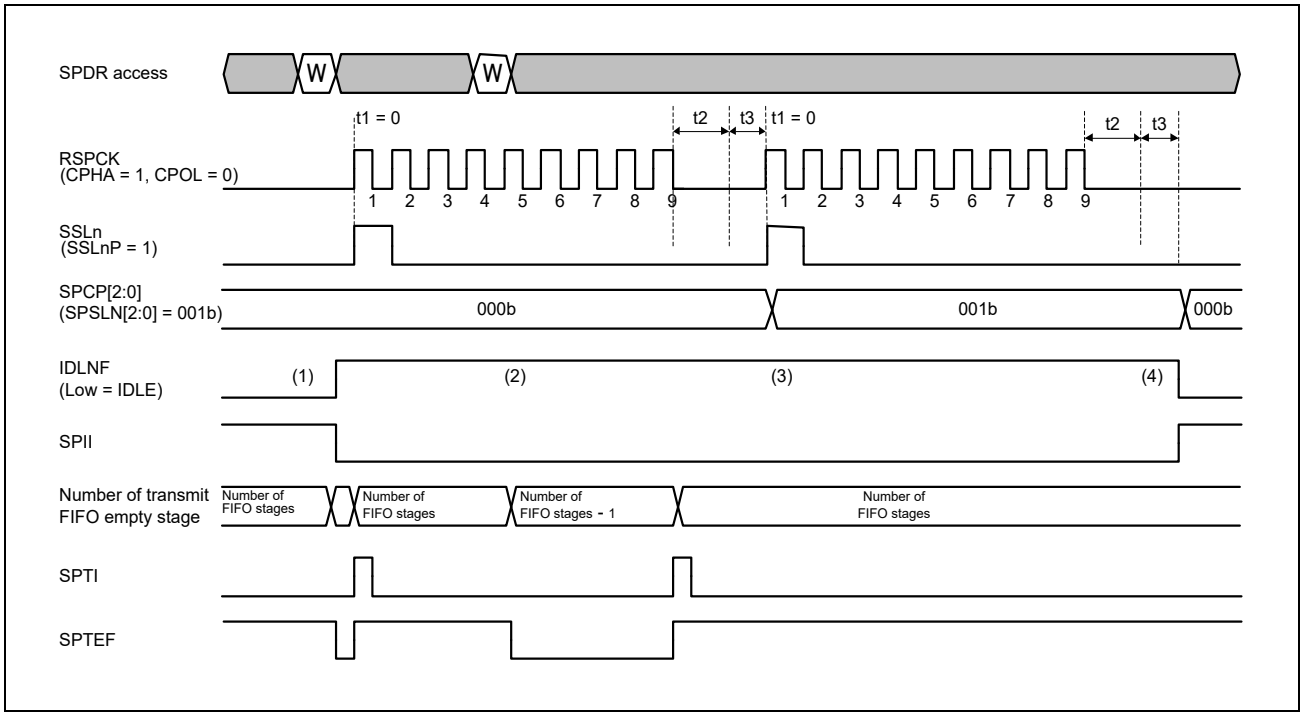


Figure 7.5-35 Example of Idle Interrupt Operation (Master Mode/TI-SSP)

(1) At the start of transmission, if the next transfer data is not set in the transmission buffer, the IDLNF flag is 0b (IDLE). Writing transmit data makes sets the IDLNF flag to 1b (BUSY). When the SPCR.SPIIE bit is set to 1b before transmit data is written, interrupt processing is required before transmission start. For this reason, set the SPIIE bit to 0b before starting transmission.

(2) After transmission has started, the IDLNF flag remains 1b (BUSY) regardless of the transmit buffer state.

(3) The SPCP[2:0] bits change the command to the next command the end of t3 cycle. When the next command is not 000b, the IDLNF flag remains unchanged even when the next transmit data has not been written.

(4) The IDLNF flag is cleared to 0b (IDLE) the end of t3 cycle because the next command is 000b and the next transmit data is not present. When SPCR.SPIIE = 1b currently, an SPII interrupt is output.

7.5.3.9 Communication End Interrupt

7.5.3.9.1 Transmit-receive/transmit-only in master mode

Refer to the description of the CENDF bit in **7.5.2.2.19 SPI Transfer FIFO Status Register (RSPIm_SPTFSR)** and **7.5.2.2.18 SPI Status Register (RSPIm_SPSR)** for the setting/clearing conditions of the communication completion flag during Transmit-Receive/Transmit-only in Master Mode.

(1) In the Motorola-SPI case

Figure 7.5-36 shows an example of communication end interrupt operation during transmit-receive/transmit master mode.

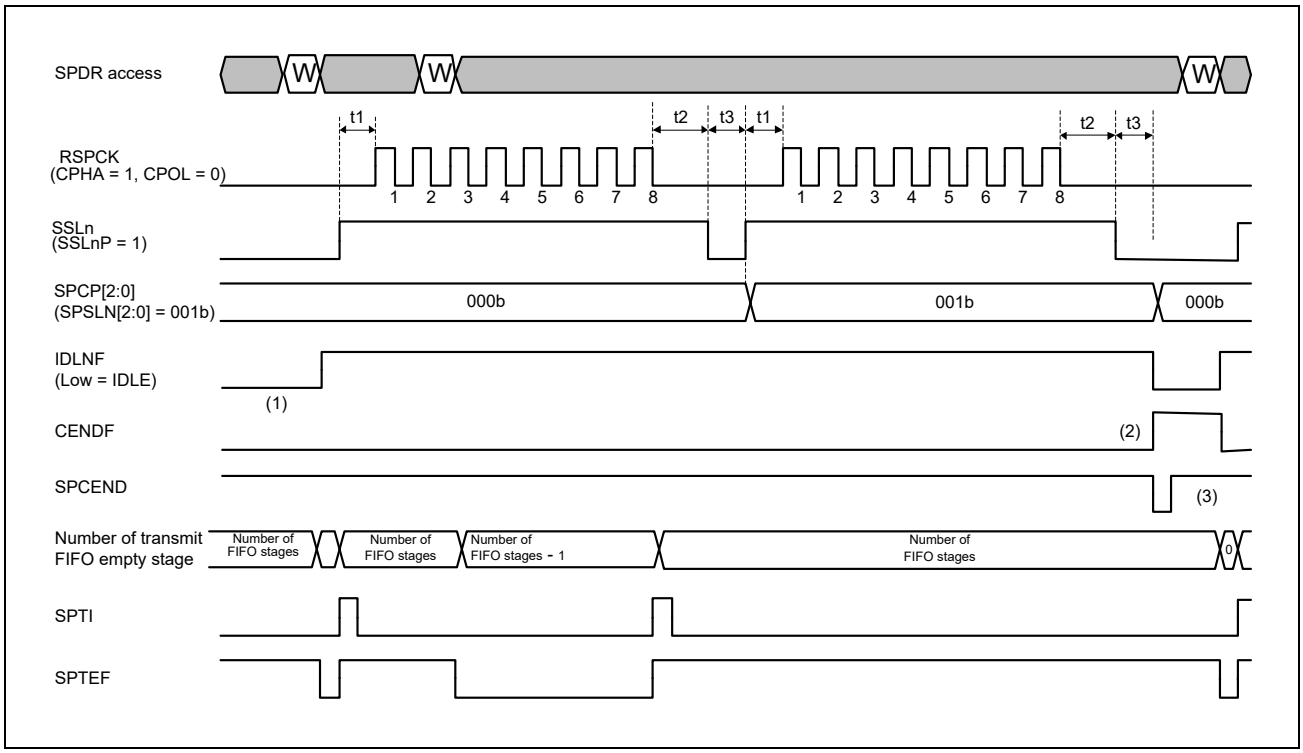


Figure 7.5-36 Example of Communication End Interrupt Operation (Transmit-Receive/Transmit-Only Master Mode/Motorola-SPI)

(1) The CENDF flag is 0 and the level of SPCEND is 1b before communication starts, and these are kept during communication.

(2) The CENDF flag is 1 (communication end) at the end of t3 cycle, because the next command is 000b and there is no next transmit data. The SPCEND interrupt is then output with 1 PCLKM cycle width if the CENDIE bit is 1b.

(3) The CENDF flag is cleared when the next transmission data is written to the transmit buffer (SPTX), or when 1b is written to the SPSRC.CENDFC bit, then the CENDF flag is 0.

(2) In the TI-SSP case

Figure 7.5-37 shows an example of communication end interrupt operation during transmit-receive/transmit-only master mode.

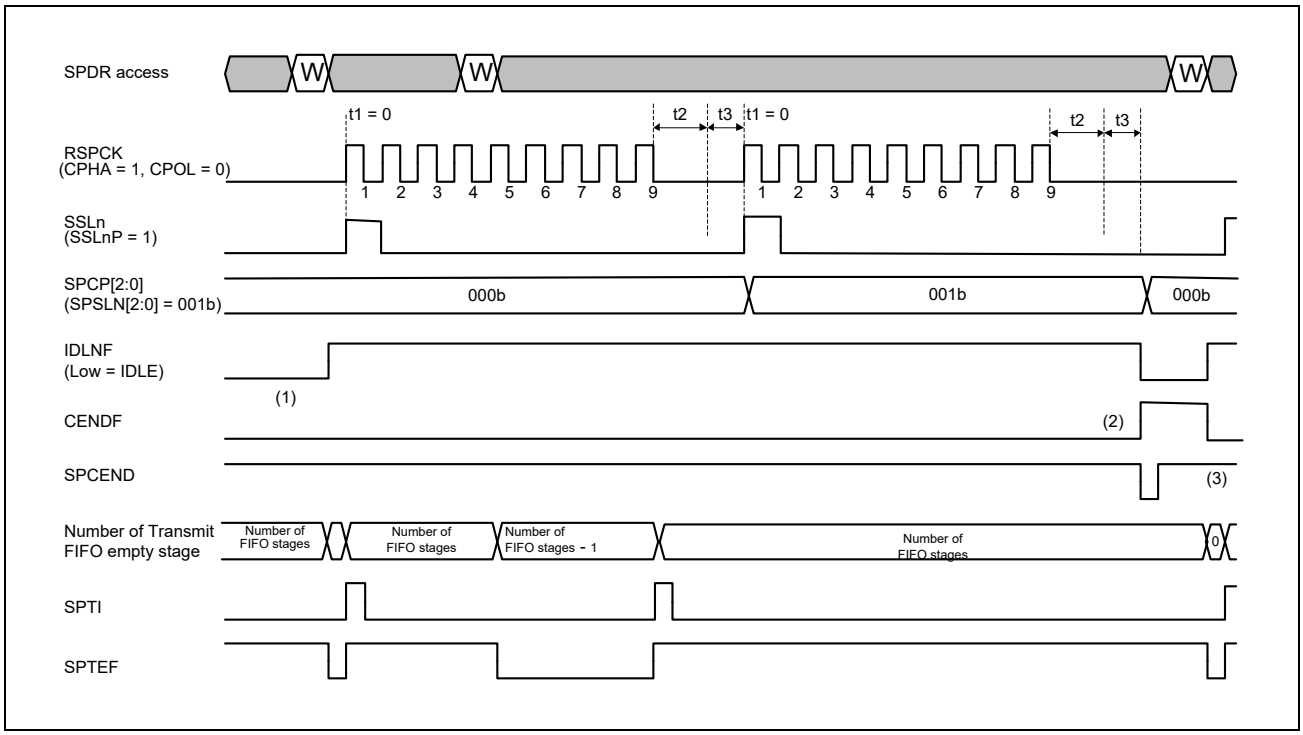


Figure 7.5-37 Example of Communication End Interrupt Operation (Transmit-Receive/Transmit-Only Master Mode/TI-SSP)

(1) The CENDF flag is 0 and the level of SPCEND is 1 before communication starts, and these are kept during communication.

(2) The CENDF flag is 1b (communication end) at the end of t3 cycle, because the next command is 000b and there is no next transmit data. The SPCEND interrupt is then output with 1 PCLKM cycle width if the CENDIE bit is 1b.

(3) The CENDF flag is cleared when the next transmission data is written to the transmit buffer (SPTX), or when 1 is written to the SPSRC.CENDFC bit, then the CENDF flag is 0.

7.5.3.9.2 Receive-only in master mode

Refer to the description of the CENDF bit in **7.5.2.2.18 SPI Status Register (RSPIm_SPSR)** for the setting/clearing conditions of the communication completion flag during Receive-only in Master Mode.

Figure 7.5-38 shows an example of communication end interrupt operation during receive-only master mode at RMFM[4:0] = 00h.

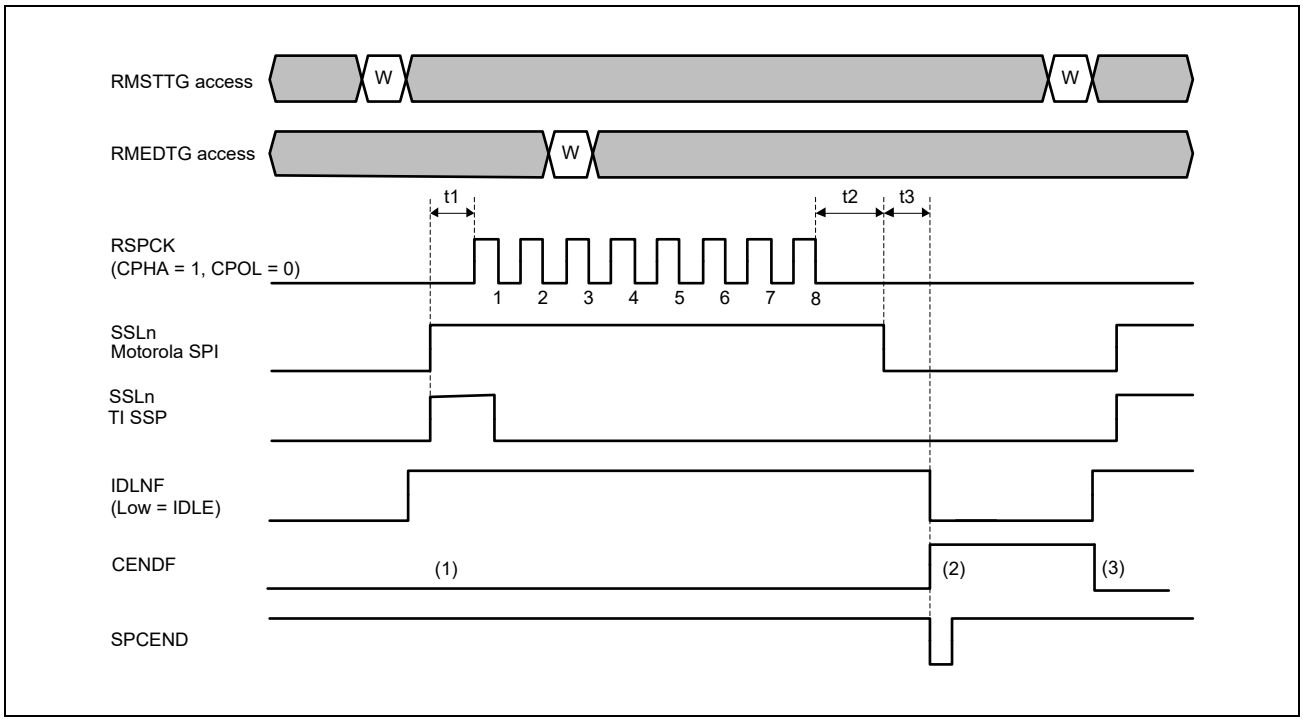


Figure 7.5-38 Example of Communication End Interrupt Operation (Receive-Only Master Mode/Motorola-SPI) at RMFM[4:0] = 00h

(1) The CENDF flag is 0 and the level of SPCEND is 1b before communication starts, and these are kept during communication.

(2) The CENDF flag is 1 (communication end) at the end of t3 cycle, by writing 1 to RMEDTG during the communication frame. The SPCEND interrupt is then output with 1 PCLKM cycle width if SPCR.CENDIE = 1.

(3) The CENDF flag is cleared when writing 1b to RMSTTG, or when 1b is written to the SPSRC.CENDFC bit, then the CENDF flag is 0.

Figure 7.5-39 shows an example of communication end interrupt operation during receive-only master mode at RMFM[4:0] ≠ 00h.

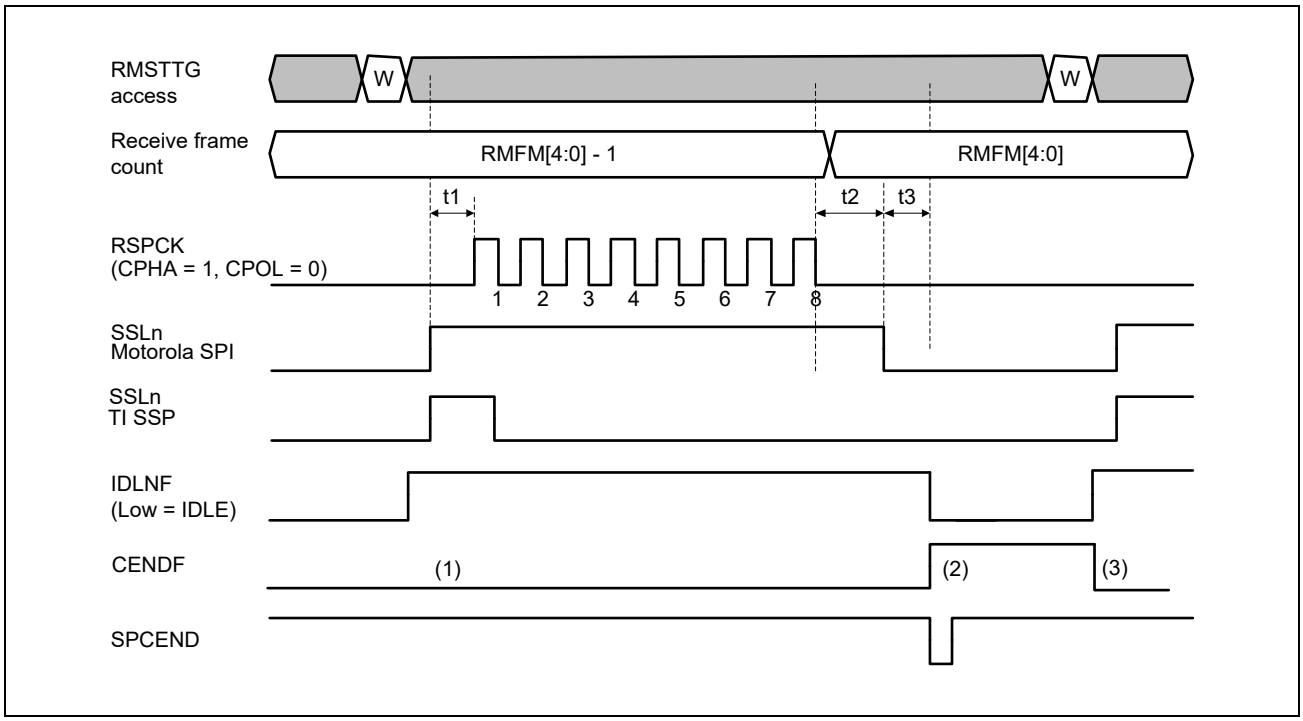


Figure 7.5-39 Example of Communication End Interrupt Operation (Receive-Only Master Mode/Motorola-SPI) at RMFM[4:0] ≠ 00h

(1) The CENDF flag is 0 and the level of SPCEND is 1b before communication starts, and these are kept during communication.

(2) The CENDF flag is 1 (communication end) at the end of t3 cycle, after receiving the number of frames set by RMFM[4:0]. The SPCEND interrupt is then output with 1 PCLKM cycle width if SPCR.CENDIE = 1b.

(3) The CENDF flag is cleared when writing 1b to RMSTTG, or when 1b is written to the SPSRC.CENDFC bit, then the CENDF flag is 0.

In slave mode operation, the output timing of the communication end interrupt is dependent on the value of the SPCR.SPMS bit (SPI mode select bit). The clear timing of the communication end interrupt is dependent on the communication mode (transmit-receive or transmit-only or receive-only).

7.5.3.9.3 Transmit-receive/transmit-only in slave mode on SPI operation (4-wire)

See the description for the CENDF bit in **7.5.2.2.18 SPI Status Register (RSPIm_SPSR)** for the setting/clearing conditions of the communication completion flag during Transmit-Receive/Transmit-only in Slave Mode (4-wire).

(1) In the Motorola-SPI case

Figure 7.5-40 shows an example of communication end interrupt operation during transmit-receive/transmit-only slave mode on SPI operation.

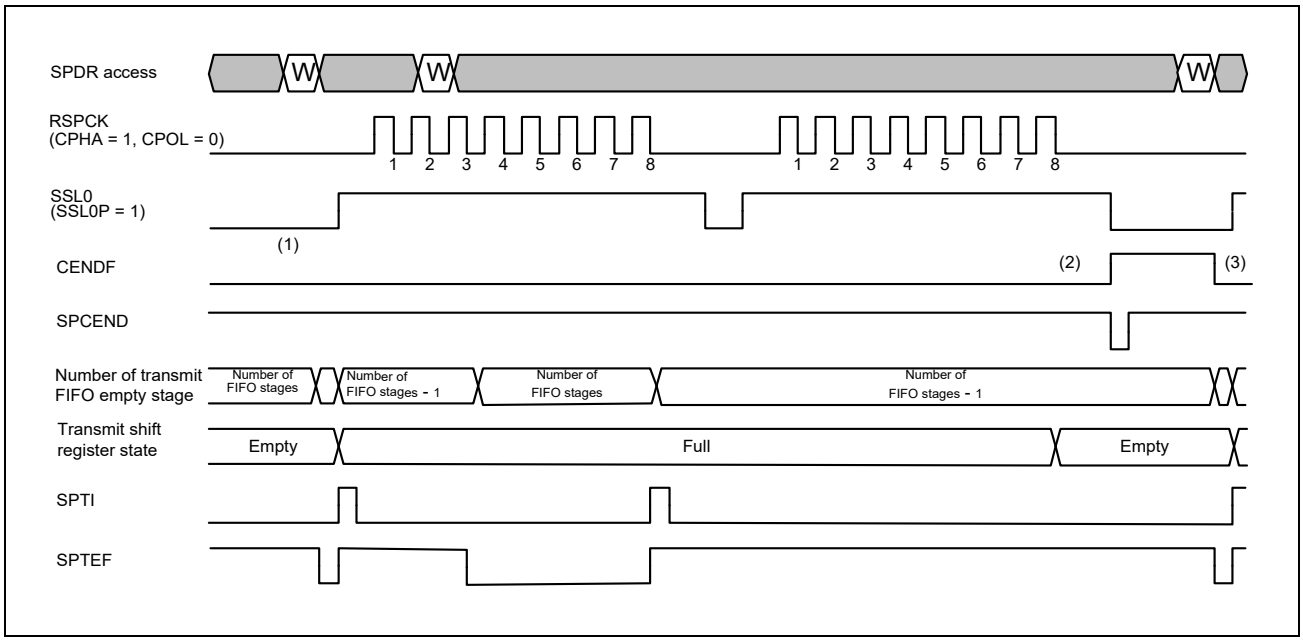


Figure 7.5-40 Example of Communication End Interrupt Operation (Transmit-Receive/Transmit-Only Slave Mode on SPI Operation/Motorola-SPI)

(1) The CENDF flag is 0 and the level of SPCEND is 1b before communication starts, and these are kept during communication.

(2) The CENDF flag is 1 (communication end) at the timing of SSL0 negate, when the next transfer data is not set in the transmit FIFO and the transmit shift register is empty. The SPCEND interrupt is then output with 1 PCLKM cycle width if SPCR.CENDIE = 1b.

(3) The CENDF flag is cleared when the next transmission data is written to the transmit buffer (SPTX), or when 1b is written to the SPSRC.CENDFC bit, then the CENDF flag is 0.

(2) In the TI-SSP case

Figure 7.5-41 shows an example of communication end interrupt operation during transmit-receive/transmit-only slave mode on SPI operation.

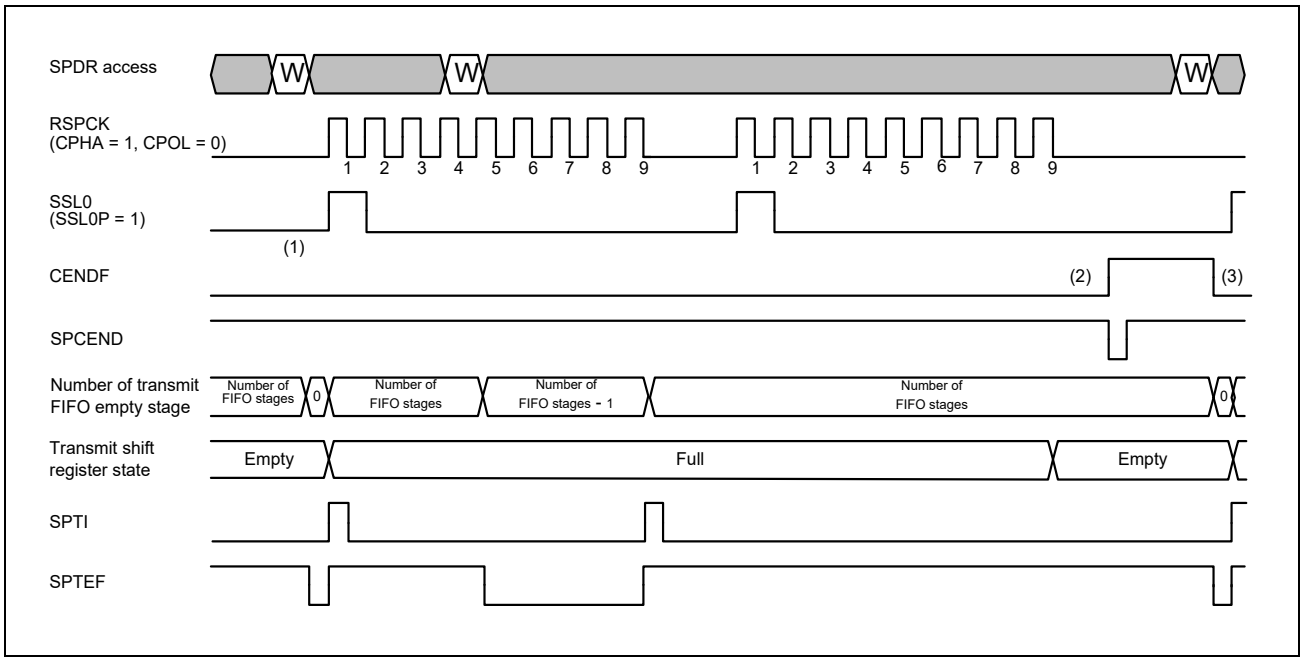


Figure 7.5-41 Example of Communication End Interrupt Operation (Transmit-Receive/Transmit-Only Slave Mode on SPI Operation/TI-SSP)

(1) The CENDF flag is 0 and the level of SPCEND is 1 before communication start, and these have kept during communication.

(2) The CENDF flag is 1 (communication end) at the RSPCK last data bit sampling, when the next transfer data is not set in the transmit FIFO and the transmit shift register is empty. The SPCEND interrupt is then outputs with PCLKM 1 cycle width if the CENDIE bit is 1.

(3) The CENDF flag is cleared when the next transmission data is written to the transmit buffer (SPTX), or when 1 is written to the SPSRC.CENDFC bit, then the CENDF flag is 0.

7.5.3.9.4 Receive only in slave mode on SPI operation (4-wire)

Refer to the description of the CENDF bit in **7.5.2.2.18 SPI Status Register (RSPIm_SPSR)** for the setting/clearing conditions of the communication completion flag during Receive-only in Slave Mode (4-wire).

(1) In the Motorola-SPI case

Figure 7.5-42 shows an example of communication end interrupt operation during receive only slave mode on SPI operation (4-wire).

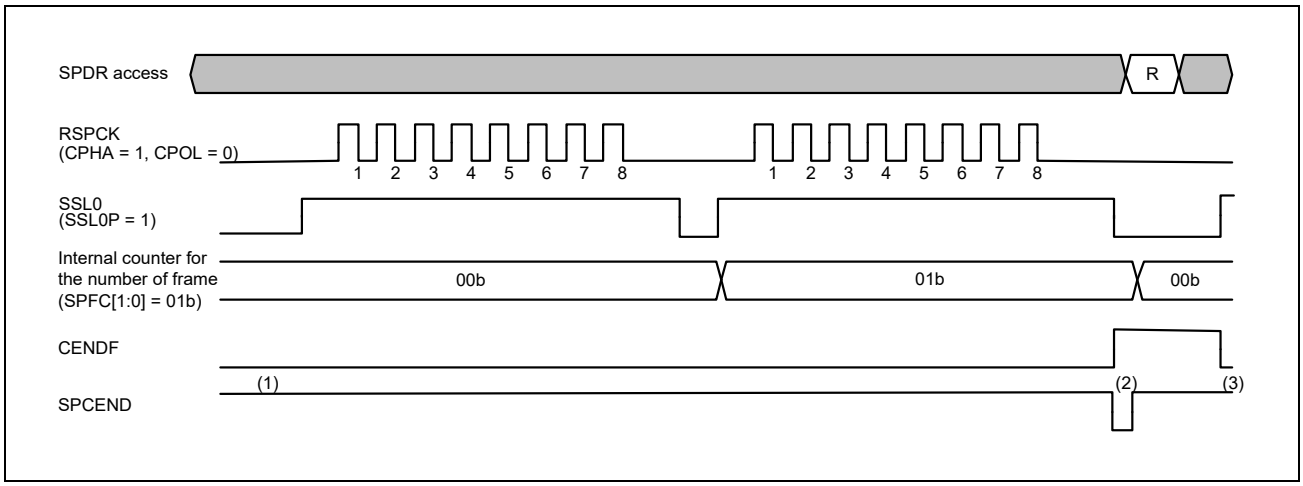


Figure 7.5-42 Example of Communication End Interrupt Operation (Receive-Only Slave Mode on SPI Operation/Motorola-SPI)

(1) The CENDF flag is 0 and the level of SPCEND is 1b before communication start, and these have kept during communication.

(2) After store the frames for the SPDCR.SPFC register set value in the receive buffer, the CENDF flag becomes 1b (communication completed) at the timing of SSL0 negation. The SPCEND interrupt is then outputs with PCLKM 1 cycle width if SPCR.CENDIE = 1b.

(3) The CENDF flag is cleared at the SSL0 assert when the next transmission start, or when 1b is written to the SPSRC.CENDFC bit, then the CENDF flag is 0.

(2) In the TI-SSP case

Figure 7.5-43 shows an example of communication end interrupt operation during receive only slave mode on SPI operation (4-wire).

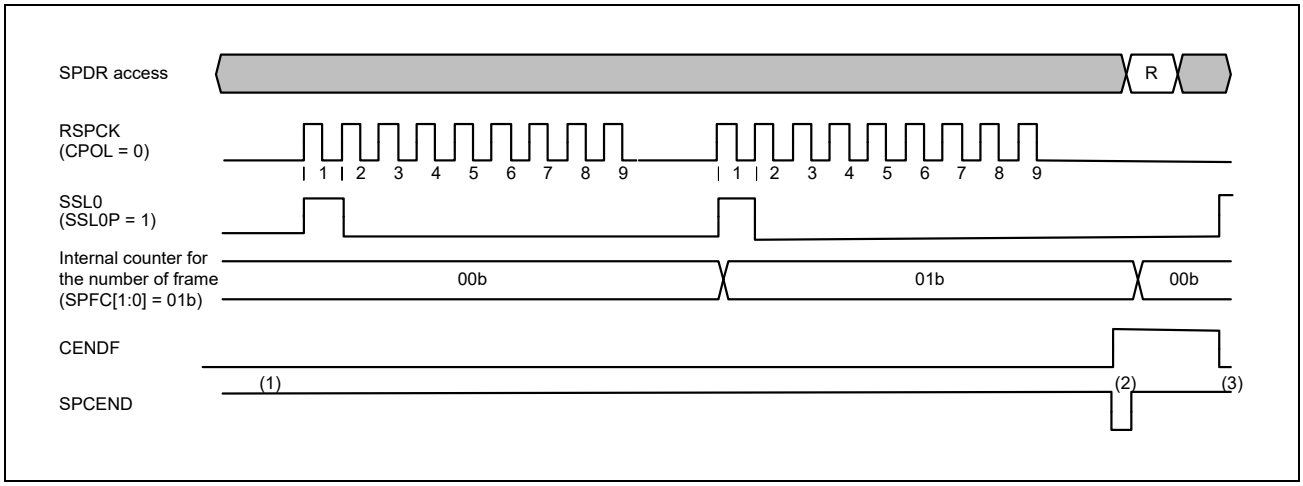


Figure 7.5-43 Example of Communication End Interrupt Operation (Receive-Only Slave Mode on SPI Operation/TI-SSP)

(1) The CENDF flag is 0 and the level of SPCEND is 1 before communication start, and these have kept during communication.

(2) The CENDF flag is 1b (communication end) at the RSPCK last data bit sampling, when the last frame transmission ends. The SPCEND interrupt is then outputs with PCLKM 1 cycle width if SPCR.CENDIE = 1b.

(3) The CENDF flag is cleared at the SSL0 assert when the next transmission start, or when 1b is written to the SPSRC.CENDFC bit, then the CENDF flag is 0.

7.5.3.9.5 Transmit-receive/transmit-only in slave mode on clock synchronous operation (3-wire)

Refer to the description of the CENDF bit in **7.5.2.2.18 SPI Status Register (RSPIm_SPSR)** for the setting/clearing conditions of the communication completion flag during Receive-only in Slave Mode (4-wire).

Figure 7.5-44 shows an example of communication end interrupt operation during transmit-receive/transmit slave mode on clock synchronous operation (3-wire).

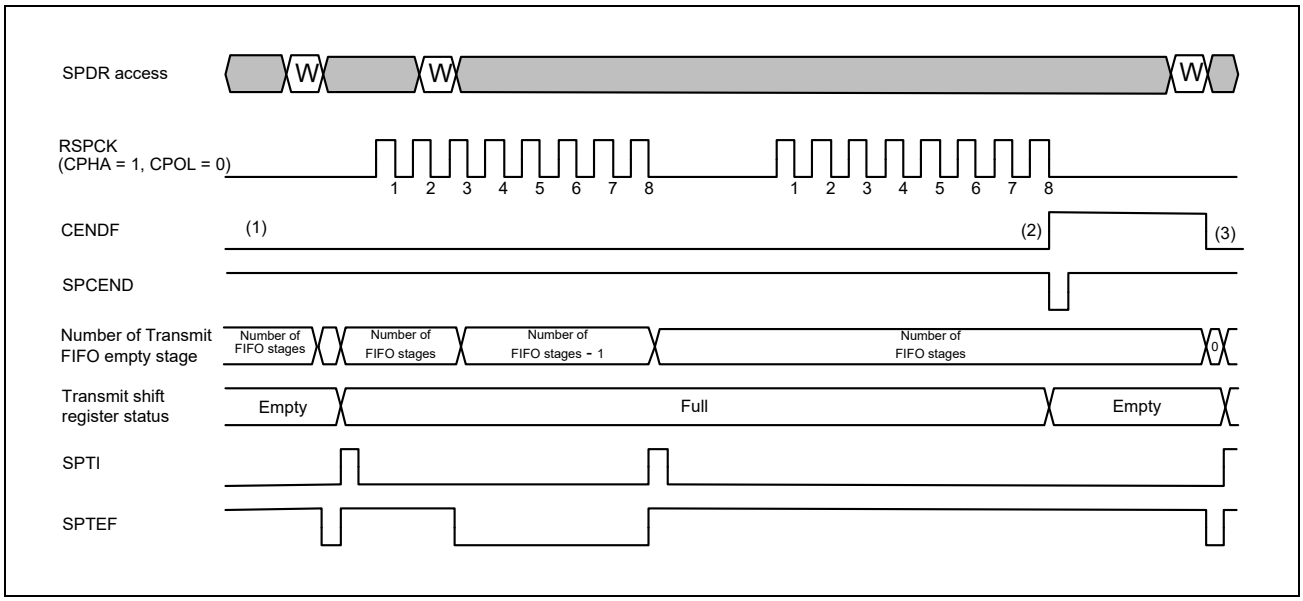


Figure 7.5-44 Example of Communication End Interrupt Operation (Receive-Only Slave Mode on SPI Operation/Motorola-SPI)

(1) The CENDF flag is 0 and the level of SPCEND is 1b before communication start, and these have kept during communication.

(2) When the next transfer data is not set in the transmit FIFO and the transmit shift register is empty, the SPCEND interrupt is then outputs with PCLKM 1 cycle width if SPCR.CENDIE = 1b.

(3) The CENDF flag is cleared when the next transmission data is written to the transmit buffer (SPTX), or when 1b is written to the SPSRC.CENDFC bit, then the CENDF flag is 0.

7.5.3.9.6 Receive only in slave mode on clock synchronous operation (3-wire)

Refer to the description of the CENDF bit in **7.5.2.2.19 SPI Transfer FIFO Status Register (RSPIm_SPTFSR)** and **7.5.2.2.18 SPI Status Register (RSPIm_SPSR)** for the setting/clearing conditions of the communication completion flag during Receive-only in Slave Mode on Clock Synchronous (3-wire).

Figure 7.5-45 shows an example of communication end interrupt operation during receive only slave mode on clock synchronous operation.

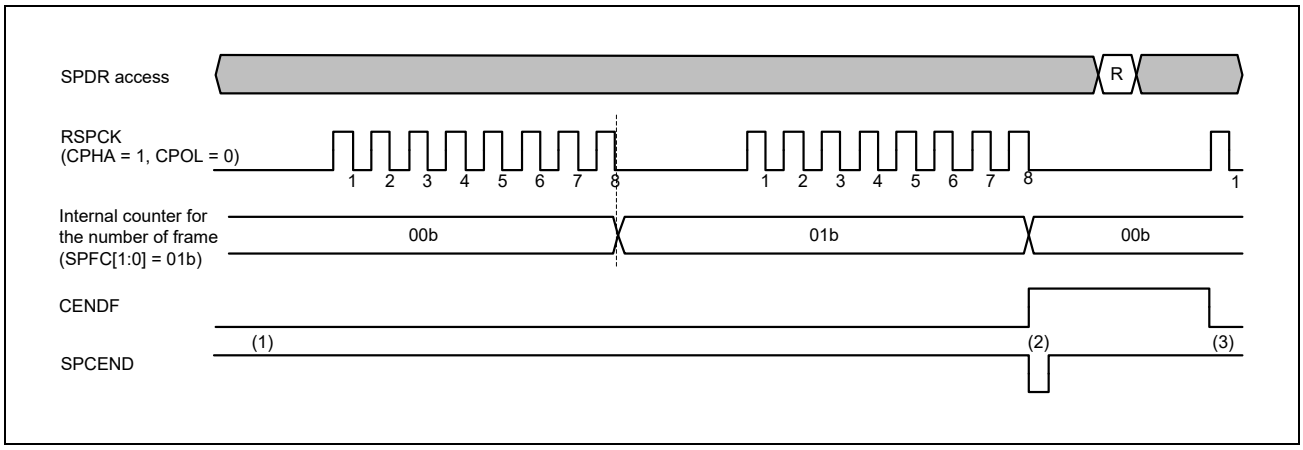


Figure 7.5-45 Example of Communication End Interrupt Operation (Receive-Only Slave Mode on Clock Synchronous Operation)

(1) The CENDF flag is 0 and the level of SPCEND is 1 before communication start, and these have kept during communication.

(2) The CENDF flag is set to 1 (communication completed) at the timing of the last data bit sampling of RSPCK in the last frame communication when the last frame of the SPDCR.SPFC bit set value is received. The SPCEND interrupt is then outputs with PCLKM 1 cycle width if SPCR.CENDIE = 1.

(3) The CENDF flag is cleared at the first edge of RSPCK for the next transmission, or when 1 is written to the SPSRC.CENDFC bit, then the CENDF flag is 0.

7.5.3.9.7 Common operation

In this Section, the operation common to each mode communication in **7.5.3.9.1 Transmit-receive/transmit-only in master mode** to **7.5.3.9.6 Receive only in slave mode on clock synchronous operation (3-wire)** is explained. When the enable of SPI communication end interrupt (CENDIE) is 0, at the time of communication completion, a flag of communication end (CENDF) is set, but no interrupt is output. However, if the enable of communication end interrupt (CENDIE) is set to 1 before clearing the flag of communication end (CENDF) while the enable of SPI function (SPE) is 1, the communication end interrupt is output.

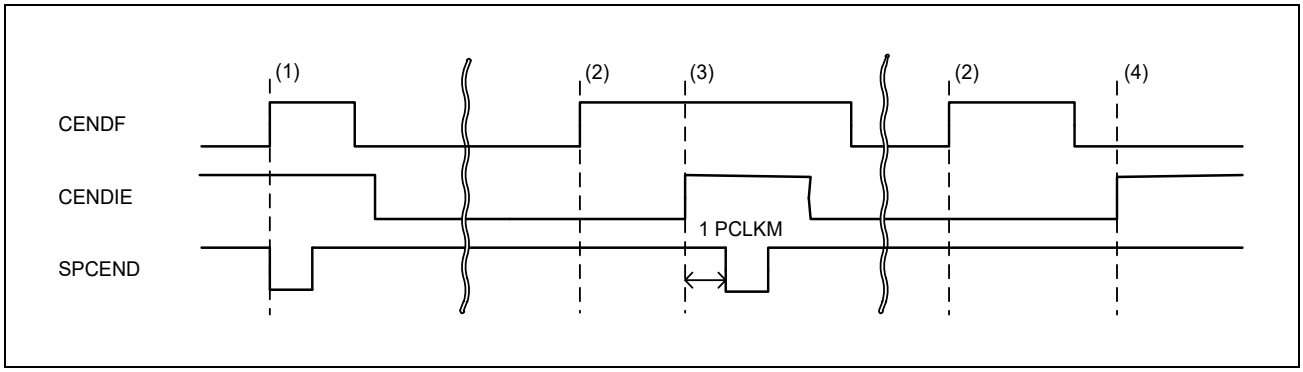


Figure 7.5-46 Example of Communication End Interrupt Operation (Enable Control)

(1) When the enable of SPI communication end interrupt (CENDIE) is 1b, at the time of communication completion, the following two are the same timing

— a flag of communication end (CENDF)
— the communication end interrupt

(2) When the enable of SPI communication end interrupt (CENDIE) is 0, at the time of communication completion, the following occurs, but no interrupt.
— a flag of communication end (CENDF)

(3) After (2), if the enable of communication end interrupt (CENDIE) is set when the enable of SPI function (SPE) and the flag of communication end (CENDF) are 1, the communication end interrupt is output after 1 PCLKM.

(4) After (2), even if the enable of communication end interrupt (CENDIE) is set when the enable of SPI function (SPE) or the flag of communication end (CENDF) is 0, the communication end interrupt is not output.

7.5.3.10 Error Detection

In normal serial transfer of SPI, data written to the transmit buffer in the SPDR register is transmitted in serial and received serial data can be read from the SPDR's receive buffer. Depending on the transmit buffer or receive buffer status when SPDR is accessed and the SPI status at the start or end of serial transfer, abnormal transfer is performed.

When some abnormal transfer operation takes place, the SPI detects it as an underrun error, overrun error, parity error, or mode fault error. **Table 7.5-10** shows the relationship between abnormal transfer operation and the SPI's error detection function.

Table 7.5-10 Abnormal Transfer Occurrence Conditions and Error Detection Function of SPI

	Transfer Occurrence Condition	SPI Operation	Error Detection
1	SPDR is written when while no empty stages in the transmit FIFO.	• Transmit buffer data is retained. • No write data is present.	None
2	SPDR is read while no data stored in receive FIFO.	Received data and previously received serial data are read.	None
3	Serial transfer starts in transmit slave mode or transmit-only slave mode, before transmit data output is ready.	• Serial transfer is suspended. • No transmit data or receive data is present. • Driving of the MISO output signal is stopped. • SPI function is disabled.	Underrun error
4	Serial transfer ends when data is stored in the receive FIFO for the number of FIFO stages.	• Receive FIFO data is retained. • No serial receive data is present.	Overrun error
5	Incorrect parity bit has been received with the parity function enabled in following mode. • Transmit/receive-only master mode • Transmit/receive slave mode • Receive-only slave mode	The parity error flag is asserted.	Parity error
6	[In the Motorola-SPI case] The SSL0 input signal is negated during serial transfer in slave mode.	• Serial transfer is suspended. • No transmit data or receive data is present. • Driving of the MISO output signal is stopped. • SPI function is disabled.	Mode fault error
7	[In the TI-SSP case] The SSL0 input signal is asserted during serial transfer in slave mode.	• Serial transfer is suspended. • No transmit data or receive data is present. • Driving of the MISO output signal is stopped. • SPI function is disabled.	Mode fault error

The SPI does not detect any error for the operation shown in No.1 in **Table 7.5-10**. To prevent missing of data during SPDR writing, write data to SPDR upon a transmit buffer empty interrupt request or while the SPSR.SPTEF flag is 1b.

The SPI does not detect any error also for the operation shown in No.2 in **Table 7.5-10**. To prevent reading of unnecessary data, read data from SPDR upon a receive buffer full interrupt request or while the SPSR.SPRF flag is 1b.

The underrun error shown in No.3 in **Table 7.5-10** is described in **7.5.3.10.4 Underrun error**.

The overrun error shown in No.4 in **Table 7.5-10** is described in **7.5.3.10.1 Overrun error**.

The parity error shown in No.5 in **Table 7.5-10** is described in **7.5.3.10.2 Parity error**.

The mode fault error shown in No.6 to No.9 in **Table 7.5-10** is described in **7.5.3.10.3 Mode fault error**.

For transmission and reception interrupts, see **7.5.3.7 Transmission buffer empty/receive buffer full interrupt**.

7.5.3.10.1 Overrun error

When serial transfer ends while the receive buffer in the SPDR register is full, the SPI detects an overrun error and sets the OVRF flag in SPSR to 1b. While the OVRF flag is 1b, the SPI does not copy shift register data to the receive buffer. Therefore, data before an error occurs is retained in the receive buffer. To clear the OVRF flag in SPSR to 0b, issue a system reset or 1b is written to the SPSRC.OVRFC bit.

Figure 7.5-47 shows operation of the SPRF flag and the OVRF flag in SPSR. “SPSR access” and “SPDR access” described in **Figure 7.5-47** show an access to SPSR and SPDR respectively. “W” shows a write cycle and “R” shows a read cycle. In the example in **Figure 7.5-47**, the SPI performs 8-bit data serial transfer with the settings of SPCMD.CPHA = 1b and SPCMD.CPOL = 0b. Numbers under the RSPCK waveform show the number of RSPCK cycles (number of transfer bits).

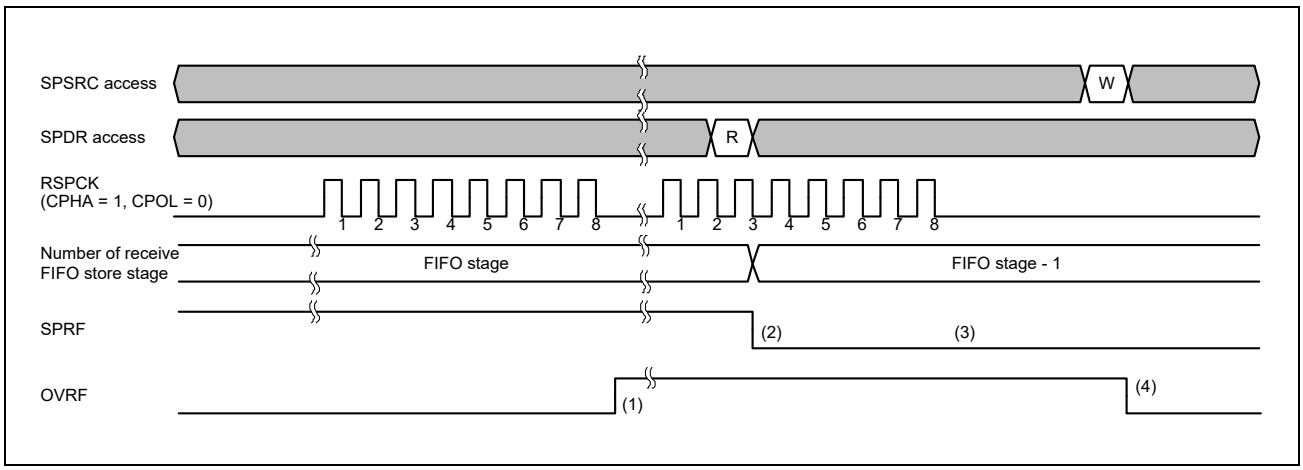


Figure 7.5-47 Example of Operation of SPRF and OVRF Flags

The following describes operation of flags at timings (1) to (4) in the figure above.

(1) When serial transfer ends while data is stored for the number of FIFO stages, the SPI detects an overrun error and sets the OVRF flag to 1b. The SPI does not copy shift register data to the receive buffer. The SPI does not detect a parity error even when SPPE = 1b. In master mode, the SPI copies the value of pointer to the SPCMD register to the SPSSR.SPECM[2:0] bits.

(2) The SPI can read receive buffer data by reading SPDR. At this time, the SPRF flag is cleared to 0 at the last access when the received data is read from SPDR in one processing routine using DMAC.

(3) When serial transfer ends while the OVRF flag is 1b (overrun error), the SPI does not copy shift register data to the receive buffer (SPRF flag remains 0) and no SPI receive buffer full interrupt is generated. The SPI does not detect a parity error even when SPPE = 1b. In master mode, the SPI does not update the SPSSR.SPECM[2:0] bits. When the SPI does not copy received data from the shift register to receive buffer while an overrun error is present, the

SPI also determines that the shift register is empty at the end of serial transfer, enabling data transfer from the transmit buffer to the shift register.

(4) When 1 is written to the SPSRC.OVRFC bit.

Whether an overrun error is present can be checked by reading SPSR or by reading an SPI error interrupt and SPSR. To perform serial transfer without using an SPI error interrupt, read SPDR and then immediately read SPSR to quickly detect an overrun error. When the SPI is used in master mode, the pointer value to SPCMD when an error is present can be checked by reading the SPSSR.SPECM bits. When an overrun error occurs and the OVRF flag is set to 1, successful reception is disabled until the OVRF flag is cleared to 0.

When the RSPCK auto-stop function is enabled in master mode, an overrun error does not occur. **Figure 7.5-48** and **Figure 7.5-49** show the clock stop waveform when a serial transfer continues while the reception buffer is full in master mode.

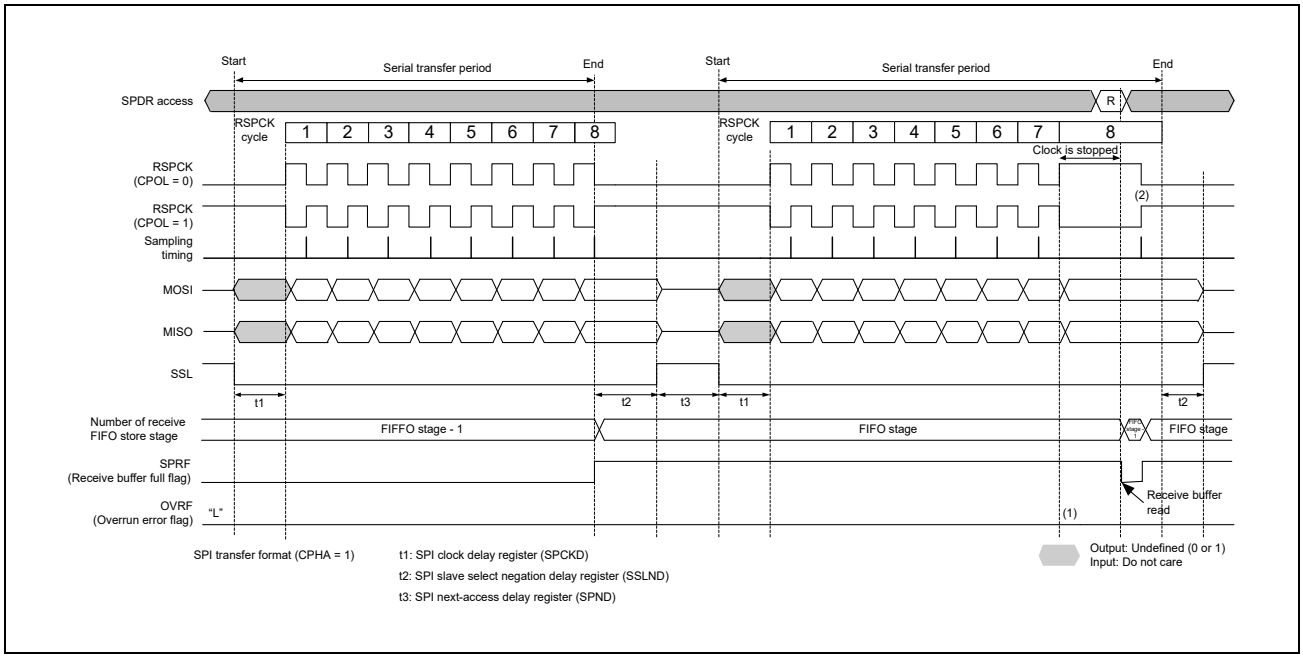


Figure 7.5-48 Clock Stop Waveform when Serial Transfer Continues with Data is Stored for the Number of FIFO Stages in Master Mode (CPHA = 1b)

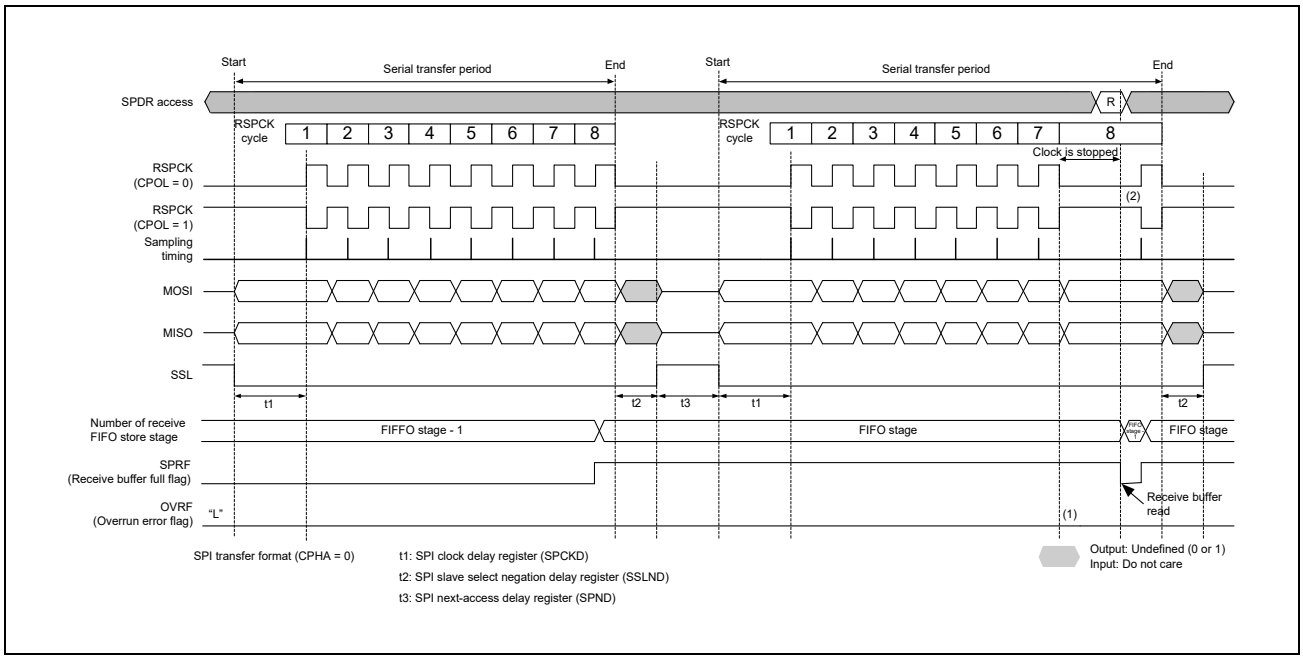


Figure 7.5-49 Clock Stop Waveform when Serial Transfer Continues with Data is Stored for the Number of FIFO Stages in Master Mode (CPHA = 0b)

The following describes operation of flags at timings (1) and (2) in the figure above.

(1) While data is stored in the receive FIFO for the number of FIFO stages, the RSPCK clock is deactivated and no overrun error occurs.

(2) Receive buffer data can be read by reading SPDR during clock stop. After the receive buffer data has been read, the RSPCK clock restarts.

Overrun error does not occur when RSPCK automatic stop function is enabled for transfer with no delay of between frames during burst transfer in master mode. The **Figure 7.5-50** and the **Figure 7.5-51** show the clock stop waveform, when there is no delay between frames at burst transfer and the serial transfer continues in the data is stored in the receive FIFO for the number of FIFO stages.

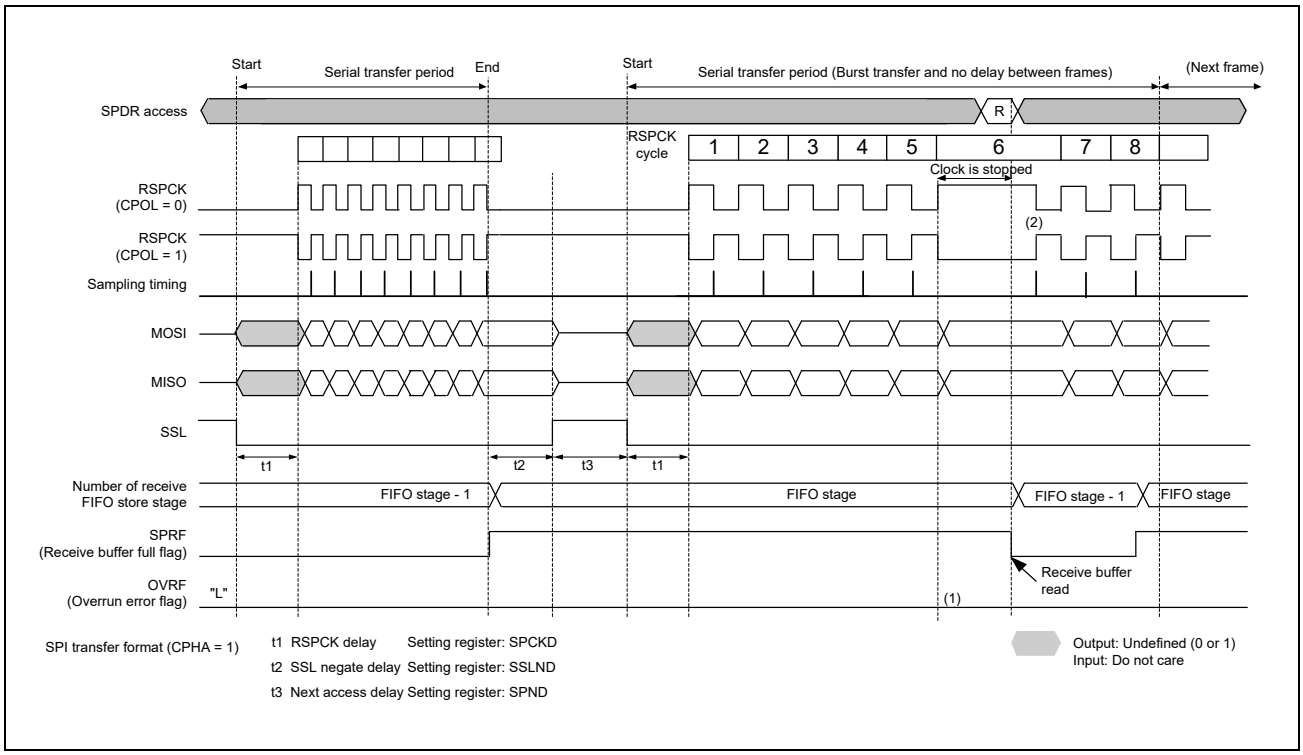


Figure 7.5-50 Clock Stop Waveform when Serial Transfer Continues in the Receive Buffer Full with Data is Stored for the Number of FIFO Stages in Master Mode (at Burst Transfer and No Delay between Frames CPHA = 1b)

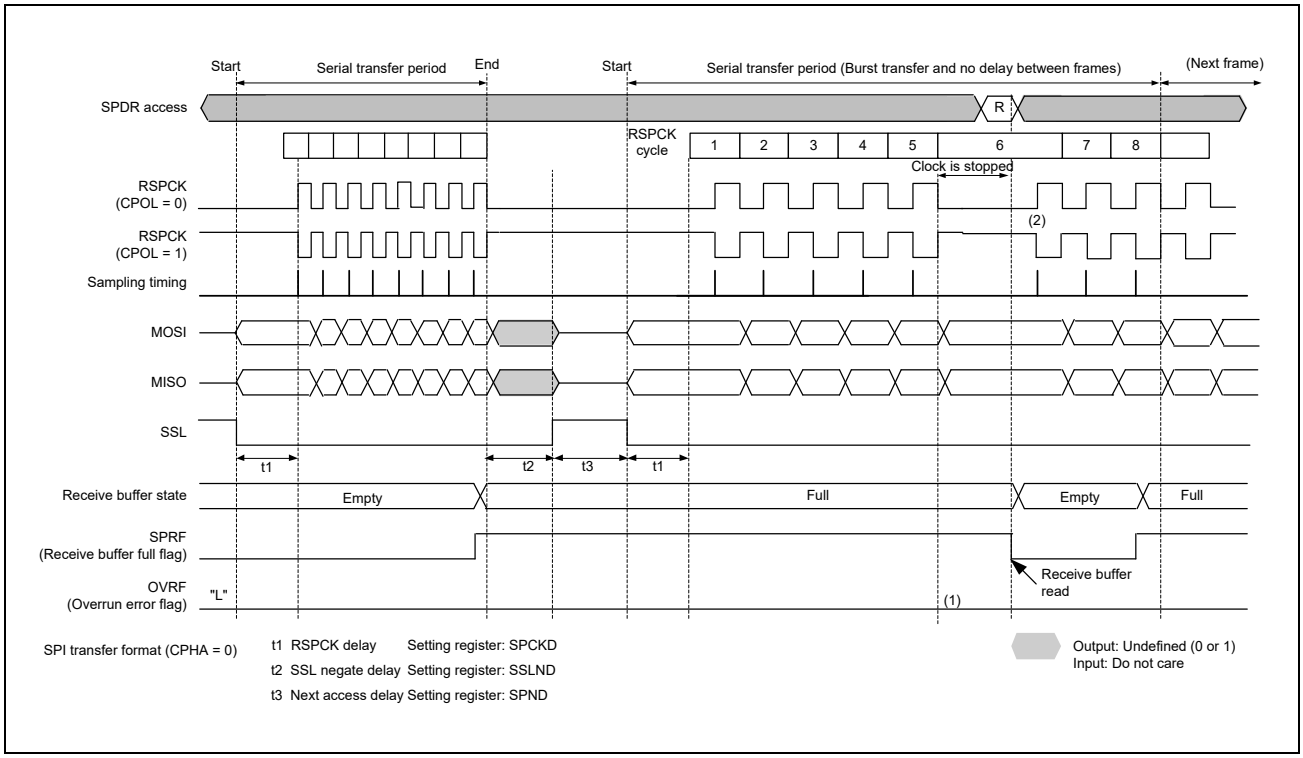


Figure 7.5-51 Clock Stop Waveform when Serial Transfer Continues with Data is Stored for the Number of FIFO Stages in Master Mode (at Burst Transfer and No Delay between Frames CPHA = 0b)

The following describes operation of flags at timings (1) and (2) in the figure above.

(1) While the data is stored for the number of FIFO stages, the RSPCK clock is deactivated and no overrun error occurs.

(2) Receive buffer data can be read by reading SPDR during clock stop. After the receive buffer data has been read, the RSPCK clock restarts.

7.5.3.10.2 Parity error

After transfer in transmit-receive or receive-only master mode, transmit-receive slave mode or receive only slave mode while the SPCR.SPPE = 1b, the SPI checks occurrence of a parity error. When the SPI detects a parity error in received data, the PERF flag in the SPSR register is set to 1b. While the OVRF flag is 1b, the SPI does not copy shift register data to the receive buffer. Therefore, parity error in received data is not detected. To clear the PERF flag in SPSR to 0b, issue a system reset or write 1b to the SPSRC.PERFC bit.

Figure 7.5-52 shows operation of the OVRF and PERF flags in SPSR. “SPSR access” described in **Figure 7.5-52** shows an access to SPSR. “W” shows a write cycle and “R” shows a read cycle. In the example in **Figure 7.5-52**, the SPI performs transmit-receive serial communication while SPCR.SPPE = 1b. The SPI performs 8-bit data serial transfer with the settings of SPCMD.CPHA = 1b and SPCMD.CPOL = 0b. Numbers under the RSPCK waveform show the number of RSPCK cycles (number of transfer bits).

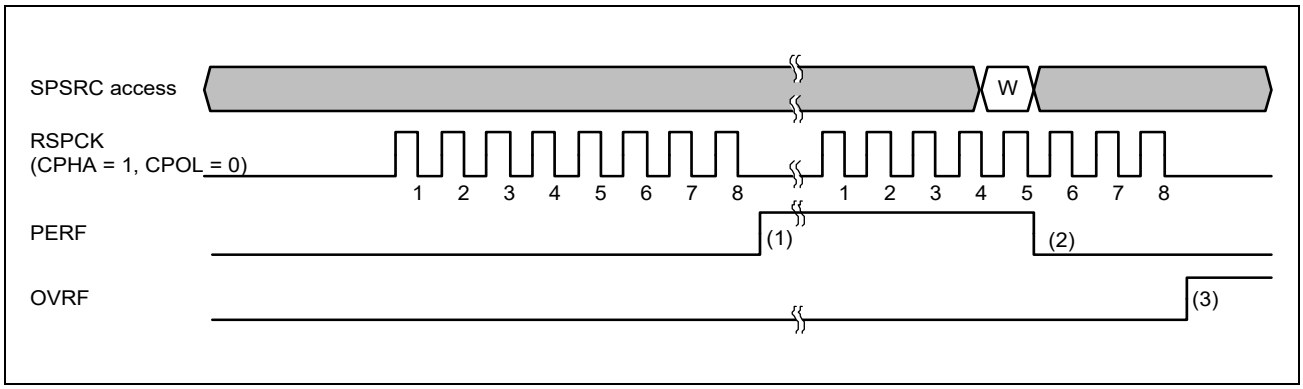


Figure 7.5-52 Example of Operation of OVRF and PERF Flags

The following describes operation of flags at timings (1) to (3) in the figure above.

(1) When the SPI does not detect an overrun error and terminates the serial transfer, the SPI copies shift register data to the receive buffer. When the SPI checks the received data and detects a parity error at this time, the PERF flag is set to 1b. In master mode, the SPI copies the value of pointer to the SPCMD register to the SPSSR.SPCM [2:0] bits.

(2) When 1 is written to the SPSRC.PERFC bit.

(3) When the SPI detects an overrun error and terminates the serial transfer, the SPI does not copy shift register data to the receive buffer and does not detect a parity error.

Whether a parity error is present can be checked by reading SPSR or by reading an SPI error interrupt and SPSR. To use an SPI error interrupt, set the SPCR.SPEIE bit to 1b. To perform serial transfer without using an SPI error interrupt, read SPDR to quickly detect a parity error. When the SPI is used in master mode, the pointer value to SPCMD when an error is present can be checked by reading the SPSSR.SPECM[2:0] bits.

7.5.3.10.3 Mode fault error

When the SPCR.MSTR = 0b, the SPI operates in slave mode. When SPCR.SPMS = 0b and SPCR.MODFEN = 1b in slave mode, if the SSL0 input signal is negated during the serial transfer period (from valid data drive start to final valid data latch), the SPI detects a mode fault error while any of the following 2 conditions is met.

(1) In the Motorola-SPI case

When the SSL0 input signal is negated while serial data transfer.

(2) In the TI-SSP case

When the SSL0 input signal is asserted while serial data transfer. However, during burst transfer, no error is detected even if the SSL0 input signal is asserted during the last bit of frame.

When the SPI detects a mode fault error, it stops driving output signals and clears the SPCR.SPE bit.

When the SPE bit is cleared, the SPI function is disabled (as described in **7.5.3.12 SPI Initialization**).

Whether a mode fault error is present can be checked by reading SPSR or by reading an SPI error interrupt and SPSR. To detect a mode fault error without using an SPI error interrupt, poll SPSR.

While the MODF flag = 1b, the SPI ignores writing 1b to the SPE bit. To enable the SPI function after a mode fault error is detected, clear the MODF flag to 0b without fail.

7.5.3.10.4 Underrun error

While the SPI is operating in slave mode (SPCR.MSTR = 0b) and the communication mode select SPCR.TXMD[1:0] bits are set to 00b or 01b, if serial transfer is started before transmit data output is ready with the SPCR.SPE bit set to 1b (SPI function enabled), the SPI detects an underrun error and sets the MODF and UDRF flags in the SPSR register to 1b.

When the SPI detects an underrun error, it stops driving output signals and clears the SPE bit to 0b. When the SPE bit is cleared, the SPI function is disabled (as described in **7.5.3.12 SPI Initialization**)

Whether an underrun error is present can be checked by reading SPSR or by reading an SPI error interrupt and SPSR. To detect an underrun error without using an SPI error interrupt, poll SPSR.

While the MODF flag = 1b, the SPI ignores writing 1b to the SPE bit. To enable the SPI function after an underrun error is detected, clear the MODF flag to 0 without fail.

7.5.3.11 Received Data Ready Detection

When SPCR.TXMD[1:0] = 00b, 01b, or 11b, and SPDRCR.SPDRC[7:0] ≠ 00h, after receiving data in the receive FIFO during communication (SPE = 1), SPDR.SPDRF flag is set to 1 when the received data is not stored even after the number of received FIFOs ≤ the threshold value and the value set in SPDRCR.SPDRC[7:0] has elapsed.

When the receive data ready is detected, the interrupt can be selected as SPRI or SPEI with the SPCR.SPDRES bit. **Figure 7.5-53** shows an example of reception data ready detection operation.

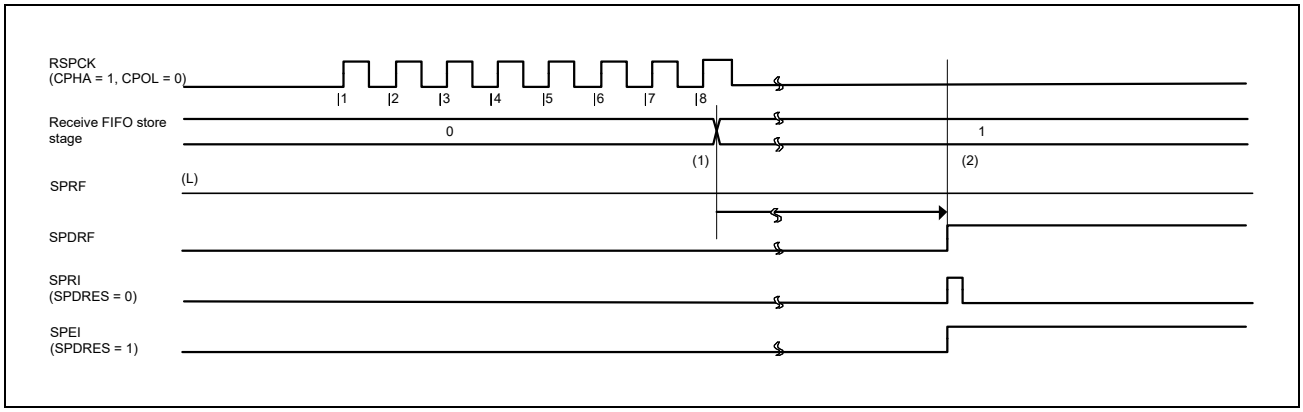


Figure 7.5-53 Received Data Ready

The following describes operation at the timings indicated by (1) and (2) in the figure.

(1) Store the received data in the receive FIFO. SPRF is 0b, because receive FIFO store stage ≤ RTRG.

(2) Set SPDRF and assert SPRI or SPEI because there is no writing to the receive FIFO for the amount of SPDRC[7:0] set from (1).

7.5.3.12 SPI Initialization

When 0b is written to the SPCR.SPE bit or when the SPI clears the SPE bit to 0b after it detects a mode fault error or an underrun error, the SPI disables the SPI function and initializes a part of module functions. Furthermore, the SPI initializes all module functions when a system reset occurs. The following describes initialization by clearing the SPCR.SPE bit and by using a system reset.

7.5.3.12.1 Initialization by clearing the SPE bit

When the SPCR.SPE bit is cleared to 0b, the SPI performs the following initialization:

- Suspends the ongoing serial transfer.
- Stops driving output signals (Hi-Z) in slave mode.
- Resets the internal SPI state.
- SPSR.SPTEF flag = 1b.

In initialization by clearing the SPE bit, control bits in SPI are not initialized. For this reason, re-setting the SPE bit to 1b activates the SPI with the pre-initialization transfer mode.

Values of the CENDF flag, the SPRF flag, OVRF flag, MODF flag, PERF flag, and UDRF flag in SPSR are not initialized. The SPSSR register value is not initialized either. For this reason, it is possible to read receive buffer data and check the communication completion status and the error status during SPI transfer also after the SPI is initialized.

SPSR.SPTEF flag is initialized to 1b. Therefore, if the SPCR.SPTIE bit is set to 1 after the SPI is initialized, an SPI transmit buffer empty interrupt is generated. When the CPU initializes the SPI, write 0b to the SPE bit and the SPTIE bit to disable an SPI transmit buffer empty interrupt.

7.5.3.12.2 System reset

In initialization by using a system reset, all SPI control bits, status bits, and data register are initialized in addition to the initialization described in **7.5.3.12.1 Initialization by clearing the SPE bit** to completely initialize the SPI.

7.5.3.13 SPI Operation

7.5.3.13.1 Master mode operation

(1) Starting a serial transfer

When data is written to the SPDR register while the next transfer data is not set in the transmit FIFO, the SPI updates the transmit buffer (SPTXn, n = 0 to 15) data in SPDR.

While the shift register is empty, the SPI copies transmit buffer data to the shift register to start serial transfer. After the SPI copies transmit data to the shift register, it changes the shift register status to full. Upon completion of serial transfer, the SPI changes the shift register status to empty. The shift register status cannot be monitored.

For details about the SPI transfer format, see **7.5.3.5 Transfer Format (Frame Format)**. The SSL output signal polarity depends on the set SSLP register value.

(2) Terminating a serial transfer

[Except Receive-only in Master Mode]

After the SPI detects the RSPCK edge corresponding to the final sampling timing regardless of the SPCMD.CPHA bit value, the SPI terminates serial transfer. When the number of data stored in the receive FIFO < the number of FIFO stages, data is copied from the shift register to the receive buffer in the SPDR register after serial transfer.
The final sampling timing varies depending on the transfer data bit length. The data length of the SPI in master mode depends on the set value of the SPCMD.SPB[4:0] bits. The SSL output signal polarity depends on the set SSLP register value. For details about the SPI transfer format, see **7.5.3.5 Transfer Format (Frame Format)**.

[Receive-only in Master Mode]

When any of the following 2 conditions is met, then SPI terminating the serial transfer.

- After the SPI detects the RSPCK edge corresponding to the final sampling timing regardless of the SPCMD.CPHA bit value, the SPI terminates serial transfer.
- When writing SPCRRM.RMEDTG = 1b during the serial transfer period, the SPI terminates the serial transfer.

When the number of data stored in the receive FIFO < the number of FIFO stages, data is copied from the shift register to the receive buffer in the SPDR register after serial transfer.

The final sampling timing varies depending on the transfer data bit length. The data length of the SPI in master mode depends on the set value of the SPCMD.SPB[4:0] bits. The SSL output signal polarity depends on the set SSLP register value. For details about the SPI transfer format, see **7.5.3.5 Transfer Format (Frame Format)**.

(3) Sequence control

The transfer format in master mode is determined as follows.

The transfer format in master mode is determined by the SPSCR register, SPCMDm (m = 0 to 7) register, SPBR register, SPCKD register, SSLND register, and SPND register.

The SPSCR register is used to determine the sequence configuration for serial transfer to be performed by the SPI in master mode. The SSL output signal value, MSB first/LSB first, data length, a part of bit rate settings, RSPCK polarity and phase, SPCKD enable/disable, SSLND enable/disable, and SPND enable/disable are set in SPCMD0 to SPCMD7.

A part of bit rate settings is set in SPBR, the SPI clock delay value is set in SPCKD, the SSL negation delay value is set in SSLND, and the access delay value is set in SPND.

The SPI configures the sequence that structures a part of or whole of SPCMD0 to SPCMD7 according to the sequence length specified in SPSCR. The SPI has a pointer to SPCMD that configures the sequence. This pointer value can be checked by reading the SPSSR.SPCP[2:0] bits. When the SPE bit in the SPCR register is set to 1 to enable the SPI function, the SPI sets the pointer to commands in SPCMD0 and applies the SPCMD0 setting to the transfer format at the start of serial transfer. The SPI increments the pointer the next-access delay period ends in each data transfer. When the serial transfer corresponding to the final command that configures the sequence is completed, the SPI sets the pointer in SPCMD0. Thus sequence is repeatedly executed.

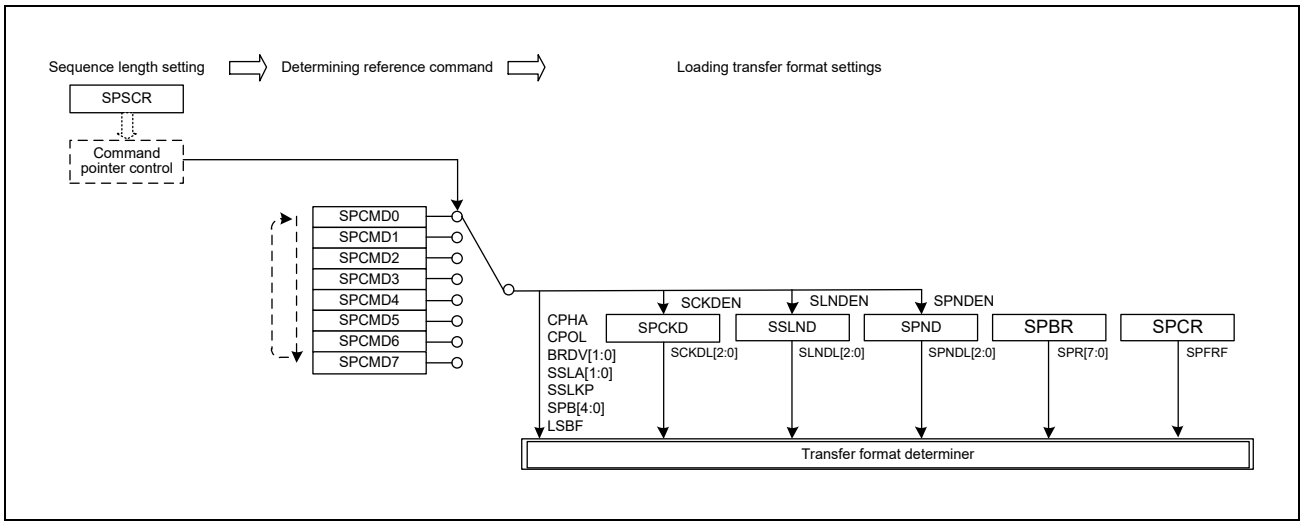


Figure 7.5-54 Determining Serial Transfer Method in Master Mode

In this section, a frame consists of data (SPDR) and configuration (SPCMDm).

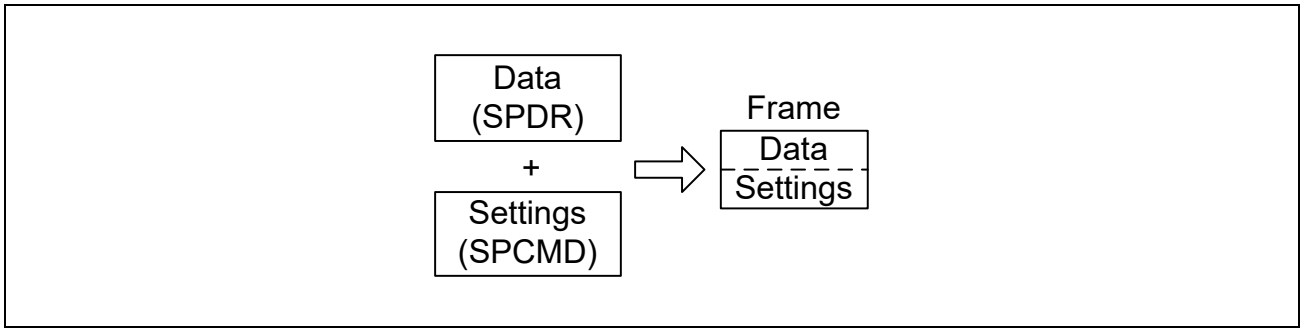


Figure 7.5-55 Conceptual Diagram of Frame

Figure 7.5-56 shows the correspondence between the SPI command register and transmit buffer (SPTXn, n = 0 to 15)/receive buffer (SPRXn, n = 0 to 15) in the sequence operation.

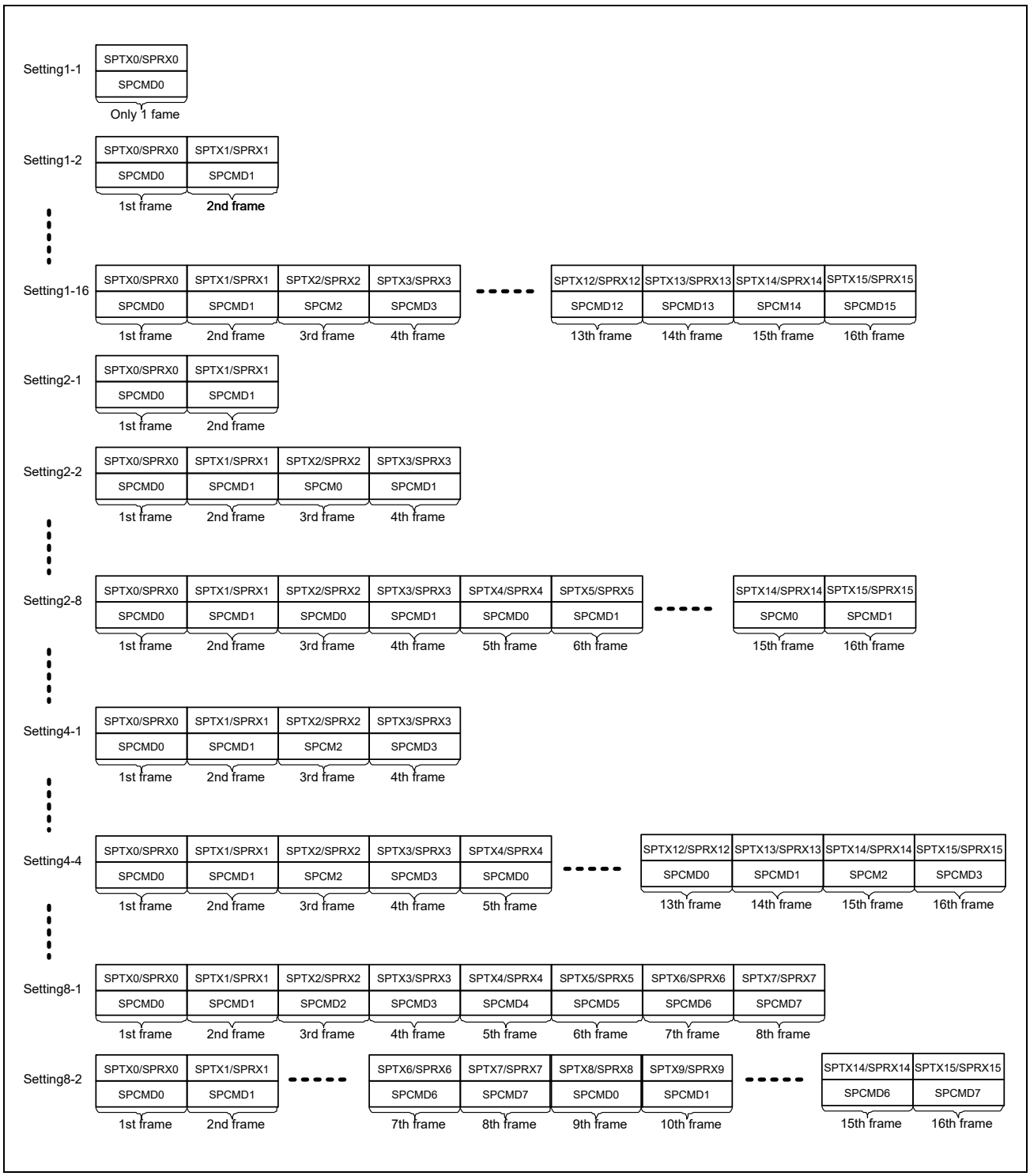


Figure 7.5-56 Correspondence between SPI Command Register and Transmit/Receive Buffers in Sequence Operation

(4) Burst transfer

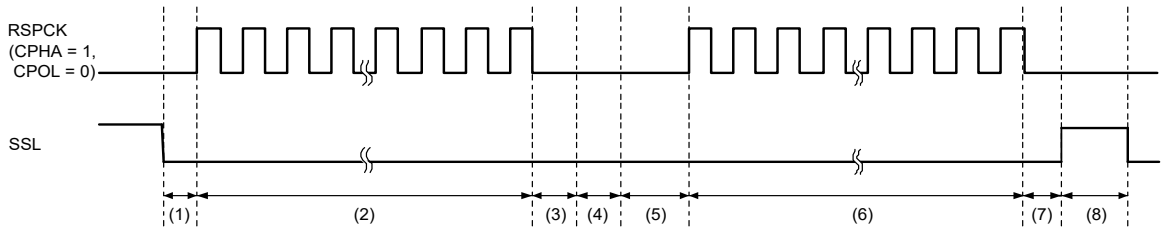
This section describes burst transfer during transmit-receive/transmit-only operation.

[In the Motorola-SPI case]

When the SPCMD.SSLKP bit, which the SPI references in the current serial transfer, is 1b, the SPI retains the SSL signal level during serial transfer until the SSL signal assertion of the next serial transfer starts. When the SSL signal level in the next serial transfer is the same as the SSL signal level in the current serial transfer, the SPI can continuously perform serial transfer while holding the SSL signal assertion status (burst transfer).

(a) SPCR.BFDS = 0b

Figure 7.5-57 shows an example of SSL signal operation when burst transfer is achieved by using the settings of SPCMD0 and SPCMD1. The following describes SPI operations of (1) to (8) shown in **Figure 7.5-57**. The SSL output signal polarity depends on the set the SSLP register.



(1) Assert the SSL signal and insert an RSPCK delay according to SPCMD0.

(2) Perform serial transfer according to SPCMD0.

(3) Insert an SSL negation delay.

(4) The SSL signal value in SPCMD0 is retained because SPCMD0.SSLKP = 1b. This period additionally continues for 5 RSPI_n_TCLK cycles (at minimum) that is the same as the next-access delay time of SPCMD0. When the shift register is empty after the minimum period has passed, this period continues until transmit data for the next transfer is stored in the shift register.

(5) Assert the SSL signal and insert an RSPCK delay according to SPCMD1.

(6) Perform serial transfer according to SPCMD1.

(7) Insert SSL negate delay.

(8) The SSL signal is negated because SPCMD1.SSLKP = 0b. Furthermore, the next-access delay is inserted according to SPCMD1.

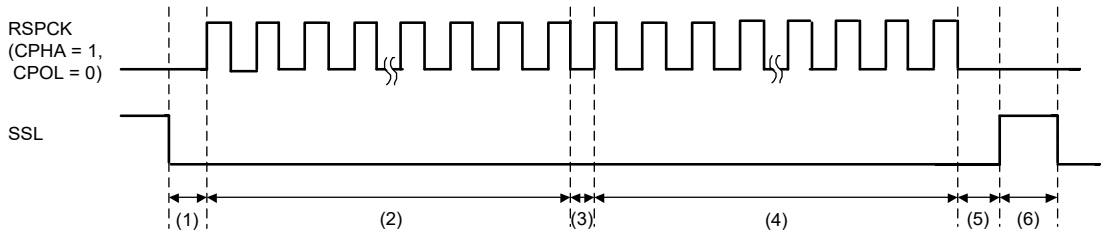
Figure 7.5-57 Example of Burst Transfer Operation Using SSLKP Bit (BFDS = 0b, SPFRF = 0b)

If the SSL signal output setting in SPCMD with the SSLKP bit set to 1 differs from the SSL signal output setting in SPCMD to be used for the next transfer, the SPI changes the SSL signal state when the SSL signal corresponding to the next-transfer command is asserted (5). Note that, if an SSL signal change like this takes place, slaves that drive the MISO signal may conflict with each other, which may cause collision of signal level.

The SPI in master mode references the SSL signal operation in the module when SSLKP is not used. Even when SPCMD.CPHA = 0b, the SPI can accurately start serial transfer by using the next-transfer SSL signal assertion detected internally. For this reason, burst transfer in master mode is enabled regardless of the set CPHA bit value. (See **7.5.3.13 SPI Operation**.)

(b) SPCR.BFDS = 1

Figure 7.5-58 shows an example of SSL signal operation when burst transfer is achieved by using the settings of SPCMD0 and SPCMD1. The following describes SPI operations of (1) to (6) shown in **Figure 7.5-58**. The SSL output signal polarity depends on the set SSSLP register value.



(1) Assert the SSL signal and insert an RSPCK delay according to SPCMD0. The RSPCK delay is inserted only the first frame of burst transmission.

(2) Perform serial transfer according to SPCMD0. Wait last clock until the next transmit data is stored in the shift register, if the shift register is empty during RSPCK negate period between frames.

(3) The value of SSL signal according to SPCMD0 was hold, because SPCMD0.SSLKP = 1b. RSPCK negate period between frames is 0.5RSPCK, if the shift register is not empty.

(4) Perform serial transfer according to SPCMD1.

(5) Insert SSL negate delay for the last frame.

(6) The SSL signal is negated because SPCMD1.SSLKP = 0b. Furthermore, the next-access delay is inserted according to SPCMD1.

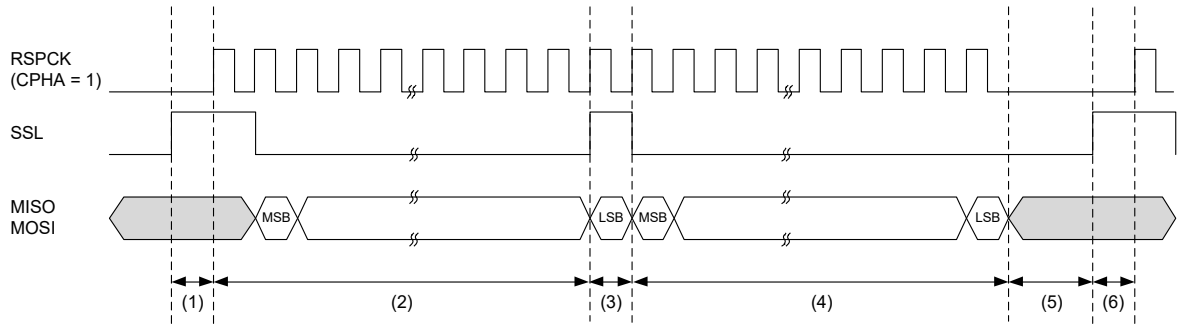
Figure 7.5-58 Example of Burst Transfer Operation Using SSLKP Bit (BFDS = 1b, SPFRF = 0b)

[In the TI-SSP case]

SPI asserts the SSL signal for one cycle at the start of serial transfer.
Serial transfer can be executed continuously by asserting the SSL signal for one cycle at the start of the next serial transfer (burst transfer).

(c) SPCMD.SSLKP = 1b and SPCR.BFDS = 1b

SPCMD0 to SPCMD1 are shown in **Figure 7.5-59**. The following shows an example of SSL signal operation and serial data MISO/MOSI when burst transfer is realized using the settings. The SSL output signal polarity depends on the set SPI slave select polarity register (SSLP) value.



(1) Assert the SSL signal and insert an RSPCK delay according to SPCMD0. The RSPCK delay is inserted only the first frame of burst transmission.

(2) Perform serial transfer according to SPCMD0.

(3) Final data transfer and SSL assertion are performed simultaneously. If the shift register is empty during the RSPCK negation period between frames, wait for the output of the last clock until the transmission data for the next transfer is stored in the shift register.

(4) Perform serial transfer according to SPCMD1.

(5) Insert SSL negate delay for the last frame.

(6) The SSL signal is negated because the SPCMD1.SSLKP = 0b.

Figure 7.5-59 Example of Burst Transfer Operation (SPFRF = 1b)

If the SSL signal output setting in SPCMD with the SSLKP bit set to 1 differs from the SSL signal output setting in SPCMD to be used for the next transfer, the SPI changes the SSL signal state when the SSL signal corresponding to the next-transfer command is asserted (5). Note that, if an SSL signal change like this takes place, slaves that drive the MISO signal may conflict with each other, which may cause collision of signal level.

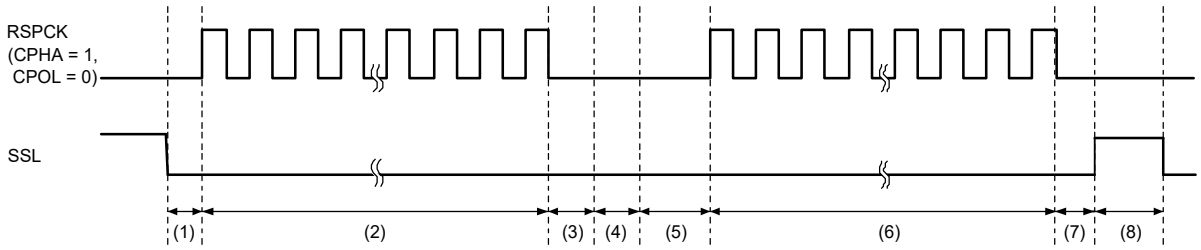
This section describes burst transfer during receive-only operation.

[In the Motorola-SPI case]

When the SPCMD.SSLKP bit, which the SPI references in the current serial transfer, is 1, the SPI retains the SSL signal level during serial transfer until the SSL signal assertion of the next serial transfer starts. When the SSL signal level in the next serial transfer is the same as the SSL signal level in the current serial transfer, the SPI can continuously perform serial transfer while holding the SSL signal assertion status (burst transfer).

(d) SPCR.BFDS = 0b

Figure 7.5-60 shows an example of SSL signal operation when burst transfer is achieved by using the settings of SPCMD0 and SPCMD1. The following describes SPI operations of (1) to (8) shown in **Figure 7.5-60**. The SSL output signal polarity depends on the set SSLP register value.



(1) Assert the SSL signal and insert an RSPCK delay according to SPCMD0.

(2) Perform serial transfer according to SPCMD0.

(3) Insert an SSL negation delay.

(4) The SSL signal value in SPCMD0 is retained because SPCMD0.SSLKP = 1b. This period additionally continues for 5 RSPI_n_TCLK cycles (at minimum) that is the same as the next-access delay time of SPCMD0.

(5) Assert the SSL signal and insert an RSPCK delay according to SPCMD1.

(6) Perform serial transfer according to SPCMD1.

(7) Insert SSL negate delay.

(8) The SSL signal is negated because SPCMD1.SSLKP = 0b. Furthermore, the next-access delay is inserted according to SPCMD1.

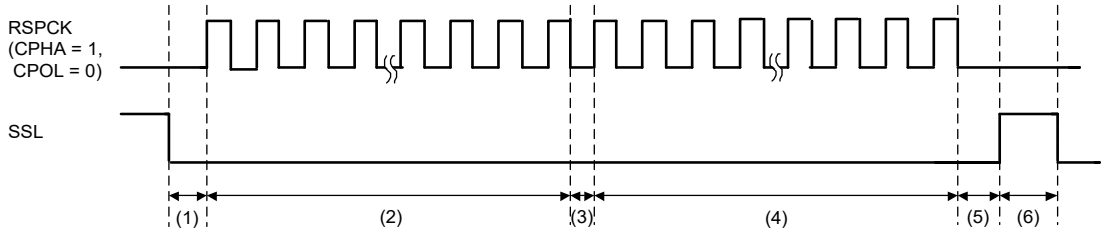
Figure 7.5-60 Example of Burst Transfer Operation Using SSLKP Bit (BFDS = 0b, SPFRF = 0b)

If the SSL signal output setting and the SSL signal output setting between SPCMDs used for burst transfer are different, SPI switches the SSL signal state when the SSL signal corresponding to the next transfer command is asserted (5). Note that, if an SSL signal change like this takes place, slaves that drive the MISO signal may conflict with each other, which may cause collision of signal level.

The SPI in master mode references the SSL signal operation in the module when SSLKP is not used. Even when SPCMD.CPHA = 0b, the SPI can accurately start serial transfer by using the next-transfer SSL signal assertion detected internally. For this reason, burst transfer in master mode is enabled regardless of the set CPHA bit value. (See **7.5.3.13, SPI Operation**.)

(e) SPCR.BFDS = 1b

Figure 7.5-61 shows an example of SSL signal operation when burst transfer is achieved by using the settings of SPCMD0 and SPCMD1. The following describes SPI operations of (1) to (6) shown in **Figure 7.5-61**. The SSL output signal polarity depends on the set SSLP register value.



(1) Assert the SSL signal and insert an RSPCK delay according to SPCMD0. The RSPCK delay is inserted only the first frame of burst.

(2) Perform serial transfer according to SPCMD0.

(3) Since it is not the last frame, the SSL signal value at SPCMD0 is retained. RSPCK negation between frames is 0.5 RSPCK for the next frame.

(4) Perform serial transfer according to SPCMD1.

(5) Insert SSL negate delay for the last time.

(6) The SSL signal is negated because SPCMD1.SSLKP = 0b. Furthermore, the next-access delay is inserted according to SPCMD1.

Note: "Last frame": Frame set by RMFM[4:0] bits when SPCRRM.RMFM[4:0] ≠ 00h
Or, a frame in which SPCRRM.RMEDTG = 1b has been accepted.

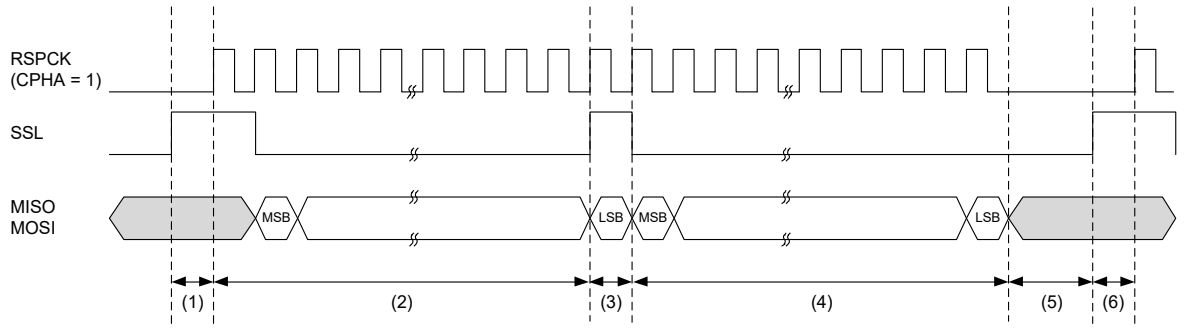
Figure 7.5-61 Example of Burst Transfer Operation Using SSLKP Bit (BFDS = 1, SPFRF = 0)

[In the TI-SSP case]

SPI asserts the SSL signal for one cycle at the start of serial transfer.
Serial transfer can be executed continuously by asserting the SSL signal for one cycle at the start of the next serial transfer (burst transfer).

(f) SPCMD.SSLKP = 1b and SPCR.BFDS = 1b

SPCMD0 to SPCMD1 are shown in **Figure 7.5-62**. The following shows an example of SSL signal operation and serial data MISO/MOSI when burst transfer is realized using the settings. The SSL output signal polarity depends on the set SSLP register value.



(1) Assert the SSL signal and insert an RSPCK delay according to SPCMD0. The RSPCK delay is inserted only the first frame of burst.

(2) Perform serial transfer according to SPCMD0.

(3) Final data transfer and SSL assertion are performed simultaneously.

(4) Perform serial transfer according to SPCMD1.

(5) Insert Output disable delay for the last frame.

(6) The SSL signal is negated because SPCMD1.SSLKP = 0b.

Note: "Last frame": Frame set by RMFM[4:0] bits when SPCRRM.RMFM[4:0] ≠ 00h
Or, a frame in which SPCRRM.RMEDTG = 1b has been accepted.

Figure 7.5-62 Example of Burst Transfer Operation (SPFRF = 0b)

If the SSL signal output setting between the SPCMDs used for burst transfer differs from the SSL signal output setting, SPI switches the SSL signal state when the SSL signal corresponding to the next transfer command is asserted (5). Note that, if an SSL signal change like this takes place, slaves that drive the MISO signal may conflict with each other, which may cause collision of signal level.

(5) RSPCK delay (t1)

The RSPCK delay value of the SPI in master mode depends on the settings of the SPCMD.SCKDEN and SPCKD.SCKDL[2:0] bits. The SPI determines SPCMD to be referenced in serial transfer by the pointer control, and then determines the RSPCK delay value for serial transfer by using the selected SPCMD.SCKDEN bit and SPCKD.SCKDL[2:0] bits as shown in **Table 7.5-11**. For the definition of RSPCK delay, see **7.5.3.5 Transfer Format (Frame Format)**.

RSPCK delay insert to only the first frame of burst transmission, when transmit without "Between Burst Transfer Frames Delay" (SPCMD.SSLKP = 1b, SPCR.BFDS = 1b).

Table 7.5-11 Relationship between SCKDEN/SPCKD and RSPCK Delay Value

SPCMDm.SCKDEN Bit	SPCKD.SCKDL[2:0] Bits	RSPCK Delay Value	
		Motorola-SPI	TI-SSP
0	000b to 111b	1 RSPCK	0 RSPCK
1	000b	1 RSPCK	1 RSPCK
	001b	2 RSPCK	2 RSPCK
	010b	3 RSPCK	3 RSPCK
	011b	4 RSPCK	4 RSPCK
	100b	5 RSPCK	5 RSPCK
	101b	6 RSPCK	6 RSPCK
	110b	7 RSPCK	7 RSPCK
	111b	8 RSPCK	8 RSPCK

(6) SSL negation delay (t2)

The SSL negation delay value of the SPI in master mode depends on the settings of the SPCMD.SLNDEN bit and SSLND.SLNDL[2:0]. The SPI determines SPCMD to be referenced in serial transfer by the pointer control, and then determines the SSL negation delay value for serial transfer by using the selected SPCMD.SLNDEN and SSLND.SLNDL[2:0] as shown in **Table 7.5-12**. For the definition of SSL negation delay, see **7.5.3.5 Transfer Format (Frame Format)**.

SSL negate delay insert to only the last frame of burst transmission, when transmit without “Between Burst Transfer Frames Delay” (SPCMD.SSLKP = 1b, SPCR.BFDS = 1b).

Table 7.5-12 Relationship between SSLND and SSL Negation Delay Value

SLNDEN	SSLND.SLNDL[2:0] Bits	SSL Negation Delay Value
0	000b to 111b	1 RSPCK cycle
1	000b	1 RSPCK cycle
	001b	2 RSPCK cycle
	010b	3 RSPCK cycle
	011b	4 RSPCK cycle
	100b	5 RSPCK cycle
	101b	6 RSPCK cycle
	110b	7 RSPCK cycle
	111b	8 RSPCK cycle

(7) Next-access delay (t3)

The next-access delay value of the SPI in master mode depends on the settings of the SPCMD.SPNDEN bit and SPND.SPNDL[2:0] bits. The SPI determines SPCMD to be referenced in serial transfer by the pointer control, and then determines the RSPCK delay value for serial transfer by using the selected SPCMD.SPNDEN bit and SPND.SPNDL[2:0] bits as shown in **Table 7.5-13**. For the definition of next-access delay, see **7.5.3.5 Transfer Format (Frame Format)**.

Next-Access delay insert to only the last frame of burst transmission, when transmit without “Between Burst Transfer Frames Delay” (SPCMD.SSLKP = 1b, SPCR.BFDS = 1b).

Table 7.5-13 Relationship between SPNDEN Bit, SPND Bits, and Next-Access Delay Value

SPNDEN Bit	SPND.SPNDL[2:0] Bits	Next-Access Delay Value
0	000b to 111b	1 RSPCK + 5 RSPI_n_TCLK cycles
1	000b	1 RSPCK + 5 RSPI_n_TCLK cycles
	001b	2 RSPCK + 5 RSPI_n_TCLK cycles
	010b	3 RSPCK + 5 RSPI_n_TCLK cycles
	011b	4 RSPCK + 5 RSPI_n_TCLK cycles
	100b	5 RSPCK + 5 RSPI_n_TCLK cycles
	101b	6 RSPCK + 5 RSPI_n_TCLK cycles
	110b	7 RSPCK + 5 RSPI_n_TCLK cycles
	111b	8 RSPCK + 5 RSPI_n_TCLK cycles

(8) Initialization flowchart

Figure 7.5-63 shows an example of initialization flow when using the SPI in master-mode SPI operation. For how to set the interrupt controller, DMAC, and input/output ports, see descriptions of each block.

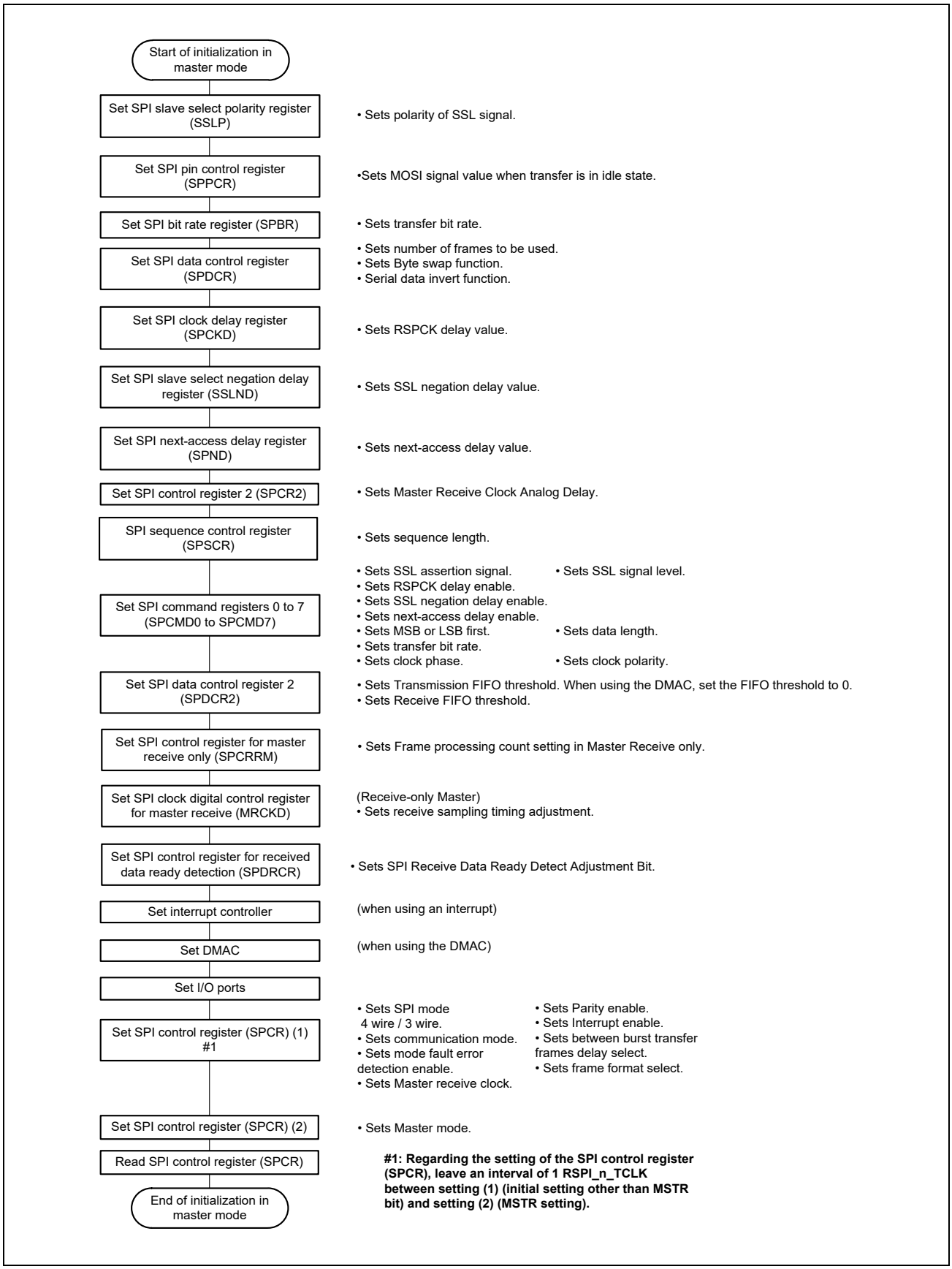


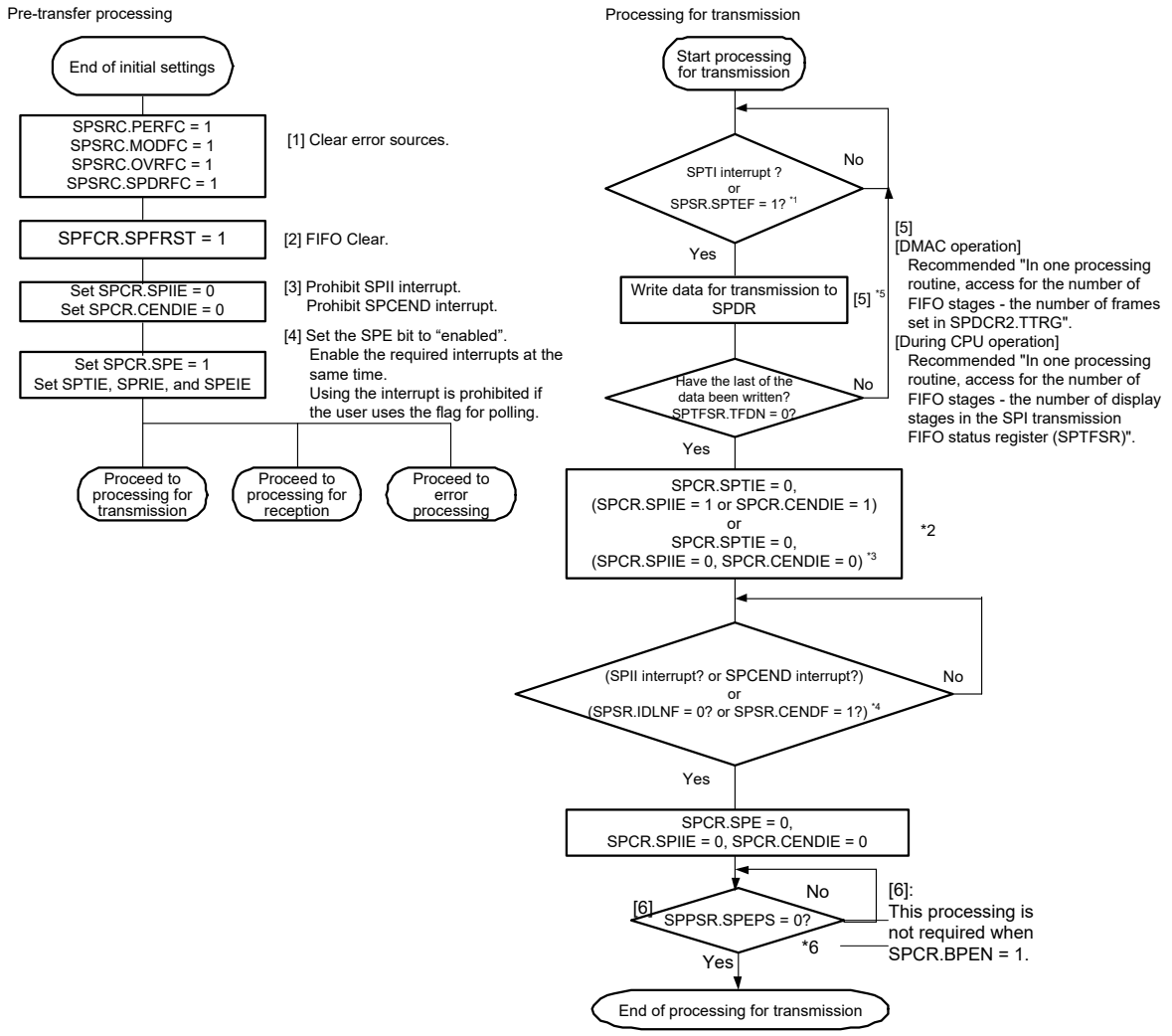
Figure 7.5-63 Example of Initialization Flowchart in Master Mode (SPI Operation)

(9) Software processing flow

Figure 7.5-64 to **Figure 7.5-67** show examples of the flow of software processing.

(a) Transmit processing flow

Enabling an SPII or SPCEND interrupt after final transmit data is written makes it possible to notify the CPU that transmission of all data has been completed.



Note: SPII: Idle interrupt
SPTI: Transmission buffer empty interrupt

Note 1. Proceed to process of writing data for transmission after reading SPSR.SPTEF flag is 1b, if the user uses the flag for polling.

Note 2. The user can know the communication end by SPII interrupt (IDLNF flag) or SPCEND interrupt (CENDF flag). Please use either SPII interrupt (IDLNF flag) or SPCEND interrupt (CENDF flag).

Note 3. Set SPII interrupt prohibited (SPCR.SPIIE = 0b) and SPCEND interrupt prohibited (SPCR.CENDIE = 0b), if the user uses the flag for polling.

Note 4. Wait more than 1 PCLKM, after writing data for transmission to SPDR, and before starting polling of SPSR.IDLNF or SPSR.CENDF, if the user uses the flag for polling.

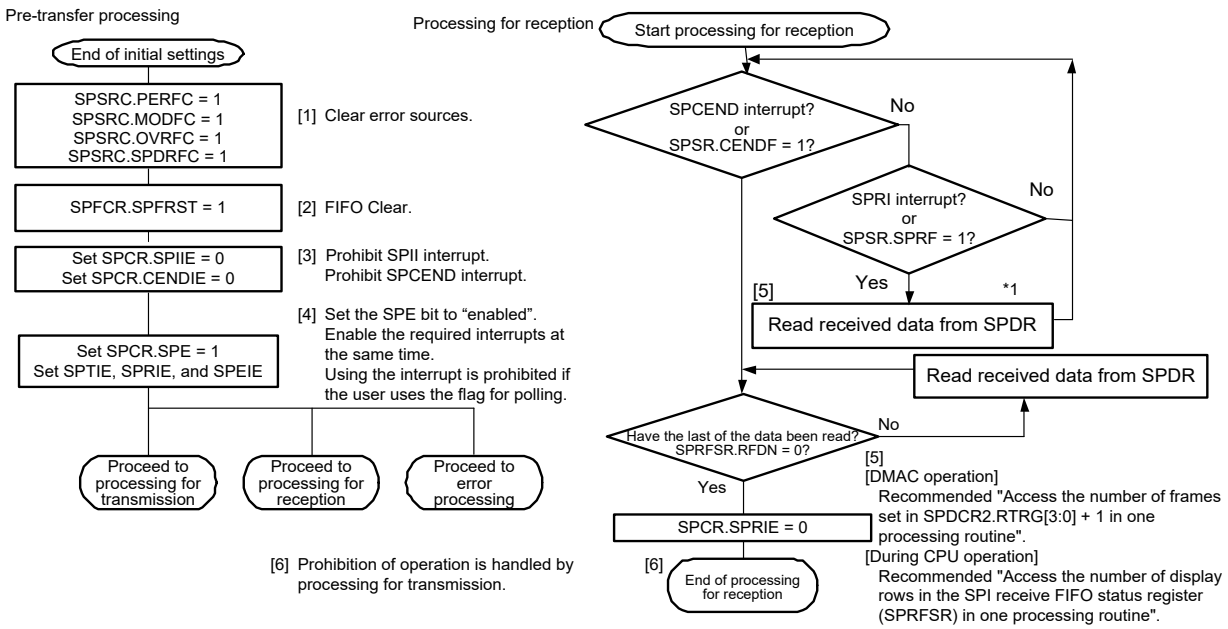
Note 5. During CPU operation, "Write 1b to SPI transmit buffer empty flag clear bit (SPSRC.SPTEFC) after one processing routine."

Note 6. When setting to 0b in slave mode and SPCR.BPEN = 0b, check that SPPSR.SPEPS has become 0b. (Not required when BPEN = 1b.)

Figure 7.5-64 Software Processing Flowchart in Master Mode (Transmission)

(b) Reception processing flow

The SPI always requires transmission because it does not have reception-only operation.

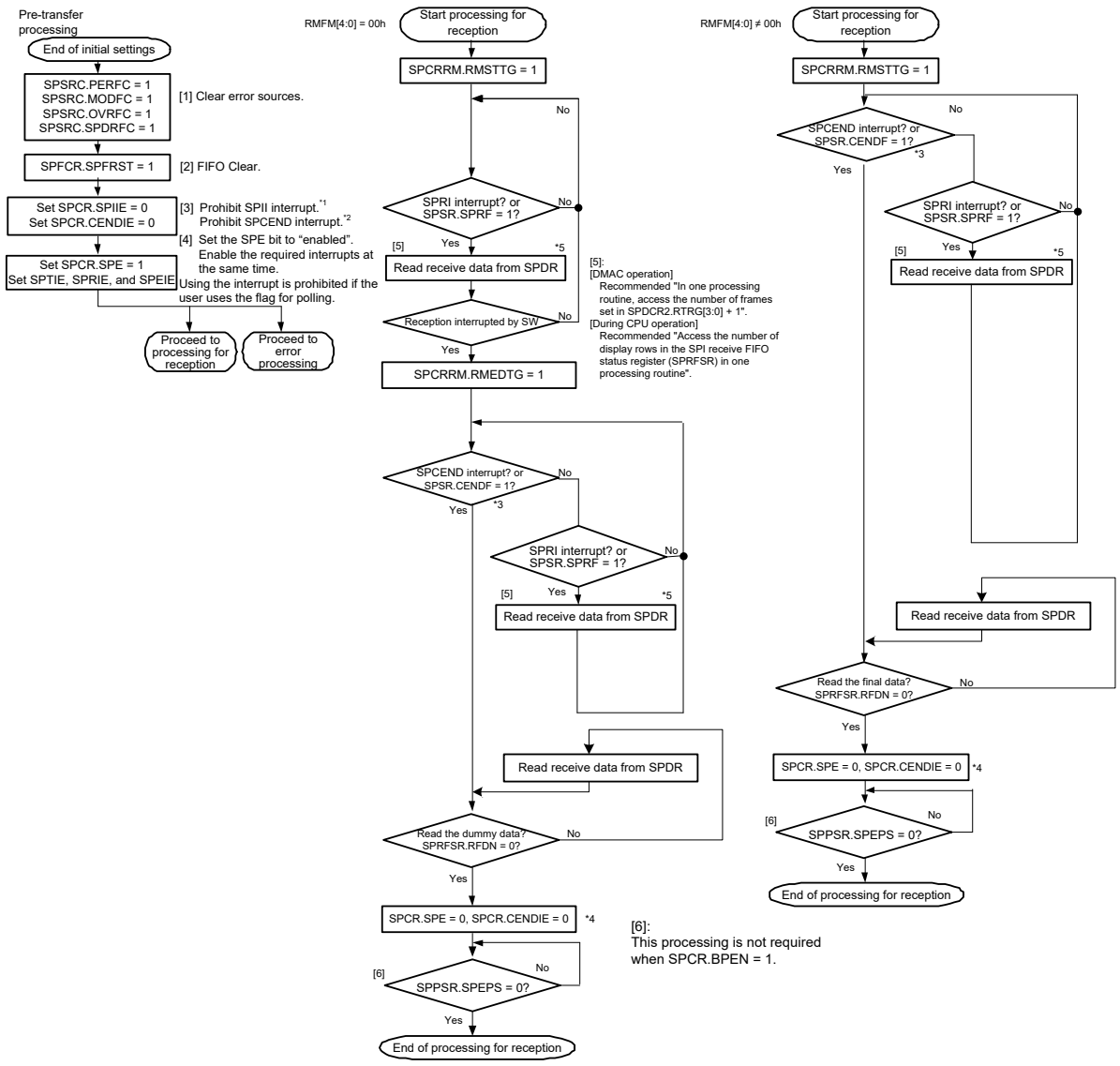


Note: SPII: Idle interrupt
SPRI: Reception buffer full interrupt

Note 1. During CPU operation, write 1 to the SPI receive buffer full flag clear bit (SPSRC.SPRFC) after one processing routine.

Figure 7.5-65 Software Processing Flowchart in Master Mode (Reception)

(c) Master reception-only processing flow



Note 1. The end of communication can be determined by the SPCEND interrupt (CENDF flag). Use the SPCEND interrupt (CENDF flag).

Note 2. Disable SPCEND interrupt (SPCR.CENDIE = 0b) during flag polling.

Note 3. A communication completion interrupt at reception only operation (Master) occurs at a predetermined timing under any of the following conditions.
- Frame number processing set by RMFM completed (RMFM[4:0] ≠ 00h)
- RMEDTG = 1b, when processing of the number of frames set by RMFM is not completed (including RMFM[4:0] = 00h)

Note 4. Internal trigger is also cleared.

Note 5. During CPU operation, write 1b to the SPI receive buffer full flag clear bit (SPSRC.SPRFC) after one processing routine.

Figure 7.5-66 Software Processing Flowchart in Master Mode (Reception-Only)

(d) Error processing flow

When a mode fault error occurs, the SPE bit is automatically cleared to stop transmission and reception operations.

However, other error sources do not clear the SPE bit or stop transmission and reception operations. For this reason, if an error occurs due to another source different from the source of the error occurred first, the SPECM[2:0] value is updated. Therefore, we recommend that you clear the SPE bit to stop the ongoing operation.

When interrupts are used and an error has occurred, clear an Interrupt flag in the error processing because an SPTI interrupt request or an SPRI interrupt request may be remaining in an Interrupt flag. Furthermore, when an SPRI interrupt request is remaining, read the receive buffer to initialize the SPI's internal sequencer.

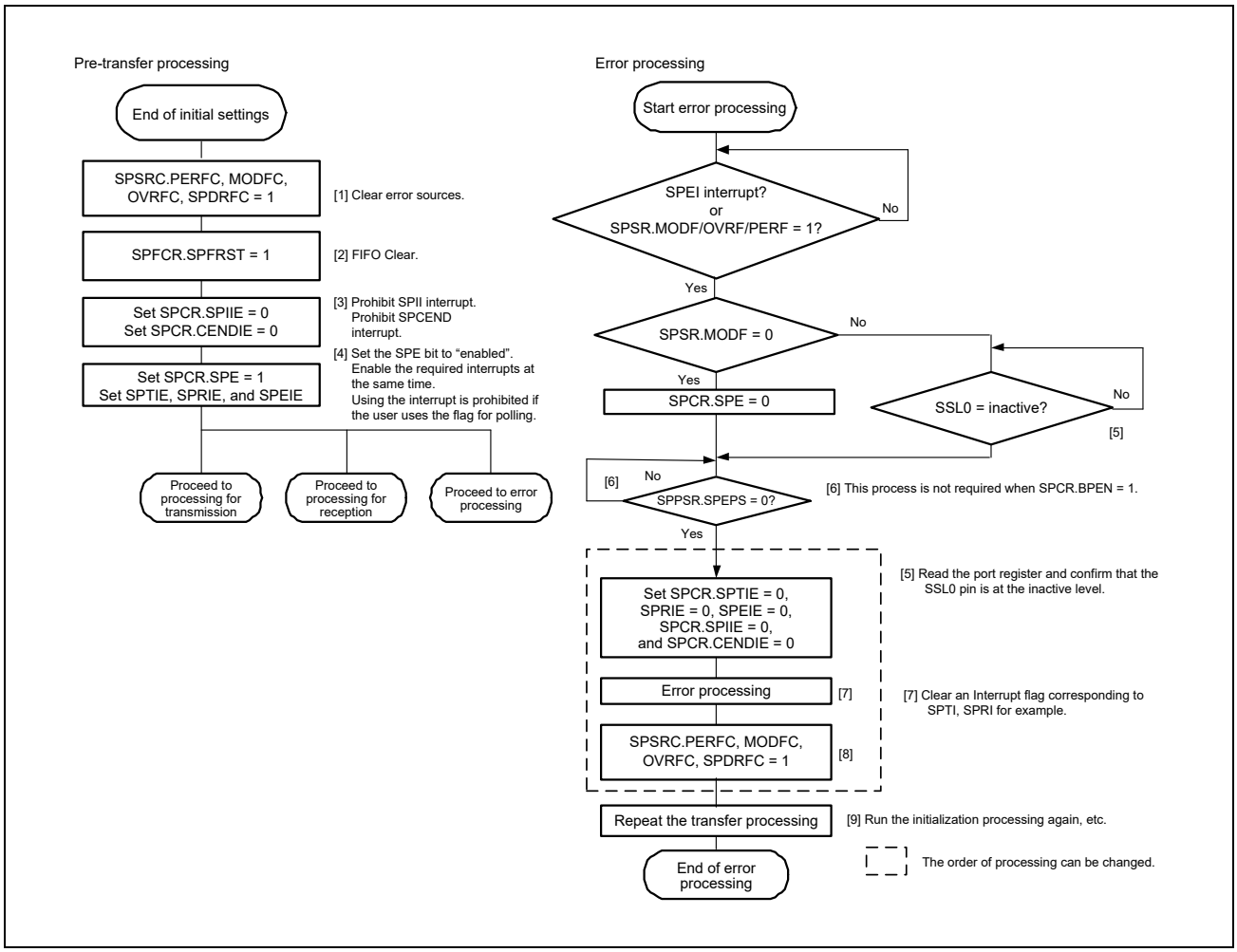


Figure 7.5-67 Flowchart for Master Mode (Error Processing)

7.5.3.13.2 Slave mode operation

(1) Starting a serial transfer

When the SPI detects SSL0 input signal assertion while SPCMD0.CPHA = 0, valid data drive to the MISO output signal must be started. For this reason, the SSL0 input signal assertion becomes a trigger to start serial transfer.

While SPCMD0.CPHA = 1b, when the SPI detects the first RSPCK edge with the SSL0 input signal asserted, valid data drive to the MISO output signal must be started. For this reason, the first RSPCK edge in the SSL0 signal assertion state becomes a trigger to start serial transfer when SPCMD0.CPHA = 1b.

The SPI starts driving the MISO output signal at the SSL0 signal assertion timing independently of the CPHA bit setting. Data that the SPI outputs is enabled or disabled depending on the set CPHA bit value.

For details about the SPI transfer format, see **7.5.3.5 Transfer Format (Frame Format)**. The SSL0 input signal polarity depends on the set SSL0P bit value in the SSLP register.

(2) Terminating a serial transfer

Upon detecting the RSPCK edge corresponding to the final sampling timing regardless of the value of SPCMD0.CPHA bit, the SPI terminates serial transfer. When the number of data stored in the receive FIFO is smaller than the number of FIFO stages, the SPI copies received data from the shift register to the receive buffer in the SPDR register after serial transfer. The SPI also changes the shift register status to empty after serial transfer regardless of the receive buffer status. When the SPI detects SSL0 input signal negation during a period from serial transfer start to the end of serial transfer, a mode fault error occurs. (See **7.5.3.10 Error Detection**.)

The final sampling timing varies depending on the transfer data bit length. The data length of the SPI in slave mode depends on the set value of the SPCMD0.SPB[4:0] bits. The SSL0 input signal polarity depends on the set value of SSLP.SSL0P bit. For details about the SPI transfer format, see **7.5.3.5 Transfer Format (Frame Format)**.

(3) Notes on single-slave operations

[In the Motorola-SPI case]

Upon detecting the SSL0 input signal assertion edge while SPCMD0.CPHA = 0b, the SPI starts serial transfer. When using the SPI in single slave mode in a configuration as shown in the example in **Figure 7.5-9**, the SPI cannot start serial transfer correctly while CPHA is set to 0b because the SSL0 input signal is always active. To correctly perform transmission and reception of the SPI in slave mode in a configuration where the SSL0 input signal is fixed to active state, set the CPHA bit to 1b. If the CPHA bit needs to be set to 0b, do not fix the SSL0 input signal.

[In the TI-SSP case]

When TI SSP is set, when SPI is used as a single slave in the configuration shown in the example of **Figure 7.5-9**, the SSL0 input signal is always fixed to the inactive state, SPI cannot start serial transfer correctly.

When using a single slave, use the configuration shown in the example in **Figure 7.5-8**.

(4) Burst transfer

[In the Motorola-SPI case]

When SPCMD0.CPHA = 1b, continuous serial transfer (burst transfer) is possible while maintaining the SSL0 input signal assertion state. When SPCMD0.CPHA = 1b, the period (from the first RSPCK edge in the SSL0 input signal active state until the sampling timing for receiving the final bit) corresponds to the serial transfer period. Even if the SSL0 input signal is always at the active level, start of an access can be detected, which allows burst transfer.
When SPCMD0.CPHA = 0b, the second and subsequent serial burst transfers cannot be performed correctly for the same reason as (3).

[In the TI-SSP case]

In serial transfer, data transfer starts after the SSL input signal is asserted for RSPCK 1 cycle. Since frame transfer starts from the SSL input signal, SSL must be asserted between frames.

(5) Initialization flowchart

Figure 7.5-68 shows an example of initialization flow when using the SPI in slave-mode SPI operation. For how to set the interrupt controller, DMAC, and input/output ports, see descriptions of each block.

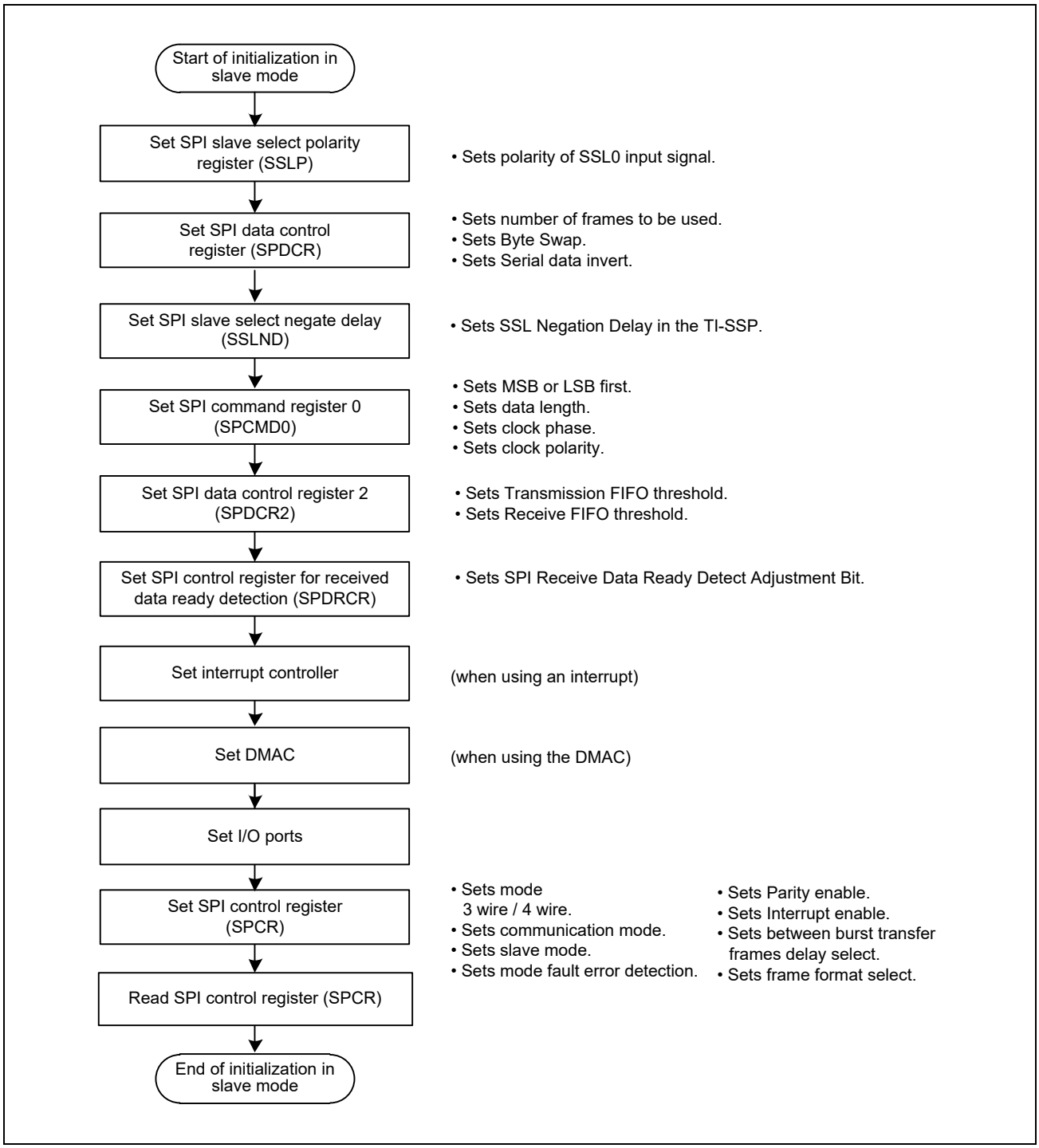
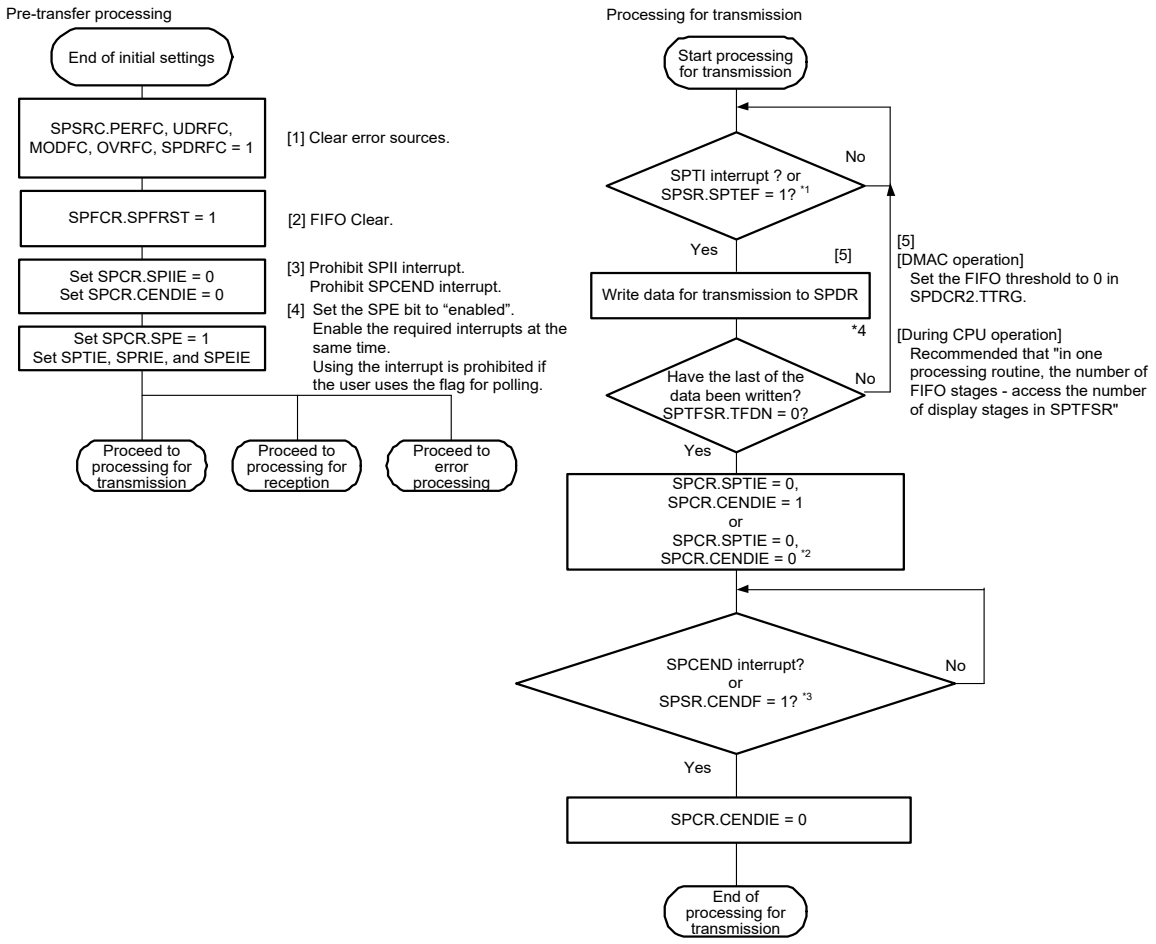


Figure 7.5-68 Example of Initialization Flow in Slave Mode

(6) Software processing flow

Figure 7.5-69 to **Figure 7.5-72** show examples of the flow of software processing.

(a) Transmit processing flow



Note: SPII: Idle interrupt

Note 1. Proceed to process of writing data for transmission after reading SPSR.SPTEF flag is 1b, if the user uses the flag for polling.

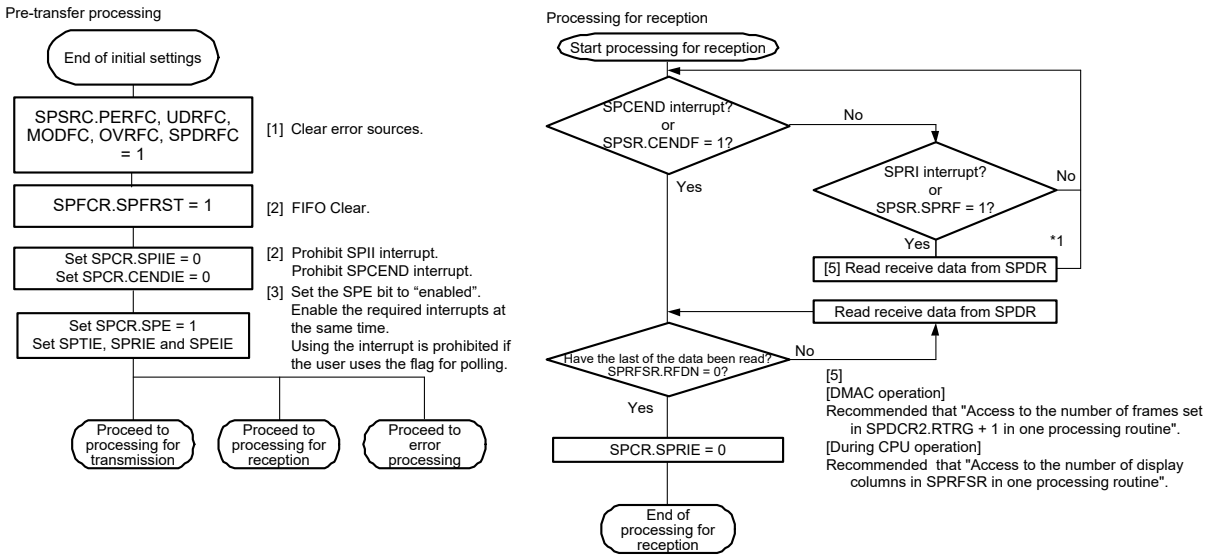
Note 2. Set SPCEND interrupt prohibited (SPCR.CENDIE = 0b), if the user uses the flag for polling.

Note 3. Wait more than 1 PCLKM, after writing data for transmission to SPDR, and before starting polling of SPSR.IDLNF or SPSR.CENDF , if the user uses the flag for polling.

Note 4. During CPU operation, write 1b to the SPI transmit buffer empty flag clear bit (SPSRC.SPTEFC) after one processing routine.

Figure 7.5-69 Software Processing Flowchart in Slave Mode (Transmission)

(b) Reception processing flow

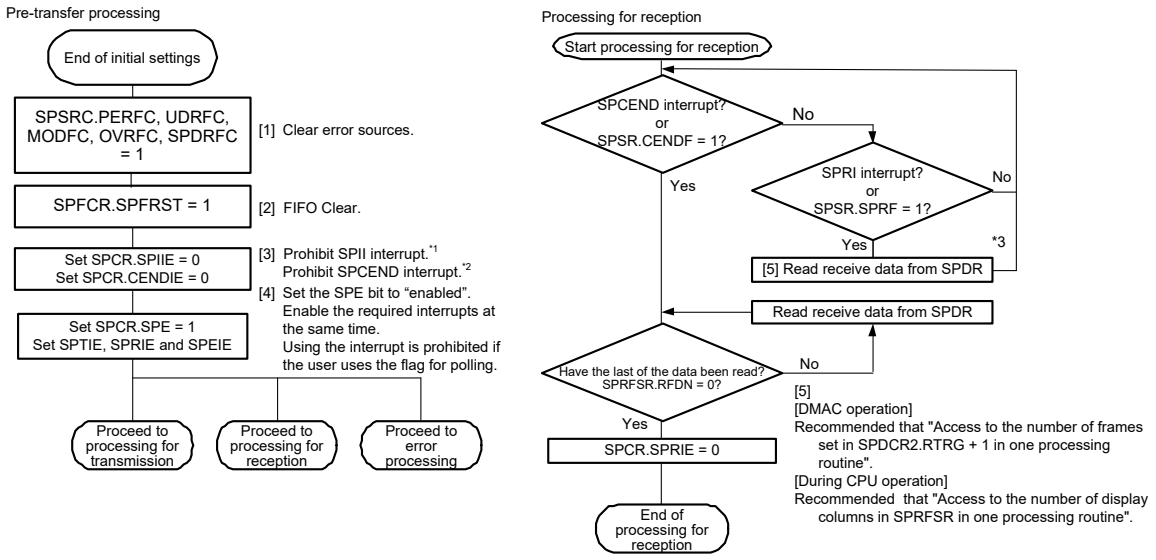


Note: SPII: Idle interrupt

Note 1. During CPU operation, "Write 1b to the SPI receive buffer full flag clear bit (SPSRC.SPTEFC) after one processing routine."

Figure 7.5-70 Flowchart in Slave Mode (Reception)

(c) Master reception-only processing flow



Note: SPII: Idle interrupt

Note 1. The end of communication can be determined by the SPCEND interrupt (CENDF flag). Use the SPCEND interrupt (CENDF flag).

Note 2. Disable the SPCEND interrupt (SPCR.CENDIE = 0b) during flag polling.

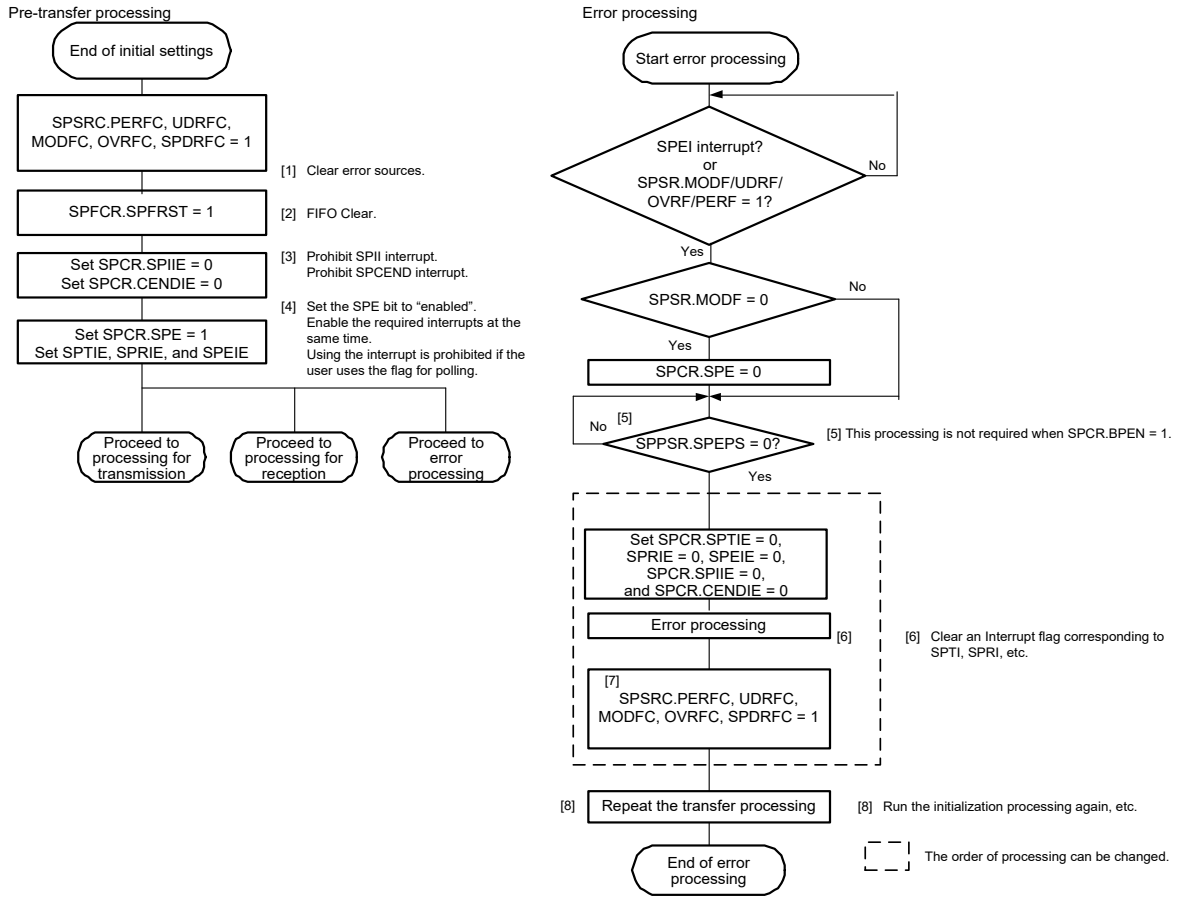
Note 3. During CPU operation, write 1 to the SPI receive buffer full flag clear bit (SPSRC.SPTEFC) after one processing routine.

Figure 7.5-71 Software Processing Flowchart in Slave Mode (Reception-Only)

(d) Error processing flow

In slave mode operation, the MODF flag can be cleared regardless of the SSL0 pin status even when a mode fault error is present.

When interrupts are used and an error has occurred, clear an Interrupt flag in the error processing because an SPTI interrupt request or an SPRI interrupt request may be remaining in an Interrupt flag. Furthermore, when an SPRI interrupt request is remaining, read the receive buffer to initialize the SPI's internal sequencer.



Note: SPEI: SPI error interrupt
SPII: Idle interrupt
SPRI: Reception buffer full interrupt
SPTI: Transmission buffer empty interrupt

Figure 7.5-72 Software Processing Flowchart in Slave Mode (Error Processing)

7.5.3.14 Clock Synchronous Operation

When SPCR.SPMS bit = 1b, the SPI operates in synchronization with clock. Clock synchronous operation enables communication by using three pins RSPCK, MOSI, and MISO without using SSL pins. SSL pins can be used as I/O ports.

Clock synchronous operation does not use SSL pins for communication, but the module operates internally in the similar manner to SPI operation. In master mode operation and slave mode operation, communication is possible according to the flow similar to SPI operation, but no mode fault error is detected because SSL pins are not used.

In clock synchronous operation, if the SPCMD.CPHA bit is set to 0b in slave mode (SPCR.MSTR = 0b), subsequent operation is not guaranteed.

7.5.3.14.1 Master mode operation

(1) Starting a serial transfer

When data is written to the SPDR register while the next transfer data is not set in the transmit FIFO, the SPI updates the transmit buffer (SPTXn, n = 0 to 15) data in SPDR. While the shift register is empty, the SPI copies transmit buffer data to the shift register to start serial transfer. After the SPI copies transmit data to the shift register, it changes the shift register status to full. Upon completion of serial transfer, the SPI changes the shift register status to empty. The shift register status cannot be monitored.

For details about the SPI transfer format, see **7.5.3.5 Transfer Format (Frame Format)**. In clock synchronous operation, however, the SSL0 output signal is not used for communication.

(2) Terminating a serial transfer

Upon detecting the RSPCK edge corresponding to the final sampling timing, the SPI terminates serial transfer. When the number of data stored in the receive FIFO < the number of FIFO stages, the SPI copies received data from the shift register to the receive buffer in the SPDR register after serial transfer.

The final sampling timing varies depending on the transfer data bit length. The data length of the SPI in master mode depends on the set value of the SPCMD.SPB[4:0] bits. For details about the SPI transfer format, see **7.5.3.5 Transfer Format (Frame Format)**. In clock synchronous operation, however, the SSL0 output signal is not used for communication.

(3) Sequence control

The transfer format in master mode is determined by the SPSCR, SPCMDm (m = 0 to 7), SPBR, SPCKD, SSLND, and SPND registers. These settings are valid though SSL signals are not output in clock synchronous operation.

The SPSCR register is used to determine the sequence configuration for serial transfer to be performed by the SPI in master mode. MSB first/LSB first, data length, a part of bit rate settings, RSPCK polarity and phase, SPCKD enable/disable, SSLND enable/disable, and SPND enable/disable are set in SPCMD0 to SPCMD7. A part of bit rate settings is set in SPBR, the SPI clock delay value is set in SPCKD, the SSL negation delay value is set in SSLND, and the access delay value is set in SPND.

The SPI configures the sequence that structures a part of or whole of SPCMD0 to SPCMD7 according to the sequence length specified in SPSCR. The SPI has a pointer to SPCMD that configures the sequence. This pointer value can be checked by reading the SPSSR.SPCP[2:0] bits. When the SPCR.SPE bit is set to 1b to enable the SPI function, the SPI sets the pointer to commands in SPCMD0 and applies the SPCMD0 setting to the transfer format at the start of serial transfer. The SPI increments the pointer when the next-access delay period ends in each data transfer. When the serial transfer corresponding to the final command that configures the sequence is completed, the SPI sets the pointer in SPCMD0. Thus sequence is repeatedly executed.

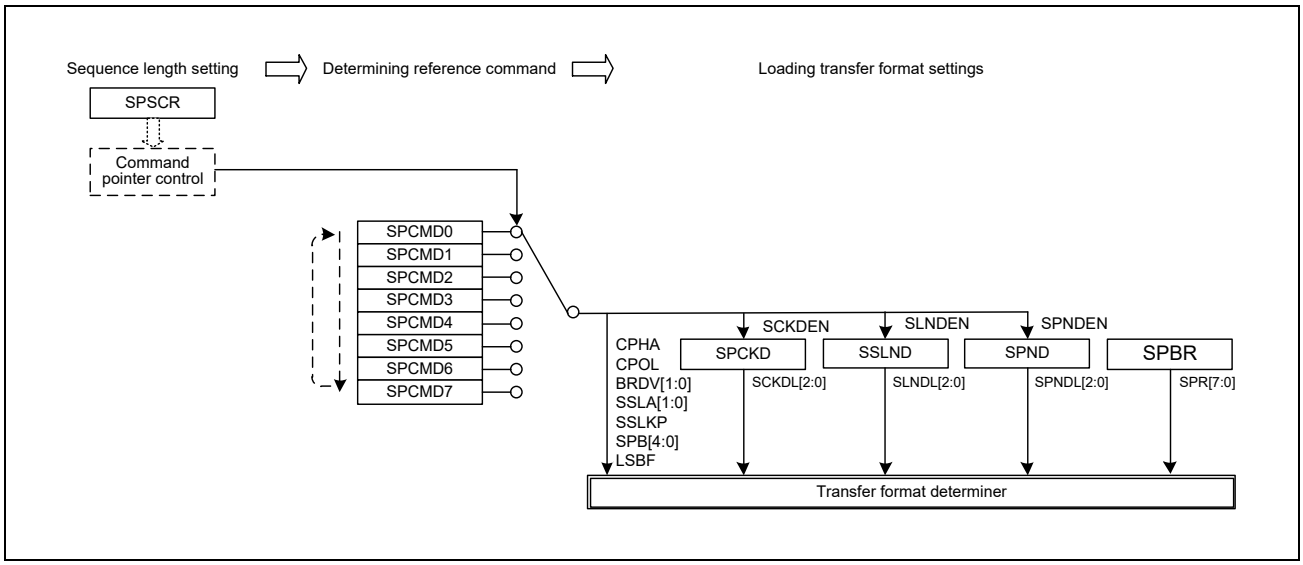


Figure 7.5-73 Determining Serial Transfer Method in Master Mode

In this section, a frame consists of data (SPDR) and configuration (SPCMDm (m = 0 to 7)).

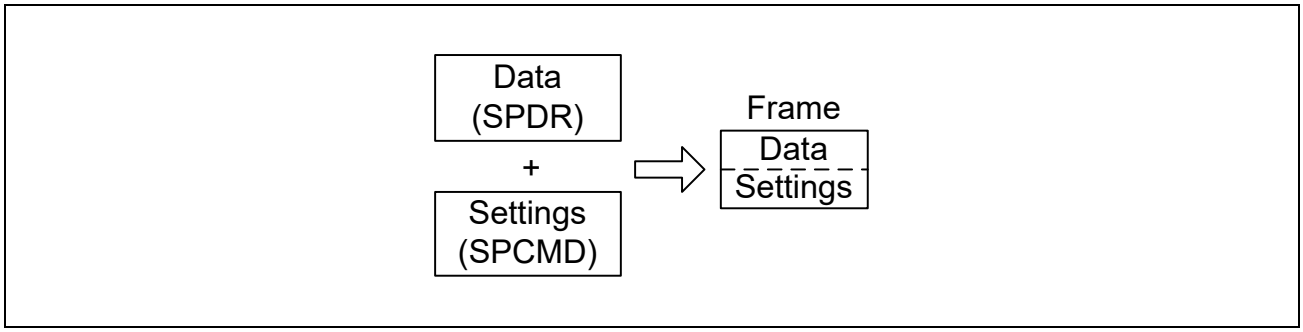


Figure 7.5-74 Conceptual Diagram of Frame

Figure 7.5-75 shows the correspondence between the command in the sequence operation performed and the transmit buffer/receive buffer.

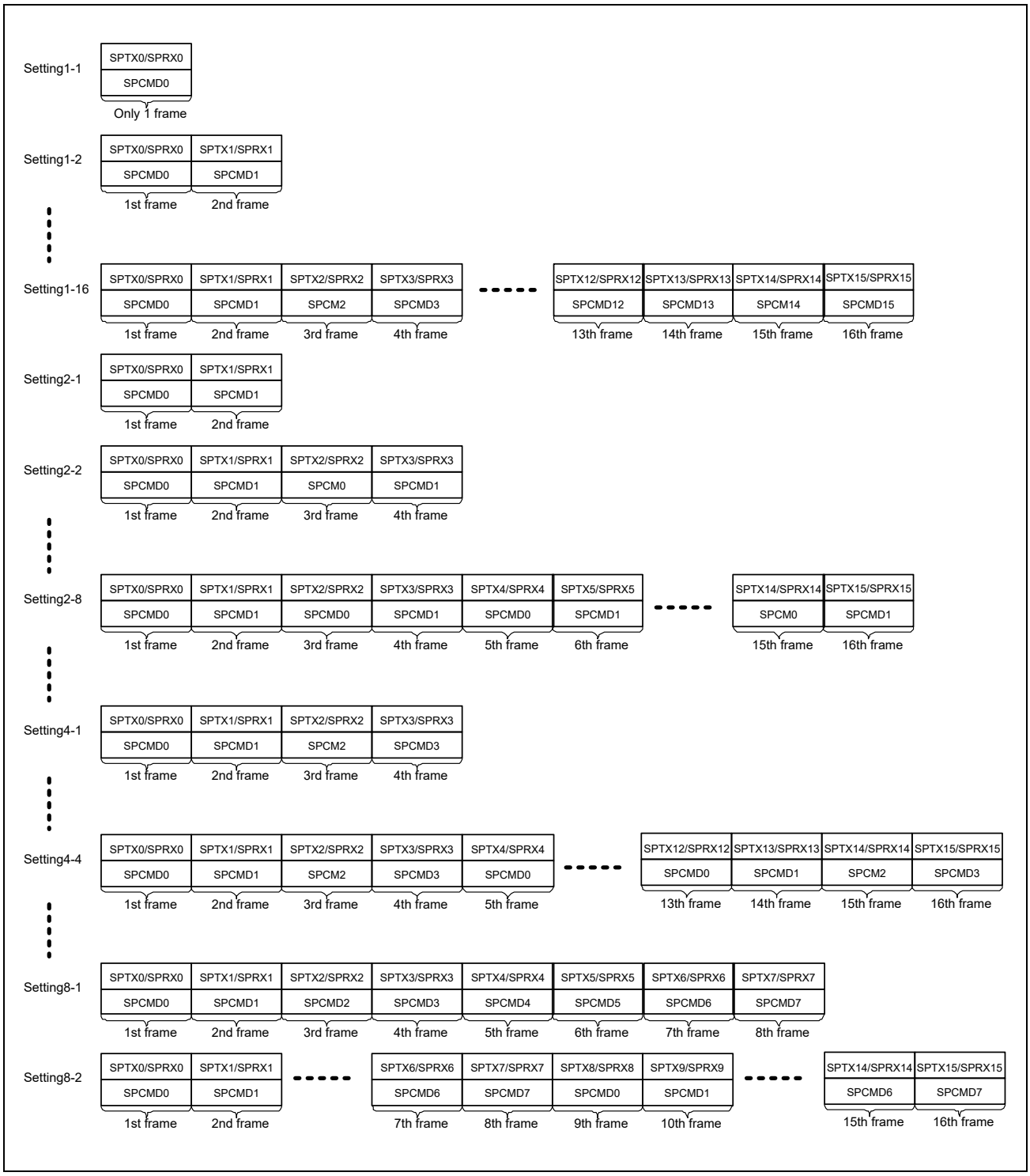


Figure 7.5-75 Correspondence between SPI Command Register and Transmit/Receive Buffers in Sequence Operation

(4) Initialization flow

Figure 7.5-76 shows an example of initialization flow when using the SPI in master-mode clock synchronous operation. For how to set the interrupt controller, DMAC, and input/output ports, see descriptions of each block.

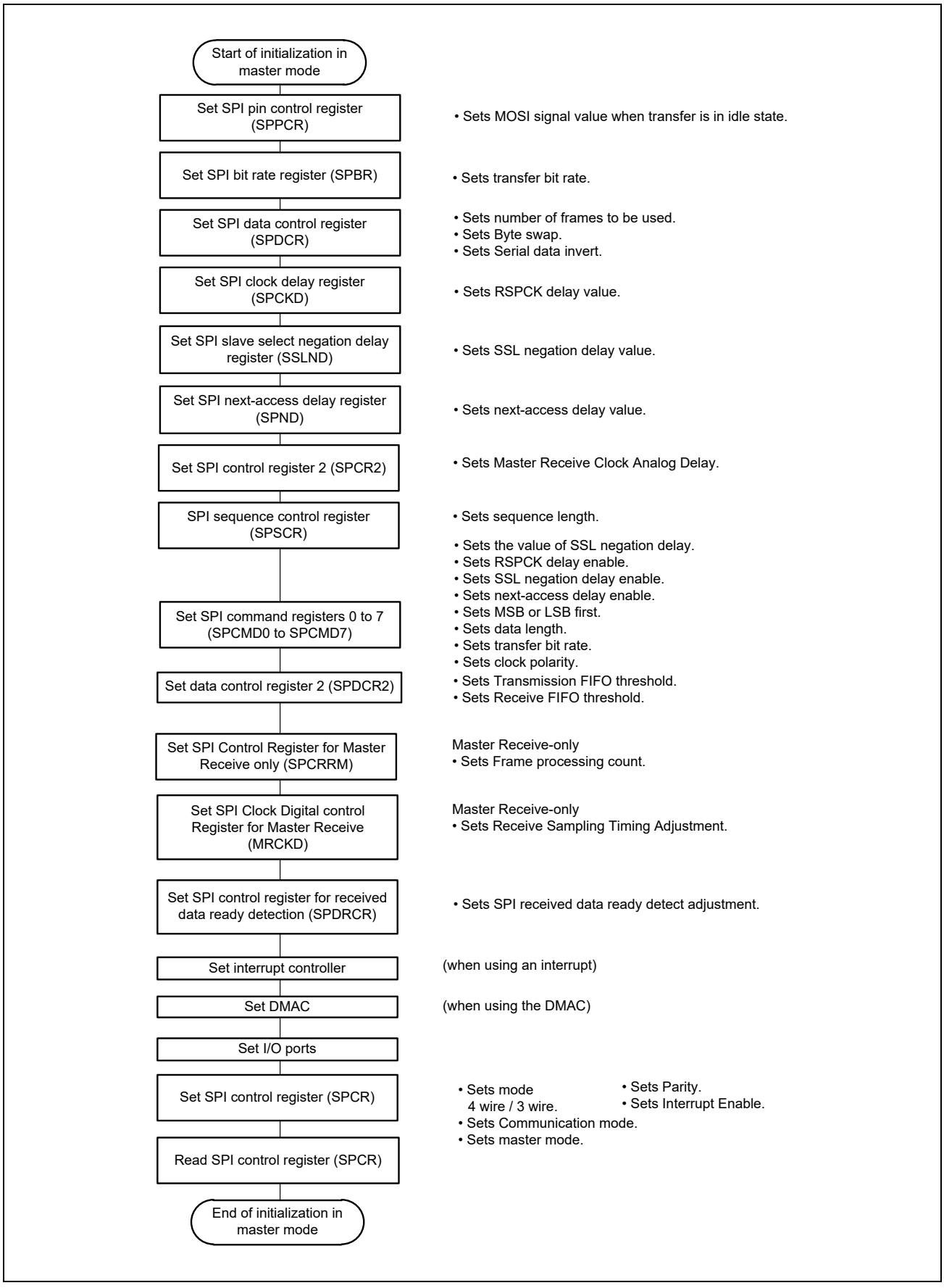


Figure 7.5-76 Example of Initialization Flowchart in Master Mode (Clock Synchronous Operation)

(5) Flow of software processing

The software processing flow is omitted here because it is similar to the flow for SPI operation. However, no mode fault error occurs.

7.5.3.14.2 Slave mode operation

(1) Starting a serial transfer

When SPCR.SPMS = 1b, the first RSPCK edge becomes a trigger to start serial transfer of the SPI.

While SPCR.SPMS = 1b, the SPI always drives the MISO output signal.

For details about the SPI transfer format, see **7.5.3.5 Transfer Format (Frame Format)**. In clock synchronous operation, however, the SSL0 input signal is not used.

(2) Terminating a serial transfer

Upon detecting the RSPCK edge corresponding to the final sampling timing, the SPI terminates serial transfer. When the number of data stored in the receive FIFO < the number of FIFO stages, the SPI copies received data from the shift register to the receive buffer in the SPDR register after serial transfer. The SPI also changes the shift register status to empty after serial transfer regardless of the receive buffer status. The final sampling timing varies depending on the transfer data bit length. The data length of the SPI in slave mode depends on the set value of the SPCMD0.SPB[4:0] bits. For details about the SPI transfer format, see **7.5.3.5 Transfer Format (Frame Format)**.

(3) Initialization flowchart

Figure 7.5-77 shows an example of initialization flow when using the SPI in slave-mode clock synchronous operation. For how to set the interrupt controller, DMAC, and input/output ports, see descriptions of each block.

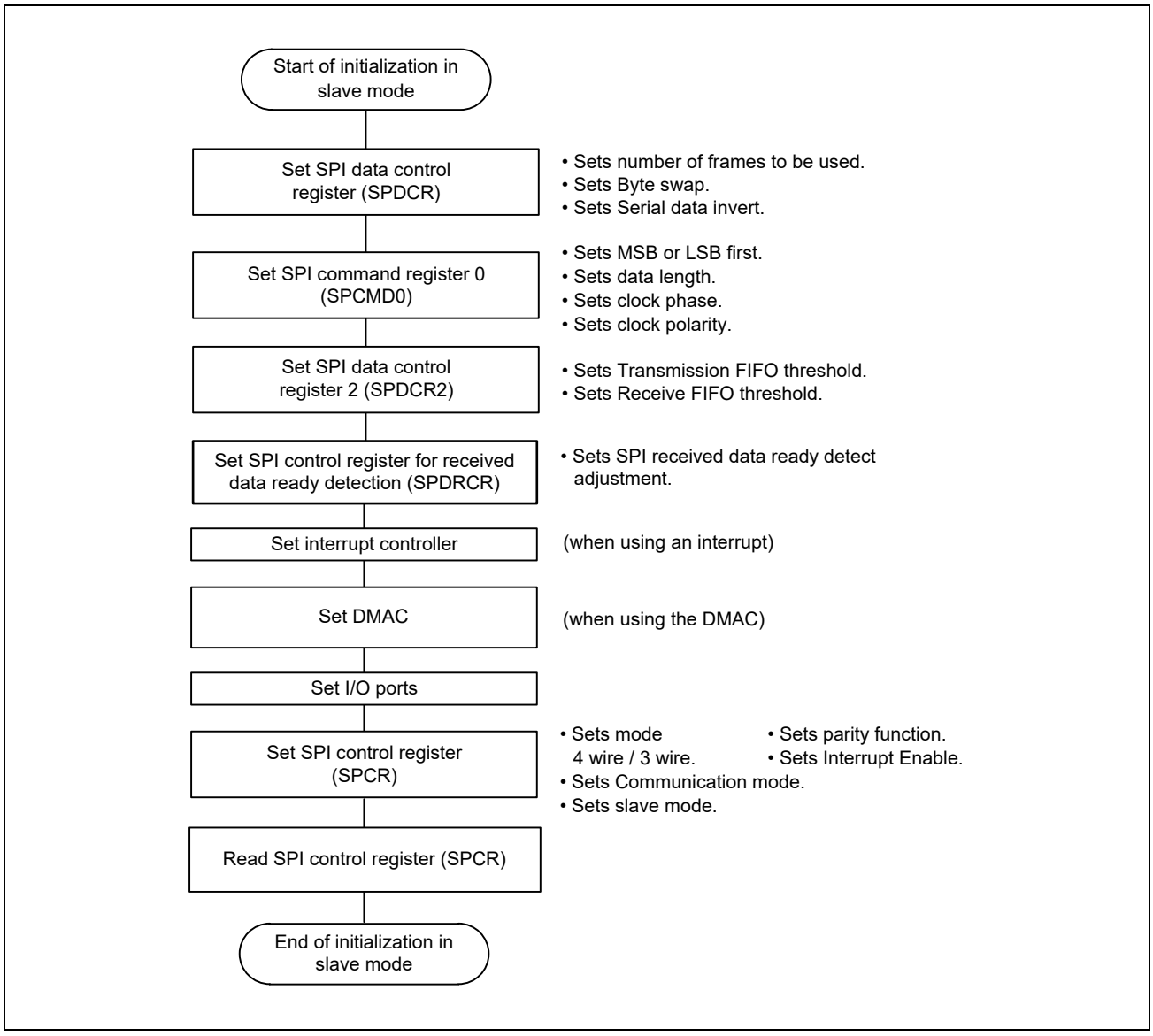


Figure 7.5-77 Example of Initialization Flow in Slave Mode

(4) Flow of software processing

The software processing flow is omitted here because it is similar to the flow for SPI operation. However, no mode fault error occurs.

7.5.3.15 Loopback Mode

When 1b is written to the SPPCR.SPLP bit or SPPCR.SPLP2 bit, the SPI disconnects the path between MISO pin and shift register (when SPCR.MSTR = 1b) or between MOSI pin and shift register (when SPCR.MSTR = 0b), and then connects the shift register input path to the output path. When SPCR.MSTR = 1, the SPI does not disconnect the path between MOSI pin and shift register. When SPCR.MSTR = 0b, the SPI does not disconnect the path between MISO pin and shift register.

When serial transfer is performed in loopback mode, transmit data or inverted transmit data of the SPI becomes received data of the SPI.

Figure 7.5-78 shows the relationship between settings of the SPLP2 and SPLP bits and received data.

Table 7.5-14 SPLP2 and SPLP Bit Settings and Received Data

SPPCR.SPLP2 Bit	SPPCR.SPLP Bit	Received Data
0b	0b	Input data from the MOSI pin or MISO pin
0b	1b	Inversed transmit data
1b	0b	Transmit data
1b	1b	Transmit data

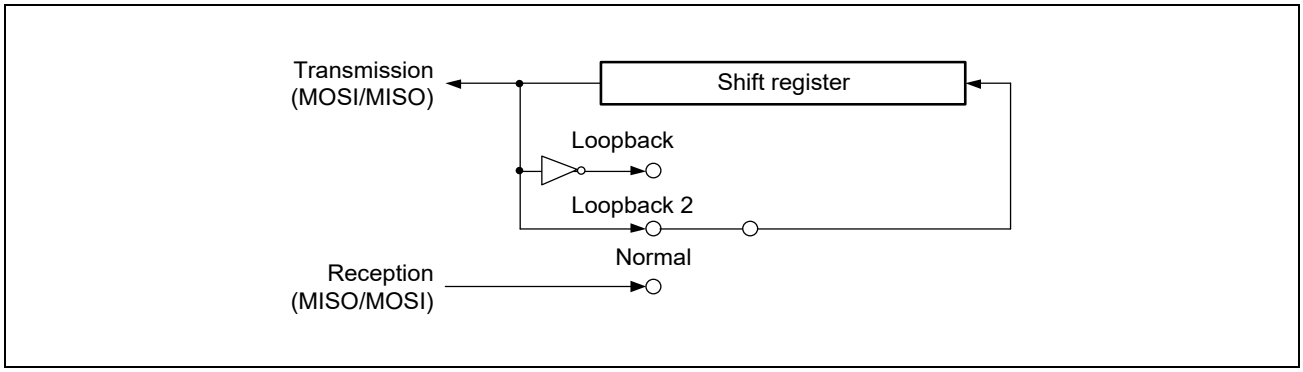


Figure 7.5-78 Configuration of Shift Register I/O Paths in Loopback Mode

7.5.3.16 Self-Diagnosis of Parity Function

The parity circuit consists of a section to add a parity bit to transmit data and a section to detect an error in receive data.

To detect a failure of these sections of the parity circuit, self-diagnosis of the parity circuit is performed according to the flow in **Figure 7.5-79**.

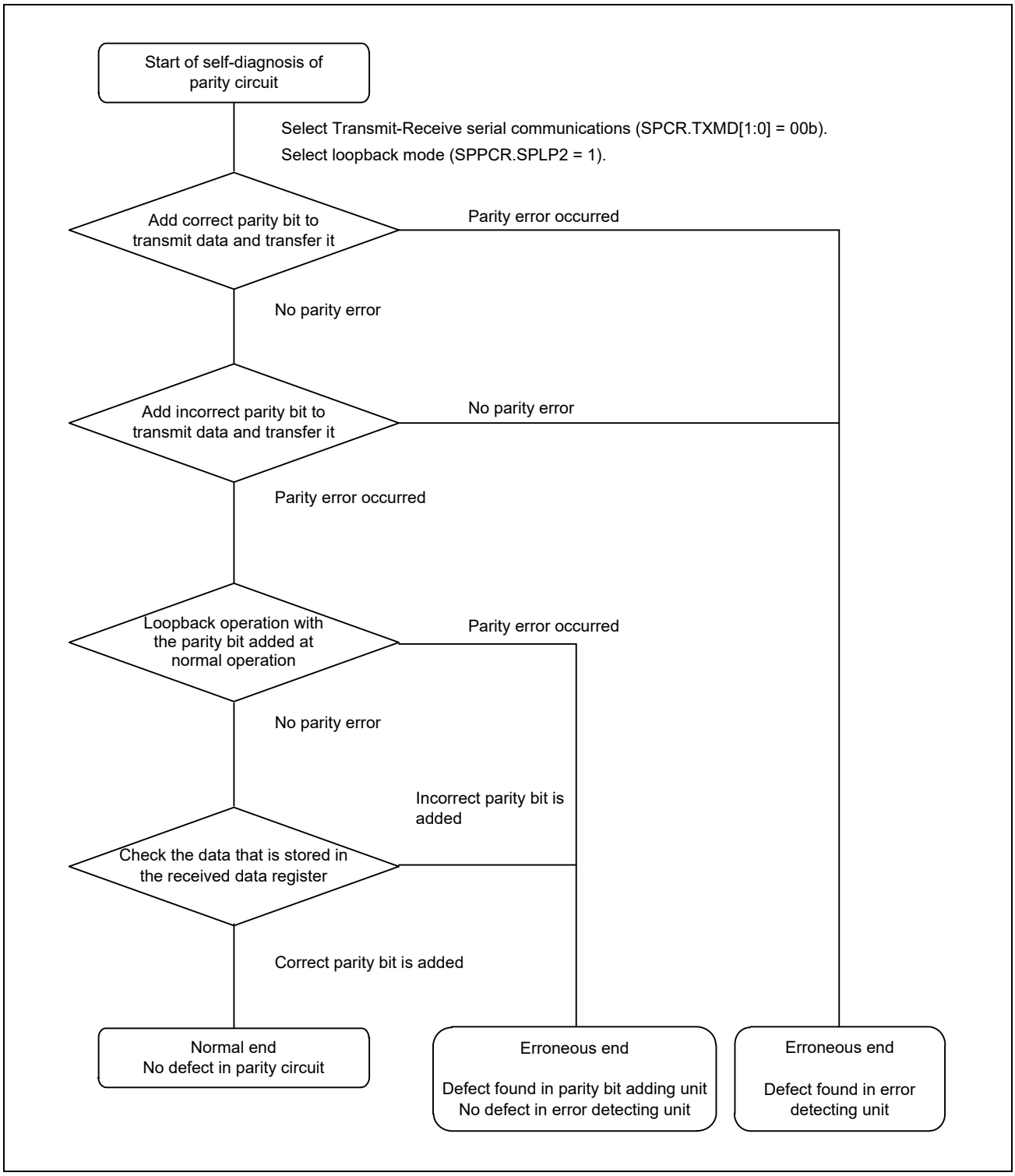


Figure 7.5-79 Parity Circuit Self-Diagnosis Flow

7.5.3.17 Interrupt Requests

The SPI has the following interrupt sources: receive buffer full, transmit buffer empty, mode fault, underrun, overrun, parity error, SPI idle, communication end, and received data ready. Furthermore, the DMAC can be activated by a receive buffer full interrupt request and a transmit buffer empty interrupt request to start data transfer.

Mode fault, underrun, overrun, parity error, and received data ready (SPCR.SPDRES = 1b) interrupt requests are allocated to vector addresses in SPEI. Therefore, identify these interrupt sources by reading respective flags. **Table 7.5-15** lists SPI's interrupt sources. When each interrupt condition in **Table 7.5-15** is met, an interrupt is generated. Clear the receive buffer full and transmit buffer empty interrupt sources by performing data transfer.

When starting transmission or reception by using the DMAC, configure the DMAC to enable it, and then configure the SPI.

Even when the transmit buffer empty interrupt or receive buffer full interrupt condition is met while an Interrupt flag is 1, an interrupt request is not output to ICU but is internally retained. (Only one request per source can be internally retained.) When an Interrupt flag is cleared to 0, the retained interrupt request is output to ICU and then automatically cleared. An internally retained interrupt request can also be cleared by setting the corresponding interrupt enable bit (SPCR.SPTIE bit or SPCR.SPRIE bit) to 0b.

Table 7.5-15 Interrupt Sources of SPI

Interrupt Source	Symbol	ICU Input Name	Interrupt Condition	DMAC Activation
Reception buffer full	SPRI	RSPI_CHn_sp_rxintpls_n	SPRIE = 1b and (receive buffer full (SPRF = 1b) | (SPDRES = 0b and Receive data ready (SPDRF = 1b)))	Activated
Transmission buffer empty	SPTI	RSPI_CHn_sp_txintpls_n	SPTIE = 1b and transmit buffer empty (SPTEF = 1b)	Activated
SPI errors (mode fault, underrun, overrun, and parity error, received data ready)	SPEI	RSPI_CHn_sp_errint_n	SPEIE = 1b and (MODF = 1b | UDRF = 1b | OVRF = 1b | PERF = 1b | (SPDRES = 1b and SPDRF = 1b))	—
SPI idle	SPII	RSPI_CHn_sp_idint_n	SPIIE = 1b and IDLNF = 0b	—
Communication end*1	SPCEND	RSPI_CHn_sp_ceintpls_n	CENDIE = 1b and CENDF = 1b	—

Note 1. See **7.5.3.9.7 Common operation** for details on the communication completion interrupt output conditions.

Note: n = 0 to 2

7.5.4 Usage Notes

7.5.4.1 Notes on Communication Start

When an Interrupt flag is 1 at the beginning of communication, perform the following procedure to clear an interrupt request before enabling operation (setting the SPCR.SPE bit to 1b).

1. Confirm that communication is not going (SPCR.SPE = 0b).
2. Set the corresponding interrupt enable bit (SPCR.SPTIE bit or SPCR.SPRIE bit) to 0b.
3. Read the corresponding interrupt enable bit (SPCR.SPTIE bit or SPCR.SPRIE bit) and confirm that it is 0b.
4. Set the Interrupt flag to 0.

7.5.4.2 Notes on the Module-Stop Function

To stop this module using the power consumption reducing function, be sure to set the SPE bit in SPCR to 0b to terminate communication and then use the power consumption reducing function.

7.5.4.3 Notes on SPRF and SPTEF Flags

When using the SPRF and SPTEF flags in SPSR by polling them, set the SPRIE and SPTIE bits in SPCR to 0b.

7.5.4.4 Notes on Receive Sampling Function

When using the receive sampling function (sampling receive data with MRIOCLK or MRCLK), set the SSLND register considering the delay of the receive data sampling clock (MRIOCLK or MRCLK) from RSPCK. Please see **7.5.2.2.3 SPI Slave Select Negation Delay Register (RSPIm_SSLND)**.

7.5.4.5 Notes on Burst Transfer in Master Mode

During burst transfer, changing the following settings between SPCMD with the SSLKP bit set to 1b and SPCMD used in the next transfer is prohibited.

1. SSL Signal Assertion setting (SSLA[1:0] bits)
2. RSPCK output setting (CPHA, CPOL, BRDV[1:0] bits)

7.5.4.6 Notes on TI-SSP in Slave Mode

In the slave TI SSP mode, the delay between frames must observe the following intervals.

Secure the following intervals on the master side.

[Master side: interval from RSPCK edge of last bit to RSPCK edge of next SSL] > [Slave side: Output enable delay time = RSPI_n_TCLK × (1 to 2) + SLNDL[2:0] setting value]

7.5.4.7 Notes at SPE = 1b

(1) When SPCR.SPE = 1b, if the following register bits are rewritten, subsequent operations are not guaranteed.

Table 7.5-16 Communication End Event Generating Conditions (Receive Only Slave Mode) (1/2)

Register Name	Bit Name
SPCKD	SCKDL[2:0]
SSLND	SLNDL[2:0]
SPND	SPNDL[2:0]
MRCKD	ARST[2:0]
SPCR	BPEN
	MSTR
	TXMD[1:0]
	SPFRF
	SPMS
	MODFEN
	BFDS
	SCKASE
	PTE
	SPOE
	SPPE
	SPSCKSEL
SPCRRM	RMFM[4:0]
SPDRCR	SPDRC[7:0]
SPPCR	MOIFE
	MOIFV
	SPLP2
	SPLP
SPCR2	SPSCKDL[2:0]
SSLP	SSL3P
	SSL2P
	SSL1P
	SSL0P
SPBR	SPR[7:0]
SPSCR	SPSLN[2:0]
SPCMD0*1	SSLA[1:0]
	SPB[4:0]
	SCKDEN
	SLNDEN
	SPNDEN
	LSBF
	SSLKP
	BRDV[1:0]
	CPOL
	CPHA

Table 7.5-16 Communication End Event Generating Conditions (Receive Only Slave Mode) (2/2)

Register Name	Bit Name
SPDCR	SPFC[1:0]
	SINV
	SPRDTD
	BYSW
SPDCR2	TTRG[1:0]
	RTRG[1:0]
SPFCR	SPFRST

Note 1. Rewriting prohibited in slave mode. In master mode, this is possible only when there is no next transfer data in the transmit FIFO.

(2) After rewriting from SPE = 1b to 0b when SPCR.BPEN = 0b, or when SPSR.MODF is set to 1b, check that the SPPSR.SPEPS bit is set to 0b, then do the next operation.

SECTION 7 LOW-SPEED INTERFACE

7.6 CRC Operation Unit (CRC)

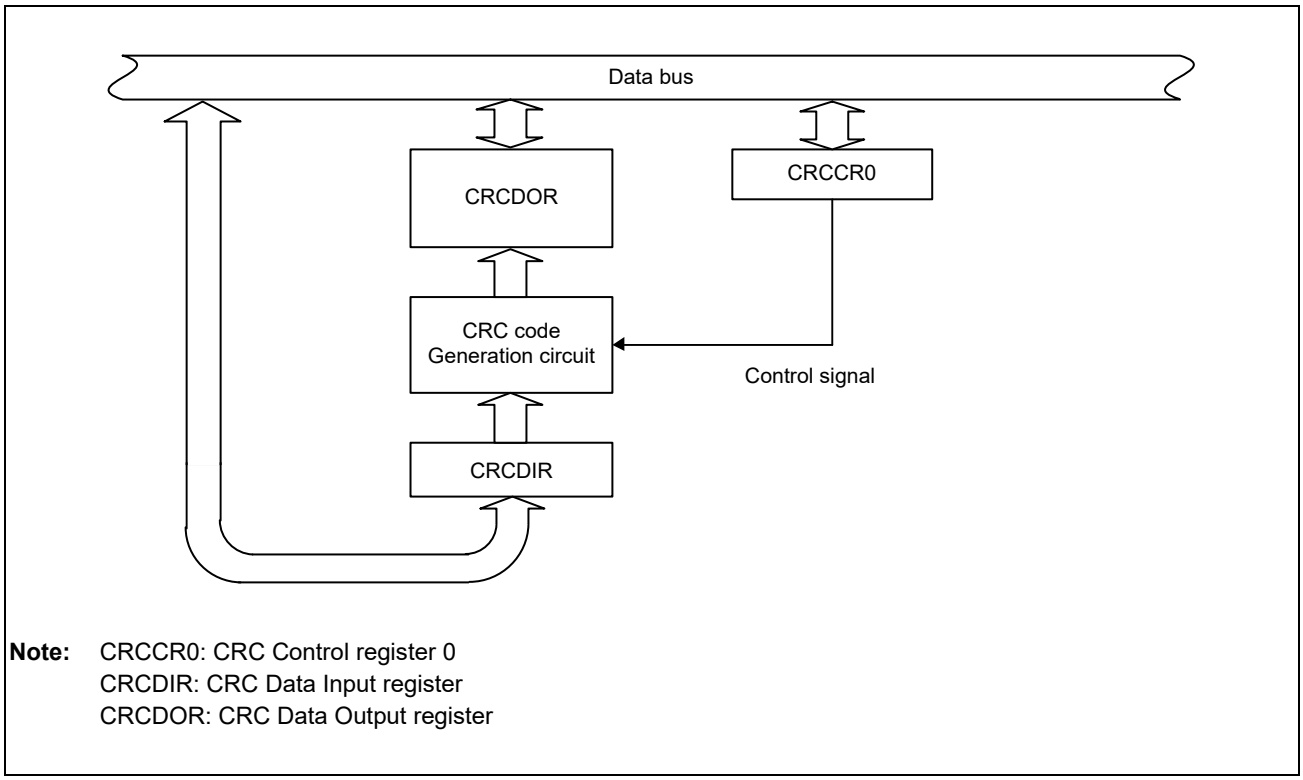
7.6.1 Overview

Cyclic redundancy check (CRC) operation units generate CRC codes. **Table 7.6-1** lists the specifications of the CRC operation unit, and **Figure 7.6-1** shows a block diagram of the CRC operation unit.

Table 7.6-1 CRC Specifications

Parameter	Description	
Number of channels	1 channel	
Data size	8 bits	32 bits
Data for CRC calculation*1	CRC codes are generated for any desired data in 8n-bit units (where n is a natural number)	CRC codes are generated for any desired data in 32n-bit units (where n is a natural number)
CRC processor unit	8-bit parallel processing	32-bit parallel processing
CRC generating polynomial	One of three generating polynomials is selectable • 8-bit CRC $X^8 + X^2 + X + 1$ (CRC-8) • 16-bit CRC $X^{16} + X^{15} + X^2 + 1$ (CRC-16) $X^{16} + X^{12} + X^5 + 1$ (CRC-CCITT)	One of two generating polynomials is selectable • 32-bit CRC $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$ (CRC-32) $X^{32} + X^{28} + X^{27} + X^{26} + X^{25} + X^{23} + X^{22} + X^{20} + X^{19} + X^{18} + X^{14} + X^{13} + X^{11} + X^{10} + X^9 + X^8 + X^6 + 1$ (CRC-32C)
CRC calculation switching	The order of the bits produced by CRC calculation can be switched for LSB first or MSB first communication	
CRC snoop	Monitoring reads from and writes to a certain register address	

Note 1. The circuit does not have a function to divide data for calculation into CRC calculation units. Write data in 8-bit or 32-bit units.



Note: CRCCR0: CRC Control register 0
CRCDIR: CRC Data Input register
CRCDOR: CRC Data Output register

Figure 7.6-1 CRC Block Diagram

7.6.2 CRC Registers

The base address for the CRC is as follows.

Table 7.6-2 Register Base Address

Base Register Name	Base Address
<CRC_base>	0_1300_0800h (5300_0800h*1, 4300_0800h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

7.6.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
CRC Control Register 0	CRCCR0	00h	0000h	8
CRC Control Register 1	CRCCR1	00h	0001h	8
Reserve	-	-	0002h to 0003h	-
CRC Data Input Register	CRCDIR	0000_0000h	0004h	32
CRC Data Output Register	CRCDOR	0000_0000h	0008h	32
Snoop Address Register	CRCSAR	0000h	000Ch	16

7.6.2.2 CRC Register Descriptions

7.6.2.2.1 CRC Control Register 0 (CRCCR0)

Access Size : 8 bits
Address : <CRC_base> + 0000h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	DORCLR	LMS	-	-	-	GPS[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	W	RW	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	DORCLR	0h	W	The read value is undefined CRCDOR Register Clear 0b: No effect 1b: Clears the CRCDOR register
6	LMS	0h	RW	CRC Calculation Switching 0b: Generates CRC for LSB first communication. 1b: Generates CRC for MSB first communication.
5 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	GPS[2:0]	0h	RW	CRC Generating Polynomial Switching 000b: No calculation is executed. 001b: 8-bit CRC-8 ($X^8 + X^2 + X + 1$) 010b: 16-bit CRC-16 ($X^{16} + X^{15} + X^2 + 1$) 011b: 16-bit CRC-CCITT ($X^{16} + X^{12} + X^5 + 1$) 100b: 32-bit CRC-32 ($X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$) 101b: 32-bit CRC-32C ($X^{32} + X^{28} + X^{27} + X^{26} + X^{25} + X^{23} + X^{22} + X^{20} + X^{19} + X^{18} + X^{14} + X^{13} + X^{11} + X^{10} + X^9 + X^8 + X^6 + 1$) Others: No calculation is executed.

GPS[2:0] bits (CRC Generating Polynomial Switching)

Set these bits to select the CRC Generating Polynomial.

LMS bit (CRC Calculation Switching)

The setting this bit selects the order of the bits of generated CRC codes. The bit selects transmission of the lower-order byte of the CRC code first for LSB first communication, or the higher-order byte first for MSB first communication. For details on the transmission and reception of CRC codes, see **7.6.3 Operation**.

DORCLR bit (CRCDOR Register Clear)

Write 1b to this bit so that the CRCDOR register is set to 0000_0000h.

7.6.2.2.2 CRC Control Register 1 (CRCCR1)

Access Size : 8 bits
Address : <CRC_base> + 0001h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	CRCSEN	CRCSWR	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	CRCSEN	0h	RW	0b: Disabled 1b: Enabled
6	CRCSWR	0h	RW	0b: Snoop-on-read 1b: Snoop-on-write
5 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

CRCSEN Bit (Snoop enable bit)

When setting this bit to 1b, CRC snoop operation is valid. When setting this bit to 0b, CRC snoop operation is invalid.

CRCSWR Bit (Snoop-on-write/read switch bit)

This bit selects the direction of the access in the address monitoring function.

When setting this bit to 0b (initial value), CRC snoop operation to reading a certain register address is valid. Similarly, when setting this bit to 1b, CRC snoop operation to writing a certain register address is valid.

7.6.2.2.3 CRC Data Input Register (CRCDIR)

CRCDIR is a readable/writable register. Write data for CRC calculation to this register.

Access Size : 32 bits
Offset Address : <CRC_base> + 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DIR[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DIR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DIR[31:0]	0h	RW	CRC Data Input

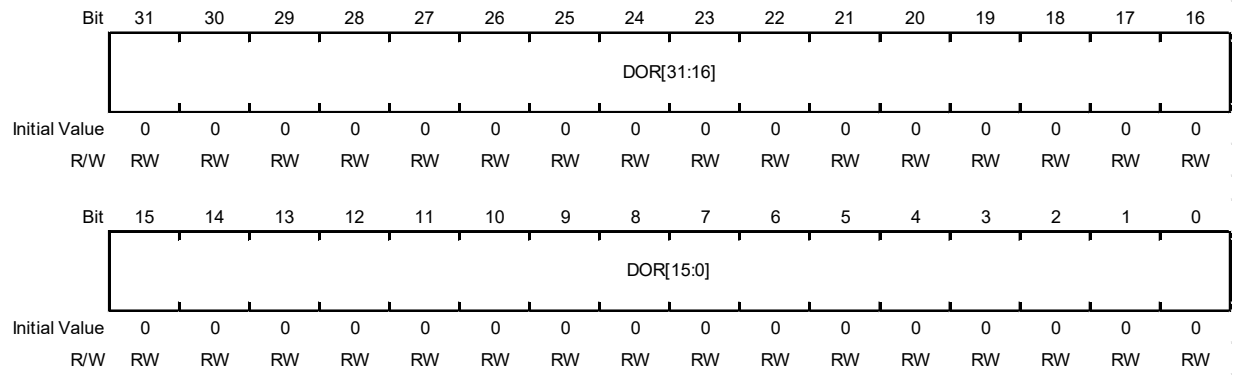
7.6.2.2.4 CRC Data Output Register (CRCDOR)

CRCDOR is a readable/writable register.

Since its initial value is 0000_0000h, rewrite the CRCDOR register to perform calculation using a value other than the initial value.

Data written to the CRCDIR register is CRC calculated and the result is stored in the CRCDOR register. If the CRC code is calculated following the transferred data and the result is 0000_0000h, there is no CRC error.

Access Size : 32 bits
Offset Address : <CRC_base> + 0008h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DOR[31:0]	0h	RW	CRC Data Output

7.6.2.2.5 Snoop Address Register (CRCSAR)

Access Size : 16 bits
Offset Address : <CRC_base> + 000Ch
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRCSAR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	CRCSAR[15:0]	0h	RW	Set the I/O register address to snoop

CRCSA Bit (register snoop address bit)

Set these bits to the certain register address for CRC snoop operation.

Only the following address can be set in CRCSA[13:0] (1300_080Ch)

0000h RSPI0.SPDR

0400h RSPI1.SPDR

0800h RSPI2.SPDR

0C00h RSCI0.RDR, 0C04h RSCI0.TDR

1000h RSCI1.RDR, 1004h RSCI1.TDR

1400h RSCI2.RDR, 1404h RSCI2.TDR

1800h RSCI3.RDR, 1804h RSCI3.TDR

1C00h RSCI4.RDR, 1C04h RSCI4.TDR

2000h RSCI5.RDR, 2004h RSCI5.TDR

2400h RSCI6.RDR, 2404h RSCI6.TDR

2800h RSCI7.RDR, 2804h RSCI7.TDR

2C00h RSCI8.RDR, 2C04h RSCI8.TDR

3000h RSCI9.RDR, 3004h RSCI9.TDR

7.6.3 Operation

7.6.3.1 Basic Operation

The CRC calculator generates CRC codes for use in LSB first or MSB first communication.

The following shows examples of generating the CRC code for input data (F0h) using the 16-bit CRC-CCITT generating polynomial ($X^{16} + X^{12} + X^5 + 1$). In these examples, the value of the CRC Data Output Register (CRCDOR_HA) is cleared before the CRC calculation (when a 16-bit CRC is in use, bits [31:16] of CRCDOR are not updated).

When an 8-bit CRC is in use, the valid bits of the CRC code is obtained in the lower-order byte of CRCDOR. When a 32-bit CRC is in use, the valid bits of the CRC code is obtained in bits [31:0] of CRCDOR.

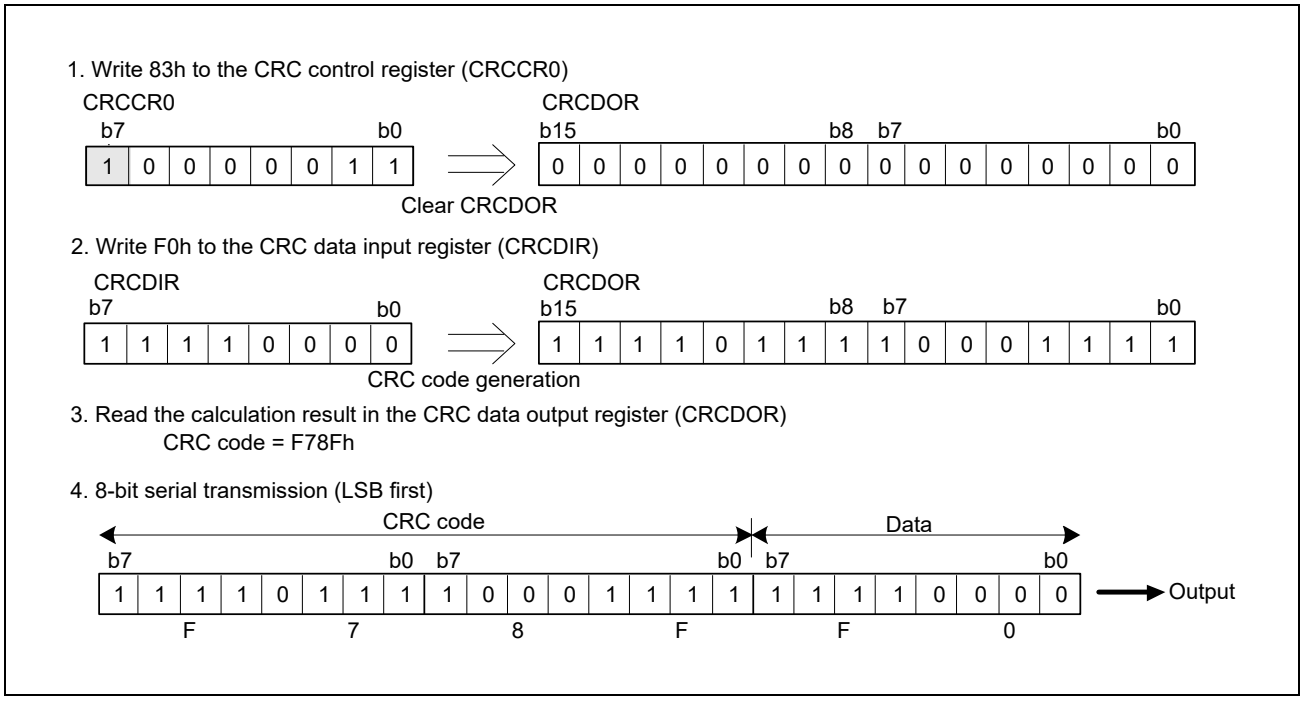


Figure 7.6-2 LSB First Data Transmission

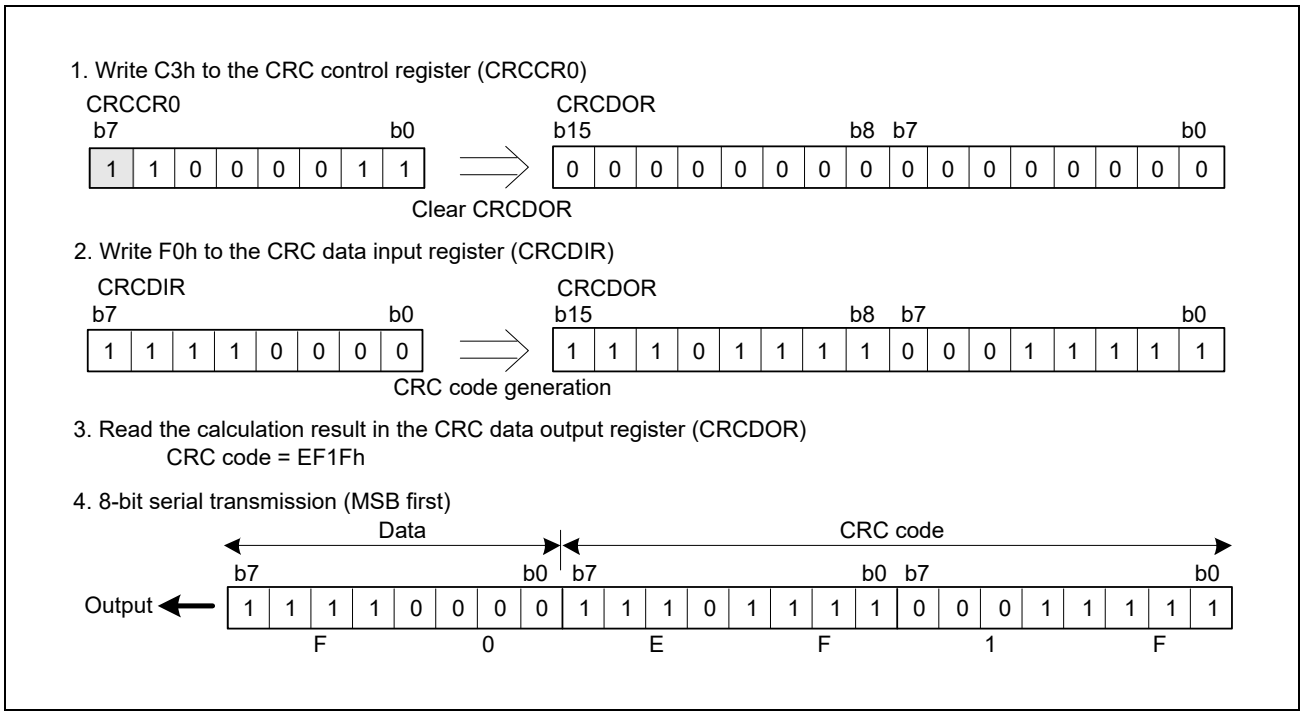


Figure 7.6-3 MSB First Data Transmission

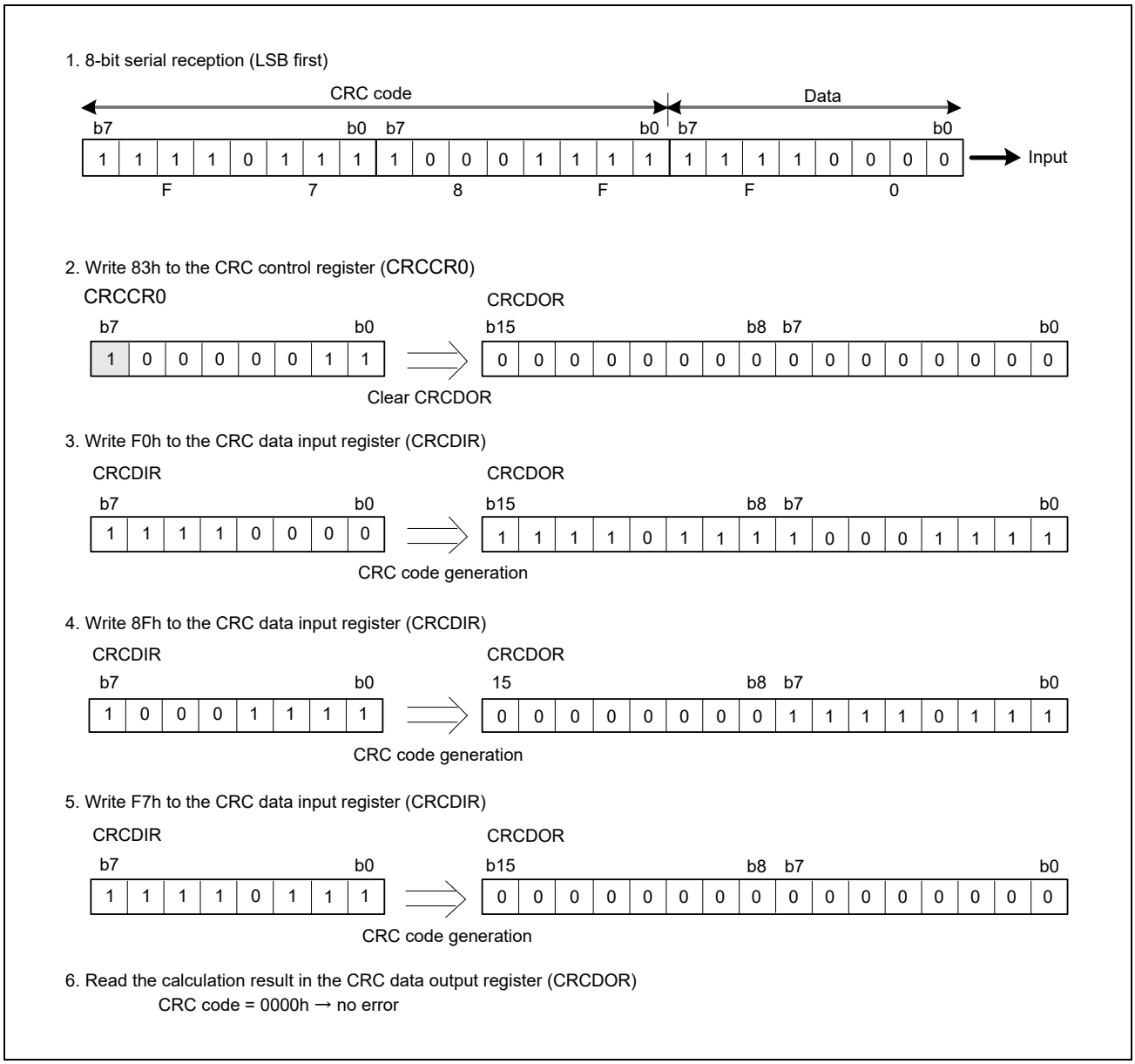


Figure 7.6-4 LSB First Data Reception

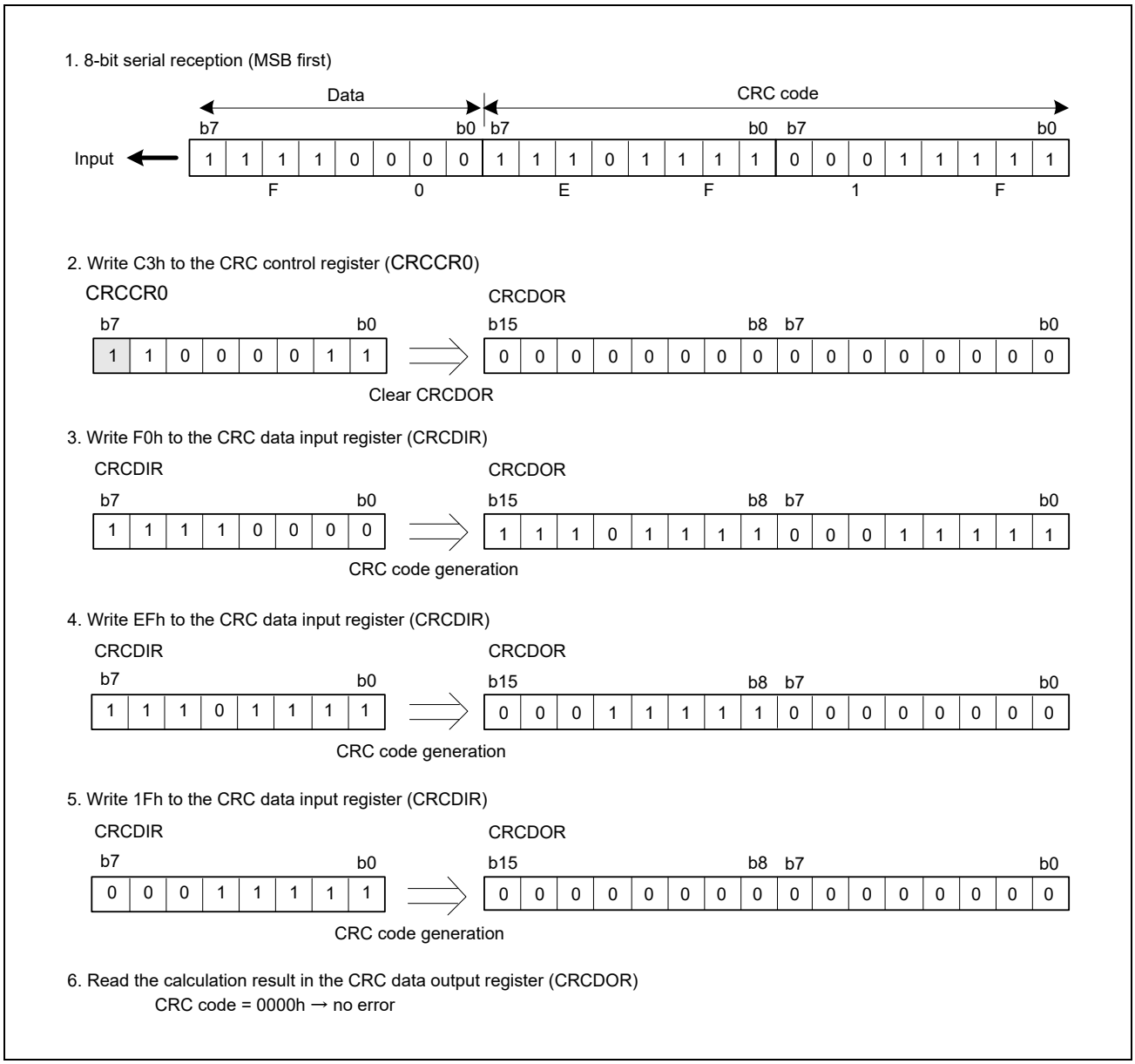


Figure 7.6-5 MSB First Data Reception

7.6.3.2 CRC Snoop

The CRC snoop monitors reads from and writes to a certain register address and performs CRC calculation on the data read from and written to the aforementioned I/O register address automatically. Because the CRC snoop recognizes writes to and reads from a certain register address as a trigger to automatically perform CRC calculation, there is no need to write data to the CRCDIR register. All I/O register addresses from 1280_0000h to 1280_33FFh are subject to the CRC snoop. The CRC snoop is useful in monitoring writes to the Serial transmit buffer, and reads from the Serial receive buffer.

To use this function, write a target I/O register address to bits CRCSA13 to CRCSA0 in the CRCSAR register, and set the CRCSEN bit in the CRCCR1 register to 1b to enable CRC snoop operation. Then, set the CRCSWR bit in the CRCCR1 register to 1b to enable snooping on writes to the target, or set the CRCSWR bit in the CRCCR1 register to 0b to enable snooping on reads from the target.

When setting the CRCSEN bit to 1b and the CRCSWR bit to 1b and writing data to a target I/O register address by the CPU or DMA, the CRC calculator stores the data in the CRCDIR register and performs CRC calculation. Similarly, when setting the CRCSEN bit to 1b and the CRCSWR bit to 0b and reading data in a target I/O register address, the CRC calculator stores the data in the CRCIN register and performs CRC calculation.

When the CRC code is generated using the CRC-8, CRC-16 and CRC-CCITT generator polynomial, the target I/O register address is accessed in bytes (8 bits). Similarly, when the CRC code is generated using the CRC-32 and CRC-32C generator polynomial, the target I/O register address is accessed in long words (32 bits).

7.6.4 Usage Notes

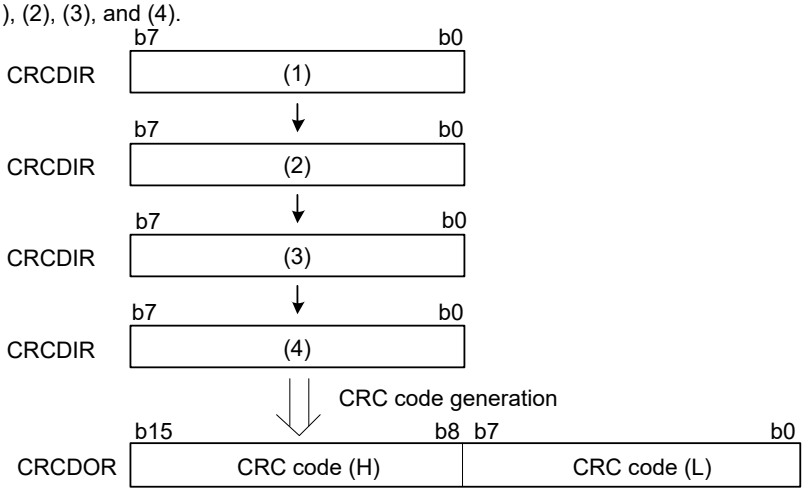
7.6.4.1 Note on Transmission

Note that the sequence of transmission for the CRC code differs according to whether transmission is LSB first or MSB first.

When transmitting 32-bit data (in case operation executed on 8 bits in parallel)

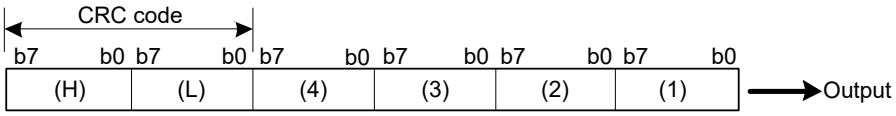
1. CRC code

After specifying the method for generation calculation, write data to the CRCDIR register in the order of (1), (2), (3), and (4).



2. Transmit data

(i) When transmission is LSB first



(ii) When transmission is MSB first

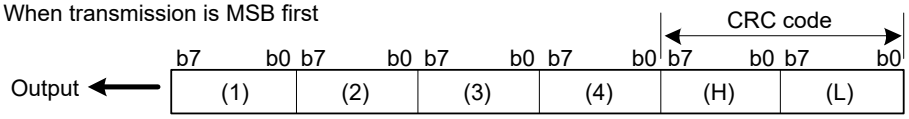


Figure 7.6-6 LSB First and MSB First Data Transmission

SECTION 7 LOW-SPEED INTERFACE

7.7 I²C Bus Interface (RIIC)

This section describes the RIIC functions of this LSI.

7.7.1 Overview

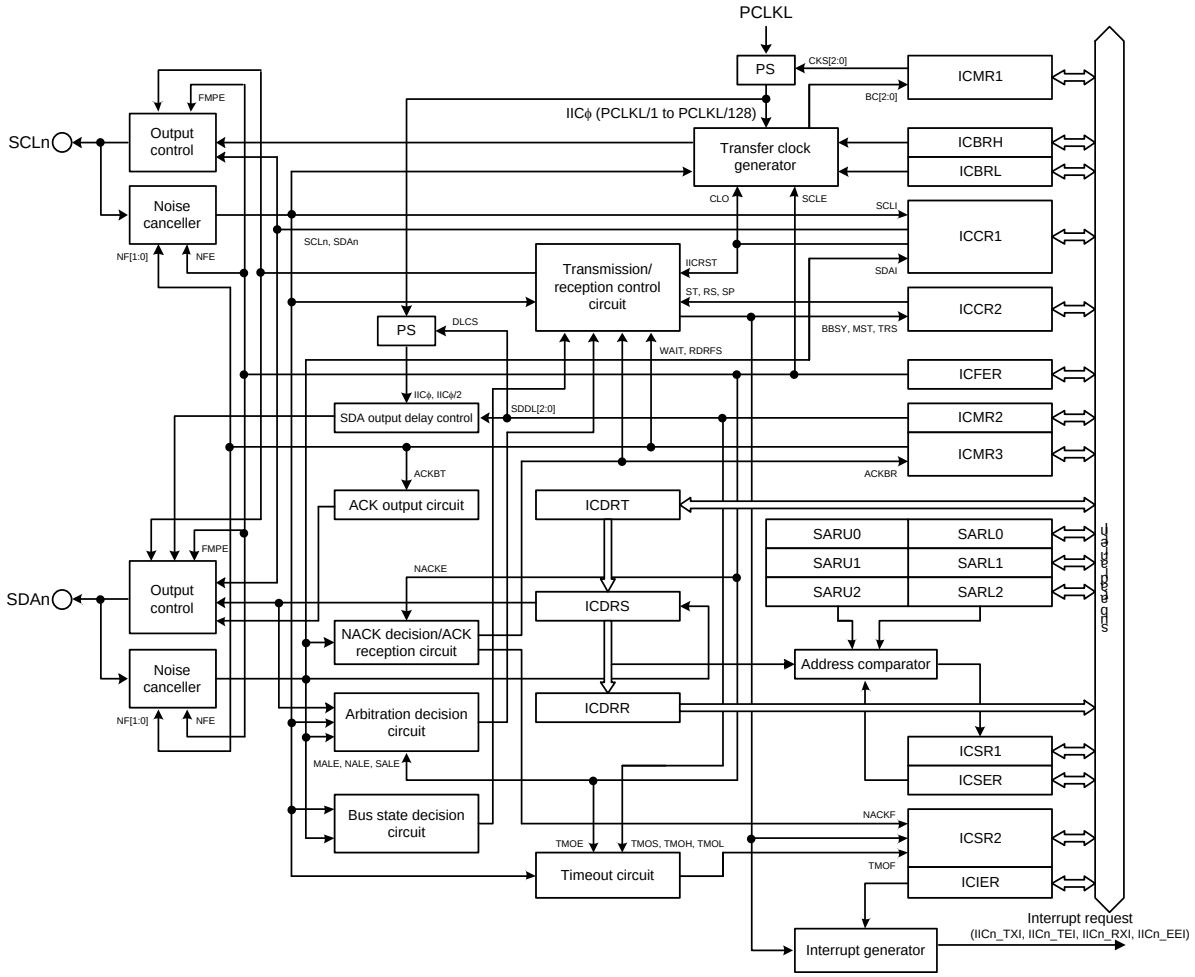
The LSI has a two-channel I²C Bus Interface (RIIC) module that conforms with and provides a subset of the NXP I²C bus (inter-integrated circuit bus) interface functions. **Table 7.7-1** lists the RIIC specifications, **Figure 7.7-1** shows a block diagram, and **Figure 7.7-2** shows an example of I/O pin connections to external circuits, with an I²C bus configuration. **Table 7.7-2** lists the I/O pins.

Table 7.7-1 RIIC Specifications (1/2)

Parameter	Specifications
Number of channels	9 channels (RIIC0-7 in the PD_OTHERS domain, RIIC8 in the PD_AWO domain)
Communications format	• I²C bus format or SMBus format. • Master or slave mode selectable. • Automatic securing of the setup times, hold times, and bus-free times for the transfer rate.
Transfer rate	Fast-mode+ supported, up to 1 Mbps.
SCL clock	For master operation, the duty cycle of the SCL clock is selectable in the range from 4% to 96%.
Issuing and detecting conditions	• Start, restart, and stop conditions are automatically generated. • Start conditions (including restart conditions) and stop conditions are detectable.
Slave address	• Configurable for up to three different slave addresses. • 7-bit and 10-bit address formats supported, including simultaneous use. • General call addresses, device ID addresses, and SMBus host addresses detectable.
Acknowledgment	• For transmission, automatic loading of the acknowledge bit. Transfer of the next transmit data can be automatically suspended on detection of a not-acknowledge bit. • For reception, automatic transmission of the acknowledge bit. If a wait between the 8th and 9th clock cycles is selected, software can control the value in the acknowledge field in response to the received value.
Wait function	During reception, the following wait periods are available by holding the SCL clock low: • Waiting between the 8th and 9th clock cycles. • Waiting between the 9th clock cycle and the 1st clock cycle of the next transfer.
SDA output delay function	Output timing of transmitted data, including the acknowledge bit, can be delayed.

Table 7.7-1 RIIC Specifications (2/2)

Parameter	Specifications
Arbitration	• For multi-master operation: – SCL clock synchronization is possible when conflict occurs with the SCL signal from another master. – When issuing the start condition creates conflict on the bus, loss of arbitration is detected by testing for mismatching between the internal signal for the SDA line and the level on the SDA line. – In master operation, loss of arbitration is detected by testing for mismatching between the signal on the SDA line and the internal signal for the SDA line. • Loss of arbitration because the start condition occurs while the bus is busy is detectable, to prevent the issuing of double start conditions. • Loss of arbitration is detectable on transfer of a not-acknowledge bit because the internal signal for the SDA line and the level on the SDA line do not match. • Loss of arbitration because mismatching of internal and line levels for data is detectable in slave transmission.
Timeout function	Internal detection of long-interval stops of the SCL clock.
Noise cancellation	• Digital noise filters for both the SCL and SDA signals. • Programmable window for noise cancellation by the filters.
Interrupt sources	• Transfer error or event generation: arbitration detection, NACK, timeout, start or restart condition, or stop condition. • Receive data full, including matching with a slave address. • Transmit data empty, including matching with a slave address. • Transmit end.
RIIC operating modes	• Master transmit • Master receive • Slave transmit • Slave receive



Note: PCLKL: RIIC_n_ckm (n = 0 to 8)

Figure 7.7-1 RIIC Block Diagram

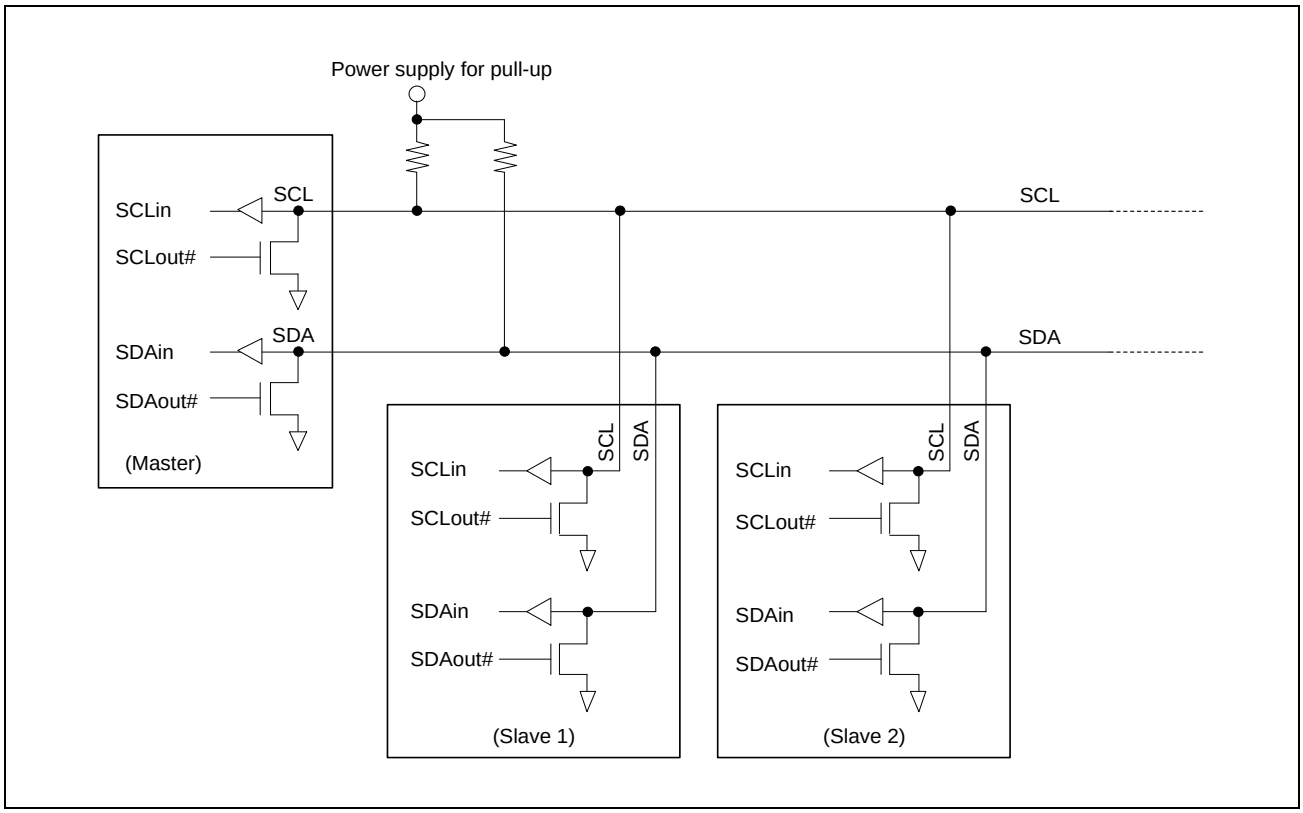


Figure 7.7-2 I/O Pin Connection to an External Circuit (I^2C Bus Configuration Example)

The input level of the signals for RIIC is CMOS. Change to TTL is not supported.

Table 7.7-2 RIIC I/O Pins

Channel	Pin Name	Input/Output	Function
RIIC0	SCL0	I/O	RIIC0 serial clock I/O pin
	SDA0	I/O	RIIC0 serial data I/O pin
RIIC1	SCL1	I/O	RIIC1 serial clock I/O pin
	SDA1	I/O	RIIC1 serial data I/O pin
RIIC2	SCL2	I/O	RIIC2 serial clock I/O pin
	SDA2	I/O	RIIC2 serial data I/O pin
RIIC3	SCL3	I/O	RIIC3 serial clock I/O pin
	SDA3	I/O	RIIC3 serial data I/O pin
RIIC4	SCL4	I/O	RIIC4 serial clock I/O pin
	SDA4	I/O	RIIC4 serial data I/O pin
RIIC5	SCL5	I/O	RIIC5 serial clock I/O pin
	SDA5	I/O	RIIC5 serial data I/O pin
RIIC6	SCL6	I/O	RIIC6 serial clock I/O pin
	SDA6	I/O	RIIC6 serial data I/O pin
RIIC7	SCL7	I/O	RIIC7 serial clock I/O pin
	SDA7	I/O	RIIC7 serial data I/O pin
RIIC8	SCL8	I/O	RIIC8 serial clock I/O pin
	SDA8	I/O	RIIC8 serial data I/O pin

Table 7.7-3 RIIC Interrupt Sources

Name	Interrupt Sources	CA55 GIC Request	CM33 GIC Request	DMAC Activation
RIIC_CHm_spi_n	Stop condition detection	Possible	Possible	Not possible
RIIC_CHm_sti_n	Start condition detection	Possible	Possible	Not possible
RIIC_CHm_naki_n	NACK detection	Possible	Possible	Not possible
RIIC_CHm_ali_n	Arbitration lost	Possible	Possible	Not possible
RIIC_CHm_tmoi_n	Timeout detection	Possible	Possible	Not possible
RIIC_CHm_ri_n	Receive data full	Possible	Possible	Possible
RIIC_CHm_ti_n	Transmit data empty	Possible	Possible	Possible
RIIC_CHm_tei_n	Transmit end	Possible	Possible	Not possible

Note: m = 0 to 8

Table 7.7-4 RIIC Event Signal Output to the ELC

Name	Event Sources
RIIC_CHm_iic_elcerrp	Transfer error or event generation
RIIC_CHm_iic_elcrdrfp	Receive data full
RIIC_CHm_iic_elctdrep	Transmit data empty
RIIC_CHm_iic_elctendp	Transmit end

Note: m = 0 to 8

7.7.2 RIIC Registers

The base addresses for the respective channels are as follows.

Table 7.7-4 Register Base Addresses

Base Register Name	RIIC ch	Base Address
<RIIC0_base>	RIIC0	0_1440_0400h (5440_0400h*1, 4440_0400h*2)
<RIIC1_base>	RIIC1	0_1440_0800h (5440_0800h*1, 4440_0800h*2)
<RIIC2_base>	RIIC2	0_1440_0C00h (5440_0C00h*1, 4440_C400h*2)
<RIIC3_base>	RIIC3	0_1440_1000h (5440_1000h*1, 4440_1000h*2)
<RIIC4_base>	RIIC4	0_1440_1400h (5440_1400h*1, 4440_1400h*2)
<RIIC5_base>	RIIC5	0_1440_1800h (5440_1800h*1, 4440_1800h*2)
<RIIC6_base>	RIIC6	0_1440_1C00h (5440_1C00h*1, 4440_1C00h*2)
<RIIC7_base>	RIIC7	0_1440_2000h (5440_2000h*1, 4440_2000h*2)
<RIIC8_base>	RIIC8	0_11C0_1000h (51C0_1000h*1, 41C0_1000h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

7.7.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
I2C Bus Control Register 1	RIICm_ICCR1	1Fh	0000h	8
I2C Bus Control Register 2	RIICm_ICCR2	00h	0001h	8
I2C Bus Mode Register 1	RIICm_ICMR1	08h	0002h	8
I2C Bus Mode Register 2	RIICm_ICMR2	06h	0003h	8
I2C Bus Mode Register 3	RIICm_ICMR3	00h	0004h	8
I2C Bus Function Enable Register	RIICm_ICFER	72h	0005h	8
I2C Bus Status Enable Register	RIICm_ICSER	09h	0006h	8
I2C Bus Interrupt Enable Register	RIICm_ICIER	00h	0007h	8
I2C Bus Status Register 1	RIICm_ICSR1	00h	0008h	8
I2C Bus Status Register 2	RIICm_ICSR2	00h	0009h	8
Slave Address Register L 0	RIICm_SARL0	00h	000Ah	8
Slave Address Register U 0	RIICm_SARU0	00h	000Bh	8
Slave Address Register L 1	RIICm_SARL1	00h	000Ch	8
Slave Address Register U 1	RIICm_SARU1	00h	000Dh	8
Slave Address Register L 2	RIICm_SARL2	00h	000Eh	8
Slave Address Register U 2	RIICm_SARU2	00h	000Fh	8
I2C Bus Bit Rate Low-Level Register	RIICm_ICBRL	FFh	0010h	8
I2C Bus Bit Rate High-Level Register	RIICm_ICBRH	FFh	0011h	8
I2C Bus Transmit Data Register	RIICm_ICDRT	FFh	0012h	8
I2C Bus Receive Data Register	RIICm_ICDRR	00h	0013h	8

7.7.2.2 RIIC Register Descriptions

The prefix (RIICm_) of the register names is omitted in this and subsequent sections.

7.7.2.2.1 I²C Bus Control Register 1 (RIICm_ICCR1)

Access Size : 8 bits
Address : <RIICm_base> + 0000h
Initial Value : 1Fh

Bit	7	6	5	4	3	2	1	0
	ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI
Initial Value	0	0	0	1	1	1	1	1
R/W	RW	RW	RW	W	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	ICE	0h	RW	IIC-Bus Interface Enable Used in combination with the IICRST bit to select either RIIC or internal reset. 0b: Disable (SCLn and SDAn pins in inactive state) 1b: Enable (SCLn and SDAn pins in active state)
6	IICRST	0h	RW	IIC-Bus Interface Internal Reset This setting clears the bit counter and the SCLn/SDAn output latch. 0b: Release RIIC reset or internal reset 1b: Initiate RIIC reset or internal reset
5	CLO	0h	RW	Extra SCL Clock Cycle Output This bit is cleared automatically after one clock cycle is output. 0b: Do not output extra SCL clock cycle (default) 1b: Output extra SCL clock cycle
4	SOWP	1h	W	The read value is undefined SCLO/SDAO Write Protect 0b: Write enable SCLO and SDAO bits 1b: Write protect SCLO and SDAO bits
3	SCLO	1h	RW	SCL Output Control/Monitor Use an external pull-up resistor to drive the signal high. 0b: Read: RIIC drives SCLn pin low Write: RIIC drives SCLn pin low 1b: Read: RIIC releases SCLn pin Write: RIIC releases SCLn pin
2	SDAO	1h	RW	SDA Output Control/Monitor Use an external pull-up resistor to drive the signal high. 0b: Read: RIIC drives SDAn pin low Write: RIIC drives SDAn pin low 1b: Read: RIIC releases SDAn pin Write: RIIC releases SDAn pin
1	SCLI	1h	R	SCL Line Monitor 0b: SCLn line is low 1b: SCLn line is high
0	SDAI	1h	R	SDA Line Monitor 0b: SDAn line is low 1b: SDAn line is high

SDAO bit (SDA Output Control/Monitor) and SCLO bit (SCL Output Control/Monitor)

The SDAO and SCLO bits directly control the SDAn and SCLn signals output from the RIIC.

When writing to these bits, also write 0b to the SOWP bit. Setting these bits results in input to the RIIC by the input buffer.

When slave mode is selected, a start condition might be detected and the bus might be released, depending on the bit settings.

Do not rewrite these bits during a start condition, stop condition, restart condition, or during transmission or reception.

Operation after rewriting under the specified conditions is not guaranteed. When reading these bits, the state of signals output from the RIIC can be read.

CLO bit (Extra SCL Clock Cycle Output)

The CLO bit allows the output of an extra SCL clock cycle for debugging or error processing. Normally, set this bit to 0b.

Setting this bit to 1 in a normal communication state causes a communication error. For details on this function, see **7.7.11.2 Extra SCL Clock Cycle Output Function**.

IICRST bit (IIC-Bus Interface Internal Reset)

The IICRST bit initiates an internal state reset of the RIIC. Setting this bit to 1b initiates an RIIC reset or internal reset.

Whether an RIIC reset or internal reset is initiated is determined by setting this bit in combination with the ICE bit.

Table 7.7-5 lists the RIIC resets.

The RIIC reset initializes all registers except ICCR1.ICE and ICCR1.IICRST bits, and internal states of the RIIC. In addition to the internal states of the RIIC, the internal reset initializes the following:

- Bit counter (ICMR1.BC[2:0] bits)
- I^2C Bus Shift Register (ICDRS)
- I^2C Bus Status Registers (ICSR1 and ICSR2)
- SDAO and SCLO Output Control/Monitor (ICCR1.SCLO and ICCR1.SDAO bits)
- I^2C Bus Control Register 2 (except ICCR2.BBSY bit)

For the reset conditions of each register, see **7.7.15 Resets, Registers, and Function States when Issuing Each Condition**.

An internal reset initiated with the IICRST bit set to 1b during operation (with the ICE bit set to 1b) resets the internal states of the RIIC without initializing the port settings and the control and setting registers of the RIIC when the bus or RIIC hangs up because of a communication error. If the RIIC hangs up in a low level output state, resetting the internal states cancels the low level output state and releases the bus with the SCLn pin and SDAn pin at high impedance.

Note: If an internal reset is initiated using the IICRST bit for a bus hang-up that occurs during communication with the master device in slave mode, the slave and master devices might enter different states, because the bit counter information differs. For this reason, do not initiate an internal reset in slave mode. Initiate recovery processing from the master device. If an internal reset is necessary because the RIIC hangs with the SCLn line in a low level output state in slave mode, initiate an internal reset, and then issue a restart condition from the master device, or issue a stop condition and resume communication from the start condition. If communication is restarted by initiating a reset solely in the slave device without issuing a start or restart condition from the master device, synchronization is lost because the master and slave devices operate asynchronously.

Table 7.7-5 RIIC Resets

IICRST	ICE	State	Specifications
1b	0b	IIC reset	Resets all registers except ICCR1.ICE and ICCR1.IICRST bits, and the internal states of the RIIC
	1b	Internal reset	Resets the following: • ICMR1.BC[2:0] bits • ICCR1, ICSR2, ICDRS registers • SDAO and SCLO Output Control/Monitor (ICCR1.SCLO and ICCR1.SDAO bits) • I^2C Bus Control Register 2 (except ICCR2.BBSY bit) • Internal states of the RIIC.

ICE bit (IIC-Bus Interface Enable)

The ICE bit selects the active or inactive state of the SCLn and SDAn pins. It can also be combined with the IICRST bit to initiate two types of resets. See **Table 7.7-5** for the reset types.

Set the ICE bit to 1b when using the RIIC. The SCLn and SDAn pins are placed in the active state when the ICE bit is set to 1b. Set the ICE bit to 0b when the RIIC is not used. The SCLn and SDAn pins are placed in the inactive state when the ICE bit is set to 0b. Do not assign the SCLn or SDAn pin to the RIIC when setting up the pin function control. Slave address comparison is performed if the pins are assigned to the RIIC.

7.7.2.2.2 I²C Bus Control Register 2 (RIICm_ICCR2)

Access Size : 8 bits
Address : <RIICm_base> + 0001h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	BBSY	MST	TRS	-	SP	RS	ST	-
Initial Value	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	BBSY	0h	R	Bus Busy Detection Flag 0b: I2C bus released (bus free state) 1b: I2C bus occupied (bus busy state)
6	MST	0h	RW	Master/Slave Mode 0b: Slave mode 1b: Master mode
5	TRS	0h	RW	Transmit/Receive Mode 0b: Receive mode 1b: Transmit mode
4	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3	SP	0h	RW	Stop Condition Issuance Request 0b: Do not issue a stop condition request 1b: Issue a stop condition request
2	RS	0h	RW	Restart Condition Issuance Request 0b: Do not issue a restart condition request 1b: Issue a restart condition request
1	ST	0h	RW	Start Condition Issuance Request 0b: Do not issue a start condition request 1b: Issue a start condition request
0	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

ST bit (Start Condition Issuance Request)

The ST bit requests transition to master mode and triggers a start condition.

When this bit is set to 1b, a start condition is issued when the BBSY flag is set to 0b (bus free state). For details on issuing a start condition, see **7.7.10 Start, Restart, and Stop Condition Issuing Function**.

[Setting condition]

- When 1b is written to the ST bit.

[Clearing conditions]

- When 0b is written to the ST bit.
- When a start condition is issued (a start condition is detected).
- When the AL (arbitration-lost) flag in ICSR2 is set to 1b.
- When 1b is written to the IICRST bit in ICCR1 to apply an RIIC reset or internal reset.

Note: Set the ST bit to 1b (start condition issuance request) when the BBSY flag is set to 0b (bus free state). Arbitration might be lost if the ST bit is set to 1b (start condition request) when the BBSY flag is 1b (bus busy state).

RS bit (Restart Condition Issuance Request)

The RS bit requests that a restart condition be issued in master mode.

When this bit is set to 1b to request a restart condition, a restart condition is issued when the BBSY flag is set to 1b (bus busy state) and the MST bit is set to 1b (master mode). For details on issuing a restart condition, see **7.7.10 Start, Restart, and Stop Condition Issuing Function**.

[Setting condition]

- When 1b is written to the RS bit with the BBSY flag in ICCR2 set to 1b.

[Clearing conditions]

- When 0b is written to the RS bit.
- When a restart condition is issued (a start condition is detected).
- When the AL (arbitration-lost) flag in ICSR2 is set to 1b.
- When 1b is written to the IICRST bit in ICCR1 to apply an RIIC reset or internal reset.

Note: Do not set the RS bit to 1b while issuing a stop condition.

Note: If 1b (restart condition request) is written to the RS bit in slave mode, the restart condition is not issued, but the RS bit remains set to 1b. If the operating mode changes to master mode without the bit being cleared, a restart condition might be issued.

SP bit (Stop Condition Issuance Request)

The SP bit requests that a stop condition be issued in master mode.

When this bit is set to 1b, a stop condition is issued when the BBSY flag is set to 1b (bus busy state) and the MST bit is set to 1b (master mode). For details on this function, see **7.7.10 Start, Restart, and Stop Condition Issuing Function**.

[Setting condition]

- When 1b is written to the SP bit with both the BBSY flag and the MST bit in ICCR2 set to 1b.

[Clearing conditions]

- When 0b is written to the SP bit.
- When a stop condition is issued (a stop condition is detected).
- When the AL (arbitration-lost) flag in ICSR2 is set to 1b.
- When a start condition and a restart condition are detected.
- When 1b is written to the IICRST bit in ICCR1 to apply an RIIC reset or internal reset.

Note: Writing to the SP bit is not possible while the BBSY flag is 0b (bus free state).

Note: Do not set the SP bit to 1b while a restart condition is being issued.

TRS bit (Transmit/Receive Mode)

The TRS bit indicates transmit or receive mode.

The RIIC is in receive mode when the TRS bit is 0b and in transmit mode when the bit is 1b. The combination of the TRS bit and the MST bit indicates the operating mode of the RIIC.

The value of the TRS bit automatically changes to 1b for transmit mode or 0b for receive mode when a start condition is issued or detected and the R/W# bit is set. Although writing to the TRS bit is possible when the MTWP bit in ICMR1 is set to 1b, writing to this bit is not required during normal usage.

[Setting conditions]

- When a start condition is issued normally because of a start condition request (when a start condition is detected with the ST bit set to 1b).
- When a restart condition is issued normally because of a restart condition request (when a restart condition is detected with the RS bit set to 1b).
- When the R/W# bit added to the slave address is set to 0b in master mode.
- When the address received in slave mode matches the address enabled in ICSER, with the R/W# bit set to 1b.
- When 1b is written to the TRS bit with the MTWP bit in ICMR1 set to 1b.

[Clearing conditions]

- When a stop condition is detected.
- When the AL (arbitration-lost) flag in ICSR2 is set to 1b.
- In master mode, on the reception of a slave address to which a R/W# bit with the value 1b is appended.
- In slave mode, on a match between the received address and the address enabled in ICSER when the value of the received R/W# bit is 0b, including when the received address is the general call address.
- In slave mode, when a restart condition is detected (a start condition is detected with ICCR2.BBSY = 1b and ICCR2.MST = 0b).
- When 0b is written to the TRS bit with the MTWP bit in ICMR1 set to 1b.
- When 1b is written to the IICRST bit in ICCR1 to apply an RIIC reset or internal reset.

MST bit (Master/Slave Mode)

The MST bit indicates master or slave mode. The RIIC is in slave mode when the MST bit is set to 0b and is in master mode when the bit is set to 1b. The combination of the MST bit and the TRS bit indicates the operating mode of the RIIC.

The value of the MST bit automatically changes to 1b for master mode or 0b for slave mode when a start condition is issued, or when a stop condition is issued or detected. Although writing to the MST bit is possible when the MTWP bit in ICMR1 is set to 1b, writing to this bit is not necessary during normal usage.

[Setting conditions]

- When a start condition is issued normally because of a start condition request (when a start condition is detected with the ST bit set to 1b).
- When 1b is written to the MST bit with the MTWP bit in ICMR1 set to 1b.

[Clearing conditions]

- When a stop condition is detected.
- When the AL (arbitration-lost) flag in ICSR2 is set to 1b.

- When 0b is written to the MST bit with the MTWP bit in ICMR1 set to 1b.
- When 1b is written to the IICRST bit in ICCR1 to apply an RIIC reset or internal reset.

BBSY bit (Bus Busy Detection Flag)

The BBSY flag indicates whether the I^2C bus is occupied (bus busy state) or released (bus free state).

The flag is set to 1b when the SDAn line changes from high to low when the SCLn line is high, assuming that a start condition was issued.

The flag is set to 0b when the SDAn line changes from low to high with the SCLn line high, if the bus free time (ICBRL setting) start condition is not detected, assuming that a stop condition was issued.

[Setting condition]

- When a start condition is detected.

[Clearing conditions]

- When the bus free time (ICBRL setting) start condition is not detected after detecting a stop condition.
- When 1b is written to the IICRST bit in ICCR1 with the ICE bit in ICCR1 set to 0b (RIIC reset).

7.7.2.2.3 I²C Bus Mode Register 1 (RIICm_ICMR1)

Access Size : 8 bits
Address : <RIICm_base> + 0002h
Initial Value : 08h

Bit	7	6	5	4	3	2	1	0
	MTWP	CKS[2:0]			BCWP	BC[2:0]		
Initial Value	0	0	0	0	1	0	0	0
R/W	RW	RW	RW	RW	W	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	MTWP	0h	RW	MST/TRS Write Protect 0b: Write protect MST and TRS bits in ICCR2 1b: Write enable MST and TRS bits in ICCR2
6 to 4	CKS[2:0]	0h	RW	Internal Reference Clock Select Select the internal reference clock source (IICΦ) for the IIC. 000b: PCLKL clock 001b: PCLKL/2 clock 010b: PCLKL/4 clock 011b: PCLKL/8 clock 100b: PCLKL/16 clock 101b: PCLKL/32 clock 110b: PCLKL/64 clock 111b: PCLKL/128 clock
3	BCWP	-	W	The read value is undefined BC Write Protect 0b: Write enable BC[2:0] bits 1b: Write protect BC[2:0] bits
2 to 0	BC[2:0]	0h	RW	Bit Counter 000b: 9 bits 001b: 2 bits 010b: 3 bits 011b: 4 bits 100b: 5 bits 101b: 6 bits 110b: 7 bits 111b: 8 bits

BC[2:0] bits (Bit Counter)

The BC[2:0] bits function as a counter that indicates the number of bits remaining to be transferred on detection of a rising edge on the SCLn line. Although BC[2:0] are read/write bits, it is not required to access these bits under normal conditions.

To write to these bits, specify the number of bits to be transferred plus one, for an additional acknowledge bit, between transferred frames when the SCLn line is at a low level.

The value in the BC[2:0] bits return to 000b at the end of a data transfer, including the acknowledge bit, or when a start or restart condition is detected.

7.7.2.2.4 I²C Bus Mode Register 2 (RIICm_ICMR2)

Access Size : 8 bits
Address : <RIICm_base> + 0003h
Initial Value : 06h

Bit	7	6	5	4	3	2	1	0
	DLCS	SDDL[2:0]			-	TMOH	TMOL	TMOS
Initial Value	0	0	0	0	0	1	1	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	DLCS	0h	RW	SDA Output Delay Clock Source Select 0b: Select internal reference clock (IICΦ) as clock source for SDA output delay counter 1b: Select internal reference clock divided by 2 (IICΦ/2) as clock source for SDA output delay counter*1
6 to 4	SDDL[2:0]	0h	RW	SDA Output Delay Counter 000b: No output delay 001b: 1 IICΦ cycle (When ICMR2.DLCS = 0 (IICΦ)) 1 or 2 IICΦ cycles (When ICMR2.DLCS = 1 (IICΦ/2)) 010b: 2 IICΦ cycles (When ICMR2.DLCS = 0 (IICΦ)) 3 or 4 IICΦ cycles (When ICMR2.DLCS = 1 (IICΦ/2)) 011b: 3 IICΦ cycles (When ICMR2.DLCS = 0 (IICΦ)) 5 or 6 IICΦ cycles (When ICMR2.DLCS = 1 (IICΦ/2)) 100b: 4 IICΦ cycles (When ICMR2.DLCS = 0 (IICΦ)) 7 or 8 IICΦ cycles (When ICMR2.DLCS = 1 (IICΦ/2)) 101b: 5 IICΦ cycles (When ICMR2.DLCS = 0 (IICΦ)) 9 or 10 IICΦ cycles (When ICMR2.DLCS = 1 (IICΦ/2)) 110b: 6 IICΦ cycles (When ICMR2.DLCS = 0 (IICΦ)) 11 or 12 IICΦ cycles (When ICMR2.DLCS = 1 (IICΦ/2)) 111b: 7 IICΦ cycles (When ICMR2.DLCS = 0 (IICΦ)) 13 or 14 IICΦ cycles (When ICMR2.DLCS = 1 (IICΦ/2))
3	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	TMOH	1h	RW	Timeout H Count Control 0b: Disable count while SCLn line is high 1b: Enable count while SCLn line is high
1	TMOL	1h	RW	Timeout L Count Control 0b: Disable count while SCLn line is low 1b: Enable count while SCLn line is low
0	TMOS	0h	RW	Timeout Detection Time Select 0b: Select long mode 1b: Select short mode

Note 1. The setting DLCS = 1 (IICϕ/2) is only valid when SCL is low. When SCL is high, the DLCS = 1 setting becomes invalid and the clock source becomes the internal reference clock (IICϕ).

TMOS bit (Timeout Detection Time Select)

The TMOS bit selects long or short mode for the timeout detection time when the timeout function is enabled (ICFER.TMOE = 1b).

When this bit is set to 0b, long mode is selected. When it is set to 1b, short mode is selected.

In long mode, the timeout detection internal counter functions as a 16 bit-counter. In short mode, the counter functions as a 14 bit counter. While the SCLn line is in the state that enables this counter as specified in the TMOH and TMOL bits, the counter counts up in synchronization with the internal reference clock (IICϕ) as a count source.

For details on this function, see **7.7.11.1 Timeout Function**.

TMOL bit (Timeout L Count Control)

The TMOL bit enables or disables up-counting on the internal counter of the timeout function while the SCLn line is held low and the timeout function is enabled (ICFER.TMOE = 1b).

TMOH bit (Timeout H Count Control)

The TMOH bit enables or disables up-counting on the internal counter of the timeout function while the SCLn line is held high and the timeout function is enabled (ICFER.TMOE = 1).

SDDL[2:0] bits (SDA Output Delay Counter)

The SDDL[2:0] bits can be used to delay the SDA output. This counter works with the clock source selected in the DLCS bit. This setting can be used for all types of SDA output, including the transmission of the acknowledge bit.

Set the SDA output delay to meet the I^2C bus standard for the data enable time/acknowledge enable time,*1 or the SMBus standard, within [data hold time (300 ns or more + the SCL-clock low-level period) - the data setup time (250 ns)]. If a value outside the standard is set, communication between the devices might malfunction or falsely indicate a start or stop condition, depending on the bus state.

For details on this function, see **7.7.5 SDA Output Delay Function**.

Note 1. Data enable time/acknowledge enable time
3,450 ns for up to 100 kbps: Standard-mode (Sm)
900 ns for up to 400 kbps: Fast-mode (Fm)
450 ns for up to 1 Mbps: Fast-mode plus (Fm+)

7.7.2.2.5 I²C Bus Mode Register 3 (RIICm_ICMR3)

Access Size : 8 bits
Address : <RIICm_base> + 0004h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	SMBS	WAIT	RDRFS	ACKWP	ACKBT	ACKBR	NF[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	SMBS	0h	RW	SMBus/IIC-Bus Select 0b: Select I2C bus 1b: Select SMBus
6	WAIT	0h	RW	WAIT Low-hold is released by reading ICDRR. 0b: No wait. SCLn is not held low during the period between 9th clock cycle and 1st clock cycle. 1b: Wait. SCLn is held low during the period between 9th clock cycle and 1st clock cycle.
5	RDRFS	0h	RW	RDRF Flag Set Timing Select Low-hold is released by writing to ACKBT. 0b: Set the RDRF flag on the rising edge of the 9th SCL clock cycle. The SCLn line is not held low on the falling edge of the 8th clock cycle. 1b: Set the RDRF flag on the rising edge of the 8th SCL clock cycle. The SCLn line is held low on the falling edge of the 8th clock cycle.
4	ACKWP	0h	RW	ACKBT Write Protect 0b: Write protect ACKBT bit 1b: Write enable ACKBT bit
3	ACKBT	0h	RW	Transmit Acknowledge 0b: Send 0 as the acknowledge bit (ACK transmission) 1b: Send 1 as the acknowledge bit (NACK transmission)
2	ACKBR	0h	R	Receive Acknowledge 0b: 0 received as the acknowledge bit (ACK reception) 1b: 1 received as the acknowledge bit (NACK reception)
1, 0	NF[1:0]	0h	RW	Noise Filter Stage Select 00b: Filter out noise of up to 1 IICΦ cycle (single-stage filter) 01b: Filter out noise of up to 2 IICΦ cycles (2-stage filter) 10b: Filter out noise of up to 3 IICΦ cycles (3-stage filter) 11b: Filter out noise of up to 4 IICΦ cycles (4-stage filter)

NF[1:0] bits (Noise Filter Stage Select)

The NF[1:0] bits select the number of stages in the digital noise filter. For details on this function, see **7.7.6 Digital Noise Filter Circuits**.

Note: Set the noise range to be filtered out by the noise filter within a range less than the SCLn line high-level period or low-level period. If the noise range is set to a value of [SCL clock width: high-level period or low-level period, whichever is shorter] - [1.5 internal reference clock (IICϕ) cycles] or more, the SCL clock is regarded as noise by the noise filter function of the RIIC, which might prevent the RIIC from operating normally.

ACKBR bit (Receive Acknowledge)

The ACKBR bit stores the acknowledge bit information received from the receive device in transmit mode.

[Setting condition]

- When 1b is received as the acknowledge bit with the TRS bit in ICCR2 set to 1b.

[Clearing conditions]

- When 0b is received as the acknowledge bit with the TRS bit in ICCR2 set to 1b.
- When 1b is written to the IICRST bit in ICCR1 while the ICE bit in ICCR1 is 0b (IIC reset).

ACKBT bit (Transmit Acknowledge)

The ACKBT bit sets the acknowledge bit to be sent in receive mode.

[Setting condition]

- When 1b is written to this bit with the ACKWP bit set to 1b.

[Clearing conditions]

- When 0b is written to this bit with the ACKWP bit set to 1b.
- When stop condition issuance is detected with the SP bit in ICCR2 set to 1b.
- When 1b is written to the IICRST bit in ICCR1 while the ICE bit in ICCR1 is 0b (IIC reset).

ACKWP bit (ACKBT Write Protect)

The ACKWP bit controls write enabling of the ACKBT bit.

RDRFS bit (RDRF Flag Set Timing Select)

The RDRFS bit selects the RDRF flag set timing in receive mode and also selects whether to hold the SCLn line low on the falling edge of the 8th SCL clock cycle.

When the RDRFS bit is 0b, the SCLn line is not held low on the falling edge of the 8th SCL clock cycle, and the RDRF flag is set to 1b on the rising edge of the 9th SCL clock cycle.

When the RDRFS bit is 1b, the RDRF flag is set to 1b on the rising edge of the 8th SCL clock cycle, and the SCLn line is held low on the falling edge of the 8th SCL clock cycle. The low-hold of the SCLn line is released by a write to the ACKBT bit.

After data is received with this setting, the SCLn line is automatically held low before the acknowledge bit is sent. This enables processing to send ACK (ACKBT = 0b) or NACK (ACKBT = 1b), based on the receive data.

WAIT bit (WAIT)

The WAIT bit controls whether to forcefully hold the period between the 9th SCL clock cycle and the 1st SCL clock cycle low until the receive data buffer (ICDRR) is completely read each time a single-byte data is received in receive mode.

When the WAIT bit is 0b, the receive operation is continued without holding the period between the 9th and the 1st SCL clock cycle low. When both the RDRFS and WAIT bits are 0b, continuous receive operation is enabled with the double buffer.

When the WAIT bit is 1b, the SCLn line is held low from the falling edge of the 9th clock cycle until the ICDRR value is read each time a single-byte data is received. This enables the receive operation in byte units.

Note: When the value of the WAIT bit is to be read, be sure to first read the ICDRR.

SMBS bit (SMBus/IIC-Bus Select)

Setting the SMBS bit to 1b selects the SMBus and enables the HOAE bit in ICSER.

7.7.2.2.6 I²C Bus Function Enable Register (RIICm_ICFER)

Access Size : 8 bits
Address : <RIICm_base> + 0005h
Initial Value : 72h

Bit	7	6	5	4	3	2	1	0
	FMPE	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE
Initial Value	0	1	1	1	0	0	1	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	FMPE	0h	RW	Fast-mode Plus Enable 0b: No Fm+ slope control circuit is used for the SCLn pin and SDAn pin. 1b: An Fm+ slope control circuit is used for the SCLn pin and SDAn pin.
6	SCLE	1h	RW	SCLn Synchronous Circuit Enable 0b: Do not use the SCLn synchronous circuit 1b: Use the SCLn synchronous circuit
5	NFE	1h	RW	Digital Noise Filter Circuit Enable 0b: Do not use the digital noise filter circuit 1b: Use the digital noise filter circuit
4	NACKE	1h	RW	NACK Reception Transfer Suspension Enable 0b: Do not suspend transfer operation during NACK reception (disable transfer suspension) 1b: Suspend transfer operation during NACK reception (enable transfer suspension)
3	SALE	0h	RW	Slave Arbitration-Lost Detection Enable 0b: Disable 1b: Enable
2	NALE	0h	RW	NACK Transmission Arbitration-Lost Detection Enable 0b: Disable 1b: Enable
1	MALE	1h	RW	Master Arbitration-Lost Detection Enable 0b: Disable the arbitration-lost detection function and disable automatic clearing of the MST and TRS bits in ICCR2 when arbitration is lost 1b: Enable the arbitration-lost detection function and enable automatic clearing of the MST and TRS bits in ICCR2 when arbitration is lost
0	TMOE	0h	RW	Timeout Function Enable 0b: Disable 1b: Enable

TMOE bit (Timeout Function Enable)

The TMOE bit enables or disables the timeout function. For details on this function, see **7.7.11.1 Timeout Function**.

MALE bit (Master Arbitration-Lost Detection Enable)

The MALE bit specifies whether to use the arbitration-lost detection function in master mode. For normal operation, set this bit to 1.

NALE bit (NACK Transmission Arbitration-Lost Detection Enable)

The NALE bit specifies whether to cause loss of arbitration when ACK is detected during the transmission of NACK in receive mode, for example, when slaves with the same address exist on the bus, or when two or more masters select the same slave device simultaneously with a different number of receive bytes.

SALE bit (Slave Arbitration-Lost Detection Enable)

The SALE bit specifies whether to cause loss of arbitration when a value different from the value being transmitted is detected on the bus in slave transmit mode, for example, when slaves with the same address exist on the bus, or when a mismatch with the transmit data occurs because of noise.

NACKE bit (NACK Reception Transfer Suspension Enable)

The NACKE bit specifies whether to continue or discontinue the data transfer when NACK is received in transmit mode. Normally, set this bit to 1b.

When NACK is received with the NACKE bit set to 1b, the next transfer operation is suspended. When the NACKE bit is 0b, the next transfer operation is continued regardless of the received acknowledge content.

For details, see **7.7.8.2 NACK Reception Transfer Suspension Function**.

SCLE bit (SCL Synchronous Circuit Enable)

The SCLE bit specifies whether to synchronize the SCL clock with the SCL input clock. For normal operation, set this bit to 1b.

When the SCLE bit is set to 0b (no SCL synchronous circuit used), the RIIC does not synchronize the SCL clock with the SCL input clock. With this setting, the RIIC outputs the SCL clock with the transfer rate set in ICBRH and ICBRL, regardless of the SCLn line state. For this reason, if the bus load of the I^2C bus line is much larger than the specification value, or if the SCL clock output overlaps in multiple masters, a short-cycle SCL clock that does not meet the specification might be output. When no SCL synchronous circuit is used, it also affects the issuance of the start, restart, and stop conditions, and the continuous output of extra SCL clock cycles.

Do not set this bit to 0b except when checking the output of the set transfer rate.

FMPE Bit (Fast-mode Plus Enable)

The FMPE bit specifies whether to use a slope control circuit for Fast-mode Plus [Fm+].

When the FMPE bit is set to 1b, a slope control circuit conforming to the Fast-mode Plus [Fm+] slope control standard (tof) of the I2C bus is selected. When the FMPE bit is set to 0b, a slope control circuit conforming to the Standard-mode [Sm] and Fast-mode [fm] slope control standard (tof) of the I2C bus is selected.

Set the FMPE bit to 1b when using the transmission rate within a range up to 1 Mbps (Fast-mode Plus [Fm+]) of the I2C bus standard. Set the FMPE bit to 0b when using the transmission rate at other rates (up to 100 kbps [Sm], up to 400 kbps [fm]) or for SMBus (10 to 100 kbps).

7.7.2.2.7 I²C Bus Status Enable Register (RIICm_ICSER)

Access Size : 8 bits
Address : <RIICm_base> + 0006h
Initial Value : 09h

Bit	7	6	5	4	3	2	1	0
	HOAE	-	DIDE	-	GCAE	SAR2E	SAR1E	SAR0E
Initial Value	0	0	0	0	1	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	HOAE	0h	RW	Host Address Enable 0b: Disable host address detection 1b: Enable host address detection
6	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	DIDE	0h	RW	Device ID Address Detection Enable 0b: Disable device ID address detection 1b: Enable device ID address detection
4	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3	GCAE	1h	RW	General Call Address Enable 0b: Disable general call address detection 1b: Enable general call address detection
2	SAR2E	0h	RW	Slave Address Register 2 Enable 0b: Disable slave address in SARL2 and SARU2 1b: Enable slave address in SARL2 and SARU2
1	SAR1E	0h	RW	Slave Address Register 1 Enable 0b: Disable slave address in SARL1 and SARU1 1b: Enable slave address in SARL1 and SARU1
0	SAR0E	1h	RW	Slave Address Register 0 Enable 0b: Disable slave address in SARL0 and SARU0 1b: Enable slave address in SARL0 and SARU0

SARyE bit (Slave Address Register y Enable) (y = 0 to 2)

The SARyE bit enables or disables the received slave address and the slave address set in SARLy and SARUy.

When this bit is set to 1b, the slave address set in SARLy and SARUy is enabled and is compared with the received slave address. When this bit is set to 0b, the slave address set in SARLy and SARUy is disabled and is ignored even if it matches the received slave address.

GCAE bit (General Call Address Enable)

The GCAE bit specifies whether to ignore the general call address (0000_000b + 0b [W]: All 0) when it is received.

When this bit is set to 1b, if the received slave address matches the general call address, the RIIC recognizes the received slave address as the general call address independently of the slave addresses set in SARLy and SARUy (y = 0 to 2), and performs the data receive operation. When this bit is set to 0b, the received slave address is ignored even if it matches the general call address.

DIDE bit (Device ID Address Detection Enable)

The DIDE bit specifies whether to recognize and execute the device ID address when a device ID (1111_100b) is received in the first frame after a start or restart condition is detected.

When this bit is set to 1b, if the received first frame matches the device ID, the RIIC recognizes that the device ID address was received. When the next R/W# bit is 0b (W), the RIIC recognizes the second and the subsequent frames as

slave addresses and continues the receive operation. When this bit is set to 0b, the RIIC ignores the received first frame even if it matches the device ID address, and recognizes the first frame as a normal slave address.

For details on the device ID function, see **7.7.7.3 Device ID Address Detection**.

HOAE bit (Host Address Enable)

The HOAE bit specifies whether to ignore the received host address (0001_000b) when the SMBS bit in ICMR3 is 1b.

When this bit is set to 1b while the SMBS bit in ICMR3 is 1b, if the received slave address matches the host address, the RIIC recognizes the received slave address as the host address independently of the slave addresses set in SARLy and SARUy (y = 0 to 2), and performs the receive operation.

When the SMBS bit in ICMR3 or the HOAE bit is set to 0b, the received slave address is ignored even if it matches the host address.

7.7.2.2.8 I²C Bus Interrupt Enable Register (RIICm_ICIER)

Access Size : 8 bits
Address : <RIICm_base> + 0007h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	TIE	0h	RW	Transmit Data Empty Interrupt Request Enable 0b: Disable transmit data empty interrupt (IICn_TXI) request 1b: Enable transmit data empty interrupt (IICn_TXI) request
6	TEIE	0h	RW	Transmit End Interrupt Request Enable 0b: Disable transmit end interrupt (IICn_TEI) request 1b: Enable transmit end interrupt (IICn_TEI) request
5	RIE	0h	RW	Receive Data Full Interrupt Request Enable 0b: Disable receive data full interrupt (IICn_RXI) request 1b: Enable receive data full interrupt (IICn_RXI) request
4	NAKIE	0h	RW	NACK Reception Interrupt Request Enable 0b: Disable NACK reception interrupt IICn_EEI (NACKF) request 1b: Enable NACK reception interrupt IICn_EEI (NACKF) request
3	SPIE	0h	RW	Stop Condition Detection Interrupt Request Enable 0b: Disable stop condition detection interrupt IICn_EEI (STOP) request 1b: Enable stop condition detection interrupt IICn_EEI (STOP) request
2	STIE	0h	RW	Start Condition Detection Interrupt Request Enable 0b: Disable start condition detection interrupt IICn_EEI (START) request 1b: Enable start condition detection interrupt IICn_EEI (START) request
1	ALIE	0h	RW	Arbitration-Lost Interrupt Request Enable 0b: Disable arbitration-lost interrupt IICn_EEI (AL) request 1b: Enable arbitration-lost interrupt IICn_EEI (AL) request
0	TMOIE	0h	RW	Timeout Interrupt Request Enable 0b: Disable timeout interrupt IICn_EEI (TMOF) request 1b: Enable timeout interrupt IICn_EEI (TMOF) request

TMOIE bit (Timeout Interrupt Request Enable)

The TMOIE bit enables or disables timeout interrupt IICn_EEI (TMOF) requests when the TMOF flag in ICSR2 is 1b. To cancel a IICn_EEI (TMOF) interrupt request, set the TMOF flag or the TMOIE bit to 0b.

ALIE bit (Arbitration-Lost Interrupt Request Enable)

The ALIE bit enables or disables arbitration-lost interrupt IICn_EEI (AL) requests when the AL flag in ICSR2 is 1b. To cancel an IICn_EEI (AL) interrupt request, set the AL flag or the ALIE bit to 0b.

STIE bit (Start Condition Detection Interrupt Request Enable)

The STIE bit enables or disables start condition detection interrupt IICn_EEI (START) requests when the START flag in ICSR2 is 1b. To cancel an IICn_EEI (START) interrupt request, set the START flag or the STIE bit to 0b.

SPIE bit (Stop Condition Detection Interrupt Request Enable)

The SPIE bit enables or disables stop condition detection interrupt IICn_EEI (STOP) requests when the STOP flag in ICSR2 is 1b. To cancel an IICn_EEI (STOP) interrupt request, set the STOP flag or the SPIE bit to 0b.

NAKIE bit (NACK Reception Interrupt Request Enable)

The NAKIE bit enables or disables NACK reception interrupt IICn_EEI (NACKF) requests when the NACKF flag in ICSR2 is 1. To cancel an IICn_EEI (NACKF) interrupt request, set the NACKF flag or the NAKIE bit to 0b.

RIE bit (Receive Data Full Interrupt Request Enable)

The RIE bit enables or disables receive data full interrupt (IICn_RXI) requests when the RDRF flag in ICSR2 is 1b.

TEIE bit (Transmit End Interrupt Request Enable)

The TEIE bit enables or disables transmit end interrupt (IICn_TEI) requests when the TEND flag in ICSR2 is 1b. To cancel an IICn_TEI interrupt request, set the TEND flag or the TEIE bit to 0b.

TIE bit (Transmit Data Empty Interrupt Request Enable)

The TIE bit enables or disables transmit data empty interrupt (IICn_TXI) requests when the TDRE flag in ICSR2 is 1b.

7.7.2.2.9 I²C Bus Status Register 1 (RIICm_ICSR1)

Access Size : 8 bits
Address : <RIICm_base> + 0008h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	HOA	-	DID	-	GCA	AAS2	AAS1	AAS0
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	HOA	0h	RW	Host Address Detection Flag This bit is set to 1b when the received slave address matches the host address (0001_000b). 0b: Host address not detected 1b: Host address detected
6	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	DID	0h	RW	Device ID Address Detection Flag This bit is set to 1 when the first frame received immediately after a start condition is detected matches a value of (device ID (1111_100b) + 0 [W]). 0b: Device ID command not detected 1b: Device ID command detected
4	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3	GCA	0h	RW	General Call Address Detection Flag 0b: General call address not detected 1b: General call address detected
2	AAS2	0h	RW	Slave Address 2 Detection Flag 0b: Slave address 2 not detected 1b: Slave address 2 detected
1	AAS1	0h	RW	Slave Address 1 Detection Flag 0b: Slave address 1 not detected 1b: Slave address 1 detected
0	AAS0	0h	RW	Slave Address 0 Detection Flag 0b: Slave address 0 not detected 1b: Slave address 0 detected

AASy bit (Slave Address y Detection Flag) (y = 0 to 2)

The AASy flag indicates whether slave address y was detected.

[Setting conditions]

For 7-bit address format (SARUy.FS = 0b):

- When the received slave address matches the SVA[6:0] value in SARLy, with the SARyE bit in ICSER set to 1 (slave address y detection enabled).
  The AASy flag is set to 1b on the rising edge of the 9th SCL clock cycle in the frame.

For 10-bit address format: (SARUy.FS = 1b):

- When the received slave address matches a value of (11110b + SVA[1:0] in SARUy), and the subsequent address matches the SARLy value, with the SARyE bit in ICSER set to 1b (slave address y detection enabled).
  The AASy flag is set to 1b on the rising edge of the 9th SCL clock cycle in the frame.

[Clearing conditions]

- When 0b is written to the AASy flag after reading AASy = 1b
- When a stop condition is detected.
- When 1b is written to the IICRST bit in ICCR1 to apply an RIIC reset or internal reset.

For 7-bit address format (SARUy.FS = 0b):

- When the received slave address does not match the SVA[6:0] value in SARLy, with the SARyE bit in ICSER set to 1b (slave address y detection enabled).
  The AASy flag is set to 0b on the rising edge of the 9th SCL clock cycle in the frame.

For 10-bit address format (SARUy.FS = 1b):

- When the received slave address does not match a value of 11110b + SVA[1:0] in SARUy, with the SARyE bit in ICSER set to 1b (slave address y detection enabled).
  The AASy flag is set to 0b on the rising edge of the 9th SCL clock cycle in the frame.
- When the received slave address matches a value of 11110b + SVA[1:0] in SARUy, and the subsequent address does not match the SARLy value, with the SARyE bit in ICSER set to 1b (slave address y detection enabled).
  The AASy flag is set to 0b on the rising edge of the 9th SCL clock cycle in the frame.

GCA bit (General Call Address Detection Flag)

The GCA flag indicates whether the general call address was detected.

[Setting condition]

- When the received slave address matches the general call address (0000_000b + 0b [W]), with the GCAE bit in ICSER set to 1b (general call address detection enabled).
  The GCA flag is set to 1b on the rising edge of the 9th SCL clock cycle in the frame.

[Clearing conditions]

- When 0b is written to the GCA flag after reading GCA = 1b.
- When a stop condition is detected.
- When the received slave address does not match the general call address (0000_000b + 0b [W]), with the GCAE bit in ICSER set to 1b (general call address detection enabled).
  The GCA flag is set to 0b on the rising edge of the 9th SCL clock cycle in the frame.
- When 1b is written to the IICRST bit in ICCR1 to apply an RIIC reset or internal reset.

DID bit (Device ID Address Detection Flag)

The DID flag indicates whether the device ID address was detected.

[Setting condition]

- When the first frame received immediately after a start or restart condition is detected matches a value of (device ID (1111_100b) + 0b [W]), with the DIDE bit in ICSER set to 1b (device ID address detection enabled).
  The DID flag is set to 1b on the rising edge of the 9th SCL clock cycle in the frame.

[Clearing conditions]

- When 0b is written to the DID flag after reading DID = 1b.
- When a stop condition is detected.
- When the first frame received immediately after a start or restart condition is detected does not match a value of the device ID (1111_100b), with the DIDE bit in ICSER set to 1b (device ID address detection enabled).
  The DID flag is set to 0b on the rising edge of the 9th SCL clock cycle in the frame.
- When the first frame received immediately after a start or restart condition is detected matches a value of (device ID (1111_100b) + 0b [W]), and the second frame does not match any slave address from 0 to 2, with the DIDE bit in ICSER set to 1b (device ID address detection enabled).
  The DID flag is set to 0b on the rising edge of the 9th SCL clock cycle in the frame.

- When 1b is written to the IICRST bit in ICCR1 to initiate an RIIC reset or internal reset.

HOA bit (Host Address Detection Flag)

The HOA flag indicates whether the host address was detected.

[Setting condition]

- When the received slave address matches the host address (0001 000b), with the HOAE bit in ICSER set to 1b (host address detection enabled).
  The HOA flag is set to 1b on the rising edge of the 9th SCL clock cycle in the frame.

[Clearing conditions]

- When 0b is written to the HOA flag after reading HOA = 1b.
- When a stop condition is detected.
- When the received slave address does not match the host address (0001 000b), with the HOAE bit in ICSER set to 1b (host address detection enabled).
  The HOA flag is set to 0b on the rising edge of the 9th SCL clock cycle in the frame.
- When 1b is written to the IICRST bit in ICCR1 to initiate an RIIC reset or internal reset.

7.7.2.2.10 I²C Bus Status Register 2 (RIICm_ICSR2)

Access Size : 8 bits
Address : <RIICm_base> + 0009h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	TDRE	TEND	RDRF	NACKF	STOP	START	AL	TMOF
Initial Value	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	TDRE	0h	R	Transmit Data Empty Flag 0b: ICDRT contains transmit data 1b: ICDRT contains no transmit data
6	TEND	0h	RW	Transmit End Flag 0b: Data being transmitted 1b: Data transmit complete
5	RDRF	0h	RW	Receive Data Full Flag 0b: ICDRR contains no receive data 1b: ICDRR contains receive data
4	NACKF	0h	RW	NACK Detection Flag 0b: NACK not detected 1b: NACK detected
3	STOP	0h	RW	Stop Condition Detection Flag 0b: Stop condition not detected 1b: Stop condition detected
2	START	0h	RW	Start Condition Detection Flag 0b: Start condition not detected 1b: Start condition detected
1	AL	0h	RW	Arbitration-Lost Flag 0b: Arbitration not lost 1b: Arbitration lost
0	TMOF	0h	RW	Timeout Detection Flag 0b: Timeout not detected 1b: Timeout detected.

TMOF bit (Timeout Detection Flag)

The TMOF flag is set to 1 when RIIC detects a timeout because the SCLn line state remains unchanged for the set period.

[Setting condition]

- When the SCLn line state remains unchanged for the period specified in the ICMR2.TMOH, TMOL, and TMOS bits while the ICFER.TMOE bit is 1b (timeout function enabled) in master or in slave mode and the received slave address matches.

[Clearing conditions]

- When 0b is written to the TMOF flag after reading TMOF = 1b.
- When 1b is written to the IICRST bit in ICCR1 to initiate an RIIC reset or internal reset.

AL bit (Arbitration-Lost Flag)

The AL flag indicates that the bus mastership is lost in arbitration because of a bus conflict or because of some other reason, when a start condition was issued, or an address and data was transmitted.

The RIIC monitors the level on the SDAn line during transmission. If the level on the line does not match the value of the bit being output, the RIIC sets the AL flag to 1b to indicate that the bus is occupied by another device. The RIIC can also detect loss of arbitration during NACK transmission in receive mode or during data transmission in slave mode.

[Setting conditions]

When master arbitration-lost detection is enabled (ICFER.MALE = 1b):

- When the internal SDA output state does not match the SDAn line level on the rising edge of the SCL clock except for the ACK period during data transmission in master transmit mode.
- When a start condition is detected while the ST bit in ICCR2 is 1b (start condition requested) or the internal SDA output state does not match the SDAn line level.
- When the ST bit in ICCR2 is 1b (start condition requested), with the BBSY flag in ICCR2 set to 1.

When NACK arbitration-lost detection is enabled (ICFER.NALE = 1b):

- When the internal SDA output state does not match the SDAn line level on the rising edge of the SCL clock in the ACK period during NACK transmission in receive mode.

When slave arbitration-lost detection is enabled (ICFER.SALE = 1b):

- When the internal SDA output state does not match the SDAn line level on the rising edge of the SCL clock, except for the ACK period during data transmission in slave transmit mode.

[Clearing conditions]

- When 0b is written to the AL flag after reading AL = 1b.
- When 1b is written to the IICRST bit in ICCR1 to apply an RIIC reset or internal reset.

Table 7.7-6 Relationship between Arbitration-Lost Generation Sources and Arbitration-Lost Enable Functions

ICFER			ICSR2		
MALE	NALE	SALE	AL	Error	Arbitration-Lost Generation Source
1	×	×	1	Start condition issuance error	When internal SDA output state does not match SDAn line level when a start condition is detected, while the ST bit in ICCR2 is 1.
					When ST in ICCR2 is set to 1 while BBSY in ICCR2 is 1.
			1	Transmit data mismatch	When transmit data (including slave address) does not match the bus state in master transmit mode.
×	1	×	1	NACK transmission mismatch	When ACK is detected during transmission of NACK in master or slave receive mode.
×	×	1	1	Transmit data mismatch	When transmit data does not match the bus state in slave transmit mode.

Note: × = Don't care

START bit (Start Condition Detection Flag)

The START flag indicates whether a start or restart condition is detected.

[Setting condition]

- When a start or restart condition is detected.

[Clearing conditions]

- When 0b is written to the START flag after reading START = 1b.
- When a stop condition is detected.
- When 1b is written to the IICRST bit in ICCR1 to apply an RIIC reset or internal reset.

STOP bit (Stop Condition Detection Flag)

The STOP flag indicates whether a stop condition is detected.

[Setting condition]

- When a stop condition is detected.

[Clearing conditions]

- When 0b is written to the STOP flag after reading STOP = 1b.
- When 1b is written to the IICRST bit in ICCR1 to apply an RIIC reset or internal reset.

NACKF bit (NACK Detection Flag)

The NACKF flag indicates whether a NACK was detected.

[Setting condition]

- When acknowledge is not received (NACK received) from the receive device in transmit mode, with the NACKE bit in ICFER set to 1b (transfer suspension enabled).

[Clearing conditions]

- When 0b is written to the NACKF flag after reading NACKF = 1b.
- When 1b is written to the IICRST bit in ICCR1 to apply an RIIC reset or internal reset.

Note: When the NACKF flag is set to 1b, the RIIC suspends data transmission and reception. Writing to ICDRT in transmit mode or reading from ICDRR in receive mode with the NACKF flag set to 1b does not enable data transmit or receive operation. To restart data transmission or reception, set the NACKF flag to 0b.

RDRF bit (Receive Data Full Flag)

The RDRF flag indicates whether the IDCRR contains receive data.

[Setting conditions]

- When receive data is transferred from ICDRS to ICDRR.
  The RDRF flag is set to 1b on the rising edge of the 8th or 9th SCL clock cycle (selected in the RDRFS bit in ICMR3).
- When the received slave address matches after a start (or restart) condition is detected with the TRS bit in ICCR2 set to 0b.

[Clearing conditions]

- When 0b is written to the RDRF flag after reading RDRF = 1b.
- When data is read from ICDRR.
- When 1b is written to the IICRST bit in ICCR1 to initiate an RIIC reset or internal reset.

TEND bit (Transmit End Flag)

The TEND flag indicates whether data transmission is still being transmitted or is complete.

[Setting condition]

- On the rising edge of the 9th SCL clock cycle while the TDRE flag is 1b.

[Clearing conditions]

- When 0b is written to the TEND flag after reading TEND = 1b.
- When data is written to ICDRT.
- When a stop condition is detected.
- When 1b is written to the IICRST bit in ICCR1 to initiate an RIIC reset or internal reset.

TDRE bit (Transmit Data Empty Flag)

The TDRE flag indicates whether the ICDRT contains transmit data.

[Setting conditions]

- When data is transferred from ICDRT to ICDRS and ICDRT becomes empty.
- When the TRS bit in ICCR2 is set to 1b.
- When the received slave address matches while the TRS bit is 1b.

[Clearing conditions]

- When data is written to ICDRT.
- When the TRS bit in ICCR2 is set to 0b.
- When 1b is written to the IICRST bit in ICCR1 to apply an RIIC reset or internal reset.

Note: The NACKF flag becoming 1 while the ICFER.NACKE bit is 1 suspends data transmission and reception by the RIIC. Even if the next data for transmission has already been written to the ICDRT register (the TDRE flag is 0b), the data in the ICDRT register is retained but not transferred to the ICDRS register. At this point, the TDRE flag does not become 1.

7.7.2.2.11 Slave Address Register L y (RIICm_SARLy) (y = 0 to 2)

Access Size : 8 bits

Address : <RIICm_base> + 000Ah + 0002h × y

Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	SVA[6:0]							SVA0
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	SVA[6:0]	0h	RW	7-bit Address/10-bit Address Lower Bits Slave address setting
0	SVA0	0h	RW	10-bit Address LSB Slave address setting

SVA0 bit (10-bit Address LSB)

When the 10-bit address format is selected (SARUy.FS = 1b), the SVA0 bit functions as the LSB of a 10-bit address and is combined with the SVA[6:0] bits to form the lower 8 bits of a 10-bit address.

This bit is valid when the SARyE bit in ICSER is set to 1b (SARLy and SARUy enabled) and the SARUy.FS bit is 1b. When the SARUy.FS or SARyE bit is 0b, the setting in this bit is ignored.

SVA[6:0] bits (7-bit Address/10-bit Address Lower Bits)

When the 7-bit address format is selected (SARUy.FS = 0b), the SVA[6:0] bits function as a 7-bit address. When the 10-bit address format is selected (SARUy.FS = 1b), these bits combined with the SVA0 bit to form the lower 8 bits of a 10-bit address. When the SARyE bit in ICSER is 0b, the setting in these bits is ignored.

7.7.2.2.12 Slave Address Register U y (RIICm_SARUy) (y = 0 to 2)

Access Size : 8 bits

Address : <RIICm_base> + 000Bh + 0002h × y

Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	SVA[1:0]		FS
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2, 1	SVA[1:0]	0h	RW	10-bit Address Upper Bits Slave address setting
0	FS	0h	RW	7-bit/10-bit Address Format Select 0b: Select 7-bit address format 1b: Select 10-bit address format

FS bit (7-bit/10-bit Address Format Select)

The FS bit selects a 7-bit or 10-bit address format for slave address y (in SARLy and SARUy).

When the SARyE bit in ICSER is set to 1b (SARLy and SARUy enabled) and the SARUy.FS bit is 0b, the 7-bit address format is selected for slave address y, the SVA[6:0] setting in SARLy is valid, and the SVA[1:0] and SVA0 settings in SARLy are ignored.

When the SARyE bit in ICSER is set to 1b (SARLy and SARUy enabled) and the SARUy.FS bit is 1b, the 10-bit address format is selected for slave address y and the SVA[1:0] and SARLy settings are valid.

When the SARyE bit in ICSER is 0b (SARLy and SARUy disabled), the SARUy.FS setting is invalid.

SVA[1:0] bits (10-bit Address Upper Bits)

When the 10-bit address format is selected (FS = 1b), the SVA[1:0] bits function as the upper 2 bits of a 10-bit address.

These bits are valid when the SARyE bit in ICSER is set to 1b (SARLy and SARUy enabled) and the SARUy.FS bit is 1b. When the SARUy.FS or SARyE bit is 0b, the setting in these bits is ignored.

7.7.2.2.13 I²C Bus Bit Rate Low-Level Register (RIICm_ICBRL)

Access Size : 8 bits
Address : <RIICm_base> + 0010h
Initial Value : FFh

Bit	7	6	5	4	3	2	1	0
	-	-	-	BRL[4:0]				
Initial Value	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	-	All 1	RW	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
4 to 0	BRL[4:0]	1Fh	RW	Bit Rate Low-Level Period Low-level period setting of SCL clock

ICBRL is a 5-bit register that sets the low-level period of the SCL clock. ICBRL also works to generate the data setup time for the automatic SCL low-hold operation (see **7.7.8 Automatic Low-Hold Function for SCL**). When the RIIC is used only in slave mode, this register must be set to a value longer than the data setup time.*1

ICBRL counts the low-level period with the internal reference clock source (IICϕ) specified in the CKS[2:0] bits in ICMR1.

If the digital noise filter is enabled (the NFE bit in ICFER is 1), set the ICBRL register to a value at least one greater than the number of stages in the noise filter. For this number, see the description of the ICMR3.NF[1:0] bits.

Note 1. Data setup time (tSU: DAT)
250 ns for up to 100 kbps: Standard-mode (Sm)
100 ns for up to 400 kbps: Fast-mode (Fm)
50 ns for up to 1 Mbps: Fast-mode plus (Fm+)

7.7.2.2.14 I²C Bus Bit Rate High-Level Register (RIICm_ICBRH)

Access Size : 8 bits
Address : <RIICm_base> + 0011h
Initial Value : FFh

Bit	7	6	5	4	3	2	1	0
	-	-	-	BRH[4:0]				
Initial Value	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	-	All 1	RW	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
4 to 0	BRH[4:0]	1Fh	RW	Bit Rate High-Level Period High-level period setting of SCL clock

ICBRH is a 5-bit register that sets the high-level period of the SCL clock. ICBRH is valid in master mode. If the RIIC is used only in slave mode, no setting is required in this register.

ICBRH counts the high-level period with the internal reference clock source (IICϕ) specified in the CKS[2:0] bits in ICMR1.

If the digital noise filter is enabled (the NFE bit in ICFER is 1b), set the ICBRH register to a value at least one greater than the number of stages in the noise filter. For this number, see the description of the ICMR3.NF[1:0] bits.

The RIIC transfer rate and the SCL clock duty are calculated using the following expressions (1) to (5):

(1) ICFER.SCLE = 0b
Transfer rate = 1/{[(BRH + 1) + (BRL + 1)]/IICϕ*[1] + tr*[2] + tf*[2]}
Duty cycle = {tr + [(BRH + 1)/IICϕ]}/{tr + tf + [(BRH + 1) + (BRL + 1)]/IICϕ}

(2) ICFER.SCLE = 1b and ICFER.NFE = 0b and CKS[2:0] = 000b (IICϕ = PCLKL*[4])
Transfer rate = 1/{[(BRH + 3) + (BRL+ 3)]/IICϕ + tr + tf}
Duty cycle = {tr + [(BRH + 3)/IICϕ]}/{tr + tf + [(BRH + 3) + (BRL + 3)]/IICϕ}

(3) ICFER.SCLE = 1b and ICFER.NFE = 1b and CKS[2:0] = 000b (IICϕ = PCLKL)
Transfer rate = 1/{[(BRH + 3 + nf*[3]) + (BRL + 3 + nf)]/IICϕ + tr + tf}
Duty cycle = {tr + [(BRH + 3 + nf)/IICϕ]}/{tr + tf + [(BRH + 3 + nf) + (BRL + 3 + nf)]/IICϕ}

(4) ICFER.SCLE = 1b and ICFER.NFE = 0b and CKS[2:0] ≠ 000b
Transfer rate = 1/{[(BRH + 2) + (BRL + 2)]/IICϕ + tr + tf}
Duty cycle = {tr + [(BRH + 2)/IICϕ]}/{tr + tf + [(BRH + 2) + (BRL + 2)]/IICϕ}

(5) ICFER.SCLE = 1b and ICFER.NFE = 1b and CKS[2:0] ≠ 000b
Transfer rate = 1/{[(BRH + 2 + nf) + (BRL + 2 + nf)]/IICϕ + tr + tf}
Duty cycle = {tr + [(BRH + 2 + nf)/IICϕ]}/{tr + tf + [(BRH + 2 + nf) + (BRL + 2 + nf)]/IICϕ}

Note 1. IICϕ = PCLKL × Division ratio.
Note 2. The SCLn line rise time (tr) and SCLn line fall time (tf) depend on the total bus line capacitance (Cb) and the pull-up resistor (Rp).
For details, see the I²C bus standard from NXP Semiconductors.
Note 3. nf = Number of digital noise filter stages selected in the ICMR3.NF bit.
Note 4. PCLKL: RIIC_n_ckm [n = 0 to 8]

Table 7.7-7 Example of ICBRH/ICBRL Settings for Transfer Rate when SCLE = 0b

Transfer Rate (kbps)	CKS [2:0]	BRH [4:0]	BRL [4:0]	PCLKL (MHz)	NF[1:0]	Computation Expression
97	101b	13 (0Dh)	13 (0Dh)	100	—	(1) tr = 1300 ns, tf = 300 ns
397	010b	23 (17h)	23 (17h)	100	—	(1) tr = 300 ns, tf = 300 ns

Table 7.7-8 Example of ICBRH/ICBRL Settings for Transfer Rate when SCLE = 1b and NFE = 0b

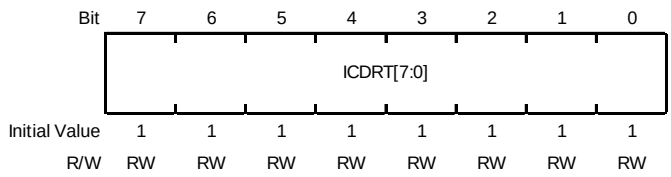
Transfer Rate (kbps)	CKS [2:0]	BRH [4:0]	BRL [4:0]	PCLKL (MHz)	NF [1:0]	Computation Expression
97	100b	26 (1Ah)	26 (1Ah)	100	—	(4) tr = 1300 ns, tf = 300 ns
397	010b	22 (16h)	22 (16h)	100	—	(4) tr = 300 ns, tf = 300 ns

Table 7.7-9 Example of ICBRH/ICBRL Settings for Transfer Rate when SCLE = 1b and NFE = 1b

Transfer Rate (kbps)	CKS [2:0]	BRH [4:0]	BRL [4:0]	PCLKL (MHz)	NF [1:0]	Computation Expression
97	100b	25 (19h)	25 (19h)	100	01b	(5) tr = 1300 ns, tf = 300 ns
397	010b	21 (15h)	21 (15h)	100	01b	(5) tr = 300 ns, tf = 300 ns

7.7.2.2.15 I²C Bus Transmit Data Register (RIICm_ICDRT)

Access Size : 8 bits
Address : <RIICm_base> + 0012h
Initial Value : FFh



Bit	Bit Name	Initial Value	R/W	Description
7 to 0	ICDRT[7:0]	FFh	RW	Transmit Data

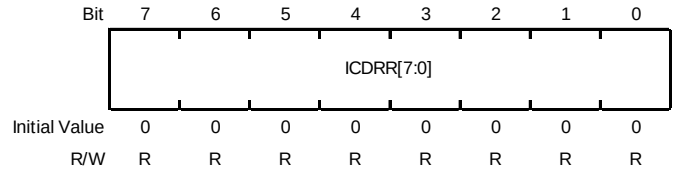
When ICDRT detects a space in the I²C Bus Shift Register (ICDRS), it transfers the transmit data that is written to ICDRT to ICDRS and starts transmitting data in transmit mode.

The double-buffer structure of ICDRT and ICDRS allows continuous transmit operation if the next transmit data is written to ICDRT while the ICDRS data is being transmitted.

ICDRT can always be read from and written to. Write transmit data to ICDRT when a transmit data empty interrupt (IICn_TXI) request is generated.

7.7.2.2.16 I²C Bus Receive Data Register (RIICm_ICDRR)

Access Size : 8 bits
Address : <RIICm_base> + 0013h
Initial Value : 00h



Bit	Bit Name	Initial Value	R/W	Description
7 to 0	ICDRR[7:0]	0h	R	Receive Data

When 1 byte of data is received, the received data is transferred from the I²C Bus Shift Register (ICDRS) to ICDRR to enable the next data to be received.

The double-buffer structure of ICDRS and ICDRR allows continuous receive operation if the received data is read from ICDRR while ICDRS is receiving data. ICDRR cannot be written to. Read data from ICDRR when a receive data full interrupt (IICn_RXI) request is generated.

If ICDRR receives the next receive data before the current data is read from ICDRR while the RDRF flag in ICSR2 is 1b, the RIIC automatically holds the SCL clock low for 1 cycle before the RDRF flag is set to 1b again.

7.7.2.2.17 I²C Bus Shift Register (RIICm_ICDRS)

Access Size : 8 bits
Address : <RIICm_base> + n/a
Initial Value : xxh

Bit	7	6	5	4	3	2	1	0
	[7:0]							
Initial Value	x	x	x	x	x	x	x	x
R/W	-	-	-	-	-	-	-	-

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	[7:0]	x	-	The read value is undefined

ICDRS is an 8-bit shift register to transmit and receive data.

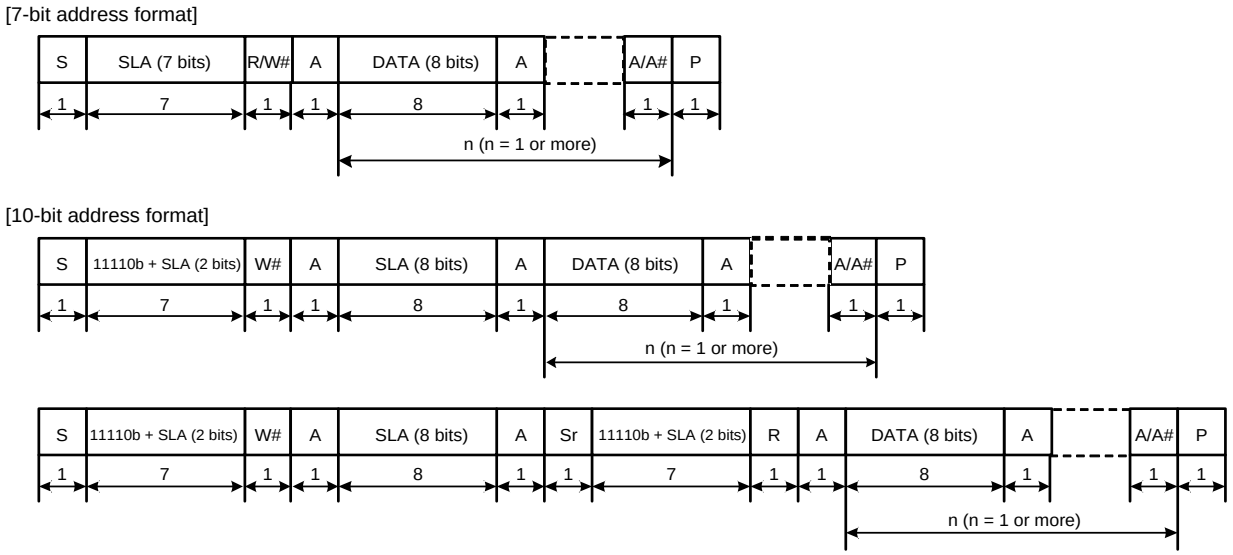
During transmission, transmit data is transferred from ICDRT to ICDRS and is sent from the SDAn pin. During reception, data is transferred from ICDRS to ICDRR after 1 byte of data is received. ICDRS cannot be accessed directly.

7.7.3 Operation

7.7.3.1 Communication Data Format

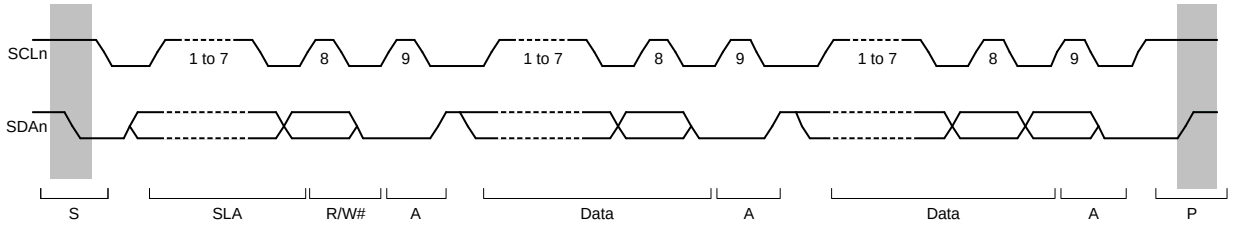
The I²C bus format consists of 8-bit data and 1-bit acknowledge. The frame following a start or restart condition is an address frame that specifies a slave device with which the master device communicates. The specified slave is valid until a new slave is specified or a stop condition is issued.

Figure 7.7-3 shows the I²C bus format, and **Figure 7.7-4** shows the I²C bus timing.



Note: n: Number of transfer frames

Figure 7.7-3 I²C Bus Format



S: Start condition. The master device drives the SDAn line low from high while the SCLn line is high.

SLA: Slave address, by which the master device selects a slave device.

R/W#: Indicates the direction of data transfer: from the slave device to the master device when R/W is 1, or from the master device to the slave device when R/W is 0.

A: Acknowledge. The receive device drives the SDAn line low. In master transmit mode, the slave device returns acknowledge. In master receive mode, the master device returns acknowledge.

A#: Not Acknowledge. The receive device drives the SDAn line high.

Sr: Restart condition. The master device drives the SDAn line low from the high level after the setup time has elapsed with the SCLn line high.

DATA: Transmitted or received data

P: Stop condition. The master device drives the SDAn line high from low while the SCLn line is high.

Figure 7.7-4 I^2C Bus Timing when the SLA Setting = 7 Bits

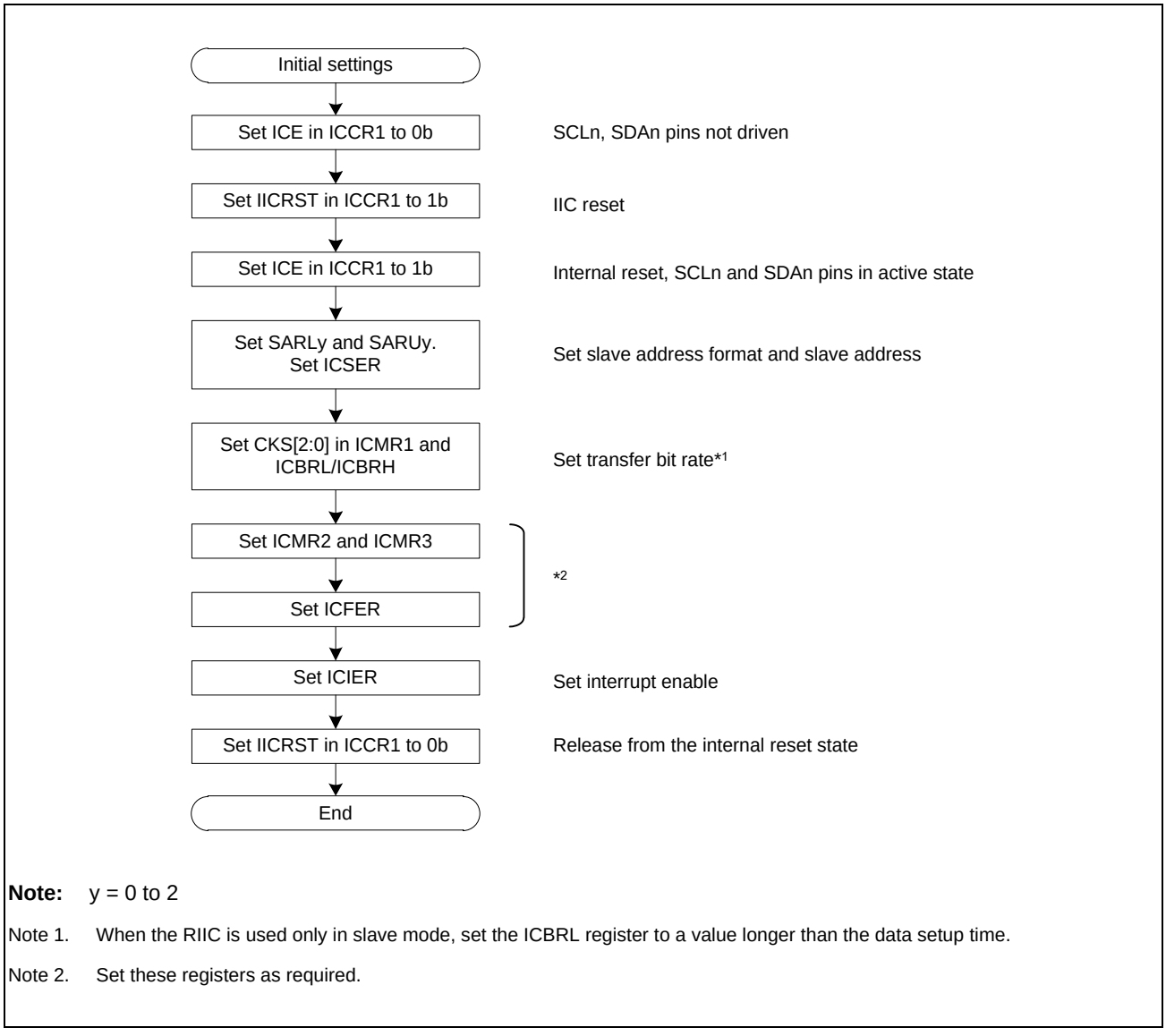
7.7.3.2 Initial Settings

Before starting data transmission and reception, initialize the RIIC using the procedure shown in **Figure 7.7-5**.

1. Set the ICCR1.ICE bit set to 0b to set the SCLn and SDAn pins to the inactive state.
2. Set the ICCR1.IICRST bit to 1b to initiate RIIC reset.
3. Set the ICCR1.ICE bit to 1b to initiate internal reset, which initializes the flags and the internal state of the ICSR1 register.
4. Set the SARLy, SARUy, ICSER, ICMR1, ICBRH, and ICBRL registers (y = 0 to 2), and set the other registers as required. For the initial RIIC settings, see **Figure 7.7-5**.
5. When the required register settings are complete, set the ICCR1.IICRST bit to 0b, to release the RIIC reset.

NOTE

This procedure is not necessary if the RIIC initialization is already complete.



Note: y = 0 to 2

Note 1. When the RIIC is used only in slave mode, set the ICBRL register to a value longer than the data setup time.

Note 2. Set these registers as required.

Figure 7.7-5 Example RIIC Initialization Flow

7.7.3.3 Master Transmit Operation

In master transmit operation, the RIIC outputs the SCL clock and transmit data signals as the master device, and the slave device returns the acknowledgments. **Figure 7.7-6** shows an example of master transmission, and **Figure 7.7-7** to **Figure 7.7-9** show the operation timing in master transmission.

To set up and perform master transmission:

1. To initialize the RIIC, follow the procedure in **7.7.3.2 Initial Settings**.
2. Read the BBSY flag in ICCR2 to check that the bus is free, and then set the ST bit in ICCR2 to 1b (start condition request). On receiving the request, the RIIC issues a start condition. At the same time, the BBSY and START flags in ICSR2 automatically set to 1b and the ST bit is automatically set to 0b. If the start condition is detected and the internal levels for the SDA output state and the levels on the SDAn line match while the ST bit is 1b, the RIIC recognizes that the start condition requested by the ST bit is successfully complete, and the MST and TRS bits in ICCR2 automatically set to 1b, placing the RIIC in master transmit mode. The TDRE flag in ICSR2 also automatically sets to 1b in response to the setting of the TRS bit to 1b.
3. Check that the TDRE flag in ICSR2 is 1b, and then write the value for transmission (the slave address and the R/W# bit) to ICDRT. When the transmit data is written to ICDRT, the TDRE flag is automatically set to 0b, the data is transferred from ICDRT to ICDRS, and the TDRE flag again sets to 1b. After the byte containing the slave address and R/W# bit is transmitted, the value of the TRS bit automatically updates to select master transmit or master receive mode based on the value of the transmitted R/W# bit. If the value of the R/W# bit is 0b, the RIIC continues in master transmit mode.
   If the ICSR2.NACKF flag is 1b, indicating that no slave device recognized the address or that there was an error in communications, write 1b to the ICCR2.SP bit to issue a stop condition.
   To transmit data with an address in the 10-bit format, start by writing 11110b, the 2 upper bits of the slave address, and W to ICDRT as the first address transmission. For the second address transmission, write the 8 lower bits of the slave address to ICDRT.
4. After confirming that the TDRE flag in ICSR2 is 1b, write the transmit data to the ICDRT register. The RIIC automatically holds the SCLn line low until the transmit data is ready or a stop condition is issued.
5. After all bytes of transmit data are written to the ICDRT register, wait until the value in the TEND flag in ICSR2 returns to 1b, and then set the SP bit in ICCR2 to 1b (stop condition requested). On receiving a stop condition request, the RIIC issues the stop condition.
6. On detecting the stop condition, the RIIC automatically sets the MST and TRS bits in ICCR2 to 00b and enters slave receive mode. In addition, the RIIC automatically sets the TDRE and TEND flags to 0b, and sets the STOP flag in ICSR2 to 1b.
7. After checking that the ICSR2.STOP flag is 1b, set the ICSR2.NACKF and STOP flags to 0b for the next transfer operation.

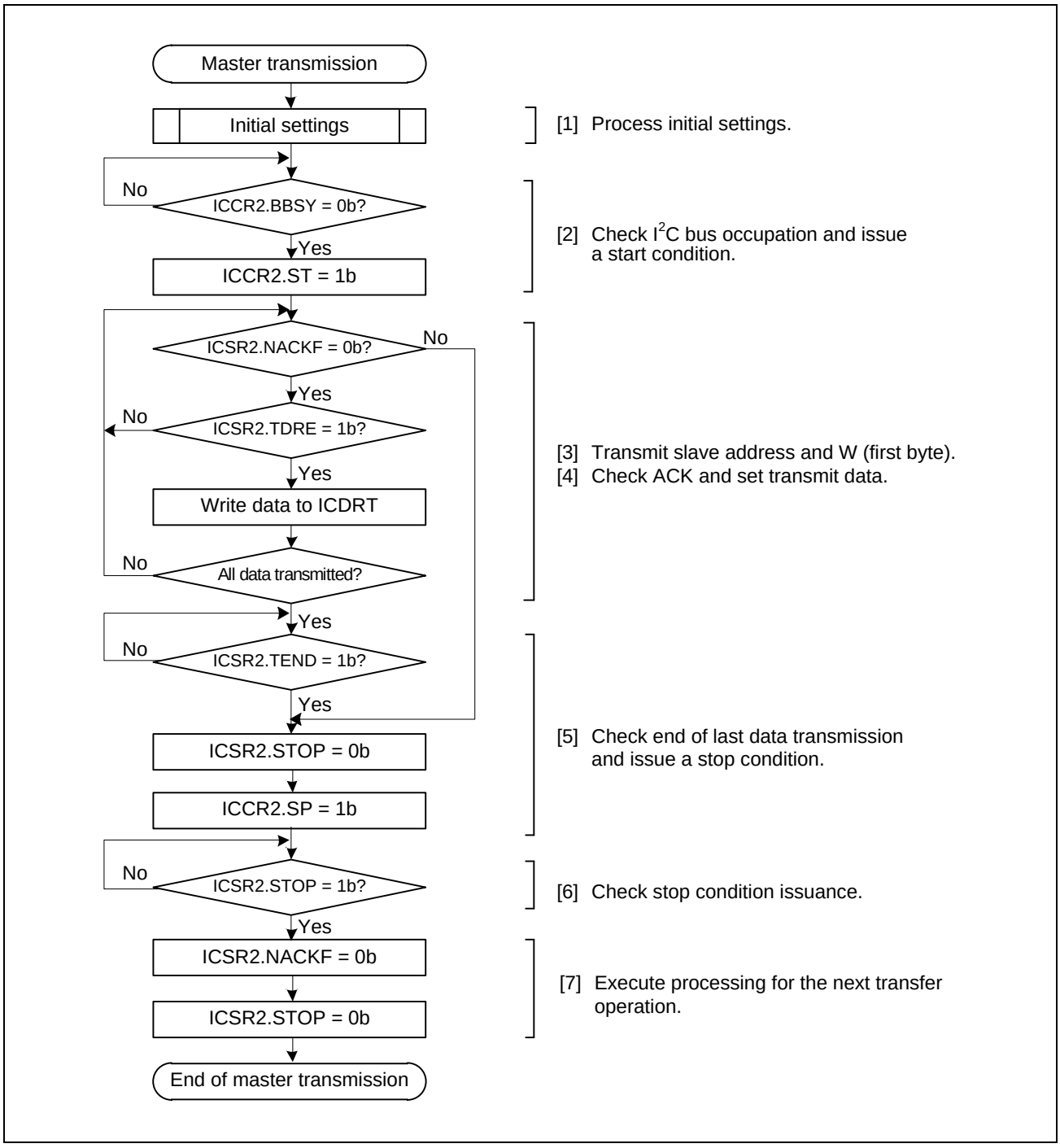


Figure 7.7-6 Example Master Transmission Flow

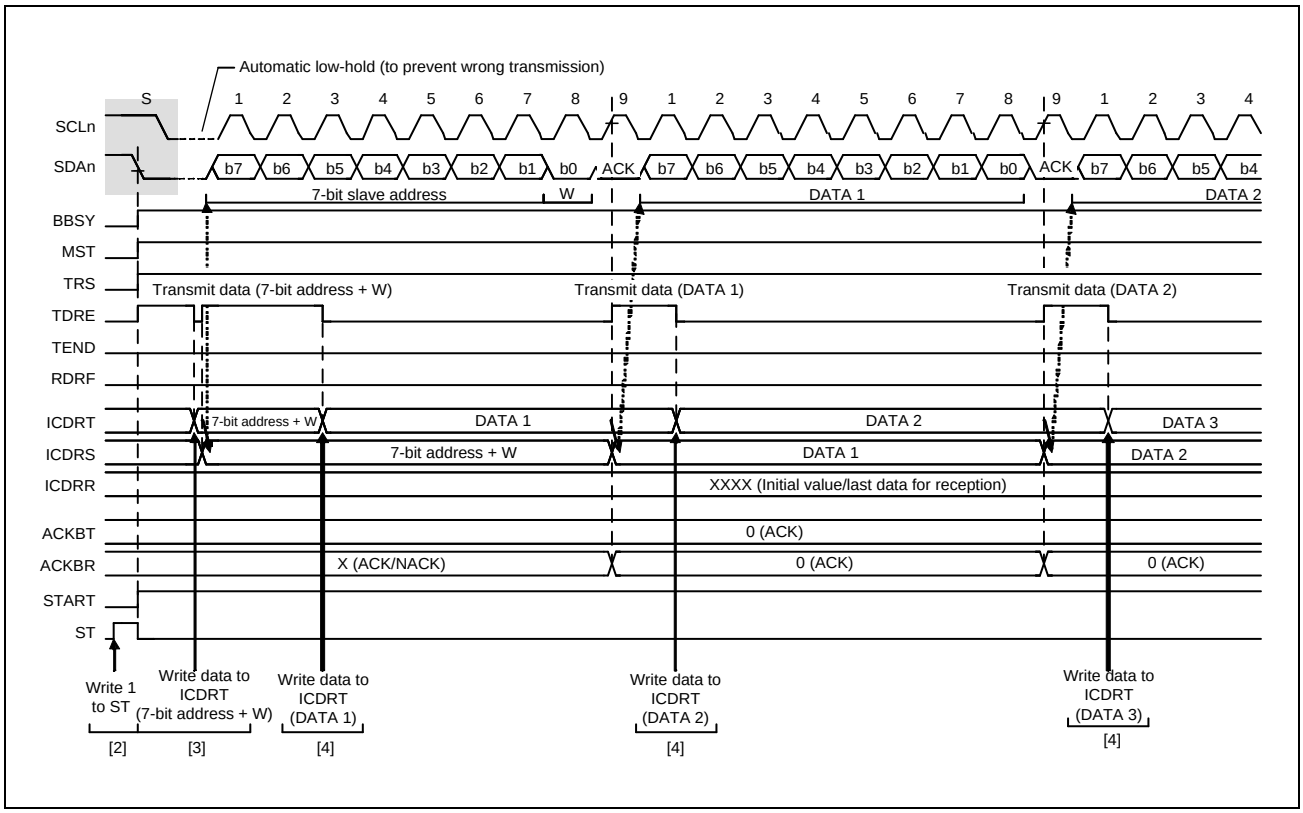


Figure 7.7-7 Master Transmit Operation Timing (1) with 7-Bit Address Format

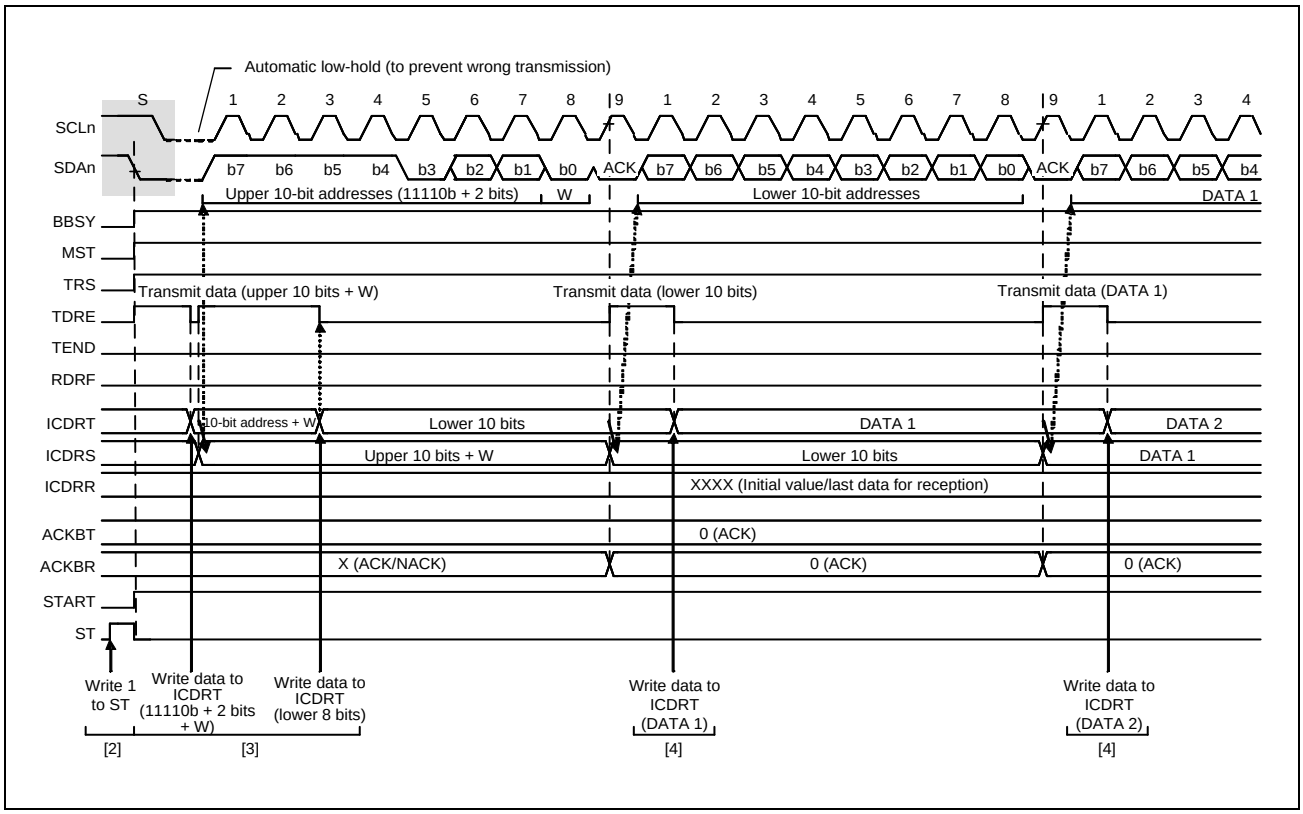


Figure 7.7-8 Master Transmit Operation Timing (2) with 10-Bit Address Format

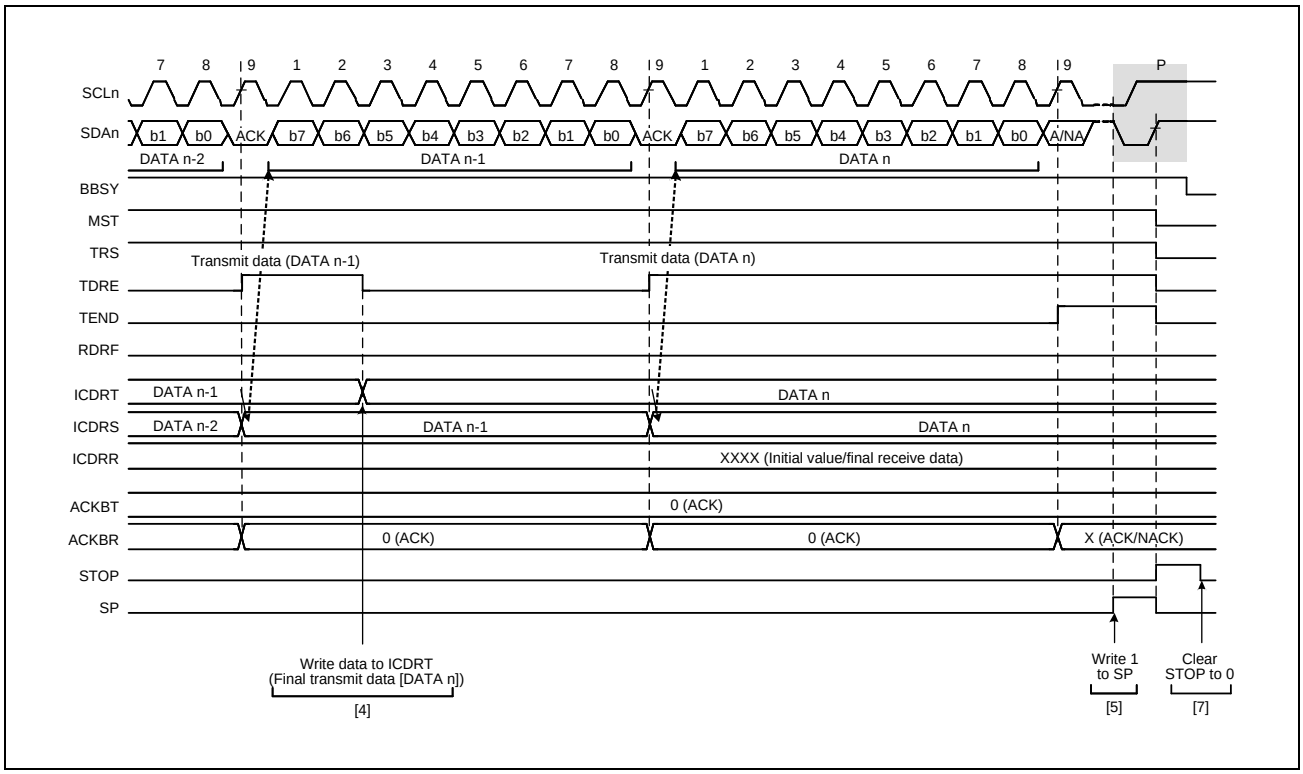


Figure 7.7-9 Master Transmit Operation Timing (3)

7.7.3.4 Master Receive Operation

In master receive operation, the RIIC as a master device outputs the SCL clock, receives data from the slave device, and returns acknowledgments. Because the RIIC must start by sending a slave address to the associated slave device, the slave address part of the procedure is performed in master transmit mode, but the subsequent steps are in master receive mode.

Figure 7.7-10 and **Table 7.7-11** show examples of master reception (7-bit address format), and **Figure 7.7-12** to **Figure 7.7-14** show the operation timing in master reception.

To set up and perform master reception:

1. To initialize the RIIC, follow the procedure in **7.7.3.2 Initial Settings**.
2. Read the BBSY flag in ICCR2 to check that the bus is free, and then set the ST bit in ICCR2 to 1b to request the issue of a start condition. On receiving the request, the RIIC issues a start condition. When the RIIC detects the start condition, the BBSY and START flags in ICSR2 automatically set to 1b, and the ST bit automatically sets to 0b. If the start condition is detected and the levels for the SDA output and the levels on the SDAn line match while the ST bit is 1b, the RIIC recognizes that issuance of the start condition as requested by the ST bit is successfully complete, and the MST and TRS bits in ICCR2 automatically set to 1b, placing the RIIC in master transmit mode. The TDRE flag in ICSR2 also is automatically set to 1b in response to the setting of the TRS bit to 1b.
3. Check that the TDRE flag in ICSR2 is 1b, and then write the value for transmission (the first byte indicates the slave address and value of the R/W# bit) to ICDRT. When the transmit data is written to ICDRT, the TDRE flag is automatically set to 0b, the data is transferred from ICDRT to ICDRS, and the TDRE flag again is set to 1b. When the byte containing the slave address and R/W# bit is transmitted, the value of the ICCR2.TRS bit automatically updates to select transmit or receive mode based on the value of the transmitted R/W# bit. If the value of the R/W# bit is 1b, the TRS bit is set to 0b on the rising edge of the 9th cycle of the SCL clock, placing the RIIC in master receive mode. At this time, the TDRE flag is set to 0b and the ICSR2.RDRF flag is automatically set to 1b.
   If the ICSR2.NACKF flag is 1b, indicating that no slave device recognized the address or that there is an error in communications, write 1b to the ICCR2.SP bit to issue a stop condition.
   For master reception from a device with a 10-bit address, start by using master transmission to issue the 10-bit address, and then issue a restart condition. After that, transmit 11110b, the two upper bits of the slave address, and the R bit to place the RIIC in master receive mode.
4. Dummy read ICDRR after confirming that the RDRF flag in ICSR2 is 1b. Doing so causes the RIIC to start output of the SCL clock and start data reception.
5. After 1 byte of data is received, the RDRF flag in ICSR2 is set to 1b on the rising edge of the 8th or 9th cycle of the SCL clock, as selected in the RDRFS bit in ICMR3. Reading ICDRR produces the received data, and automatically sets the RDRF flag to 0b. The value of the acknowledgment field received during the 9th cycle of the SCL clock is returned as the value set in the ICMR3.ACKBT bit. If the next byte to be received is the second-to-last byte, set the ICMR3.WAIT bit to 1b for wait insertion before reading ICDRR, containing the second-to-last byte. In addition to enabling NACK output, even when interrupts or other operations result in delays in setting the ICMR3.ACKBT bit to 1b (NACK) in step (6), this fixes the SCLn line to the low level on the rising edge of the 9th clock cycle in reception of the last byte, which enables the issuing of a stop condition.
6. When the ICMR3.RDRFS bit is 0b, and the slave device must be notified that it is to end transfer for data reception after transfer of the next and final byte, set the ICMR3.ACKBT bit to 1b (NACK).
7. After reading the second-to-last byte from the ICDRR register, if the value of the ICSR2.RDRF flag is 1b, write 1b to the SP bit in ICCR2 (stop condition requested), and then read the last byte from ICDRR. When ICDRR is read, the RIIC is released from the wait state and issues the stop condition after low-level output in the 9th clock cycle is complete or the SCLn line is released from the low-hold state.

8. On detecting the stop condition, the RIIC automatically sets the MST and TRS bits in ICCR2 to 00b and enters slave receive mode. Additionally, detection of the stop condition sets the ICSR2.STOP flag to 1b.
9. Check that the ICSR2.STOP flag is 1b, and then set the ICSR2.NACKF and ICSR2.STOP flags to 0b for the next transfer operation.

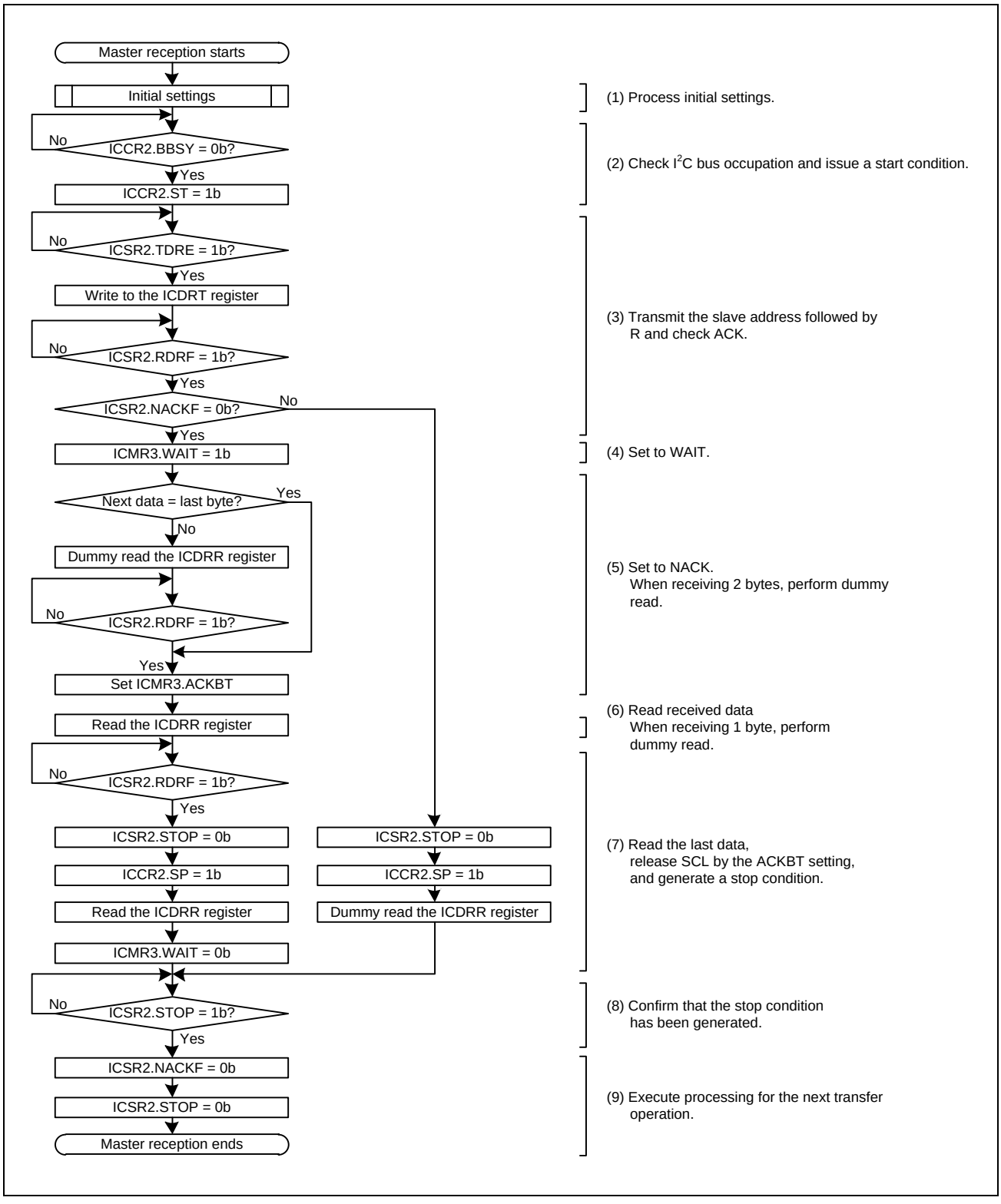


Figure 7.7-10 Example Master Reception Flow with 7-Bit Address Format and 1 or 2 Bytes

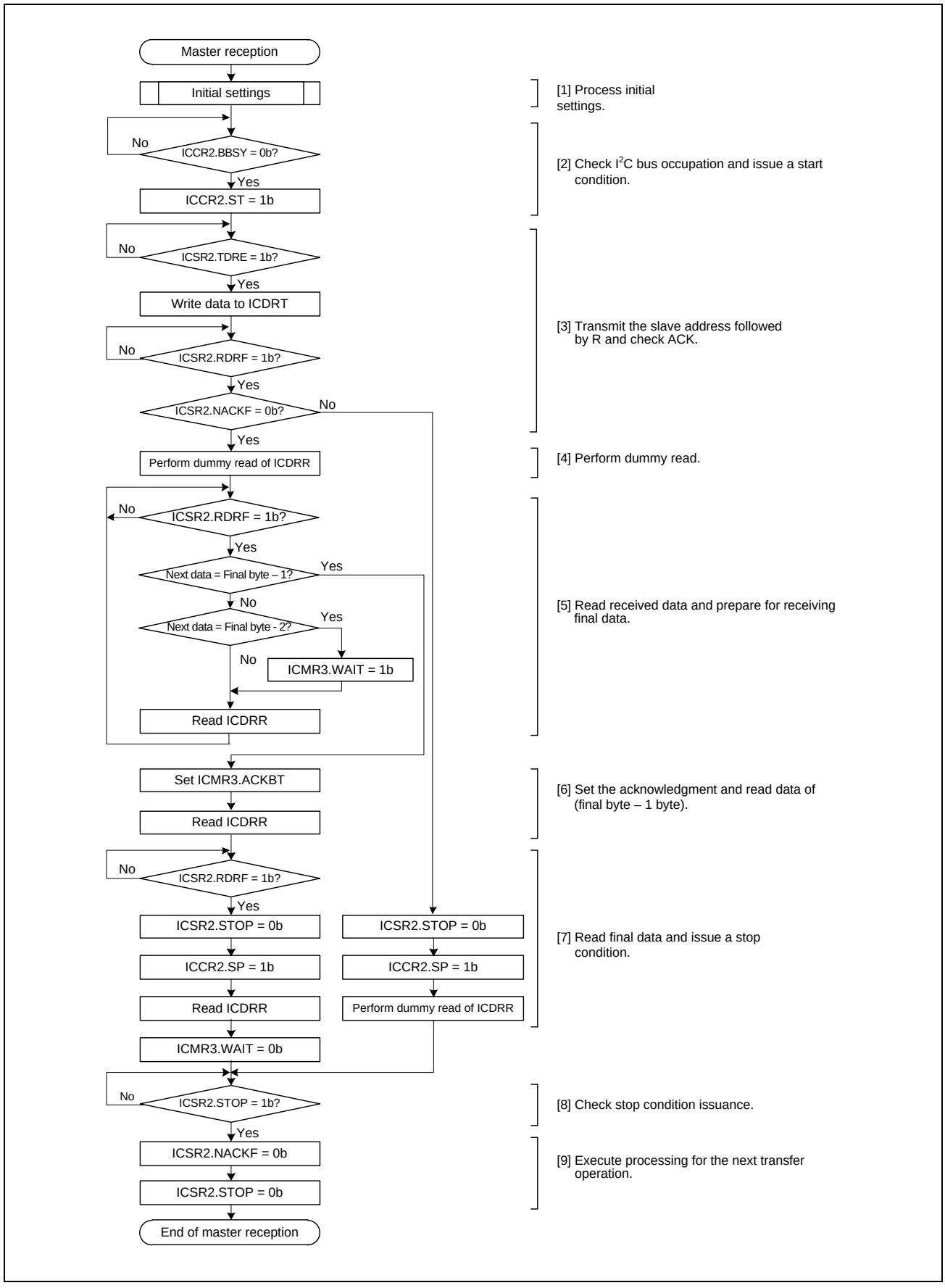


Figure 7.7-11 Example Master Reception Flow with 7-Bit Address Format and 3 or More Bytes

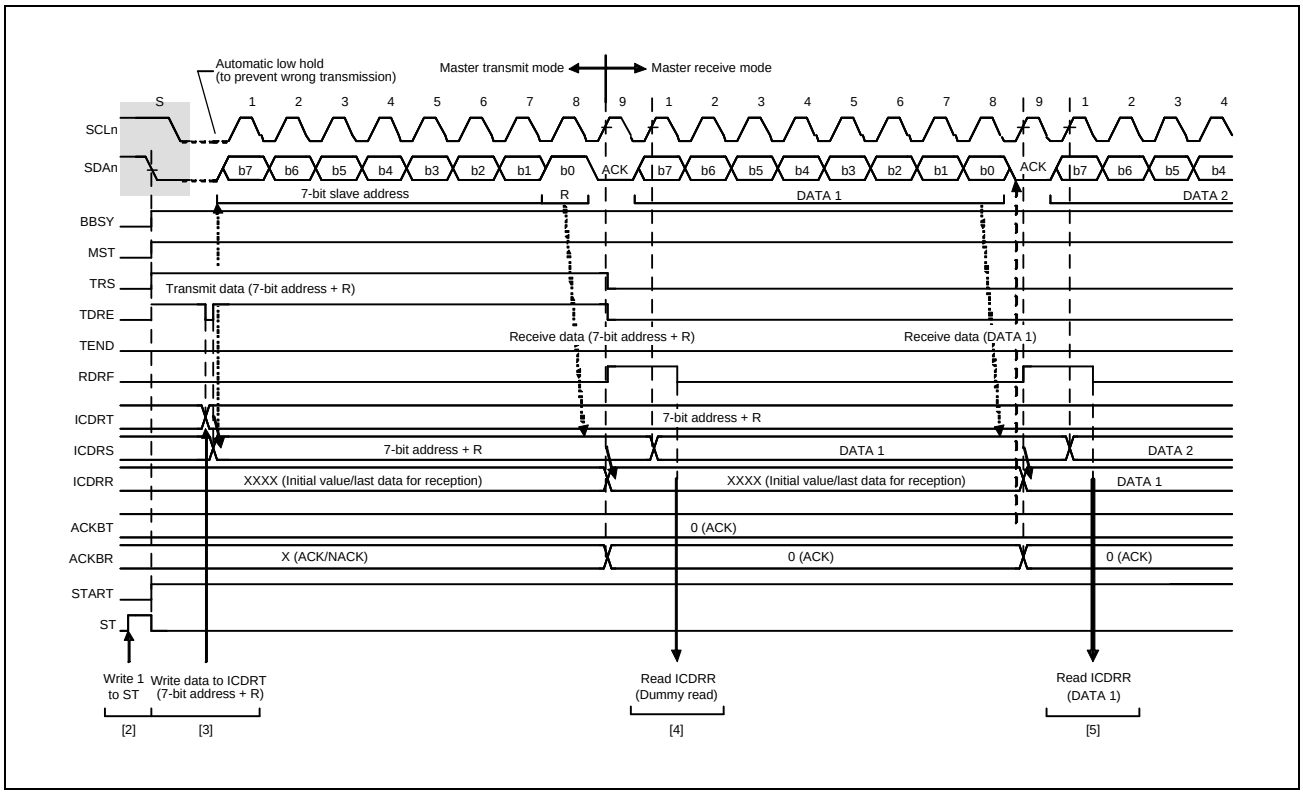


Figure 7.7-12 Master Receive Operation Timing (1) with 7-Bit Address Format, when RDRFS = 0b

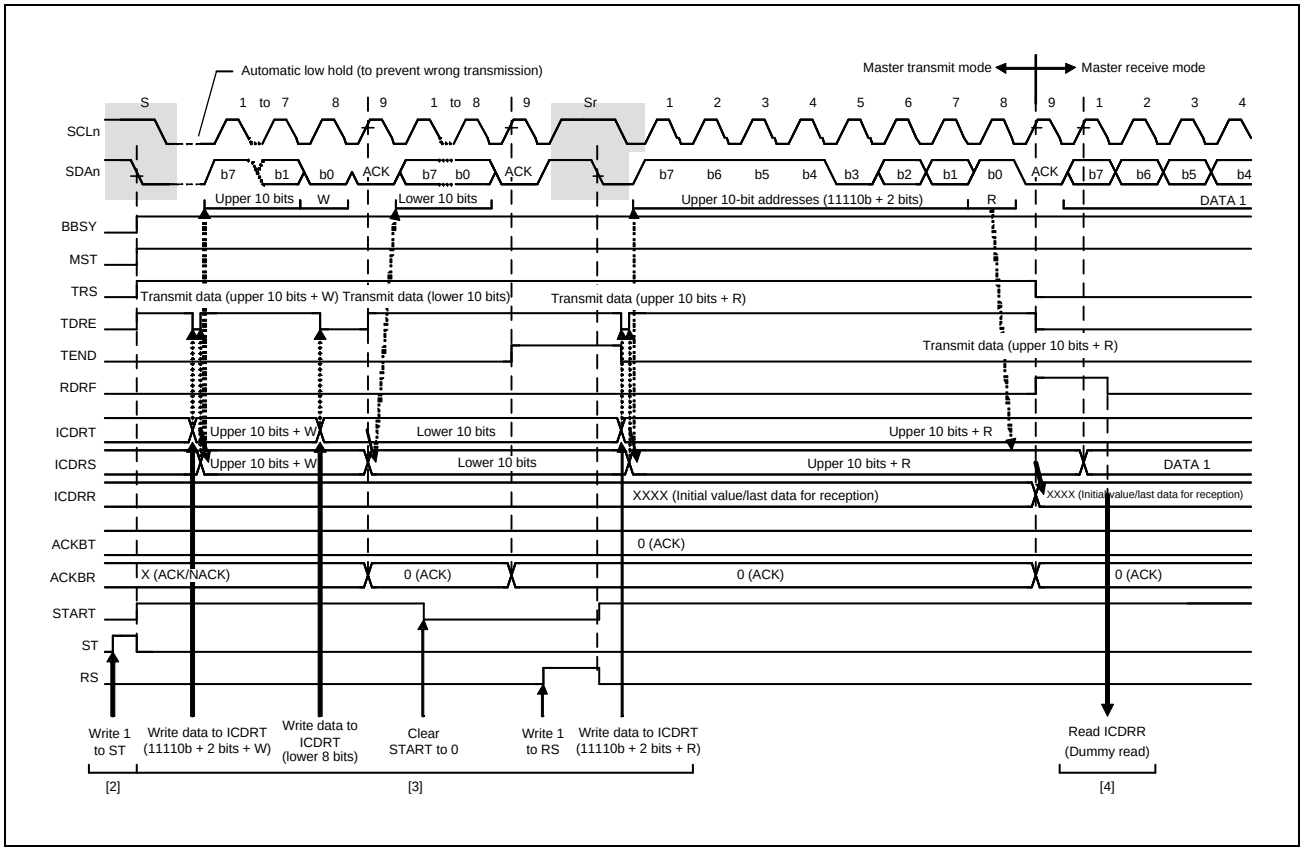


Figure 7.7-13 Master Receive Operation Timing (2) with 10-Bit Address Format, when RDRFS = 0b

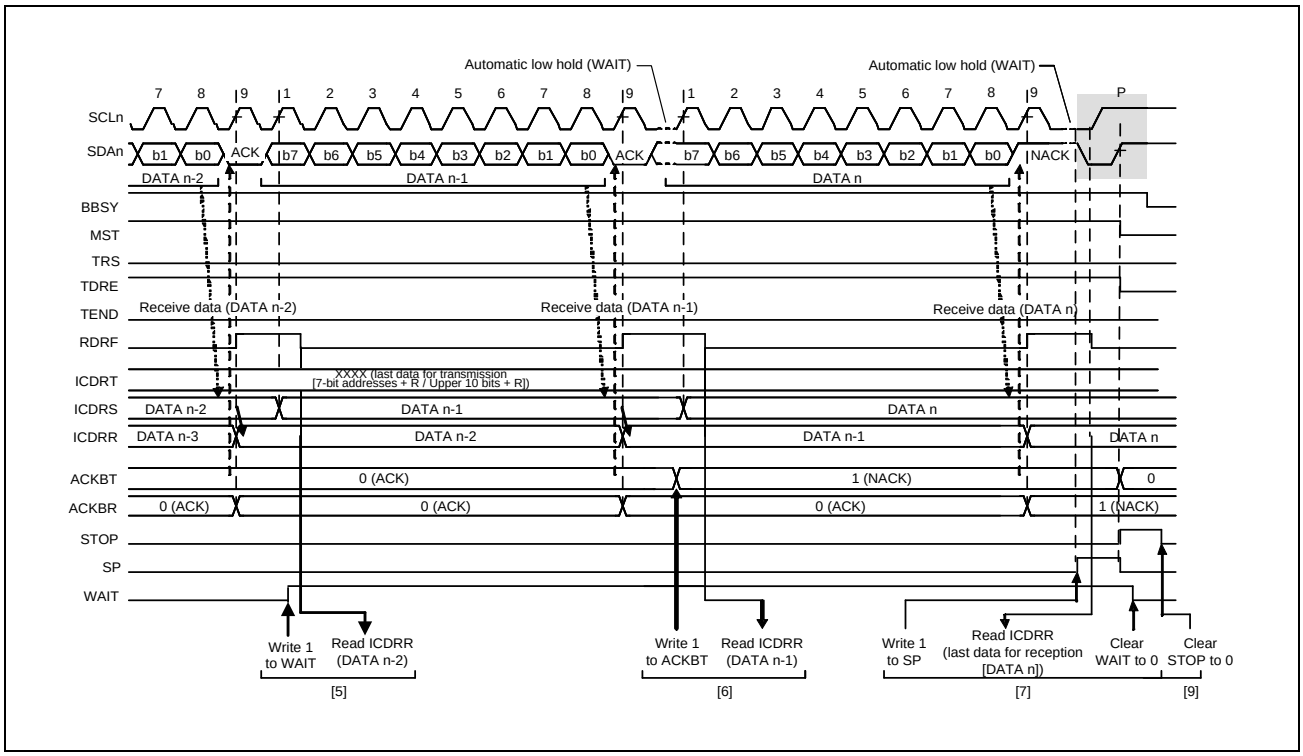


Figure 7.7-14 Master Receive Operation Timing (3) when RDRFS = 0b

7.7.3.5 Slave Transmit Operation

In slave transmit operation, the master device outputs the SCL clock, the RIIC transmits data as a slave device, and the master device returns the acknowledgments.

Figure 7.7-15 shows an example of slave transmission, and **Figure 7.7-16** and **Figure 7.7-17** show the operation timing in slave transmission.

To set up and perform slave transmission:

1. To initialize the RIIC, follow the procedure in **7.7.3.2 Initial Settings**.
   After the initialization, the RIIC stays in the standby state until it receives a slave address that matches.
2. After receiving a matching slave address, the RIIC sets one of the associated bits ICSR1.HOA, GCA, and AASy (y = 0 to 2) flags to 1b on the rising edge of the 9th cycle of the SCL clock and outputs the value set in the ICMR3.ACKBT bit to the acknowledge bit on the 9th cycle of the SCL clock. If the value of the received R/W# bit is 1b, the RIIC automatically places itself in slave transmit mode by setting both the ICCR2.TRS bit and the ICSR2.TDRE flag to 1b.
3. Check that the ICSR2.TEND flag is 1b, and write the transmit data to the ICDRT register. If the RIIC receives no acknowledge from the master device (receives a NACK signal) while the ICFER.NACKE bit is 1,b the RIIC suspends transfer of the next data.
4. Wait until the ICSR2.TEND flag is set to 1b while the ICSR2.TDRE flag is 1b, after the ICSR2.NACKF flag is set to 1b or the last byte for transmission is written to the ICDRT register. When the ICSR2.NACKF flag or the TEND flag is 1b, the RIIC drives the SCLn line low on the 9th falling edge of the SCL clock.
5. When the ICSR2.NACKF flag or the ICSR2.TEND flag is 1b, dummy read ICDRR to complete the processing. This releases the SCLn line.
6. On detecting the stop condition, the RIIC automatically sets the ICSR1.HOA, GCA, and AASy (y = 0 to 2) flags, the ICSR2.TDRE and TEND flags, and the ICCR2.TRS bit to 0b, and enters slave receive mode.
7. Check that the ICSR2.STOP flag is 1b, and then set the ICSR2.NACKF and STOP flags to 0b for the next transfer operation.

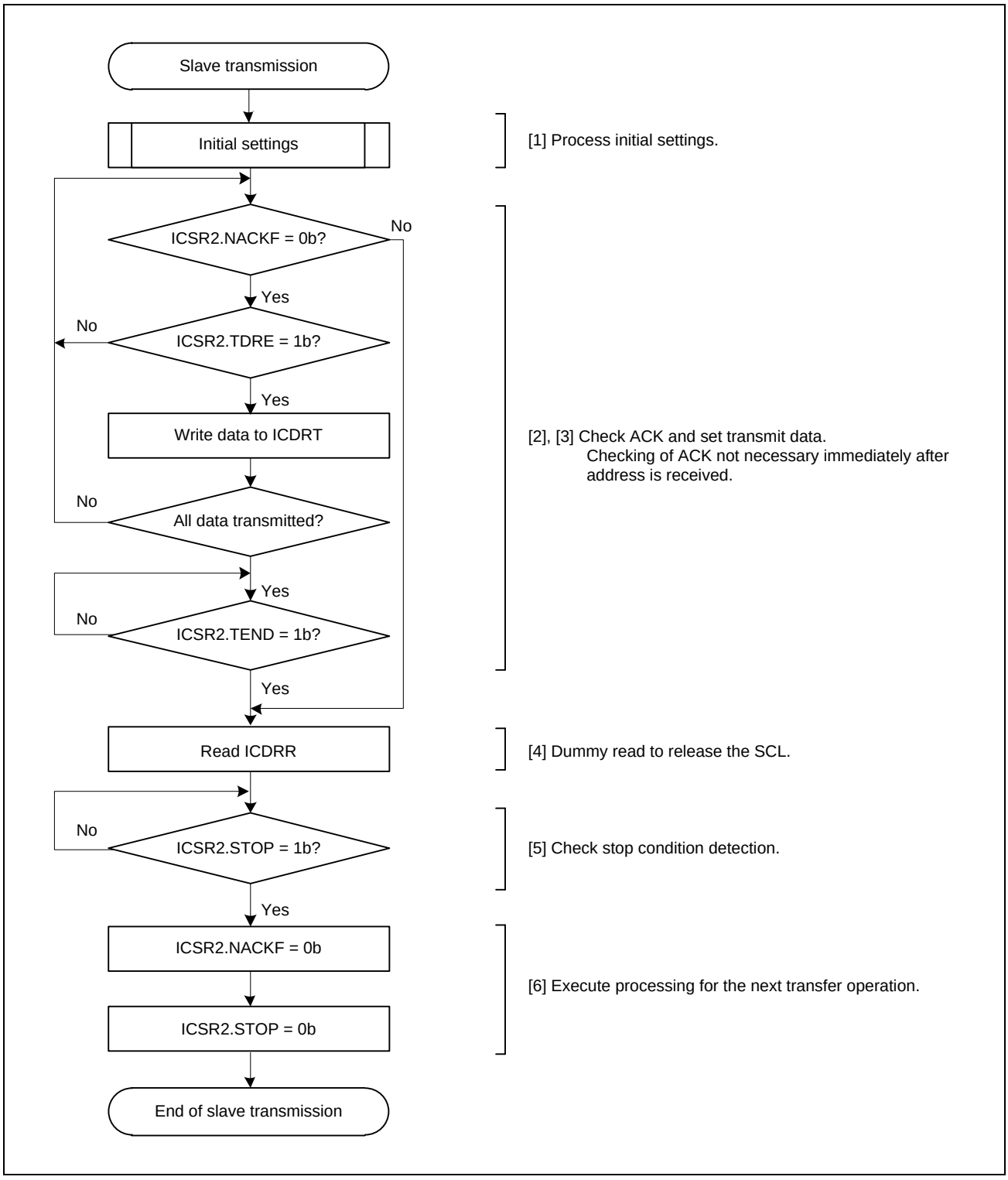


Figure 7.7-15 Example Slave Transmission Flow

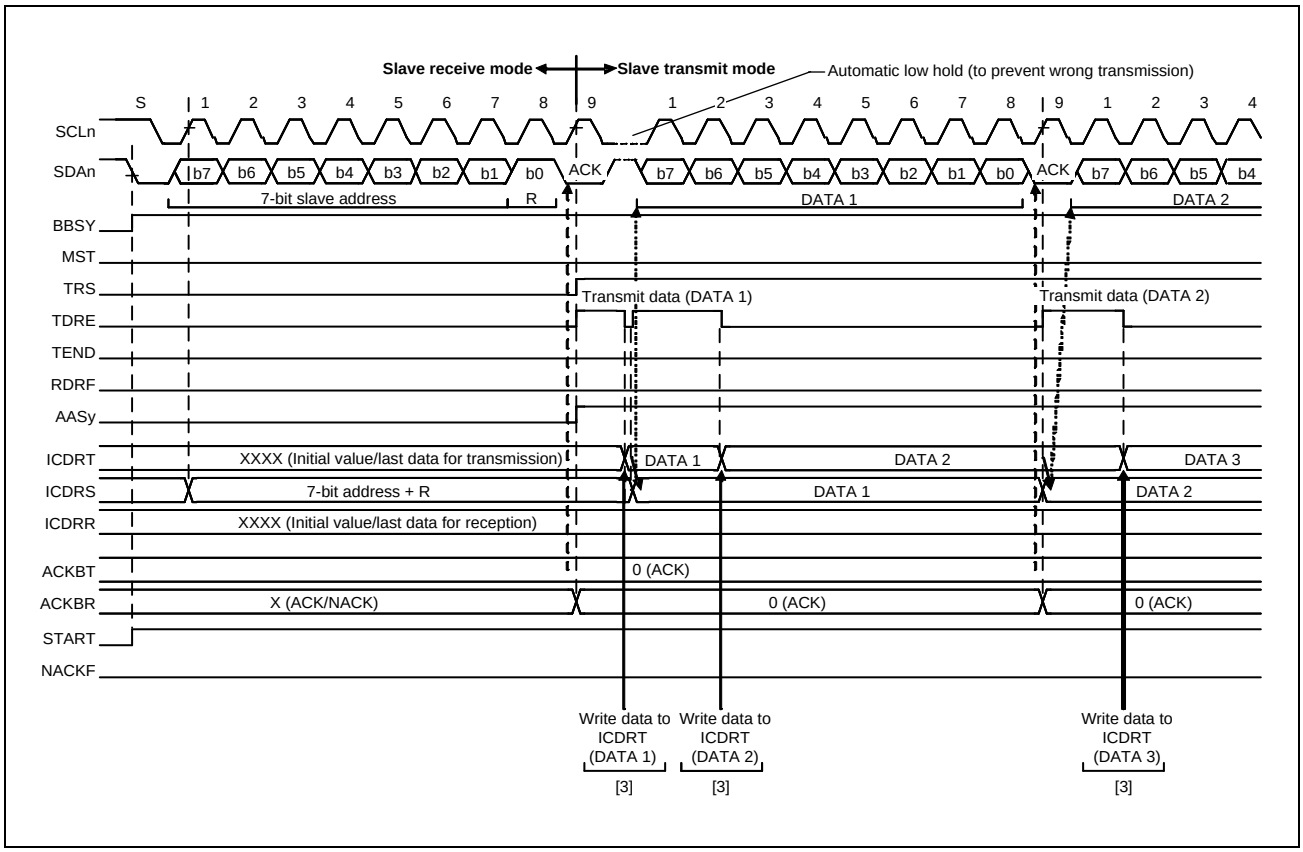


Figure 7.7-16 Slave Transmit Operation Timing (1) with 7-Bit Address Format

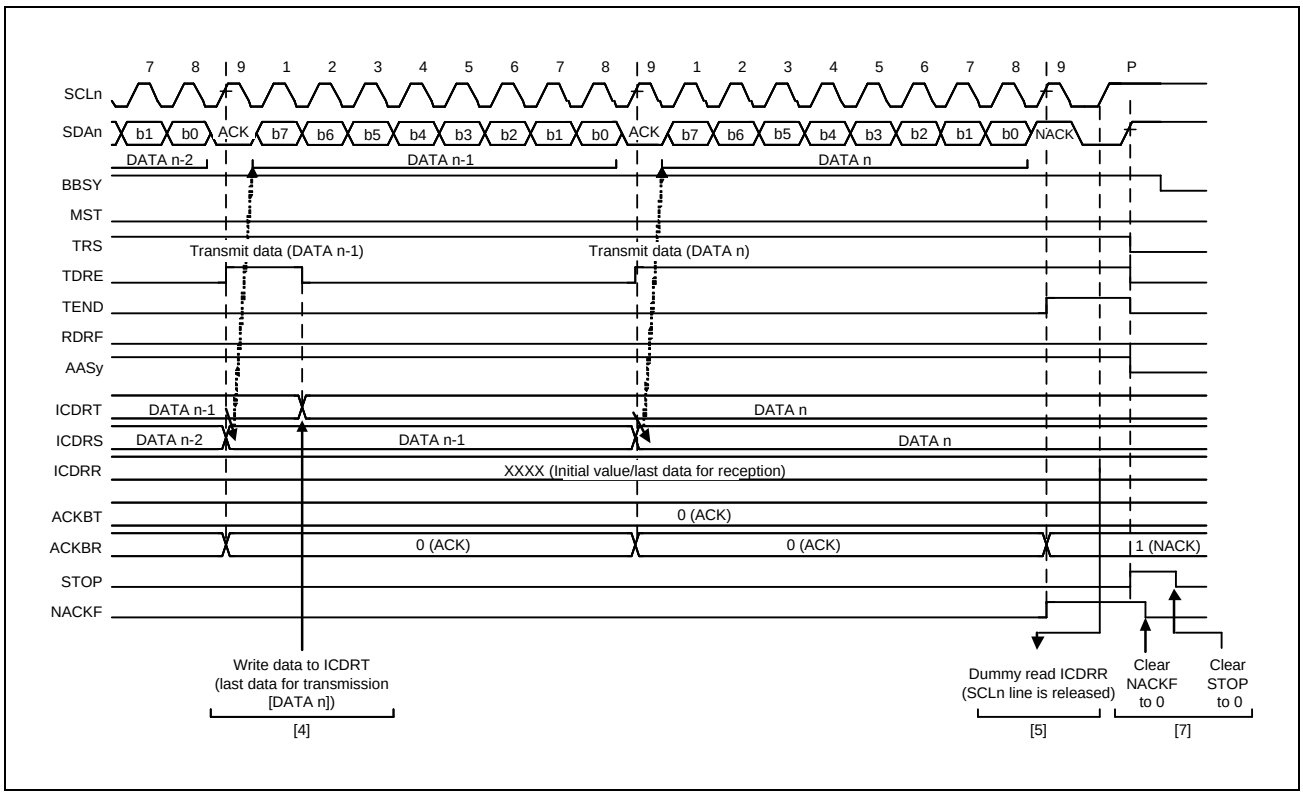


Figure 7.7-17 Slave Transmit Operation Timing (2)

7.7.3.6 Slave Receive Operation

In slave receive operation, the master device outputs the SCL clock and transmit data, and the RIIC returns acknowledgments as a slave device.

Figure 7.7-18 shows an example of slave reception, and **Figure 7.7-19** and **Figure 7.7-20** show the operation timing in slave reception.

To set up and perform slave reception:

1. To initialize the RIIC, follow the procedure in **7.7.3.2 Initial Settings**.
   After the initialization, the RIIC stays in the standby state until it receives a slave address that matches.
2. After receiving a matching slave address, the RIIC sets one of the associated ICSR1.HOA, GCA, and AASy (y = 0 to 2) flags to 1b on the rising edge of the 9th cycle of the SCL clock and outputs the value set in the ICMR3.ACKBT bit to the acknowledge bit on the 9th cycle of the SCL clock. If the value of the received R/W# bit is 0b, the RIIC continues to place itself in slave receive mode and sets the RDRF flag in ICSR2 to 1b.
3. Check that the ICSR2.STOP flag is 0b and the ICSR2.RDRF flag is 1b, and then dummy read the ICDRR register. The dummy value consists of the slave address and R/W# bit when the 7-bit address format is selected, or the lower 8 bits when the 10-bit address format is selected.
4. When ICDRR is read, the RIIC automatically sets the ICSR2.RDRF flag to 0b. If reading of ICDRR is delayed and the next byte is received while the RDRF flag is still set to 1b, the RIIC holds the SCLn line low until 1 SCL cycle before the point where RDRF must be set. In this case, reading ICDRR releases the SCLn line from being held at the low level. When the ICSR2.STOP flag is 1b and the ICSR2.RDRF flag is also 1b, read ICDRR until all the data is completely received.
5. On detecting the stop condition, the RIIC automatically clears the ICSR1.HOA, GCA, and AASy (y = 0 to 2) flags to 0b.
6. Check that the ICSR2.STOP flag is 1b, then set the ICSR2.STOP flag to 0b for the next transfer operation.

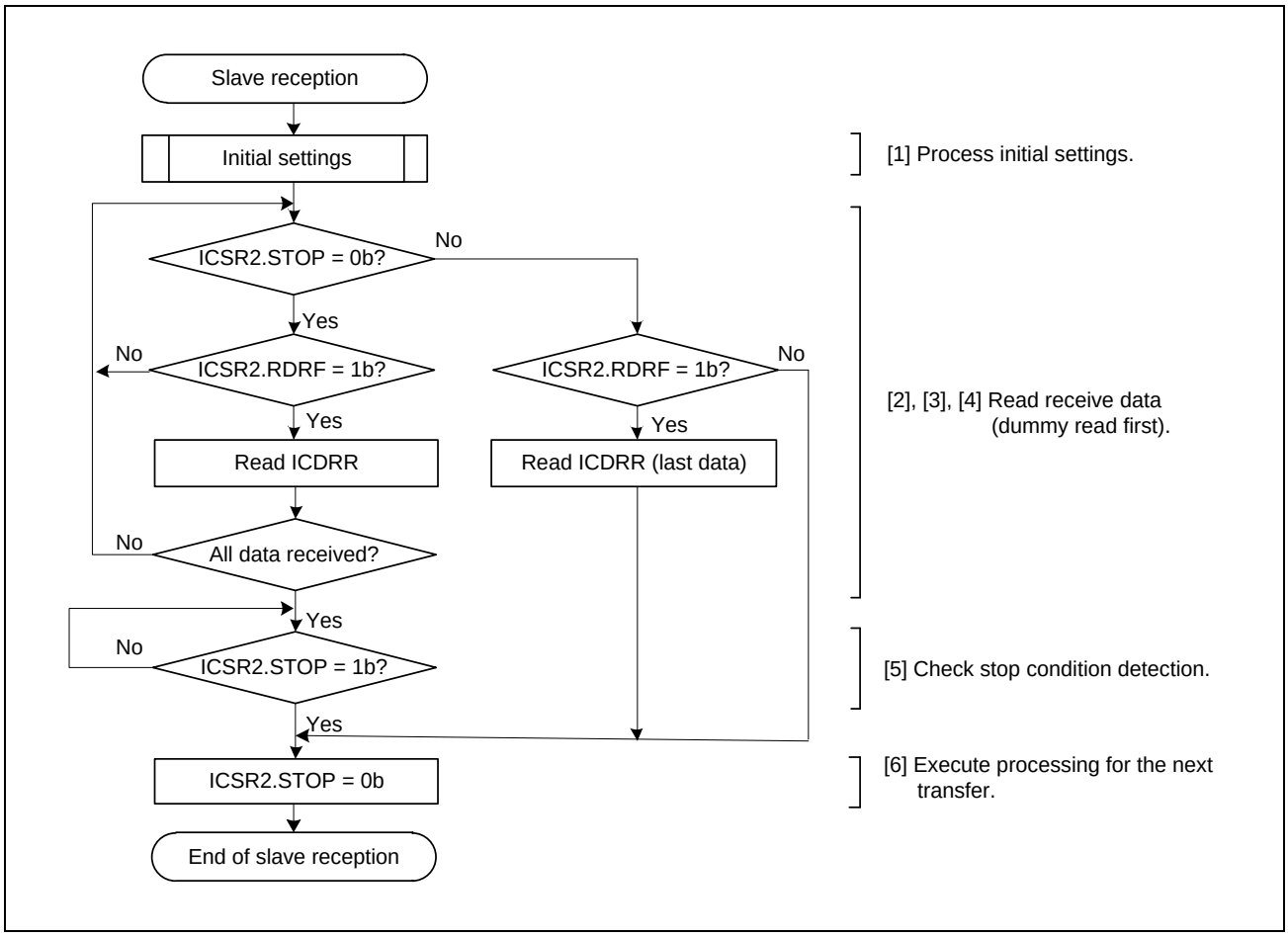


Figure 7.7-18 Example Slave Reception Flow

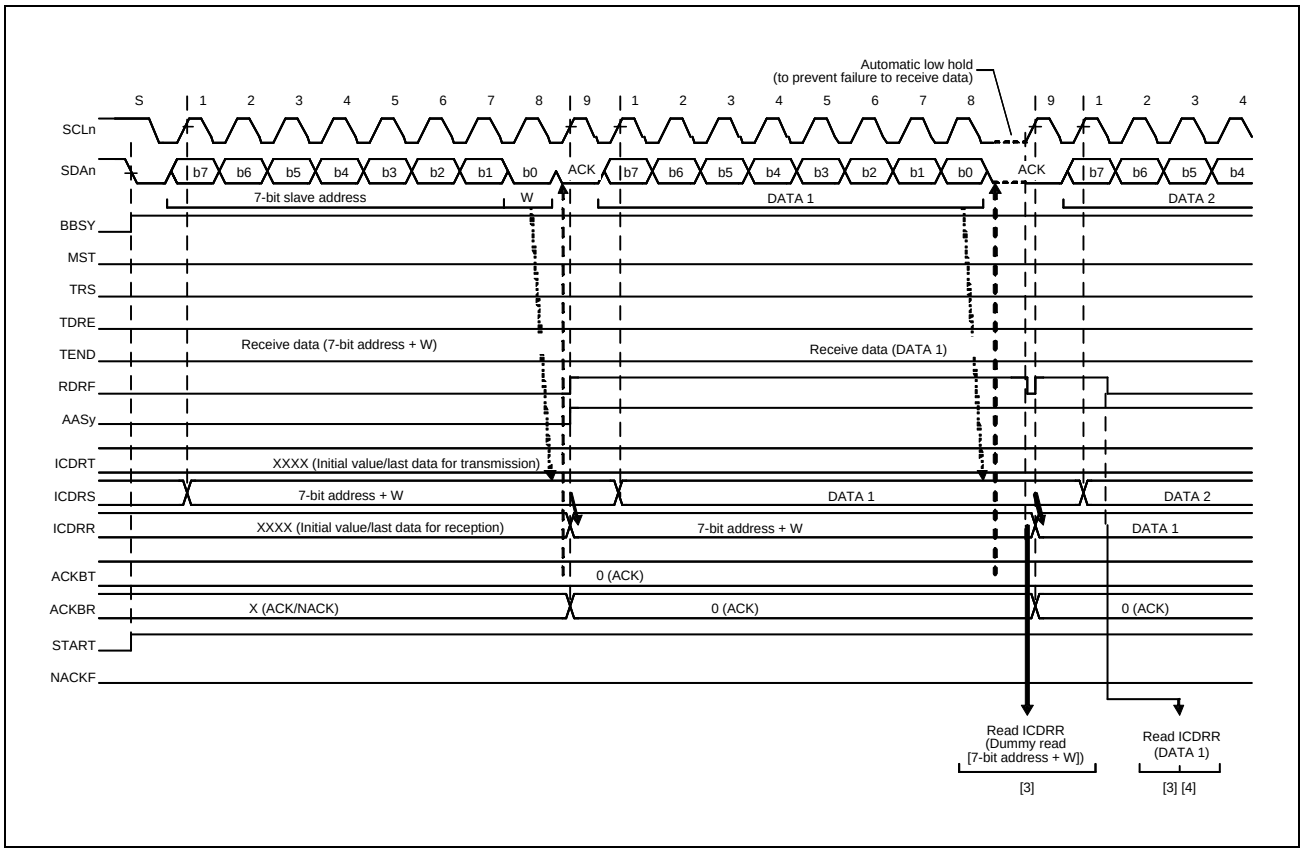


Figure 7.7-19 Slave Receive Operation Timing (1) with 7-Bit Address Format, when RDRFS = 0b

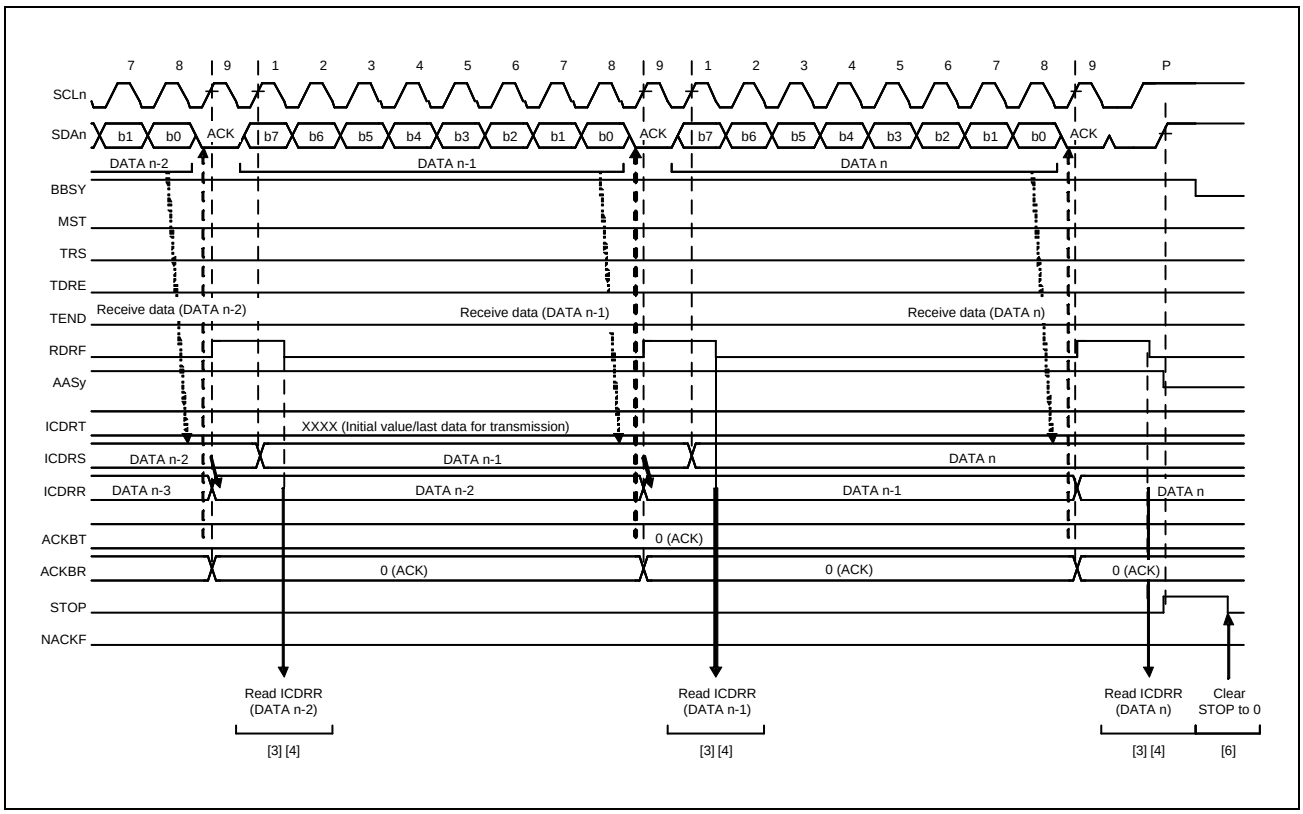


Figure 7.7-20 Slave Receive Operation Timing (2) when RDRFS = 0b

7.7.4 SCL Synchronization Circuit

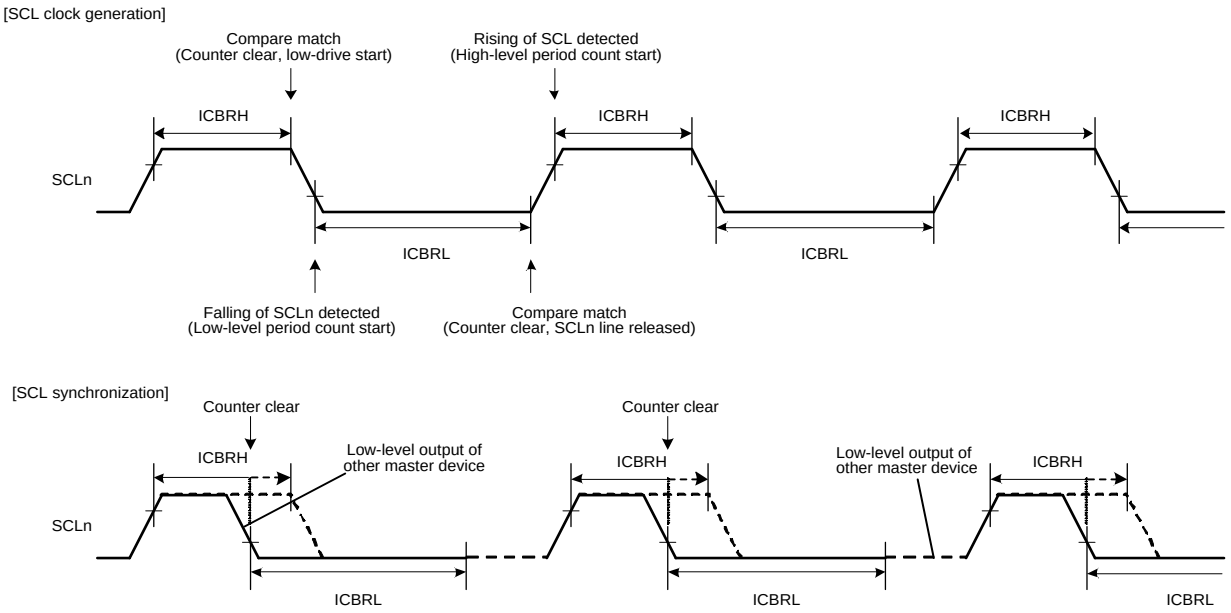
For generation of the SCL clock, RIIC starts counting the value for the high-level period specified in ICBRH when it detects a rising edge on the SCLn line, and drives the SCLn line low when it completes counting. When RIIC detects the falling edge of the SCLn line, it starts counting the value for the low-level period specified in ICBRL, and then stops driving the SCLn line, releasing the line when it completes counting. The RIIC repeats this process to generate the SCL clock.

If multiple master devices are connected to the I^2C bus, a collision of SCL signals might arise because of contention with another master device. In such cases, the master devices must synchronize their SCL signals. Because this synchronization of SCL signals must be bit by bit, RIIC is equipped with an SCL synchronization circuit to obtain bit-by-bit synchronization of the SCL clock signals by monitoring the SCLn line while in master mode.

When RIIC detects a rising edge on the SCLn line and starts counting the high-level period specified in ICBRH, and the level on the SCLn line falls because an SCL signal is being generated by another master device, RIIC performs the following:

1. Stops counting when it detects the falling edge.
2. Drives the level on the SCLn line low.
3. Starts counting the low-level period specified in ICBRL.

When the RIIC finishes counting the low-level period, it stops driving the SCLn line low to release the line. If the low-level period of the SCL clock signal from the other master device is longer than the low-level period set in the RIIC, the low-level period of the SCL signal is extended. When the low-level period for the other master device ends, the SCL signal rises because the SCLn line is released. When the RIIC finishes outputting the low-level period of the SCL clock, the SCLn line is released and the SCL clock rises. That is, when SCL signals from more than one master are contending, the high-level period of the SCL signal is synchronized with that of the clock with the narrower period, and the low-level period of the SCL signal is synchronized with that of the clock with the broader period. However, such synchronization of the SCL signal is only enabled when the SCLE bit in ICFER is set to 1b.



Note: ICBRH: I^2C Bus Bit Rate High-Level Register (SCL clock high-level period counter)
ICBRL: I^2C Bus Bit Rate Low-Level Register (SCL clock low-level period counter)

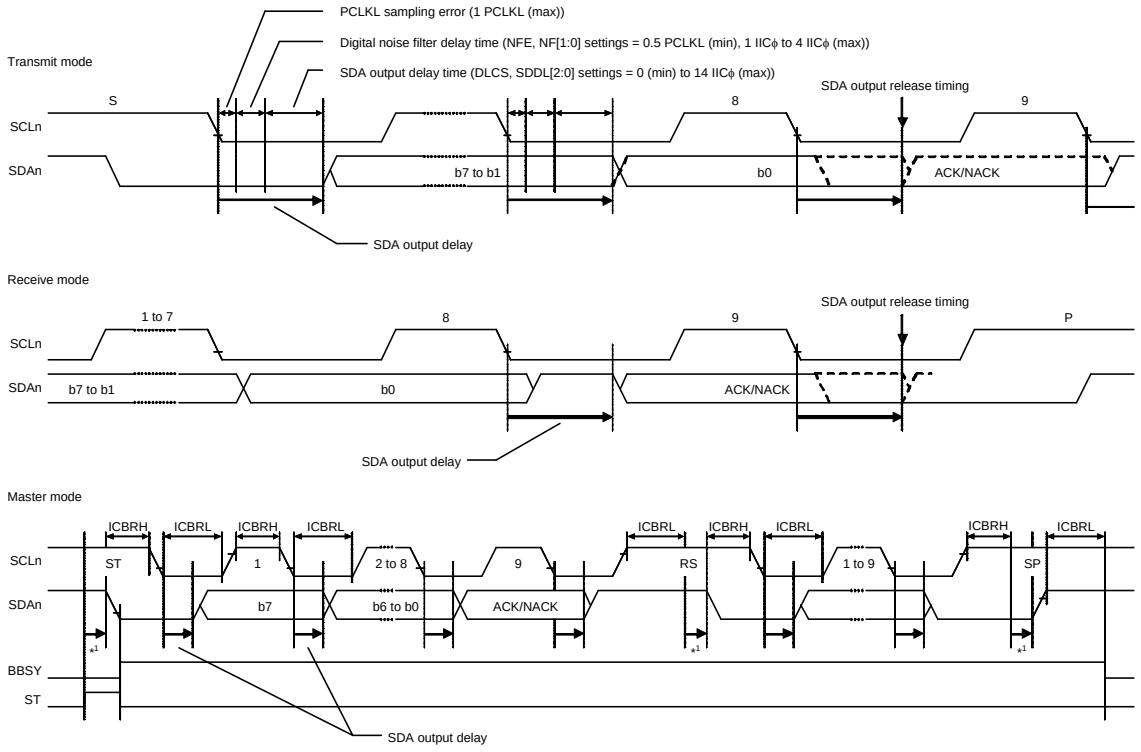
Figure 7.7-21 Generation and Synchronization of SCL Signal from RIIC

7.7.5 SDA Output Delay Function

The RIIC module provides a function for delaying the output on the SDA line. The delay can be applied to all outputs on the SDA line, including the issuing of start, restart, and stop conditions, data, and the ACK and NACK signals.

With this function, SDA output is delayed from the detection of a falling edge of the SCL signal to ensure that the SDA signal is output within the interval during which the SCL clock is low. This approach helps prevent erroneous operation of communications devices, with the aim of satisfying the 300 ns minimum data-hold time requirement of the SMBus specification. The output delay function is enabled by setting the SDDL[2:0] bits in ICMR2 to any value other than 000b, and disabled by setting the same bits to 000b.

When the SDA output delay function is enabled, for example, the DLCS bit in ICMR2 selects the clock source for the SDA output delay counter, either as the internal base clock (IICϕ) for the RIIC module or as the internal base clock divided by two (IICϕ/2). The counter counts the number of cycles set in the SDDL[2:0] bits in ICMR2. When the delay count is reached, the RIIC module places the required output (start, restart, or stop condition, data, or an ACK or NACK signal) on the SDA line.



Note 1. The output delay function is set by the DLCS and SDDL[2:0] bits when a start (ST), restart (RS), or stop (SP) condition is issued.

Figure 7.7-22 SDA Output Delay Function

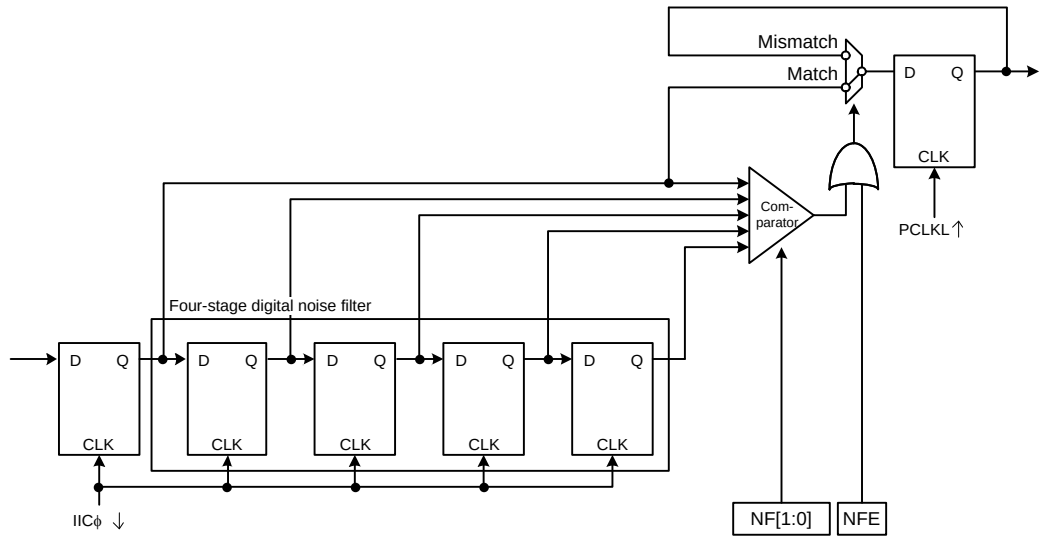
7.7.6 Digital Noise Filter Circuits

The internal circuitry sees the states of the SCLn and SDAn pins through analog and digital noise-filter circuits. **Figure 7.7-23** shows a block diagram of the digital noise-filter circuit.

The on-chip digital noise-filter circuit of the RIIC consists of four flip-flop circuit stages connected in series, and a match- detection circuit.

The number of valid stages in the digital noise filter is selected in the NF[1:0] bits in ICMR3. The selected number of valid stages determines the noise-filtering capability as a period from 1 to 4 IICϕ cycles.

The input signal to the SCLn pin (or SDAn pin) is sampled on falling edges of the RIICϕ signal. When the input signal level matches the output level of the number of valid flip-flop circuit stages as selected in the NF[1:0] bits in ICMR3, the signal level is conveyed to the subsequent stage. If the signal levels do not match, the previous value is saved.



Note: NFE: Digital Noise Filter Circuit Enable bit
NF[1:0]: Noise Filter Stage Select bits

Figure 7.7-23 Digital Noise Filter Circuit Block Diagram

7.7.7 Address Match Detection

IIC can set three unique slave addresses in addition to the general call address and host address. The slave addresses can be 7-bit or 10-bit slave addresses.

7.7.7.1 Slave-Address Match Detection

IIC can set three unique slave addresses and has a slave address detection function for each unique slave address. When the SARyE bit (y = 0 to 2) in ICSER is set to 1b, the slave addresses set in SARUy and SARLy (y = 0 to 2) can be detected.

When the RIIC detects a match of the set slave address, the associated AASy (y = 0 to 2) flag in ICSR1 is set to 1b on the rising edge of the 9th SCL clock cycle, and the RDRF flag in ICSR2 or the TDRE flag in ICSR2 is set to 1b by the subsequent R/W# bit. This causes a receive data full interrupt (IICn_RXI) or transmit data empty interrupt (IICn_TXI) to be generated. The AASy flag identifies which slave address is specified.

Figure 7.7-24 to **Figure 7.7-26** show the AASy flag set timing in three cases.

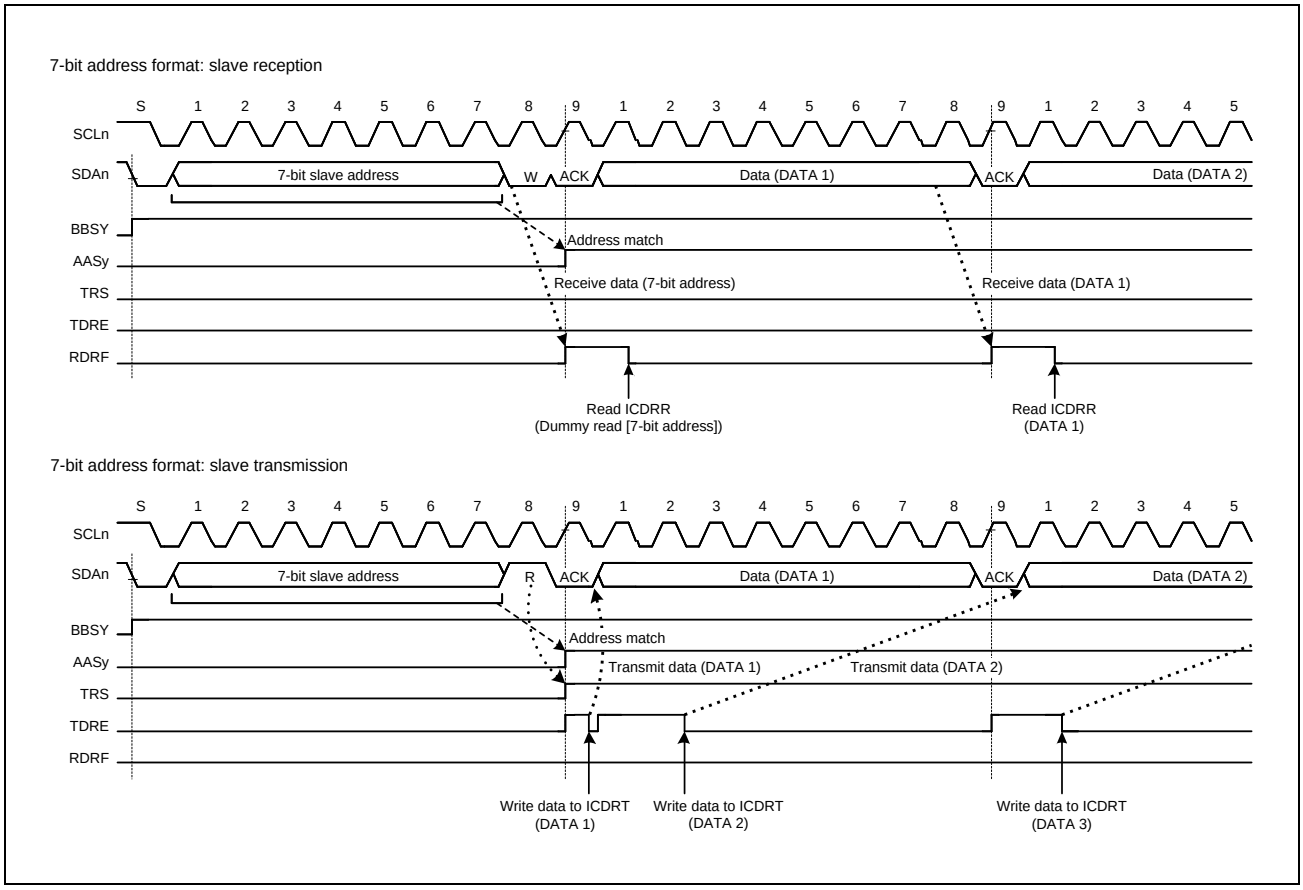


Figure 7.7-24 AASy Flag Set Timing with 7-Bit Address Format

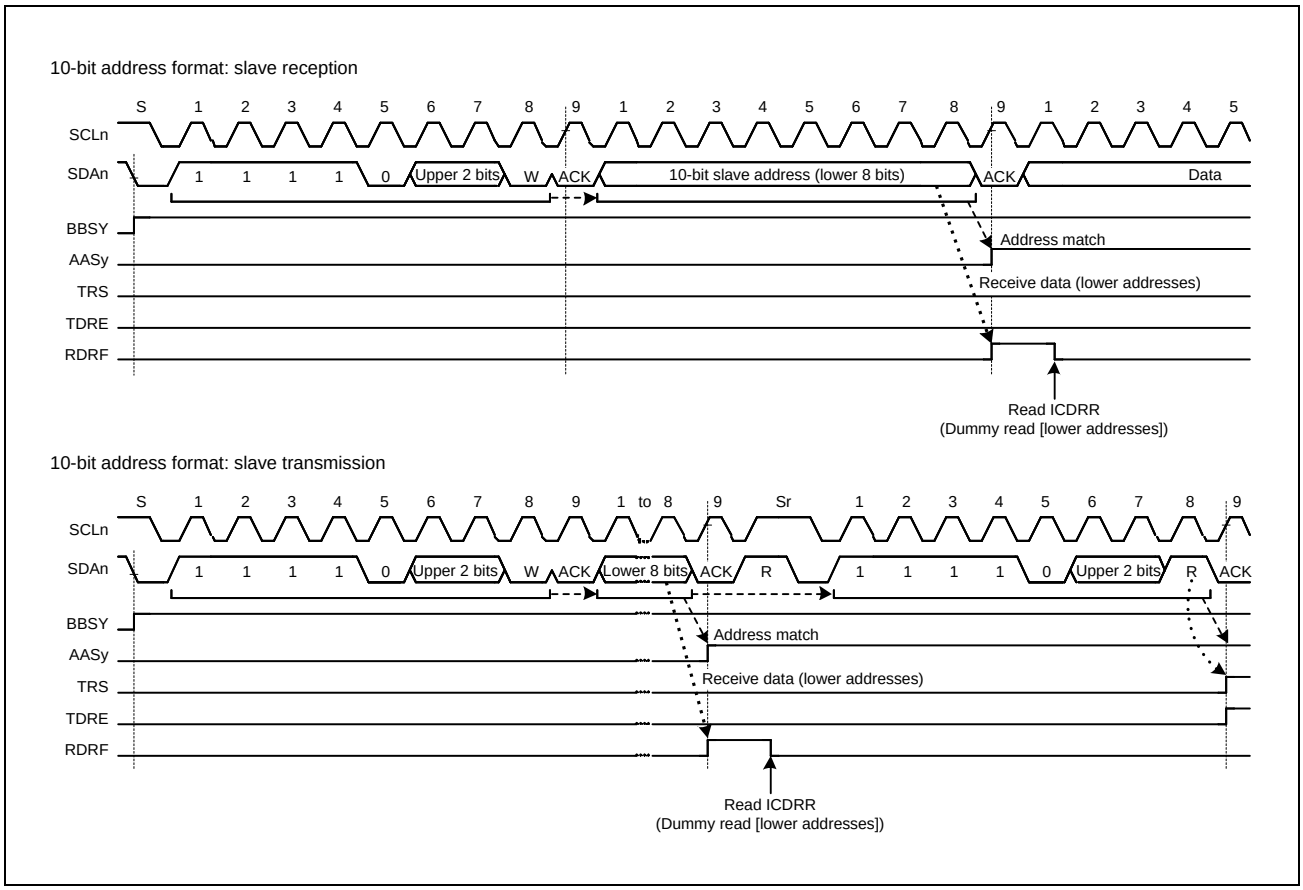


Figure 7.7-25 AASy Flag Set Timing with 10-Bit Address Format

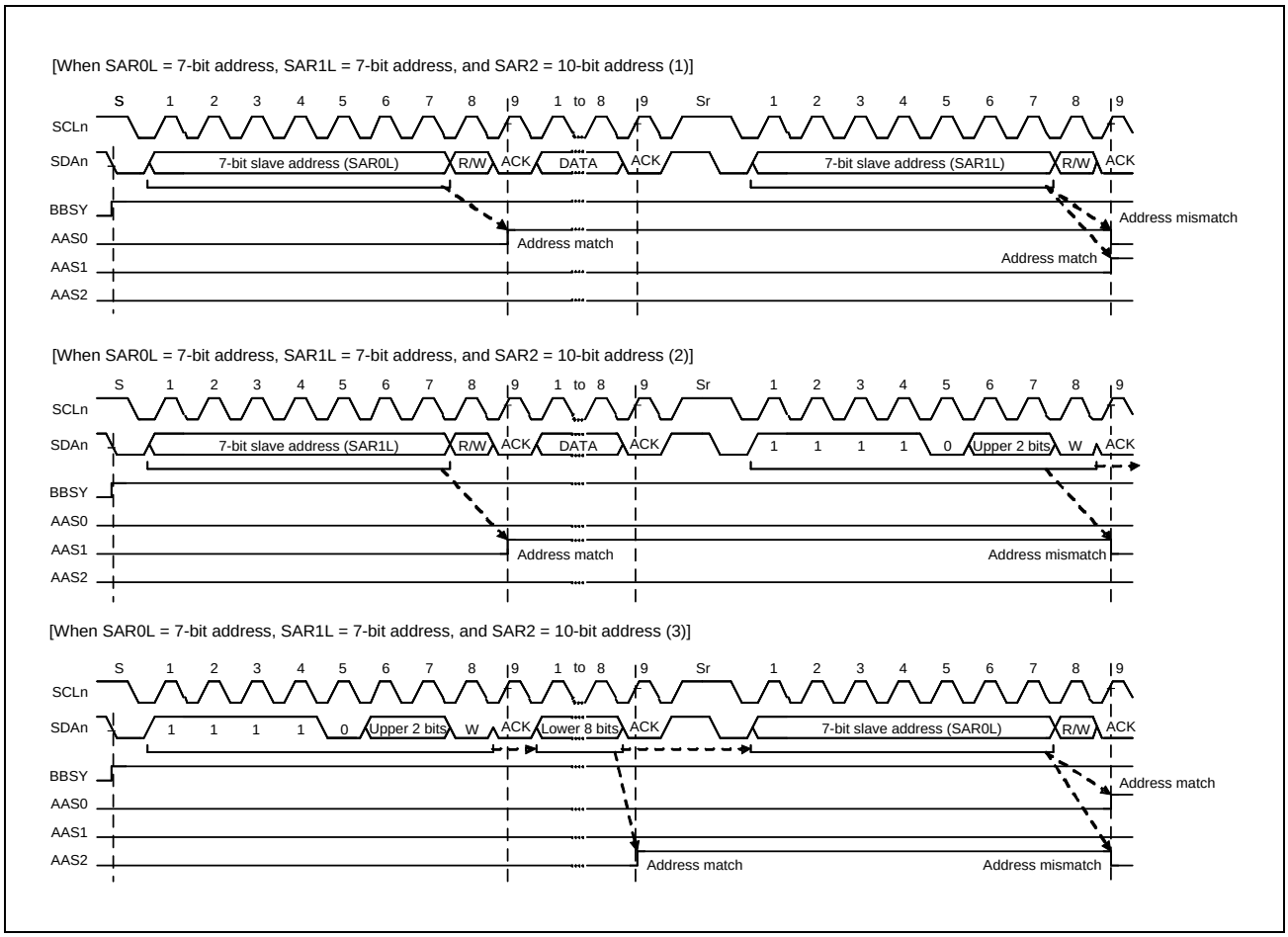


Figure 7.7-26 AASy Flag Set and Clear Timing with 7- and 10-Bit Address Formats Mixed

7.7.7.2 Detection of General Call Address

The RIIC provides detection of the general call address (0000_000b + 0b [W]). General call address detection is enabled by setting the GCAE bit in ICSER to 1b.

If the address received after a start or restart condition is issued is 0000_000b + 1b [R] (start byte), the RIIC recognizes this as the address of a slave device with an all-zero address, but not as the general call address.

When the RIIC detects the general call address, both the GCA flag in ICSR1 and the RDRF flag in ICSR2 set to 1b on the rising edge of the 9th cycle of the SCL clock. This leads to the generation of a receive data full interrupt (IICn_RXI). The value of the GCA flag can be checked to confirm that the general call address is transmitted.

Operation after detection of the general call address is the same as normal slave receive operation.

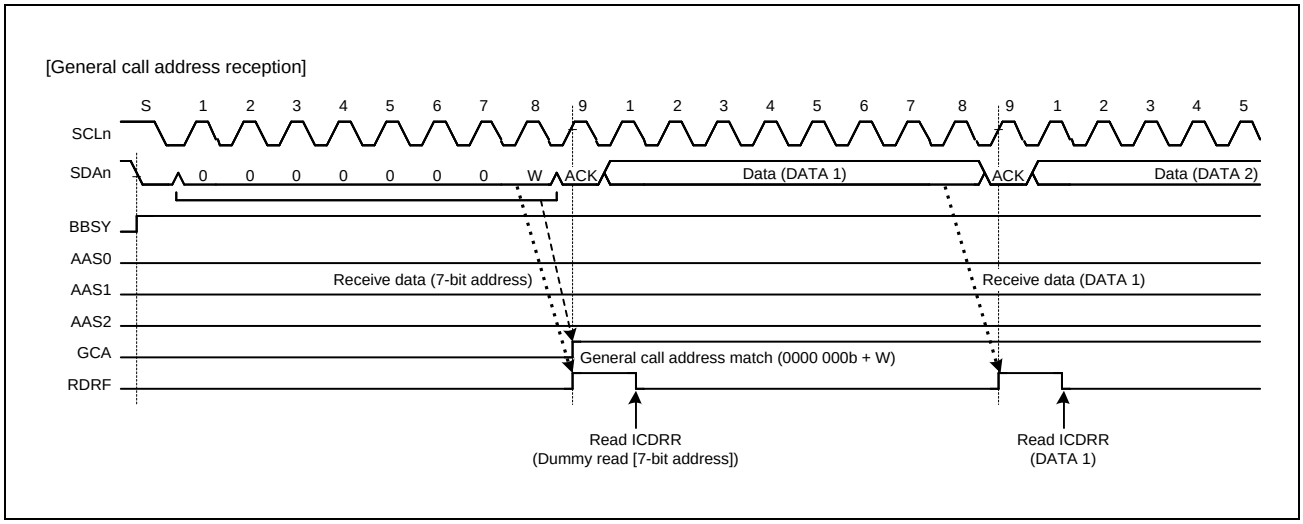


Figure 7.7-27 Timing of GCA Flag Setting during Reception of General Call Address

7.7.7.3 Device ID Address Detection

The RIIC module provides detection of the device ID address in compliance with the I^2C bus specification.

When RIIC receives 111_1100b as the first seven bits of the first byte after a start or restart condition is issued with the DIDE bit in ICSER set to 1b, it recognizes the address as a device ID address, sets the DID flag in ICSR1 to 1b on the rising edge of the 9th SCL clock cycle when the subsequent R/W# bit is 0b, and then compares the second and subsequent bytes with its own slave address. If the address matches the value in the slave address register, the RIIC sets the associated AASy (y = 0 to 2) flag in ICSR1 to 1b.

When the first byte received after the issue of a start or restart condition matches the device ID address (1111_100b) again and the subsequent R/W# bit is 1b, RIIC does not compare the second and subsequent bytes, and sets the ICSR2.TDRE flag to 1b.

In the device ID address detection function, RIIC sets the DID flag to 0b if a match with the RIIC slave address is not obtained, or a match with the device ID address is not obtained after a match with the RIIC slave address, and a restart condition is detected. If the first byte after the detection of a start or restart condition matches the device ID address (1111_100b) and the R/W# bit is 0b, RIIC sets the DID flag to 1b and compares the second and subsequent bytes with the slave address of RIIC. If the R/W# bit is 1b, the DID flag holds the previous value and RIIC does not compare the second and subsequent bytes. In this way, reception of a device ID address can be checked by reading the DID flag after confirming that TDRE = 1b.

Additionally, prepare the device ID fields (3 bytes: 12 bits indicating the manufacturer + 9 bits identifying the part + 3 bits indicating the revision) that must be sent to the host after reception of a continuous device ID field as normal transmit data. For details on the information that must be included in device ID fields, contact NXP Semiconductors.

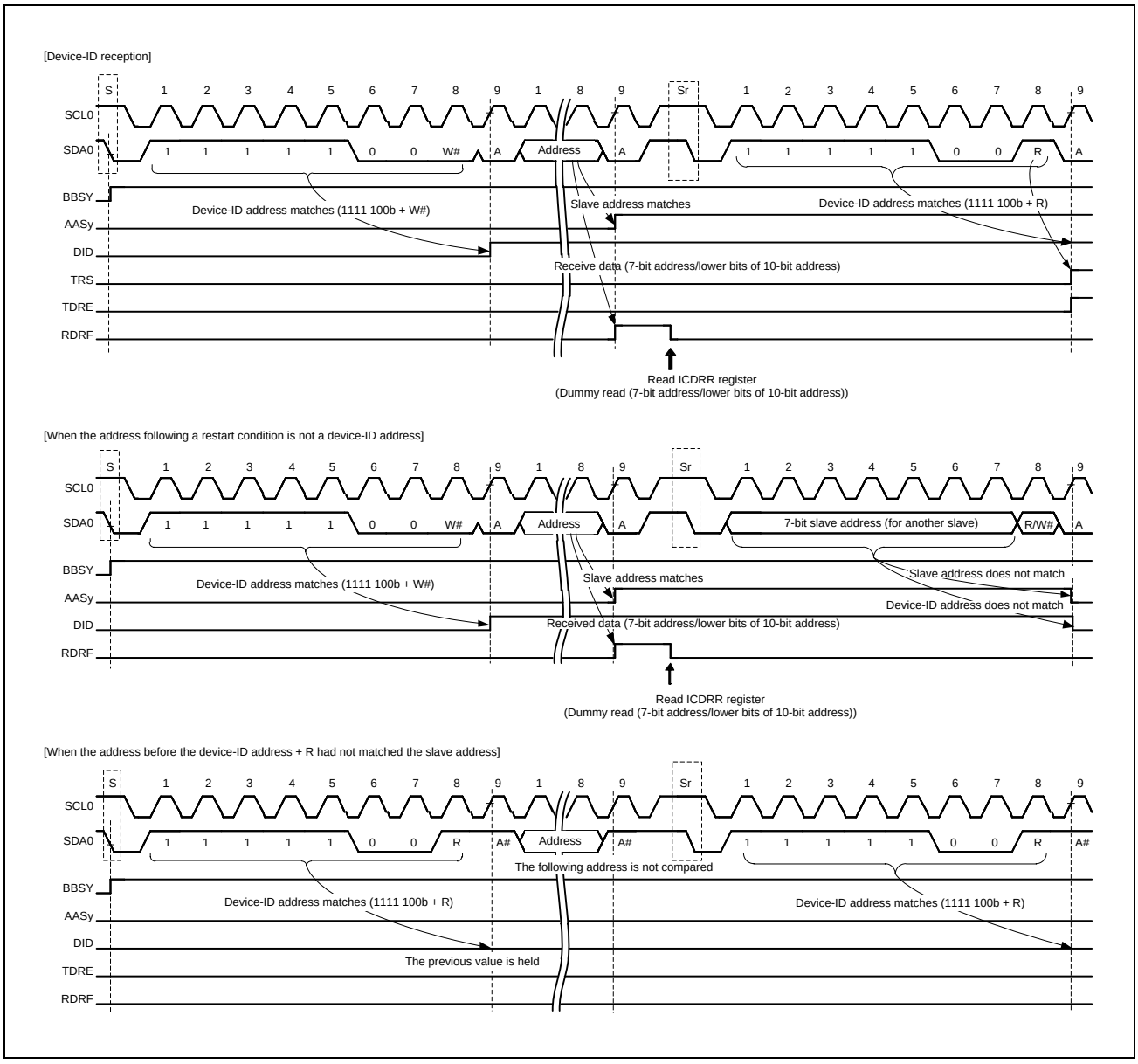


Figure 7.7-28 AASy and DID Flag Set and Clear Timing during Reception of Device ID

7.7.7.4 Host Address Detection

The RIIC provides host address detection while the SMBus is operating. When the HOAE bit in ICSER is set to 1 while the SMBS bit in ICMR3 is 1b, the RIIC can detect the host address (0001 000b) in slave receive mode (MST and TRS bits = 00b in ICCR2).

When the RIIC detects the host address, the HOA flag in ICSR1b is set to 1 on the rising edge of the 9th SCL clock cycle. At the same time, the RDRF flag in ICSR2 is set to 1b when the R/W# bit is 0b. This causes a receive data full interrupt (IICn_RXI) to be generated. The HOA flag indicates that the host address was sent from another device.

If the bit following the host address (0001 000b) is a read bit (R/W# bit = 1b), the RIIC can also detect the host address. After the host address is detected, the RIIC operates in the same manner as in normal slave operation.

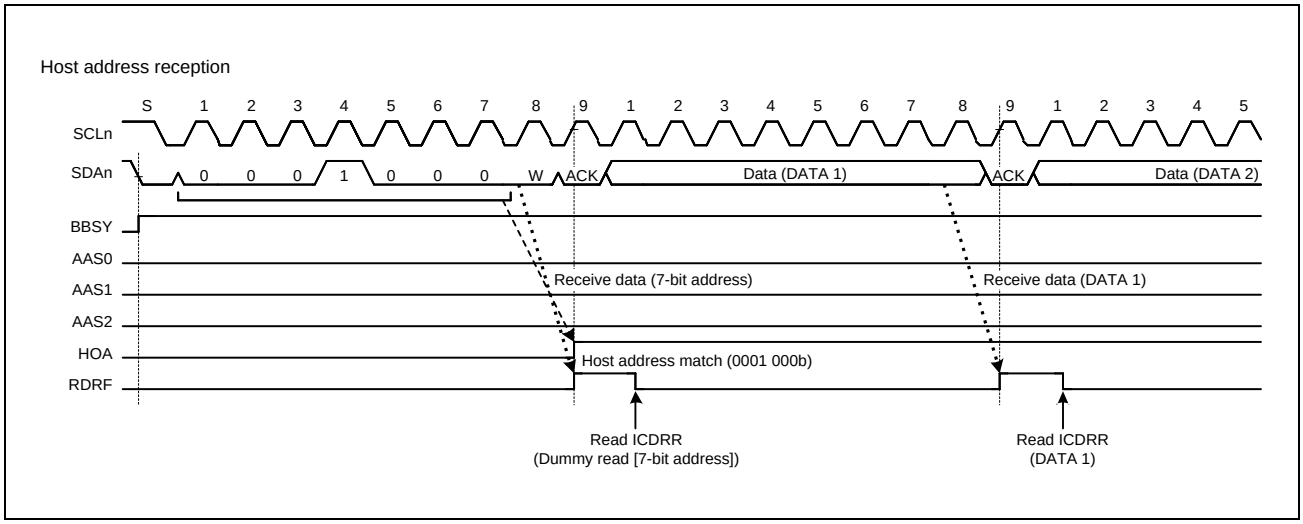


Figure 7.7-29 HOA Flag Set Timing during Reception of Host Address

7.7.8 Automatic Low-Hold Function for SCL

7.7.8.1 Function to Prevent Wrong Transmission of Transmit Data

If the I²C Bus Shift Register (ICDRS) is empty when data has not been written to the I²C Bus Transmit Data Register (ICDRT) with the RIIC in transmission mode (TRS bit = 1b in ICCR2), the SCLn line is automatically held at the low level over subsequent intervals. This low-hold period is extended until the transmit data is written, which prevents the unintended transmission of erroneous data.

- Master transmit mode
  - Low-level interval after a start or restart condition is issued
  - Low-level interval between the 9th clock cycle of one transfer and the 1st clock cycle of the next.
- Slave transmit mode
  - Low-level interval between the 9th clock cycle of one transfer and the 1st clock cycle of the next.

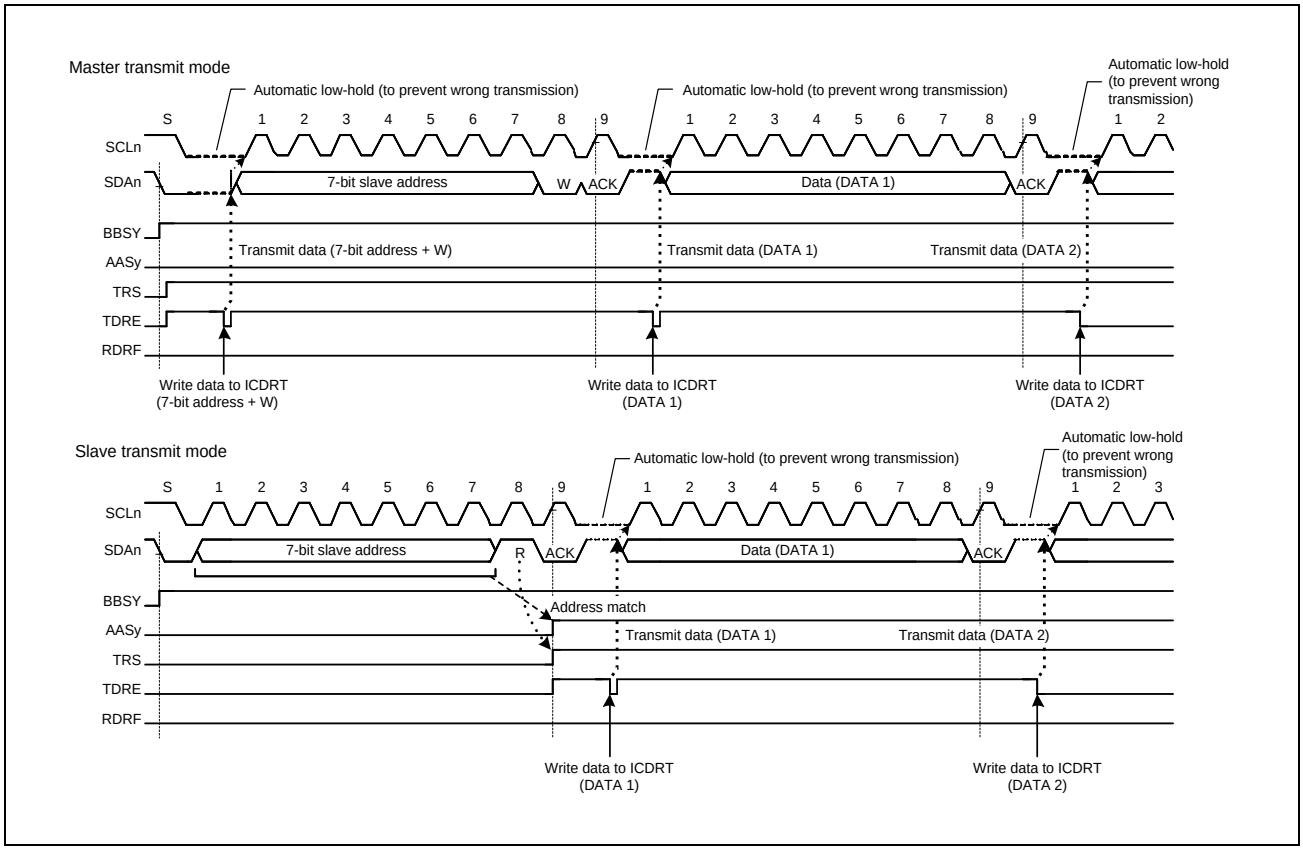


Figure 7.7-30 Automatic Low-Hold Operation in Transmit Mode

7.7.8.2 NACK Reception Transfer Suspension Function

This function suspends transfer operation when NACK is received in transmit mode (TRS bit = 1b in ICCR2). This function is enabled when the NACKE bit in ICFER is set to 1b. If the next transmit data is already written (TDRE flag = 0b in ICSR2) when NACK is received, the next data transmission on the falling edge of the 9th SCL clock cycle is automatically suspended. This prevents the SDAn line output level from being held low when the MSB of the next transmit data is 0.

If the data transmission is suspended by this function (NACKF flag = 1b in ICSR2), the following data transmission and data reception are not started. To resume data transfer, set the NACKF flag to 0b. In master transmit mode, restart data transfer by setting the NACKF flag to 0b after generating a restart condition, or restart data transfer from a start condition after generating a stop condition and setting the NACKF flag to 0b.

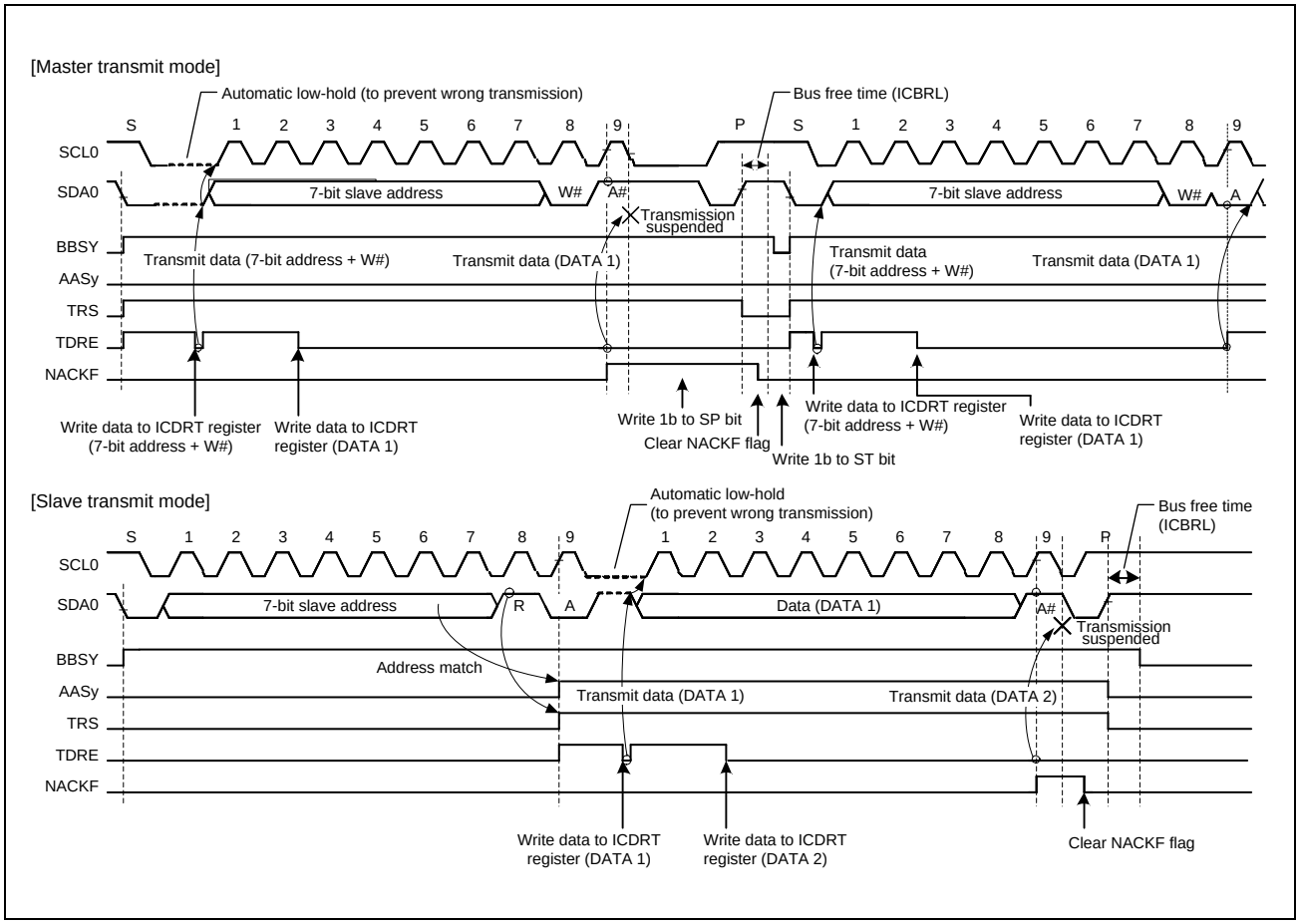


Figure 7.7-31 Suspension of Data Transfer when NACK is Received, when NACKE = 1

7.7.8.3 Function to Prevent Failure to Receive Data

If response processing is delayed when receive data (ICDRR) read is delayed for a period of one transfer frame or more with receive data full (RDRF flag = 1b in ICSR2) in receive mode (TRS = 0b in ICCR2), the RIIC holds the SCLn line low automatically immediately before the next data is received to prevent failure to receive data.

This function is also enabled even if the read processing of the final receive data is delayed and, in the meantime, the RIIC slave address is designated after a stop condition is issued. This function does not interfere with other communication because the RIIC does not hold the SCLn line low when a mismatch with its own slave address occurs after a stop condition is issued.

Periods in which the SCLn line is held low can be selected with a combination of the WAIT and RDRFS bits in ICMR3.

7.7.8.3.1 1-byte receive operation and automatic low-hold function using the WAIT bit

When the WAIT bit in ICMR3 is set to 1b, RIIC performs a 1-byte receive operation using the WAIT bit function. Additionally, when the ICMR3.RDRFS bit is 0b, RIIC automatically sends the ACKBT bit value in ICMR3 for the acknowledge bit in the period from the falling edge of the 8th SCL clock cycle to the falling edge of the 9th SCL clock cycle, and automatically holds the SCLn line low on the falling edge of the 9th SCL clock cycle using the WAIT bit function. This low-hold is released by reading data from ICDRR, which enables byte-wise receive operation.

The WAIT bit function is enabled for receive frames after a match with the RIIC slave address, including the general call address and host address, is obtained in master or slave receive mode.

7.7.8.3.2 1-byte receive operation (ACK/NACK transmission control) and automatic low-hold function using the RDRFS bit

When the RDRFS bit in ICMR3 is set to 1b, the RIIC performs a 1-byte receive operation using the RDRFS bit function.

When the RDRFS bit is set to 1b, the RDRF (receive data full) flag in ICSR2 is set to 1 on the rising edge of the 8th SCL clock cycle, and the SCLn line is automatically held low on the falling edge of the 8th SCL clock cycle. This low-hold is released by writing a value to the ACKBT bit in ICMR3, but cannot be released by reading data from ICDRR, which enables receive operation through the ACK or NACK transmission control based on the data received in byte units.

The RDRFS bit function is enabled for receive frames after a match with the RIIC slave address, including the general call address and host address, is obtained in master or slave receive mode.

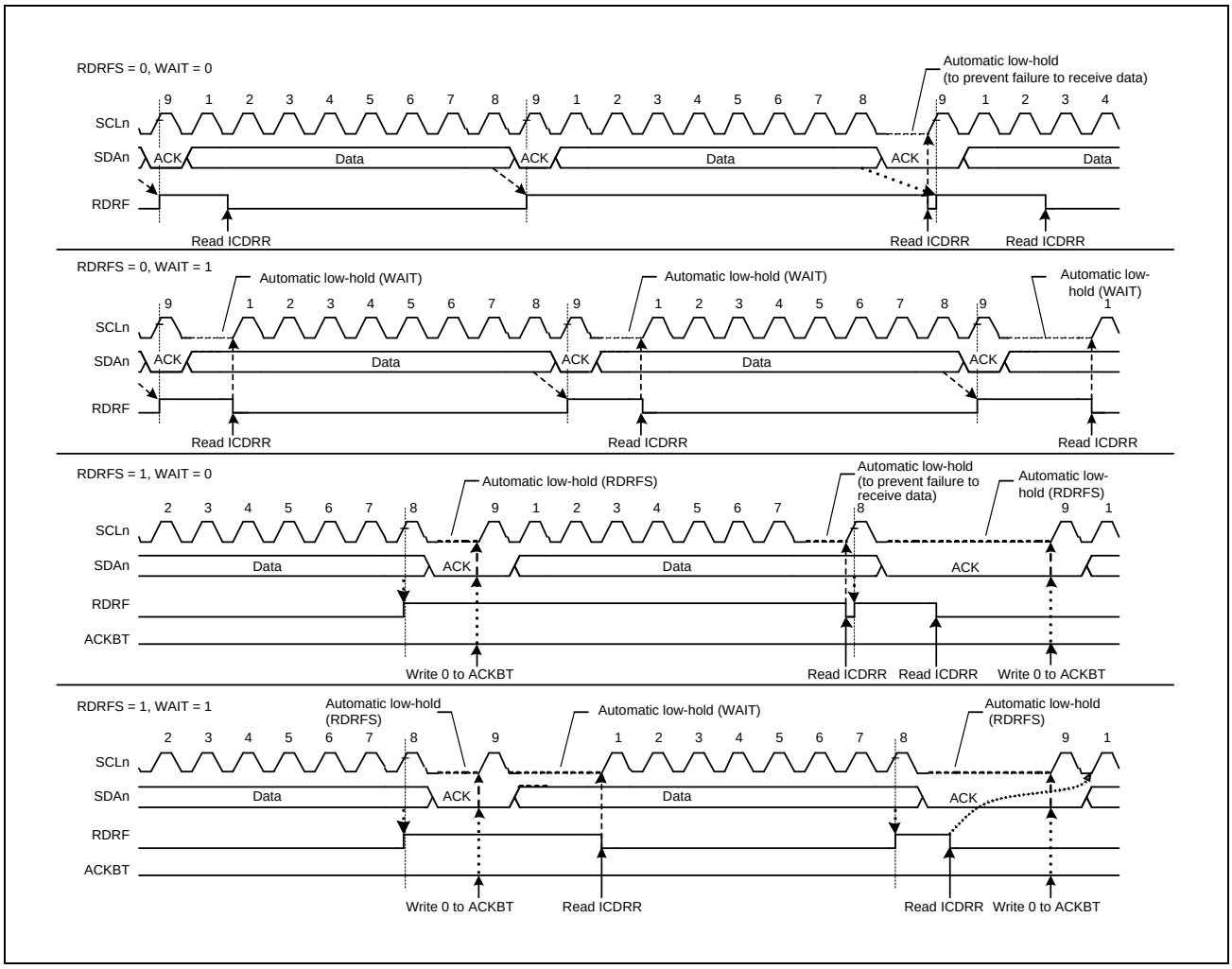


Figure 7.7-32 Automatic Low-Hold Operation in Receive Mode using the RDRFS and WAIT Bits

7.7.9 Arbitration-Lost Detection Functions

In addition to the normal arbitration-lost detection function defined by the I^2C bus standard, RIIC provides functions to prevent double-issue of a start condition, detect arbitration-lost during the transmission of NACK, and detect arbitration lost in slave transmit mode.

7.7.9.1 Master Arbitration-Lost Detection (MALE Bit)

RIIC drives the SDAn line low to issue a start condition. However, if the SDAn line was already driven low by another master device issuing a start condition, RIIC regards its own start condition as an error and considers this a loss in arbitration.

Priority is given to transfer by the other master device. Similarly, if a request to issue a start condition is made by setting the ST bit in ICCR2 to 1b while the bus is busy (BBSY flag = 1b in ICCR2), RIIC regards this as a double-issuing-of-start- condition error and considers itself to have lost the arbitration. This prevents a failure of transfer resulting from a start condition issued while transfer is in progress.

When a start condition is issued successfully, if the transmit data including the address bits (internal SDA output level) and the level on the SDAn line do not match (high output as the internal SDA output, meaning the SDAn pin is in the high- impedance state) and a low level is detected on the SDAn line, RIIC loses the arbitration.

After a loss in arbitration of mastership, the RIIC immediately enters slave receive mode. If a slave address, including the general call address, matches its own address at this time, the RIIC continues in slave operation.

A loss in arbitration of mastership is detected when the following conditions are met while the MALE bit in ICFER is 1b (master arbitration-lost detection enabled).

[Master arbitration-lost conditions]

- Mismatching of the internal level for output on SDA and the level on the SDAn line after a start condition was issued by setting the ST bit in ICCR2 to 1b while the BBSY flag in ICCR2 is set to 0b (erroneous issuing of a start condition)
- Setting of the ST bit in ICCR2 to 1b (start condition double-issue error) while the BBSY flag is 1b
- When the transmit data excluding acknowledge (internal SDA output level) does not match the level on the SDAn line in master transmit mode (MST and TRS bits = 11b in ICCR2).

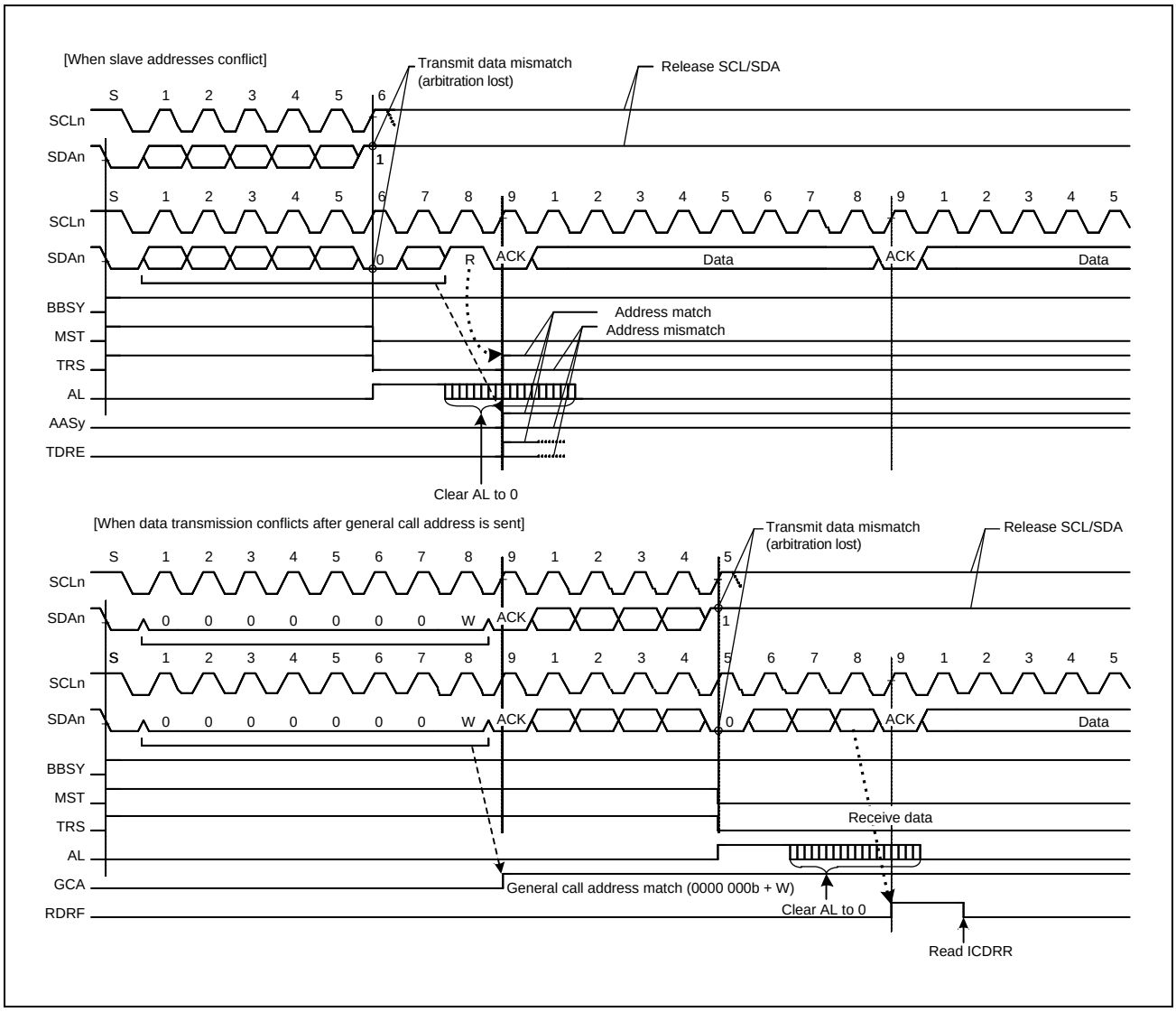


Figure 7.7-33 Examples of Master Arbitration-Lost Detection when MALE = 1

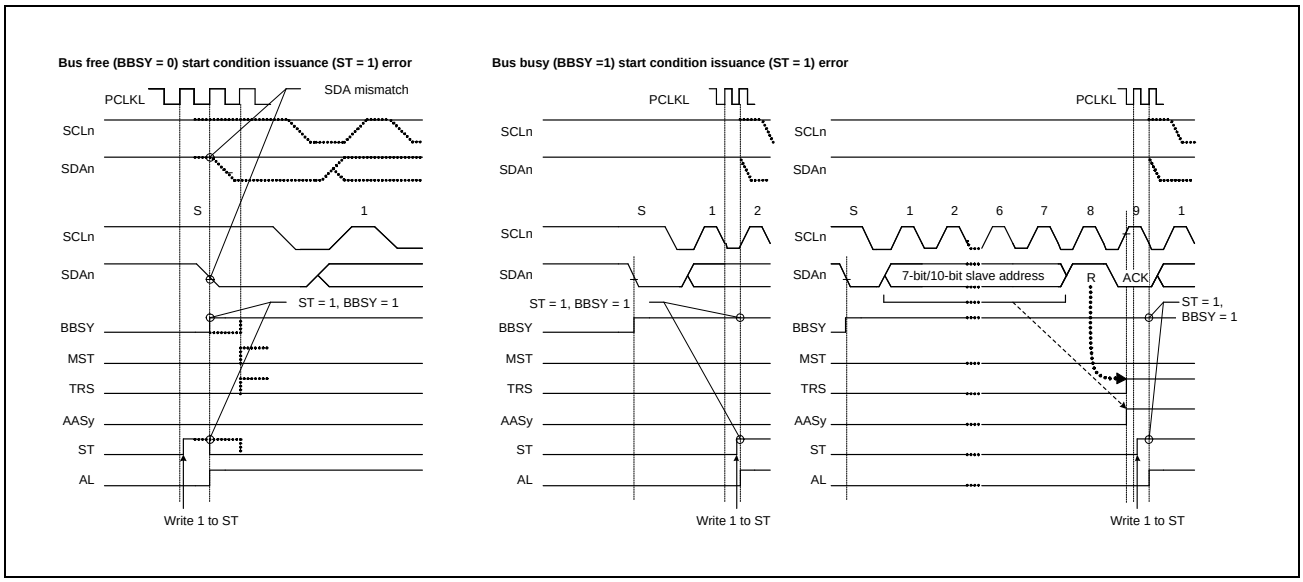


Figure 7.7-34 Arbitration-Lost when Start Condition is issued when MALE = 1

7.7.9.2 Function to Detect Loss of Arbitration during NACK Transmission (NALE Bit)

This function causes arbitration to be lost if the internal SDA output level does not match the level on the SDAn line during transmission of NACK in receive mode. Arbitration is lost because of a conflict between NACK and ACK transmissions when two or more master devices receive data from the same slave device simultaneously in a multi-master system. Such conflict occurs when multiple master devices send or receive the same information through a single slave device. **Figure 7.7-35** shows an example of arbitration-lost detection during the transmission of NACK.

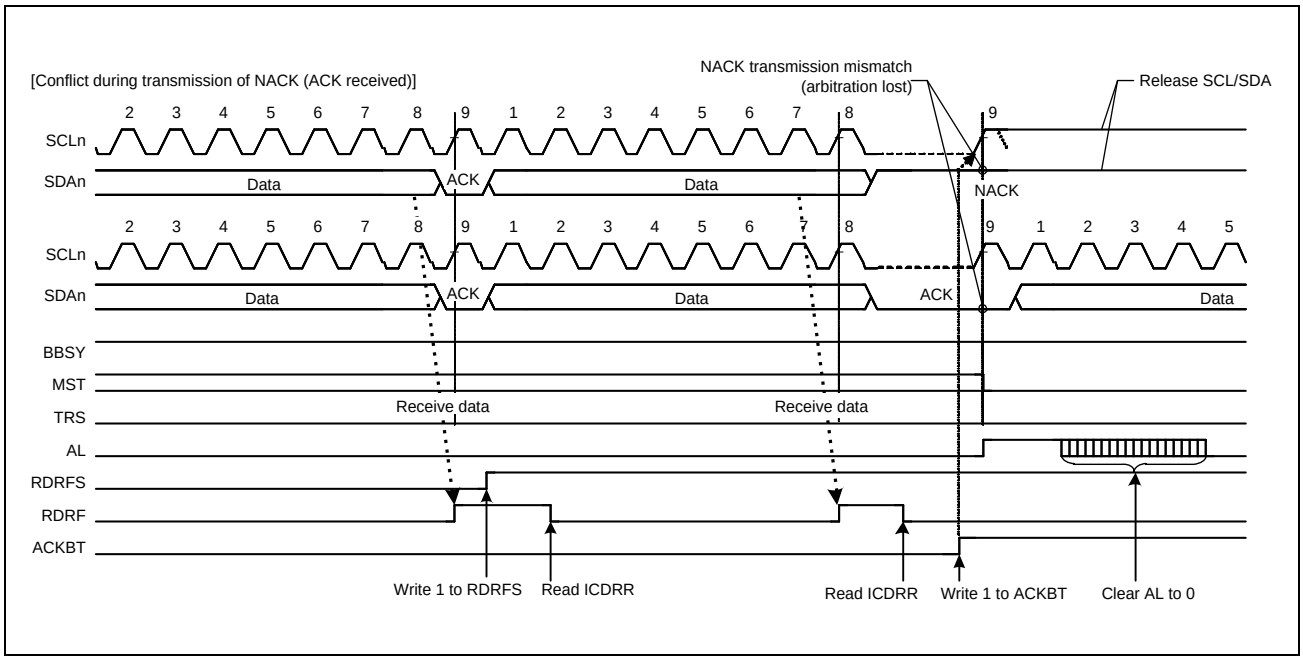


Figure 7.7-35 Example of Arbitration-Lost Detection during Transmission of NACK when NALE = 1

The following description explains arbitration-lost detection using an example in which two master devices (master A and master B) and a single slave device are connected through the bus. In this example, master A receives 2 bytes of data from the slave device, and master B receives 4 bytes of data from the slave device.

If master A and master B access the slave device simultaneously, because the slave address is identical, arbitration is not lost in either master A or B during access to the slave device. Therefore, both master A and master B recognize that they have obtained the bus mastership and operate as such. Master A sends NACK when it has received the 2 final bytes of data from the slave device. Meanwhile, master B sends ACK because it has not received the required 4 bytes of data.

The NACK transmission from master A and the ACK transmission from master B conflict. In general, if a conflict like this occurs, master A cannot detect the ACK transmitted by master B and issues a stop condition. The stop condition issue conflicts with the SCL clock output of master B, which disrupts communication.

When RIIC receives ACK during transmission of NACK, it detects a defeat in conflict with other master devices and causes arbitration to be lost. If arbitration is lost during transmission of NACK, RIIC immediately cancels the slave match condition and enters slave receive mode. This prevents a stop condition from being issued, preventing a communication failure on the bus.

Similarly, in the ARP command processing of SMBus, the function to detect loss of arbitration during transmission of NACK is also available for eliminating the extra clock cycle processing, such as FFh transmission processing, which is required if the UDID (Unique Device Identifier) of the assigned address does not match in the Get UDID general processing after the Assign Address command.

The RIIC detects arbitration-lost during transmission of NACK when the following condition is met with the NALE bit in ICFER set to 1b (arbitration-lost detection during NACK transmission enabled).

[Condition for arbitration-lost during NACK transmission]

- When the internal SDA output level does not match the SDAn line (ACK is received) during transmission of NACK (ACKBT bit = 1b in ICMR3).

7.7.9.3 Slave Arbitration-Lost Detection (SALE Bit)

This function causes arbitration to be lost if the transmit data (internal SDA output level) and the level on the SDAn line do not match (high output as the internal SDA output, meaning the SDAn pin is in the high-impedance state), and the low level is detected on the SDAn line in slave transmit mode. This arbitration-lost detection function is mainly used when transmitting a UDID (Unique Device Identifier) over an SMBus.

When RIIC loses slave arbitration, RIIC is immediately released from the slave-matched state and enters slave receive mode.

This function can detect conflicts of data during transmission of UDIDs over an SMBus and eliminates subsequent redundant processing for the transmission of FFh.

IIC detects slave arbitration-lost when the following condition is met with the SALE bit in ICFER set to 1b (slave arbitration- lost detection enabled).

[Condition for slave arbitration-lost]

- When transmit data excluding acknowledge (internal SDA output level) does not match the SDAn line in slave transmit mode (MST and TRS bits = 01b in ICCR2).

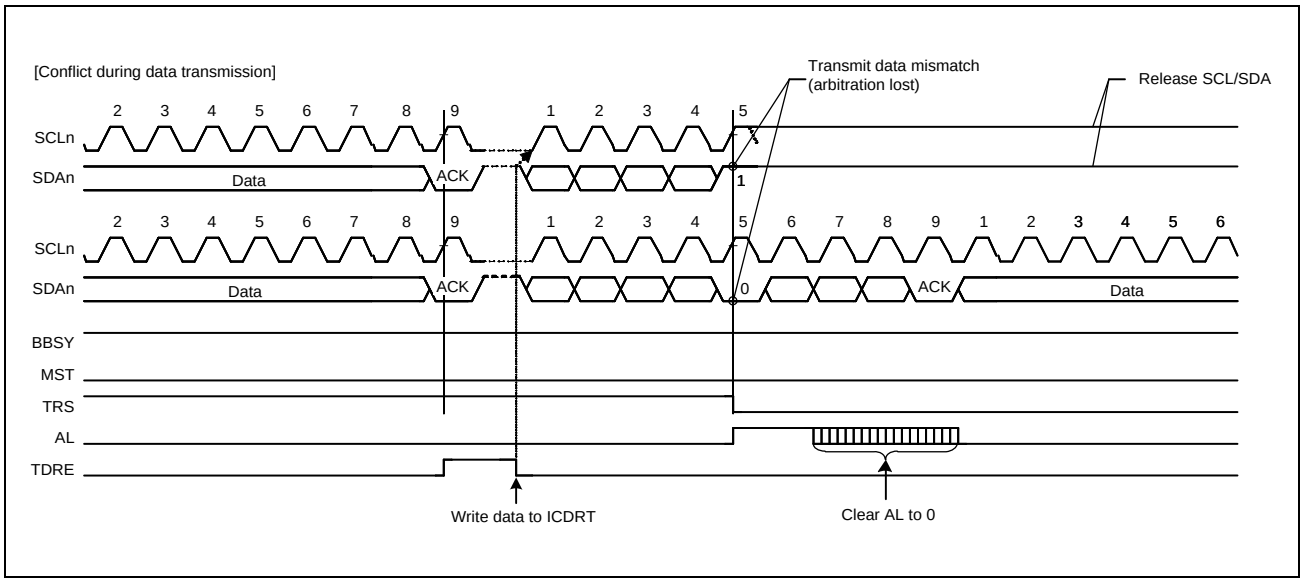


Figure 7.7-36 Example of Slave Arbitration-Lost Detection when SALE = 1

7.7.10 Start, Restart, and Stop Condition Issuing Function

7.7.10.1 Issuing a Start Condition

IIC issues a start condition when the ST bit in ICCR2 is set to 1b. When the ST bit is set to 1b, a start condition request is made, and RIIC issues a start condition when the BBSY flag in ICCR2 is 0b (bus free state). When a start condition is issued normally, RIIC automatically shifts to the master transmit mode.

To issue a start condition:

1. Drive the SDAn line low (high level to low level).
2. Ensure that the time set in ICBRH and the start condition hold time elapse.
3. Drive the SCLn line low (high level to low level).
4. Detect low level on the SCLn line and ensure the low-level period of the SCLn line set in ICBRL elapses.

7.7.10.2 Issuing a Restart Condition

IIC issues a restart condition when the RS bit in ICCR2 is set to 1.

When the RS bit is set to 1b, a restart condition request is made, and RIIC issues a restart condition when the BBSY flag in ICCR2 is 1b (bus busy state) and the MST bit in ICCR2 is 1b (master mode).

To issue a restart condition:

1. Release the SDAn line.
2. Ensure that the low-level period of the SCLn line set in ICBRL elapses.
3. Release the SCLn line (low level to high level).
4. Detect a high level on the SCLn line and ensure the time set in ICBRL and the restart condition setup time elapse.
5. Drive the SDAn line low (high level to low level).
6. Ensure the time set in ICBRH and the restart condition hold time elapse.
7. Drive the SCLn line low (high level to low level).
8. Detect a low level on the SCLn line and ensure the low-level period of the SCLn line set in ICBRL elapses.

Note: When issuing restart condition requests, write the slave address to ICDRT after confirming that ICCR2.RS = 0b. Data written while ICCR2.RS = 1b is not forwarded because of the retransmission condition before the occurrence.

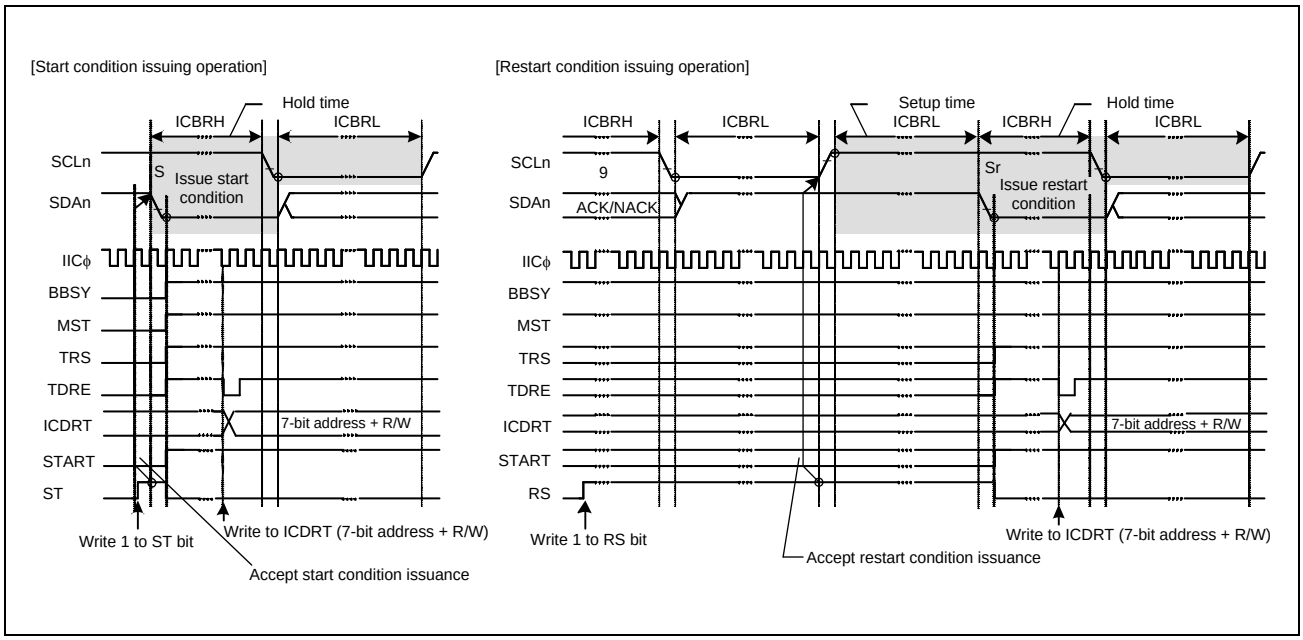


Figure 7.7-37 Start and Restart Condition Issue Timing using the ST and RS Bits

Figure 7.7-38 shows the operation timing when a restart condition is issued after the master transmission.

To issue a restart condition after the master transmission:

1. Initialize the RIIC using the details provided in **7.7.3.2 Initial Settings**.
2. Read the BBSY flag in IICR2 to check that the bus is free, and then set the ST bit in ICCR2 to 1b (start condition issuance request). On receiving the request, the RIIC issues a start condition. At the same time, the BBSY flag and the START flag in ICSR2 are automatically set to 1b and the ST bit is automatically set to 0b. If the start condition is detected and the internal levels for the SDA output state and the levels on the SDAn line match while the ST bit is 1b, the RIIC recognizes that a start condition is successfully issued as requested by the ST bit. The MST and TRS bits in ICCR2 are automatically set to 1b, placing the RIIC in master transmit mode. The TDRE flag in ICSR2 is also automatically set to 1b when the TRS bit is set to 1b.
3. Check that the TDRE flag in ICSR2 is 1b, and then write the value for transmission (the slave address and the R/W# bit) to ICDRT. After the transmit data is written to ICDRT, the TDRE flag is automatically set to 0b, the data is transferred from ICDRT to ICDRS, and the TDRE flag is again set to 1b. After the byte containing the slave address and R/W# bit has been transmitted, the value of the TRS bit is automatically updated to select master transmit or master receive mode according to the value of the transmitted R/W# bit. If the value of the R/W# bit is 0b, the RIIC continues in master transmit mode. If the ICSR2.NACKF flag is 1b at this time, indicating that no slave device recognized the address or that there was an error in communications, write 1b to the ICCR2.SP bit to issue a stop condition. To transmit data with an address in the 10-bit format, start by writing 11110b, the 2 upper bits of the slave address, and W to ICDRT as the first address transmission. Then, as the second address transmission, write the 8 lower bits of the slave address to ICDRT.
4. After confirming that the TDRE flag in ICSR2 is 1b, write the data for transmission to the ICDRT register. The RIIC automatically holds the SCLn line low until the data for transmission is ready, and a restart or stop condition is issued.
5. After all bytes of data for transmission are written to the ICDRT register, wait until the value of the TEND flag in ICSR2 returns to 1b. Then, after checking that the START flag in ICSR2 is 1b, set the START flag in ICSR2 to 0b.

6. Set the RS bit in ICCR2 to 1b (restart condition issuance request). Upon receiving the request, the RIIC issues a restart condition.
7. After checking that the START flag in ICSR2 is 1b, write the value for transmission (the slave address and the R/W# bit) to ICDRT.

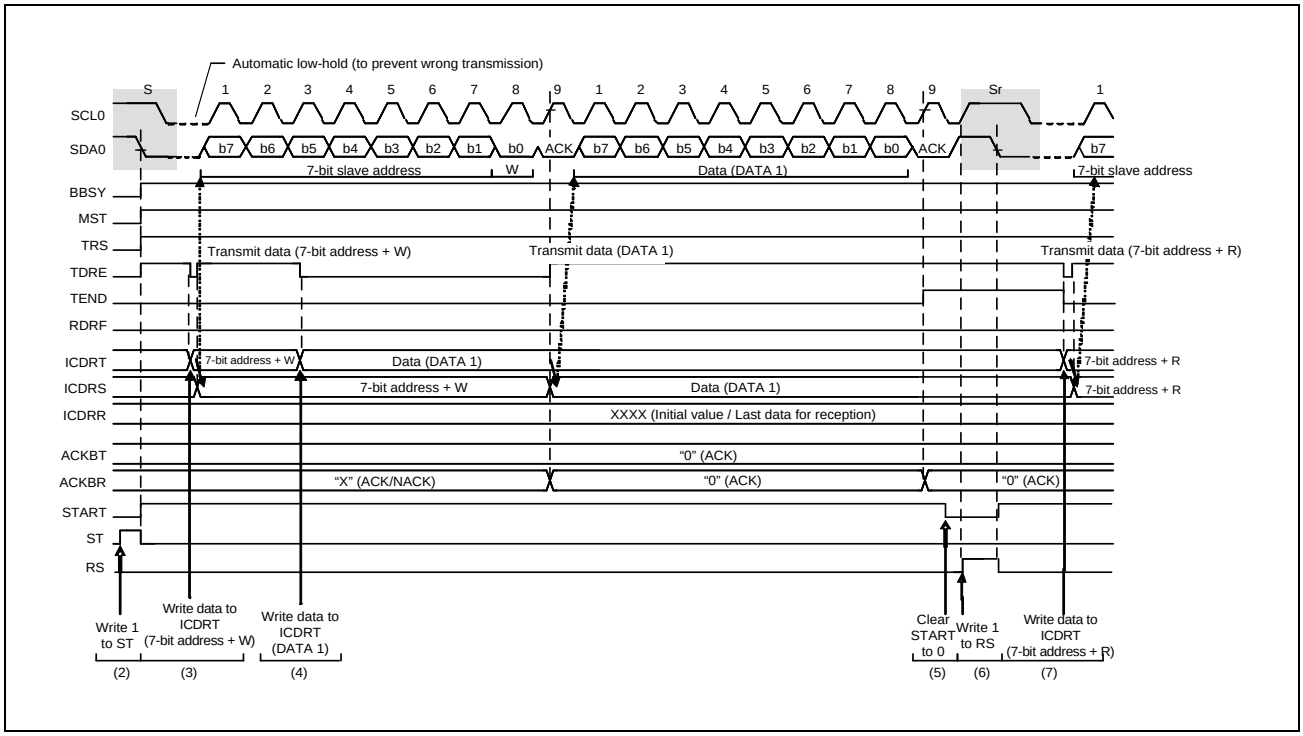


Figure 7.7-38 Restart Condition Issue Timing after Master Transmission

7.7.10.3 Issuing a Stop Condition

IIC issues a stop condition when the SP bit in ICCR2 is set to 1. When the SP bit is set to 1b, a stop condition request is made, and the RIIC issues a stop condition when the BBSY flag in ICCR2 is 1 (bus busy state) and the MST bit in ICCR2 is 1b (master mode).

To issue a stop condition:

1. Drive the SDAn line low (high level to low level).
2. Ensure that the low-level period of the SCLn line set in ICBRL elapses.
3. Release the SCLn line (low level to high level).
4. Detect a high level on the SCLn line and ensure that the time set in ICBRH and the stop condition setup time elapse.
5. Release the SDAn line (low level to high level).
6. Ensure the time set in ICBRL and the bus free time elapse.
7. Clear the BBSY flag to 0b to release the bus mastership.

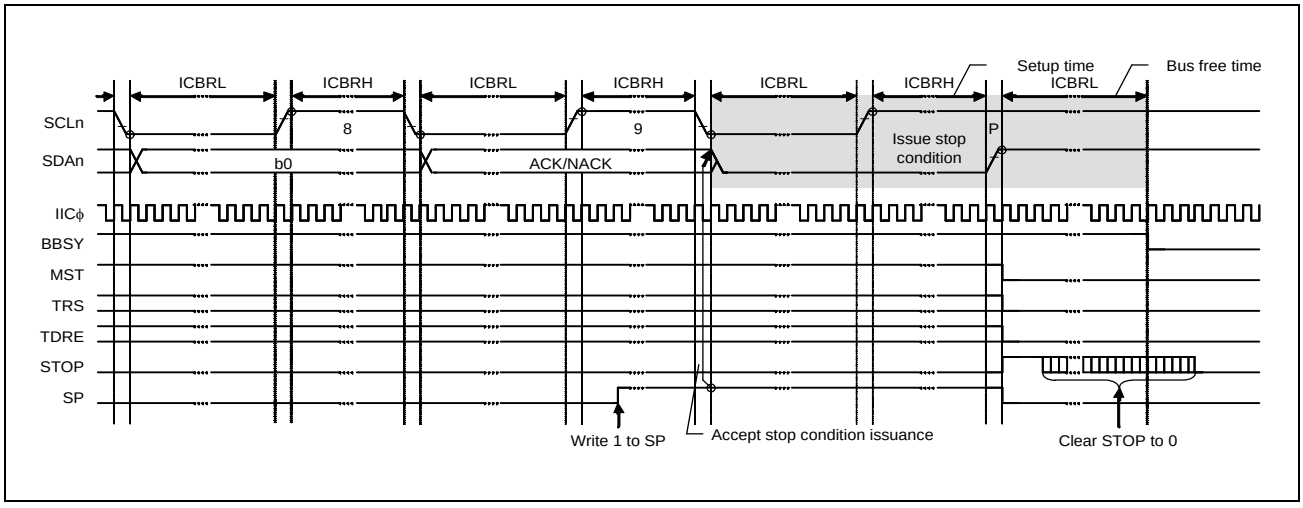


Figure 7.7-39 Stop Condition Issue Timing using the SP Bit

7.7.11 Bus Hanging

If the clock signals from the master and slave devices are out of synchronization because of noise or other factors, the I^2C bus might hang with a fixed level on the SCLn line or SDAn line.

To manage bus hanging, the RIIC has the following:

- A timeout function to detect hanging by monitoring the SCLn line
- A function for the output of an extra SCL clock cycle to release the bus from a hung state because of clock signals being out of synchronization
- An RIIC reset function
- An internal reset function

By checking the SCLO, SDAO, SCLI, and SDAI bits in ICCR1, it is possible to see whether the RIIC or its communicating partner is placing the SCLn or SDAn lines at the low level.

7.7.11.1 Timeout Function

The timeout function can detect when the SCLn line is stuck longer than the predetermined time. The RIIC can detect an abnormal bus state by monitoring that the SCLn line is stuck low or high for a predetermined time.

The timeout function monitors the SCLn line state and counts the low-level or high-level period using the internal counter.

The timeout function resets the internal counter each time the SCLn line changes (rises or falls), but continues to count unless the SCLn line changes. If the internal counter overflows because no SCLn line changes, the RIIC can detect the timeout and report the bus hung state.

This timeout function is enabled when the ICFER.TMOE bit is 1b. It detects a hung state when the SCLn line is stuck low or high during the following conditions:

- The bus is busy (ICCR2.BBSY flag is 1b) in master mode (ICCR2.MST bit is 1b)
- IIC slave address is detected (ICSR1 register is not 00h) and the bus is busy (ICCR2.BBSY flag is 1) in slave mode (ICCR2.MST bit is 0b)
- The bus is free (ICCR2.BBSY flag is 0b) while a start condition is requested (ICCR2.ST bit is 1b)

The internal counter of the timeout function uses the internal reference clock (IICΦ) set in the CKS[2:0] bits in ICMR1 as a count source. It functions as a 16-bit counter when long mode is selected (TMOS bit = 0b in ICMR2) or a 14-bit counter when short mode is selected (TMOS bit = 1b).

The SCLn line level (low, high, or both levels) during which this counter is activated can be selected in the TMOH and TMOL bits in ICMR2. If both TMOL and TMOH bits are set to 0b, the internal counter is disabled.

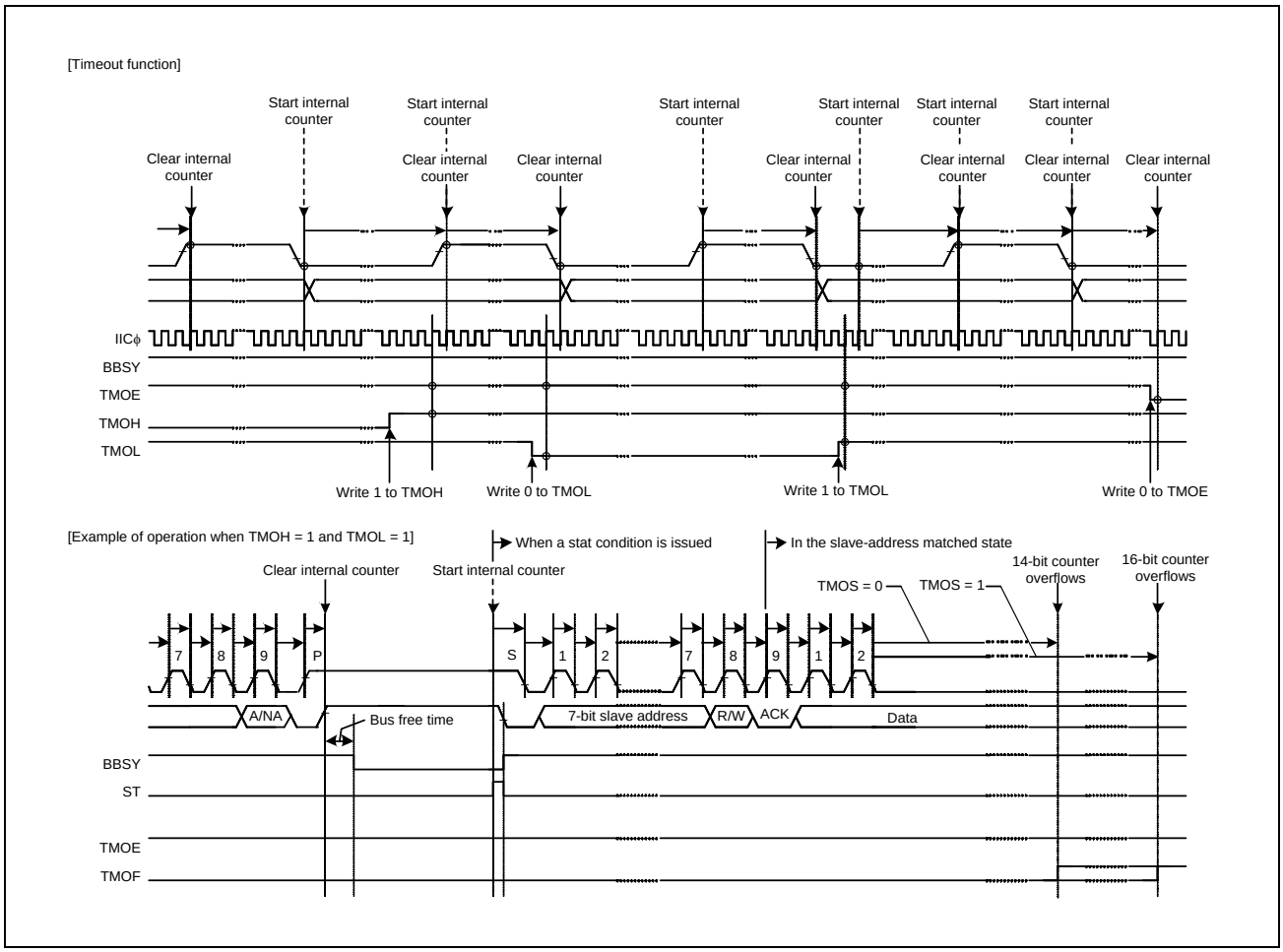


Figure 7.7-40 Timeout Function using the TMOE, TMOS, TMOH, and TMOL Bits

7.7.11.2 Extra SCL Clock Cycle Output Function

In master mode, this function outputs extra SCL clock cycles from the RIIC to release the SDAn line of the slave device from being held at the low level because the master is out of synchronization with the slave device.

It uses single cycles of the SCL clock for a bus error where the RIIC cannot issue a stop condition because the slave device is holding the SDAn line at the low level. Do not use this function in normal situations.

Using it when communications are proceeding correctly leads to malfunctioning.

When the CLO bit in ICCR1 is set to 1b in master mode, an additional clock pulse at the frequency set by the CKS[2:0] bits in ICMR1 and the ICBRH and ICBRL registers is output from the SCL0 pin. After output of this clock pulse, the CLO bit automatically becomes 0b. The SCL0 pin is held low when the BBSY bit in ICCR2 flag is 1b and held high when the BBSY flag is 0b. Consecutive additional clock pulses can be output by writing 1b to the CLO bit after confirming the CLO bit to be 0b.

When the RIIC module is in master mode and the slave device is holding the SDAn line at the low level because synchronization with the slave device is lost because of effects like noise, the output of a stop condition is not possible.

This function can be used to output extra cycles of SCL one by one to make the slave device release the SDAn line from being held at the low level, and recover the bus from an unusable state. Release of the SDAn line by the slave device can be monitored by reading the SDAI bit in ICCR1. After confirming the release of the SDAn line by the slave device, complete the communication by reissuing the stop condition.

Use this function with the MALE bit in ICFER set to 0b (master arbitration-lost detection disabled).

[Output conditions for using the CLO bit in ICCR1]

- When the bus is free (BBSY flag in ICCR2 = 0b) or in master mode (MST bit = 1b and BBSY flag = 1b in ICCR2)
- When the communication device does not hold the SCLn line low.

Figure 7.7-41 shows the operation timing of the extra SCL clock cycle output function (CLO bit).

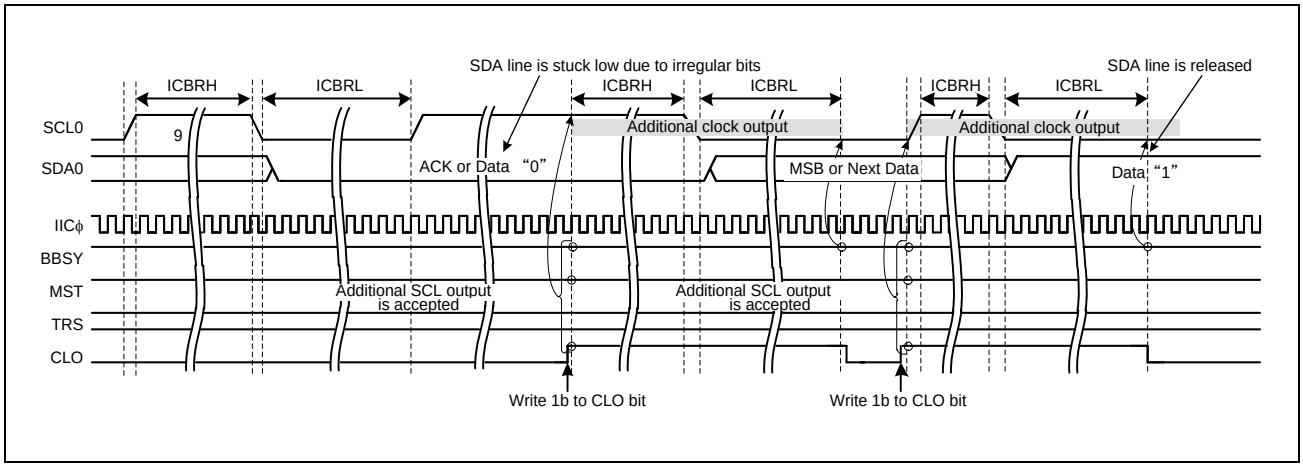


Figure 7.7-41 Extra SCL Clock Cycle Output Function using the CLO Bit

7.7.11.3 IIC Reset and Internal Reset

The RIIC module has two types of resets:

- IIC reset, which initializes all registers, including the BBSY flag in ICCR2
- Internal reset, which releases the RIIC from the slave-address matched state and initializes the internal counter while saving other settings.

After issuing a reset, always set the IICRST bit in ICCR1 to 0b.

Both types of resets are valid for release from bus-hung states, because both restore the output state of the SCLn and SDAn pins to the high-impedance state.

Issuing a reset during slave operation might lead to a loss of synchronization between the master device clock and the slave device clock, so avoid this when possible. In addition, monitoring of the bus state, such as for the presence of a start condition, is not possible during an RIIC reset (ICE and IICRST bits = 01b in ICCR1).

For a detailed description of the RIIC and internal resets, see **7.7.15 Resets, Registers, and Function States when Issuing Each Condition**.

7.7.12 SMBus Operation

IIC supports data communication conforming to the SMBus Specification, version 2.0. To perform SMBus communication, set the SMBS bit in ICMR3 to 1b. To use the transfer rate within a range of 10 kbps to 100 kbps of the SMBus standard, set the CKS[2:0] bits in ICMR1, ICBRH, and ICBRL. In addition, specify the values in the DLCS bit in ICMR2 and the SDDL[2:0] bits in ICMR2 to meet the data hold time specification of 300 ns or more. When the RIIC is used only as a slave device, the transfer rate setting is not required, but ICBRL must be set to a value longer than the data setup time (250 ns).

For the SMBus device default address (1100_001b), use one of the slave address registers L0 to L2 (SARL0, SARL1, and SARL2), and set the associated FS bit (7- or 10-bit address format select) in SARUy (y = 0 to 2) to 0 (7-bit address format).

When transmitting the UDID (Unique Device Identifier), set the SALE bit in ICFER to 1b to enable the slave arbitration-lost detection function.

7.7.12.1 SMBus Timeout Measurement

7.7.12.1.1 Measuring slave device timeout

The following period (timeout interval: $T_{LOW:SEXT}$) must be measured for slave devices in SMBus communication:

- From start condition to stop condition

To measure timeout for slave devices, measure the period from start condition detection to stop condition detection with the internal timer using the RIIC start condition detection interrupt (STIn) and stop condition detection interrupt (SPIn). The measured timeout period must be within the total clock low-level period [slave device] $T_{LOW:SEXT}$: 25 ms (maximum) of the SMBus standard.

If the time measured with the internal timer exceeds the clock low-level detection timeout $T_{TIMEOUT}$: 25 ms (minimum) of the SMBus standard, the slave device must release the bus by writing 1b to the IICRST bit in ICCR1 to issue an internal reset of the RIIC. When an internal reset is issued, the RIIC stops driving the bus for the SCLn and SDAn pins and makes the SCLn/ SDAn pin output high-impedance, which releases the bus.

7.7.12.1.2 Measuring master device timeout

The following periods (timeout interval: $T_{LOW:MEXT}$) must be measured for master devices in SMBus communication:

- From start condition to acknowledge bit
- Between acknowledge bits
- From acknowledge bit to stop condition

To measure timeout for master devices, measure these periods with the internal timer using the RIIC start condition detection interrupt (STIn), stop condition detection interrupt (SPIn), transmit end interrupt (IICn_TEI), or receive data full interrupt (IICn_RXI). The measured timeout period must be within the total clock low-level extended period (master device) $T_{LOW:MEXT}$: 10 ms (maximum) of the SMBus standard, and the total of all $T_{LOW:MEXT}$ values from the start condition to stop condition must be within $T_{LOW:SEXT}$: 25 ms (maximum).

For the ACK receive timing (rising edge of the 9th SCL clock cycle), monitor the TEND flag in ICSR2 in master transmit mode (master transmitter) and the RDRF flag in ICSR2 in master receive mode (master receiver). Perform byte-wise transmit operations in master transmit mode, and hold the RDRFS bit in ICMR3 at 0b until the byte

immediately before the reception of the final byte in master receive mode. While the RDRFS bit is 0b, the RDRF flag is set to 1b on the rising edge of the 9th SCL clock cycle.

If the period measured with internal timer exceeds the total clock low-level extended period (master device), $T_{LOW:MEXT}$: 10 ms (maximum) of the SMBus standard or the total of measured periods exceeds the clock low-level detection timeout, $T_{TIMEOUT}$: 25 ms (minimum) of the SMBus standard, the master device must stop the transaction by issuing a stop condition. In master transmit mode, immediately stop the transmit operation (stop writing data to ICDRT).

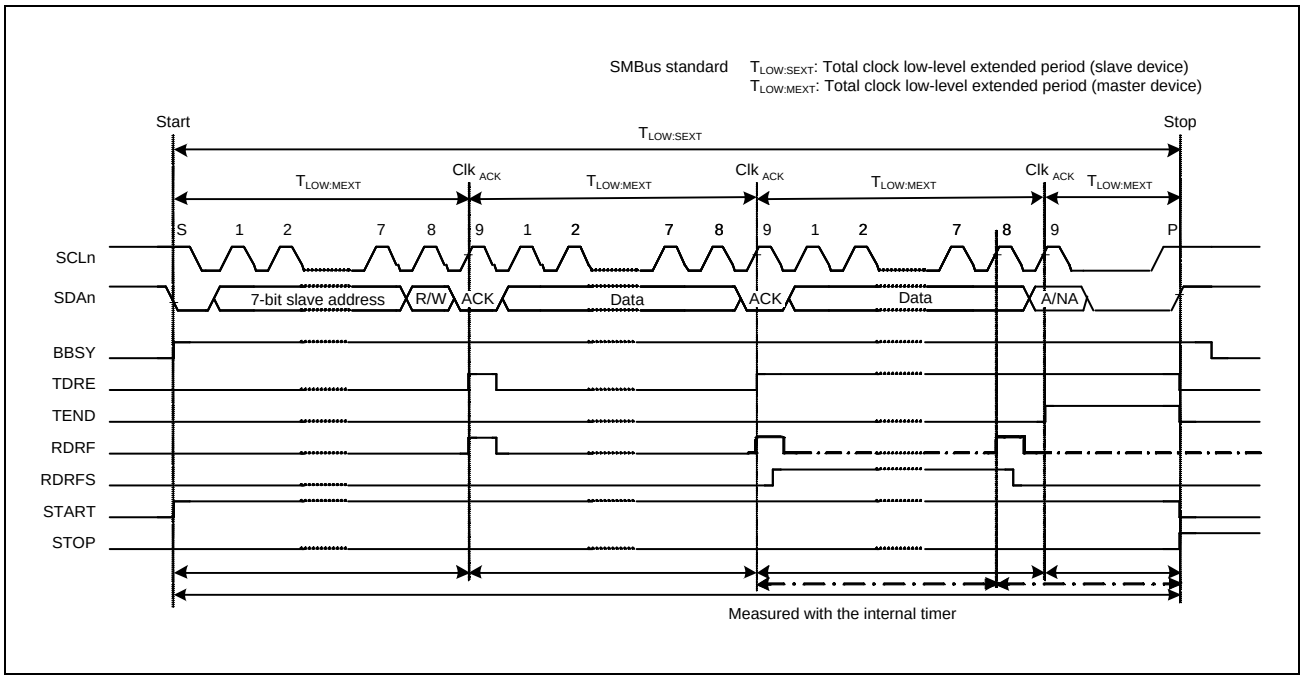


Figure 7.7-42 SMBus Timeout Measurement

7.7.12.2 Packet Error Code (PEC)

The LSI provides a CRC calculator, which enables transmission of a packet error code (PEC) or allows checking the received data in SMBus data communication for the RIIC. For the CRC-generating polynomials of the CRC calculator, see **7.6 CRC Operation Unit (CRC)**.

In master transmit mode, the PEC data can be generated by writing all transmit data to the CRC Data Input register (CRCDIR) in the CRC calculator.

In master receive mode, the PEC data can be checked by writing all receive data to CRCDIR in the CRC calculator and comparing the obtained value in the CRC Data Output Register (CRCDOR) with the received PEC data.

To send ACK or NACK based on the match or mismatch result when the final byte is received as a result of the PEC code check, set the RDRFS bit in ICMR3 to 1b before the rising edge of the 8th SCL clock cycle during reception of the final byte, and hold the SCLn line low on the falling edge of the 8th clock cycle.

7.7.12.3 SMBus Host Notification Protocol (Notify ARP Master Command)

In communications over an SMBus, a slave device can temporarily act as a master device to notify the SMBus host (or ARP master) of its own slave address or to request its own slave address from the SMBus host.

For a product using the LSI to operate as an SMBus host or ARP master, the host address (0001 000b) sent from the slave device must be detected as a slave address, and so the RIIC has a function for detecting the host address. To detect the host address as a slave address, set the SMBS bit in ICMR3 and the HOAE bit in ICSER to 1b. Operation after the host address is detected is the same as normal slave operation.

7.7.13 Interrupt Sources

The RIIC issues 9types of interrupt request :transmit end, receive data full, transmit data empty, arbitration-lost, NACK detection, timeout detection, start condition detection, stop condition detection and slave address match during wake up function.

Table 7.7-10 lists details about the interrupt requests. The receive data full and transmit data empty interrupts can activate data transfer by the DMAC.

Table 7.7-10 Interrupt Sources

Symbol	Interrupt Source	Interrupt Flag	DMAC Activation	Interrupt Condition
RIIC_CHm_ali_n*4	Arbitration lost	AL	Not possible	AL = 1b, ALIE = 1b
RIIC_CHm_naki_n*4	NACK detection	NACKF	Not possible	NACKF = 1b, NAKIE = 1b
RIIC_CHm_tmoi_n*4	Timeout detection	TMOF	Not possible	TMOF = 1b, TMOIE = 1b
RIIC_CHm_sti_n*4	Start condition detection	START	Not possible	START = 1b, STIE = 1b
RIIC_CHm_spi_n*4	Stop condition detection	STOP	Not possible	STOP = 1b, SPIE = 1b
RIIC_CHm_ri_n*2,*4	Receive data full	RDRF	Possible	RDRF = 1b, RIE = 1b
RIIC_CHm_ti_n*1,*4	Transmit data empty	TDRE	Possible	TDRE = 1b, TIE = 1b
RIIC_CHm_tei_n*3,*4	Transmit end	TEND	Not possible	TEND = 1b, TEIE = 1b

Note: There is a delay between the execution of a write instruction for a peripheral module by the CPU and actual writing to the module. When an interrupt flag is cleared or masked, read the relevant flag again to check whether clearing or masking is complete, and then return from interrupt handling. Not doing so creates the possibility of repeated processing of the same interrupt.

Note 1. Because IICn_TXI is an edge-detected interrupt, it does not require clearing. Additionally, the TDRE flag in ICSR2 (condition for IICn_TXI) is automatically set to 0b when transmit data is written to ICDRT or a stop condition is detected (STOP flag = 1b in ICSR2).

Note 2. Because IICn_RXI is an edge-detected interrupt, it does not require clearing. Additionally, the RDRF flag in ICSR2 (condition for IICn_RXI) automatically is set to 0b when data is read from ICDRR.

Note 3. When using the IICn_TEI interrupt, clear the TEND flag in ICSR2 in the IICn_TEI interrupt handling. The TEND flag in ICSR2 automatically is set to 0b when transmit data is written to ICDRT or a stop condition is detected (STOP flag = 1b in ICSR2).

Note 4. Channel number (m = 0 to 8).

Clear or mask each flag during interrupt handling.

7.7.14 Even Link Output

The RIIC handles event output for the event link controller (ELC) corresponding to the following sources.

■ **Transfer error event**

When a transfer error event occurs, the corresponding event signal can be output for another module via the ELC.

■ **Receive data full**

When a receive data register becomes full, the corresponding event signal can be output for another module via the ELC.

■ **Transmit data empty**

When a transmit data register becomes empty, the corresponding event signal can be output for another module via the ELC.

■ **Transmit end**

On completion of transfer, the corresponding event signal can be output for another module via the ELC.

Table 7.7-2 Event Signal Output to the ELC

Name	Sources
RIIC_CHm_iic_elcerrp	Transfer error or event generation
RIIC_CHm_iic_elcrdrfp	Receive data full
RIIC_CHm_iic_elctdrep	Transmit data empty
RIIC_CHm_iic_elctendp	Transmit end

Note: m = 0 to 8

7.7.15 Resets, Registers, and Function States when Issuing Each Condition

The RIIC provides chip reset, RIIC reset, and internal reset functions. **Table 7.7-11** lists the RIIC resets, registers, and function states when issuing each condition.

Table 7.7-11 Resets, Registers, and Function States when Issuing Each Condition

Registers		Chip Reset	RIIC Reset (ICE = 0b, IICRST = 1b)	Internal Reset (ICE = 1b, IICRST = 1b)	Start or Restart Condition Detection	Stop Condition Detection
ICCR1	ICE, IICRST	Reset	Saved	Saved	Saved	Saved
	SCLO, SDAO		Reset	Reset		
	Others			Saved		
ICCR2	BBSY	Reset	Reset	Saved	*1	Reset
	ST,RS			Reset	Reset	Saved
	TRS, MST				*1	Reset
	SP				Reset	Reset
ICMR1	BC[2:0]	Reset	Reset	Reset	Reset	Saved
	Others			Saved	Saved	
ICMR2		Reset	Reset	Saved	Saved	Saved
ICMR3		Reset	Reset	Saved	Saved	Saved
ICFER		Reset	Reset	Saved	Saved	Saved
ICSER		Reset	Reset	Saved	Saved	Saved
ICIER		Reset	Reset	Saved	Saved	Saved
ICSR1		Reset	Reset	Reset	Saved	Reset
ICSR2	TDRE	Reset	Reset	Reset	*1	Reset
	TEND				Saved	
	START				Set	Reset
	STOP				Saved	Set
	Others				Saved	Saved
SARL0, SARL1, SARL2, SARU0, SARU1, SARU2		Reset	Reset	Saved	Saved	Saved
ICBRH, ICBRL		Reset	Reset	Saved	Saved	Saved
ICDRT		Reset	Reset	Saved	Saved	Saved
ICDRR		Reset	Reset	Saved	Saved	Saved
ICDRS		Reset	Reset	Reset	Saved	Saved
Timeout function		Reset	Reset	Operating	Operating	Operating
Bus free time measurement		Reset	Reset	Operating	Operating	Operating

Note 1. This bit is not reset. This bit becomes 0 or 1 in accordance with the conditions.

SECTION 7 LOW-SPEED INTERFACE

7.8 I3C Bus Interface (I3C)

7.8.1 Overview

7.8.1.1 Functional Overview

The I3C bus interface (I3C) has 1 channel. The I3C module conform with and provide a subset of the NXP I^2C (Inter-Integrated Circuit) bus interface functions and a subset of the MIPI I3C.

Table 7.8-1 lists the I^2C specifications, and **Table 7.8-2** lists the I3C specifications.

Table 7.8-1 I^2C Specifications (1/2)

Item	Description
Operation mode	Master mode and slave mode selectable
Data handler	Single buffer transfer
Communication protocol	• I^2C bus format – Standard-mode (Sm): 0 to 100 kbps – Fast-mode (Fm): 0 to 400 kbps – Fast-mode Plus (Fm+): 0 to 1 Mbps – High-speed mode (Hs-mode): 0 to 3.4 Mbps • SMBus format: 10 to 100 kbps
Address format	• 7-bit address • 10-bit address
Address detection	• Slave address (static address) (max 3 addresses) • General call address • Hs-mode master code • Device ID • Host address • 10-bit slave addressing
Clock stretching	Clock stretching capability
Noise-filter	• Digital noise-filter
Interrupt source	• Receive data buffer full • Transmit data buffer empty • START condition detection • STOP condition detection • Transmit end • NACK detection • Arbitration lost • Timeout detection • Wake-up condition detection
Error detection	• NACK received • Arbitration lost error • Timeout error

Table 7.8-1 I^2C Specifications (2/2)

Item	Description
Event link output	• Communication event • Receive data buffer full event • Transmit data buffer empty event • Transmit end event
Wake-up source	• Address detection of slave address

Table 7.8-2 I3C Specifications (1/2)

Item	Description
Operation mode	Master (main master/secondary master) mode and slave mode selectable
Data handler	• Master: – Normal FIFO buffer transfer • Slave: – Normal FIFO buffer transfer
Communication protocol	• SDR (I3C single data rate) mode – Private message – Broadcast message (common command code) – Direct message (common command code) • Legacy I^2C message – Fast-mode (Fm): 0 to 400 kbps – Fast-mode Plus (Fm+): 0 to 1 Mbps
In-band interrupt	• Slave interrupt request • Master ship request (secondary master only) • Hot-join event
Address format	7-bit address
Address detection	• Slave address (static address or dynamic address) • Broadcast address (7Eh)
Clock stalling	Clock stalling capability
Timing Control	• Synchronous Timing Control – Sync Mode: Synchronous Basic Mode • Asynchronous Timing Control – Async Mode 0: Asynchronous Basic Mode – Async Mode 1: Asynchronous Advanced Mode
Interrupt source	• Non-recoverable internal error • Transfer error • Transfer abort • Response queue full • Command queue empty • IBI status queue full • Receive data buffer full • Transmit data buffer empty • Receive status queue full • START condition detection • STOP condition detection • Timeout detection • Wake-up condition detection

Table 7.8-2 I3C Specifications (2/2)

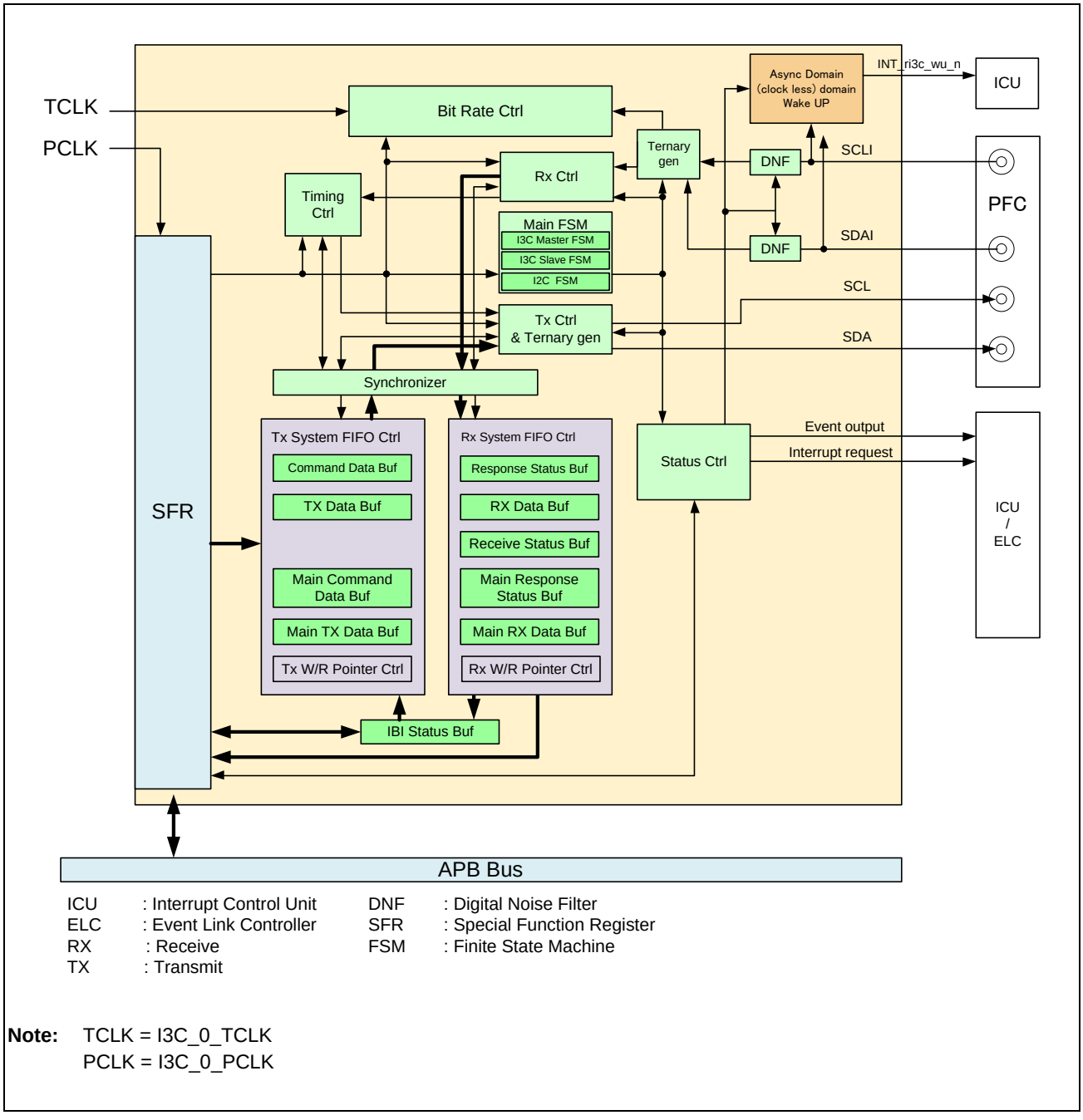
Item	Description
Error detection	• Non-recoverable internal error • CRC error • Parity error • Frame error • Address header error • Address NACKed or dynamic address assignment NACKed • Receive overflow or transfer underflow error • Aborted • NACK received for the I^2C write data transfer • Timeout error
Event link output	• Communication event • Response buffer full event • Command buffer empty event • IBI Status buffer full event • Receive data buffer full event • Transmit data buffer empty event • Receive status buffer full event • Synchronous timing event • MREF counter overflow event • MREF capture event • Additional master-initiated bus event
Wake-up source	• Master: – SDA30 assertion of IBI (START condition detection) • Slave: – Address detection of broadcast address (0x7E) and slave address

Table 7.8-3 I3C I/O Pins

Function	Pin name	Input/ Output	Description
I3C	SCL30	I/O	Input/output pins for clock
	SDA30	I/O	Input/output pins for data

7.8.2 Block Diagram [I²C/I3C common]

Figure 7.8-1 shows the main components of this I3C.



Note: TCLK = I3C_0_TCLK
PCLK = I3C_0_PCLK

Figure 7.8-1 I3C Block Diagram

7.8.3 Registers

This section describes details of the register functions.

Table 7.8-4 shows the base address of I3C.

Table 7.8-4 Register Base Address

Base Register Name	The Base Address
<I3C_base>	0_1240_0000h (5240_0000h*1, 4240_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

7.8.3.1 List of Registers

I3C registers are listed in the following table. (m = 0)

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Protocol Selection Register	I3Cm_PRTS	0000_0001h	0000h	32
Reserve	-	-	0004h to 0013h	-
Bus Control Register	I3Cm_BCTL	0000_0000h	0014h	32
Master Device Address Register	I3Cm_MSDVAD	0000_0000h	0018h	32
Reserve	-	-	001Ch to 001Fh	-
Reset Control Register	I3Cm_RSTCTL	0000_0000h	0020h	32
Present State Register	I3Cm_PRSST	0000_0000h	0024h	32
Reserve	-	-	0028h to 002Fh	-
Internal Status Register	I3Cm_INST	0000_0000h	0030h	32
Internal Status Enable Register	I3Cm_INSTE	0000_0000h	0034h	32
Internal Interrupt Enable Register	I3Cm_INIE	0000_0000h	0038h	32
Internal Status Force Register	I3Cm_INSTFC	0000_0000h	003Ch	32
Reserve	-	-	0040h to 0043h	-
Device Characteristic Table Register	I3Cm_DVCT	0000_0000h	0044h	32
Reserve	-	-	0048h to 0057h	-
IBI Notify Control Register	I3Cm_IBINCTL	0000_0000h	0058h	32
Reserve	-	-	005Ch to 005Fh	-
Bus Function Control Register	I3Cm_BFCTL	0000_0101h	0060h	32
Slave Control Register	I3Cm_SVCTL	0000_0000h	0064h	32
Reserve	-	-	0068h to 006Fh	-
Reference Clock Control Register	I3Cm_REFCKCTL	0000_0000h	0070h	32
Standard Bit Rate Register	I3Cm_STDBR	3F3F_FFFFh	0074h	32
Extended Bit Rate Register	I3Cm_EXTBR	3F3F_FFFFh	0078h	32
Bus Free Condition Detection Time Register	I3Cm_BFRECDT	0000_0000h	07Ch	32
Bus Available Condition Detection Time Register	I3Cm_BAVLCDT	0000_0000h	0080h	32
Bus Idle Condition Detection Time Register	I3Cm_BIDLCDT	0000_0000h	0084h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Output Control Register	I3Cm_OUTCTL	0000_0003h	0088h	32
Input Control Register	I3Cm_INCTL	0000_00D0h	008Ch	32
Timeout Control Register	I3Cm_TMOCTL	0000_0030h	0090h	32
Reserve	-	-	0094h to 0097h	-
Wake-Up Unit Control Register	I3Cm_WUCTL	0000_0041h	0098h	32
Reserve	-	-	009Ch to 009Fh	-
Acknowledge Control Register	I3Cm_ACKCTL	0000_0000h	00A0h	32
SCL Stretch Control Register	I3Cm_SCSTRCTL	0000_0000h	00A4h	32
Reserve	-	-	00A8h to 00AFh	-
SCL Stalling Control Register	I3Cm_SCSTLCTL	0000_0000h	00B0h	32
Reserve	-	-	00B4h to 00BFh	-
Slave Transfer Data Length Register	I3Cm_SVTDLG0	0000_0000h	00C0h	32
Reserve	-	-	00C4h to 011Fh	-
Synchronous Timing Control Register	I3Cm_STCTL	0000_0000h	0120h	32
Asynchronous Timing Control Register	I3Cm_ATCTL	0000_0000h	0124h	32
Asynchronous Timing Trigger Register	I3Cm_ATTRG	0000_0000h	0128h	32
Asynchronous Timing Control Counter Enable Register	I3Cm_ATCCNTE	0000_0000h	012Ch	32
Reserve	-	-	0130h to 013Fh	-
Condition Control Register	I3Cm_CNDCTL	0000_0000h	0140h	32
Reserve	-	-	0144h to 014Fh	-
Normal Command Queue Port Register	I3Cm_NCMDQP	0000_0000h	0150h	32
Normal Response Queue Port Register	I3Cm_NRSPQP	0000_0000h	0154h	32
Normal Transfer Data Buffer Port Register	I3Cm_NTDTBP0/NTDTBP0_BY	0000_0000h	0158h	32
Reserve	-	-	015Ch to 017Bh	-
Normal IBI Queue Port Register	I3Cm_NIBIQP	0000_0000h	017Ch	32
Normal Receive Status Queue Port Register	I3Cm_NRSQP	0000_0000h	0180h	32
Reserve	-	-	0184h to 018Fh	-
Normal Queue Threshold Control Register	I3Cm_NQTHCTL	0101_0101h	0190h	32
Normal Transfer Data Buffer Threshold Control Register	I3Cm_NTBTHCTL0	0101_0101h	0194h	32
Reserve	-	-	0198h to 01BFh	-
Normal Receive Status Queue Threshold Control Register	I3Cm_NRQTHCTL	0000_0001h	01C0h	32
Reserve	-	-	01C4h to 01CFh	-
Bus Status Register	I3Cm_BST	0000_0000h	01D0h	32
Bus Status Enable Register	I3Cm_BSTE	0000_0000h	01D4h	32
Bus Interrupt Enable Register	I3Cm_BIE	0000_0000h	01D8h	32
Bus Status Force Register	I3Cm_BSTFC	0000_0000h	01DCh	32
Normal Transfer Status Register	I3Cm_NTST	0000_0000h	01E0h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Normal Transfer Status Enable Register	I3Cm_NTSTE	0000_0000h	01E4h	32
Normal Transfer Interrupt Enable Register	I3Cm_NTIE	0000_0000h	01E8h	32
Normal Transfer Status Force Register	I3Cm_NTSTFC	0000_0000h	01ECh	32
Reserve	-	-	01F0h to 020Fh	-
Bus Condition Status Register	I3Cm_BCST	0000_0000h	0210h	32
Slave Status Register	I3Cm_SVST	0000_0000h	0214h	32
Wake-Up Unit Operating Status Register	I3Cm_WUST	0000_0000h	0218h	32
MsyncCNT Counter Capture Register	I3Cm_MRCCPT	0000_0000h	021Ch	32
Reserve	-	-	0220h to 0223h	-
Device Address Table Basic Register 0	I3Cm_DATBAS0	0000_0000h	0224h	32
Device Address Table Basic Register 1	I3Cm_DATBAS1	0000_0000h	0228h	32
Device Address Table Basic Register 2	I3Cm_DATBAS2	0000_0000h	022Ch	32
Device Address Table Basic Register 3	I3Cm_DATBAS3	0000_0000h	0230h	32
Device Address Table Basic Register 4	I3Cm_DATBAS4	0000_0000h	0234h	32
Device Address Table Basic Register 5	I3Cm_DATBAS5	0000_0000h	0238h	32
Device Address Table Basic Register 6	I3Cm_DATBAS6	0000_0000h	023Ch	32
Device Address Table Basic Register 7	I3Cm_DATBAS7	0000_0000h	0240h	32
Reserve	-	-	0244h to 02AFh	-
Slave Device Address Table Basic Register 0	I3Cm_SDATBAS0	0000_0000h	02B0h	32
Slave Device Address Table Basic Register 1	I3Cm_SDATBAS1	0000_0000h	02B4h	32
Slave Device Address Table Basic Register 2	I3Cm_SDATBAS2	0000_0000h	02B8h	32
Reserve	-	-	02BCh to 02CFh	-
Master Device Characteristic Table Register 0	I3Cm_MSDCT0	0000_0000h	02D0h	32
Master Device Characteristic Table Register 1	I3Cm_MSDCT1	0000_0000h	02D4h	32
Master Device Characteristic Table Register 2	I3Cm_MSDCT2	0000_0000h	02D8h	32
Master Device Characteristic Table Register 3	I3Cm_MSDCT3	0000_0000h	02DCh	32
Master Device Characteristic Table Register 4	I3Cm_MSDCT4	0000_0000h	02E0h	32
Master Device Characteristic Table Register 5	I3Cm_MSDCT5	0000_0000h	02E4h	32
Master Device Characteristic Table Register 6	I3Cm_MSDCT6	0000_0000h	02E8h	32
Master Device Characteristic Table Register 7	I3Cm_MSDCT7	0000_0000h	02ECh	32
Reserve	-	-	02F0h to 030Fh	-
Extended Device Address Table Basic Register	I3Cm_EXDATBAS	0000_0000h	0310h	32
Reserve	-	-	0314h to 031Fh	-
Slave Device Characteristic Table Register	I3Cm_SVDCT	0000_0000h	0320h	32
Slave Device Characteristic Table Provisional ID Low Register	I3Cm_SDCTPIDL	0000_0000h	0324h	32
Slave Device Characteristic Table Provisional ID High Register	I3Cm_SDCTPIDH	0000_0000h	0328h	32
Reserve	-	-	032Ch to 032Fh	-
Slave Device Address Register n (n = 0 to 2)	I3Cm_SVDVADn	0000_0000h	0330h + n x 0004h	32
Reserve	-	-	033Ch to 034Fh	-
CCC Slave Events Command Register	I3Cm_CSECMD	0000_0000h	0350h	32
CCC Enter Activity State Register	I3Cm_CEACTST	0000_0000h	0354h	32
CCC Max Write Length Register	I3Cm_CMWLG	0000_0000h	0358h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
CCC Max Read Length Register	I3Cm_CMRLG	0000_0000h	035Ch	32
CCC Enter Test Mode Register	I3Cm_CETSTMD	0000_0000h	0360h	32
CCC Get Device Status Register	I3Cm_CGDVST	0000_0000h	0364h	32
CCC Max Data Speed W (Write) Register	I3Cm_CMDSPW	0000_0000h	0368h	32
CCC Max Data Speed R (Read) Register	I3Cm_CMDSPR	0000_0000h	036Ch	32
CCC Max Data Speed T (Turnaround) Register	I3Cm_CMDSPT	0000_0000h	0370h	32
CCC Exchange Timing Support Information M (Mode) Register	I3Cm_CETSM	0000_0000h	0374h	32
CCC Exchange Timing Support Information S (State) Register	I3Cm_CETSS	0000_0000h	0378h	32
Reserve	-	-	037Ch to 037Fh	-
Bit Count Register	I3Cm_BITCNT	0000_0000h	0380h	32
Reserve	-	-	0384h to 0393h	-
Normal Queue Status Level Register	I3Cm_NQSTLV	0000_0000h	0394h	32
Normal Data Buffer Status Level Register	I3Cm_NDBSTLV0	0000_0001h	0398h	32
Reserve	-	-	039Ch to 03BFh	-
Normal Receive Status Queue Status Level Register	I3Cm_NRSQSTLV	0000_0000h	03C0h	32
Reserve	-	-	03C4h to 03CBh	-
Present State Debug Register	I3Cm_PRSTDBG	0000_0000h	03CCh	32
Master Error Counters Register	I3Cm_MSERRCNT	0000_0000h	03D0h	32
Reserve	-	-	03D4h to 03DFh	-
SC1 Capture monitor Register	I3Cm_SC1CPT	0000_0000h	03E0h	32
SC2 Capture monitor Register	I3Cm_SC2CPT	0000_0000h	03E4h	32

7.8.3.2 Register Description

The prefix (I3Cm_) of the register names is omitted in this and subsequent sections.

7.8.3.2.1 Protocol Selection Register (I3Cm_PRTS)

Access Size : 32 bits
Address : <I3C_base> + 0000h
Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PRTMD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	PRTMD	1h	RW	Protocol Mode 0b: I3C protocol mode 1b: I2C protocol mode

PRTMD bit (Protocol Mode)

PRTMD = 0b: I3C FIFO buffer transfer (Equivalent to HCI)

PRTMD = 1b: I^2C single buffer transfer

7.8.3.2.2 Bus Control Register (I3Cm_BCTL)

Access Size : 32 bits
Address : <I3C_base> + 0014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BUSE	RSM	ABT	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	HJACK CTL	-	-	-	-	-	-	-	INCBA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	BUSE	0h	RW	Bus Enable*2 0b: I3C bus operation is disabled. 1b: I3C bus operation is enabled.
30	RSM	0h	RW	Resume*2 Values when read: 0b: I3C is running. 1b: I3C is suspended (RW1C).
29	ABT	0h	RW	Abort*1 0b: I3C is running. 1b: I3C has aborted a transfer.
28 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8	HJACKCTL	0h	RW	Hot-Join Acknowledge Control*1 0b: ACK the Hot-Join request 1b: NACK and send broadcast CCC to disable Hot-Join
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	INCBA	0h	RW	Include I3C Broadcast Address*1 0b: Do not include I3C broadcast address for private transfers 1b: Include I3C broadcast address for private transfers

Note 1. This bit supports I3C master mode and I3C secondary master mode.

Note 2. This bit supports all I3C mode.

INCBA bit (Include I3C Broadcast Address)

This bit controls whether the I3C broadcast address (7Eh) is included for private transfers.

If the I3C broadcast address is not included for private transfers, then IBIs driven from Slaves might not win the arbitration, potentially delaying acceptance of the IBIs.

HJACKCTL bit (Hot-Join Acknowledge Control)

This bit acts as global control to either ACK (0) or NACK (1) all Hot-Join Requests arriving from the Devices on the I3C Bus. If set to NACK (1), then the NACK will be followed by the broadcast CCC to disable Hot-Join.

ABT bit (Abort)

When set to 1b, this bit allows I3C to relinquish control of the I3C Bus before completing the currently issued transfer.

In response to an ABORT request, I3C issues the STOP condition on the I3C Bus after the complete data byte is transferred or received.

The Driver shall clear the ABT bit to allow operation on the Bus.

If BCTL.ABT is set and ABORT processing is performed, please ignore ERR_STATUS of Response Descriptor.

RSM bit (Resume)

This bit is used to resume I3C operation following the Halt state.

I3C enters the Halt state (as indicated in register PRSTDBG) as a result of any type of error occurring in a transfer.

The error type is indicated by the field ERR_STATUS in register NRSPQP, NRSQP and NIBIQP).

After I3C has entered the Halt state, the application must write the value 1 to the RSM bit to resume I3C operation. I3C shall auto-clear the RSM bit once it has resumed making transfers (it has initiated the next Command).

BUSE bit (Bus Enable)

Enables or disables the operation on the I3C Bus by I3C.

Set the BUSE bit to 1b when using I3C. The SCL and SDA pins are placed in the active state when the BUSE bit is set to 1b. Set the BUSE bit to 0b when I3C is not to be used. The SCL and SDA pins are placed in the inactive state when the BUSE bit is set to 0b.

If the software sets this bit, then it also confirms that initialization is done, and that I3C can use the programmed register values (For example, generation of SCL on IBI detection, etc.). If this bit is not set, then I3C shall not generate SCL for incoming IBI.

Software may disable I3C bus operation while it is active, However:

- If a disable request occurs while receiving IBI, the actual disabling will not occur until reception of the IBI is complete.
- When the software reads the value 0 from this field, this indicates that I3C bus operation disable operation has completed.
  If commands remain in the command queue, do not set BUSE = 0b.

7.8.3.2.3 Master Device Address Register (I3Cm_MSDVAD)

Access Size : 32 bits
Address : <I3C_base> + 0018h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MDYADV	-	-	-	-	-	-	-	-	MDYAD[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	MDYADV	0h	RW	Master Dynamic Address Valid 0b: The master dynamic address field is not valid. 1b: The master dynamic address field is valid.
30 to 23	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
22 to 16	MDYAD[6:0]	0h	RW	Master Dynamic Address
15 to 0	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Note: This register supports I3C master mode.

MDYAD[6:0] bits (Master Dynamic Address)

This field is used to program I3C master dynamic address. I3C uses this address to respond to master transactions in I3C interface mode (slave or secondary master role).

In I3C main master mode, the software shall program the dynamic address as it self-assigns its dynamic address.

MDYADV bit (Master Dynamic Address Valid)

This bit indicates whether or not the value in the MDYAD field is valid.

In I3C main master mode, the user sets this bit to 1b as it self-assigns its dynamic address.

Note: After setting MSDVAD, and setting BCTL. BUSE = 1b, the device will act as main master.
Without setting MSDVAD, setting SVDCT.TBCR[7:6] = 00b (Device Role Slave), and setting BCTL.BUSE = 1b, the device will act as slave.
Without setting MSDVAD, setting MSDCTm.RBCR[7:6] = 01b (Device Role Master), and setting BCTL.BUSE = 1b, the device will act as slave.

7.8.3.2.4 Reset Control Register (I3Cm_RSTCTL)

For details on reset for each register, see **7.8.7 Reset Descriptions**.

Access Size : 32 bits
Address : <I3C_base> + 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INTLRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	RSQRST	IBIQRST	RDBRST	TDBRST	RSPQRST	CMDQRST	RI3CRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	INTLRST	0h	RW	Internal Software Reset 0b: Release of some registers and internal state. 1b: Reset of some registers and internal state.
15 to 7	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6	RSQRST	0h	RW	Receive Status Queue Software Reset*2 0b: The Receive Status Queue in I3C is not flushed. 1b: The Receive Status Queue in I3C is flushed.
5	IBIQRST	0h	RW	IBI Queue Software Reset*1 0b: The IBI Queues in I3C are not flushed. 1b: The IBI Queues in I3C are flushed.
4	RDBRST	0h	RW	Receive Data Buffer Software Reset*1 0b: The Receive Queues in I3C are not flushed. 1b: The Receive Queues in I3C are flushed.
3	TDBRST	0h	RW	Transmit Data Buffer Software Reset*1 0b: The Transmit Queues in I3C are not flushed. 1b: The Transmit Queues in I3C are flushed.
2	RSPQRST	0h	RW	Response Queue Software Reset*1 0b: The Response Queues in I3C are not flushed. 1b: The Response Queues in I3C are flushed.
1	CMDQRST	0h	RW	Command Queue Software Reset*1 0b: The Command Queues in I3C are not flushed. 1b: The Command Queues in I3C are flushed.
0	RI3CRST	0h	RW	I3C Software Reset 0b: Release I3C reset. 1b: Initiate I3C reset.

Note 1. This bit supports all I3C mode.

Note 2. This bit supports I3C secondary master mode and I3C slave mode.

RI3CRST bit (I3C Software Reset)

On Driver setting this bit to 1b, I3C shall be reset and disabled.

All registers shall return to their reset values, and the software shall re-initialize I3C.

This field is cleared automatically upon I3C reset completion. This field also resets all Queues in I3C.

CMDQRST bit (Command Queue Software Reset)

On software setting this bit to 1b, the Command Queues in I3C shall be flushed.

This field shall be cleared automatically upon Command Queue reset completion.

RSPQRST bit (Response Queue Software Reset)

On software setting this bit to 1b, the Response Queues in I3C shall be flushed.

This field shall be cleared automatically upon Response Queue reset completion.

TDBRST bit (Transmit Data Buffer Software Reset)

On software setting this bit to 1b, the Transmit Data Buffers in I3C shall be flushed.

This field shall be cleared automatically upon Transmit Data Buffer reset completion.

RDBRST bit (Receive Data Buffer Software Reset)

On software setting this bit to 1b, the Receive Data Buffers in I3C shall be flushed.

This field shall be cleared automatically upon completion of Receive Data Buffer reset.

IBIQRST bit (IBI Queue Software Reset)

On software setting this bit to 1b, the IBI Queues in I3C shall be flushed.

This field shall be cleared automatically upon completion of IBI Queue reset.

RSQRST bit (Receive Status Queue Software Reset)

On software setting this bit to 1b, the Receive Status Queues in I3C shall be flushed. This field shall be cleared automatically upon Receive Status Queue reset completion.

INTLRST bit (Internal Software Reset)

When set to 1b, some of registers is reset. For details on the registers to be reset, see **7.8.7 Reset Descriptions**.

Note: When setting internal software reset during bus operation enable, use DISEC CCC in advance to disable IBI transmission to I3C Slave in order to avoid conflict with IBI from I3C Slave connected to I3C Bus.

7.8.3.2.5 Present State Register (I3Cm_PRSST)

Access Size : 32 bits
Address : <I3C_base> + 0024h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	PRSSTWP	-	-	TRMD	-	CRMS	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	W	RW	RW	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7	PRSSTWP	-	W	The read value is undefined Present State Write Protect*2 0b: CRMS bit is protected. 1b: CRMS bit can be written when writing simultaneously with the value of the target bit.
6 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	TRMD	0h	R	Transmit/Receive Mode*3 0b: Receive mode 1b: Transmit mode
3	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	CRMS	0h	RW*1	Current Master*2 0b: The Master is not the Current Master, and must request and acquire bus ownership before initiating any transfer. 1b: The Master is the Current Master, and as a result can initiate transfers.
1 to 0	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Note 1. When the PRSSTWP bit is set to 1b, the CRMS bit can be written to.

Note 2. This bit supports I^2C, I3C master, and I3C secondary master mode.

Note 3. This bit supports I^2C mode.

CRMS bit (Current Master)

Indicates the set condition and reset condition of each operation mode.

- Operation Mode [I^2C/I3C common]

  [Clearing conditions]

  - When 0b written to the PRSST.CRMS by the software.

  [Setting conditions]

  - When 1b written to the PRSST.CRMS by the software.

- Operation Mode [I^2C]

  [Clearing conditions]

  - When STOP is issued.

- When Master arbitration-lost is detected.

[Setting conditions]

- When START is issued.

- Operation Mode [I3C Main Master]

[Clearing conditions]

- When 0b written to the MSDVAD.MDYADV by the software.
- When GETACCMST transmission is successfully completed by issuing STOP, after responding ACK to the Mastership-Request received from the Secondary Master.

[Setting conditions]

- When 1b written to the MSDVAD.MDYADV by the software.
- When GETACCMST reception is successfully completed by issuing STOP, after the ACK is responded to the Mastership-Request transmitted to the Secondary Master.

- Operation Mode [I3C Secondary Master]

[Clearing condition]

- When GETACCMST transmission is successfully completed by issuing STOP, after responding ACK to the Mastership-Request received from the Non-Current Master.

[Setting condition]

- When GETACCMST reception is successfully completed by issuing STOP, after the ACK is responded to the Mastership-Request transmitted to the Current Master.

The PRSST register returns I3C current state.

State has two parts: this register which is mandatory, and an additional optional PRSST_DEBUG register intended for debug purposes (see the Debug Capability registers in the Extended Capabilities list).

TRMD bit (Transmit/Receive Mode)

This bit indicates transmit or receive mode.

I3C is in receive mode when the TRMD bit is set to 0b and is in transmit mode when the bit is set to 1b. Combination of this bit and the CRMS bit indicates the operating mode of I3C.

The value of TRMD bit is automatically changed to 1b for transmit mode or 0b for receive mode by issuing or detection of a START condition and setting of the R/W# bit.

[Setting conditions]

- When a START condition is issued normally according to the START condition issuance request (when a START condition is detected with the CNDCTL.STCND bit set to 1b).
- When a Repeated START condition is issued normally according to the Repeated START condition issuance request (when a Repeated START condition is detected with the CNDCTL.SRCND bit set to 1b).
- When the R/W# bit added to the slave address is set to 0b in master mode.
- When the address received in slave mode matches the address enabled in SVCTL, with the R/W# bit set to 1b.

[Clearing conditions]

- When a STOP condition is detected.
- The ALF (arbitration-lost) flag in BST being set to 1b.

- In master mode, reception of a slave address to which an R/W# bit with the value 1 is appended.
- In slave mode, a match between the received address and the address enabled in SVCTL when the value of the received R/W# bit is 0b (including cases where the received address is the general call address).
- In slave mode, a Repeated START condition is detected (a Repeated START condition is detected with BCST.BFREF = 0b and CRMS = 0b).

PRSSTWP bit (Present State Write Protect)

PRSSTWP is always 0b when reading.

When writing to PRSST, writing 1 to this bit at the same time enables writing to CRMS bit.

7.8.3.2.6 Internal Status Register (I3Cm_INST)

The Interrupt Status register reflects the status of outstanding interrupt(s).

Access Size : 32 bits
Address : <I3C_base> + 0030h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	INEF	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10	INEF	0h	RW*1	Internal Error Flag 0b: I3C Internal Error has not detected. 1b: I3C Internal Error has detected.
9 to 0	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Note: This register supports all I3C mode.

Note 1. Clearing (to 0) condition: Writing 0b after 1b is read.

INEF bit (Internal Error Flag)

When this bit is 1b, it indicates that I3C Internal Error has detected.

When this bit is 0b, it indicates that I3C Internal Error has not detected.

[Setting conditions]

- The following 1 is satisfied and any of the following 2) to 9) are satisfied.

1) The INSTE.INEE bit = 1b
2) When transmit data is written to the Transmit Data Buffer that is completely full.
3) When received data is read from the Receive Data Buffer that is completely empty.
4) When Command Descriptor is written to the Command Queue that is completely full.
5) When Response Descriptor is read from the Response Status Queue that is completely empty.
6) When Receive Status Descriptor is read from the Receive Status Queue that is completely empty.
7) When IBI Status Descriptor is read from the IBI Queue under the condition that the IBI Queue is completely empty and PRSST.CRMS = 1b.
8) When IBI Data is written to the IBI Queue under the condition that the IBI Queue is completely full and PRSST.CRMS = 0b.
9) When the Response Status Queue, IBI Status Queue or Receive Status Queue overflows.

[Clearing condition]

- When 0b is written to the INEF bit after reading INEF bit = 1b.

7.8.3.2.7 Internal Status Enable Register (I3Cm_INSTE)

Access Size : 32 bits

Address : <I3C_base> + 0034h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	INEE	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10	INEE	0h	RW	Internal Error Enable 0b: Disable INST.INEF 1b: Enable INST.INEF
9 to 0	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Note: This register supports all I3C mode.

INEE bit (Internal Error Enable)

When this bit set to 1b, it enables detection of I3C Internal Error.

When this bit set to 0b, it disables detection of I3C Internal Error.

7.8.3.2.8 Internal Interrupt Enable Register (I3Cm_INIE)

Access Size : 32 bits

Address : <I3C_base> + 0038h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	INEIE	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10	INEIE	0h	RW	Internal Error Interrupt Enable 0b: Disables Non-recoverable Internal Error Interrupt Signal. 1b: Enables Non-recoverable Internal Error Interrupt Signal.
9 to 0	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Note: This register supports all I3C mode.

INEIE bit (Internal Error Interrupt Enable)

When set to 1b and register INEF is set, the hardware Controller asserts an interrupt to the Host.

7.8.3.2.9 Internal Status Force Register (I3Cm_INSTFC)

Access Size : 32 bits
Address : <I3C_base> + 003Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	INEFC	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	-	All 0	W	Reserved The written value should always be 0b.
10	INEFC	0h	W	Internal Error Force 0b: Not force a specific interrupt 1b: Force a specific interrupt
9 to 0	-	All 0	W	Reserved The written value should always be 0b.

Note: This register supports all I3C mode.

INEFC bit (Internal Error Force)

For debug, helps to force this interrupt.

7.8.3.2.10 Device Characteristic Table Register (I3Cm_DVCT)

Access Size : 32 bits

Address : <I3C_base> + 0044h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	IDX[4:0]					-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read.
23 to 19	IDX[4:0]	0h	R	DCT Table Index Current index of the DCT, which is used as the starting index for the I3C ENTDAA CCC.
18 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read.

Note: This register supports I3C master mode and I3C secondary master mode.

IDX[4:0] bits (DCT Table Index)

Once the complete characteristics of device that won the arbitration are written to the DCT (during ENTDAA using Address Assignment Command) this index is incremented by 1.

7.8.3.2.11 IBI Notify Control Register (I3Cm_IBINCTL)

Access Size : 32 bits
Address : <I3C_base> + 0058h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	NRSIRCTL	-	NRMRCTL	NRHJCTL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3	NRSIRCTL	0h	RW	Notify Rejected Slave Interrupt Request Control 0b: Do not pass rejected IBI Status to the IBI Queue/Rings, if the incoming SIR is NACKed and is auto-disabled based on DVSIRRJ field in relevant DATBASn register. 1b: Pass rejected IBI Status to the IBI Queue/Rings, if the incoming SIR is NACKed and is auto-disabled based on DVSIRRJ field in relevant DATBASn register.
2	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	NRMRCTL	0h	RW	Notify Rejected Master Request Control 0b: Do not pass rejected IBI Status to IBI Queue/Ring, if the incoming Master Request is NACKed and is auto-disabled based on DVMRRJ field in relevant DATBASn register. 1b: Pass rejected IBI Status to the IBI Queue, if the incoming Master Request is NACKed and is autodisabled based on DVMRRJ field in relevant DATBASn register.
0	NRHJCTL	0h	RW	Notify Rejected Hot-Join Control 0b: Do not pass rejected IBI Status to IBI Queue, if the incoming Hot-Join request is NACKed and is auto-disabled based on field HJACKCTL of BCTL. 1b: Pass rejected IBI Status to the IBI Queue, if the incoming Hot-Join request is NACKed and is auto-disabled based on field HJACKCTL of BCTL.

Note: This register supports I3C master mode and I3C secondary master mode.

NRHJCTL bit (Notify Rejected Hot-Join Control)

Enables or disables reporting rejection of individual Hot Join requests.

NRMRCTL bit (Notify Rejected Master Request Control)

Enables or disables reporting rejection of individual Master Requests.

NRSIRCTL bit (Notify Rejected Slave Interrupt Request Control)

Enables or disables reporting rejection of individual Slave Interrupt Requests (SIR).

7.8.3.2.12 Bus Function Control Register (I3Cm_BFCTL)

Access Size : 32 bits
Address : <I3C_base> + 0060h
Initial Value : 0000_0101h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HSME	FMPE	-	SMBS	-	-	-	SCSYNE	-	-	-	-	-	SALE	NALE	MALE
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15	HSME	0h	RW	High Speed Mode Enable*1 0b: Disable High Speed Mode. 1b: Enable High Speed Mode.
14	FMPE	0h	RW	Fast-mode Plus Enable*1 0b: No Fm+ slope control circuit is used for the SCL30 pin and SDA30 pin. 1b: An Fm+ slope control circuit is used for the SCL30 pin and SDA30 pin.
13	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12	SMBS	0h	RW	SMBus/I2C Bus Selection*1 0b: The I2C bus is selected. 1b: The SMBus is selected.
11 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8	SCSYNE	1h	RW	SCL30 Synchronous Circuit Enable*1 0b: No SCL30 synchronous circuit is used. 1b: An SCL30 synchronous circuit is used.
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	SALE	0h	RW	Slave Arbitration-Lost Detection Enable*1 0b: Slave arbitration-lost detection disabled 1b: Slave arbitration-lost detection enabled
1	NALE	0h	RW	NACK Transmission Arbitration-Lost Detection Enable*1 0b: NACK transmission arbitration-lost detection disabled 1b: NACK transmission arbitration-lost detection enabled
0	MALE	1h	RW	Master Arbitration-Lost Detection Enable*1 0b: Master arbitration-lost detection disabled Disables the arbitration-lost detection function and does not clear the CRMS and TRMD bits in PRSST automatically when arbitration is lost. 1b: Master arbitration-lost detection enabled Enables the arbitration-lost detection function and clears the CRMS and TRMD bits in PRSST automatically when arbitration is lost.

Note 1. This bit supports I^2C mode.

MALE bit (Master Arbitration-Lost Detection Enable)

This bit is used to specify whether to use the arbitration-lost detection function in master mode. Normally, set this bit to 1b.

NALE bit (NACK Transmission Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when ACK is detected during transmission of NACK in receive mode (such as when slaves with the same address exist on the bus or when two or more masters select the same slave device simultaneously with different number of receive bytes).

SALE bit (Slave Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when a value different from the value being transmitted is detected on the bus in slave transmit mode (such as when slaves with the same address exist on the bus or when a mismatch with the transmit data occurs due to noise).

SCSYNE bit (SCL Synchronous Circuit Enable)

This bit is used to specify whether to synchronize the SCL clock with the SCL input clock. Normally, set this bit to 1b.

When the SCSYNE bit set to 0b (no SCL synchronous circuit used), I3C does not synchronize the SCL clock with the SCL input clock. In this setting, I3C outputs the SCL clock with the transfer rate set in STDBR and EXTBR regardless of the SCL line state. For this reason, if the bus load of the I^2C bus line is much larger than the specification value or if the SCL clock output overlaps in multiple masters, the short-cycle SCL clock that does not meet the specification may be output. When no SCL synchronous circuit uses, it also affects the issuance of a START condition, Repeated START condition, and STOP condition, and the continuous output of extra SCL clock cycles.

This bit must not be set to 0b except for checking the output of the set transfer rate.

FMPE bit (Fast-mode Plus Enable)

This bit is used to specify whether to use a slope control circuit for Fast-mode Plus [Fm+].

When this bit is set to 1b, a slope control circuit conforming to the Fast-mode Plus [Fm+] slope control specification (tof) of the I3C-bus is selected. When this bit is set to 0b, a slope control circuit conforming to the Standard-mode [Sm] and Fast- mode [fm] slope control specification (tof) of the I3C-bus is selected.

Set this bit to 1b when using the transmission rate within a range up to 1 Mbps (Fast-mode Plus [Fm+]) of the I3C-bus specification. Set this bit to 0b when using the transmission rate at other rates (up to 100 kbps [Sm], up to 400 kbps [Fm]) or for SMBus (10 to 100 kbps).

Note: When communicating in Hs-mode, set as follows.
- Set FMPE to 0b when sending Hs-mode master code (0000 1XXXb) with Fast-mode.
- Set FMPE to 1b when sending Hs-mode master code (0000 1XXXb) with Fast-mode Plus.

HSME bit (High Speed Mode Enable)

This bit is used for communicating in Hs-mode.

When this bit is set to 1b, the Hs-mode master code is recognized and Hs-mode communication is possible.

After the START condition is detected, if Hs-mode master code (0000 1XXXb) transmission is recognized, Hs-mode communication starts from Repeated START after receiving the NACK response.

It communicates at the bit rate set in STDBR until the NACK response, and automatically switches from Repeated START condition issuance after receiving the NACK response to the bit rate set in EXTBR.

Hs-mode continues until a STOP condition is detected.

When the STOP condition is detected, the bit rate is automatically switched to the bit rate set in STDBR.

Note: When this bit is set to 1b, the BST.NACKDF bit will not be set even if a NACK response is received after sending the Hs-mode master code.

7.8.3.2.13 Slave Control Register (I3Cm_SVCTL)

Access Size : 32 bits
Address : <I3C_base> + 0064h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	SVAE2	SVAE1	SVAE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HOAE	-	-	-	-	-	-	-	-	DVIDE	HSMCE	-	-	-	-	GCAE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18	SVAE2	0h	RW	Slave Address Enable 2*2 0b: Slave 2 disabled 1b: Slave 2 enabled
17	SVAE1	0h	RW	Slave Address Enable 1*2 0b: Slave 1 disabled 1b: Slave 1 enabled
16	SVAE0	0h	RW	Slave Address Enable 0*2 0b: Slave 0 disabled 1b: Slave 0 enabled
15	HOAE	0h	RW	Host Address Enable*1 0b: Host address detection disabled 1b: Host address detection enabled
14 to 7	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6	DVIDE	0h	RW	Device-ID Address Enable*1 0b: Device-ID address detection disabled 1b: Device-ID address detection enabled
5	HSMCE	0h	RW	Hs-mode Master Code Enable*1 0b: Hs-mode Master Code Detection disabled 1b: Hs-mode Master Code Detection enabled
4 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	GCAE	0h	RW	General Call Address Enable*1 0b: General call address detection disabled 1b: General call address detection enabled

Note 1. This bit supports I^2C mode.

Note 2. These bits support I^2C, I3C secondary master, and I3C slave mode.

GCAE bit (General Call Address Enable)

This bit is used to specify whether to ignore the general call address (0000 000 + 0 (write): All 0) when it is received. When this bit is set to 1b, if the received slave address matches the general call address, I3C recognizes the received slave address as the general call address independently of the slave addresses set in the SVDVADn.SVAD[9:0] bits (n = 0 to 2) and performs data receive operation.

When this bit is set to 0b, the received slave address is ignored even if it matches the general call address.

HSMCE bit (Hs-mode Master Code Enable)

This bit is used to specify whether to recognize and execute the Hs-mode master code (00001xxxb) is received in the first byte after a START condition is detected.

When this bit is set to 1b, if the received first byte matches the Hs-mode master code, IIC recognizes that the Hs-mode master code has been received.

The first byte after Repeated START after NACK response to Hs-mode master code is recognized as a slave address and compared with the slave address set by SVDVADn.SVAD[9:0].

If the addresses match, the transmission / reception operation continues according to the R/W# bit value.

Hs-mode continues until a STOP condition is detected.

When this bit is set to 0b, I3C will ignore the pattern until a STOP condition is detected, even if it matches the Hs-mode master code.

Note: When this bit is set to 1b, SCSTRCTL.ACKTWE bit must be set to 0b and SCSTRCTL.RWE bit must be set to 1b.

DVIDE bit (Device-ID Address Enable)

This bit is used to specify whether to recognize and execute the Device-ID address when a device ID (1111 100) is received in the first byte after a START condition or Repeated START condition is detected.

When this bit is set to 1b, if the received first byte matches the Device-ID, I3C recognizes that the Device-ID address has been received. When the following R/W# bit is 0b (write), I3C recognizes the second and the following bytes as slave addresses and continues the receive operation.

When this bit is set to 0b, I3C ignores the received first byte even if it matches the Device ID address and recognizes the first byte as a normal slave address.

For details on the Device-ID address detection, see **(c) Device-ID Address Detection [I2C mode]**.

HOAE bit (Host Address Enable)

This bit is used to specify whether to ignore received host address (0001 000) when the BFCTL.SMBS bit = 1b.

When this bit is set to 1b while the SMBS bit = 1b, if the received slave address matches the host address, I3C recognizes the received slave address as the host address independently of the slave addresses set in the SVDVADn.SVAD[9:0] bits (n = 0 to 2) and performs the receive operation.

When the SMBS bit or the HOAE bit is set to 0b, the received slave address is ignored even if it matches the host address.

SVAEn bits (Slave Address Enable n) (n = 0 to 2)

This bit is used to enable or disable the slave address set in the SVDVADn.SVAD[9:0] bits.

When this bit is set to 1b, the slave address set in the SVAD[9:0] bits is enabled and is compared with the received slave address.

When this bit is set to 0b, the slave address set in the SVAD[9:0] bits is disabled and is ignored even if it matches the received slave address.

7.8.3.2.14 Reference Clock Control Register (I3Cm_REFCKCTL)

Access Size : 32 bits

Address : <I3C_base> + 0070h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	IREFCKS[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	IREFCKS*[1] [2:0]	0h	RW	Internal Reference Clock Selection Selects the internal reference clock source (I3Cϕ) for I3C. 000b: I3C_0_TCLK/1 clock 001b: I3C_0_TCLK/2 clock 010b: I3C_0_TCLK/4 clock 011b: I3C_0_TCLK/8 clock 100b: I3C_0_TCLK/16 clock 101b: I3C_0_TCLK/32 clock 110b: I3C_0_TCLK/64 clock 111b: I3C_0_TCLK/128 clock

Note 1. Set the IREFCKS[2:0] bits to 000b in I3C mode.

7.8.3.2.15 Standard Bit Rate Register (I3Cm_STDBR)

The STDBR register sets the bit rate according to the operating speed.

- I²C mode: Bit rate setting when communicating with Standard-mode / Fast-mode / Fast-mode plus
- I3C master mode: Bit rate setting selected by mode bit of command descriptor
- I3C slave mode: I3C bit rate setting

Access Size : 32 bits
Address : <I3C_base> + 0074h
Initial Value : 3F3F_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSBRPO	-	SBRHP[5:0]						-	-	SBRLP[5:0]					
Initial Value	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SBRHO[7:0]								SBRLO[7:0]							
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	DSBRPO	0h	RW	Double the Standard Bit Rate Period for Open-Drain*4 0b: The time period set for SBRHO[7:0] and SBRLO[7:0] is not doubled. 1b: The time period set for SBRHO[7:0] and SBRLO[7:0] is doubled.
30	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
29 to 24	SBRHP[5:0]	3Fh	RW	Standard Bit Rate High-Level Period Push-Pull*3 Count value of the high-level period of SCL30 clock
23, 22	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
21 to 16	SBRLP[5:0]	3Fh	RW	Standard Bit Rate Low-level Period Push-Pull*2 Count value of the low-level period of SCL30 clock
15 to 8	SBRHO[7:0]	FFh	RW	Standard Bit Rate High-Level Period Open-Drain*1 Count value of the high-level period of SCL30 clock
7 to 0	SBRLO[7:0]	FFh	RW	Standard Bit Rate Low-Level Period Open-Drain*1 Count value of the low-level period of SCL30 clock

Note 1. These bits support I²C, I3C master, and I3C secondary master mode.

Note 2. These bits support I3C master mode and I3C secondary master mode.

Note 3. These bits support all I3C mode.

Note 4. This bit supports I²C, I3C master, and I3C secondary master mode.

The I²C transfer rate and the SCL clock duty are calculated using the following expression.

$$\text{Transfer rate} = 1/\{[(\text{High} - \text{Level Period} + \alpha^{*1}) + (\text{Low} - \text{Level Period} + \alpha)]/\text{I3C}\phi^{*2} + \text{SCL line rising time [tr]}^{*3} + \text{SCL line falling time [tf]}^{*3}\}$$

$$\text{Duty cycle} = \{\text{SCL line rising time [tr]} + (\text{High} - \text{Level Period} + \alpha)/\text{I3C}\phi\}/\{\text{SCL line falling time [tf]} + (\text{Low} - \text{Level Period} + \alpha)/\text{I3C}\phi\}$$

Note 1. α depend on the number of stages in the noise filter.

Note 2. I3Cϕ = I3C_0_TCLK × Division ratio

Note 3. The SCL line rising time [tr] and SCL line falling time [tf] depend on the total bus line capacitance [Cb] and the pull-up resistor [Rp]. For details, see the I²C-bus specification from NXP Semiconductors.

The I3C transfer rate and the SCL clock duty are calculated using the following expression.

$$\text{Transfer rate} = 1/[(\text{High} - \text{Level Period} + \text{Low} - \text{Level Period})/\text{I3C}\phi + \text{SCL line rising time [tr]} + \text{SCL line falling time [tf]}]$$

$$\text{Duty cycle} = [\text{SCL line rising time [tr]} + \text{High} - \text{Level Period}/\text{I3C}\phi]/[\text{SCL line falling time [tf]} + \text{Low} - \text{Level Period}/\text{I3C}\phi]$$

SBRLO[7:0] bits (Count value of the Low-level period of SCL clock)

The SBRLO[7:0] bits are used to set the low-level period of SCL clock in Open-Drain mode.

I3C counts the low-level period with the internal reference clock source (I3Cϕ) specified by the REFCKCTL.IREFCKS[2:0] bits. It also works to generate the data setup time for automatic SCL low-hold operation (see **(6) Clock Stretching [I2C mode]**); when I3C is used in I²C slave mode, these bits need to be set to a value longer than the data setup time*[1].

If the digital noise filter is enabled (INCTL.DNFE = 1b), set the SBRLO[7:0] bits to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the INCTL.DNFS[3:0] bits.

Note 1. Data setup time (tSU: DAT)
250 ns (up to 100 kbps: Standard-mode [Sm])
100 ns (up to 400 kbps: Fast-mode [Fm])
50 ns (up to 1 Mbps: Fast-mode plus [Fm+])
10 ns (up to 3.4 Mbps: Hs-mode [HS])

SBRHO[7:0] bits (Count value of the High-level period of SCL clock)

The SBRHO[7:0] bits use to set the high-level period of SCL clock in Open-Drain mode. SBRHO[7:0] bits are valid in master mode. If I3C is used only in I²C slave mode, these bits need not to set the high-level period.

I3C counts the high-level period with the internal reference clock source (I3Cϕ) specified by the REFCKCTL.IREFCKS[2:0] bits.

If the digital noise filter is enabled (the INCTL.DNFE bit = 1b), set the SBRHO[7:0] bits to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the INCTL.DNFS[3:0] bits.

SBRLP[5:0] bits (Standard Bit Rate Low-level Period Push-Pull)

SBRLP[5:0] bits are used to set the low-level period of SCL clock in Push-Pull.

I3C counts the low-level period with the internal reference clock source (I3Cϕ) specified by the REFCKCTL.IREFCKS[2:0] bits.

If the digital noise filter is enabled (the INCTL.DNFE bit = 1b), set the SBRLP[5:0] bits to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the INCTL.DNFS[3:0] bits.

SBRHP[5:0] bits (Standard Bit Rate High-Level Period Push-Pull)

SBRHP[5:0] bits is used to set the high-level period of SCL clock in Push-Pull mode.

SBRHP[5:0] bits are valid in master mode. If I3C is used only in I^2C slave mode, these bits need not to set the high-level period.

I3C counts the high-level period with the internal reference clock source (I3Cϕ) specified by the REFCKCTL.IREFCKS[2:0] bits.

If the digital noise filter is enabled (the INCTL.DNFE bit = 1b), set the SBRHP[5:0] bits to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the INCTL.DNFS[3:0] bits.

DSBRPO bit (Double the Standard Bit Rate Period for Open-Drain)

When DSBRPO = 1b, double the high-level period that is set in SBRHO[7:0] and double the low-level period that is set in SBRLO[7:0].

Table 7.8-5 Requirement and Usage of Setting in Each Mode

	Device Mode				
Bit Name	I^2C Master	I^2C Slave	I3C Master	I3C Secondary Master	I3C Slave
SBRHP[5:0]	Not used	Do not use	Setting required*3	Setting required*4	Not used
SBRLP[5:0]	Not used	Do not use	Setting required*3	Setting required*5	Not used
SBRHO[7:0]	Setting required*1	Do not use	Setting required*3	Setting required*5	Not used
SBRLO[7:0]	Setting required*1	Setting required*2	Setting required*3	Setting required*5	Not used

Note 1. The setting value is used for the data rate of ST, FM, and FM+ mode.

Note 2. The setting value is used for the data setup time of automatic SCL low-hold operation.

Note 3. The setting value is used for the data rate of each communication.

Note 4. When operating with I3C Master, the setting value is used for the data rate of each communication.
When operating with I3C Slave, the setting value is not used.

Note 5. When operating with I3C Master, the setting value is used for the data rate of each communication.
When operating with I3C Slave, the setting value is not used.

7.8.3.2.16 Extended Bit Rate Register (I3Cm_EXTBR)

The EXTBR register sets the bit rate according to the operating speed.

- I^2C mode: Bit rate setting for communicating in high-speed mode
- I3C master mode: Bit rate setting selected by mode bit of command descriptor
- I3C slave mode: Unused

Access Size : 32 bits
Address : <I3C_base> + 0078h
Initial Value : 3F3F_FFFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	EBRHP[5:0]						-	-	EBRLP[5:0]					
Initial Value	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EBRHO[7:0]								EBRLO[7:0]							
Initial Value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
29 to 24	EBRHP[5:0]	3Fh	RW	Extended Bit Rate Low-Level Period Push-Pull*2 Count value of the high-level period of SCL30 clock
23, 22	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
21 to 16	EBRLP[5:0]	3Fh	RW	Extended Bit Rate Low-Level Period Push-Pull*2 Count value of the low-level period of SCL30 clock
15 to 8	EBRHO[7:0]	FFh	RW	Extended Bit Rate High-Level Period Open-Drain*1 Count value of the high-level period of SCL30 clock
7 to 0	EBRLO[7:0]	FFh	RW	Extended Bit Rate Low-Level Period Open-Drain*1 Count value of the low-level period of SCL30 clock

Note 1. These bits support I^2C, I3C master, and I3C secondary master mode.

Note 2. These bits support I3C master mode and I3C secondary master mode.

EBRLO[7:0] bits (Extended Bit Rate Low-Level Period Open-Drain)

See SBRLO[7:0] bits of **7.8.3.2.15 Standard Bit Rate Register (I3Cm_STDBR)** for details. Watch SBRHO, SBRLO as EBRHO[7:0], EBRLO[7:0].

EBRHO[7:0] bits (Extended Bit Rate High-Level Period Open-Drain)

See SBRHO[7:0] bits of **7.8.3.2.15 Standard Bit Rate Register (I3Cm_STDBR)** for details. Watch SBRHO, SBRLO as EBRHO[7:0], EBRLO[7:0].

EBRLP[5:0] bits (Extended Bit Rate Low-Level Period Push-Pull)

See SBRLP[5:0] bits of **7.8.3.2.15 Standard Bit Rate Register (I3Cm_STDBR)** for details. Watch SBRHP, SBRLP as EBRHP[5:0], EBRLP[5:0].

EBRHP[5:0] bits (Extended Bit Rate High-Level Period Push-Pull)

See SBRHP[5:0] bits of **7.8.3.2.15 Standard Bit Rate Register (I3Cm_STDBR)** for details. Watch SBRHP, SBRLP as EBRHP[5:0], EBRLP[5:0].

Table 7.8-6 Requirement and Usage of Setting in Each Mode

Bit Name	Device Mode				
	I^2C Master	I^2C Slave	I3C Master	I3C Secondary Master	I3C Slave
EBRHP[5:0]	Not used	Not used	Setting required*3	Setting required*4	Not used
EBRLP[5:0]	Not used	Not used	Setting required*3	Setting required*4	Not used
EBRHO[7:0]	Setting required*1	Not used	Setting required*3	Setting required*4	Not used
EBRLO[7:0]	Setting required*1	Setting required*2	Setting required*3	Setting required*4	Not used

Note 1. The setting value is used for the data rate of High-Speed mode.

Note 2. The setting value is used for the data setup time of automatic SCL low-hold operation in Hs-mode.

Note 3. The setting value is used for the data rate of each communication.

Note 4. When operating with I3C Master, the setting value is used for the data rate of each communication.
When operating with I3C Slave, the setting value is not used.

7.8.3.2.17 Bus Free Condition Detection Time Register (I3Cm_BFRECDT)

Access Size : 32 bits
Address : <I3C_base> + 007Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	FRECYC[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8 to 0	FRECYC[8:0]	0h	RW	Bus Free Condition Detection Cycle The count value is a period for detecting the Bus free condition.

FRECYC[8:0] bits (Bus Free Condition Detection Cycle)

I3C counts the period for detecting the Bus free condition with the I3Cϕ.

These bits set the Bus Free period. This Bus Free period is counted by the internal reference clock (I3Cϕ) selected by the REFCKCTL.IREFCKS[2:0] bits. See the BCST.BFREF flag for Bus Free detection behavior.

7.8.3.2.18 Bus Available Condition Detection Time Register (I3Cm_BAVLCDT)

Access Size : 32 bits
Address : <I3C_base> + 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	AVLCYC[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8 to 0	AVLCYC[8:0]	0h	RW	Bus Available Condition Detection Cycle The count value is a period for detecting the Bus available condition.

Note: This register supports all I3C mode.

AVLCYC[8:0] bits (Bus Available Condition Detection Cycle)

I3C counts the period for detecting the Bus available condition with the I3Cϕ.

These bits set the Bus Available period. This Bus Available period is counted by the internal reference clock (I3Cϕ) selected by the REFCKCTL.IREFCKS[2:0] bits. See the BCST.BAVLF flag for Bus Available detection behavior.

7.8.3.2.19 Bus Idle Condition Detection Time Register (I3Cm_BIDLCDT)

Access Size : 32 bits
Address : <I3C_base> + 0084h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	IDLCYC[17:16]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IDLCYC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
17 to 0	IDLCYC[17:0]	0h	RW	Bus Idle Condition Detection Cycle The count value is a period for detecting the Bus idle condition.

Note: This register supports all I3C mode.

IDLCYC[17:0] bits (Bus Idle Condition Detection Cycle)

I3C counts the period for detecting the Bus idle condition with the I3Cϕ.

These bits set the Bus Idle period. This Bus Idle period is counted by the internal reference clock (I3Cϕ) selected by the REFCKCTL.IREFCKS[2:0] bits. See the BCST.BIDLF flag for Bus Available detection behavior.

7.8.3.2.20 Output Control Register (I3Cm_OUTCTL)

Access Size : 32 bits

Address : <I3C_base> + 0088h

Initial Value : 0000_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDODCS	-	-	-	-	SDOD[2:0]			-	-	-	EXCYC	-	SOCWP	SCOC	SDOC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	W	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15	SDODCS	0h	RW	SDA30 Output Delay Clock Source Selection*3 0b: The internal reference clock (I3Cϕ) is selected as the clock source of the SDA30 output delay counter. 1b: The internal reference clock divided by 2 (I3Cϕ/2) is selected as the clock source of the SDA30 output delay counter.*4
14 to 11	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10 to 8	SDOD[2:0]	0h	RW	SDA30 Output Delay*2 000b: No output delay 001b: 1 I3Cϕ cycle (When OUTCTL.SDODCS = 0b (I3Cϕ)) 1 or 2 I3Cϕ cycles (When OUTCTL.SDODCS = 1b (I3Cϕ/2)) 010b: 2 I3Cϕ cycles (When OUTCTL.SDODCS = 0b (I3Cϕ)) 3 or 4 I3Cϕ cycles (When OUTCTL.SDODCS = 1b (I3Cϕ/2)) 011b: 3 I3Cϕ cycles (When OUTCTL.SDODCS = 0b (I3Cϕ)) 5 or 6 I3Cϕ cycles (When OUTCTL.SDODCS = 1b (I3Cϕ/2)) 100b: 4 I3Cϕ cycles (When OUTCTL.SDODCS = 0b (I3Cϕ)) 7 or 8 I3Cϕ cycles (When OUTCTL.SDODCS = 1b (I3Cϕ/2)) 101b: 5 I3Cφ cycles (When OUTCTL.SDODCS = 0b (I3Cφ)) 9 or 10 I3Cϕ cycles (When OUTCTL.SDODCS = 1b (I3Cϕ/2)) 110b: 6 I3Cφ cycles (When OUTCTL.SDODCS = 0b (I3Cϕ)) 11 or 12 I3Cϕ cycles (When OUTCTL.SDODCS = 1b (I3Cϕ/2)) 111b: 7 I3Cφ cycles (When OUTCTL.SDODCS = 0b (I3Cφ)) 13 or 14 I3Cϕ cycles (When OUTCTL.SDODCS = 1b (I3Cϕ/2))
7 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	EXCYC	0h	RW	Extra SCL30 Clock Cycle Output*3 The EXCYC bit is cleared automatically after one clock cycle is output. 0b: Does not output an extra SCL30 clock cycle (default). 1b: Outputs an extra SCL30 clock cycle.
3	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	SOCWP	0h	W	SCL30/SDA30 Output Control Write Protect*1 0b: Bits SCOC and SDOC are protected. 1b: Bits SCOC and SDOC can be written (When writing simultaneously with the value of the target bit). This bit is read as 0b.
1	SCOC	1h	RW	SCL30 Output Control*1 High level output is achieved through an external pull-up resistor. 0b: I3C drives the SCL30 pin low. 1b: I3C releases the SCL30 pin.
0	SDOC	1h	RW	SDA30 Output Control*1 0b: I3C drives the SDA30 pin low. 1b: I3C releases the SDA30 pin.

Note 1. This bit supports I^2C, I3C master, and I3C secondary master mode.

Note 2. These bits support I^2C mode.

Note 3. This bit supports I^2C mode.

Note 4. The setting SDODCS = 1b (I3Cϕ/2) only becomes valid when SCL is at the low level. When SCL is at the high level, the setting SDODCS = 1b becomes invalid and the clock source becomes the internal reference clock (I3Cϕ).

SDOC bit (SDA Output Control) and SCOC bit (SCL Output Control)

These bits are used to directly control the SDA and SCL signals output from this module.

When writing to these bits, also write 1b to the SOCWP bit at the same time.

The result of setting these bits is input to I3C via the input buffer. When slave mode is selected, a START condition may be detected and the bus may be released depending on the bit settings.

Do not rewrite these bits during a START condition, STOP condition, Repeated START condition, or during transmission or reception. Operation after rewriting under the above conditions is not guaranteed.

EXCYC bit (Extra SCL Clock Cycle Output)

This bit is used to output an extra SCL clock cycle for debugging or error processing.

Normally, set the bit to 0b. Setting the bit to 1 in a normal communication state causes a communication error.

For details on this function, see **(10) Port Control**, **(a) Extra SCL Clock Cycle Output Function**.

7.8.3.2.21 Input Control Register (I3Cm_INCTL)

Access Size : 32 bits
Address : <I3C_base> + 008Ch
Initial Value : 0000_00D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	DNFE	DNFS[3:0]			
Initial Value	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 6	-	All 1	RW	Reserved Whenever it is read, 1b is read. The written value should always be 1b.
5	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	DNFE	1h	RW	Digital Noise Filter Circuit Enable 0b: No digital noise filter circuit is used. 1b: A digital noise filter circuit is used.
3 to 0	DNFS[3:0]	0h	RW	Digital Noise Filter Stage Selection 0h: Noise of up to one I3Cϕ cycle is filtered out (single-stage filter). 1h: Noise of up to two I3Cϕ cycles is filtered out (2-stage filter). 2h: Noise of up to three I3Cϕ cycles is filtered out (3-stage filter). 3h: Noise of up to four I3Cϕ cycles is filtered out (4-stage filter). 4h: Noise of up to five I3Cϕ cycles is filtered out (5-stage filter). ⋮ Fh: Noise of up to sixteen I3Cϕ cycles is filtered out (16-stage filter).

Note: This register supports I^2C mode.

DNFS[3:0] bits (Digital Noise Filter Stage Selection)

These bits are used to select the number of stages in the digital noise filter.

For details on the digital noise filter function, see **(3) Digital Noise-Filter Circuits [I2C mode]**.

In I^2C High Speed mode, the module changes the number of noise filter stage to a quarter of the number of noise filter stage automatically.

Note: Set the noise range to be filtered out by the noise filter within a range less than the SCL line high-level period or low-level period. If the noise range is set to a value of (SCL clock width: high-level period or low-level period, whichever is shorter) - [1.5 internal reference clock (I3Cϕ) cycles] or more, the SCL clock is regarded as noise by the noise filter function of I3C, which may prevent I3C from operating normally.

Note: In I^2C High Speed mode, the lower 2 bits of the DNFS [3:0] bits are ignored, and the number of filter stages for 1 to 4 stages is selected by the upper 2 bits.

7.8.3.2.22 Timeout Control Register (I3Cm_TMOCTL)

Access Size : 32 bits
Address : <I3C_base> + 0090h
Initial Value : 0000_0030h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TOMDS[1:0]		TOHCTL	TOLCTL	-	-	TODTS[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7, 6	TOMDS[1:0]	0h	RW	Timeout Operation Mode Selection 00b: Timeout is detected during the following conditions: The bus is busy (BCST.BFREF = 0) in master mode. I3C's own slave address is detected and the bus is busy in slave mode. The bus is free (BCST.BFREF = 1) while generation of a START condition is requested (CNDCTL.STCND = 1). 01b: Timeout is detected while the bus is busy. 10b: Timeout is detected while the bus is free. 11b: Setting prohibited
5	TOHCTL	1h	RW	Timeout H Count Control 0b: Count is disabled while the SCL30 line is at a high level. 1b: Count is enabled while the SCL30 line is at a high level.
4	TOLCTL	1h	RW	Timeout L Count Control 0b: Count is disabled while the SCL30 line is at a low level. 1b: Count is enabled while the SCL30 line is at a low level.
3, 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1, 0	TODTS[1:0]	0h	RW	Timeout Detection Time Selection 00b: 16-bit timeout 01b: 14-bit timeout 10b: 8-bit timeout 11b: 6-bit timeout

TODTS[1:0] bits (Timeout Detection Time Selection)

These bits are used to select for the timeout detection time when the timeout function is enabled (BSTE.TODE bit = 1b).

When these bits are set to 00b, the timeout detection internal counter functions as a 16-bit counter.

When these bits are set to 01b, the counter functions as a 14-bit counter.

When these bits are set to 10b, the counter functions as an 8-bit counter.

When these bits are set to 11b, the counter functions as a 6-bit counter.

While the SCL line is in the state that enables this counter as specified by bits TOHCTL and TOLCTL, the counter counts up in synchronization with the internal reference clock (I3Cϕ) as a count source.

For details on the timeout function, see **(3) Timeout Error Detection**.

TOLCTL bit (Timeout L Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL line is held low when the timeout function is enabled (BSTE.TODE = 1b).

TOHCTL bit (Timeout H Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL line is held high when the timeout function is enabled (BSTE.TODE = 1b).

TOMDS[1:0] bits (Timeout Operation Mode Selection)

These bits are used to select the detection condition for timeout when the timeout function is enabled.

7.8.3.2.23 Wake-Up Unit Control Register (I3Cm_WUCTL)

Access Size : 32 bits
Address : <I3C_base> + 0098h
Initial Value : 0000_0041h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	WUFE	WUFSYNE	-	WUANFS	-	-	-	WUACKS
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7	WUFE	0h	RW	Wake-Up Function Enable 0b: Wake-up function disabled 1b: Wake-up function enabled Do not set WUFE = 0 during wake-up operation.
6	WUFSYNE	1h	RW	Wake-Up Function Synchronous Enable 0b: I3C asynchronous circuit enabled 1b: I3C synchronous circuit enabled
5	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	WUANFS	0h	RW	Wake-Up Analog Noise Filter Selection*1 0b: Do not add the Wake-Up analog filter. 1b: Add the Wake-Up analog filter.
3 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	WUACKS	1h	RW	Wake-Up Acknowledge Selection*1 Choice of four response modes with a combination of RSTCTL.INTLRST bit and WUACKS bit. Shown in the **Table 7.8-7**.

Note 1. This bit supports I^2C mode.

Table 7.8-7 Wake-Up Mode

INTLRST	WUACKS	Operation Mode	Description
0	0	Normal Wake-up mode 1	ACK response at 9th SCL and SCL low-hold after at 9th SCL.
0	1	Normal Wake-up mode 2	No ACK response immediately and SCL low-hold between 8th and 9th SCL. Release SCL low-hold and ACK response at 9th SCL.
1	0	Command recovery mode	ACK response at 9th SCL and not SCL low-hold.
1	1	EEP response mode	NACK response at 9th SCL and not SCL low-hold.

Note: In Wake-up mode 2, HS mode cannot be used.

WUFSYNE bit (Wake-Up Function Synchronous Enable)

This bit is used to switch between the I3C_0_PCLK synchronous operation and the I3C_0_PCLK asynchronous operation.

The bit is used in combination with the WUASYNF flag when Wake-up function is effective (WUCTL.WUFE bit =1b).

- When switching from the I3C_0_PCLK synchronous operation to the I3C_0_PCLK asynchronous operation:

- I3C operation changes into the I3C_0_PCLK asynchronous operation while the BCST.BFREF flag = 1, when the WUASYNF flag set to 1 while the WUFSYNE bit = 0b.
- The reception can operate without depending on the state of operation of I3C_0_PCLK (with I3C_0_PCLK stopped) after it switches to the I3C_0_PCLK asynchronous operation (Wake-up event detection operation).

- When switching from the I3C_0_PCLK asynchronous operation to the I3C_0_PCLK synchronous operation:
  - I3C operation changes into the I3C_0_PCLK synchronous operation at the following conditions. (At the same timing when the WUASYNF flag becomes 0)
  - In the case Wake-up event is detected: right after the WUFSYNE bit is set to 1b.
  - In the case Wake-up event is not detected: when STOP condition is detected after the WUFSYNE bit is set to 1b.

[Setting conditions]

- When 1b is written to the WUFSYNE bit.
- WUCTL.WUFE=0b

[Clearing condition]

- When 0b is written to the WUFSYNE bit.

7.8.3.2.24 Acknowledge Control Register (I3Cm_ACKCTL)

Access Size : 32 bits
Address : <I3C_base> + 00A0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	ACKTWP	ACKT	ACKR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	W	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	ACKTWP	0h	W	ACKT Write Protect 0b: The ACKT bit are protected. 1b: The ACKT bit can be written (when writing simultaneously with the value of the target bit). This bit is read as 0b.
1	ACKT	0h	RW	Acknowledge Transmission 0b: A 0 is sent as the acknowledge bit (ACK transmission). 1b: A 1 is sent as the acknowledge bit (NACK transmission).
0	ACKR	0h	R	Acknowledge Reception 0b: A 0 is received as the acknowledge bit (ACK reception). 1b: A 1 is received as the acknowledge bit (NACK reception).

Note: This register supports I^2C mode.

ACKR bit (Acknowledge Reception)

This bit is used to store the acknowledge bit information received from the receive device in transmit mode.

[Setting condition]

- When 1b is received as the acknowledge bit with the PRSST.TRMD bit set to 1b.

[Clearing condition]

- When 0b is received as the acknowledge bit with the PRSST.TRMD bit set to 1b.

ACKT bit (Acknowledge Transmission)

[Setting condition]

- When 1b is written to the ACKT bit and 1b is written to the ACKTWP bit at the same time.

[Clearing conditions]

- When 0b is written to the ACKT bit and 1b is written to the ACKTWP bit at the same time.
- When a STOP condition is detected. (when a STOP condition is detected with the CNDCTL.SPCND bit set to 1b.)

Note: Set the ACKT bit to 0b in I^2C Slave mode.

ACKTWP bit (ACKT Write Protect)

This bit is used to control the modification of the ACKT bit.

When changing the ACKT bit, setting this bit to 1b at the same time can change the ACKT bit.

When this bit is read, 0b is always read.

7.8.3.2.25 SCL Stretch Control Register (I3Cm_SCSTRCTL)

Access Size : 32 bits
Address : <I3C_base> + 00A4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RWE	ACKTWE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
1	RWE	0h	RW	Receive Wait Enable 0b: No WAIT (The period between ninth clock cycle and first clock cycle is not held low.) 1b: WAIT (The period between ninth clock cycle and first clock cycle is held low.) Low-hold is released by reading NTDTBP0.
0	ACKTWE	0h	RW	Acknowledge Transmission Wait Enable 0b: NTST.RDBFF0 is set at the rising edge of the ninth SCL30 clock cycle. (The SCL30 line is not held low at the falling edge of the eighth clock cycle.) 1b: NTST.RDBFF0 is set at the rising edge of the eighth SCL30 clock cycle. (The SCL30 line is held low at the falling edge of the eighth clock cycle.) Low-hold is released by writing a value to the ACKCTL.ACKT bit.

Note: This register supports I^2C mode.

ACKTWE bit (Acknowledge Transmission Wait Enable)

This bit is used to select the NTST.RDBFF0 flag set timing in receive mode and also to select whether to hold the SCL line low at the falling edge of the eighth SCL clock cycle.

When ACKTWE = 0b, the SCL line is not held low at the falling edge of the eighth SCL clock cycle, and the NTST.RDBFF0 flag is set to 1 at the rising edge of the ninth SCL clock cycle.

When ACKTWE = 1b, the NTST.RDBFF0 flag is set to 1b at the rising edge of the eighth SCL clock cycle and the SCL line is held low at the falling edge of the eighth SCL clock cycle. The low-hold of the SCL line is released by writing a value to the ACKCTL.ACKT bit.

After data is received with this setting, the SCL line is automatically held low before the acknowledge bit is sent. This enables processing to send ACK (ACKCTL.ACKT = 0b) or NACK (ACKCTL.ACKT = 1b) according to receive data.

RWE bit (Receive Wait Enable)

This bit is used to control whether to hold the period between the ninth SCL clock cycle and the first SCL clock cycle low until the receive data buffer (NTDTBP0) is completely read each time single-byte data is received in receive mode.

When RWE = 0b, the receive operation is continued without holding the period between the ninth and the first SCL clock cycle low. When both the ACKTWE and RWE bits = 0b, continuous receive operation is enabled with the double buffer.

When RWE = 1b, the SCL line is held low from the falling edge of the ninth clock cycle until the NTDTBP0 value is read each time single-byte data is received.

This enables receive operation in byte units.

Note: When the value of the RWE bit is to be read, be sure to read the NTDTBP0 beforehand.

7.8.3.2.26 SCL Stalling Control Register (I3Cm_SCSTLCTL)

When setting this register, follow Chapter 5.1.2.5 Master Clock Stalling of MIPI I3C Spec V1.0, and use it only when necessary because of its negative impacts on bus performance.

Access Size : 32 bits
Address : <I3C_base> + 00B0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ACKPE	PARPE	-	AAPE	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STLCYC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	ACKPE	0h	RW	ACK phase Enable Stall enable bit during ACK/NACK phase 0b: Does not stall the SCL30 clock during the ACK/NACK phase. 1b: Stall the SCL30 clock during the ACK/NACK phase.
30	PARPE	0h	RW	Parity Phase Enable Stall enable bit in parity bit period 0b: Does not stall the SCL30 clock during the parity bit period. 1b: Stall the SCL30 clock during the parity bit period.
29	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
28	AAPE	0h	RW	Assigned Address Phase Enable Enable bit that allows stall by the first bit at address assignment 0b: Does not stall the SCL30 clock during the address assignment phase. 1b: Stall the SCL30 clock during address assignment phase.
27 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 0	STLCYC[15:0]	0h	RW	Stalling Cycle Counter setting of stall period (I3Cϕ cycle). Common use for each phase.

Note: This register supports I3C master mode and I3C secondary master mode.

STLCYC[15:0] bits (Stalling Cycle)

These bits set the SCL stall period. The SCL stall period is counted by the internal reference clock (I3Cϕ). This is a counter common to the enable bits of each phase.

AAPE bit (Assigned Address Phase Enable)

The master can stall SCL during the low period of the first bit of the assigned address phase of the Enter Dynamic Address Assignment CCC command. It can gain time in assigning dynamic address to the device based on the BCR and DCR of the slave. However, because the Dynamic Address Assignment procedure sends the dynamic address set in the DATBASm register in sequence, it is not necessary to set this bit and it is prohibited.

PARPE bit (Parity Phase Enable)

The parity bit of the transmission data of I3C write transfer can be used for SCL stalling to avoid underrun of the transmission data FIFO. However, when the transmission data FIFO of the I3C master becomes empty, SCL stalling is

performed regardless of the setting of this bit, it is not necessary to set this bit and it is prohibited. It is necessary to set this bit when the I3C slave requires preparation time to receive data.

ACKPE bit (ACK phase Enable)

Determine the need to perform SCL stalling in the ACK/NACK phase based on the following criteria:

- It is necessary to set this bit when the I3C and I^2C slaves connected to the bus require preparation time to receive or transmit data.
- In legacy I^2C communication, if there is a possibility that the data FIFO of the I3C master might underrun or overflow, it is not necessary to set this bit because SCL Stalling is performed by FIFO Empty or Full regardless of the setting of this bit.
- Other than legacy I^2C communication, the data FIFO of I3C master might underrun or overflow, and if SCL stalling is required in ACK phase, this bit can be set. However, it is necessary to build the software so that the FIFO does not underrun or overflow due to the interrupt generated according to the FIFO threshold setting (NQTHCTL, NTBTHCTL, NRQTHCTL).
- When I3C master responds ACK/NACK to IBI, it is not necessary to set this bit because ACK/NACK response can be set in advance by BCTL.HJACK, DATBASm.DVMRRJ and DATBASm.DVS IRRJ.
- It is necessary to set this bit when the I3C slave connected to the bus requires preparation time to transmit data for Direct GET CCC.

7.8.3.2.27 Slave Transfer Data Length Register (I3Cm_SVTDLG0)

Access Size : 32 bits

Address : <I3C_base> + 00C0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	STDLG[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	STDLG[15:0]	0h	RW	Slave Transfer Data Length Indicates the number of bytes to be transferred.
15 to 0	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Note: This register supports I3C secondary master mode and I3C slave mode.

7.8.3.2.28 Synchronous Timing Control Register (I3Cm_STCTL)

Access Size : 32 bits

Address : <I3C_base> + 0120h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STOE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	STOE	0h	RW	Synchronous Timing Output Enable 0b: Disable 1b: Enable

Note: This register supports all I3C mode.

7.8.3.2.29 Asynchronous Timing Control Register (I3Cm_ATCTL)

Access Size : 32 bits

Address : <I3C_base> + 0124h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDIV[7:0]								-	-	-	-	-	AMEOE	MREFOE	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 8	CDIV[7:0]	0h	RW	I3C_0_TCLK Counter Divide Setting*1
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	AMEOE	0h	RW	Additional Master-initiated bus Event Output Enable*2 0b: Disable 1b: Enable
1	MREFOE	0h	RW	MREF Output Enable (Capture Event / Counter Overflow)*2 0b: Disable 1b: Enable
0	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Note 1. This bit supports all I3C mode.

Note 2. This bit supports I3C master mode and I3C secondary master mode.

7.8.3.2.30 Asynchronous Timing Trigger Register (I3Cm_ATTRG)

Access Size : 32 bits

Address : <I3C_base> + 0128h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ATSTRG
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	W	Reserved The written value should always be 0b.
0	ATSTRG	0	W	The read value is undefined Asynchronous Timing Software Trigger 0b Write: Do nothing 1b Write: Software trigger (one-shot pulse) output This bit is always read as 0b

Note: This register supports I3C secondary master mode and I3C slave mode.

7.8.3.2.31 Asynchronous Timing Control Counter Enable Register (I3Cm_ATCCNTE)

Access Size : 32 bits

Address : <I3C_base> + 012Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ATCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	ATCE	0h	RW	Asynchronous Timing Counter Enable for MREF, MC2, SC1, SC2. 0b: Disable 1b: Enable

Note: This register supports all I3C mode.

7.8.3.2.32 Condition Control Register (I3Cm_CNDCTL)

Access Size : 32 bits
Address : <I3C_base> + 0140h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SPCND	SRCND	STCND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	SPCND	0h	RW	STOP (P) Condition Issuance 0b: Does not request to issue a STOP condition. 1b: Requests to issue a STOP condition.
1	SRCND	0h	RW	Repeated START (Sr) Condition Issuance 0b: Does not request to issue a Repeated START condition. 1b: Requests to issue a Repeated START condition.
0	STCND	0h	RW	START (S) Condition Issuance 0b: Does not request to issue a START condition. 1b: Requests to issue a START condition.

Note: This register supports I^2C mode.

STCND bit (START (S) Condition Issuance)

This bit is used to request transition to master mode and issuance of a START condition.

For details on the START condition issuance, see **(3) START Condition / Repeated START Condition / STOP Condition Issuing Function**.

[Setting condition]

- When 1b is written to the STCND bit

[Clearing conditions]

- When 0b is written to the STCND bit
- When a START condition has been issued (A START condition is detected)
- When the BST.ALF (arbitration-lost) flag is set to 1b

Note: Set the STCND bit to 1b (START condition issuance request) when the BCST.BFREF flag is set to 1b (bus free state). Note that arbitration may be lost due to a START condition issuance error if the STCND bit is set to 1b (START condition issuance request) when the BFREF flag is set to 0b (bus busy state).

SRCND bit (Repeated START (Sr) Condition Issuance)

This bit is used to request that a Repeated START condition be issued in master mode.

When this bit is set to 1b to request to issue a Repeated START condition, a Repeated START condition is issued when the BFREF flag is set to 0b (bus busy state) and the PRSST.CRMS bit is set to 1b (master mode).

For details on the Repeated START condition issuance, see **(3) START Condition / Repeated START Condition / STOP Condition Issuing Function**.

[Setting condition]

- When 1b is written to the SRCND bit with the BCST.BFREF flag set to 0b

[Clearing conditions]

- When 0b is written to the SRCND bit
- When a Repeated START condition has been issued (A Repeated START condition is detected)
- When the BST.ALF (arbitration-lost) flag is set to 1b

Note: Do not set the SRCND bit to 1b while issuing a STOP condition.
If 1b (requests to issue a Repeated START condition) is written to the SRCND bit in slave mode, the Repeated START condition is not issued but the SRCND bit remains set to 1b.
If the operating mode changes to master mode with the bit not being cleared, note that the Repeated START condition may be issued.

SPCND bit (STOP (P) Condition Issuance)

This bit is used to request that a STOP condition be issued in master mode.

When this bit is set to 1b to request to issue a STOP condition, a STOP condition is issued when the BCST.BFREF flag is set to 0b (bus busy state) and the PRSST.CRMS bit is set to 1b (master mode).

For details on the STOP condition issuance, see **(3) START Condition / Repeated START Condition / STOP Condition Issuing Function**.

[Setting condition]

- When 1b is written to the SPCND bit with the BCST.BFREF flag set to 0b and the PRSST.CRMS bit set to 1b

[Clearing conditions]

- When 0b is written to the SPCND bit
- When a STOP condition has been issued (A STOP condition is detected)
- When the BST.ALF (arbitration-lost) flag is set to 1b
- When a START condition and a Repeated START condition are detected

Note: Writing to the SPCND bit is not possible while the setting of the BCST.BFREF flag = 1b (bus free state).
Do not set the SPCND bit to 1b while a Repeated START condition is being issued.

7.8.3.2.33 Normal Command Queue Port Register (I3Cm_NCMDQP)

32-bit mailbox register NCMDQP contains a command descriptor structure that depends on the requested transfer type:

- Address Assignment Command (see **(1) Address Assign Command**)
- Immediate Data Transfer (see **(2) Immediate Transfer Command**)
- Regular Data Transfer (see **(3) Regular Transfer Command**)
- Write + Write/Read Combo Transfer (see **(4) Combo Transfer Command**)
- Internal Control Command (see **(5) Internal Control Command**)

Within the command descriptor, DWORDs appear starting with the Least Significant DWORD, in order until the Most Significant DWORD.

Access Size : 32 bits
Address : <I3C_base> + 0150h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NCMDQP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCMDQP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NCMDQP[31:0]	-	W	Normal Command Queue Port

Note: This register supports all I3C mode.

7.8.3.2.34 Normal Response Queue Port Register (I3Cm_NRSPQP)

32-bit mailbox register NRSPQP contains a response structure (see **7.8.4.1.4 Receive Status Descriptor**).

Access Size : 32 bits
Address : <I3C_base> + 0154h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NRSPQP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NRSPQP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NRSPQP[31:0]	0h	R	Normal Response Queue Port

Note: This register supports all I3C mode.

7.8.3.2.35 Normal Transfer Data Buffer Port Register (I3Cm_NTDTBP0/NTDTBP0_BY)

32-bit mailbox register NTDTBP0 is a 32-bit bi-directional data transfer register which is used both to read from the Normal Receive Data Buffer, and to write to the Normal Transmit Data Buffer.

In other words, the Normal Receive Data Buffer and the Normal Transmit Data Buffer have the same offset, forming a single bidirectional port for transmitting or receiving I3C data.

Access Size : 32 bits
Address : <I3C_base> + 0158h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NTDTBP0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NTDTBP0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NTDTBP0[31:0]	0h	RW	Normal Transfer Data Buffer Port NTDTBP0 is a 32-bit read/write register. NTDTBP0_BY (NTDTBP0[7:0]) is a 8-bit read/write register.

Note: NTDTBP0 is for 32-bit access in I3C mode.
NTDTBP0_BY is for 8-bit access in I^2C mode.

Read Operations:

[I3C protocol mode]

Data Read from the Normal Receive Data Buffer. It should be read based on Normal Queue Status Level indications. The Receive data is always aligned to a 4-byte boundary, and stored in the Normal Receive Data Buffer. If the length of the data transfer is not aligned to a 4-byte boundary, then there will be extra (unused) bytes at the end of the transferred data. The valid data must be identified using the DATA_LENGTH field in the Response Descriptor.

[I^2C protocol mode]

When 1 byte of data has been received, the received data is transferred from the internal shift register to NTDTBP0 to enable the next data to be received. The double-buffer structure of the internal shift register and NTDTBP0 allows continuous receive operation if the received data has been read from NTDTBP0 while the internal shift register is receiving data. Read data from NTDTBP0 once when a receive data full interrupt (INT_ri3c_rx_n) request is generated. If NTDTBP0 receives the next receive data before the current data is read from NTDTBP0 (while the RDBFF0 flag in NTST is 1b), this module automatically holds the SCL clock low one cycle before the RDBFF0 flag is set to 1b next. The lower 8 bits of the read 32-bit data are valid as received data.

Write Operations:

[I3C protocol mode]

Data Written to the Normal Transmit Data Buffer. Data DWORDs written to the Normal Transmit Data Buffer are placed onto the I3C bus one byte at a time, with the DWORD LSB first. Within each byte, bits are placed onto the I3C bus in big-endian order, with bit 7 going out first on the bus. The transmit data should always start aligned to a 4-byte

boundary, and written to the NTDTBP0 register. If the length of the transfer is not aligned to a 4-byte boundary, then there will be extra (unused) bytes at the end of the transferred data. I3C shall only send the valid number of bytes indicated in the DATA_LENGTH field of the Command Descriptor.

[I^2C protocol mode]

When NTDTBP0 detects a space in the internal shift register, it transfers the transmit data that has been written to NTDTBP0 to the internal shift register and starts transmitting data in transmit mode. The double-buffer structure of NTDTBP0 and the internal shift register allows continuous transmit operation if the next transmit data has been written to NTDTBP0 while the internal shift register data is being transmitted. Write transmit data to NTDTBP0 once when a transmit data empty interrupt (INT_ri3c_tx_n) request is generated. The lower 8 bits of the written 32-bit data are valid as transmission data.

7.8.3.2.36 Normal IBI Queue Port Register (I3Cm_NIBIQP)

Access Size : 32 bits
Address : <I3C_base> + 017Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NIBIQP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NIBIQP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NIBIQP[31:0]	0h	RW	Normal IBI Queue Port

Note: This register supports all I3C mode.

When receiving an IBI, 32-bit mailbox register NIBIQP is used for both:

- Read the IBI status descriptor (see **7.8.4.1.3 IBI Status Descriptor**)
- Read the IBI data (which is raw/opaque data).

The IBI status descriptor is a read-only structure describing an IBI event received from a Slave device on the I3C bus.

Remark If the I3C HCI auto-read feature is used, then the IBI data includes the data received from the auto-generated private read operation.
Even if LAST_STATUS is set to 0b, the driver software still evaluates the data payload length by examining the CHUNKS field.

7.8.3.2.37 Normal Receive Status Queue Port Register (I3Cm_NRSQP)

32-bit mailbox register NRSQP contains a receive status structure (see **7.8.4.1.4 Receive Status Descriptor**).

Access Size : 32 bits
Address : <I3C_base> + 0180h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NRSQP[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NRSQP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NRSQP[31:0]	0h	R	Normal Receive Status Queue Port

Note: This register supports I3C secondary master mode and I3C slave mode.

7.8.3.2.38 Normal Queue Threshold Control Register (I3Cm_NQTHCTL)

The Queue Threshold Control register controls the interrupt trigger thresholds for the Command Queue, the Response Queue, and the IBI Queue.

Access Size : 32 bits
Address : <I3C_base> + 0190h
Initial Value : 0101_0101h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	IBIQTH[2:0]			IBIDSSZ[7:0]							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	RSPQTH[1:0]		-	-	-	-	-	-	CMDQTH[1:0]	
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
26 to 24	IBIQTH[2:0]	1h	RW	Normal IBI Queue Threshold*1 000b: I3C Protocol mode (Master): Interrupt is generated when the Outstanding IBI Status count is 1 or more. I3C Protocol mode (Slave): Interrupt is issued when IBI Data Buffer is completely empty. Others: I3C Protocol mode (Master): Interrupt is generated when the Outstanding IBI Status count is N + 1 or more. (N = IBIQTH[2:0]) I3C Protocol mode (Slave): Interrupt is issued when IBI Data Buffer contains N empties. (N = IBIQTH[2:0])
23 to 16	IBIDSSZ[7:0]	1h	RW	Normal IBI Data Segment Size*2 Supported Values: Minimum: 1 (4 bytes) Maximum: 63 (252 bytes), provided that the configured IBI Queue depth is 64 or more. When ATCCNTE.ATCE = 1, restrict to the number of segments is 2 or more.
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9, 8	RSPQTH[1:0]	1h	RW	Normal Response Queue Threshold*1 00b: Interrupt is issued when Response Queue contains 1 entry (DWORD). 01b: Interrupt is issued when Response Queue contains 2 entries (DWORD). 10b: Interrupt is issued when Response Queue contains 3 entries (DWORD). 11b: Interrupt is issued when Response Queue contains 4 entries (DWORD).
7 to 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1, 0	CMDQTH[1:0]	1h	RW	Normal Command Ready Queue Threshold*1 00b: Interrupt is issued when Command Queue is completely empty. 01b: Interrupt is issued when Command Queue contains 1 empty. 10b: Interrupt is issued when Command Queue contains 2 empties. 11b: Interrupt is issued when Command Queue contains 3 empties.

Note: This register supports I3C secondary master mode and I3C slave mode.

Note 1. These bits support all I3C mode.

Note 2. These bits support I3C master mode and I3C secondary master mode.

CMDQTH[1:0] bits (Normal Command Ready Queue Threshold)

Controls the minimum number of Command Queue empties needed to trigger the INTCMD interrupt.

RSPQTH[1:0] bits (Normal Response Queue Threshold)

Controls the minimum number of Response Queue entries needed to trigger the INTRESP interrupt.

IBIDSSZ[7:0] bits (Normal IBI Data Segment Size)

This is the IBI data segment size, in DWORDs (4 bytes).

In PIO mode, this field allows the incoming IBI data to be sliced into multiple segments generating status individually, to support cutthrough readout of a long IBI payload data.

When Asynchronous Timing Control mode is supported, this field should be set to a value other than 1 or 3 to allow the single data segment to contain the entire Master time-stamp value (i.e., both MREF and MC2).

IBIQTH[2:0] bits (Normal IBI Queue Threshold)

For I3C protocol mode (Master): PRTS.PRTMD = 0b and PRSST.CRMS = 1b.

Controls generation of the INTIBI interrupt, based on the value of the IBI Queue’s Outstanding IBI status count.

Each IBI status entry can represent either the complete IBI payload (if the IBI payload byte size is 4×IBIDSSZ or less), or a segment of the IBI payload (if the IBI payload byte size is more than 4 × IBIDSSZ).

For I3C protocol mode (Slave): PRTS.PRTMD bit = 0b, PRSST.CRMS bit = 0b.

Controls the minimum number of IBI Data Buffer empties needed to trigger the INTIBI interrupt.

7.8.3.2.39 Normal Transfer Data Buffer Threshold Control Register (I3Cm_NTBTHCTL0)

The Data Buffer Control register controls the interrupt trigger thresholds for the Receive Data Buffer Queue and the Transmit Data Buffer Queue.

Access Size : 32 bits
Address : <I3C_base> + 0194h
Initial Value : 0101_0101h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	RXSTTH[2:0]			-	-	-	-	-	TXSTTH[2:0]		
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	RXDBTH[2:0]			-	-	-	-	-	TXDBTH[2:0]		
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
26 to 24	RXSTTH[2:0]	1h	RW	Normal Receive Start Threshold*2 000b: Wait for 2 empty DWORDs 001b: Wait for 4 empty DWORDs 010b: Wait for 8 empty DWORDs 011b: Wait for 16 empty DWORDs Others: Setting prohibited
23 to 19	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18 to 16	TXSTTH[2:0]	1h	RW	Normal Transmit Start Threshold*2 000b: Wait for 2 entry DWORDs 001b: Wait for 4 entry DWORDs 010b: Wait for 8 entry DWORDs 011b: Wait for 16 entry DWORDs Others: Setting prohibited
15 to 11	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10 to 8	RXDBTH[2:0]	1h	RW	Normal Receive Data Buffer Threshold*1 000b: Interrupt triggers at 2 Receive Buffer entries, in DWORDs 001b: Interrupt triggers at 4 Receive Buffer entries, in DWORDs 010b: Interrupt triggers at 8 Receive Buffer entries, in DWORDs Others: Setting prohibited
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	TXDBTH[2:0]	1h	RW	Normal Transmit Data Buffer Threshold*1 000b: Interrupt triggers at 2 Transmit Buffer empties, in DWORDs 001b: Interrupt triggers at 4 Transmit Buffer empties, in DWORDs 000b: Interrupt triggers at 8 Transmit Buffer empties, in DWORDs Others: Setting prohibited

Note 1. These bits support all I3C mode.

Note 2. These bits support I3C master mode and I3C secondary master mode.

TXDBTH[2:0] bits (Normal Transmit Data Buffer Threshold)

Minimum number of Transmit FIFO empties, in DWORDs, that will trigger the INT_ri3c_tx_n interrupt.

RXDBTH[2:0] bits (Normal Receive Data Buffer Threshold)

Minimum number of Receive FIFO entries in DWORDs that will trigger the INT_ri3c_rx_n interrupt.

TXSTTH[2:0] bits (Normal Transmit Start Threshold)

When preparing to initiate a Write Transfer on the I3C Bus, I3C shall wait until the Transmit Buffer has at least the indicated number of locations available.

Two optional configurable Modes are available:

(1) Store and Forward Mode
If the TXSTTH[2:0] field is set to the Transmit Buffer size (011b), then I3C shall delay initiation of the Write Command as follows:

- If the data length to be transferred is more than the Transmit Buffer size (16 DWORDs), then this module shall wait until the Transmit FIFO is completely full.
- If the data length to be transferred is less than the Transmit Buffer size (16 DWORDs), then I3C shall wait until enough Transmit FIFO locations are available to store the data to be transferred.

(2) Threshold Mode
If the TXSTTH[2:0] field value is less than the Transmit Buffer size (011b), then I3C shall initiate the Write Command as soon as the indicated number of Transmit FIFO locations are entries.

RXSTTH[2:0] bits (Normal Receive Start Threshold)

When preparing to initiate a Read Transfer on the I3C bus, I3C shall wait until the Receive Buffer has at least the indicated number of empty locations in DWORDs.

Two optional configurable Modes are available:

(1) Store and Forward Mode
If the RXSTTH[2:0] field is set to the Receive Buffer size (011b), then I3C shall delay initiation of the Read Command as follows:

- If the data length to be transferred is more than the Receive Buffer size (16 DWORDs), then this module shall wait until the Receive FIFO is completely empty.
- If the data length to be transferred is less than the Receive Buffer size (16 DWORDs), then I3C shall wait until enough Receive FIFO locations are available to store the data to be transferred.

(2) Threshold Mode
If the RXSTTH[2:0] field value is less than the Receive Buffer size (011b), then I3C shall initiate the Read Command as soon as the indicated number of Receive FIFO locations are empty.

7.8.3.2.40 Normal Receive Status Queue Threshold Control Register (I3Cm_NRQTHCTL)

Access Size : 32 bits

Address : <I3C_base> + 01C0h

Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RSQTH
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	RSQTH	1h	RW	Normal Receive Status Queue Threshold 0b: Interrupt is issued when Receive Status Queue contains 1 entry (DWORD). 1b: Interrupt is issued when Receive Status Queue contains 2 entries (DWORD).

RSQTH bit (Normal Receive Status Queue Threshold)

Controls the minimum number of receive status queue entries needed to trigger the INTRCV interrupt.

7.8.3.2.41 Bus Status Register (I3Cm_BST)

Access Size : 32 bits

Address : <I3C_base> + 01D0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	WUCNDDF	-	-	-	TODF	-	-	-	ALF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TENDF	-	-	-	NACKDF	-	-	SPCNDDF	STCNDDF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
24	WUCNDDF	0h	RW*2	Wake-Up Condition Detection Flag 0b: Wake-Up Condition is not detected. 1b: Wake-Up Condition is detected.
23 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20	TODF	0h	RW*2	Timeout Detection Flag 0b: Timeout is not detected. 1b: Timeout is detected.
19 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	ALF	0h	RW*2	Arbitration Lost Flag*1 0b: Arbitration is not lost 1b: Arbitration is lost.
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8	TENDF	0h	RW*2	Transmit End Flag*1 0b: Data is being transmitted. 1b: Data has been transmitted.
7 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	NACKDF	0h	RW*2	NACK Detection Flag*1 0b: NACK is not detected. 1b: NACK is detected.
3, 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	SPCNDDF	0h	RW*2	STOP Condition Detection Flag 0b: STOP condition is not detected. 1b: STOP condition is detected.
0	STCNDDF	0h	RW*2	START Condition Detection Flag 0b: START condition is not detected. 1b: START condition is detected.

Note 1. This bit supports I^2C mode.

Note 2. Clearing (to 0) condition: Writing 0b after 1b is read.

STCNDDF bit (START Condition Detection Flag)

[Setting conditions]

- All of the followings are satisfied:

1) The BSTE.STCNDDE bit = 1b.
   Thexxx
2) When a START condition (or a Repeated START condition) is detected.

[Clearing conditions]

- When 0b is written to the STCNDDF flag after reading STCNDDF flag = 1b.
- When a STOP condition is detected.

SPCNDDF bit (STOP Condition Detection Flag)

[Setting conditions]

- All of the followings are satisfied:
  1) The BSTE.SPCNDDE bit = 1b.
  2) When a STOP condition is detected.

[Clearing condition]

- When 0b is written to the SPCNDDF flag after reading SPCNDDF flag = 1b.

NACKDF bit (NACK Detection Flag)

[Setting conditions]

- All of the followings are satisfied:
  1) The PRTS.PRTMD bit = 1b (I^2C protocol mode).
  2) The BSTE.NACKDE bit = 1b (Enables NACK detection interrupt status logging).
  3) When acknowledge is not received (NACK is received) from the receive device in transmit mode.

[Clearing condition]

- When 0b is written to the NACKDF flag after reading NACKDF flag = 1b.

TENDF bit (Transmit End Flag)

[Setting conditions]

- All of the followings are satisfied:
  1) The PRTS.PRTMD bit = 1b (I^2C protocol mode).
  2) The BSTE.TENDE bit = 1b (Enables Transmit End Interrupt Status logging).
  3) At the rising edge of the ninth SCL clock cycle while the NTST.TDBEF0 flag = 1b. Excluding when sending an address.

[Clearing conditions]

- When 0b is written to the TENDF flag after reading TENDF flag = 1b.
- When data is written to the NTDTBP0 register.
- When a STOP condition is detected.

ALF bit (Arbitration Lost Flag)

[Setting conditions]

- When master arbitration-lost detection is enabled: BSTE.ALE bit = 1b, BFCTL.MALE = 1b.

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data (including slave address) transmission in master transmit mode (when the SDA line is driven low while the internal SDA output is at a high level (the SDA pin is in the high-impedance state)).
- All of the followings are satisfied.
  1) When the START condition is detected while the CNDCTL.STCND bit = 1b.
  2) When the internal SDA output state does not match the SDA line level.
- When the CNDCTL.STCND bit is set to 1 (START condition issuance request) while the BCST.BFREF flag = 0b.

- When NACK arbitration-lost detection is enabled: BSTE.ALE bit = 1b, BFCTL.NALE = 1b.
  - When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock in the ACK period during NACK transmission in receive mode.
- When slave arbitration-lost detection is enabled: BSTE.ALE bit = 1b, BFCTL.SALE = 1b.
  - When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data transmission in slave transmit mode.

[Clearing condition]

- When 0b is written to the ALF flag after reading ALF flag = 1b.

TODF bit (Timeout Detection Flag)

[Setting conditions]

- All of the followings are satisfied.
  1) The BSTE.TODE bit = 1b (Enables Timeout Detection Interrupt Status logging).
  2) When the master mode or the received slave address matches the slave address n in Slave mode.
  3) When the SCL line state remains unchanged for the period specified by TMOCTL register.

[Clearing condition]

- When 0b is written to the TODF flag after reading TODF flag = 1b.

WUCNDDF bit (Wake-Up Condition Detection Flag)

[Setting conditions]

- For I^2C Protocol mode: PRTS.PRTMD bit = 1b.

  When I3C_0_PCLK and I3C_0_TCLK are supplied after all of the following 1) to 4) are satisfied.
  1) The WUCTL.WUFE bit = 1b (Wake-up function is enabled).
  2) The BSTE.WUCNDDE bit = 1b (Enables Wake-up Condition Detection Status logging).
  3) The WUST.WUASYNF flag = 1b.
  4) When the address received in slave mode matches the address of slave enabled in the SVCTL.SVAEn bit (except for the Device-ID address).
- For I3C Protocol mode (Master): PRTS.PRTMD bit = 0b, PRSST.CRMS bit = 1b.

  When I3C_0_PCLK and I3C_0_TCLK are supplied after all of the following 1) to 4) are satisfied.
  1) The WUCTL.WUFE bit = 1b (Wake-up function is enabled).
  2) The BSTE.WUCNDDE bit = 1b (Enables Wake-up Condition Detection Status logging).

3) The WUST.WUASYNF flag = 1b.

4) When low level of the SDA line is detected (When the START condition is detected).

- For I3C Protocol mode (Slave) : PRTS.PRTMD bit = 0b, PRSST.CRMS bit = 0b.

  When I3C_0_PCLK and I3C_0_TCLK are supplied after all of the following 1) to 4) are satisfied.

  1) The WUCTL.WUFE bit = 1b (Wake-up function is enabled).

  2) The BSTE.WUCNDDE bit = 1b (Enables Wake-up Condition Detection Status logging).

  3) The WUST.WUASYNF flag = 1b.

  4) When the Broadcast Address (7Eh) is detected after a START (or Repeated START) condition and the own Dynamic Address is detected after the Repeated START condition following the Broadcast Address.

[Clearing condition]

- When 0b is written to the WUCNDDF flag after reading WUCNDDF flag = 1b while the WUST.WUASYNF flag = 0b.

7.8.3.2.42 Bus Status Enable Register (I3Cm_BSTE)

Access Size : 32 bits

Address : <I3C_base> + 01D4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	WUCNDDE	-	-	-	TODE	-	-	-	ALE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TENDE	-	-	-	NACKDE	-	-	SPCNDDE	STCNDDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
24	WUCNDDE	0h	RW	Wake-Up Condition Detection Enable 0b: Disables Wake-Up Condition Detection Status logging 1b: Enables Wake-Up Condition Detection Status logging
23 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20	TODE	0h	RW	Timeout Detection Enable 0b: Disables Timeout Detection Interrupt Status logging. 1b: Enables Timeout Detection Interrupt Status logging.
19 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	ALE	0h	RW	Arbitration Lost Enable*1 0b: Disables Arbitration Lost Interrupt Status logging. 1b: Enables Arbitration Lost Interrupt Status logging.
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8	TENDE	0h	RW	Transmit End Enable*1 0b: Disables Transmit End Interrupt Status logging. 1b: Enables Transmit End Interrupt Status logging.
7 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	NACKDE	0h	RW	NACK Detection Enable*1 0b: Disables NACK Detection Interrupt Status logging. 1b: Enables NACK Detection Interrupt Status logging.
3, 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	SPCNDDE	0h	RW	STOP Condition Detection Enable 0b: Disables STOP condition Detection Interrupt Status logging. 1b: Enables STOP condition Detection Interrupt Status logging.
0	STCNDDE	0h	RW	START Condition Detection Enable 0b: Disables START condition Detection Interrupt Status logging. 1b: Enables START condition Detection Interrupt Status logging.

Note 1. This bit supports I^2C mode.

STCNDDE bit (START Condition Detection Enable)

When this bit is 1b, operation of BST.STCNDDF is enabled. For the setting conditions and clearing conditions of the BST.STCNDDF flag, see the details of BST.STCNDDF.

SPCNDDE bit (STOP Condition Detection Enable)

When this bit is 1b, operation of BST.SPCNDDF is enabled. For the setting conditions and clearing conditions of the BST.SPCNDDF flag, see the details of BST.SPCNDDF.

NACKDE bit (NACK Detection Enable)

When this bit is 1b, the operation of BST.NACKDF is enabled. This bit is used to specify whether to continue or discontinue the transfer operation when NACK is received from the slave device in transmit mode. Normally, set this bit to 1b. For the setting conditions and clearing conditions of the BST.NACKDF flag, see the details of BST.NACKDF.

TENDE bit (Transmit End Enable)

When this bit is 1b, the operation of BST.TENDF is enabled. For the setting conditions and clearing conditions of the BST.TENDF flag, see the details of BST.TENDF.

ALE bit (Arbitration Lost Enable)

When this bit is 1b, the operation of BST.ALF is enabled. For the setting conditions and clearing conditions of the BST.ALF flag, see the details of BST.ALF.

TODE bit (Timeout Detection Enable)

When this bit is 1b, the operation of BST.TODF is enabled. For the setting conditions and clearing conditions of the BST.TODF flag, see the details of BST.TODF.

WUCNDDE bit (Wake-Up Condition Detection Enable)

When this bit is 1b, the operation of BST.WUCNDDF is enabled.

For the setting conditions and clearing conditions of the BST.WUCNDDF flag, see the details of BST.WUCNDDF.

7.8.3.2.43 Bus Interrupt Enable Register (I3Cm_BIE)

The BIE register enables signaling of outstanding bus interrupts received by I3C.

Access Size : 32 bits
Address : <I3C_base> + 01D8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	WUCNDDIE	-	-	-	TODIE	-	-	-	ALIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TENDIE	-	-	-	NACKDIE	-	-	SPCNDDIE	STCNDDIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
24	WUCNDDIE	0h	RW	Wake-Up Condition Detection Interrupt Enable 0b: Disables Wake-Up Condition Detection Interrupt Signal. 1b: Enables Wake-Up Condition Detection Interrupt Signal.
23 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20	TODIE	0h	RW	Timeout Detection Interrupt Enable 0b: Disables Timeout Detection Interrupt Signal. 1b: Enables Timeout Detection Interrupt Signal.
19 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	ALIE	0h	RW	Arbitration Lost Interrupt Enable*1 0b: Disables Arbitration Lost Interrupt Signal. 1b: Enables Arbitration Lost Interrupt Signal.
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8	TENDIE	0h	RW	Transmit End Interrupt Enable*1 0b: Disables Transmit End Interrupt Signal. 1b: Enables Transmit End Interrupt Signal.
7 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	NACKDIE	0h	RW	NACK Detection Interrupt Enable*1 0b: Disables NACK Detection Interrupt Signal. 1b: Enables NACK Detection Interrupt Signal.
3, 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	SPCNDDIE	0h	RW	STOP Condition Detection Interrupt Enable 0b: Disables STOP condition Detection Interrupt Signal. 1b: Enables STOP condition Detection Interrupt Signal.
0	STCNDDIE	0h	RW	START Condition Detection Interrupt Enable 0b: Disables START condition Detection Interrupt Signal. 1b: Enables START condition Detection Interrupt Signal.

Note 1. This bit supports I^2C mode.

STCNDDIE bit (START Condition Detection Interrupt Enable)

This bit enables or disables the START Condition Detection interrupt requests when the BST.STCNDDF flag is set to 1.

SPCNDDIE bit (STOP Condition Detection Interrupt Enable)

This bit enables or disables the STOP Condition Detection interrupt requests when the BST.SPCNDDF flag is set to 1b.

NACKDIE bit (NACK Detection Interrupt Enable)

This bit enables or disables the NACK Detection interrupt requests when the BST.NACKDF flag is set to 1b.

TENDIE bit (Transmit End Interrupt Enable)

This bit enables or disables the Transmit End interrupt (I3C_TEND) requests when the BST.TENDF flag is set to 1b.

ALIE bit (Arbitration Lost Interrupt Enable)

This bit enables or disables the Arbitration Lost interrupt requests when the BST.ALF flag is set to 1b.

TODIE bit (Timeout Detection Interrupt Enable)

This bit enables or disables the Timeout Detection interrupt requests when the BST.TODF flag is set to 1b.

WUCNDDIE bit (Wake-Up Condition Detection Interrupt Enable)

This bit is used to enable or disable the Wake-up condition detection interrupt (INTWU) requests when the BST.WUCNDDF flag is set to 1b.

7.8.3.2.44 Bus Status Force Register (I3Cm_BSTFC)

Access Size : 32 bits

Address : <I3C_base> + 01DCh

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	WUCNDDFC	-	-	-	TODFC	-	-	-	ALFC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	W	R	R	R	W	R	R	R	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TENDFC*2	-	-	-	NACKDFC	-	-	SPCNDDFC	STCNDDFC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W	R	R	R	W	R	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
24	WUCNDDFC	0h	W	Wake-Up Condition Detection Force 0b: Not Force Wake-Up Condition Detection Interrupt for software testing. 1b: Force Wake-Up Condition Detection Interrupt for software testing.
23 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	TODFC	0h	W	The read value is undefined Timeout Detection Force 0b: Not Force Timeout Detection Interrupt for software testing. 1b: Force Timeout Detection Interrupt for software testing.
19 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	ALFC	0h	W	The read value is undefined Arbitration Lost Force*1 0b: Not Force Arbitration Lost Interrupt for software testing. 1b: Force Arbitration Lost Interrupt for software testing.
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	TENDFC*2	0h	W	The read value is undefined Transmit End Force*1 0b: Not Force Transmit End Interrupt for software testing. 1b: Force Transmit End Interrupt for software testing.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	NACKDFC	0h	W	The read value is undefined NACK Detection Force*1 0b: Not Force NACK Detection Interrupt for software testing. 1b: Force NACK Detection Interrupt for software testing.
3	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	-	0h	W	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	SPCNDDFC	0h	W	The read value is undefined STOP condition Detection Force 0b: Not Force STOP condition Detection Interrupt for software testing. 1b: Force STOP condition Detection Interrupt for software testing.
0	STCNDDFC	0h	W	The read value is undefined START condition Detection Force 0b: Not Force START condition Detection Interrupt for software testing. 1b: Force START condition Detection Interrupt for software testing.

Note 1. This bit supports I^2C mode.

Note 2. TENDFC does not work unless TDBEF0 = 1b.

7.8.3.2.45 Normal Transfer Status Register (I3Cm_NTST)

Access Size : 32 bits

Address : <I3C_base> + 01E0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	RSQFF	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	TEF	-	-	-	TABTF	RSPQFF	CMDQEF	IBIQEFF	RDBFF0	TDBEF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20	RSQFF	0h	RW*3	Normal Receive Status Queue Full Flag*2 0b: The number of Receive Status Queue entries is less than or equal to the NRQTHCTL.RSQTH threshold. 1b: The number of Receive Status Queue entries is more than the NRQTHCTL.RSQTH threshold.
19 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	TEF	0h	RW*3	Normal Transfer Error Flag*1 0b: Transfer Error does not occur. 1b: Transfer Error occurs. To clear this bit, write 0 after 1 is read.
8 to 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	TABTF	0h	RW*3	Normal Transfer Abort Flag*1 0b: Transfer Abort does not occur. 1b: Transfer Abort occur. To clear this bit, write 0 after 1 is read.
4	RSPQFF	0h	RW*3	Normal Response Queue Full Flag*1 0b: The number of Response Queue entries is less than or equal to the NQTHCTL.RSPQTH threshold. 1b: The number of Response Queue entries is more than the NQTHCTL.RSPQTH threshold.
3	CMDQEF	0h	RW*3	Normal Command Queue Empty Flag*1 0b: If the NQTHCTL.CMDQTH = 0b: The number of Command Queue empties is less than the Command Queue size. If the NQTHCTL.CMDQTH is other than 0b: The number of Command Queue empties is less than the NQTHCTL.CMDQTH threshold. 1b: If the NQTHCTL.CMDQTH = 0b: The number of Command Queue empties is the Command Queue size. If the NQTHCTL.CMDQTH is other than 0b: The number of Command Queue empties is more than or equal to the NQTHCTL.CMDQTH threshold.

Bit	Bit Name	Initial Value	R/W	Description
2	IBIQEFF	0h	RW*3	Normal IBI Queue Empty/Full Flag*1 0b: For I3C protocol mode (Master): PRTS.PRTMD bit = 0b, PRSST.CRMS bit = 1b. The number of IBI Status Queue entries is less than or equal to the NQTHCTL.IBIQTH threshold. For I3C protocol mode (Slave) : PRTS.PRTMD bit = 0b, PRSST.CRMS bit = 0b. If the NQTHCTL.IBIQTH = 0b: The number of IBI Data Buffer empties is less than the IBI Data Buffer size. If the NQTHCTL.IBIQTH is other than 0b: The number of IBI Data Buffer empties is less than the NQTHCTL.IBIQTH threshold. 1b: For I3C protocol mode (Master): PRTS.PRTMD bit = 0, PRSST.CRMS bit = 1b. The number of IBI Status Queue entries is more than the NQTHCTL.IBIQTH threshold. For I3C protocol mode (Slave) : PRTS.PRTMD bit = 0b, PRSST.CRMS bit = 0b. If the NQTHCTL.IBIQTH = 0b: The number of IBI Data Buffer empties is the IBI Data Buffer size. If the NQTHCTL.IBIQTH is other than 0b: The number of IBI Data Buffer empties is more than or equal to the NQTHCTL.IBIQTH threshold.
1	RDBFF0	0h	RW*3	Normal Receive Data Buffer Full Flag*1 0b: For I2C protocol mode: PRTS.PRTMD bit = 1b. Normal Receive Data Buffer contains no receive data. For I3C Protocol mode: PRTS.PRTMD bit = 0b. The number of entries in the Normal Receive Data Buffer is less than the NTBTHCTL0.RXDBTH[2:0] threshold. 1b: For I2C protocol mode: PRTS.PRTMD bit = 1b. Normal Receive Data Buffer contains receive data. For I3C Protocol mode: PRTS.PRTMD bit = 0b. The number of entries in the Normal Receive Data Buffer is more than or equal to the NTBTHCTL0.RXDBTH[2:0] threshold.
0	TDBEF0	0h	RW*3	Normal Transmit Data Buffer Empty Flag*1 0b: For I2C protocol mode: PRTS.PRTMD bit = 1b. Normal Transmit Data Buffer contains transmit data. For I3C protocol mode: PRTS.PRTMD bit = 0b. The number of empties in the Normal Transmit Data Buffer is less than the NTBTHCTL0.TXDBTH[2:0] threshold. 1b: For I2C protocol mode: PRTS.PRTMD bit = 1b. Normal Transmit Data Buffer contains no transmit data. For I3C protocol mode: PRTS.PRTMD bit = 0b. The number of empties in the Normal Transmit Data Buffer is more than or equal to the NTBTHCTL0.TXDBTH[2:0] threshold.

Note 1. This bit supports all I3C mode.

Note 2. This bit supports I3C secondary master mode and I3C slave mode.

Note 3. Clearing (to 0) condition: Writing 0b after 1b is read.

TDBEF0 bit (Normal Transmit Data Buffer Empty Flag)

[Setting conditions]

- For I^2C Protocol mode: PRTS.PRTMD bit = 1b.

  The following condition 1 is satisfied and any of the following conditions 2) to 4) are satisfied:

  1) The NTSTE.TDBEE0 bit = 1b (enables Transmit Data Buffer Empty Interrupt Status logging).
  2) When data has been transferred from the Normal Transmit Data Buffer to the Shift Register and the Normal Transmit Data Buffer becomes empty*1.
  3) When the PRSST.TRMD bit is set to 1b.
  4) 4When the received slave address matches while the TRMD bit = 1b.

- For I3C Protocol mode: PRTS.PRTMD bit = 0b.

  The following conditions 1) and 2) are satisfied:

  1) The NTSTE.TDBEE0 bit = 1b (enables Transmit Data Buffer Empty Interrupt Status logging).

2) When the number of empties in the Normal Transmit Data Buffer is the NTBTHCTL0.TXDBTH[2:0] threshold or more (see NTBTHCTL0 register).

[Clearing conditions]

- For I^2C Protocol mode: PRTS.PRTMD bit = 1b.
  - When data is written to NTDTBP0.
  - When the TRMD bit in PRSST is set to 0b.
- For I3C protocol mode: PRTS.PRTMD bit = 0b.
  - Write 0b to this bit after 1b is read.
  - On completion of the last write access to Normal Transmit Data by DMAC.
  - When the number of empties in the Normal Transmit Data Buffer is less than the NTBTHCTL0.TXDBTH[2:0] threshold (see NTBTHCTL0 register).

Note 1. When the BST.NACKDF flag is set to 1b while the BSTE.NACKDE bit = 1b, I3C aborts data transmission/reception. If the TDBEF0 flag = 0b (next transmit data has been written), data is transferred to the Shift Register and the Normal Transmit Data Buffer register becomes empty at the rising edge of the 9th clock cycle, but the TDBEF0 flag is not set to 1b.

RDBFF0 bit (Normal Receive Data Buffer Full Flag)

[Setting conditions]

- For I^2C Protocol mode: PRTS.PRTMD bit = 1b.

  The following condition 1 is satisfied and any of the following condition 2) or 3) is satisfied:

  1) The NTSTE.RDBFE0 bit = 1b (enables Receive Data Buffer Full Interrupt Status logging).
  2) When receive data is transferred from Shift Register to Normal Receive Data Buffer.
     The RDBFF0 flag is set to 1b on the rising edge of the 8th or 9th SCL clock cycle (selected in the ACKTWE bit in SCSTRCTL).
  3) When the received slave address matches after a START (or Repeated START) condition is detected with the TRMD bit in PRSST set to 0b.
- For I3C Protocol mode: PRTS.PRTMD bit = 0b. The following conditions 1) and 2) are satisfied:
  1) The NTSTE.RDBFE0 bit = 1b (enables Receive Data Buffer Full Interrupt Status logging).
  2) When the number of Normal Receive Data Buffer entries is the NTBTHCTL0.RXDBTH[2:0] threshold or more (see NTBTHCTL0 register).

[Clearing conditions]

- For I^2C Protocol mode: PRTS.PRTMD bit = 1b.
  - When data is read from NTDTBP0.
- For I3C Protocol mode: PRTS.PRTMD bit = 0b.
  - Write 0b to this bit after 1b is read.
  - On completion of the last read access to Normal Receive Data by DMAC.
  - When the number of Normal Receive Data Buffer entries is less than the NTBTHCTL0.RXDBTH[2:0] threshold (see NTBTHCTL0 register).

IBIQEFF bit (Normal IBI Queue Empty/Full Flag)

[Setting conditions]

- For I3C Protocol mode (Master): PRTS.PRTMD bit = 0b, PRSST.CRMS bit = 1b.

  The following 1) and 2) are satisfied.

  1) The NTSTE.IBIQEFE bit = 1b. (Enables IBI Status Buffer Empty/Full Interrupt Status logging)
  2) When the number of IBI Status Queue entries is more than the NQTHCTL.IBIQTH threshold (see register NQTHCTL).

- For I3C Protocol mode (Slave) : PRTS.PRTMD bit = 0b, PRSST.CRMS bit = 0b.

  If the NQTHCTL.IBIQTH = 0b:

  The following 1) and 2) are satisfied.

  1) The NTSTE.IBIQEFE bit = 1b. (Enables IBI Status Buffer Empty/Full Interrupt Status logging)
  2) When IBI Data Buffer is completely empty

- If the NQTHCTL.IBIQTH is other than 0b:

  The following 1) and 2) are satisfied.

  1) The NTSTE.IBIQEFE bit = 1b. (Enables IBI Status Buffer Empty/Full Interrupt Status logging)
  2) When the number of IBI Data Buffer empties is the NQTHCTL.IBIQTH threshold or more (see register NQTHCTL).

[Clearing conditions]

- For I3C protocol mode (master): PRTS.PRTMD bit = 0b, PRSST.CRMS bit = 1b.
  - Write 0b to this bit after 1b is read.
  - On completion of the last read access to IBI Status by DMAC.
  - When the number of IBI Status Queue entries is the NQTHCTL.IBIQTH threshold or less (see NQTHCTL register).
- For I3C protocol mode (slave): PRTS.PRTMD bit = 0b, PRSST.CRMS bit = 0b.
  - Write 0b to this bit after 1b is read.
  - On completion of the last write access to IBI Status by DMAC.
- If the NQTHCTL.IBIQTH = 0b:
  - When IBI Data Buffer is not completely empty.
- If the NQTHCTL.IBIQTH is other than 0b:
  - When the number of IBI Data Buffer empties is less than the NQTHCTL.IBIQTH threshold (see NQTHCTL register).

CMDQEF bit (Normal Command Queue Empty Flag)

[Setting conditions]

- If the NQTHCTL.CMDQTH = 0b:

  The following 1) and 2) are satisfied.

  1) The NTSTE.CMDQEE bit = 1b. (Enables Command Buffer Empty Interrupt Status logging)
  2) When Command Queue is completely empty

- If the NQTHCTL.CMDQTH is other than 0b:

  The following 1) and 2) are satisfied.

  1) The NTSTE.CMDQEE bit = 1b. (Enables Command Buffer Empty Interrupt Status logging)
  2) When the number of Command Queue empties is the NQTHCTL.CMDQTH threshold or more (see register NQTHCTL).

[Clearing conditions]

- Write 0b to this bit after 1b is read.
- On completion of the last write access to Normal Command by DMAC.

If the NQTHCTL.CMDQTH = 0b:

- When Command Queue is not completely empty.

If the NQTHCTL.CMDQTH is other than 0b:

- When the number of Command Queue empties is less than the NQTHCTL.CMDQTH threshold (see NQTHCTL register).

RSPQFF bit (Normal Response Queue Full Flag)

[Setting conditions]

- The following 2 conditions are satisfied:
  1) The NTSTE.RSPQFE bit = 1b (enables Response Buffer Full Interrupt Status logging).
  2) When the number of Response Queue entries is more than the NQTHCTL.RSPQTH threshold (see NQTHCTL register).

[Clearing conditions]

- Write 0b to this bit after 1b is read.
- On completion of the last read access to Normal Receive Status by DMAC.
- When the number of Response Queue entries is the NQTHCTL.RSPQTH threshold or less (see NQTHCTL register).

TABTF bit (Normal Transfer Abort Flag)

[Setting conditions]

- The following 2 conditions are satisfied:
  1) The NTSTE.TABTE bit = 1b (enables Transfer Abort Interrupt Status logging).
  2) When any transfer is aborted.

[Clearing condition]

- Write 0b to this bit after 1b is read.

TEF bit (Normal Transfer Error Flag)

[Setting conditions]

- The following 2 conditions are satisfied:
  1) The NTSTE.TEE bit = 1b (enables Transfer Error Interrupt Status logging).
  2) When any transfer error occurs on the I3C bus. The Error type for this error is available in the Response or Receive Status structure corresponding to the Transfer command.

[Clearing condition]

- Write 0b to this bit after 1b is read.

RSQFF bit (Normal Receive Status Queue Full Flag)

[Setting conditions]

- The following 2 conditions are satisfied:
  1) The NTSTE.RSQFE bit = 1b (Normal Receive Status Queue Full Enable).
  2) When the number of Receive Status Queue entries is more than the NRQTHCTL.RSQTH threshold (see NRQTHCTL register).

[Clearing conditions]

- Write 0 to this bit after 1 is read.
- On completion of the last read access to Normal Receive Status by DMAC.
- When the number of Receive Status Queue entries is the NRQTHCTL.RSQTH threshold or less (see register NRQTHCTL).

7.8.3.2.46 Normal Transfer Status Enable Register (I3Cm_NTSTE)

Access Size : 32 bits

Address : <I3C_base> + 01E4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	RSQFE	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	TEE	-	-	-	TABTE	RSPQFE	CMDQEE	IBIQEFE	RDBFE0	TDBEE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20	RSQFE	0h	RW	Normal Receive Status Queue Full Enable*2 0b: Disables Receive Status Buffer Full Interrupt Status logging. 1b: Enables Receive Status Buffer Full Interrupt Status logging.
19 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	TEE	0h	RW	Normal Transfer Error Enable*1 0b: Disables Transfer Error Interrupt Status logging. 1b: Enables Transfer Error Interrupt Status logging.
8 to 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	TABTE	0h	RW	Normal Transfer Abort Enable*1 0b: Disables Transfer Abort Interrupt Status logging. 1b: Enables Transfer Abort Interrupt Status logging.
4	RSPQFE	0h	RW	Normal Response Queue Full Enable*1 0b: Disables Response Buffer Full Interrupt Status logging. 1b: Enables Response Buffer Full Interrupt Status logging.
3	CMDQEE	0h	RW	Normal Command Queue Empty Enable*1 0b: Disables Command Buffer Empty Interrupt Status logging. 1b: Enables Command Buffer Empty Interrupt Status logging.
2	IBIQEFE	0h	RW	Normal IBI Queue Empty/Full Enable*1 0b: Disables IBI Status Buffer Empty/Full Interrupt Status logging. 1b: Enables IBI Status Buffer Empty/Full Interrupt Status logging.
1	RDBFE0	0h	RW	Normal Receive Data Buffer Full Enable 0b: Disables Receive Data Buffer Full Interrupt Status logging. 1b: Enables Receive Data Buffer Full Interrupt Status logging.
0	TDBEE0	0h	RW	Normal Transmit Data Buffer Empty Enable 0b: Disables Transmit Data Buffer Empty Interrupt Status logging. 1b: Enables Transmit Data Buffer Empty Interrupt Status logging.

Note 1. This bit supports all I3C mode.

Note 2. This bit supports I3C secondary master mode and I3C slave mode.

TDBEE0 bit (Normal Transmit Data Buffer Empty Enable)

When this bit is 1b, the operation of NTST.TDBEF0 is enabled.

For the setting conditions and clearing conditions of the NTST.TDBEF0 flag, see the details of NTST.TDBEF0.

RDBFE0 bit (Normal Receive Data Buffer Full Enable)

When this bit is 1b, the operation of NTST.RDBFF0 is enabled.

For the setting conditions and clearing conditions of the NTST.RDBFF0 flag, see the details of NTST.RDBFF0.

IBIQEFE bit (Normal IBI Queue Empty/Full Enable)

When this bit is 1b, the operation of NTST.IBIQEFF is enabled.

For the setting conditions and clearing conditions of the NTST.IBIQEFF flag, see the details of NTST.IBIQEFF.

CMDQEE bit (Normal Command Queue Empty Enable)

When this bit is 1b, the operation of NTST.CMDQEF is enabled.

For the setting conditions and clearing conditions of the NTST.CMDQEF flag, see the details of NTST.CMDQEF.

RSPQFE bit (Normal Response Queue Full Enable)

When this bit is 1b, the operation of NTST.RSPQFF is enabled.

For the setting conditions and clearing conditions of the NTST.RSPQFF flag, see the details of NTST.RSPQFF.

TABTE bit (Normal Transfer Abort Enable)

When this bit is 1b, the operation of NTST.TABTF is enabled.

For the setting conditions and clearing conditions of the NTST.TABTF flag, see the details of NTST.TABTF.

TEE bit (Normal Transfer Error Enable)

When this bit is 1b, the operation of NTST.TEF is enabled.

For the setting conditions and clearing conditions of the NTST.TEF flag, see the details of NTST.TEF.

RSQFE bit (Normal Receive Status Queue Full Enable)

When this bit is 1b, the operation of NTST.RSQFF is enabled.

For the setting conditions and clearing conditions of the NTST.RSQFF flag, see the details of NTST.RSQFF.

7.8.3.2.47 Normal Transfer Interrupt Enable Register (I3Cm_NTIE)

The PIO Interrupt Signal Enable register enables signaling of outstanding interrupts received by I3C.

Access Size : 32 bits
Address : <I3C_base> + 01E8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	RSQFIE	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	TEIE	-	-	-	TABTIE	RSPQFIE	CMDQEIE	IBIQEFIE	RDBFIE0	TDBEIE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20	RSQFIE	0h	RW	Normal Receive Status Queue Full Interrupt Enable*2 0b: Disables Receive Status Buffer Full Interrupt Signal. 1b: Enables Receive Status Buffer Full Interrupt Signal.
19 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	TEIE	0h	RW	Normal Transfer Error Interrupt Enable*1 0b: Disables Transfer Error Interrupt Signal. 1b: Enables Transfer Error Interrupt Signal.
8 to 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	TABTIE	0h	RW	Normal Transfer Abort Interrupt Enable*1 0b: Disables Transfer Abort Interrupt Signal. 1b: Enables Transfer Abort Interrupt Signal.
4	RSPQFIE	0h	RW	Normal Response Queue Full Interrupt Enable*1 0b: Disables Response Buffer Full Interrupt Signal. 1b: Enables Response Buffer Full Interrupt Signal.
3	CMDQEIE	0h	RW	Normal Command Queue Empty Interrupt Enable*1 0b: Disables Command Buffer Empty Interrupt Signal. 1b: Enables Command Buffer Empty Interrupt Signal.
2	IBIQEFIE	0h	RW	Normal IBI Queue Empty/Full Interrupt Enable*1 0b: Disables IBI Status Buffer Empty/Full Interrupt Signal. 1b: Enables IBI Status Buffer Empty/Full Interrupt Signal.
1	RDBFIE0	0h	RW	Normal Receive Data Buffer Full Interrupt Enable 0b: Disables Receive Data Buffer Full Interrupt Signal. 1b: Enables Receive Data Buffer Full Interrupt Signal.
0	TDBEIE0	0h	RW	Normal Transmit Data Buffer Empty Interrupt Enable 0b: Disables Transmit Data Buffer Empty Interrupt Signal. 1b: Enables Transmit Data Buffer Empty Interrupt Signal.

Note 1. This bit supports all I3C mode.

Note 2. This bit supports I3C secondary master mode and I3C slave mode.

TDBEIE0 bit (Normal Transmit Data Buffer Empty Interrupt Enable)

This bit is used to enable or disable the Normal Transmit Data buffer empty interrupt (INT_ri3c_tx_n) requests when the NTST.TDBEF0 flag is set to 1b.

RDBFIE0 bit (Normal Receive Data Buffer Full Interrupt Enable)

This bit is used to enable or disable the Normal Receive Data buffer full interrupt (INT_ri3c_rx_n) requests when the NTST.RDBFF0 flag is set to 1b.

IBIQEFIE bit (Normal IBI Queue Empty/Full Interrupt Enable)

This bit is used to enable or disable the Normal IBI Status buffer full interrupt (INTIBI) requests when the NTST.IBIQEFF flag is set to 1b.

CMDQEIE bit (Normal Command Queue Empty Interrupt Enable)

This bit is used to enable or disable the Normal Command buffer empty interrupt (INTCMD) requests when the NTST.CMDQEF flag is set to 1b.

RSPQFIE bit (Normal Response Queue Full Interrupt Enable)

This bit is used to enable or disable the Normal Response Status buffer full interrupt (INTRESP) requests when the NTST.RSPQFF flag is set to 1b.

TABTIE bit (Normal Transfer Abort Interrupt Enable)

This bit is used to enable or disable the Normal Transfer Abort interrupt (INTABORT) requests when the NTST.TABTF flag is set to 1b.

TEIE bit (Normal Transfer Error Interrupt Enable)

This bit is used to enable or disable the Normal Transfer Error interrupt (INTTERR) requests when the NTST.TEF flag is set to 1b.

RSQFIE bit (Normal Receive Status Queue Full Interrupt Enable)

This bit is used to enable or disable the Normal Receive Status buffer full interrupt (INTRCV) requests when the NTST.RSQFF flag is set to 1b.

7.8.3.2.48 Normal Transfer Status Force Register (I3Cm_NTSTFC)

The PIO Interrupt Force register is used to force specific interrupt. It can be used for debug purposes.

Access Size : 32 bits
Address : <I3C_base> + 01ECh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	RSQFFC	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	TEFC	-	-	-	TABTFC	RSPQFFC	CMDQEFC	IBIQEFFC	RDBFFC0	TDBEFC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	W	Reserved The written value should always be 0b.
20	RSQFFC	0h	W	Normal Receive Status Queue Full Force*2 0b: Not Force Receive Status Buffer Full Interrupt for software testing. 1b: Force Receive Status Buffer Full Interrupt for software testing.
19 to 10	-	All 0	W	Reserved The written value should always be 0b.
9	TEFC	0h	W	Normal Transfer Error Force*1 0b: Not Force Transfer Error Interrupt for software testing. 1b: Force Transfer Error Interrupt for software testing.
8 to 6	-	All 0	W	Reserved The written value should always be 0b.
5	TABTFC	0h	W	Normal Transfer Abort Force*1 0b: Not Force Transfer Abort Interrupt for software testing. 1b: Force Transfer Abort Interrupt for software testing.
4	RSPQFFC	0h	W	Normal Response Queue Full Force*1 0b: Not Force Response Buffer Full Interrupt for software testing. 1b: Force Response Buffer Full Interrupt for software testing.
3	CMDQEFC	0h	W	Normal Command Queue Empty Force*1 0b: Not Force Command Buffer Empty Interrupt for software testing. 1b: Force Command Buffer Empty Interrupt for software testing.
2	IBIQEFFC	0h	W	Normal IBI Queue Empty/Full Force*1 0b: Not Force IBI Status Buffer Full Interrupt for software testing. 1b: Force IBI Status Buffer Full Interrupt for software testing.
1	RDBFFC0	0h	W	Normal Receive Data Buffer Full Force 0b: Not Force Receive Data Buffer Full Interrupt for software testing. 1b: Force Receive Data Buffer Full Interrupt for software testing.
0	TDBEFC0	0h	W	Normal Transmit Data Buffer Empty Force 0b: Not Force Transmit Data Buffer Empty Interrupt for software testing. 1b: Force Transmit Data Buffer Empty Interrupt for software testing.

Note 1. This bit supports all I3C mode.

Note 2. This bit supports I3C secondary master mode and I3C slave mode.

TDBEFC0 bit (Normal Transmit Data Buffer Empty Force)

For software testing, when set to 1b, forces the corresponding interrupt, subject to TDBEE0 and TDBEIE0 configuration.

RDBFFC0 bit (Normal Receive Data Buffer Full Force)

For software testing, when set to 1b, forces the corresponding interrupt, subject to RDBFE0 and RDBFIE0 configuration.

IBIQEFFC bit (Normal IBI Queue Empty/Full Force)

For software testing, when set to 1b, forces the corresponding interrupt, subject to IBIQEFE and IBIQEFIE configuration.

CMDQEFC bit (Normal Command Queue Empty Force)

For software testing, when set to 1b, forces the corresponding interrupt, subject to CMDQEE and CMDQEIE configuration.

RSPQFFC bit (Normal Response Queue Full Force)

For software testing, when set to 1b, forces the corresponding interrupt, subject to RSPQFE and RSPQFIE configuration.

TABTFC bit (Normal Transfer Abort Force)

For software testing, forces the corresponding interrupt, subject to TABTE and TABTIE configuration.

TEFC bit (Normal Transfer Error Force)

For software testing, when set to 1b, forces the corresponding interrupt, subject to TEE and TEIE configuration.

RSQFFC bit (Normal Receive Status Queue Full Force)

For software testing, when set to 1b, forces the corresponding interrupt, subject to RSQFE and RSQFIE configuration.

7.8.3.2.49 Bus Condition Status Register (I3Cm_BCST)

Access Size : 32 bits
Address : <I3C_base> + 0210h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	BIDLF	BAVLF	BFREF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read.
2	BIDLF	0h	R	Bus Idle Detection Flag*1 0b: Have not Detected Bus Idle 1b: Have Detected Bus Idle
1	BAVLF	0h	R	Bus Available Detection Flag*1 0b: Have not Detected Bus Available 1b: Have Detected Bus Available
0	BFREF	0h	R	Bus Free Detection Flag 0b: Have not Detected Bus Free 1b: Have Detected Bus Free

Note 1. This bit supports all I3C mode.

BFREF bit (Bus Free Detection Flag)

The Bus Free Condition is a period occurring after a STOP and before a START, and with the following duration:

- For Pure Bus: A duration of at least tCAS
- For Mixed Bus (at least one Legacy I^2C is present on the I3C Bus): A duration of at least tBUF

[Setting conditions]

- After a STOP condition is detected, when the number of cycles (I3Cϕ) that are set by BFRECDT.FRECYC[8:0] has passed in the state of SCL = SDA = 1.
- After setting BCTL.BUSE to 1, when the number of cycles (I3Cϕ) that are set by BFRECDT.FRECYC[8:0] has passed in the state of SCL = SDA = 1.

[Clearing conditions]

- When SCL and SDA are other than high.
- When the BCTL.BUSE bit is set to 0b.

BAVLF bit (Bus Available Detection Flag)

The Bus Available Condition is a period during which the Bus Free Condition is sustained continuously for a duration of at least tAVAL. A Slave can only issue a START Request (for an In-Band Interrupt, or for a Master Handoff Request) after a Bus Available Condition.

[Setting conditions]

- After a STOP condition is detected, when the number of cycles (I3Cϕ) that are set by BAVLCDT.AVLCYC[8:0] has passed in the state of SCL = SDA = 1.
- After setting BCTL.BUSE to 1, when the number of cycles (I3Cϕ) that are set by BAVLCDT.AVLCYC[8:0] has passed in the state of SCL = SDA = 1.

[Clearing conditions]

- When SCL and SDA are other than high.
- When the BCTL.BUSE bit is set to 0b.

BIDLF bit (Bus Idle Detection Flag)

The I3C Bus Idle Condition is in order to help ensure Bus stability during Hot-Join events. The Bus Idle Condition is a period during which the Bus Available Condition is sustained continuously for a duration of at least tIDLE.

If a Hot-Join Device is powered up onto the I3C Bus at the same time as the Main Master, then the Hot-Join Device may pull SDA Low after 1 ms if (1) the Main Master has SCL and SDA pulled up, and (2) the Master does not act on the I3C Bus within the same Idle period.

[Setting conditions]

- After a STOP condition is detected, when the number of cycles (I3Cϕ) that are set by BIDLCDT.IDLCYC[17:0] has passed in the state of SCL = SDA = 1.
- After setting BCTL.BUSE to 1, when the number of cycles (I3Cϕ) that are set by BIDLCDT.IDLCYC[17:0] has passed in the state of SCL = SDA = 1.

[Clearing conditions]

- When SCL and SDA are other than high.
- When the BCTL.BUSE bit is set to 0b.

7.8.3.2.50 Slave Status Register (I3Cm_SVST)

Access Size : 32 bits
Address : <I3C_base> + 0214h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	SVAF2	SVAF1	SVAF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HOAF	-	-	-	-	-	-	-	-	DVIDF	HSMCF	-	-	-	-	GCAF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18	SVAF2	0h	RW*1	Slave Address Detection Flag 2 0b: Slave 2 not detected 1b: Slave 2 detected
17	SVAF1	0h	RW*1	Slave Address Detection Flag 1 0b: Slave 1 not detected 1b: Slave 1 detected
16	SVAF0	0h	RW*1	Slave Address Detection Flag 0 0b: Slave 0 not detected 1b: Slave 0 detected
15	HOAF	0h	RW*1	Host Address Detection Flag 0b: Host address not detected 1b: Host address detected This bit set to 1b when the received slave address matches the host address (0001_000b).
14 to 7	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6	DVIDF	0h	RW*1	Device-ID Address Detection Flag 0b: Device-ID command not detected 1b: Device-ID command detected This bit set to 1b when the first frame received immediately after a START condition is detected matches a value of (device ID (1111_100) + 0[W]).
5	HSMCF	0h	RW*1	Hs-mode Master Code Detection Flag 0b: Hs-mode Master Code not detected 1b: Hs-mode Master Code detected
4 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	GCAF	0h	RW*1	General Call Address Detection Flag 0b: General call address not detected 1b: General call address detected

Note: This register supports I^2C mode.

Note 1. Clearing (to 0) condition: Writing 0b after 1b is read.

GCAF bit (General Call Address Detection Flag)

I^2C Normal Wake-Up Mode1/2 sets GCAF to 1b when switching from asynchronous operation to synchronous unit.

[Setting conditions]

- This flag is set to 1b at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.

1) The SVCTL.GCAE bit = 1b (General call address detection is enabled).
2) When the received slave address matches the general call address (0000 000 + 0 (write)).

[Clearing conditions]

- When 0b is written to the GCAF flag after reading GCAF flag to be 1b.
- When a STOP condition is detected.
- When a Repeated START condition is detected.

HSMCF bit (Hs-mode Master Code Detection Flag)

I^2C Normal Wake-Up Mode1/2 sets HSMCF to 1b when switching from asynchronous operation to synchronous unit.

[Setting conditions]

- This flag is set to 1b at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.
  1) The SVCTL.HSMCE bit = 1b (Hs-mode master code detection is enabled).
  2) When the first byte received immediately after a START condition is detected matches a value of Hs-mode master code (0000 1XXX) + 1 (NACK).

[Clearing conditions]

- When 0b is written to the HSMCF flag after reading HSMCF flag to be 1b.
- When a STOP condition is detected.

DVIDF bit (Device-ID Address Detection Flag)

[Setting conditions]

- This flag is set to 1b at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.
  1) The SVCTL.DVIDE bit = 1b (Device-ID address detection is enabled).
  2) When the first byte received immediately after a START condition or Repeated START condition is detected matches a value of Device ID (1111 100) + 0 (write).

[Clearing conditions]

- When 0b is written to the DVIDF flag after reading DVIDF flag to be 1b.
- When a STOP condition is detected.
- This flag is set to 0b at the rising edge of the ninth SCL clock cycle in the first byte when the following 1 and 2 or 1 and 3 are satisfied.
  1) The SVCTL.DVIDE bit = 1b (Device-ID address detection is enabled).
  2) When the first byte received immediately after a START condition or Repeated START condition is detected does not match a value of Device ID (1111 100).
  3) When the first byte received immediately after a START condition or Repeated START condition is detected matches a value of (device ID (1111 100) + 0 [W]) and the second byte does not match any of slave addresses 0 to 2.

HOAF bit (Host Address Detection Flag)

I²C Normal Wake-Up Mode1/2 sets HOAF to 1 at the time of switching from asynchronous operation to synchronous unit.

[Setting conditions]

- This flag is set to 1b at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.
  1) The SVCTL.HOAE bit = 1b (Host address detection is enabled).
  2) When the received slave address matches the host address (0001 000).

[Clearing conditions]

- When 0b is written to the HOAF flag after reading HOAF flag to be 1b.
- When a STOP condition is detected.
- When a Repeated START condition is detected.

SVAFn bits (Slave Address Detection Flag n) (n = 0 to 2)

I²C Normal Wake-Up Mode1/2 sets SVAF2/1/0 to 1 when switching from asynchronous operation to synchronous unit.

[Setting conditions]

- For 7-bit address format: SVDVADn.SADLG bit = 0b.

  This flag is set to 1b at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.
  1) The SVCTL.SVAEn bit = 1b (Slave n enabled).
  2) When the received slave address matches the SVDVADn.SVAD[6:0] bits value.
- For 10-bit address format: SVDVADn.SADLG bit = 1b.

  This flag is set to 1b at the rising edge of the ninth SCL clock cycle in the second byte when all of the followings are satisfied.
  1) The SVCTL.SVAEn bit = 1b (Slave n enabled).
  2) When the received slave address matches a value of 11110 + SVDVADn.SVAD[9:8] bits and the following address matches the SVDVADn.SVAD[7:0] value.

[Clearing conditions]

- When 0b is written to the SVAFn flag after reading SVAFn flag to be 1b.
- When a STOP condition is detected.

For 7-bit address format: SVDVADn.SADLG bit = 0b.

This flag is set to 0b at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.

1) The SVCTL.SVAEn bit = 1b (Slave n enabled).
2) When the received slave address does not match SVDVADn.SVAD[6:0] bits value.

For 10-bit address format: SVDVADn.SADLG bit = 1b.

This flag is set to 0b at the rising edge of the ninth SCL clock cycle in the first byte when all of the followings are satisfied.

1) The SVCTL.SVAEn bit = 1b (Slave n enabled).

2) When the received slave address does not match a value of 11110 + SVDVADn.SVAD[9:8] bits.

- This flag is set to 0b at the rising edge of the ninth SCL clock cycle in the second byte when all of the followings are satisfied.

1) The SVCTL.SVAEn bit = 1b (Slave n enabled).

2) When the received slave address matches a value of 11110 + SVDVADn.SVAD[9:8] bits and the following address does not match the SVDVADn.SVAD[7:0] value.

7.8.3.2.51 Wake-Up Unit Operating Status Register (I3Cm_WUST)

Access Size : 32 bits
Address : <I3C_base> + 0218h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WUASYNF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read.
0	WUASYNF	0h	R	Wake-Up Function Asynchronous Operation Status Flag 0b: I3C synchronous circuit enable condition 1b: I3C asynchronous circuit enable condition

WUASYNF bit (Wake-Up Function Asynchronous Operation Status Flag)

This bit shows whether I3C is in the I3C_0_TCLK asynchronous operation (WUCTL.WUFE bit = 1b).

[Setting conditions]

- The following 1) and 2) are satisfied.
  1) The WUCTL.WUFE bit = 1b (Wake-up function is enabled)
  2) When the BCST.BFREF flag = 1b after 0b is written to the WUCTL.WUFSYNE bit

[Clearing condition: I^2C slave]

- The WUCTL.WUFE bit = 0b (Wake-up function is disabled)
- All of the following 1) to 3) are satisfied.
  1) The WUCTL.WUFE bit = 1b (Wake-up function is enabled)
  2) Wake-up event is detected
  3) When 1b is written to the WUCTL.WUFSYNE bit while the WUASYNF flag = 1b

[Clearing condition: I3C slave]

- The WUCTL.WUFE bit = 0b (Wake-up function is disabled)
- All of the following 1) to 4) are satisfied.
  1) The WUCTL.WUFE bit = 1b (Wake-up function is enabled)
  2) Wake-up event is detected
  3) When 1b is written to the WUCTL.WUFSYNE bit while the WUASYNF flag =1b
  4) When a STOP condition is detected.

[Clearing condition: I^2C/I3C slave]

- All of the following 1) to 5) are satisfied.

1) The WUCTL.WUFE bit = 1b (Wake-up function is enabled)
2) Wake-up event is not detected
3) The WUASYNF flag = 1b
4) The WUCTL.WUFSYNE bit = 1b
5) When a STOP condition is detected.

[Clearing condition: I3C master]

- The WUCTL.WUFE bit = 0b (Wake-up function is disabled)
- All of the following 1) to 4) are satisfied.
  1) The WUCTL.WUFE bit = 1b (Wake-up function is enabled)
  2) Wake-up event is detected
  3) The WUASYNF flag = 1b
  4) The WUCTL.WUFSYNE bit = 1b

7.8.3.2.52 MsyncCNT Counter Capture Register (I3Cm_MRCCPT)

Access Size : 32 bits
Address : <I3C_base> + 021Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MRCCPT[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MRCCPT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MRCCPT[31:0]	0h	R	MSyncCNT Counter Capture Used in Async Mode 1, not used in Async Mode0.

Note: This register supports I3C master mode and I3C secondary master mode.

MRCCPT[31:0] bits (MSyncCNT Counter Capture)

- Async Mode 1 (Asynchronous Advanced Mode)

When ATCCNTE.ATCE is enabled, the counter starts counting. Its value is captured as MSyncCNT for each aME (SDA falling edge of START condition), and stored in the capture register.

7.8.3.2.53 Device Address Table Basic Register n (I3Cm_DATBASn) (n = 0 to 7)

Access Size : 32 bits

Address : <I3C_base> + 0224h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DVTYP	DVNACK[1:0]		-	-	-	-	-	DVDYAD[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DVIBITS	DVMRRJ	DVSIRRJ	DVIBIPL	-	-	-	-	-	DVSTAD[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	DVTYP	0h	RW	Device Type 0b: I3C Device 1b: I2C Device
30 to 29	DVNACK[1:0]	0h	RW	Device NACK Retry Count Device-specific retry count
28 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23 to 16	DVDYAD[7:0]	0h	RW	Device I3C Dynamic Address Bit 23 is the parity bit, per the I3C specification, computed and updated by the software driver.
15	DVIBITS	0h	RW	Device IBI Time-stamp 0b: The Master shall not time-stamp IBIs from this Device with Master Time-stamps. 1b: The Master shall time-stamp IBIs for this Device with Master Time-stamps.
14	DVMRRJ	0h	RW	Device In-Band Master Request Reject 0b: This Device shall ACK Master Requests. 1b: This Device shall NACK Master Requests and send the auto-disable command.
13	DVSIRRJ	0h	RW	Device In-Band Slave Interrupt Request Reject 0b: This Device shall ACK the SIR. 1b: This Device shall NACK the SIR and send the auto-disable CCC.
12	DVIBIPL	0h	RW	Device IBI Payload 0b: IBIs from this Device do not carry a Data Payload. 1b: IBIs from this Device do carry a Data Payload.
11 to 7	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6 to 0	DVSTAD[6:0]	0h	RW	Device Static Address I3C Static Address

Note: This register supports I3C master mode and I3C secondary master mode.

DVIBIPL bit (Device IBI Payload)

Indicates whether IBIs from this Device have a Data Payload. This field reflects the IBI Payload bit in the Device's Bus Characteristics Register (BCR).

During IBI handling for this Device, the Master shall use this field to determine whether or not to drive reception of the IBI Data Payload. Data continuation is indicated by the T-Bit.

DVSIRRJ bit (Device In-Band Slave Interrupt Request Reject)

Controls whether this Device, when operating in the Master role, will accept vs. reject Slave Interrupt Requests from other Devices.

DVMRRJ bit (Device In-Band Master Request Reject)

Controls whether this Device, when operating in the Master role, will accept vs. reject Master requests from other Devices. This bit is only valid if I3C declares Non-Current Master Capability.

DVIBITS bit (Device IBI Time-stamp)

Enables or disables IBI time-stamping for a specific Device.

Note: The IBI Status Descriptor for each IBI event indicates whether or not the individual IBI event was actually time- stamped. Set to 0 except for Async mode 0 and Async mode 1 of timing control.

DVNACK[1:0] bits (Device NACK Retry Count)

These bits set the number of retries when a NACK response is received from the slave for the transaction set in the Command Descriptor.

Note: When ENTDAA is executed by Address Assign Command, the setting of this bit is ignored and the transaction ends when NACK is received once.

Note: I3C will retry according to the setting of DVNACK[1:0], even if it receives a NACK for the broadcast address.

Note: If DVNACK[1:0] is 00b, I3C will not retry even for Direct CCCs.

7.8.3.2.54 Slave Device Address Table Basic Register n (I3Cm_SDATBASn) (n = 0 to 2)

Access Size : 32 bits
Address : <I3C_base> + 02B0h + n x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	SDDYAD[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SDIBIPL	-	SDADLS	SDSTAD[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
22 to 16	SDDYAD*1 [6:0]	0h	RW	Slave Device I3C Dynamic Address*5
15 to 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12	SDIBIPL*1	0h	RW	Slave Device IBI Payload*4 This bit is the mirror bit of the SVDCT.TBCR[2]. 0b: IBIs from this device do not carry a data payload. 1b: IBIs from this device carry a data payload.
11	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10	SDADLS	0h	RW	Slave Device Address Length Selection*3 0b: Slave device address length 7 bits selected. 1b: Slave device address length 10 bits selected. (I2C device only)
9 to 0	SDSTAD[9:0]	0h	RW	Slave Device Static Address*2 I3C / I2C Static Address

Note: SW write to the SDATBAS register of the main master is prohibited.

Note 1. This bit is valid only in SDATBAS0 register.

Note 2. These bits support I^2C, I3C secondary master, and I3C slave mode.

Note 3. This bit supports I^2C mode.

Note 4. This bit supports I3C secondary master mode and I3C slave mode.

Note 5. These bits support I3C secondary master mode and I3C slave mode.

SDSTAD[9:0] bits (Slave Device Static Address)

When the 7-bit address format is selected (SDADLS bit is 0b), the lower 7 bits of SDSTAD[9:0] function as the 7-bit address.

When the 10-bit address format is selected (SDADLS bit is 1b), the SDSTAD[9:0] function as the 10-bit address. While the SVCTL.SVAEn bit is 0b, the setting of this bit is ignored.

SDIBIPL bit (Slave Device IBI Payload)

Indicates whether IBIs from this Device have a Data Payload. This field reflects the IBI Payload bit in the Device's Bus Characteristics Register (BCR).

During IBI handling for this Device, the Master shall use this field to determine whether or not to drive reception of the IBI Data Payload. Data continuation is indicated by the T-Bit.

SDDYAD[6:0] bits (Slave Device I3C Dynamic Address)

[Update conditions]

- When writing Dynamic Address value.
- When Slave Address value is its own Static Address in receiving SETDASA CCC (Direct), these bits are updated to Dynamic Address value.*1
- When Dynamic Address Assignment procedure that starts by receiving ENTDAA CCC (Broadcast) is established.*1
- When receiving RSTDAA CCC (Broadcast), all bits are cleared to 0.*1
- When Slave Address value is its own Dynamic Address in receiving RSTDAA CCC (Direct), all bits are cleared to 0.*1
- When Slave Address value is its own Dynamic Address in receiving SETNEWDA CCC (Direct), these bits are updated to the Dynamic Address value.*1
- When receiving SETAASA CCC (Broadcast), these bits are updated to the value of SDSTAD[6:0] bits*2.

Note 1. See the MIPI I3C Specification v1.0.

Note 2. See the MIPI I3C Basic Specification v1.0.

7.8.3.2.55 Master Device Characteristic Table Register n (I3Cm_MSDCTn) (n = 0 to 7)

Access Size : 32 bits

Address : <I3C_base> + 02D0h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RBCR76[1:0]		-	-	RBCR3	RBCR2	RBCR1	RBCR0	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15, 14	RBCR76[1:0]	0h	RW	Device Role 00b: I3C Slave 01b: I3C Master*3 Others: Setting prohibited
13, 12	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
11	RBCR3	0h	RW	Offline Capable*2 0b: Device will always respond to I3C bus commands. 1b: Device will not always respond to I3C bus commands.
10	RBCR2	0h	RW	IBI Payload 0b: No data byte follows the accepted IBI. 1b: Mandatory one or more data bytes follow the accepted IBI. Data byte continuation is indicated by T-Bit.
9	RBCR1	0h	RW	IBI Request Capable 0b: Not Capable 1b: Capable
8	RBCR0	0h	RW	Max Data Speed Limitation*1 0b: No Limitation 1b: Limitation
7 to 0	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Note: This register supports I3C master mode and I3C secondary master mode.

Note 1. Master shall use the GETMXDS CCC to interrogate the Slave for specific limitation.

Note 2. Offline Capable Devices retain the Dynamic Address.

Note 3. For an I3C Device acting as I3C Main Master, the BCR Device Role bits will contain the value 01.

RBCRn bits (Received Bus Characteristic Register)

Each I3C Device that is connected to the I3C Bus shall have an associated read-only Bus Characteristics Register (BCR). This read-only register describes the I3C compliant Device's role and capabilities for use in Dynamic Address assignment and Common Command Codes.

Note: When RBCR[2] is 0b and when ACK response to Slave Interrupt Request from I3C Slave while DATBASm.DVSIRRJ is set to 0b, STOP Condition is issued after ACK response. When RBCR[2] is 1b and when ACK response to Slave Interrupt Request from I3C Slave while DATBASm.DVSIRRJ is set to 0b, IBI Payload is received after ACK response. STOP Condition is issued after end of IBI Payload.

[Update condition]

When receiving of Bus Characteristics Register (BCR) from Device in the Dynamic Address Assignment procedure starting by receiving ENTDAA CCC (Broadcast).*1

Note 1. See the MIPI I3C Specification v1.0

7.8.3.2.56 Extended Device Address Table Basic Register (I3Cm_EXDATBAS)

Access Size : 32 bits
Address : <I3C_base> + 0310h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EDTYP	EDNACK[1:0]		-	-	-	-	-	EDDYAD[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	EDSTAD[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	EDTYP	0h	RW	Extended Device Type 0b: I3C Device 1b: I2C Device
30, 29	EDNACK[1:0]	0h	RW	Extended Device NACK Retry Count Device-specific retry count
28 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23 to 16	EDDYAD[7:0]	0h	RW	Extended Device I3C Dynamic Address Bit 23 is the parity bit, per the I3C specification, computed and updated by the software driver.
15 to 7	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6 to 0	EDSTAD[6:0]	0h	RW	Extended Device Static Address I3C / I2C static address

Note: This register supports I3C master mode and I3C secondary master mode.

7.8.3.2.57 Slave Device Characteristic Table Register (I3Cm_SVDCT)

Access Size : 32 bits
Address : <I3C_base> + 0320h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TBCR76[1:0]		-	-	TBCR3	TBCR2	TBCR1	TBCR0	TDCR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 14	TBCR76[1:0]	0h	RW	Device Role 0 0b: I3C Slave 0 1b: I3C Master*3 1 0b: Reserved for future definition by MIPI Sensor WG 1 1b: Reserved for future definition by MIPI Sensor WG
13, 12	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
11	TBCR3	0h	RW	Offline Capable*2 0b: Device will always respond to I3C bus commands. 1b: Device will not always respond to I3C bus commands.
10	TBCR2	0h	RW	IBI Payload 0b: No data byte follows the accepted IBI. 1b: Mandatory one or more data bytes follow the accepted IBI. Data byte continuation is indicated by T-Bit.
9	TBCR1	0h	RW	IBI Request Capable 0b: Not Capable 1b: Capable
8	TBCR0	0h	RW	Max Data Speed Limitation*1 0b: No Limitation 1b: Limitation
7 to 0	TDCR[7:0]	0h	RW	Transfer Device Characteristic Register 255 available codes for describing the type of sensor, or device. Examples: Accelerometer, gyroscope, composite devices Default value is 0b: Generic Device

Note: This register supports I3C secondary master mode and I3C slave mode.

Note 1. Master shall use the GETMXDS CCC to interrogate the Slave for specific limitation.

Note 2. Offline Capable Devices retain the Dynamic Address.

Note 3. For an I3C Device acting as I3C Main Master, the BCR Device Role bits will contain the value 01.

TDCR[7:0] bits (Transfer Device Characteristic Register)

Each I3C device that is connected to the I3C bus has an associated Device Characteristics Register (DCR). This register describes the I3C compliant device type such as accelerometer and gyroscope, for use in Dynamic Address assignment and Common Command Codes.

TBCRn bits (Transfer Bus Characteristic Register)

Each I3C device that is connected to the I3C bus has an associated Bus Characteristics Register (BCR). This register describes the role and capabilities of the I3C compliant device for use in Dynamic Address assignment and Common Command Codes.

When I3C Slave issues IBI by Command Descriptor, the setting values of TBCR[7:0] are as follows:

[Slave Interrupt Request : No IBI Payload follow the accepted IBI]

- TBCR1 = 1b
- TBCR2 = 0b

Note: Set DATA_LENGTH[15:0] of Command Descriptor to 0.

[Slave Interrupt Request : IBI Payload follow the accepted IBI]

- TBCR1 = 1b
- TBCR2 = 1b

Note: Set DATA_LENGTH[15:0] of Command Descriptor to any value.

[Mastership Request]

- TBCR1 = 1b
- TBCR76[1:0] = 01b

[Hot-join Event]

- TBCR1 = 1b

When I3C Slave receives CCC from I3C Master, it performs the following operations according to the setting of TBCR[7:0]:

- When TBCR2 = 1b, CMRLG.IBIPSZ[7:0] is sent as the 3rd byte data to GETMRL CCC from I3C Master
- When TBCR0 = 0b, NACK responses to GETMXDS CCC from I3C Master
- When TBCR0 = 1, ACK responses to GETMXDS CCC from I3C Master and sends data from CMDSPW, CMDSPR, and CMDSPT registers

7.8.3.2.58 Slave Device Characteristic Table Provisional ID Low Register (I3Cm_SDCTPIDL)

Access Size : 32 bits

Address : <I3C_base> + 0324h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDCTPIDL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDCTPIDL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SDCTPIDL [31:0]	0h	RW	Transfer Device Provisional ID Low Bits 31 to 16 are read as 0b. Bits 15 to 0 are bits [15:0] of device's I3C PID.

Note: This register supports I3C secondary master mode and I3C slave mode.

7.8.3.2.59 Slave Device Characteristic Table Provisional ID High Register (I3Cm_SDCTPIDH)

Access Size : 32 bits

Address : <I3C_base> + 0328h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDCTPIDH[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDCTPIDH[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SDCTPIDH [31:0]	0h	RW	Transfer Device Provisional ID High Bits [47:16] of device's I3C PID.

Note: This register supports I3C secondary master mode and I3C slave mode.

7.8.3.2.60 Slave Device Address Register n (I3Cm_SVDVADn) (n = 0 to 2)

Access Size : 32 bits

Address : <I3C_base> + 0330h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDYADV	SSTADV	-	-	SADLG	-	SVAD[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	SDYADV*4	0h	R	Slave Dynamic Address Valid*3 0b: Dynamic Address is disabled. 1b: Dynamic Address is enabled.
30	SSTADV	0h	R	Slave Static Address Valid*1 0b: Slave address is disabled. 1b: Slave address is enabled.
29, 28	-	All 0	R	Reserved Whenever it is read, 0b is read.
27	SADLG	0h	R	Slave Address Length*2 0b: The 7-bit address format is selected. 1b: The 10-bit address format is selected.
26	-	0h	R	Reserved Whenever it is read, 0b is read.
25 to 16	SVAD[9:0]	0h	R	Slave Address*1 A slave address is set. When rewriting SVAD, change to SVAE = 0b and rewrite.
15 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read.

Note 1. These bits support I^2C, I3C secondary master, and I3C slave mode.

Note 2. This bit supports I^2C mode.

Note 3. This bit supports I3C secondary master mode and I3C slave mode.

Note 4. This bit is valid only in SVDVAD0 register.

SVAD[9:0] bits (Slave Address)

The SVAD[9:0] bits indicate a valid slave address.

[The SVDVAD0.SDYADV bit = 1b]

Note: This condition is only for SVDVAD0.SVAD[9:0].

- The SVAD[9:7] bits = 0b
- The SVAD[6:0] bits = the SDATBAS0.SDDYAD[6:0] bits

[The SVDVADn.SSTADV bit = 1b and the SVDVADn.SADLG bit = 0b]

- The SVAD[9:7] bits = 0b
- The SVAD[6:0] bits = the SDATBASn.SDSTAD[6:0] bits

[The SVDVADn.SSTADV bit = 1b and the SVDVADn.SADLG bit = 1b]

- The SVAD[9:0] bits = the SDATBASn.SDSTAD[9:0] bits

SADLG bit (Slave Address Length)

[Setting conditions]

- All of the followings are satisfied:
  1) The PRTS.PRTMD bit = 1b (I^2C Protocol mode)
  2) The SVCTL.SVAEn bit = 1b (Slave n is enabled)
  3) The SDATBASn.SDADLS bit = 1b (The address length is 10 bits)

[Clearing condition]

- [Setting condition] is not satisfied.

SSTADV bit (Slave Static Address Valid)

[Setting conditions]

- All of the followings are satisfied:
  1) The SVCTL.SVAEn bit = 1b (Slave n is enabled)
  2) The SVDVAD0.SDYADV bit = 0b (Dynamic Address is disabled)

     Note: This condition is only for SVDVAD0.SSTADV.

  3) If the SVDVADn.SADLG bit = 0b, the SDATBASn.SDSTAD[6:0] bits are not all 0
     If the SVDVADn.SADLG bit = 1b, the SDATBASn.SDSTAD[9:0] bits are not all 0

[Clearing condition]

- "[Setting condition]" is not satisfied.

SDYADV bit (Slave Dynamic Address Valid)

[Setting conditions]

- All of the followings are satisfied:
  1) The PRTS.PRTMD bit = 0b (I3C Protocol mode)
  2) The SVCTL.SVAEn bit = 1b (Slave n is enabled)
  3) The SDATBAS0.SDDYAD[6:0] bits are not all 0

     Note: This condition is only for SVDVAD0.SDYADV.

[Clearing condition]

- "[Setting condition]" is not satisfied.

7.8.3.2.61 CCC Slave Events Command Register (I3Cm_CSECMD)

Access Size : 32 bits
Address : <I3C_base> + 0350h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	HJEVE	-	MSRQE	SVIRQE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3	HJEVE	0h	RW	Hot-Join Event Enable 0b: DISABLED: Slave-initiated Hot-Join is Disabled by the Master to control. 1b: ENABLED: Slave-initiated Hot-Join is Enabled by the Master to control.
2	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	MSRQE	0h	RW	Mastership Requests Enable 0b: DISABLED: Mastership requests from Secondary Masters is Disabled by the Current Master to control. 1b: ENABLED: Mastership requests from Secondary Masters is Enabled by the Current Master to control.
0	SVIRQE	0h	RW	Slave Interrupt Requests Enable 0b: DISABLED: Slave-initiated Interrupts is Disabled by the Master to control. 1b: ENABLED: Slave-initiated Interrupts is Enabled by the Master to control.

Note: This register supports I3C secondary master mode and I3C slave mode.

SVIRQE bit (Slave Interrupt Requests Enable)

This bit allows the Master to control when Slave-initiated Interrupts are allowed on the I3C Bus.

These four Direct (ENEC/DISEC Format 1) or Broadcast (ENEC/DISEC Format 2) CCCs allows the Master to control when Slave- initiated traffic is (Enable) vs. is not (Disable) allowed on the I3C Bus. This control governs a Slave's attempts to request an Interrupt (ENI), to request Mastership (ENMR), or to signify a Hot-Join event (ENHJ).

[Setting conditions]

- When writing 1b
- When receiving ENEC CCC (Broadcast) with ENINT bit = 1b.*1
- When ENINT bit = 1b with own Slave Address in receiving ENEC CCC (Direct).*1

[Clearing conditions]

- When writing 0b.
- When receiving DISEC CCC (Broadcast) with DISINT bit = 1b.*1
- When DISINT bit = 1b with own Slave Address in receiving DISEC CCC (Direct).*1

MSRQE bit (Mastership Requests Enable)

This bit allows the Current Master to control when Mastership requests from Secondary Masters are allowed on the I3C Bus.

[Setting conditions]

- When writing 1b.
- When receiving ENEC CCC (Broadcast) with ENMR bit = 1b.*1
- When ENMR bit = 1b with own Slave Address in receiving ENEC CCC (Direct).*1

[Clearing conditions]

- When writing 0b.
- When receiving DISEC CCC (Broadcast) with DISMR bit = 1b.*1
- When DISMR bit = 1b with own Slave Address in receiving DISEC CCC (Direct).*1

HJEVE bit (Hot-Join Event Enable)

This bit allows the Master to control when Slave-initiated Hot-Join is allowed on the I3C Bus. [Setting conditions]

- When writing 1b.
- When receiving ENEC CCC (Broadcast) with ENHJ bit = 1b.*1
- When ENHJ bit = 1b with own Slave Address in receiving ENEC CCC (Direct).*1

[Clearing conditions]

- When writing 0b.
- When receiving DISEC CCC (Broadcast) with DISHJ bit = 1b.*1
- When DISHJ bit = 1b with own Slave Address in receiving DISEC CCC (Direct)*1

Note 1. See the MIPI I3C Specification v1.0

7.8.3.2.62 CCC Enter Activity State Register (I3Cm_CEACTST)

Access Size : 32 bits
Address : <I3C_base> + 0354h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	ACTST[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3 to 0	ACTST[3:0]	0h	RW	Activity State 1h: ENTAS0 (1 µs: Latency-free operation) 2h: ENTAS1 (100 µs) 4h: ENTAS2 (2 ms) 8h: ENTAS3 (50 ms: Lowest-activity operation) Others: Setting prohibited

Note: This register supports I3C secondary master mode and I3C slave mode.

ACTST[3:0] bits (Activity State)

[Update conditions]

- When writing Activity State value.
- When receiving ENTAS0 CCC (Broadcast), these bits are updated to 1h.*1
- When receiving ENTAS1 CCC (Broadcast), these bits are updated to 2h.*1
- When receiving ENTAS2 CCC (Broadcast), these bits are updated to 4h.*1
- When receiving ENTAS3 CCC (Broadcast), these bits are updated to 8h.*1
- When Slave Address value is its own Slave Address in receiving ENTAS0 CCC (Direct), these bits are updated to 1h.*1
- When Slave Address value is its own Slave Address in receiving ENTAS1 CCC (Direct), these bits are updated to 2h.*1
- When Slave Address value is its own Slave Address in receiving ENTAS2 CCC (Direct), these bits are updated to 4h.*1
- When Slave Address value is its own Slave Address in receiving ENTAS3 CCC (Direct), these bits are updated to 8h.*1

Note 1. See the MIPI I3C Specification v1.0.

7.8.3.2.63 CCC Max Write Length Register (I3Cm_CMWLG)

Access Size : 32 bits
Address : <I3C_base> + 0358h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MWLG[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 0	MWLG[15:0]	0h	RW	Max Write Length

Note: This register supports I3C secondary master mode and I3C slave mode.

MWLG[15:0] bits (Max Write Length)

These bits use for the I3C Master to set or get a maximum data write length in bytes for one Slave Device.

This Max Write Length does not affect data write lengths for Broadcast CCCs. The Set/Get Max Write Length value is transmitted over two bytes, with the most significant byte (MSB) transmitted first. The minimum value that Max Write Length can be set to is 8.

[Update conditions]

- When writing Max Write Length value.
- When receiving SETMWL CCC (Broadcast), these bits are updated to MWL value.*1
- When Slave Address value is its own Slave Address in receiving SETMWL CCC (Direct), these bits are updated to MWL value.*1

Note 1. See the MIPI I3C Specification v1.0

7.8.3.2.64 CCC Max Read Length Register (I3Cm_CMRLG)

Access Size : 32 bits
Address : <I3C_base> + 035Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	IBIPSZ[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MRLG[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23 to 16	IBIPSZ[7:0]	0h	RW	IBI Payload Size
15 to 0	MRLG[15:0]	0h	RW	Max Read Length

Note: This register supports I3C secondary master mode and I3C slave mode.

MRLG[15:0] bits (Max Read Length)

These bits use for the I3C Master to set or get a maximum data read length for one Slave Device.

The Set/Get Max Read Length value is transmitted over the first two bytes, with most significant byte (MSB) transmitted first. The minimum value to which Max Read Length can be set is 16.

[Update conditions]

- When writing Max Read Length value.
- When receiving SETMRL CCC (Broadcast), these bits are updated to MRL value.*[1]
- When Slave Address value is its own Slave Address in receiving SETMRL CCC (Direct), these bits are updated to MRL value.*[1]

IBIPSZ[7:0] bits (IBI Payload Size)

These bits use for the I3C Master to set or get optionally a maximum IBI payload size.

For devices with BCR bit 2 set to 1b, the Max IBI payload size value is added as a third-byte, where a value of 0 indicates an unlimited payload size. If Timing Control is used, then the minimum IBI payload size is either four bytes or five bytes. If Timing Control is not used, then the minimum IBI payload size is one (one byte).

This CCC is optional for the Slave, with two exceptions:

1. This CCC is required if both (a) any private Read Request Message (s) and/or any extended Read Request CCC (s) implemented by the Slave support a variable limit on the maximum number of data bytes that the Slave may return per Message, and (b) this limit is greater than 16 bytes.
2. This CCC is required if the Slave both (a) supports an IBI Payload (as indicated with BCR bit 1), and (b) will transmit more than one byte of private payload (not counting Timing Control bytes, when Timing Control is used).

[Update conditions]

- When writing Max IBI payload size value.
- When receiving SETMRL CCC (Broadcast), these bits are updated to IBI payload size value.*1
- When Slave Address value is its own Slave Address in receiving SETMRL CCC (Direct), these bits are updated to IBI payload size value.*1

Note 1. See the MIPI I3C Specification v1.0.

7.8.3.2.65 CCC Enter Test Mode Register (I3Cm_CETSTMD)

Access Size : 32 bits
Address : <I3C_base> + 0360h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TSTMD[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read.
7 to 0	TSTMD[7:0]	0h	R	Test Mode 00h: Exit Test Mode This value removes all I3C devices from Test Mode. 01h: Vendor Test Mode This value indicates that I3C devices shall return a random 32bit value in the provisional ID during the Dynamic Address Assignment procedure. Others: MIPI reserved Reserved for future use by the MIPI Alliance

Note: This register supports I3C secondary master mode and I3C slave mode.

TSTMD[7:0] bits (Test Mode)

When these bits set to 00h, all I3C Devices remove from Test Mode.

When these bits set to 01h, I3C Devices shall return a random 32bit value in the Provisional ID during the Dynamic Address Assignment procedure.

The Broadcast CCC informs all I3C Devices that the Master is entering a specified Test Mode during manufacturing or Device test. The Enter Test Mode command Frame format includes a byte that specifies which Test Mode to enter.

Supporting I3C Devices shall enter the indicated Test Mode upon receipt of the Enter Test Mode CCC.

[Update condition]

- When receiving ENTTM CCC (Broadcast), these bits are updated to Test Mode Byte value.*[1]

Note 1. See the MIPI I3C Specification v1.0.

7.8.3.2.66 CCC Get Device Status Register (I3Cm_CGDVST)

Access Size : 32 bits
Address : <I3C_base> + 0364h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VDRSV[7:0]								ACTMD[1:0]		PRTE	-	PNDINT[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 8	VDRSV[7:0]	0h	RW	Vendor Reserved Reserved for vendor-specific meaning
7 to 6	ACTMD[1:0]	0h	RW	Slave Device's current Activity Mode 00b: Activity Mode 0 01b: Activity Mode 1 10b: Activity Mode 2 11b: Activity Mode 3
5	PRTE	0h	RW	Protocol Error 0b: The Slave has not detected a protocol error since the last Status read. 1b: The Slave has detected a protocol error since the last Status read.
4	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3 to 0	PNDINT[3:0]	0h	RW	Pending Interrupt Contains the interrupt number of any pending interrupt, or 0 if no interrupts are pending. This encoding allows for up to 15 numbered interrupts. If more than one interrupt is set, then the highest priority interrupt shall be returned.

Note: This register supports I3C secondary master mode and I3C slave mode.

PRTE bit (Protocol Error)

If this bit set to 1b, then the Slave detects a protocol error since the last Status read.

The Slave checks for such errors. Note that this value self-clears by the hardware upon every successful completion of a Master read of the Slave's Status.

The Direct CCC is a Get request for one I3C Slave Device to return its current Status, in the two-byte format detailed. Note that byte 0 is the LSB, and byte 1 is the MSB.

[Setting condition]

- When the Slave detected a protocol error.*1

[Clearing condition]

- When transmission by own Slave Address is completed without error after receiving GETSTATUS CCC (Direct).*1

ACTMD[1:0] bits (Slave Device’s current Activity Mode)

Contains the two-bit ID of the Slave Device’s current Activity Mode (readiness to support data read of sensor or related information).

Note 1. See the MIPI I3C Specification v1.0.

7.8.3.2.67 CCC Max Data Speed W (Write) Register (I3Cm_CMDSPW)

Access Size : 32 bits

Address : <I3C_base> + 0368h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	MSWDR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	MSWDR[2:0]	0h	RW	Maximum Sustained Write Data Rate 000b: fscl Max (default value) 001b: 8 MHz 010b: 6 MHz 011b: 4 MHz 100b: 2 MHz Others: Setting prohibited

Note: This register supports I3C secondary master mode and I3C slave mode.

7.8.3.2.68 CCC Max Data Speed R (Read) Register (I3Cm_CMDSPR)

Access Size : 32 bits

Address : <I3C_base> + 036Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	CDTTIM[2:0]			MSRDR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5 to 3	CDTTIM[2:0]	0h	RW	Clock to Data Turnaround Time (TSCO) 000b: 8 ns or less (default value) 001b: 9 ns or less 010b: 10 ns or less 011b: 11 ns or less 100b: 12 ns or less 111b: TSCO is more than 12 ns, and is reported by private agreement. Others Setting prohibited
2 to 0	MSRDR[2:0]	0h	RW	Maximum Sustained Read Data Rate 000b: fscl Max (default value) 001b: 8 MHz 010b: 6 MHz 011b: 4 MHz 100b: 2 MHz Others: Setting prohibited

Note: This register supports I3C secondary master mode and I3C slave mode.

7.8.3.2.69 CCC Max Data Speed T (Turnaround) Register (I3Cm_CMDSPT)

Access Size : 32 bits

Address : <I3C_base> + 0370h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MRTE	-	-	-	-	-	-	-	MRTTIM[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MRTTIM[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	MRTE	0h	RW	Maximum Read Turnaround Time Enable 0b: Disables transmission of the Maximum Read Turnaround Time. (GETMXDS Format 1: Without Turnaround) 1b: Enables transmission of the Maximum Read Turnaround Time. (GETMXDS Format 2: With Turnaround)
30 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23 to 0	MRTTIM[23:0]	0h	RW	Maximum Read Turnaround Time 24-bit field can encode turnaround times from 0.0 seconds to 16 seconds. F40000h: 0 µs (minimum value) F40001h: 1 µs (resolution) ⋮ F42400h: 16 seconds (maximum value) F42401h: Setting prohibited ⋮ FFFFFFh: Setting prohibited

Note: This register supports I3C secondary master mode and I3C slave mode.

7.8.3.2.70 CCC Exchange Timing Support Information M (Mode) Register (I3Cm_CETSM)

Access Size : 32 bits

Address : <I3C_base> + 0374h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	INAC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FREQ[7:0]								-	-	-	-	-	SPTASYN1	SPTASYN0	SPTSYN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
23 to 16	INAC[7:0]	0h	RW	Inaccuracy Byte This byte represents the maximum variation of the Slave's internal oscillator in 1/10th percent (0.1%) increments, up to 25.5%. 00h: 0.0% 0Fh: 1.5% 1Fh: 3.1% 2Fh: 4.7% 3Fh: 6.3% 4Fh: 7.9% 5Fh: 9.5% 6Fh: 11.1% 7Fh: 12.7% 8Fh: 14.3% 9Fh: 15.9% AFh: 17.5% BFh: 19.1% CFh: 20.7% DFh: 22.3% EFh: 23.9% FFh: 25.5%
15 to 8	FREQ[7:0]	0h	RW	Frequency Byte This byte represents the Slave's internal oscillator frequency in increments of 0.5 MHz (500 kHz), up to 127.5 MHz. 00h: 32.0 KHz 0Fh: 7.5 MHz 1Fh: 15.5 MHz 2Fh: 23.5 MHz 3Fh: 31.5 MHz 4Fh: 39.5 MHz 5Fh: 47.5 MHz 6Fh: 55.5 MHz 7Fh: 63.5 MHz Others: Setting prohibited
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	SPTASYN1	0h	RW	Support Async Mode 1 0b: Async Mode 1 is not supported. 1b: Async Mode 1 is supported.
1	SPTASYN0	0h	RW	Support Async Mode 0 0b: Async Mode 0 is not supported. 1b: Async Mode 0 is supported.

Bit	Bit Name	Initial Value	R/W	Description
0	SPTSYN	0h	RW	Supports Sync Mode 0b: Sync Mode is not supported. 1b: Sync Mode is supported.

Note: This register supports I3C secondary master mode and I3C slave mode.

SPTASYN1 bit (Support Async Mode 1)

Bit mask indicating Async Mode 1 of Timing Control Mode(s) the target Slave supports.

If this bit set (has value 1), then that Slave supports Async Mode 1 of Timing Control Mode.

SPTASYN0 bit (Support Async Mode 0)

Bit mask indicating Async Mode 0 of Timing Control Mode(s) the target Slave supports.

If this bit set (has value 1), then that Slave supports Async Mode 0 of Timing Control Mode.

SPTSYN bit (Supports Sync Mode)

Bit mask indicating Sync Mode of Timing Control Mode(s) the target Slave supports.

If this bit set (has value 1), then that Slave supports Sync Mode of Timing Control Mode.

The Directed CCC provides the framework for the Master to query the Exchange Timing capabilities supported by the I3C Slaves. The Get Exchange Timing Support Information CCC causes the addressed Slave to return four data bytes containing key information on supported current state, and internal oscillator/clock frequency and inaccuracy.

7.8.3.2.71 CCC Exchange Timing Support Information S (State) Register (I3Cm_CETSS)

Bit mask indicating which Timing Control Mode (if any) is currently enabled for the target Slave, and whether any counter overflows have occurred since the most recent previous check. If a Timing Control Mode bit is set (has value 1), then that Slave has currently enabled the corresponding Timing Control Mode. If the Overflow bit is set (has value 1), then that Slave experienced a counter overflow since the most recent previous check.

Access Size : 32 bits
Address : <I3C_base> + 0378h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ICOVF	-	-	-	-	ASYNE[1:0]		SYNE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7	ICOVF	0h	RW	Internal Counter Overflow 0b: Slave has not experienced a counter overflow since the most recent previous check. 1b: Slave experienced a counter overflow since the most recent previous check.
6 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2, 1	ASYNE[1:0]	0h	RW	Async Mode Enable 00b: All Mode disabled 01b: Async Mode 0 enabled 10b: Async Mode 1 enabled 11b: Setting prohibited
0	SYNE	0h	RW	Sync Mode Enable 0b: Sync Mode disabled 1b: Sync Mode enabled

Note: This register supports I3C secondary master mode and I3C slave mode.

ASYNE[1] bit (Async Mode Enabled 1)

Slave Timing Control Async Mode 1 is enabled.

[Setting condition]

- When writing 1b.
- When CETSM.SPTASYN[1] bit = 1b and either of the following 1) or 2) is satisfied.
  1) When receiving SETXTIME CCC(Broadcast) with Defining byte value EFh.
  2) When Slave Address value is its own Slave Address in receiving SETXTIME CCC(Direct) with Defining byte value EFh.

[Clearing condition]

- When writing 0b.
- When CETSM.SPTASYN[1] bit = 1b and either of the following 1) or 2) is satisfied.

1) When receiving SETXTIME CCC(Broadcast) with Defining byte value DFh.
2) When Slave Address value is its own Slave Address in receiving SETXTIME CCC(Direct) with Defining byte value DFh.

ASYNE[0] bit (Async Mode Enabled 2)

Slave Timing Control Async Mode 0 is enabled.

[Setting condition]

- When writing 1b.
- When CETSM.SPTASYN[0] bit = 1b and either of the following 1) or 2) is satisfied.
  1) When receiving SETXTIME CCC(Broadcast) with Defining byte value DFh.
  2) When Slave Address value is its own Slave Address in receiving SETXTIME CCC(Direct) with Defining byte value DFh.

[Clearing condition]

- When writing 0b.
- When CETSM.SPTASYN[0] bit = 1 and either of the following 1) or 2) is satisfied.
  1) When receiving SETXTIME CCC(Broadcast) with Defining byte value EFh.
  2) When Slave Address value is its own Slave Address in receiving SETXTIME CCC(Direct) with Defining byte value EFh.

7.8.3.2.72 Bit Count Register (I3Cm_BITCNT)

Access Size : 32 bits
Address : <I3C_base> + 0380h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	BCNT[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read.
4 to 0	BCNT[4:0]	0h	R	Bit Counter Indicates the number of bits remaining to be transferred. For details on the values, see **Table 7.8-8** and **Table 7.8-9**.

BCNT[4:0] bits (Bit Counter)

These bits function as a counter that indicates the number of bits remaining to be transferred at the detection of a sampling edge on the SCL line.

Table 7.8-8 I^2C transfer / Legacy I^2C transfer

	Master		Slave	
BCNT[4:0]	Address phase	Data phase	Address phase	Data phase
00h	2 to 1 bits	2 to 1 bits	3 to 1 bits	2 to 1 bits
01h	3 bits	3 bits	4 bits	3 bits
02h	4 bits	4 bits	5 bits	4 bits
03h	5 bits	5 bits	6 bits	5 bits
04h	6 bits	6 bits	7 bits	6 bits
05h	7 bits	7 bits	8 bits	7 bits
06h	8 bits	8 bits	9 bits	8 bits
07h	9 bits	9 bits	—	9 bits

Table 7.8-9 I3C transfer

	SDR*1	
BCNT[4:0]	Transmission	Reception
00h	1 bit	2 to 1 bits
01h	2 bits	3 bits
02h	3 bits	4 bits
03h	4 bits	5 bits
04h	5 bits	6 bits

	SDR*1	
BCNT[4:0]	Transmission	Reception
05h	6 bits	7 bits
06h	7 bits	8 bits
07h	8 bits	9 bits
08h	9 bits	—

Note 1. The address phase is the same as in **Table 7.8-8**.

7.8.3.2.73 Normal Queue Status Level Register (I3Cm_NQSTLV)

Access Size : 32 bits

Address : <I3C_base> + 0394h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	IBISCNT[4:0]					IBIQLV[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSPQLV[7:0]								CMDQFLV[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read.
28 to 24	IBISCNT[4:0]	0h	R	Normal IBI Status Count*2 Number of IBI Status entries currently in the IBI Queue.
23 to 16	IBIQLV[7:0]	0h	R	Normal IBI Queue Level*1 Number of buffer entries currently in the IBI Queue.
15 to 8	RSPQLV[7:0]	0h	R	Normal Response Queue Level*1 Number of buffer entries currently in the Response Queue.
7 to 0	CMDQFLV[7:0]	0h	R	Normal Command Queue Free Level*1 Number of free buffer entries currently in the Command Queue. Reset value is the depth of the Command Queue.

Note 1. These bits support all I3C mode.

Note 2. These bits support I3C master mode and I3C secondary master mode.

7.8.3.2.74 Normal Data Buffer Status Level Register (I3Cm_NDBSTLV0)

Access Size : 32 bits

Address : <I3C_base> + 0398h

Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDBLV[7:0]								TDBFLV[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read.
15 to 8	RDBLV[7:0]	0h	R	Normal Receive Data Buffer Level Indicates the number of Receive Data Buffer entries in the Receive Data Queue.
7 to 0	TDBFLV[7:0]	1h	R	Normal Transmit Data Buffer Free Level Indicates the number of free Transmit Data Buffer entries in the Transmit Data Queue. Reset value is the depth of the Transmit Data Queue.

Note: This register supports all I3C mode.

7.8.3.2.75 Normal Receive Status Queue Status Level Register (I3Cm_NRSQSTLV)

Access Size : 32 bits

Address : <I3C_base> + 03C0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RSQLV[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read.
7 to 0	RSQLV[7:0]	0h	R	Normal Receive Status Queue Level

Note: This register supports I3C secondary master mode and I3C slave mode.

7.8.3.2.76 Present State Debug Register (I3Cm_PRSTDBG)

Access Size : 32 bits

Address : <I3C_base> + 03CCh

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	SDOLV	SCOLV	SDILV	SCILV
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read.
3	SDOLV	0h	R	SDA30 Output Level 0b: I3C has driven the SDA30 pin low. 1b: I3C has released the SDA30 pin.
2	SCOLV	0h	R	SCL30 Output Level 0b: I3C has driven the SCL30 pin low. 1b: I3C has released the SCL30 pin.
1	SDILV	0h	R	SDA30 Line Signal Level This bit is used to check the SDA30 Line level, in order to recover from errors and for debugging.
0	SCILV	0h	R	SCL30 Line Signal Level This bit is used to check the SCL30 Line level, in order to recover from errors and for debugging.

SDOLV bit (SDA Output Level) and SCOLV bit (SCL Output Level)

When reading these bits, the state of signals output from I3C can be read.

7.8.3.2.77 Master Error Counters Register (I3Cm_MSERRCNT)

Access Size : 32 bits

Address : <I3C_base> + 03D0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	M2ECNT[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read.
7 to 0	M2ECNT[7:0]	0h	R	M2 Error Counter Counts I3C Type M2 errors on the I3C Bus. Cleared upon read out.

Note: This register supports I3C secondary master mode and I3C slave mode.

7.8.3.2.78 SC1 Capture monitor Register (I3Cm_SC1CPT)

Access Size : 32 bits
Address : <I3C_base> + 03E0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1CPT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read.
15 to 0	SC1CPT[15:0]	0h	R	SC1 Capture

Note: This register supports I3C secondary master mode and I3C slave mode.

SC1C[15:0] bits (SC1 Capture)

- Async Mode 0 (Asynchronous Basic Mode)

  After enabling ATCCNTE.ATCE, SC1 Counter counts up from SC1 count trigger*1 to SCL rise edge next to ACK for the IBI, and capture it as SC1.

- Async Mode 1 (Asynchronous Advanced Mode)

  After enabling ATCCNTE.ATCE, SC1 Counter counts up from SC1 count trigger*1 to the first aME, and capture it as SC1.

Note 1. SW or external trigger can be selected by selection bits.

Remark As the timing control specification, the SC1 counter value is included in the IBI frame as IBI data and is sent to I3C Master, therefore it is not necessary for the I3C Slave to read this register. If the I3C Slave needs to read this register, read it after completing the IBI frame.

7.8.3.2.79 SC2 Capture monitor Register (I3Cm_SC2CPT)

Access Size : 32 bits
Address : <I3C_base> + 03E4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC2CPT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read.
15 to 0	SC2CPT[15:0]	0h	R	SC2 Capture

Note: This register supports I3C secondary master mode and I3C slave mode.

SC2C[15:0] bits (SC2 Capture)

- Async Mode 0 (Asynchronous Basic Mode)

  After enabling ATCCNTE.ATCE, SC2 Counter counts up from SCL rise edge next to ACK for the IBI transmitted from I3C Slave to SCL rise edge next to Tbit after Mandatory Byte, and capture it as SC2.

- Async Mode 1 (Asynchronous Advanced Mode)

  After enabling ATCCNTE.ATCE, SC2 Counter counts up from SCL rise edge next to ACK for the IBI transmitted from I3C Slave to SCL rise edge next to Tbit after Mandatory Byte, and capture it as SC2.

Remark As the timing control specification, the SC2 counter value is included in the IBI frame as IBI data and is sent to I3C Master, therefore it is not necessary for the I3C Slave to read this register. If the I3C Slave needs to read this register, read it after completing the IBI frame.

7.8.4 Operation

7.8.4.1 Data Structures

7.8.4.1.1 Command Descriptor

The write-only Command Descriptor structure is 64 bits in length. The Command Descriptor is put to the Command Queue with writes to the Command Queue Port.

Write to the Command Queue Port in the following order:

1. First write: The least significant DWORD (Command Descriptor Structure Low).
2. Second write: The most significant DWORD (Command Descriptor Structure High).

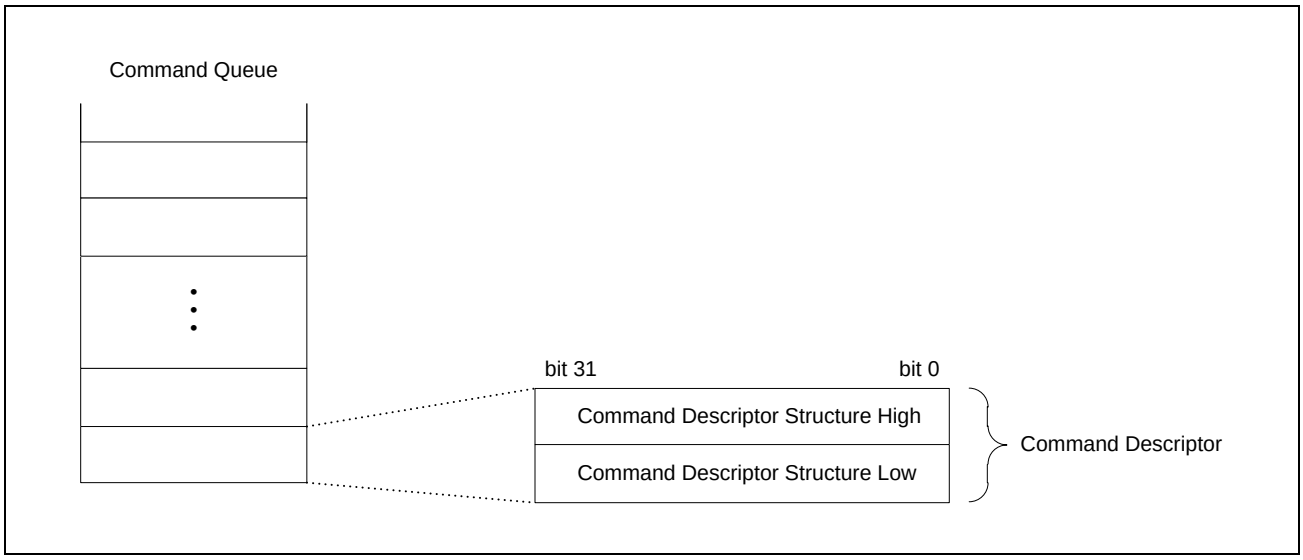


Figure 7.8-2 Command descriptor data structure

I3C provides a Command Descriptor structure for each command type as follows:

- Address Assign Command
- Immediate Transfer Command
- Regular Transfer Command
- Combo Transfer Command
- Internal Control Command

Details are explained in the following sections.

(1) Address Assign Command

This command is used for address assignment (ENTDAA, SETDASA).

Remark When issuing SETAASA CCC, use the Immediate Transfer command.

The I3C provides an address assign command for the following mode:

- I3C Master Mode

Details of the Address Assign command structure are as follows.

Bit position	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
Bit field	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
Bit field	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field	TOC	ROC	DEV_COUNT[3:0]				—	—	—	—	EXT_DEVICE	DEV_INDEX[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field	—	CMD[7:0]								TID[3:0]				CMD_ATTR[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
2:0	CMD_ATTR[2:0]	Command Attributes 0h: XFER: Regular Transfer 1h: IMMED_DATA_XFER: Immediate Data Transfer 2h: ADDR_ASSGN_CMD: Address Assignment Command 3h: WWR_COMBO_XFER: Write + Write/Read Combo Transfer 7h: INTERNAL_CONTROL: Internal Control command Others: Setting prohibited	W
6:3	TID[3:0]	Transaction ID	W
14:7	CMD[7:0]	Transfer Command CCC Value	W
15	—	The write value should be 0b.	W
20:16	DEV_INDEX[4:0]	Device Index	W
21	EXT_DEVICE	Extended Device Index 0b: Use the DATBASm table indicated by DEV_INDEX[4:0]. 1b: Use the EXDATBAS table.	W
25:22	—	The write value should be 0b.	W
29:26	DEV_COUNT[3:0]	Device Count	W

Bit	Symbol	Function	R/W
30	ROC	Response on Completion 0b: NOT_REQUIRED: Response Status is not required. 1b: REQUIRED: Response Status is required.	W
31	TOC	Terminate on Completion 0b: RESTART: Issue Repeated START (Sr) at end of transfer 1b: STOP: Issue STOP (P) at end of transfer	W
63:32	—	The write value should be 0b.	W

CMD_ATTR[2:0] bits (Command Attributes)

Command Type, defining the format of the other fields.

TID[3:0] bits (Transaction ID)

Used as an identification tag for this command. This field shall be populated by the software Driver, and the same value shall be reflected in the Response Descriptor.

CMD[7:0] bits (Transfer Command CCC Value)

Specifies CCC code indicating whether Address Assignment uses ENTDAA or SETDASA commands. The field comprises the entire command code (ENTDAA or SETDASA).

DEV_INDEX[4:0] bits (Device Index)

Indicates the DATBASm table index for the Slave device being addressed with the transfer. Static and device addressing related information are stored to this index in the DATBASm.

DEV_COUNT[3:0] bits (Device Count)

Indicates the number of devices that a dynamic address is assigned to.

ROC bit (Response on Completion)

Controls whether Response Status is sent after successful completion of the Transfer command. The successful completion is read from register NRSPQP. Upon unsuccessful transfer the Response Status is sent.

TOC bit (Terminate on Completion)

Controls what bus condition to issue after the Transfer command completes.

For ENTDAA, a STOP condition is issued regardless of the setting value of TOC. It is meaningful for SETDASA transfers.

When sending SETDASA CCC by TOC = 0b (RESTART), the next command must be set to SETDASA CCC with the Address Assign Command.

When the next command is not the same SETDASA CCC flame, it must be set to TOC = 1b (STOP).

(2) Immediate Transfer Command

This structure directly contains data (max 4 bytes) to be transferred, and as a result is only useful for Transfers/CCCs that write data. This structure shall not be used for Read operations (for example, to receive data).

When transmitting data of 4 bytes or less, use this Immediate Transfer Command to communicate.

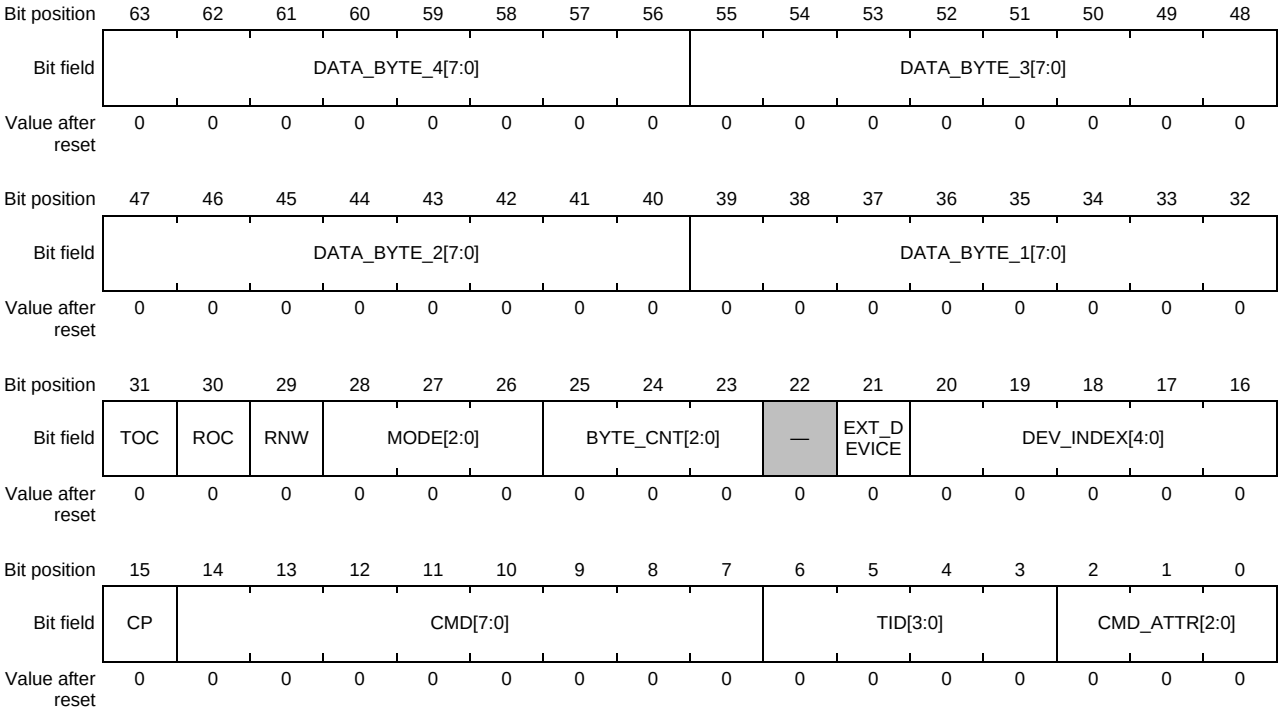
When transmitting data of 5 bytes or more, use the Regular Transfer Command to communicate.

For the Regular Transfer Command, see **(3) Regular Transfer Command**.

I3C provides an Immediate Transfer Command for the following mode:

- I3C Master Mode

Details of the Immediate Transfer Command Structure of each mode are shown in this section.



Bit	Symbol	Function	R/W
2:0	CMD_ATTR[2:0]	Immediate Data Transfer Command Attribute 0h: XFER: Regular Transfer 1h: IMMED_DATA_XFER: Immediate Data Transfer 2h: ADDR_ASSGN_CMD: Address Assignment Command 3h: WWR_COMBO_XFER: Write + Write/Read Combo Transfer 7h: INTERNAL_CONTROL: Internal Control command Others: Setting prohibited	W
6:3	TID[3:0]	Immediate Data Transfer Transaction ID	W
14:7	CMD[7:0]	Immediate Data Transfer CCC Value	W
15	CP	Immediate Data Transfer Command Present 0b: TRANFSER: This structure describes an SDR transfer, so the CMD field is not valid. 1b: CCC: This structure describes a CCC transfer, so the CMD field is valid.	W
20:16	DEV_INDEX[4:0]	Immediate Data Transfer Device Index	W

Bit	Symbol	Function	R/W
21	EXT_DEVICE	Immediate Data Transfer Extended Device Index 0b: Use the DATBASm table indicated by DEV_INDEX[4:0]. 1b: Use the EXDATBAS table.	W
22	—	The write value should be 0b.	W
25:23	BYTE_CNT[2:0]	Immediate Data Transfer Byte Count 0h: No payload 1h: 1 byte is valid. 2h: 2 bytes are valid. 3h: 3 bytes are valid. 4h: 4 bytes are valid. Others: Setting prohibited	W
28:26	MODE[2:0]	Immediate Data Transfer Mode and Speed Values 0h: I3C SDR0/Data rate: STDBR (I3C mode) I^{2}C Message 0/Data rate: STDBR (I^{2}C mode) 1h: I3C SDR1/Data rate: EXTBR (I3C mode) I^{2}C Message 0/Data rate: EXTBR (I^{2}C mode) 2h: I3C SDR2/Data rate: STDBR × 2 (I3C mode) Reserved (I^{2}C mode) 3h: I3C SDR3/Data rate: EXTBR × 2 (I3C mode) Reserved (I^{2}C mode) 4h: I3C SDR4/Data rate: EXTBR × 4 (I3C mode) Reserved (I^{2}C mode) Others: Setting prohibited	W
29	RNW	Immediate Data Transfer R/W 0b: WRITE: Write transfer 1b: READ: Read transfer	W
30	ROC	Immediate Data Transfer Response on Completion 0b: NOT_REQUIRED: Response Status is not required. 1b: REQUIRED: Response Status is required.	W
31	TOC	Immediate Data Transfer Terminate on Completion 0b: RESTART: Issue Repeated START (Sr) at end of data transfer 1b: STOP: Issue STOP (P) at end of data transfer	W
39:32	DATA_BYTE_1[7:0]	Immediate Data Transfer Data Byte 1 Direct argument	W
47:40	DATA_BYTE_2[7:0]	Immediate Data Transfer Data Byte 2 Direct argument	W
55:48	DATA_BYTE_3[7:0]	Immediate Data Transfer Data Byte 3 Direct argument	W
63:56	DATA_BYTE_4[7:0]	Immediate Data Transfer Data Byte 4 Direct argument	W

CMD_ATTR[2:0] bits (Immediate Data Transfer Command Attribute)

Command Type, defining the format of the other fields.

TID[3:0] bits (Immediate Data Transfer Transaction ID)

Used as an identification tag for this command. This field shall be populated by the software Driver, and the same value shall be reflected in the Response Descriptor.

CP bit (Immediate Data Transfer Command Present)

Indicates whether CMD field is valid for CCC Transfer.

DEV_INDEX[4:0] bits (Immediate Data Transfer Device Index)

Indicates the DATBASm Table index for the Slave Device being addressed with the transfer. Static and Device addressing related information will be stored to this index in the DATBASm.

BYTE_CNT[2:0] bits (Immediate Data Transfer Byte Count)

Number of valid data bytes to use in this Immediate Data Transfer Descriptor.

This field must be set to non-zero value, except for CCCs that does not have payload defined.

MODE[2:0] bits (Immediate Data Transfer Mode and Speed Values)

Sets the mode and speed for the I3C or I^2C transfer.

Interpretation of this field depends on whether the Device is in I3C Mode vs. I^2C Mode (see the DEVICE field in the DATBASm Table entry indexed by field DEV_INDEX).

RNW bit (Immediate Data Transfer R/W)

Identifies direction of the transfer.

This field shall always be set to 0b, because Immediate transfers are valid for Write transactions only.

ROC bit (Immediate Data Transfer Response on Completion)

Controls whether Response Status is required after successful completion of the data transfer command. The successful completion shall be read from NRSPQP register. Upon unsuccessful transfer the Response Status shall always be sent.

TOC bit (Immediate Data Transfer Terminate on Completion)

Controls what Bus condition is issued after completion of the data transfer.

When sending Direct CCC by TOC = 0b (RESTART), next command must be set to same Direct CCC.

When the next command is not the same Direct CCC, must be set to TOC = 1b (STOP).

(3) Regular Transfer Command

This structure does not contain data to be transferred.

For Master Mode, the data buffer is available through Transfer Data Queue Port (Receive Data Queue Port and Transmit Data Queue Port).

When transmitting data of 5 bytes or more, use this Regular Transfer Command to communicate.

When transmitting data of 4 bytes or less, use the Immediate Transfer Command to communicate.

For the Regular Transfer Command, see **(2) Immediate Transfer Command**.

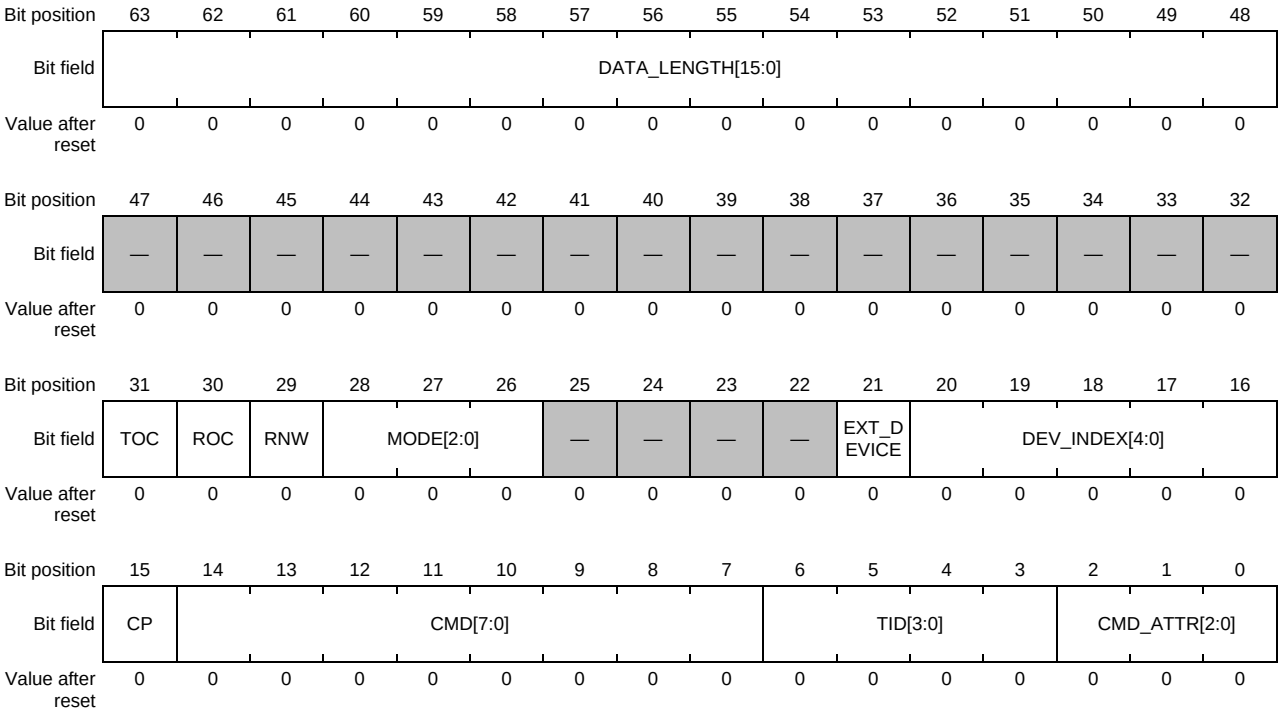
For I3C Slave Mode, the IBI Payload buffer is available through IBI Status Queue Port.

I3C provides a Regular Transfer Command for each mode below.

- I3C Master Mode
- I3C Slave Mode

Details of the regular transfer command structure of each mode are shown below.

(1) I3C Master Mode



Bit	Symbol	Function	R/W
2:0	CMD_ATTR[2:0]	Data Transfer Command Attribute Command Type, defining the format of the other fields. Values: 0h: XFER: Regular Transfer 1h: IMMED_DATA_XFER: Immediate Data Transfer 2h: ADDR_ASSGN_CMD: Address Assignment Command 3h: WWR_COMBO_XFER: Write + Write/Read Combo Transfer 7h: INTERNAL_CONTROL: Internal Control command Others: Setting prohibited	W
6:3	TID[3:0]	Data Transfer Transaction ID Identification tag for this command	W

Bit	Symbol	Function	R/W
14:7	CMD[7:0]	Data Transfer CCC Code Value Specifies the I3C Command code	W
15	CP	Data Transfer Command Present 0b: TRANFSER: This structure describes an SDR transfer, so the CMD field is not valid. 1b: CCC: This structure describes a CCC transfer, so the CMD field is valid.	W
20:16	DEV_INDEX[4:0]	Data Transfer Device Index	W
21	EXT_DEVICE	Data Transfer Extended Device Index 0b: Use the DATBASm Table indicated by DEV_INDEX[4:0]. 1b: Use the EXDATBAS table.	W
25:22	—	The write value should be 0b.	W
28:26	MODE[2:0]	Data Transfer Speed and Mode 0h: I3C SDR0/Data rate: STDBR (I3C mode) I²C Message 0/Data rate: STDBR (I²C mode) 1h: I3C SDR1/Data rate: EXTBR (I3C mode) I²C Message 0/Data rate: EXTBR (I²C mode) 2h: I3C SDR2/Data rate: STDBR × 2 (I3C mode) Reserved (I²C mode) 3h: I3C SDR3/Data rate: EXTBR × 2 (I3C mode) Reserved (I²C mode) 4h: I3C SDR4/Data rate: EXTBR × 4 (I3C mode) Reserved (I²C mode) Others: Setting prohibited	W
29	RNW	Data Transfer R/W 0b: WRITE: Write transfer 1b: READ: Read transfer	W
30	ROC	Data Transfer Response on Completion 0b: NOT_REQUIRED: Response Status is not required. 1b: REQUIRED: Response Status is required.	W
31	TOC	Data Transfer Terminate on Completion 0b: RESTART: Issue Repeated START (Sr) at end of transfer 1b: STOP: Issue STOP (P) at end of transfer	W
47:32	—	The write value should be 0.	W
63:48	DATA_LENGTH [15:0]	Data Transfer Data Length Indicates the number of bytes to be transferred. This field must be set to non-zero value, except for CCCs that does not have payload defined.	W

CMD_ATTR[2:0] bits (Data Transfer Command Attribute)

Command Type, defining the format of the other fields.

TID[3:0] bits (Data Transfer Transaction ID Identification tag for this command)

Used as an identification tag for this command. This field shall be populated by the software Driver, and the same value shall be reflected in the Response Descriptor.

CP bit (Data Transfer Command Present)

Indicates whether he contents of the CMD field is valid for a CCC Transfer.

DEV_INDEX[4:0] bits (Data Transfer Device Index)

Indicates the DATBASm Table index for the Slave Device being addressed with the transfer. Static and Device addressing related information will be stored to this index in the DATBASm.

MODE[2:0] bits (Data Transfer Speed and Mode)

Sets the mode and speed for the I3C or I^2C transfer.

Interpretation of this field depends on whether the Device is in I3C Mode vs. I^2C Mode (see the DEVICE field in the DATBASm Table entry indexed by field DEV_INDEX).

RNW bit (Data Transfer R/W)

Identifies direction of the transfer.

ROC bit (Data Transfer Response on Completion)

Controls whether Response Status is required after successful completion of the transfer command. The successful completion shall be read from NRSPQP register. Upon unsuccessful transfer the Response Status shall always be sent.

TOC bit (Data Transfer Terminate on Completion)

Controls what Bus condition will be issued after completion of the transfer.

When sending Direct CCC by TOC = 0b (RESTART), next command must be set to same Direct CCC.

When the next command is not the same Direct CCC, must be set to TOC = 1b (STOP).

DATA_LENGTH[15:0] bits (Data Transfer Data Length)

Number of valid data bytes to use in this Regular Transfer Descriptor.

This field must be set to non-zero value, except for CCCs that does not have payload defined.

Length setting of GETMXDS command should be fixed to 5.

(2) I3C Slave Mode

Bit position	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
Bit field	DATA_LENGTH[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
Bit field	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field	—	ROC	RNW	—	—	—	—	—	—	ITS	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field	HJ	—	—	—	—	—	—	—	—	TID[3:0]				CMD_ATTR[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
2:0	CMD_ATTR[2:0]	Data Transfer Command Attribute Command Type, defining the format of the other fields. Values: 0h: XFER: Regular Transfer 1h: IMMED_DATA_XFER: Immediate Data Transfer 2h: ADDR_ASSGN_CMD: Address Assignment Command 3h: WWR_COMBO_XFER: Write + Write/Read Combo Transfer 7h: INTERNAL_CONTROL: Internal Control command Others: Setting prohibited	W
6:3	TID[3:0]	Data Transfer Transaction ID Identification tag for this command	W
14:7	—	The write value should be 0b.	W
15	HJ	Data Transfer Hot-Join Event 0b: Slave Interrupt Request or Mastership Request, so the RNW field is valid. 1b: Hot-Join Event, so the RNW field is not valid.	W
21:16	—	The write value should be 0b.	W
22	ITS	Include timestamp for Async Mode 0b: Do not include timestamp. 1b: Include timestamp.	W
28:23	—	The write value should be 0b.	W
29	RNW	Data Transfer R/W 0b: WRITE: Write transfer (Mastership Request) 1b: READ: Read transfer (Slave Interrupt Request)	W
30	ROC	Data Transfer Response on Completion 0b: NOT_REQUIRED: Response Status is not required. 1b: REQUIRED: Response Status is required.	W
47:31	—	The write value should be 0b.	W
63:48	DATA_LENGTH [15:0]	Data Transfer Data Length Indicates the number of bytes to be transferred. This field must be set to non-zero value, except for CCCs that does not have payload defined.	W

CMD_ATTR[2:0] bits (Data Transfer Command Attribute)

Command Type, defining the format of the other fields.

TID[3:0] bits (Data Transfer Transaction ID Identification tag for this command)

Used as an identification tag for this command. This field shall be populated by the software Driver, and the same value shall be reflected in the Response Descriptor.

HJ bit (Data Transfer Hot-Join Event)

Indicates whether Hot-Join Event is valid in this IBI Data transfer.

RNW bit (Data Transfer R/W)

Identifies direction of the transfer.

ROC bit (Data Transfer Response on Completion)

Controls whether Response Status is required after successful completion of the transfer command. The successful completion shall be read from NRSPQP register. Upon unsuccessful transfer the Response Status shall always be sent.

(4) Combo Transfer Command

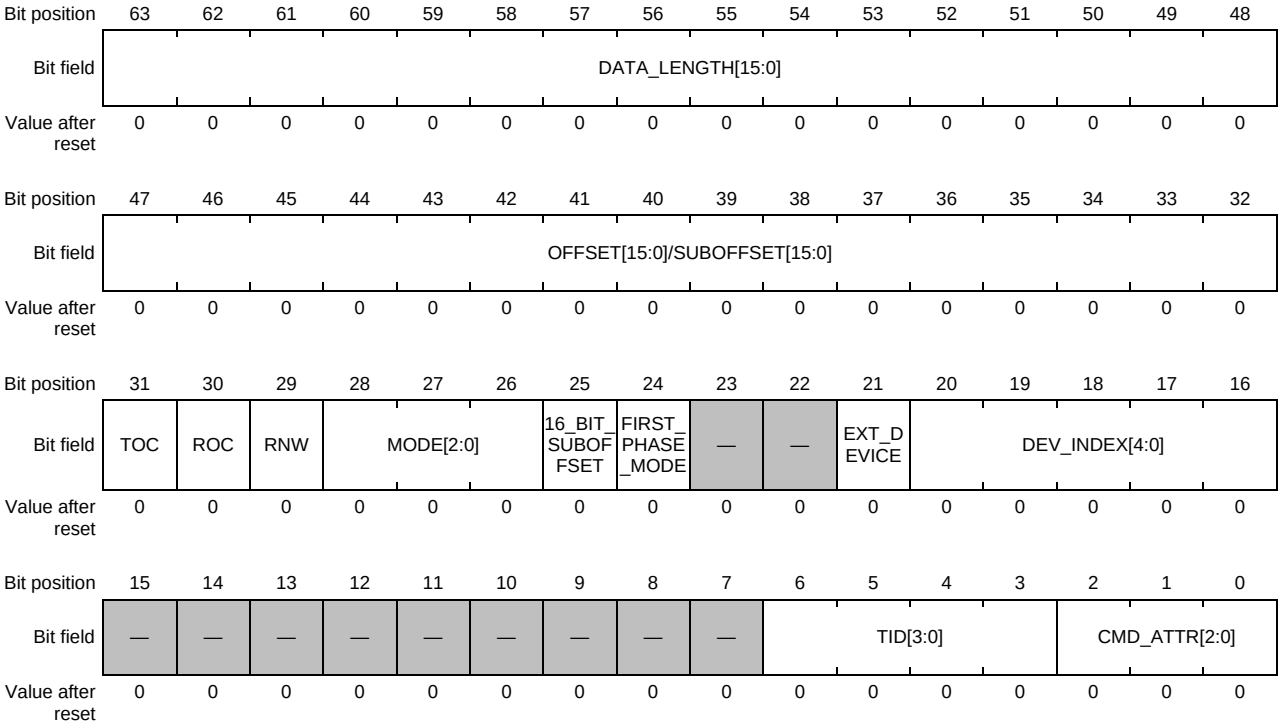
This structure contains a combined Write + Read/Write operation.

The data buffer is available through Transfer Data Queue Port (Receive Data Queue Port and Transmit Data Queue Port).

I3C provides a Combo Transfer Command for the following mode:

- I3C Master Mode

Details of the Combo Transfer Command Structure of each mode are as follows.



Bit	Symbol	Function	R/W
2:0	CMD_ATTR[2:0]	Combo Transfer Command Attribute Command Type, defining the format of the other fields. 0h: XFER: Regular Transfer 1h: IMMED_DATA_XFER: Immediate Data Transfer 2h: ADDR_ASSGN_CMD: Address Assignment Command 3h: WWR_COMBO_XFER: Write + Write/Read Combo Transfer 7h: INTERNAL_CONTROL: Internal Control command Others: Setting prohibited	W
6:3	TID[3:0]	Combo Transfer Transaction ID Identification tag for the command	W
15:7	—	The write value should be 0b.	W
20:16	DEV_INDEX[4:0]	Combo Transfer Device Index	W
21	EXT_DEVICE	Combo Transfer Extended Device Index 0b: Use the DATBASm table indicated by DEV_INDEX[4:0]. 1b: Use the EXDATBAS table.	W
23:22	—	The write value should be 0b.	W

Bit	Symbol	Function	R/W
24	FIRST_PHASE_MODE	Combo Transfer First Phase Mode 0b: SDR: First phase is executed in SDR mode. 1b: MODE: First phase is executed in the mode indicated by the MODE field.	W
25	16_BIT_SUBOFFSET	Combo Transfer Sub Offset Size 0b: 8_BIT_SUBOFFSET: Sub-offset is 8-bits long. Value is encoded in Lower Byte of OFFSET/SUBOFFSET field. 1b: 16_BIT_SUBOFFSET: Sub-offset is 16-bits long.	W
28:26	MODE[2:0]	Combo Transfer Speed and Mode Values for I3C Mode 0h: I3C SDR0/Data rate: STDBR 1h: I3C SDR1/Data rate: EXTBR 2h: I3C SDR2/Data rate: STDBR × 2 3h: I3C SDR3/Data rate: EXTBR × 2 4h: I3C SDR4/Data rate: EXTBR × 4 Others: Setting prohibited	W
29	RNW	Combo Transfer R/W Identifies direction of the transfer 0b: WRITE: Write transfer 1b: READ: Read transfer	W
30	ROC	Combo Transfer Response on Completion 0b: NOT_REQUIRED: Response Status is not required. 1b: REQUIRED: Response Status is required.	W
31	TOC	Combo Transfer Terminate on Completion 0b: RESTART: Issue Repeated START (Sr) at end of transfer 1b: STOP: Issue STOP (P) at end of transfer	W
47:32	OFFSET[15:0]/SUBOFFSET[15:0]	Combo Transfer Offset/Sub-Offset Offset of the target operation	W
63:48	DATA_LENGTH[15:0]	Combo Transfer Data Length Number of bytes to be transferred. This field must be set to non-zero value.	W

CMD_ATTR[2:0] bits (Combo Transfer Command Attribute)

Command Type, defining the format of the other fields.

TID[3:0] bits (Combo Transfer Transaction ID Identification tag for the command)

Used as an identification tag for this command. This field shall be populated by the software Driver, and the same value shall be reflected in the Response Descriptor.

DEV_INDEX[4:0] bits (Combo Transfer Device Index)

Indicates the DATBASm table index for the Slave Device being addressed with the transfer. Static and Device addressing related information will be stored to this index in the DATBASm.

FIRST_PHASE_MODE bits (Combo Transfer First Phase Mode)

Indicates whether the first phase of the Combo Transfer is executed in SDR Mode, vs. the Mode indicated by the MODE field.

MODE[2:0] bits (Combo Transfer Speed and Mode Values for I3C Mode)

Sets the mode and speed for the I3C or I^2C transfer.

Interpretation of this field depends on whether the Device is in I3C Mode vs. I^2C Mode (see the DEVICE field in the DATBASm Table entry indexed by field DEV_INDEX).

RNW bit (Combo Transfer R/W Identifies direction of the transfer)

Identifies direction of the transfer.

ROC bit (Combo Transfer Response on Completion)

Controls whether Response Status is required after successful completion of the data transfer command. The successful completion shall be read from NRSPQP register. Upon unsuccessful transfer the Response Status shall always be sent.

TOC bit (Combo Transfer Terminate on Completion)

Controls what Bus condition is issued after completion of the data transfer.

DATA_LENGTH[15:0] bit (Combo Transfer Data Length Number of bytes to be transferred. This field must be set to non-zero value.)

Number of valid data bytes to use in this Combo Transfer Descriptor.

This field must be set to non-zero value.

(5) Internal Control Command

This structure is used for controlling I3C itself (not for transfer commands).

I3C provides an Internal Control Command for the following mode:

- I3C Master Mode

Details of the Internal Control Command Structure are as follows:

Bit position	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
Bit field	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
Bit field	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field	—	—	—	ON_OFF	MIPI_CMD[3:0]				—	TID[3:0]				CMD_ATTR[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
2:0	CMD_ATTR[2:0]	Command Attribute*2 Command Type, defining the format of the other fields. 0h: XFER: Regular Transfer 1h: IMMED_DATA_XFER: Immediate Data Transfer 2h: ADDR_ASSGN_CMD: Address Assignment Command 3h: WWR_COMBO_XFER: Write + Write/Read Combo Transfer 7h: INTERNAL_CONTROL: Internal Control command Others: Setting prohibited	W
6:3	TID[3:0]	Transaction ID Identification tag for the command	W
7	—	The write value should be 0b.	W
11:8	MIPI_CMD[3:0]	MIPI Alliance Command 00h: NoOp, so the ON_OFF field is not valid. 02h: Include 7E (IBA), so the ON_OFF field is valid. Others: Setting prohibited	W
12	ON_OFF	Bus Instance 7E On/Off*1 Enables or disables automatic transmission of the I3C Broadcast Header after every START condition on this I3C Bus instance. 0b: IBA_INCLUDE off 1b: IBA_INCLUDE on	W
63:13	—	The write value should be 0b.	W

Note 1. The IBA_INCLUDE on state set by MIPI_CMD [3:0] = 2h and ON_OFF = 1 is cleared by setting RSTCTL.INTLRST to 1b.

Note 2. The Response descriptor is not stored when the Internal Control Command is executed.

7.8.4.1.2 Response Descriptor

The Response Descriptor is a read-only structure describing the success or failure of a command, and the amount of data transferred.

The Response Descriptor is read from Response Queue with reads from Response Queue Port.

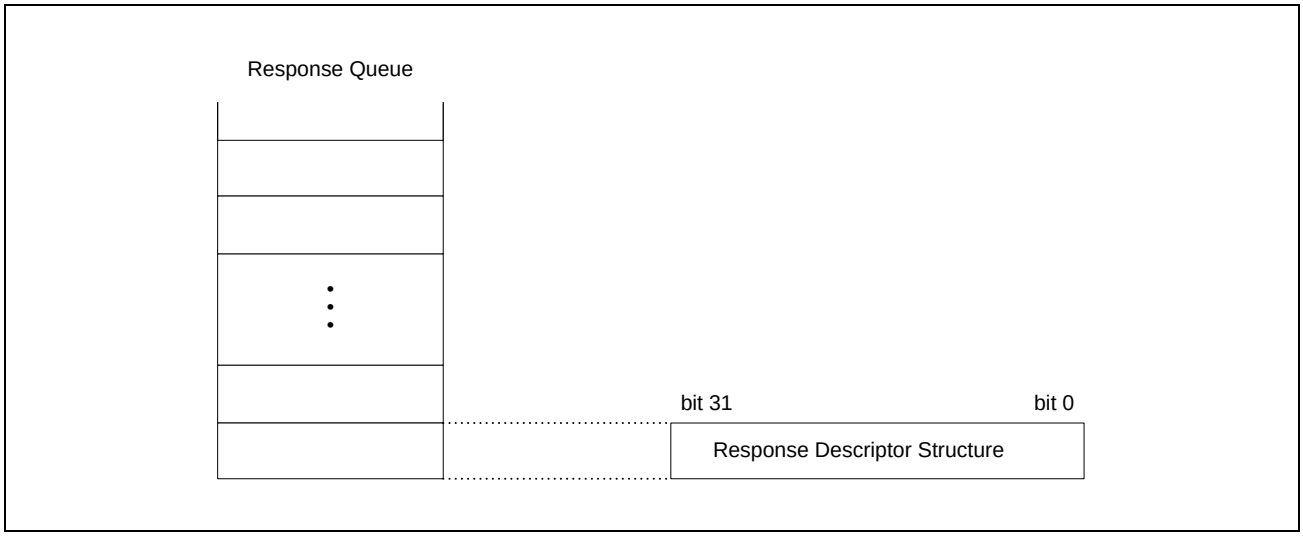


Figure 7.8-3 Response descriptor data structure

I3C provides a Response Descriptor for the following modes:

- I3C Master Mode
- I3C Slave Mode

Details of the Response Descriptor structure of each mode are shown in the following sections.

(1) I3C Master Mode

Bit position	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field	ERR_STATUS[3:0]				TID[3:0]				—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field	DATA_LENGTH[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
15:0	DATA_LENGTH [15:0]	Data Length/Device Count The meaning of this field depends on the context: For Write Transfer: Remaining data length (in bytes) For Read Transfer: Received data length (in bytes) For Address Assignment: Remaining Device count	R
23:16	—	These bits are read as 0b.	R

Bit	Symbol	Function	R/W
27:24	TID[3:0]	Command/Response Transaction ID Identification tag for the command. This value shall match one of commands sent on the Bus. 0h-7h: Valid Transaction IDs Others: Setting prohibited	R
31:28	ERR_STATUS[3:0]	MIPI Alliance Command 0h: SUCCESS: Transfer successful, no error 1h: CRC: CRC Error 2h: PARITY: Parity Error 3h: FRAME: Frame Error 4h: ADDR_HEADER: Address Header Error 5h: NACK: Address NACKed or Dynamic Address Assignment NACKed 6h: OVL: Receive Overflow or Transfer Underflow Error 8h: ABORTED: Aborted 9h: I^2C_WR_DATA_NACK: NACK received for the I^2C Write Data transfer Ah: NOT_SUPPORTED: Command with specific parameters not supported by I3C implementation (for example, specific Internal Control codes may not be supported) Others: Setting prohibited	R

Note: In I3C Master mode, when an abnormal command with a specific parameter that is not supported is stored in Command Descriptor, it is indicated as NOT_SUPPORTED (Ah) in ERR_STATUS [3:0].

(2) I3C Slave Mode

Bit position	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field	ERR_STATUS[3:0]				TID[3:0]				—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field	DATA_LENGTH[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
15:0	DATA_LENGTH [15:0]	Data Length Remaining data length (in bytes) for Slave Interrupt Request	R
23:16	—	These bits are read as 0b.	R
27:24	TID[3:0]	Command/Response Transaction ID Identification tag for the command. This value matches one of commands sent on the bus. 0h-7h: Valid Transaction IDs Others: Setting prohibited	R

Bit	Symbol	Function	R/W
31:28	ERR_STATUS[3:0]	Response Error Status 0h: SUCCESS: Transfer successful, no error. 3h: FRAME: Frame Error 4h: ADDR_HEADER: Address Header Error 5h: NACK: Address NACK'ed or Dynamic Address Assignment NACK'ed 6h: OVL: Receive Overflow or Transfer Underflow Error 8h: ABORTED: Aborted Ah: NOT_SUPPORTED: Command with specific parameters not supported by I3C implementation (for example, specific Internal Control codes may not be supported) Others: Setting prohibited	R

Note: In I3C Slave mode, it is indicated as NOT_SUPPORTED (Ah) in ERR_STATUS[3:0] in the following cases:
- When an abnormal command with a specific parameter that is not supported is stored in the Command Descriptor.
- When the IBI to be transmitted is disabled in the CSECMD register.
- After the normal command for IBI transmission is prepared in the Command Queue, when that IBI is disabled in the CSECMD register by the DISEC CCC frame from the I3C Master.

7.8.4.1.3 IBI Status Descriptor

The IBI Status Descriptor is a read-only structure describing an IBI event received from a Slave device on the I3C Bus.

The IBI Status Descriptor is read from IBI Status Queue with reads from IBI Status Queue Port.

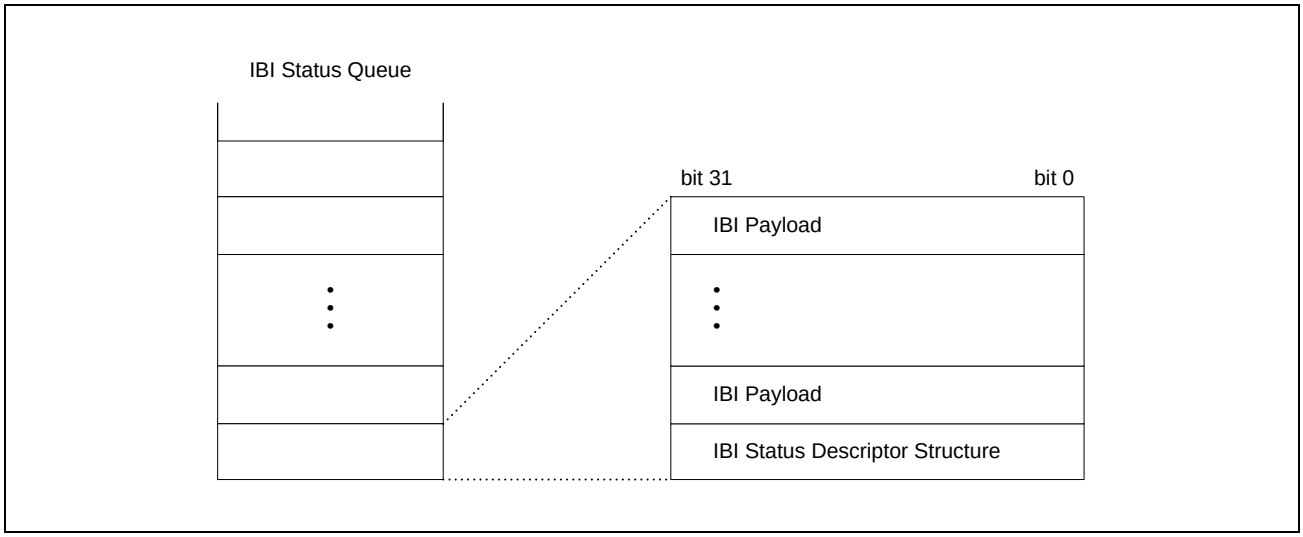


Figure 7.8-4 IBI status descriptor data structure

I3C provides a IBI Status Descriptor for the following mode:

- I3C Master Mode

Details of the IBI Status Descriptor Structure are as follows.

Bit position	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field	IBI_ST	—	—	ERR_STATUS[2:0]			TS	LAST_STATUS	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field	IBI_ID[7:0]								DATA_LENGTH[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
7:0	DATA_LENGTH [7:0]	IBI Data Length Number of data bytes in IBI Data.	R
15:8	IBI_ID[7:0]	IBI Received ID The meaning of this field depends on the context: For Slave Interrupt or Master Request: Bits 15:9 contain the Slave's Device Address, and bit 8 contains the R/W bit. For Hot-Join IBI: Bits 15:8 contain the Hot-Join ID for the IBI.	R
23:16	—	These bits are read as 0b. The write value should be 0b.	R
24	LAST_STATUS	Last IBI Status Last IBI status for the IBI transaction.	R

Bit	Symbol	Function	R/W
25	TS	IBI Time-stamp Present Indicates whether a time-stamp is available for the IBI. 0b: OFF: IBI is not time-stamped. 1b: ON: IBI is time-stamped.	R
28:26	ERR_STATUS[2:0]	IBI Error Status 0h: SUCCESS 3h: ERROR: FRAME (Frame Error) 4h: ERROR: ADDR_HEADER (Address Header Error) 5h: NACK: Address NACKed 7h: ERROR: ABORT (Aborted to Master) Others: Setting prohibited	R
30:29	—	These bits are read as 0b.	R
31	IBI_ST	IBI Received Status Indicates how the received IBI was handled. 0b: The IBI was handled with ACK. 1b: NACK: The IBI was handled with NACK, and then Auto-Disabled.	R

LAST_STATUS bits (Last IBI Status)

Even if LAST_STATUS is set to 0b, the software driver still evaluates the data payload length by examining the CHUNKS field.

7.8.4.1.4 Receive Status Descriptor

The Receive Status Descriptor is a read-only structure describing the success or failure of read/write operation from the master, and the amount of data transferred.

The Receive Status Descriptor is read from Receive Status Queue with reads from Receive Status Queue Port.

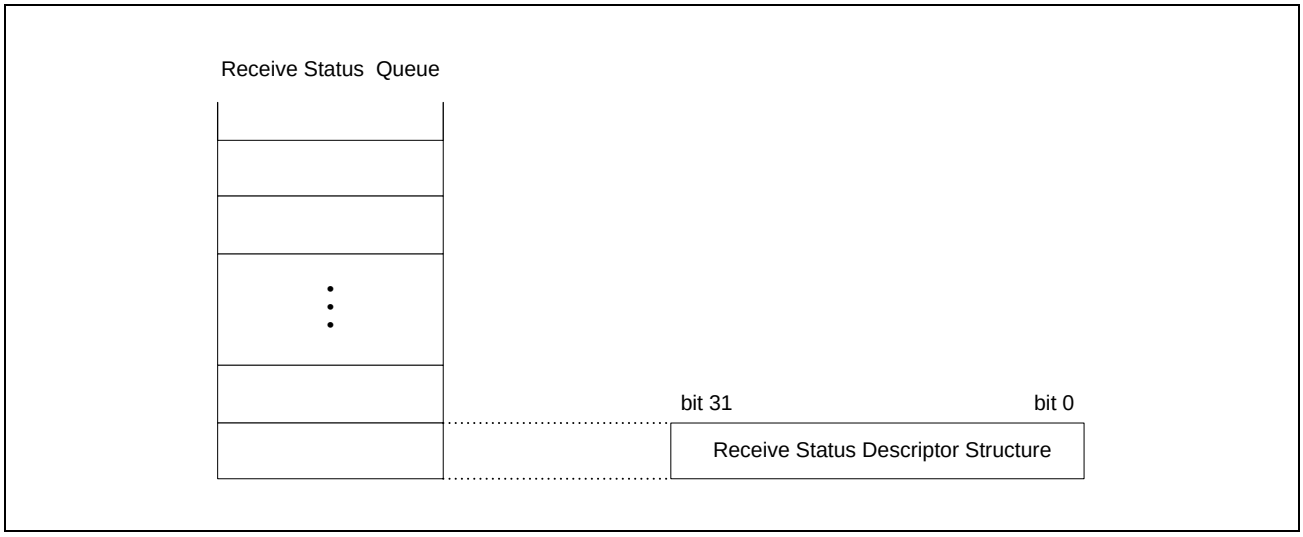


Figure 7.8-5 Receive status descriptor data structure

I3C provides a Receive Status Descriptor for the following mode:

- I3C Slave Mode

Details of the Receive Status Descriptor structure of each mode are as follows.

Bit position	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field	DEV_INDEX[2:0]			TRANSFER_TYPE[1:0]		ERR_STATUS[2:0]			CMD[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field	DATA_LENGTH[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
15:0	DATA_LENGTH [15:0]	Data Length The meaning of this field depends on the context. For Write Transfer: Received data length (in bytes) For Read Transfer: Transmitted data length (in bytes)	R

Bit	Symbol	Function	R/W
23:16	CMD[7:0]	The contents are different depending on the operation mode. Details are follows: [SDR Private Message Mode] bit 23: R/W Type bit 22 to 20: Setting prohibited bit 19: I3C_I^{2}C Type bit 18 to 16: Setting prohibited [SDR CCC Mode] CCC code[7:0]	R
26:24	ERR_STATUS[2:0]	Error Status 0h: SUCCESS 1h: ERROR: CRC (CRC Error) 2h: ERROR: PARITY (Parity Error) 3h: ERROR: FRAME (Frame Error) 4h: ERROR: ADDR_HEADER (Address Header Error) 5h: ERROR: NACK (Slave NACKed) 6h: ERROR: OVL (FIFO Overflow/Underflow) 7h: ERROR: ABORT (Aborted to Master)	R
28:27	TRANSFER_TYPE [1:0]	Transfer Type 0 0b: I3C SDR/I^{2}C Message 0 1b: I3C CCC 1 0b: Setting prohibited 1 1b: Setting prohibited	R
31:29	DEV_INDEX[2:0]	Device Index Indicates the SVDVADn index for the response with the transfer.	R

7.8.4.2 Details of Function

7.8.4.2.1 Operation Mode

The support relationship between the mode select (I3C mode / I²C mode) and operation mode (Master / Slave) on the I3C bus or the I²C bus is shown in **Table 7.8-10**.

Table 7.8-10 Support of operating mode

I3C/I²C Bus	I3C mode		I²C mode	
	Master	Slave	Master	Slave
I3C Bus	✓	✓	—	✓
I²C Bus	—	—	✓	✓

Note: ✓: Supported
—: Un-Supported

(1) Master Mode Operation

(a) I²C Master Operation

(a-1) Data Write Transfer (Single Buffer transfer)

In master transmit operation, I3C outputs the SCL clock and transmitted data signals as the master device, and the slave device returns acknowledgments. **Figure 7.8-114** shows an example of usage of master transmission and **Figure 7.8-6** to **Figure 7.8-8** show the timing of operations in master transmission.

The following describes the procedure and operations for master transmission.

1. Initial settings. For details, see **7.8.4.3.1 Initial Setting Flow**.
2. Read the BCST.BFREF flag to check that the bus is open, and then set the CNDCTL.STCND bit to 1b (START condition issuance request). Upon receiving the request, I3C issues a START condition. At the same time, the BFREF flag bit is automatically set to 0b, the BST.STCNDDF flag is automatically set to 1b and the STCND bit is automatically set to 0b. At this time, if the START condition is detected and the internal levels for the SDA output state and the levels on the SDA line have matched while the STCND bit = 1b, I3C recognizes that issuing of the START condition as requested by the STCND bit has been successfully completed, and bits CRMS and TRMD in the PRSST register are automatically set to 1b, placing I3C in master transmit mode. The NTST.TDBEF0 flag is also automatically set to 1b in response to setting of the TRMD bit to 1b.
3. Check that the NTST.TDBEF0 flag = 1b, and then write the value for transmission (the slave address and the R/W# bit) to the NTDTBP0 register. Once the data for transmission are written to the NTDTBP0 register, the TDBEF0 flag is automatically set to 0b, the data are transferred from the Normal Transmit Data Buffer to the Shift Register, and the TDBEF0 flag is again set to 1b. After the byte containing the slave address and R/W# bit has been transmitted, the value of the TRMD bit is automatically updated to select master transmit or master receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 0, I3C continues in master transmit mode. Because the BST.NACKDF flag being 1b at this time indicates that no slave device recognized the address or there was an error in communications, write 1b to the CNDCTL.SPCND bit to issue a STOP condition. For data transmission with an address in the 10-bit format, start by writing 1111 0, the 2 higher-order bits of the slave address, and W to the NTDTBP0 register as the first address transmission. Then, as the second address transmission, write the 8 lower-order bits of the slave address to the NTDTBP0 register.

4. After confirming that the NTST.TDBEF0 flag = 1b, write the data for transmission to the NTDTBP0 register. I3C automatically holds the SCL line low until the data for transmission are ready or a STOP condition is issued.
5. After all bytes of data for transmission have been written to the NTDTBP0 register, wait until the value of the BST.TENDF flag returns to 1, and then set the CNDCTL.SPCND bit to 1b (STOP condition issuance request). Upon receiving a STOP condition issuance request, I3C issues the STOP condition.
6. Upon detecting the STOP condition, I3C automatically sets bits CRMS and TRMD in the PRSST register to 00b and enters slave receive mode. Furthermore, it automatically sets the TDBEF0 and TENDF flags to 0b, and sets the BST.SPCNDDF flag to 1b.
7. After checking that the BST.SPCNDDF flag = 1b, set the BST.NACKDF and SPCNDDF flags to 0b for the next transfer operation.

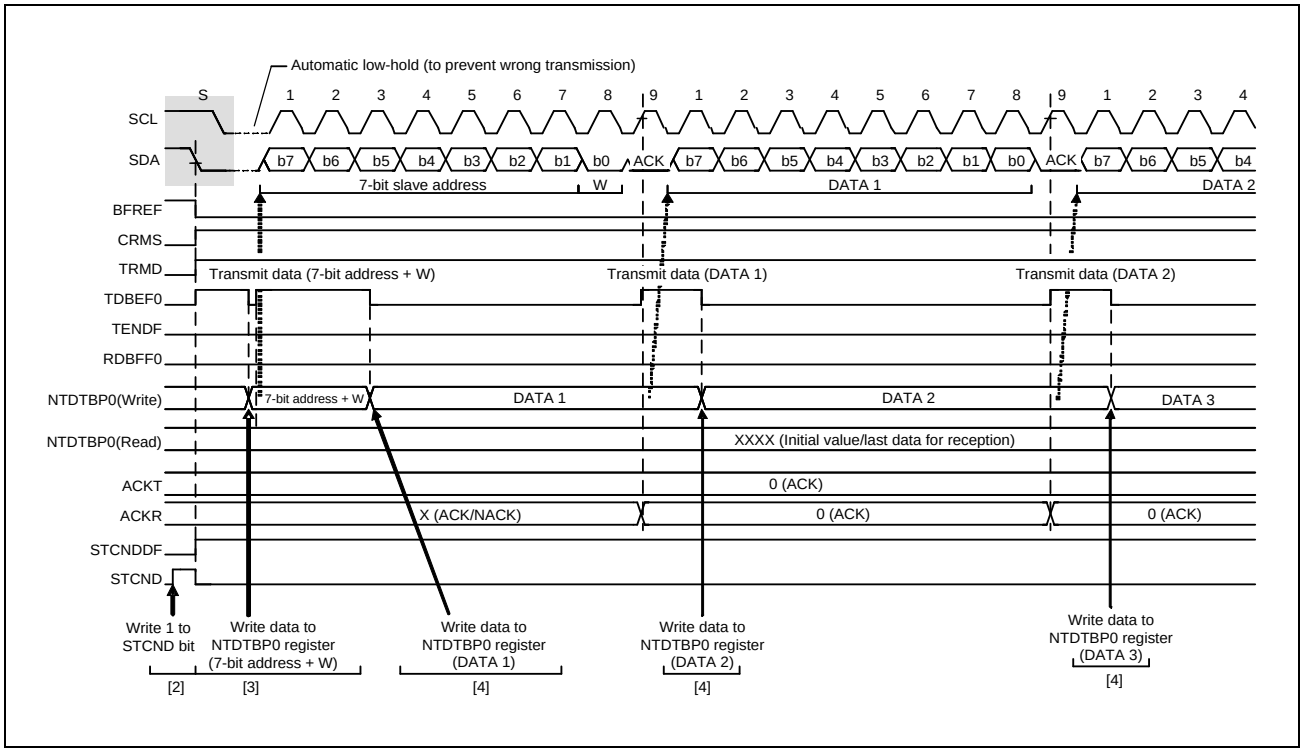


Figure 7.8-6 Master transmit operation timing (1) (7-bit address format)

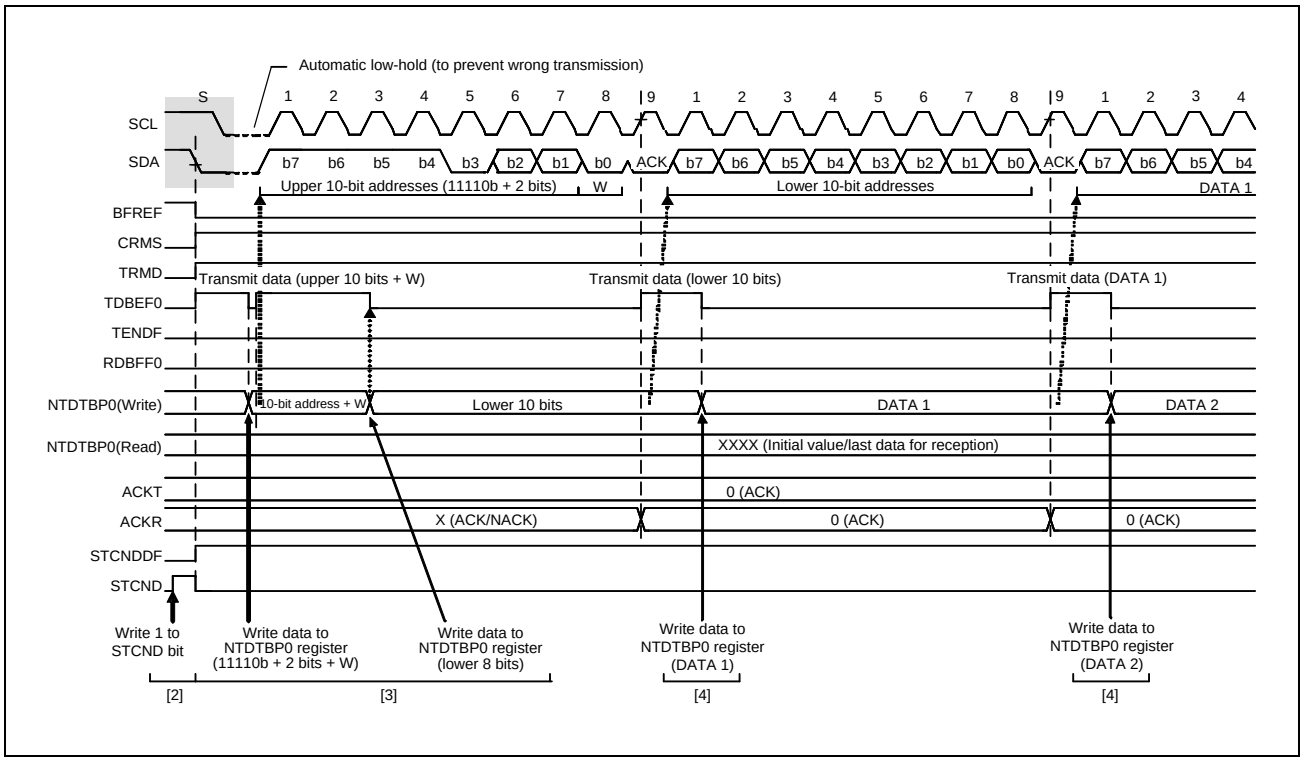


Figure 7.8-7 Master transmit operation timing (2) (10-bit address format)

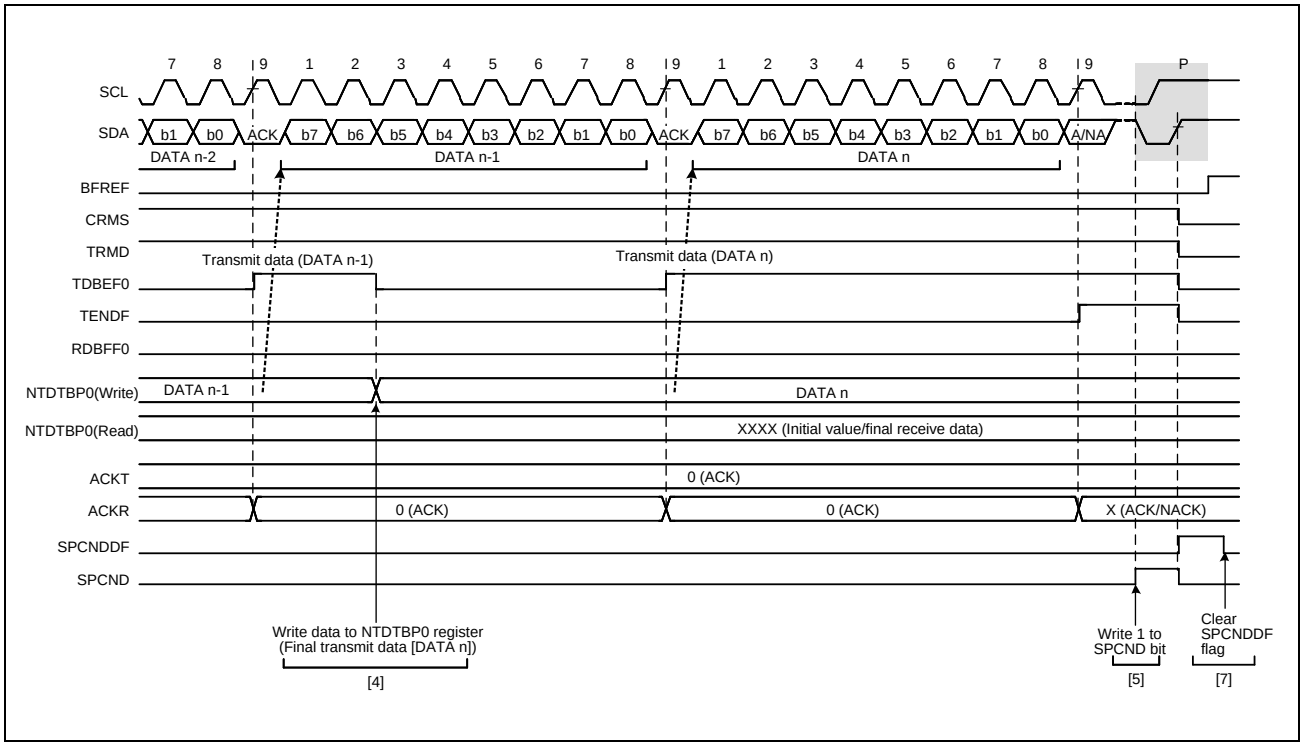


Figure 7.8-8 Master transmit operation timing (3)

(a-2) Data Read Transfer (Single Buffer transfer)

In master receive operation, I3C as a master device outputs the SCL clock, receives data from the slave device, and returns acknowledgments. Because I3C must start by sending a slave address to the corresponding slave device, this part of the procedure is performed in master transmit mode, but the subsequent steps are in master receive mode.

Figure 7.8-115 and **Figure 7.8-116** show examples of usage of master reception (7-bit address format) and **Figure 7.8-9** to **Figure 7.8-11** show the timing of operations in master reception.

The following describes the procedure and operations for master reception.

1. Initial settings. For details, see **7.8.4.3.1 Initial Setting Flow**.
2. Read the BCST.BFREF flag to check that the bus is open, and then set the CNDCTL.STCND bit to 1b (START condition issuance request). Upon receiving the request, I3C issues a START condition. When I3C detects the START condition, the BFREF flag is automatically set to 0b and the BST.STCNDDF flag is automatically set to 1b and the STCND bit is automatically set to 0b. At this time, if the START condition is detected and the levels for the SDA output and the levels on the SDA line have matched while the STCND bit = 1b, I3C recognizes that issuing of the START condition as requested by the STCND bit has been successfully completed, and bits CRMS and TRMD in the PRSST register are automatically set to 1b, placing I3C in master transmit mode. The NTST.TDBEF0 flag is also automatically set to 1b in response to setting of the TRMD bit to 1b.
3. Check that the NTST.TDBEF0 flag = 1b, and then write the value for transmission (the first byte indicates the slave address and value of the R/W# bit) to the NTDTBP0 register. Once the data for transmission are written to the NTDTBP0 register, the TDBEF0 flag is automatically set to 0b, the data are transferred from the Normal Transmit Data Buffer to the Shift Register, and the TDBEF0 flag is again set to 1b. Once the byte containing the slave address and R/W# bit has been transmitted, the value of the PRSST.TRMD bit is automatically updated to select transmit or receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 1, the TRMD bit is set to 0b on the rising edge of the ninth cycle of SCL clock, placing I3C in master receive mode. At this time, the TDBEF0 flag is set to 0b. The NTST.RDBFF0 flag is automatically set to 1b when ACK response is received from the slave device. If the slave device is not recognized or a communication failure occurs, the BST.NACKDF flag will be set to 1b. At this time, set 1b to the CNDCTL.SPCND bit to issue a STOP condition. For master reception from a device with a 10-bit address, start by using master transmission to issue the 10-bit address, and then issue a Repeated START condition. After that, transmitting 1111 0, the two higher-order bits of the slave address, and the R bit places I3C in master receive mode.
4. Dummy read the NTDTBP0 register after confirming that the NTST.RDBFF0 flag = 1b; this makes I3C start output of the SCL clock and start data reception.
5. After 1 byte of data has been received, the NTST.RDBFF0 flag is set to 1 on the rising edge of the eighth or ninth cycle of SCL clock (the clock signal) as selected by the SCSTRCTL.ACKTWE bit. Reading the NTDTBP0 register at this time will produce the received data, and the RDBFF0 flag is automatically set to 0b at the same time. Furthermore, the value of the acknowledgment field received during the ninth cycle of SCL clock is returned as the value set in the ACKCTL.ACKT bit. Furthermore, if the next byte to be received is the next to last byte, set the SCSTRCTL.RWE bit to 1b (for wait insertion) before reading the NTDTBP0 register (containing the second byte from last). As well as enabling NACK output even in the case of delays in processing to set the ACKCTL.ACKT bit to 1b (NACK) in step 6, due to other interrupts, etc., this fixes the SCL line to the low level on the falling edge of the ninth clock cycle in reception of the last byte, so the state is such that issuing a STOP condition is possible.
6. When the SCSTRCTL.ACKTWE bit = 0b and the slave device must be notified that it is to end transfer for data reception after transfer of the next (final) byte, set the ACKCTL.ACKT bit to 1b (NACK).

7. After reading the byte before last from the NTDTBP0 register, if the value of the NTST.RDBFF0 flag is confirmed to be 1b, write 1 to the CNDCTL.SPCND bit (STOP condition issuance request) and then read the last byte from the NTDTBP0 register. When 1b is written to the CNDCTL.SPCND bit, I3C is released from the wait state and issues the STOP condition after low-level output in the ninth clock cycle is completed or the SCL line is released from the low- hold state.
8. Upon detecting the STOP condition, I3C automatically sets bits CRMS and TRMD in the PRSST register to 00b and enters slave receive mode. Furthermore, detection of the STOP condition leads to setting of the BST.SPCNDDF flag to 1b.
9. After checking that the BST.SPCNDDF flag = 1b, set the BST.NACKDF and SPCNDDF flags to 0 for the next transfer operation.

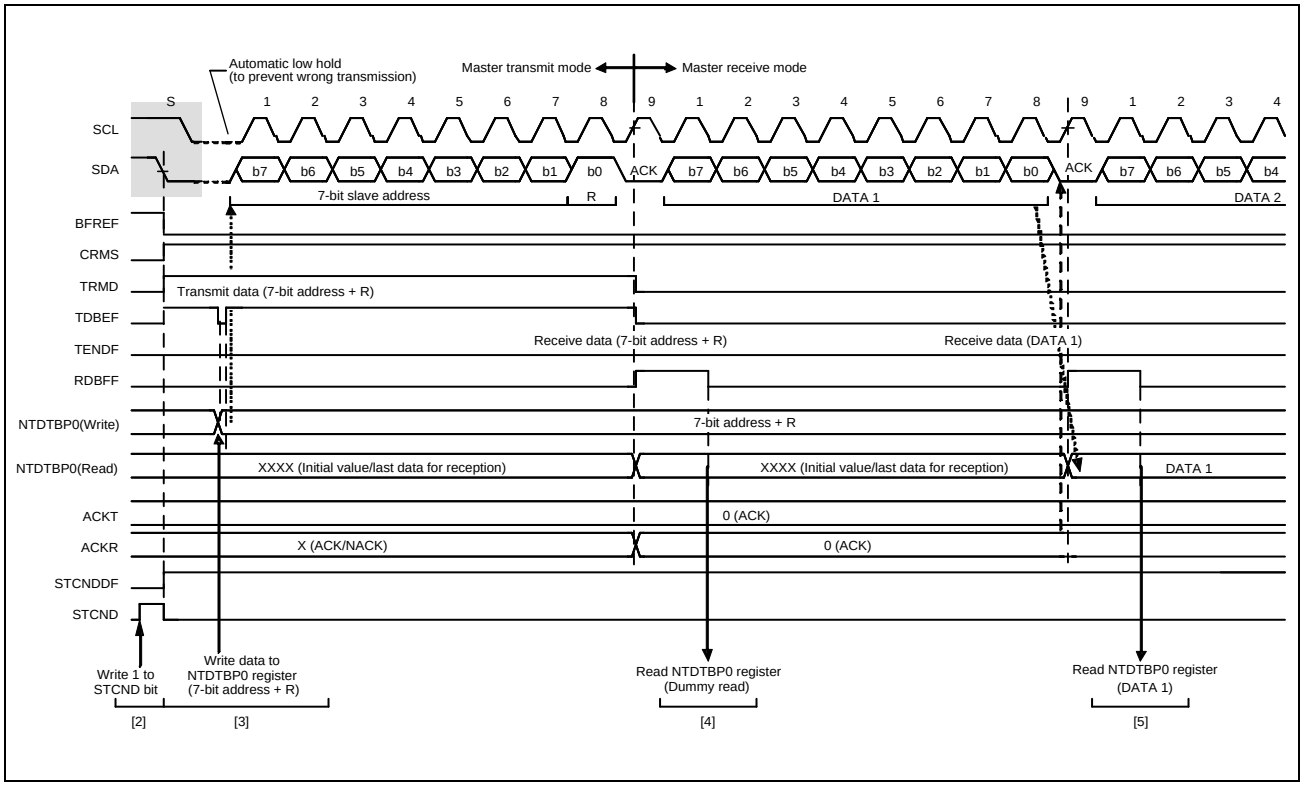


Figure 7.8-9 Master receive operation timing (1) (7-bit address format, when ACKTWE = 0)

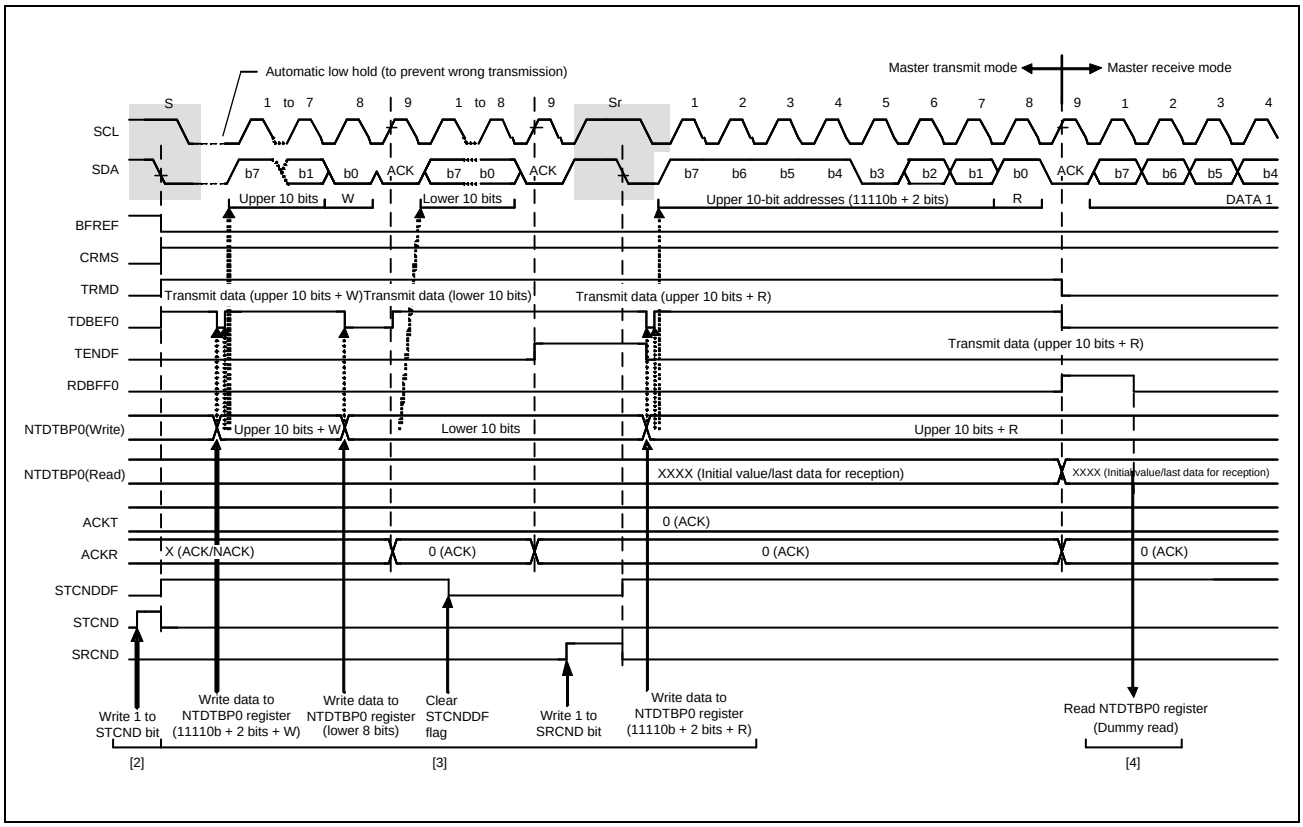


Figure 7.8-10 Master receive operation timing (2) (10-bit address format, when ACKTWE = 0)

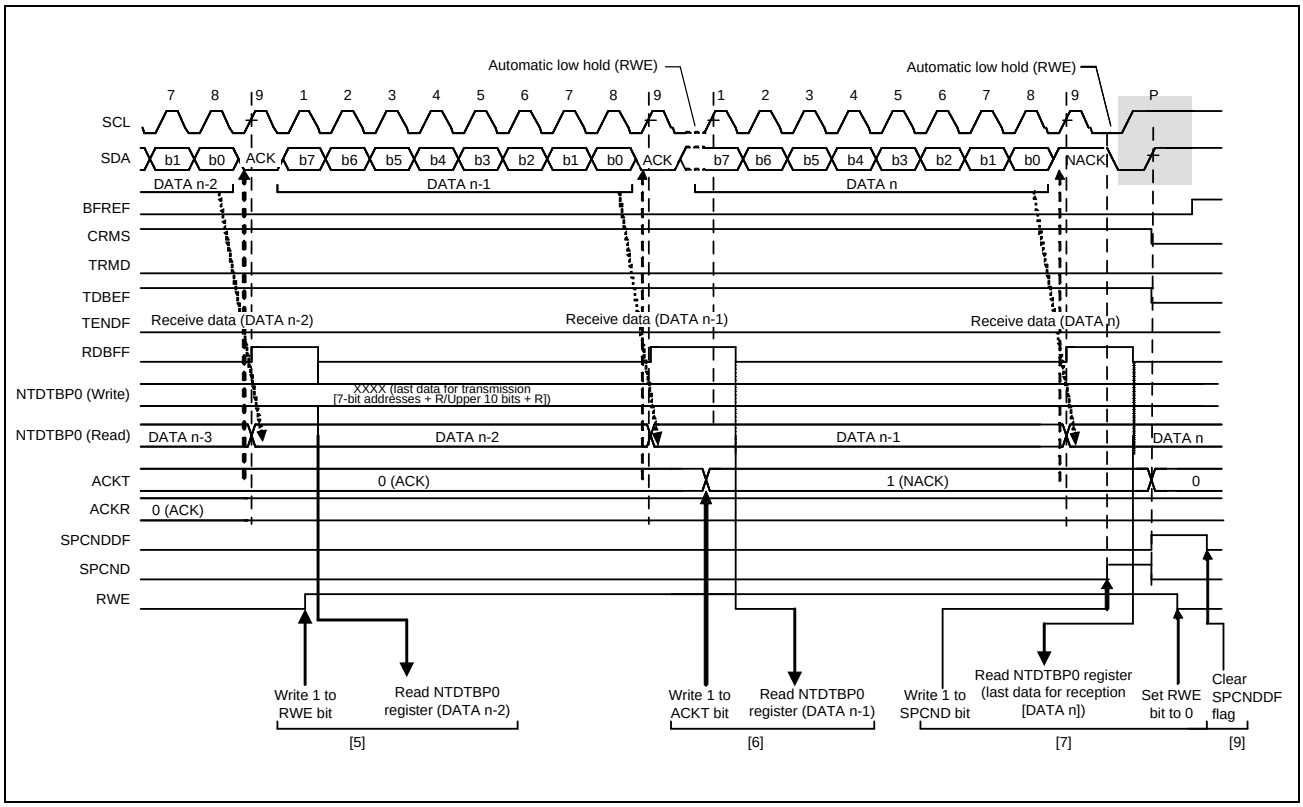


Figure 7.8-11 Master receive operation timing (3) (when ACKTWE = 0)

(b) I3C Master Operation

(b-1) Dynamic Address Assign Procedure

After initializing I3C, first execute Dynamic Address Assign Procedure for I3C Slave connected on the I3C Bus. The following describes the procedure.

1. Initial setting (see **(2) I3C Initial Setting Flow** for details)
2. Execute Dynamic Address Assign with ENTDAA or SETDASA Common Command Code (CCC) for I3C Slave set in DAT (DATBASm register).
3. Write Command Descriptor (Address Assign Command) to Command Buffer via the NCMDQP register.
   When Command Descriptor is written to Command Buffer, Transaction is issued on I3C Bus.
4. When ENTDAA is specified for CMD[7:0] of Address Assign Command:
   Execute Dynamic Address Assign for I3C Slave for the number of DATs specified by DEV_COUNT[3:0] starting with DAT specified by DEV_INDEX[4:0] of Address Assign Command.
   When SETDASA is specified for CMD[7:0] of Address Assign Command:
   Execute Dynamic Address Assign for I3C Slave indicated by DAT specified by DEV_INDEX[4:0] of Address Assign Command.
5. In case of ENTDAA, the Provisional ID, BCR, DCR transmitted from I3C Slave is stored in Receive Data Buffer (BCR is also automatically stored in the MSDCTm register).
   Read the Provisional ID, BCR, and DCR from the Receive Data Buffer via the NTDTBP0 register with an interrupt by RDBFF0 = 1b.
6. When execution of Dynamic Address Assign is completed, issue STOP condition and store the Response Descriptor into the Response Buffer.
7. Read the Response Descriptor via the NRSPQP register and check the status.
8. Check whether the value of the DATA_LENGTH[15:0] bits of the Response Descriptor matches the value of DEV_COUNT[3:0] of the Address Assign Command.

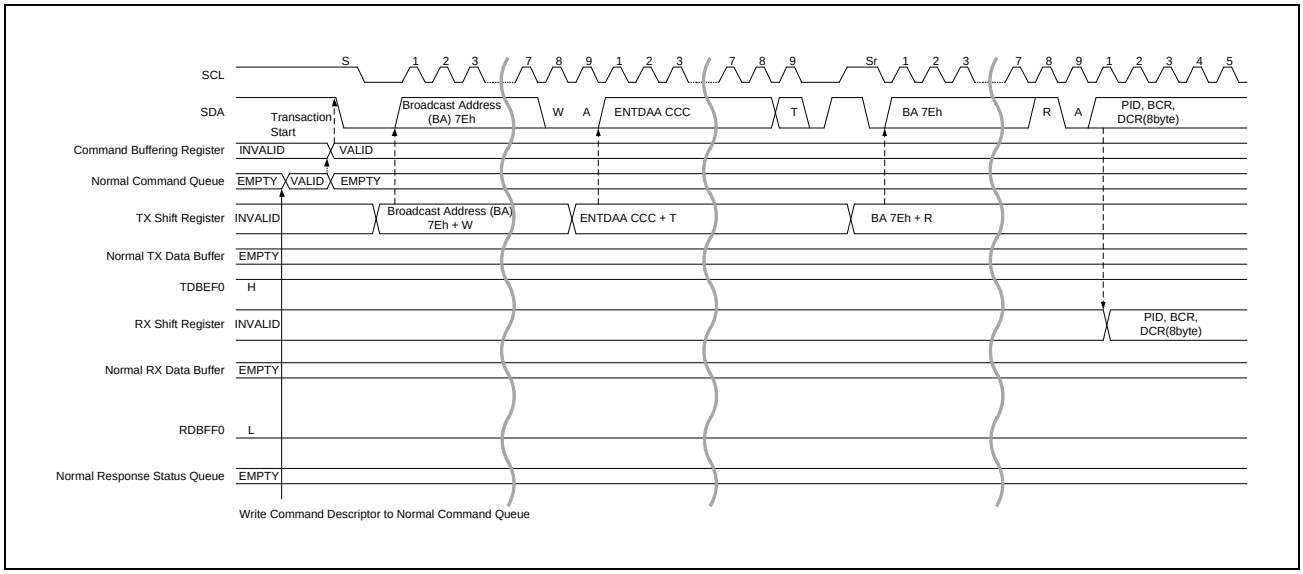


Figure 7.8-12 Dynamic address assign procedure (ENTDAA CCC) timing (1/3)

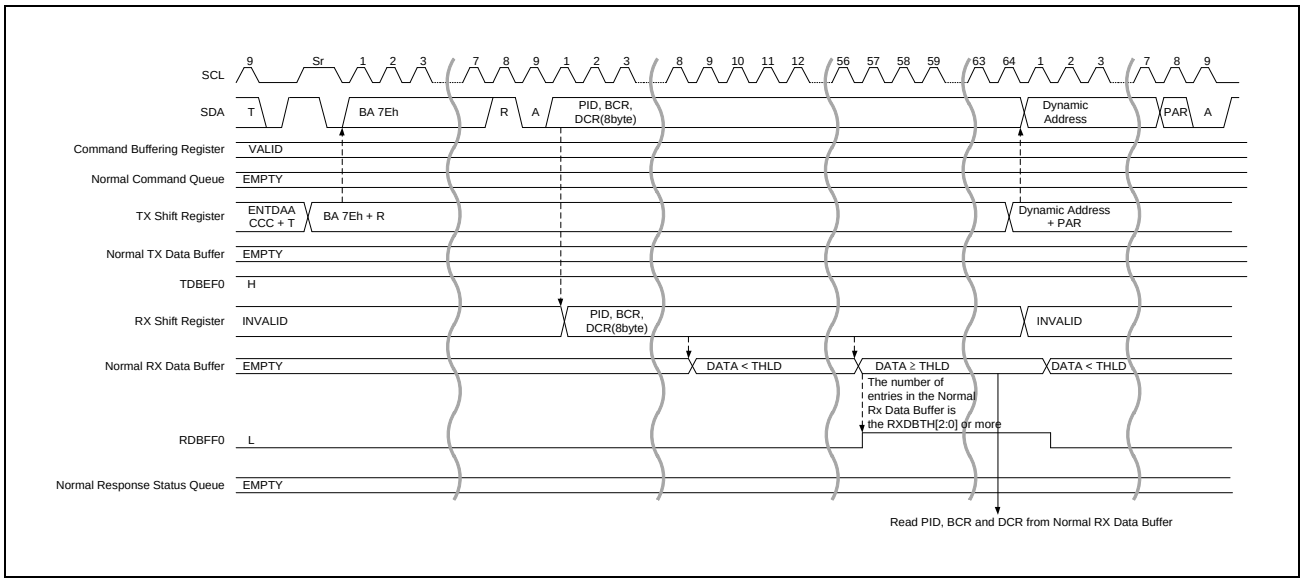


Figure 7.8-13 Dynamic address assign procedure (ENTDAA CCC) timing (2/3)

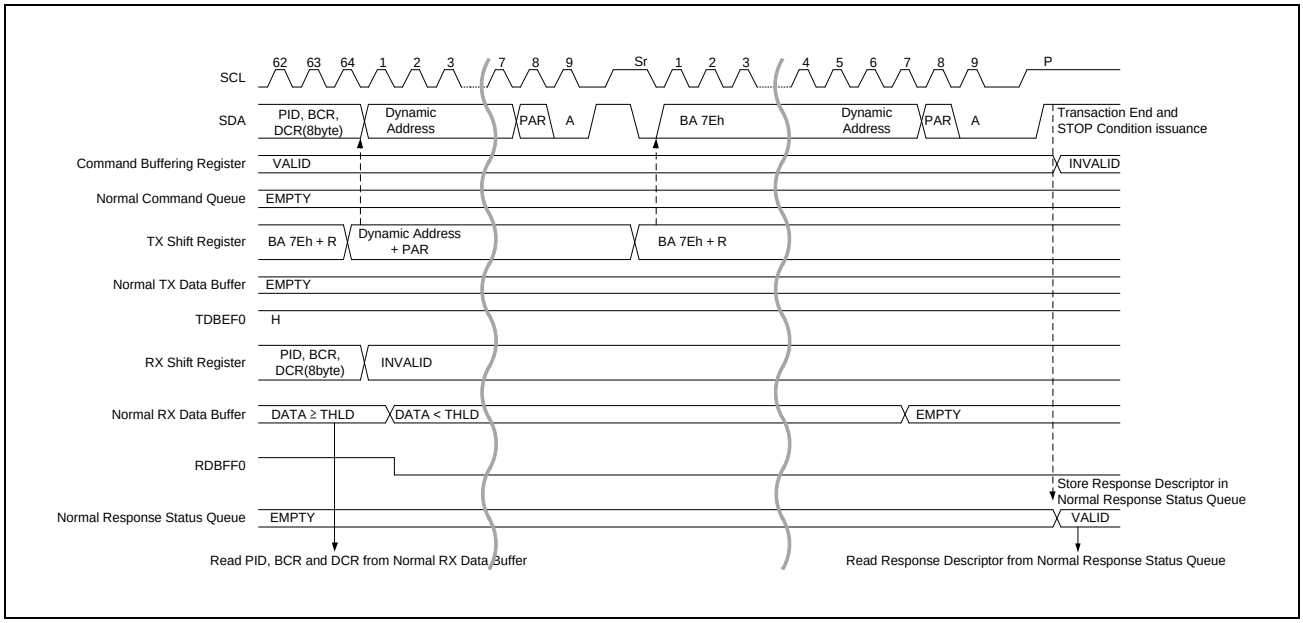


Figure 7.8-14 Dynamic address assign procedure (ENTDAA CCC) timing (3/3)

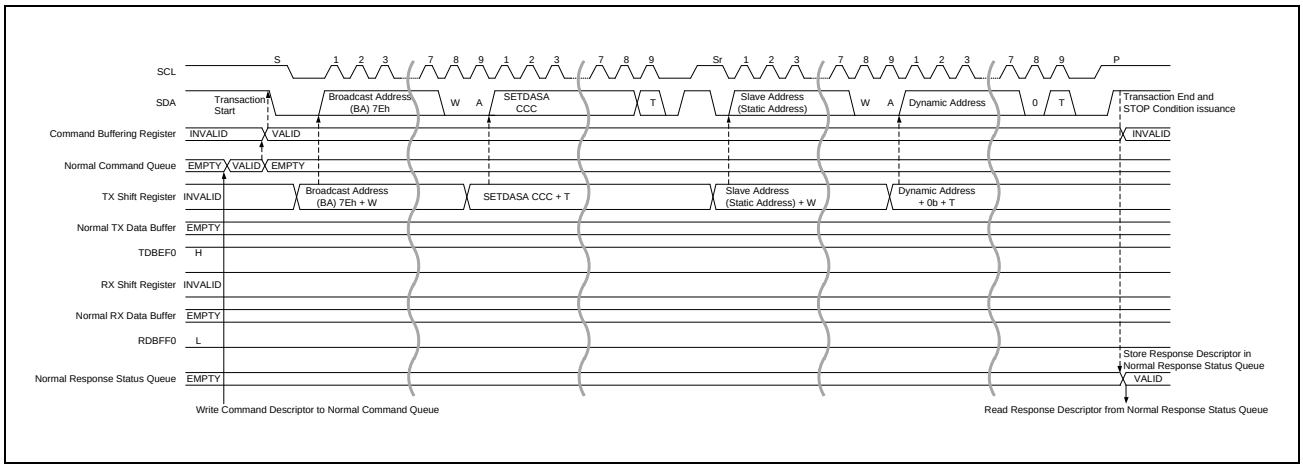


Figure 7.8-15 Dynamic address assign procedure (SETDASA CCC) timing

(b-2) SDR Data Write Transfer

1. Write data for transmission to the Transmit Data Buffer via the NTDTBP0 register.
2. Write a Command Descriptor (Immediate Transfer Command or Regular Transfer Command or Combo Transfer Command) for Data Transfer to the Command Buffer via the NCMDQP register.
3. When Command Descriptor is written to Command Buffer, transaction is issued on I3C Bus.
   When NACK is received with the Address Header, transaction of the same command is automatically issued according to the NACK Retry Count value (DATBASm.DVNACK).
4. If data for transmission still remain, write data for transmission by an interrupt with TDBEF0 = 1b to the Transmit Data Buffer via the NTDTBP0 register.
5. When data transmission for the number of Data Length specified by the DATA_LENGTH[15:0] bits of the Command Descriptor is completed, the Repeated START condition or STOP condition is issued and the Response Descriptor is stored in the Response Buffer.
6. Read the Response Descriptor via the NRSPQP register and check the status.
7. Check that the value of the DATA_LENGTH[15:0] bits of the Response Descriptor is 0.

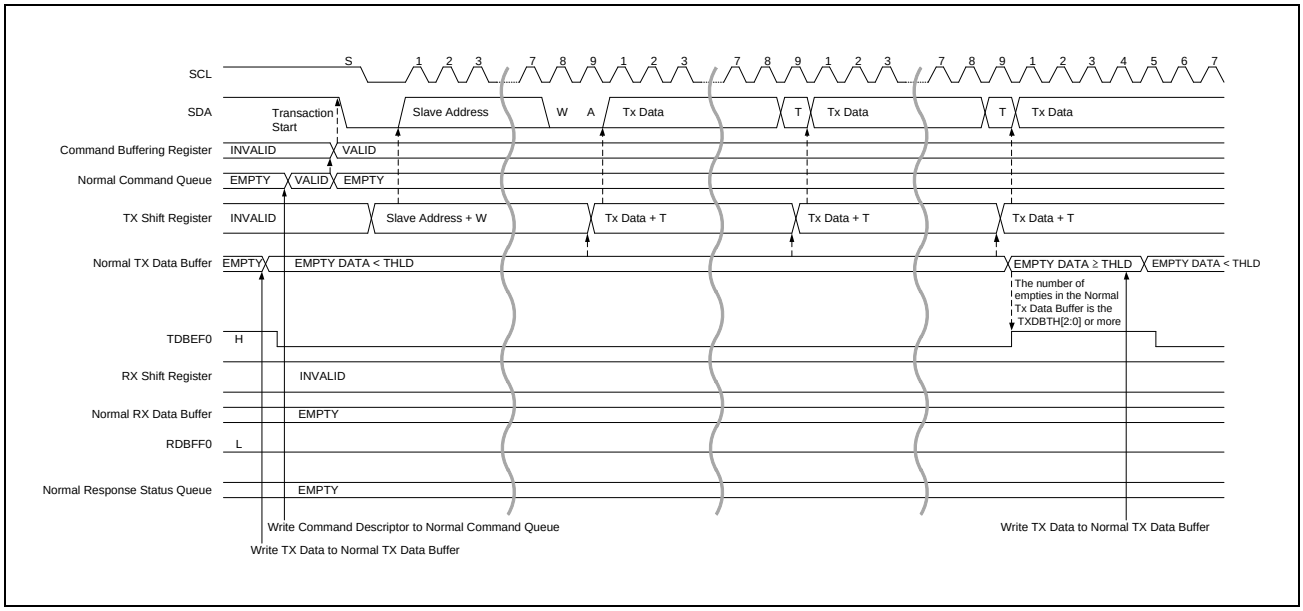


Figure 7.8-16 SDR data write transfer timing (1/2)

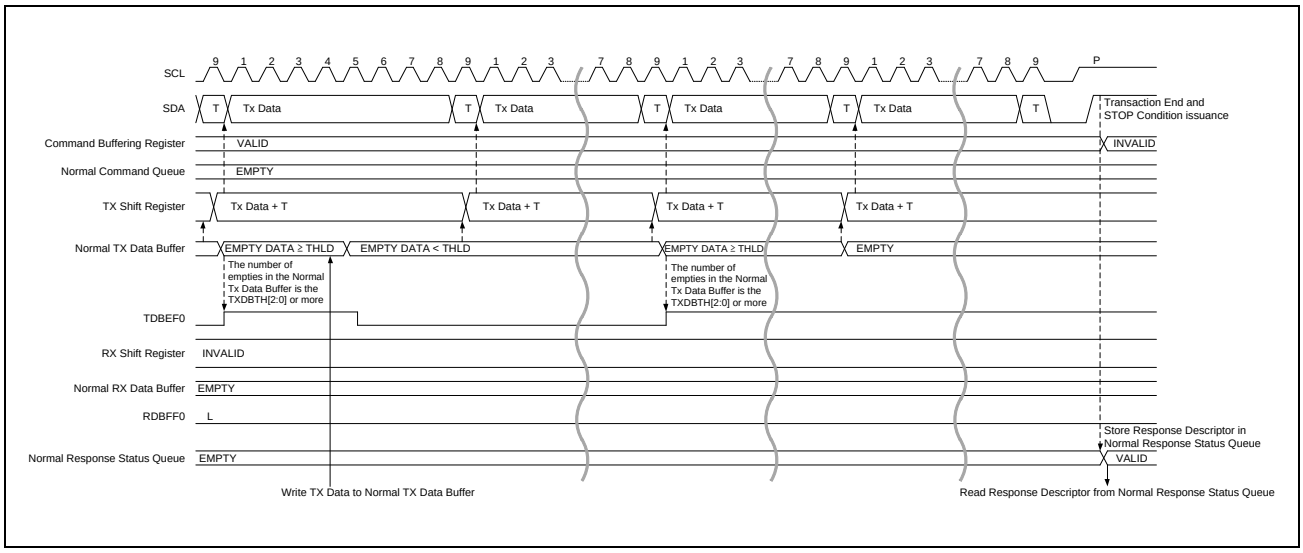


Figure 7.8-17 SDR data write transfer timing (2/2)

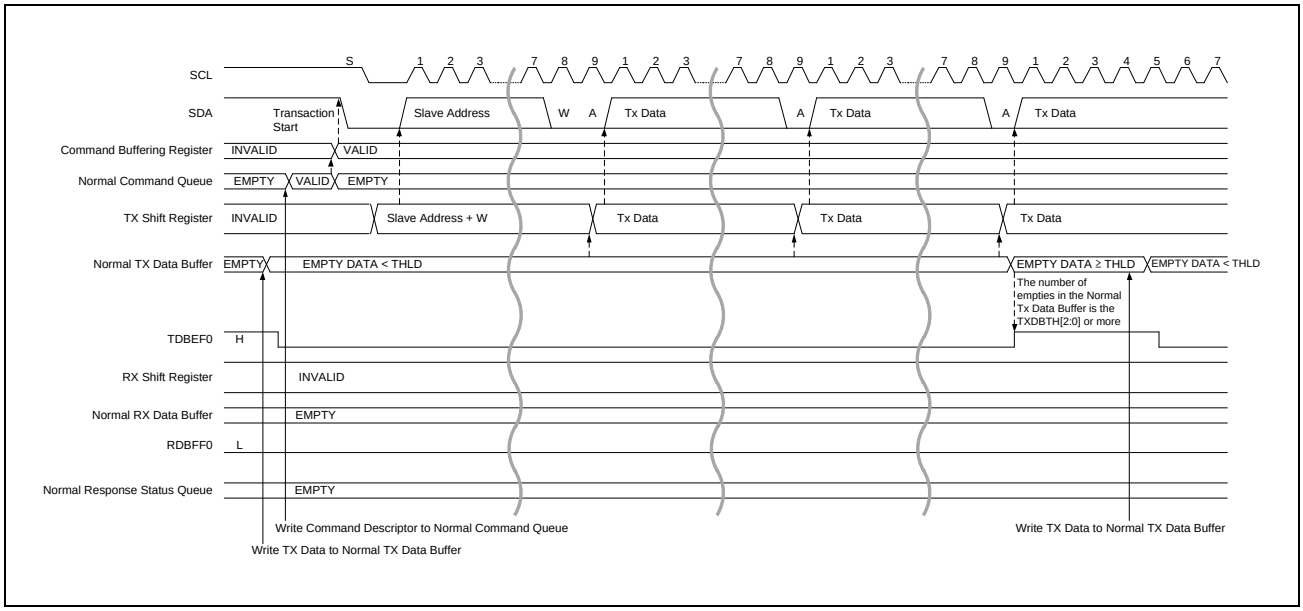


Figure 7.8-18 Legacy I^{2}C message data write timing (1/2)

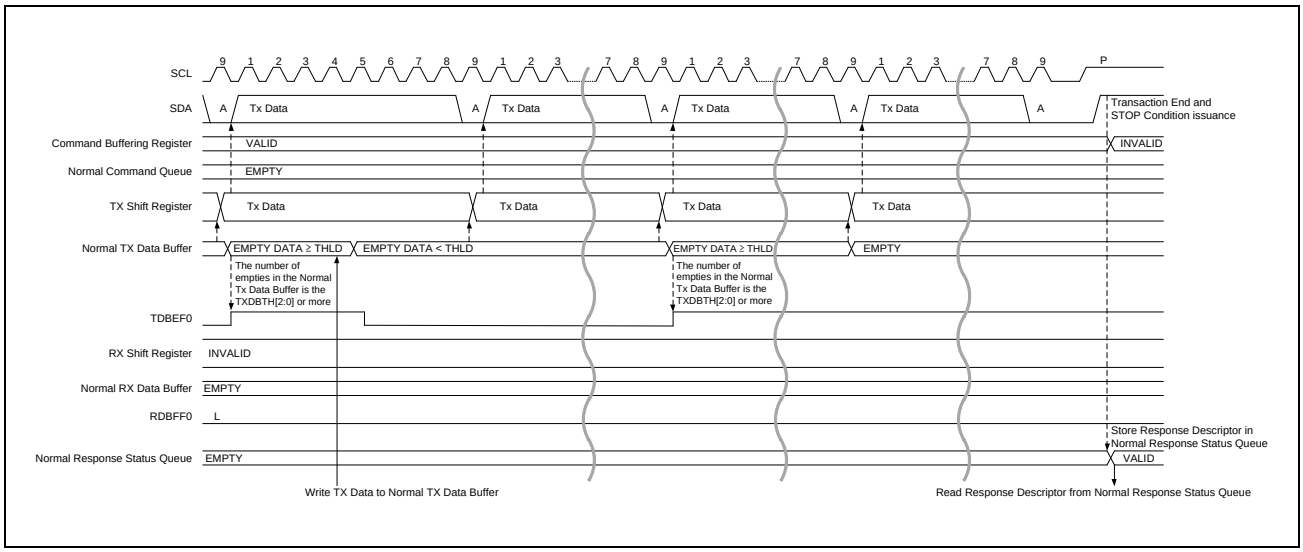


Figure 7.8-19 Legacy I²C message data write timing (2/2)

(b-3) SDR Data Read Transfer

1. Write a Command Descriptor (Immediate Transfer Command or Regular Transfer Command or Combo Transfer Command) for Data Transfer to the Command Buffer via the NCMDQP register.
2. When Command Descriptor is written to Command Buffer, transaction is issued on I3C Bus.
   When NACK is received with the Address Header, transaction of the same command is automatically issued according to the NACK Retry Count value (DATBASm.DVNACK).
3. Data received from the I3C Slave is stored in the Receive Data Buffer.
4. With the RDBFF0 = 1b interrupt, the received data is read from the Receive Data Buffer via the NTDTBP0 register.
5. SDR:
   Detecting Low in T-bit or receiving Data for the number of Data Length specified by the DATA_LENGTH[15:0] bits of Command Descriptor is completed, issue Repeated START condition or STOP condition and store the Response Descriptor into the Response Buffer.
   Legacy I^2C Message:
   When data reception for the number of Data Length specified by the DATA_LENGTH[15:0] bits of Command Descriptor is completed, NACK is issued. After that, issue a Repeated START condition or STOP condition and store the Response Descriptor into the Response Buffer.
6. Read the Response Descriptor via the NRSPQP register and check the status.
7. Check whether the value of the DATA_LENGTH[15:0] bits of the Response Descriptor matches the data length setting value of the Command Descriptor.

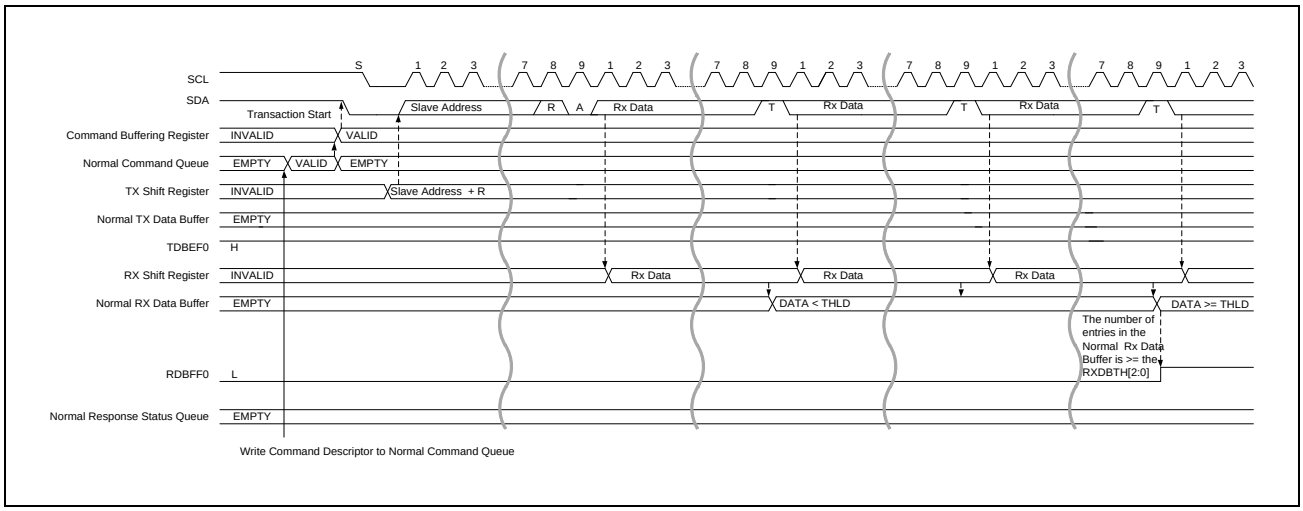


Figure 7.8-20 SDR data read transfer timing (1/2)

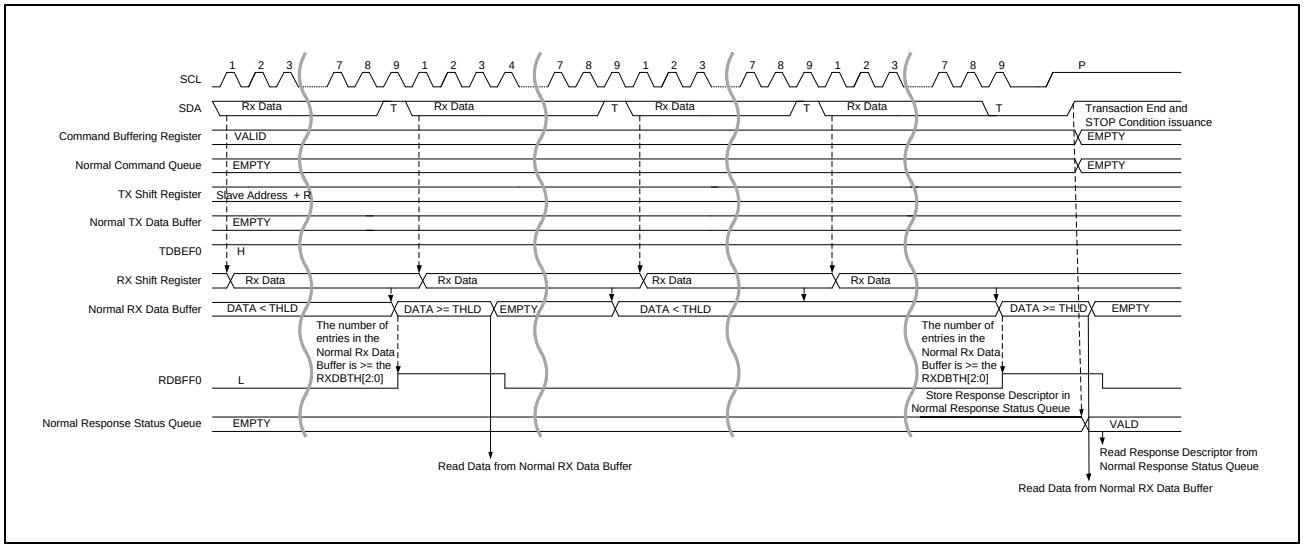


Figure 7.8-21 SDR data read transfer timing (2/2)

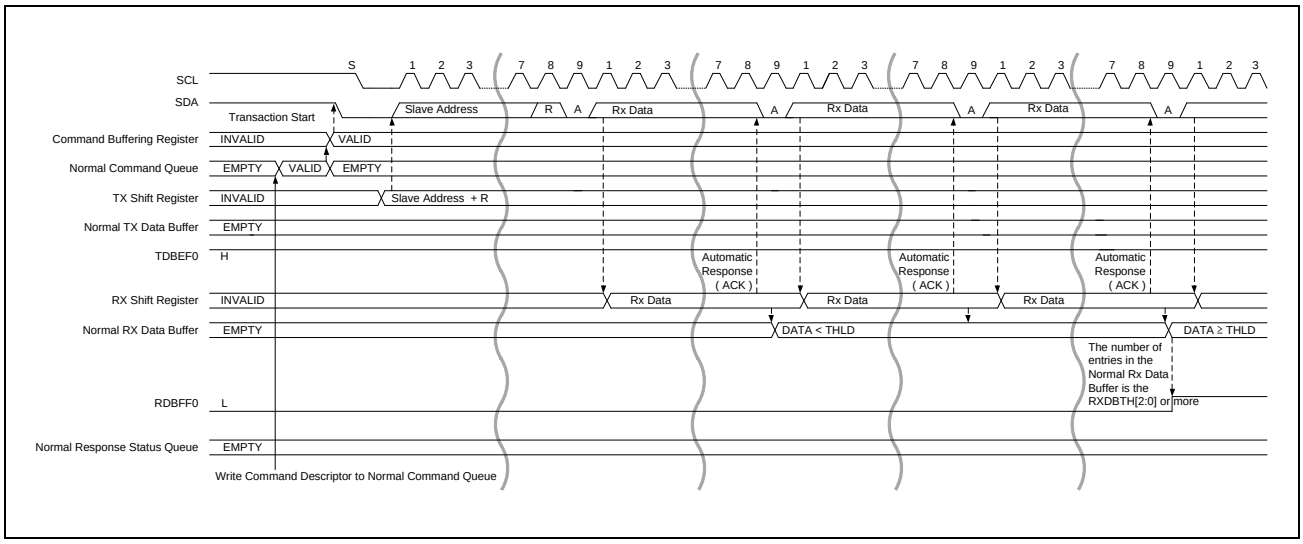


Figure 7.8-22 Legacy I²C message data read transfer timing (1/2)

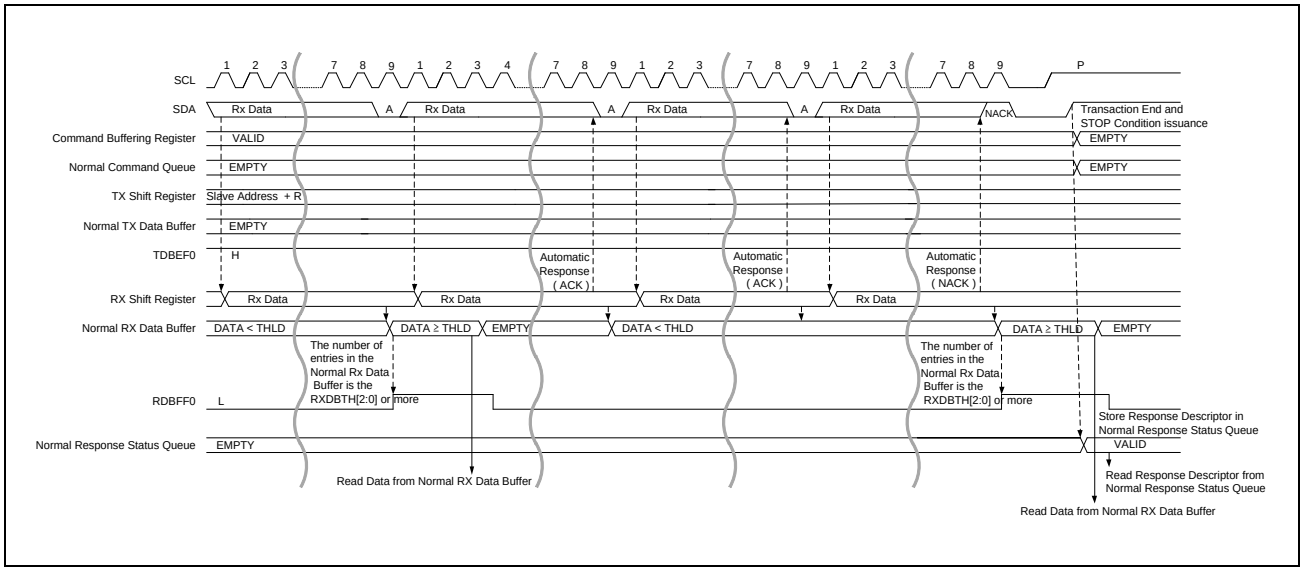


Figure 7.8-23 Legacy I²C message data read transfer timing (2/2)

(b-4) IBI Transfer

1. Write Command Descriptor to the Command Buffer and issue Transaction on I3C Bus.
   If START Request (SDA Low Drive) is issued from the slave device, I3C drives SCL to Low and completes START condition.
   Thereafter, the SCL is supplied and In-Band Interrupt Request is received.
2. In Slave Address with RnW of the Address Header, if losing Arbitration by issuing In-Band Interrupt from I3C Slave, stop issuing Transaction.
3. According to **(8) In-Band Interrupt [I3C mode]**, detect In-Band Interrupt and process.
4. In the interrupt with IBIQEFF = 1b, read the IBI Status Descriptor from the IBI Status Buffer via the NIBIQP register and check the status.
   When detected a Slave Interrupt Request and responded with ACK, Read the IBI Data for the Data Length indicated by the DATA_LENGTH[15:0] bits of the IBI Status Descriptor from the IBI Data Buffer via the NIBIQP register.
5. Restart issuing Transaction of Command of Step1.

An example of the processing procedure after detection of In-Band Interrupt is shown below.

Processing procedure for detecting Mastership Request and transferring master right to Secondary Master

1. If the I3C Secondary Master wins the Arbitration, issue a DEFSLVS CCC and notify Slave information to Secondary Master.
2. Issue a GETACCMST CCC and complete CCC by a STOP condition.

Remark After transferring master right to Secondary Master, to get master right again, issue a Mastership Request according to (f) IBI Transfer of **(b) I3C Slave Operation**.
After Mastership Request is accepted by the Current Master, to get master right again at receiving the GETACCMST CCC and complete CCC by a STOP condition.

Processing procedure when Hot-Join Event is detected

1. Issue a Broadcast Command Code Enter Dynamic Address Assignment (ENTDAA) to start the Dynamic Address Assignment process.
2. Issue a DEFSLVS CCC and notify Slave information to Secondary Master.

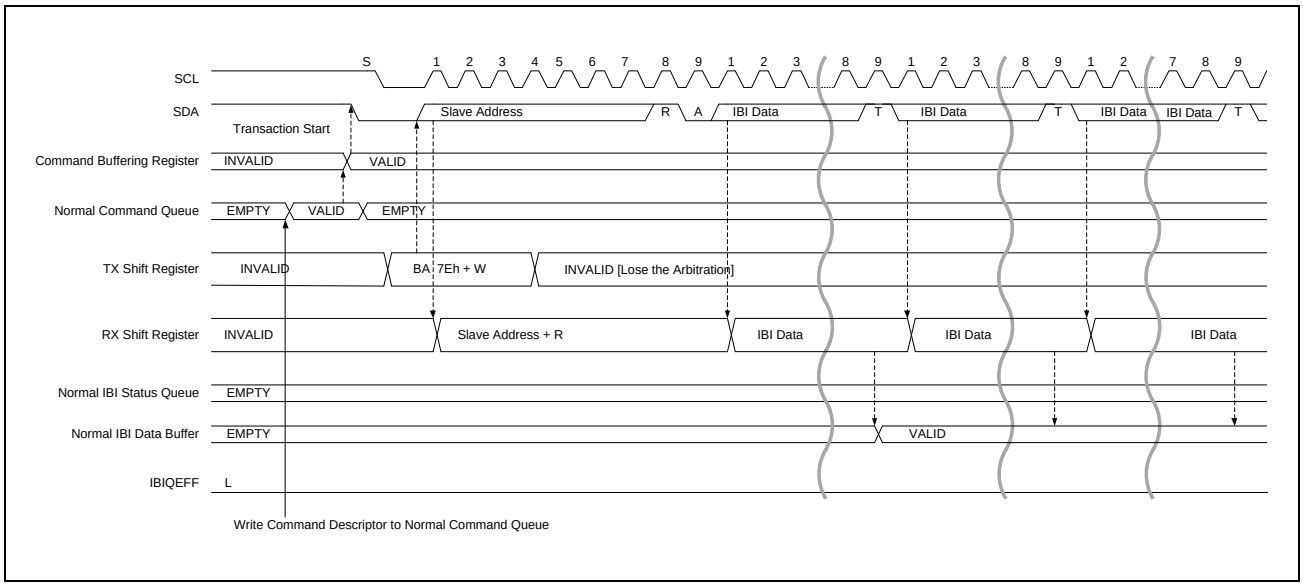


Figure 7.8-24 I3C master IBI transfer timing (1/2)

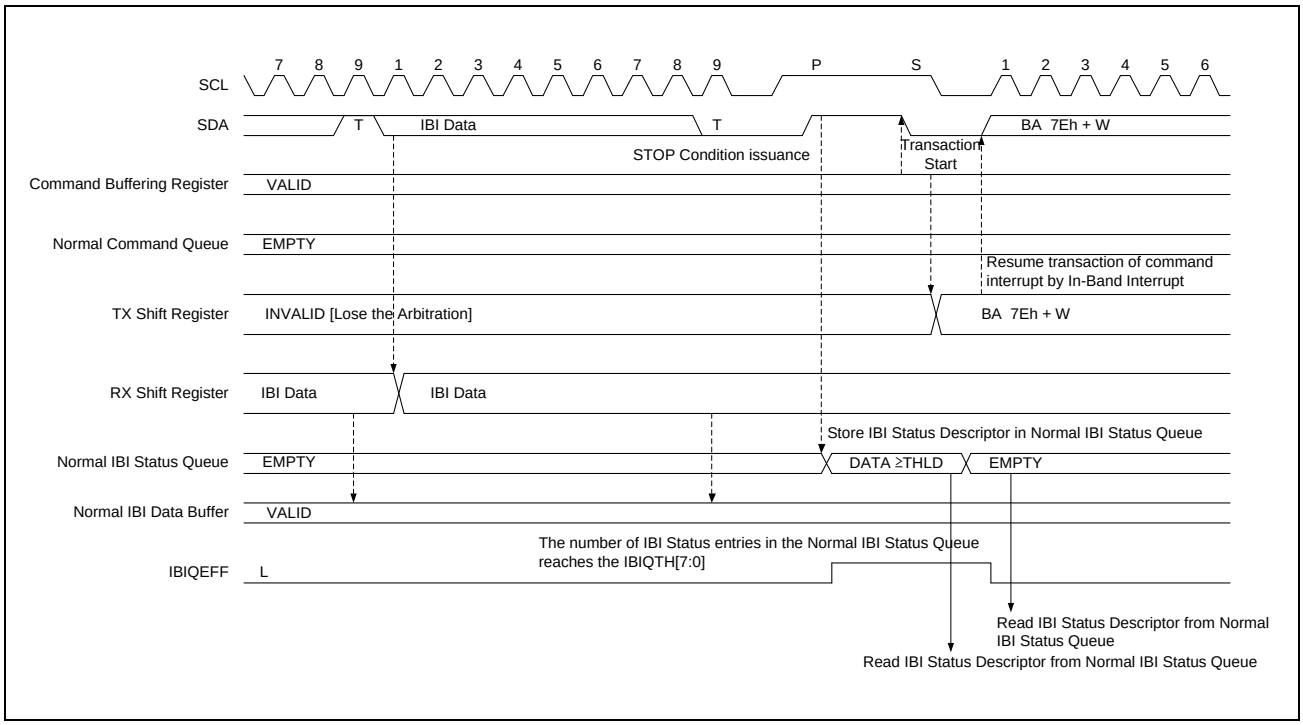
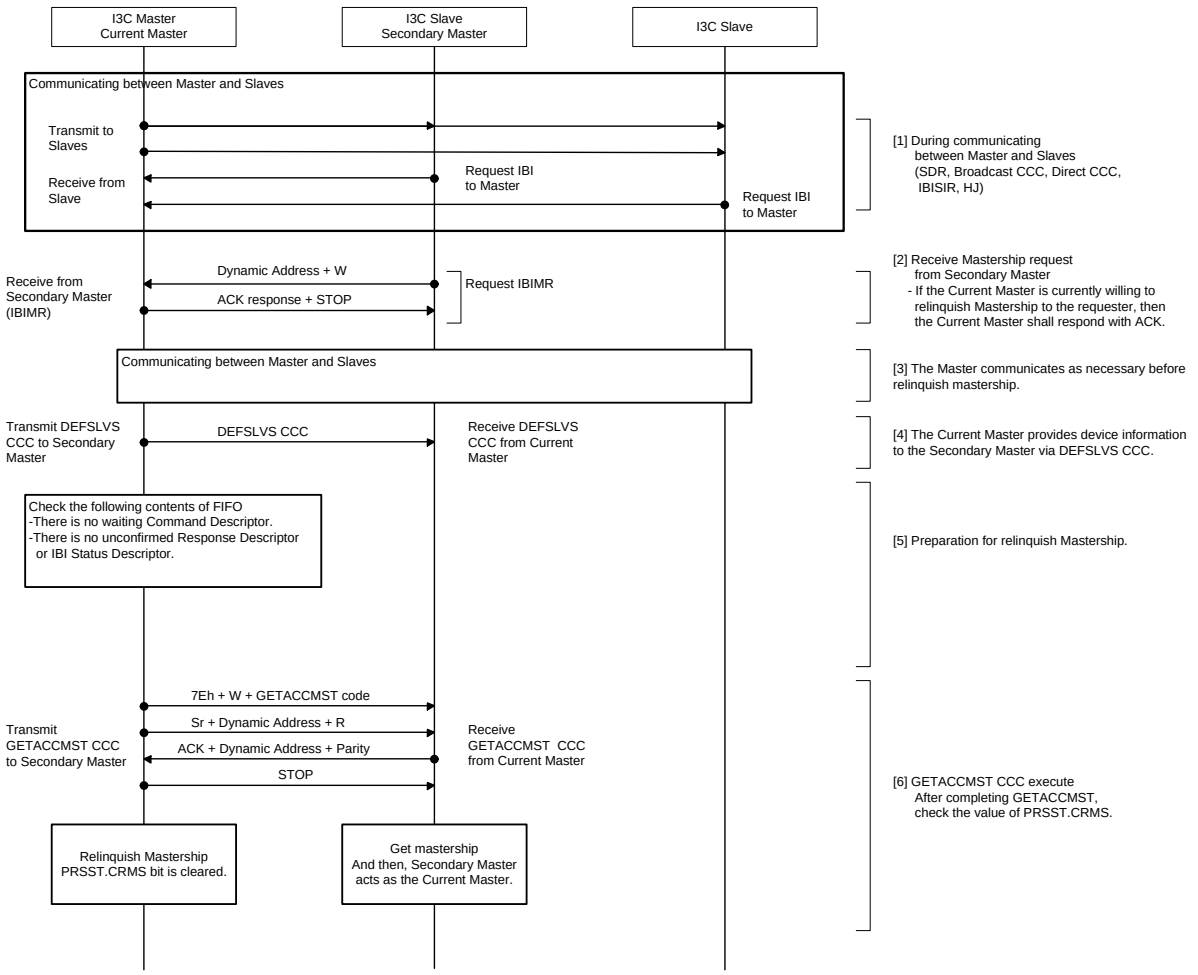


Figure 7.8-25 I3C master IBI transfer timing (2/2)



Note: The Abort request during GETACCMST by BCTL.ABT setting is prohibited due to the transfer of Mastership.

Figure 7.8-26 I3C Master Mastership processing flow

(2) Slave Mode Operation

(a) I^2C Slave Operation

(a-1) Data Write Transfer (Single Buffer transfer)

In slave receive operation, the master device outputs the SCL clock and transmit data, and I3C returns acknowledgments as a slave device.

Figure 7.8-122 shows an example of usage of slave reception and **Figure 7.8-27** and **Figure 7.8-28** show the timing of operations in slave reception.

The following describes the procedure and operations for slave reception.

1. Initial settings. For details, see **7.8.4.3.1 Initial Setting Flow**. After initial settings, I3C will stay in the standby state until it receives a slave address that it matches.
2. After receiving a matching slave address, I3C sets one of the corresponding bits SVST.HOAF, GCAF, and SVAFn (n = 0 to 2) to 1b on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the acknowledge bit (ACK) on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 0b, I3C continues to place itself in slave receive mode and sets the NTST.RDBFF0 flag to 1b.
3. After the BST.SPCNDDF flag is confirmed to be 0b and the NTST.RDBFF0 flag to be 1b, dummy read the NTDTBP0 register (the dummy value consists of the slave address and R/W# bit when the 7-bit address format is selected, or the lower 8 bits when the 10-bit address format is selected).
4. When the NTDTBP0 register is read, I3C automatically sets the NTST.RDBFF0 flag to 0b. If reading of the NTDTBP0 register is delayed and a next byte is received while the RDBFF0 flag is still set to 1b, I3C holds the SCL line low from one SCL cycle before the timing with which RDBFF0 should be set. In this case, reading the NTDTBP0 register releases the SCL line from being held at the low level. When the BST.SPCNDDF flag = 1b and the NTST.RDBFF0 flag is also 1b, read the NTDTBP0 register until all the data is completely received.
5. Upon detecting the STOP condition, I3C automatically clears bits SVST.HOAF, GCAF, and SVAFn (n = 0 to 2) to 0b.
6. After checking that the BST.SPCNDDF flag = 1b, set the BST.SPCNDDF flag to 0b for the next transfer operation.

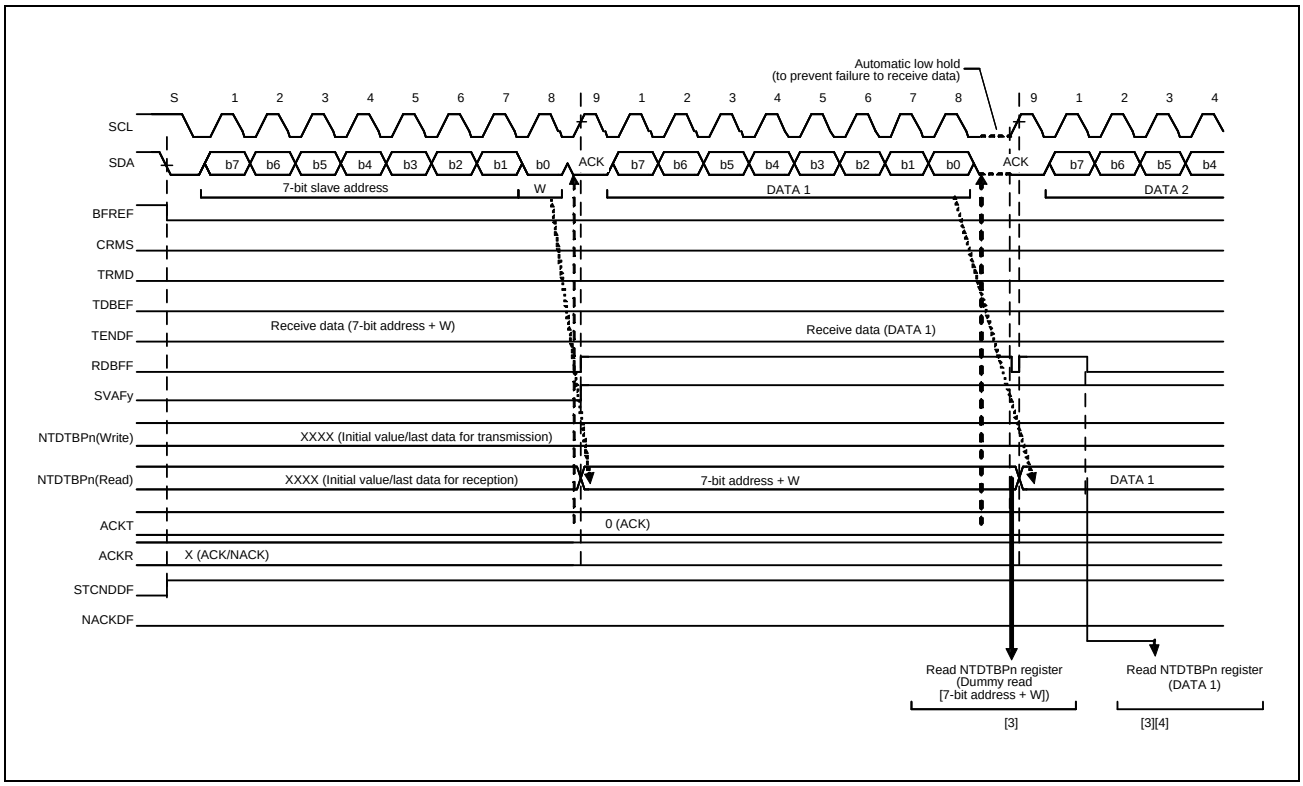


Figure 7.8-27 Slave receive operation timing (1) (7-bit address format, when ACKTWE = 0)

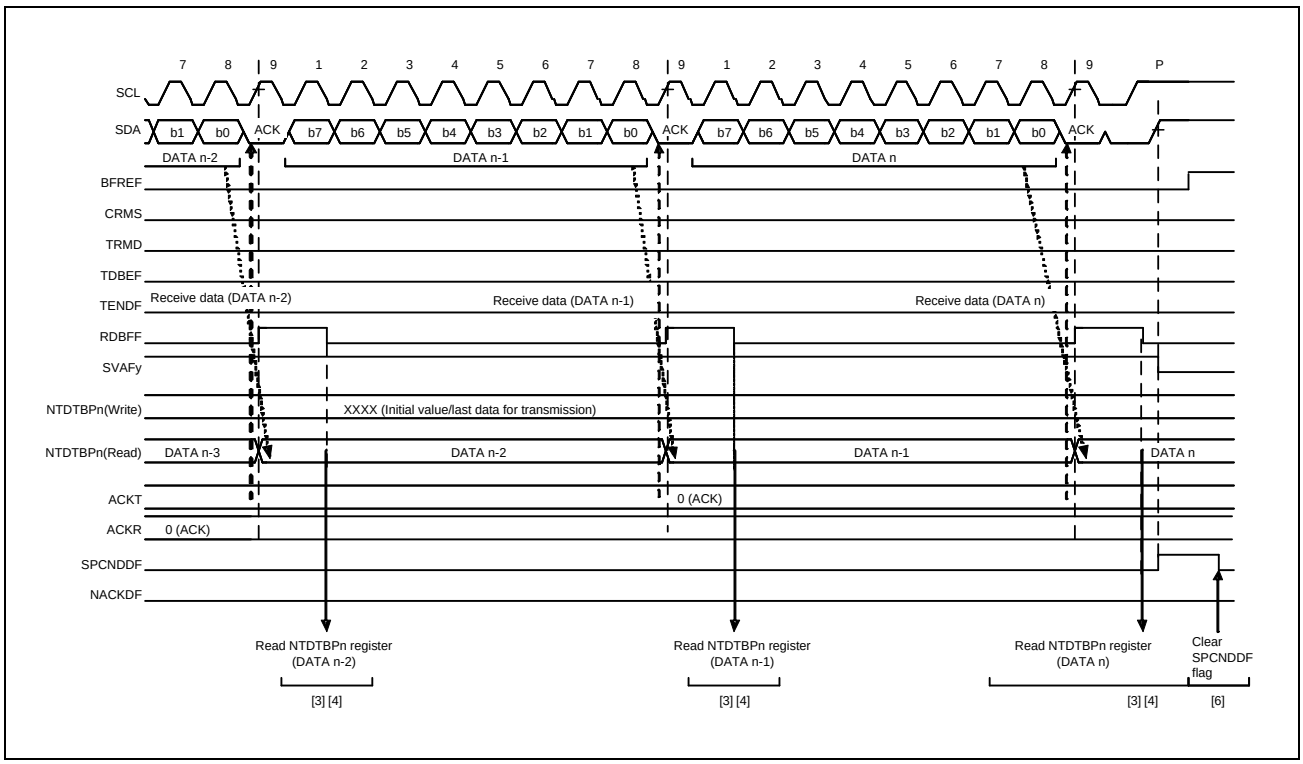


Figure 7.8-28 Slave receive operation timing (2) (when ACKTWE = 0)

(a-2) Data Read Transfer (Single Buffer transfer)

In slave transmit operation, the master device outputs the SCL clock, I3C transmits data as a slave device, and the master device returns acknowledgments.

Figure 7.8-121 shows an example of usage of slave transmission and **Figure 7.8-29** and **Figure 7.8-30** show the timing of operations in slave transmission.

The following describes the procedure and operations for slave transmission.

1. Initial settings. For details, see **7.8.4.3.1 Initial Setting Flow**.
   After initial settings, I3C will stay in the standby state until it receives a slave address that it matches.
2. After receiving a matching slave address, I3C sets one of the corresponding bits SVST.HOAF, GCAF, and SVAFn (n = 0 to 2) to 1b on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the acknowledge bit (ACK) on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 1b, I3C automatically places itself in slave transmit mode by setting both the PRSST.TRMD bit and the NTST.TDBEF0 flag to 1b.
3. After the NTST.TDBEF0 flag is confirmed to be 1b, write the data for transmission to the NTDTBP0 register. At this time, if I3C does not receive acknowledge from the master device (receives a NACK signal) while the BSTE.NACKDE bit = 1b, I3C aborts transfer of the next data.
4. Wait until the following (a) or (b) condition.
   (a) The BST.NACKDF flag is set to 1b.
   (b) The BST.TENDF flag is set to 1b while the NTST.TDBEF0 flag = 1b, after the last byte for transmission is written to the NTDTBP0 register.
5. When the BST.NACKDF flag or the BST.TENDF flag = 1b, dummy read the NTDTBP0 register to complete the processing.
6. Upon detecting the STOP condition, I3C automatically sets bits SVST.HOAF, GCAF, and SVAFn (n = 0 to 2), flags NTST.TDBEF0 and BST.TENDF, and the PRSST.TRMD bit to 0b, and enters slave receive mode.
7. After checking that the BST.SPCNDDF flag = 1b, set the BST.NACKDF and SPCNDDF flags to 0b for the next transfer operation.

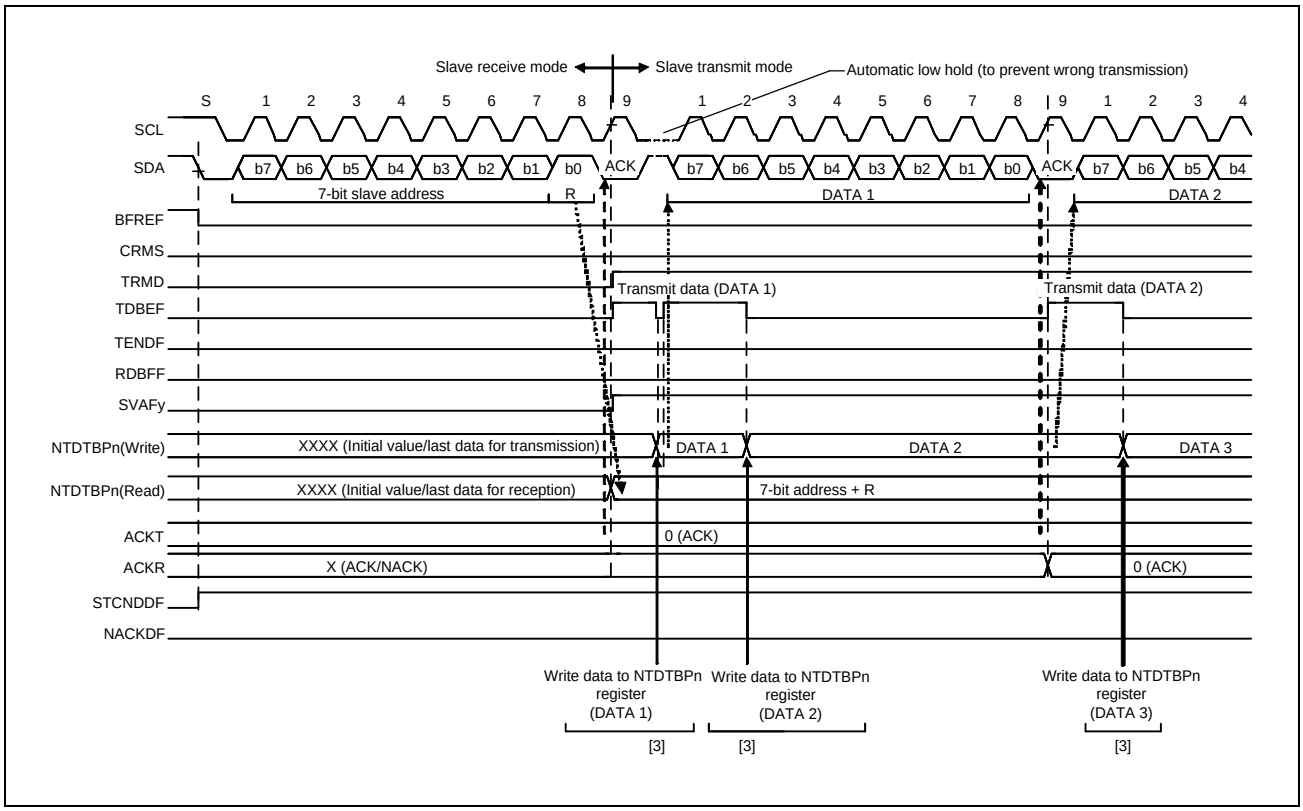


Figure 7.8-29 Slave transmit operation timing (1) (7-bit address format)

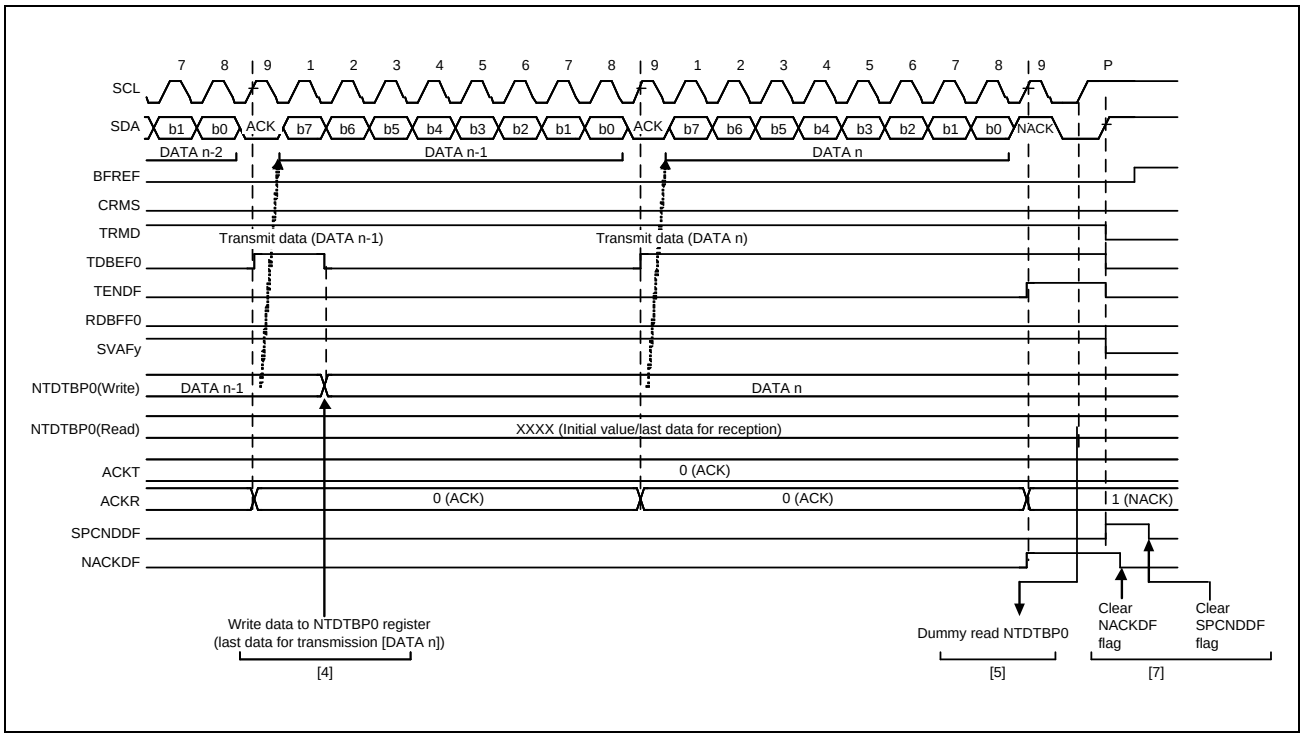


Figure 7.8-30 Slave transmit operation timing (2)

(b) I3C Slave Operation

(b-1) Dynamic Address Assign Procedure

After initializing I3C, the I3C master first performs Dynamic Address Assign Procedure.

The operation of R-I3 during the Dynamic Address Assign Procedure by ENTDAA CCC is described below.

1. Initial setting (For details, see **(2) I3C Initial Setting Flow**)
2. When ENTDAA CCC is received, I3C transmits Provisional ID (SDCTPIDH[31:0], SDCTPIDL[15:0]), BCR (SVDCT.TBCR[7:0]), DCR (SVDCT.TDCR[7:0]) until a dynamic address is assigned. (For details, see “In case of Broadcast CCC (ENTDAA)” of **(f) CCC detection function [I3C mode]**.)
3. When ENTDAA CCC is completed and a STOP condition is detected, Receive Status Descriptor is stored in Receive Status Buffer.
4. Read Receive Status Descriptor via NRSQP register and check the status.
5. Read the data for the Data Length indicated by the DATA_LENGTH[15:0] bits of the Receive Status Descriptor from the Receive Data Buffer via the NTDTBP0 register.

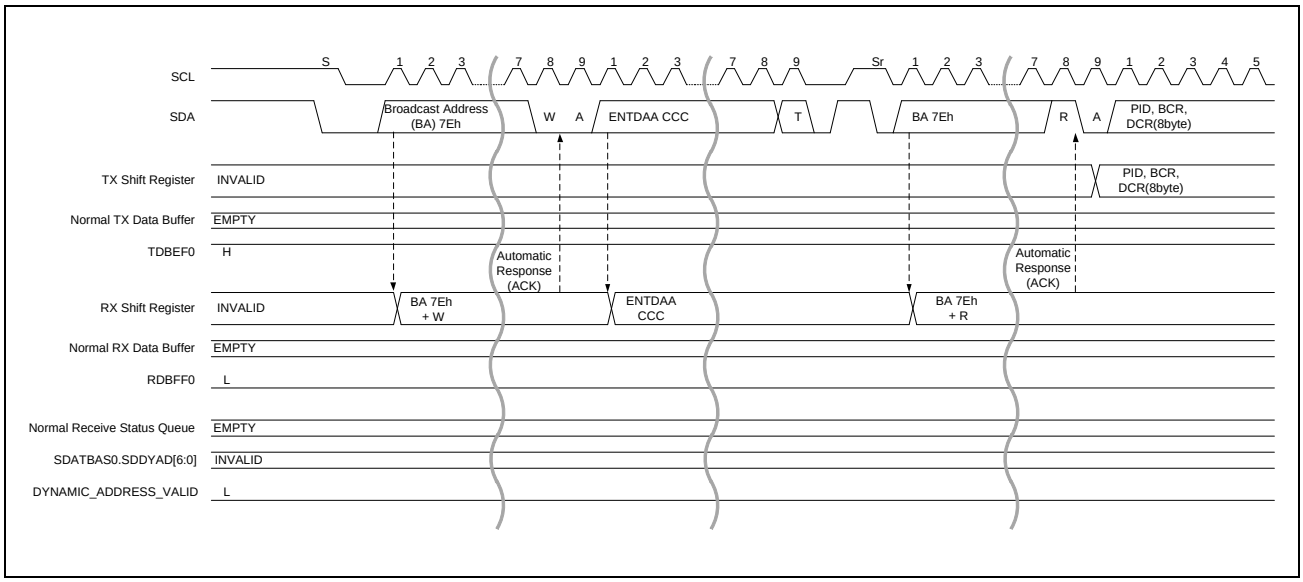


Figure 7.8-31 Dynamic address assign procedure (ENTDAA CCC) timing (1/3)

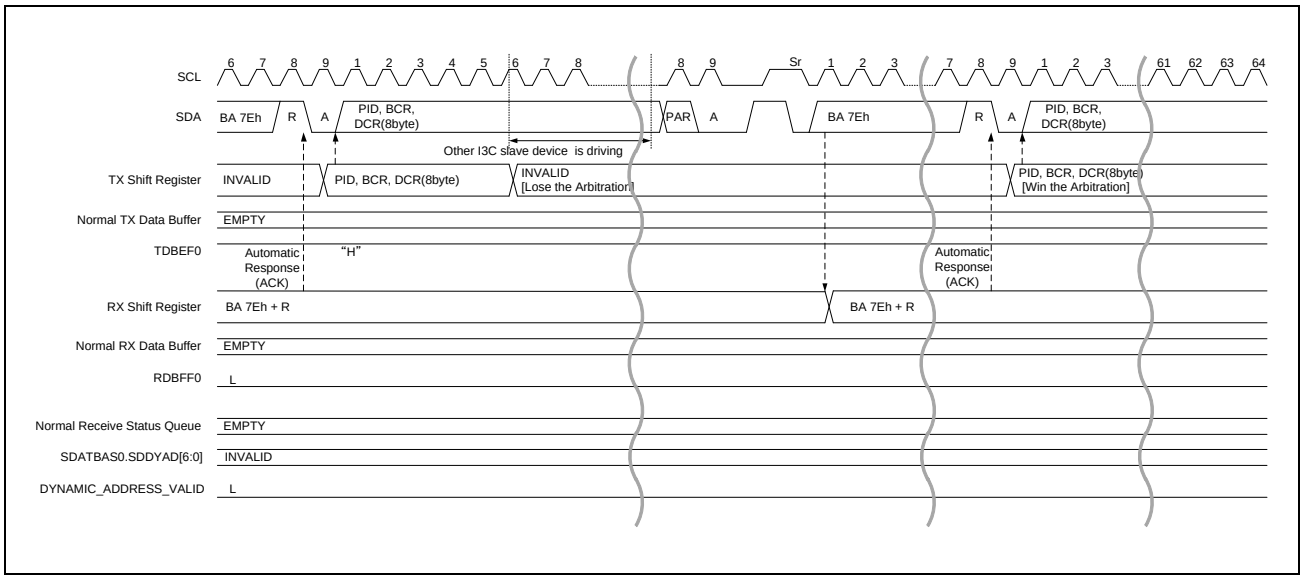


Figure 7.8-32 Dynamic address assign procedure (ENTDAA CCC) timing (2/3)

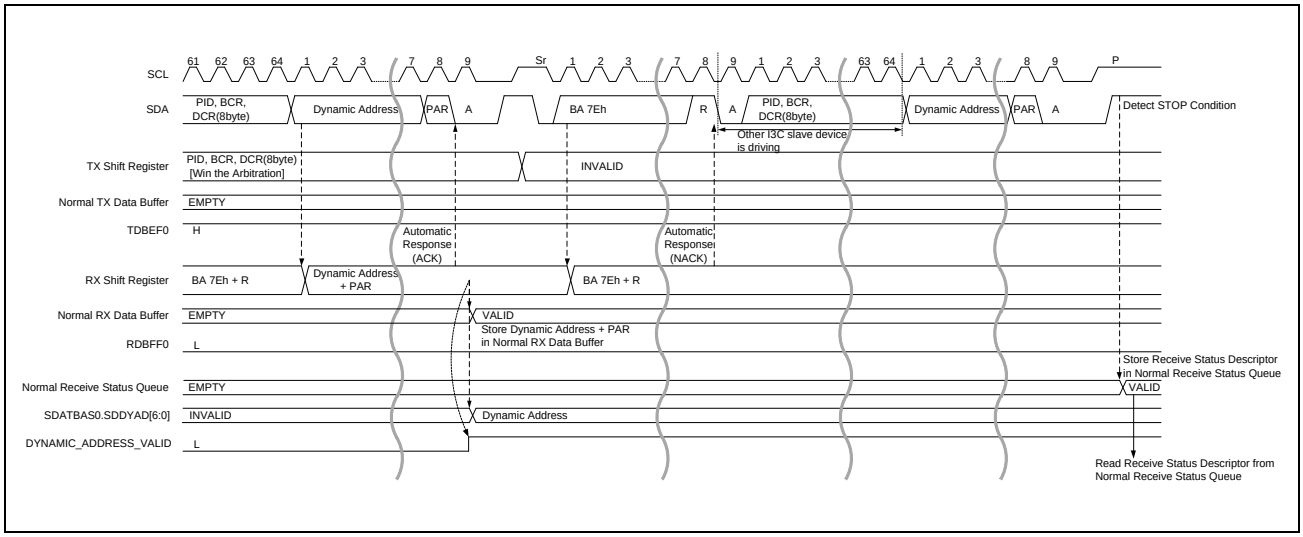


Figure 7.8-33 Dynamic address assign procedure (ENTDAA CCC) timing (3/3)

When communicating with a Static Address until the Dynamic Address is assigned from the I3C Master, by setting to the SDSTAD[6:0] bit of DAT (SDATBASn register), the SSTADV bit of the SVDVADn register is set to 1b and the Static Address Will be effective.

If the I3C Slave has a Static Address and the I3C Master executes the Dynamic Address Assign Procedure, it is possible to assign a Dynamic Address with SETDASA CCC.

The operation of I3C during SETDASA CCC Dynamic Address Assign Procedure is described below.

1. Initial setting (For details, see **(2) I3C Initial Setting Flow**)
2. When SETDASA CCC which agrees with its own Static Address is received, the SDDYAD [6:0] bit of DAT (SDATBAS0 register) is renewed and SDYADV bit of SVDVAD0 register is set in 1b. (For details, see "In case of Direct Write CCC" of **(f) CCC detection function [I3C mode]**.)

3. When SETDASA CCC is completed and a STOP condition is detected, Receive Status Descriptor is stored in Receive Status Buffer.
4. Read Receive Status Descriptor via NRSQP register and check the status.

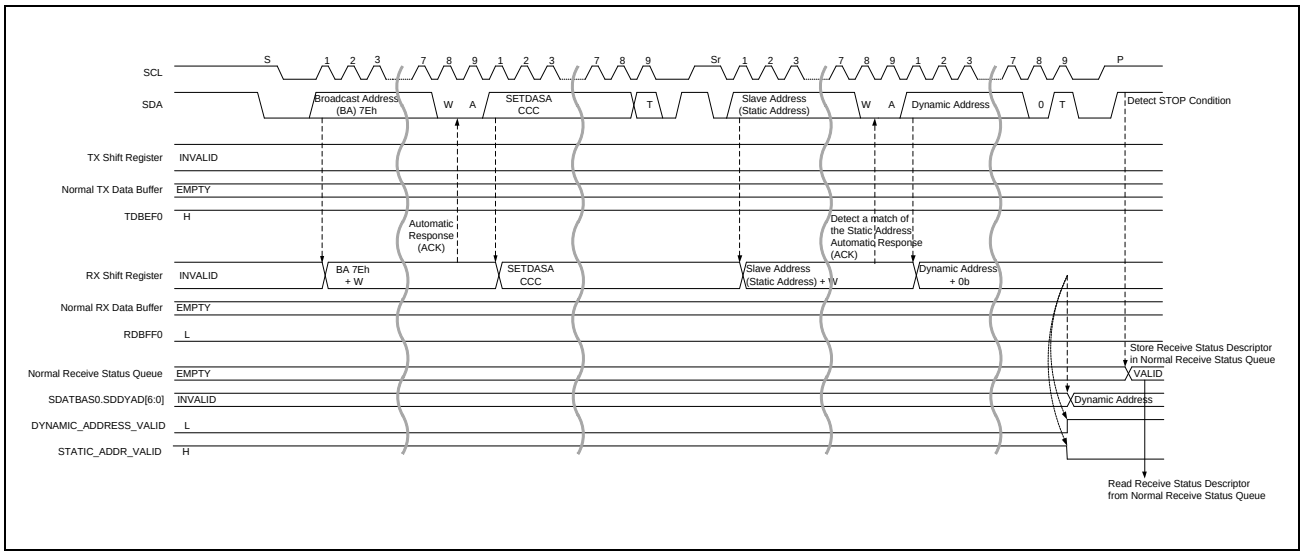


Figure 7.8-34 Dynamic address assign procedure (SETDASA CCC) timing

(b-2) SDR Data Write Transfer

1. When Transaction is issued from the I3C Master, it compares the Slave Address of Address Header with its own Slave Address, and if it matches, I3C responds with ACK.
   When a Transaction is received, if the Receive Data Buffer is full, the I3C Slave will respond with NACK in the Address Header.
   In preparation for retrying the I3C Master, read the data from the Receive Data Buffer via the NTDTBP0 register, and empty the Receive Data Buffer.
2. Data received from I3C Master is stored in the Receive Data Buffer.
3. With the RDBFF0 = 1b interrupt, the received data is read from the Receive Data Buffer via the NTDTBP0 register.
4. When Repeated START condition or STOP condition is detected, the Receive Status Descriptor is stored in the Receive Status Buffer.
5. Read Receive Status Descriptor via NRSQP register and check the status.

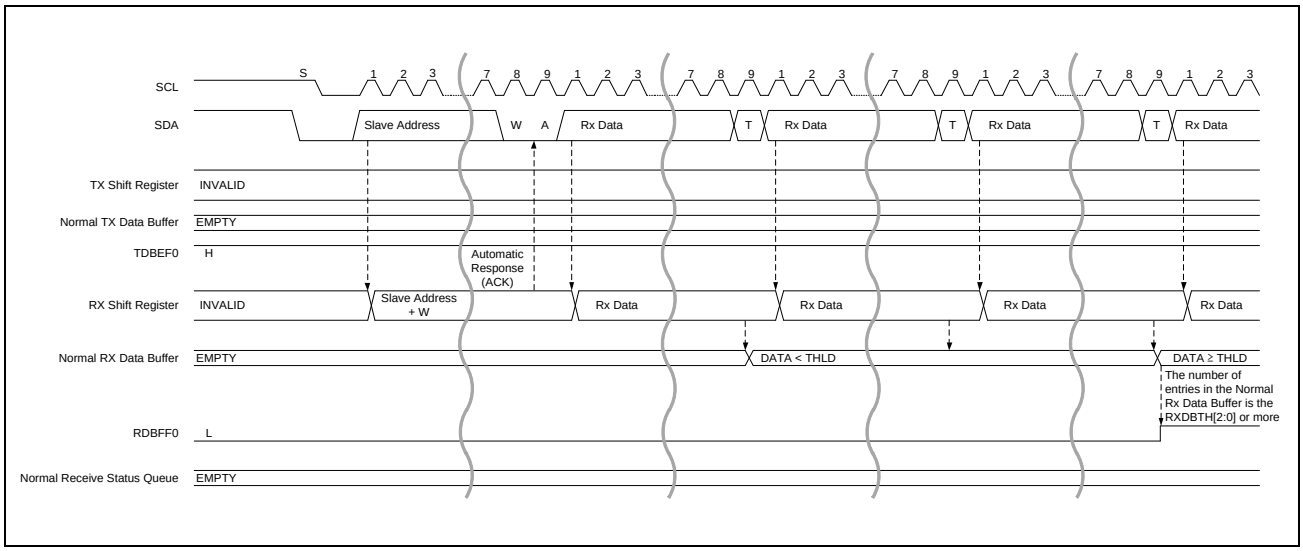


Figure 7.8-35 SDR data write transfer timing (1/2)

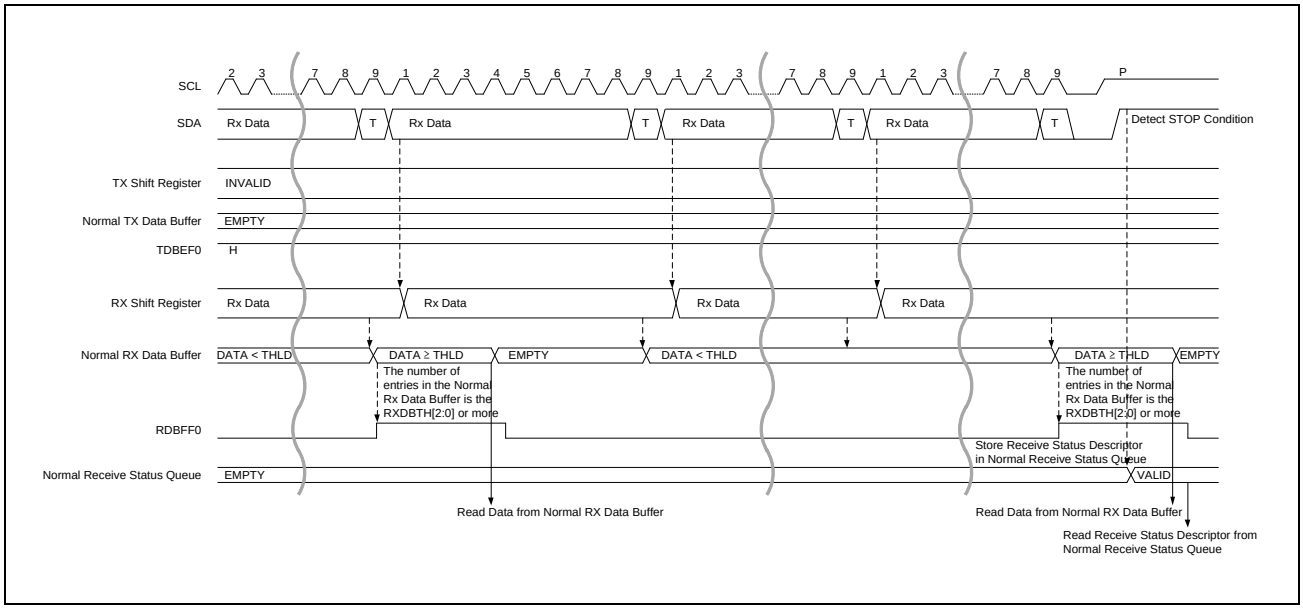


Figure 7.8-36 SDR data write transfer timing (2/2)

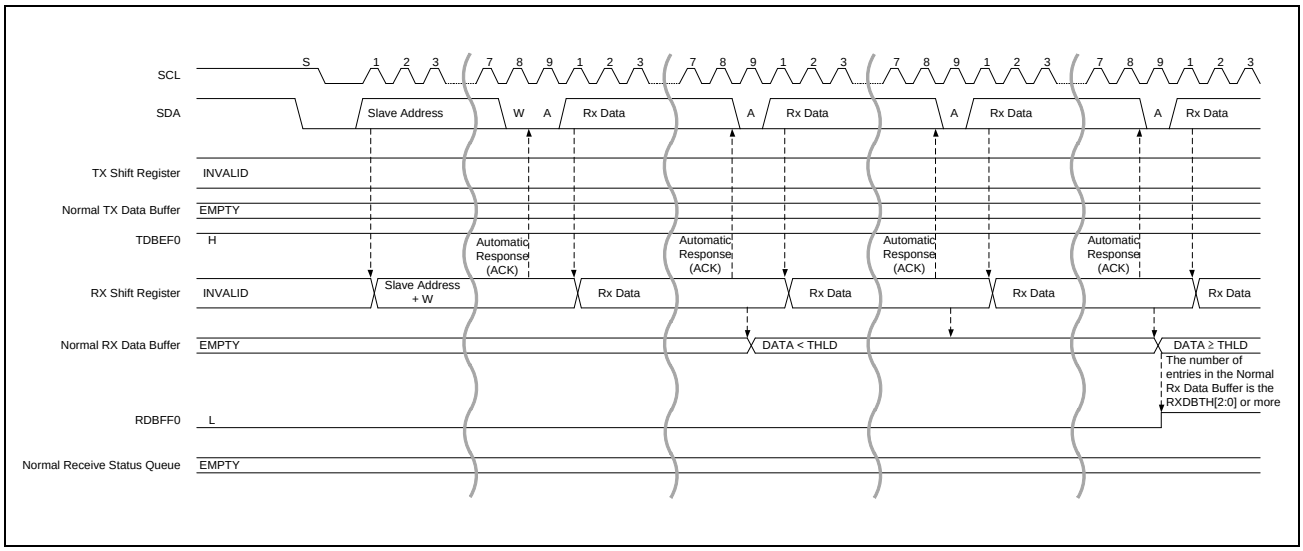


Figure 7.8-37 Legacy I²C message data write transfer timing (1/2)

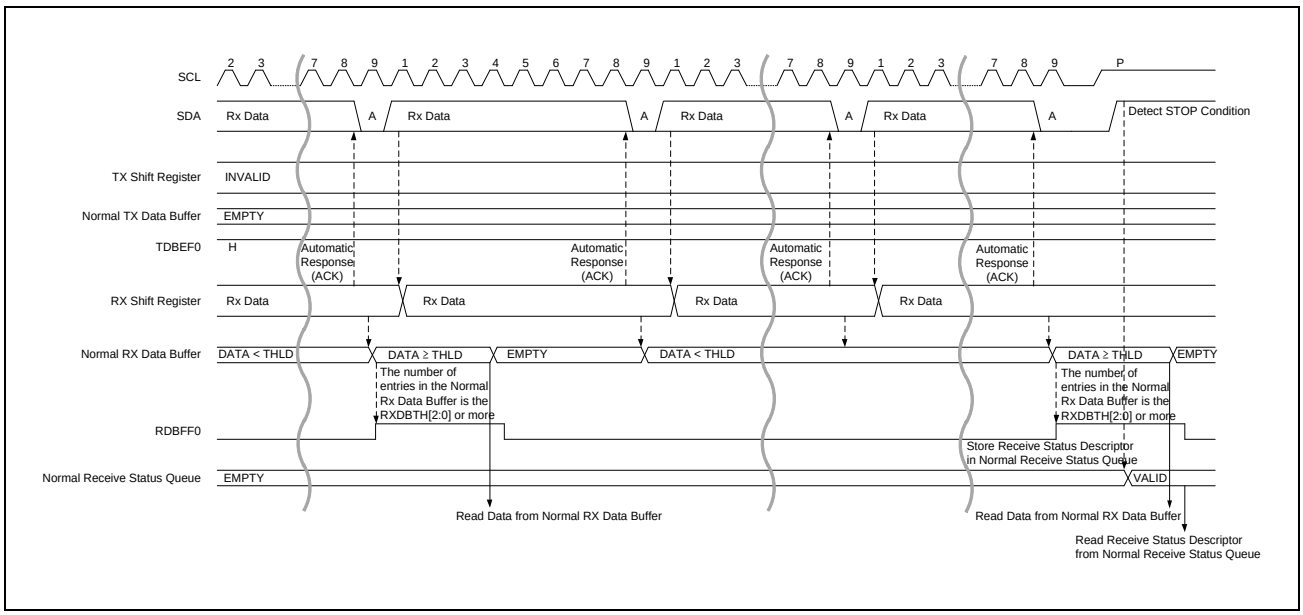


Figure 7.8-38 Legacy I²C message data write transfer timing (2/2)

(b-3) SDR Data Read Transfer

1. Write the data requested from the I3C Master to the Transmit Data Buffer via the NTDTBP0 register.
2. When Transaction is issued from the I3C Master, it compares the Slave Address of Address Header with its own Slave Address, and if it matches, I3C responds with ACK.
   When a Transaction is received, if the Transmit Data Buffer is EMPTY, I3C Slave responds with NACK with the Address Header.
   In preparation for retrying the I3C Master, write data to the Transmit Data Buffer via the NTDTBP0 register.
3. Transmit the data stored in the Transmit Data Buffer.
4. If data to be transmitted still remains, write the data to be transmitted with an interrupt by TDBEF0 = 1b to the Transmit Data Buffer via the NTDTBP0 register.
5. SDR:
   When the transmission of the data stored in the Transmit Data Buffer is completed, Low is output to the T-bit following Data, and it is notified to the I3C Master that it is the final data.
   Legacy I^2C Message:
   When NACK is detected, data transmission is terminated.
6. When a Repeated START condition or STOP condition is detected, the Receive Status Descriptor is stored in the Receive Status Buffer.
7. Read the Receive Status Descriptor via NRSQP and check the status.

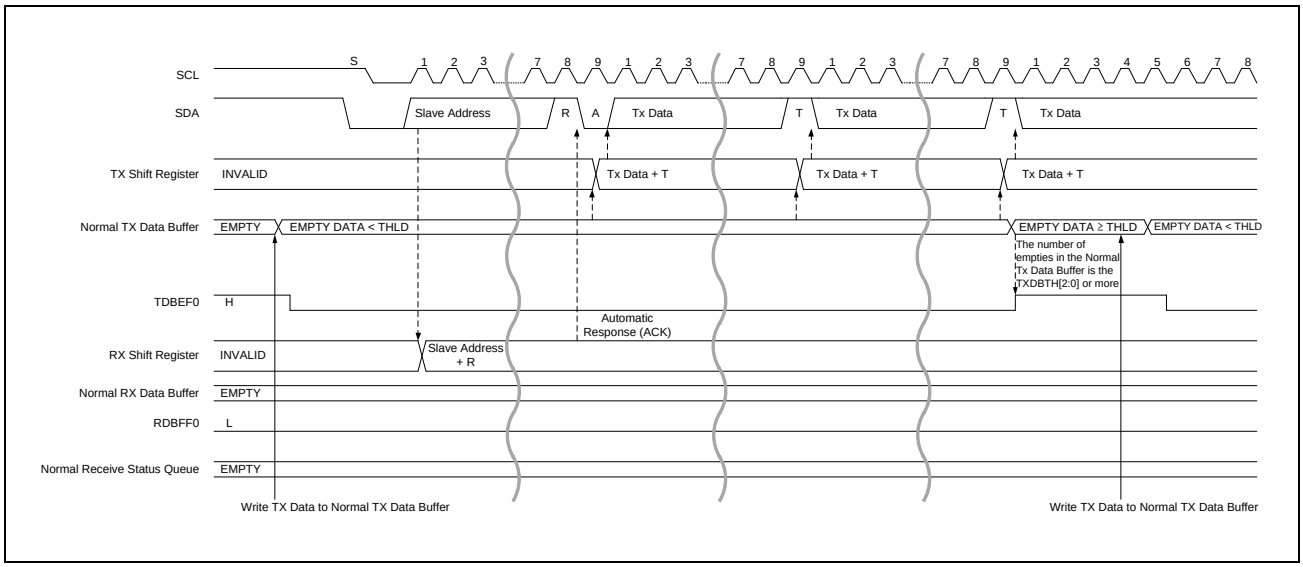


Figure 7.8-39 SDR data read transfer timing (1/2)

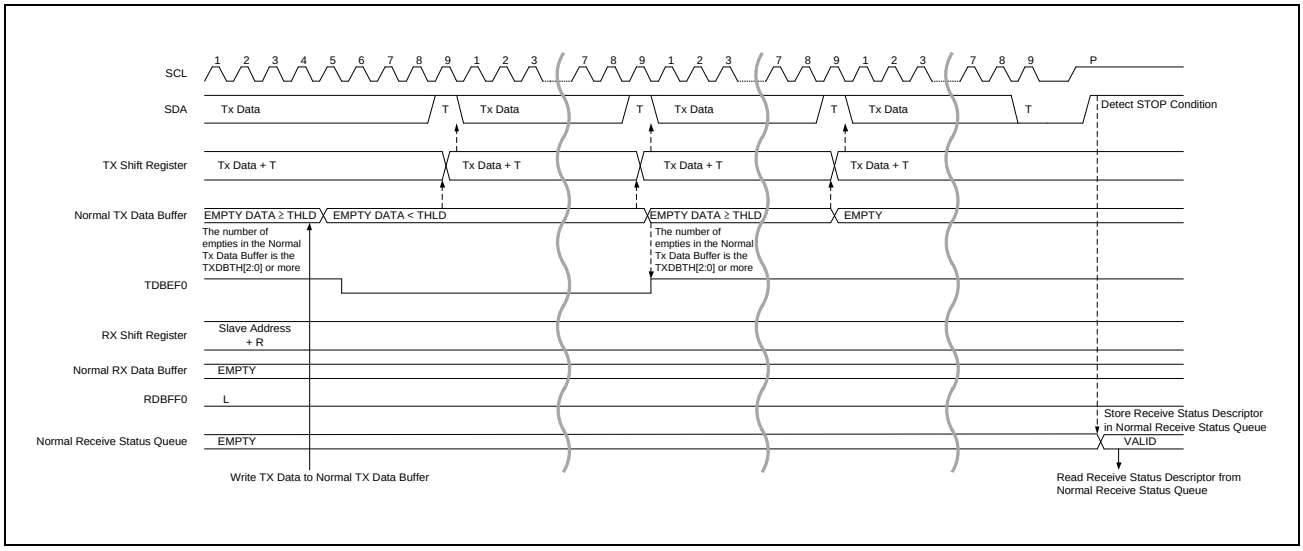


Figure 7.8-40 SDR data read transfer timing (2/2)

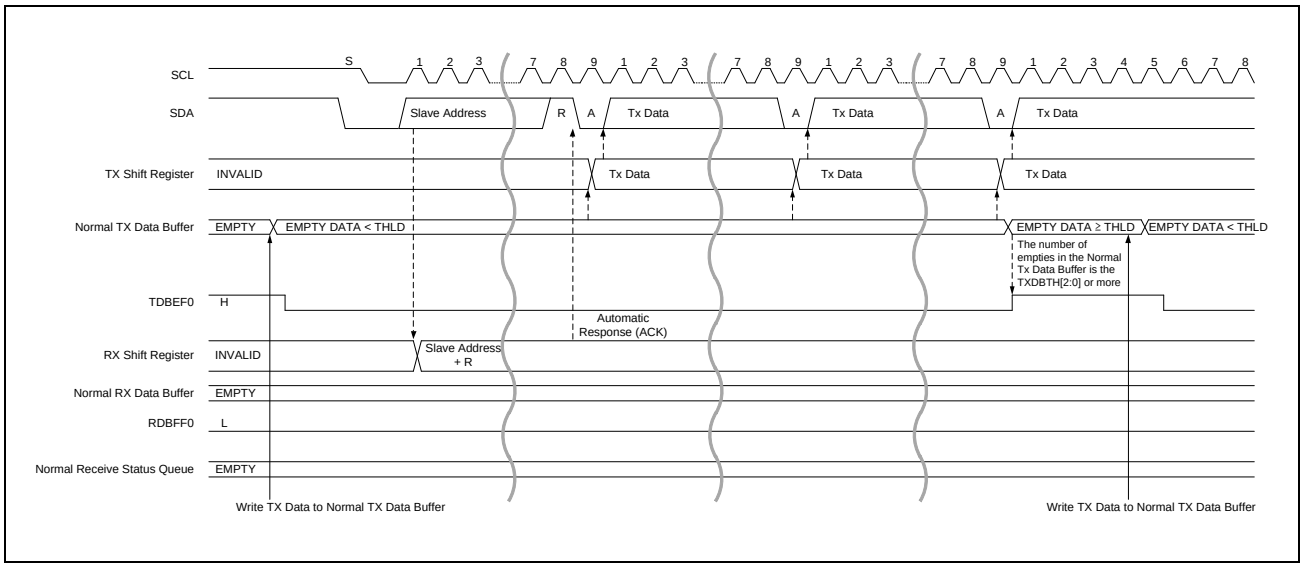


Figure 7.8-41 Legacy I²C message data read transfer timing (1/2)

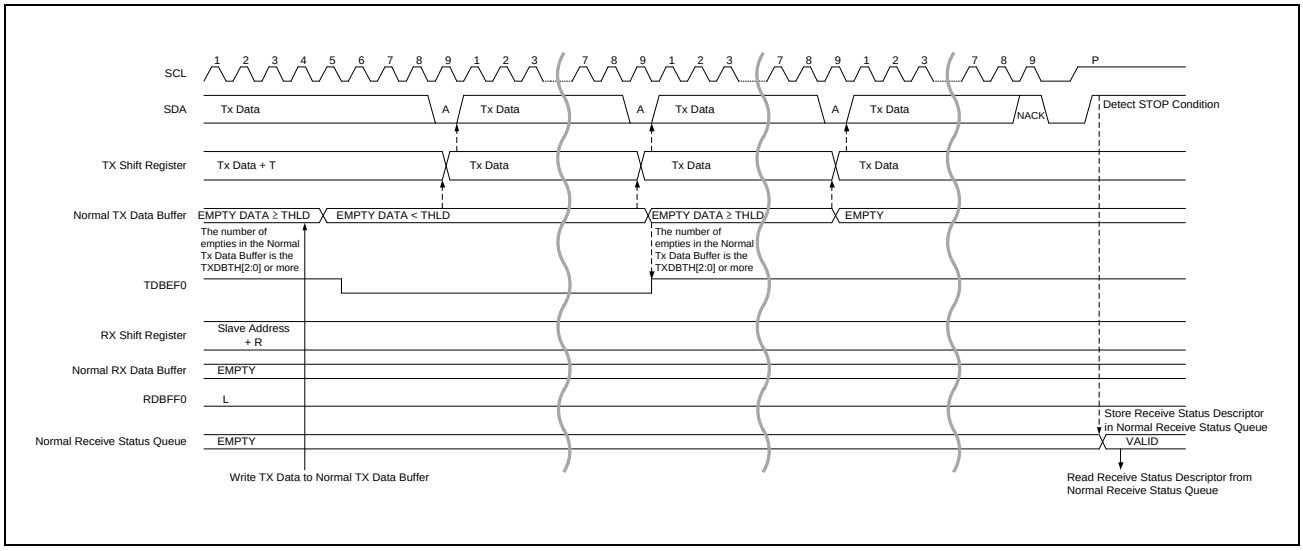


Figure 7.8-42 Legacy I²C message data read transfer timing (2/2)

(b-4) IBI Transfer

1. When sending Slave Interrupt Request.
   When transmitting IBI Data, write IBI Data to the IBI Data Buffer via the NIBIQP register.
2. Write Command Descriptor (Immediate Transfer Command or Regular Transfer Command) to the Command Buffer for IBI Transfer via the NCMDQP register.
3. When Command Descriptor is written to Command Buffer, IBI Transaction is issued under the following conditions.
   - When START condition is detected in Slave Interrupt Request or Mastership Request. (Does not apply a Repeated START condition)
   - If no START is forthcoming within the following Bus Condition, then this module issue a START Request by pulling the SDA line Low.
     a) Slave Interrupt Request, Mastership Request: Bus Available
     b) Hot-Join Event: Bus Idle
4. In Slave Address with RnW of the Address Header, if losing Arbitration by issuing a Transaction from I3C Master, stop issuing Transaction.
   When detecting Repeated START condition or STOP condition, store the Response Descriptor into the Response Buffer.
5. When sending Slave Interrupt Request:
   - When IBI data for transmission still remain, write IBI data with an interrupt by IBIQEFF = 1b to the IBI Data Buffer via the NIBIQP register.
   - When the transmission of IBI Data for the number of Data Length specified by the DATA_LENGTH[15:0] bits of the Command Descriptor is completed, output Low to the T-bit following IBI Data and notify the I3C Master that it is the final IBI Data.
6. When detecting Repeated START condition or STOP condition, store the Response Descriptor into the Response Buffer.
7. Read the Response Descriptor form the Response Buffer with the NRSPQP register and check the status. If NACK is responded, repeat steps 1 to 7.
8. When sending Slave Interrupt Request:
   Check that the value of the DATA_LENGTH[15:0] bit of the Response Descriptor is 0b.

The Mastership processing flow is shown in **Figure 7.8-45**. When joining the I3C Bus by Hot-Join after the I3C Bus has already been configured, issue the Hot-Join according to the flow shown in **Figure 7.8-126**.

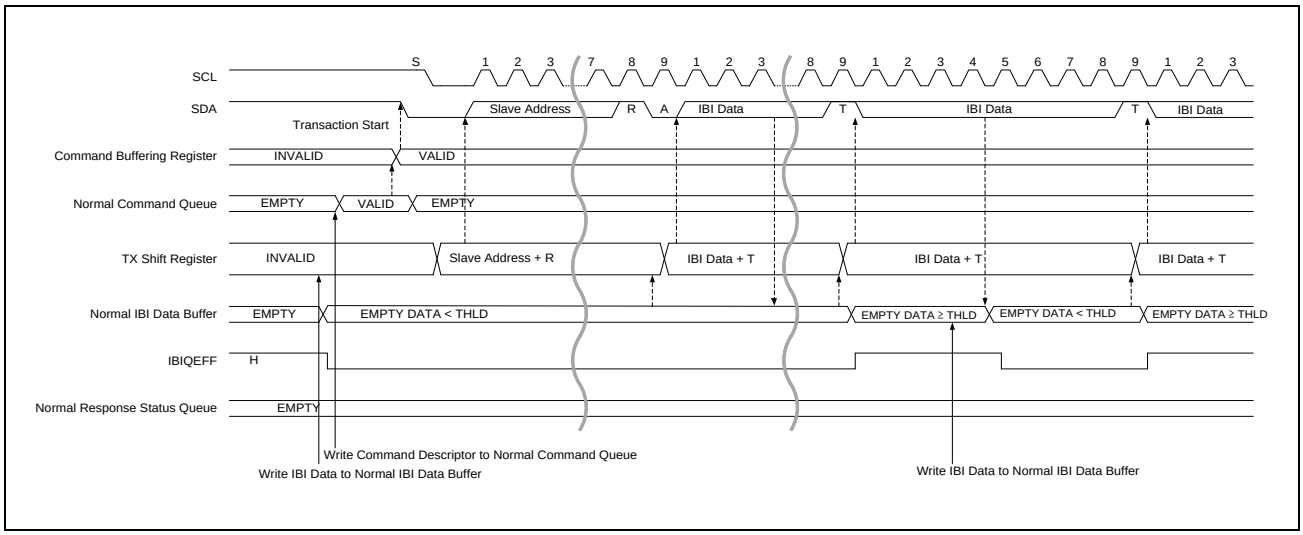


Figure 7.8-43 I3C slave IBI transfer timing (1/2)

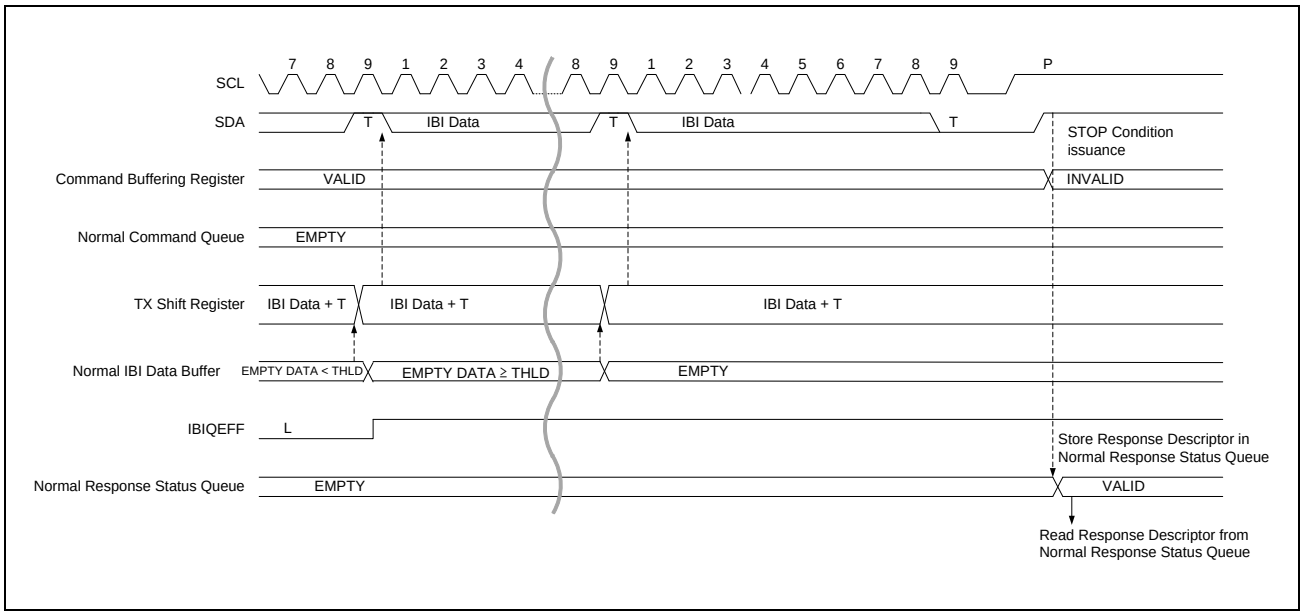


Figure 7.8-44 I3C slave IBI transfer timing (2/2)

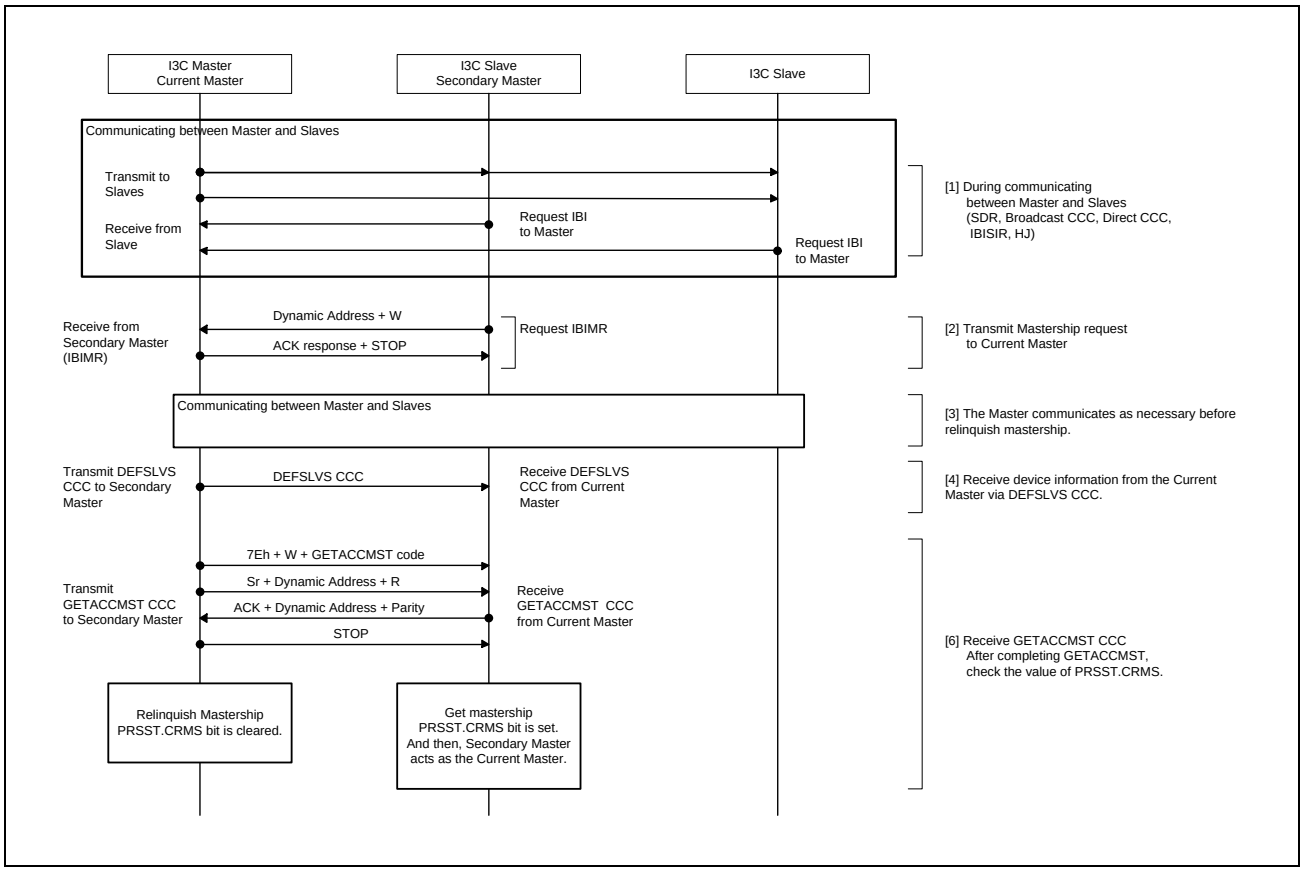


Figure 7.8-45 I3C Slave Mastership processing flow

7.8.4.2.2 Data Handler

The relationship between the transfer method and the queue is shown in **Table 7.8-11**.

Table 7.8-11 Transfer method and queue

Protocol	Transfer method	Queue/Buffer	size	Master	Slave	Secondary Master
I^2C Mode	Single buffer transfer	Normal Transmit Data	1 byte	✓	✓	—
		Normal Receive Data	1 byte	✓	✓	—
I3C Mode	Normal FIFO buffer transfer	Normal Command	4 QUEUEs	✓	✓	✓
		Normal Response Status	4 QUEUEs	✓	✓	✓
		Normal Transmit Data	16 DWORDs	✓	✓	✓
		Normal Receive Data	16 DWORDs	✓	✓	✓
		Normal Receive Status	2 QUEUEs	—	✓	✓
		Normal IBI Status	2 QUEUEs	✓	—	✓
		Normal IBI Data	8 DWORDs	✓	✓	✓

(1) Transfer Method in I²C Mode

(a) Single Buffer transfer

Each process (condition issue, data transfer, ACK/NACK response) is controlled by software.

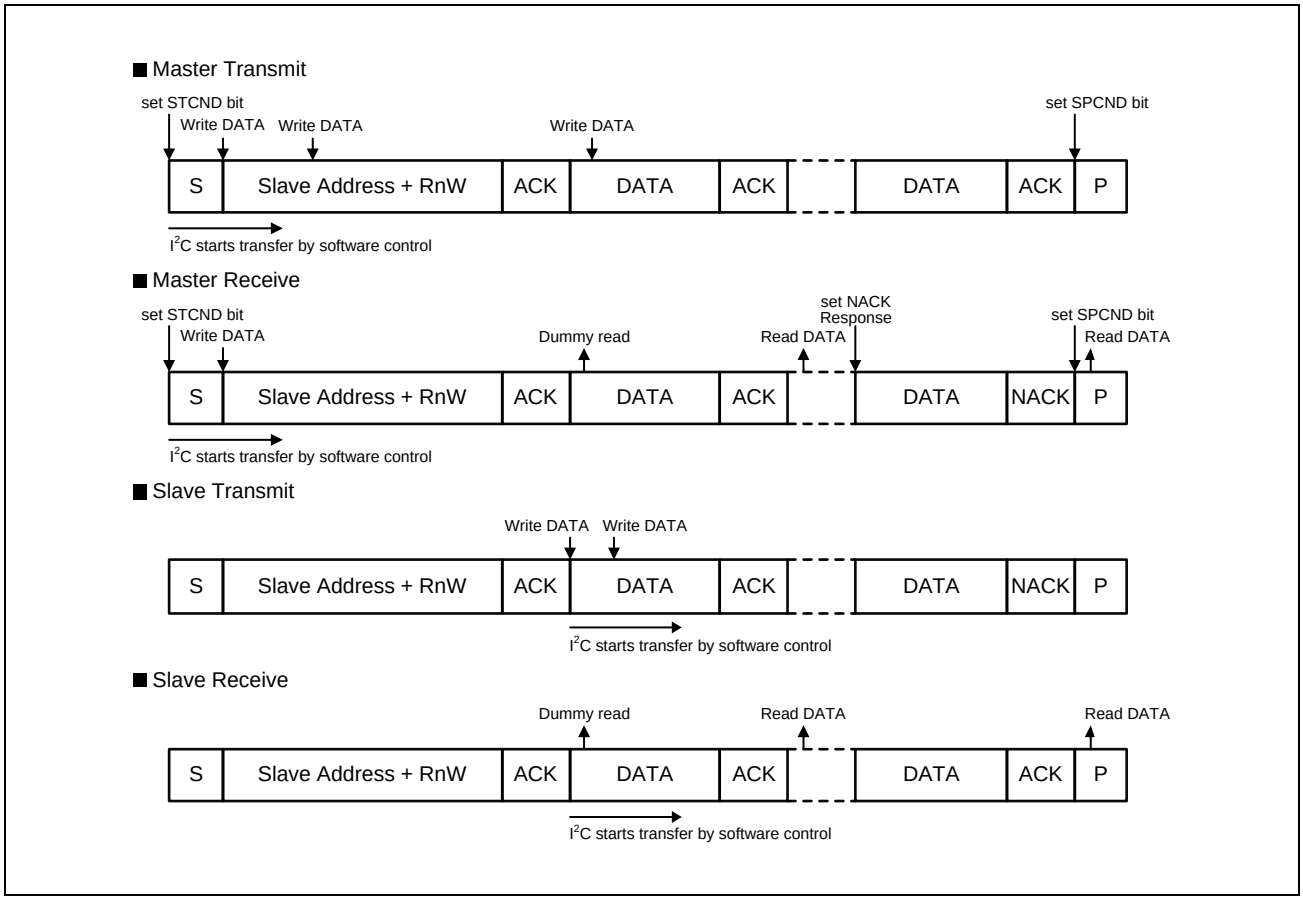


Figure 7.8-46 Data handler with single buffer transfer

(2) Transfer Method in I3C Mode

(a) Normal FIFO Buffer Transfer

I3C autonomously starts transfer when data and command are written.

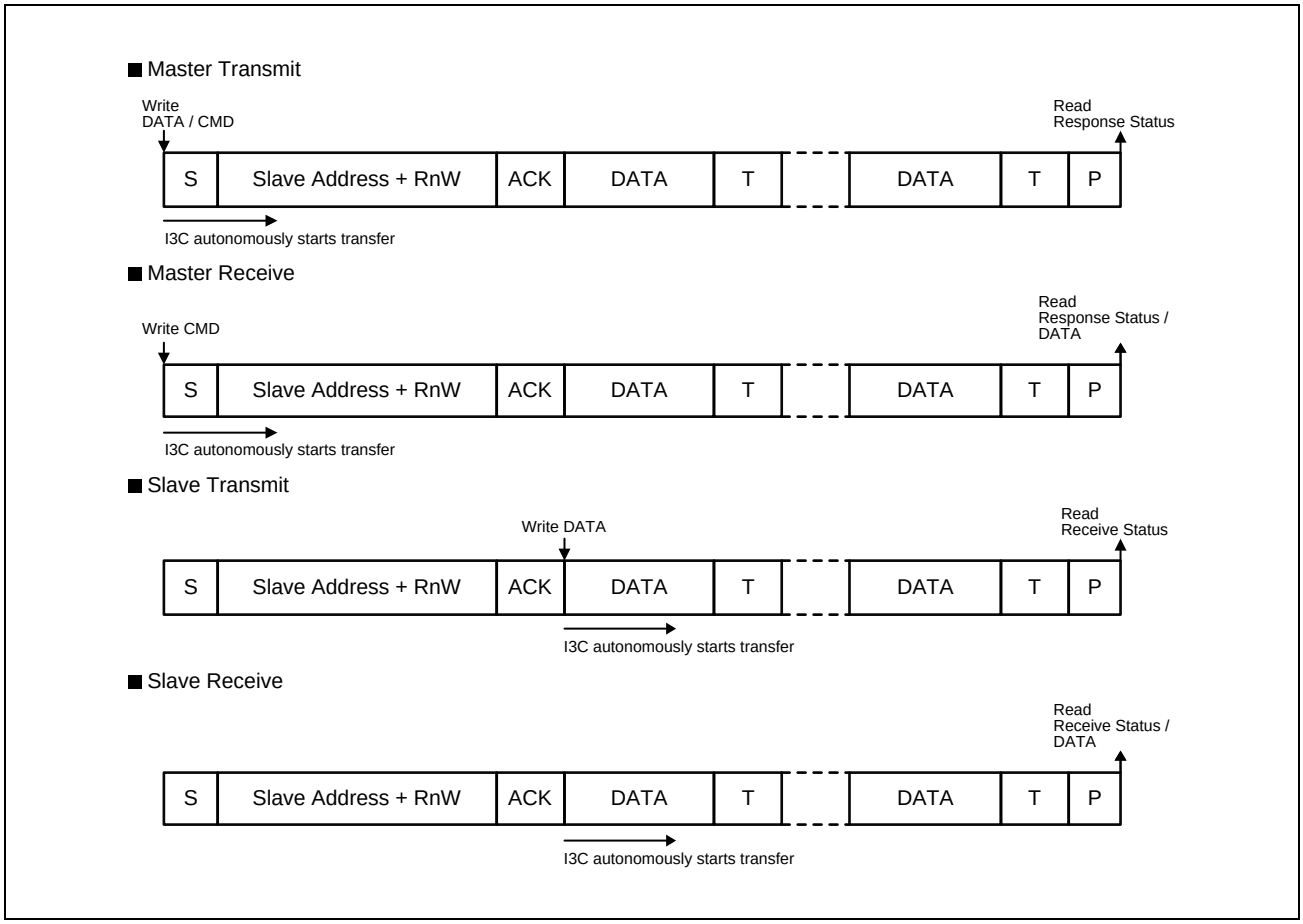


Figure 7.8-47 Data handler with normal FIFO buffer transfer

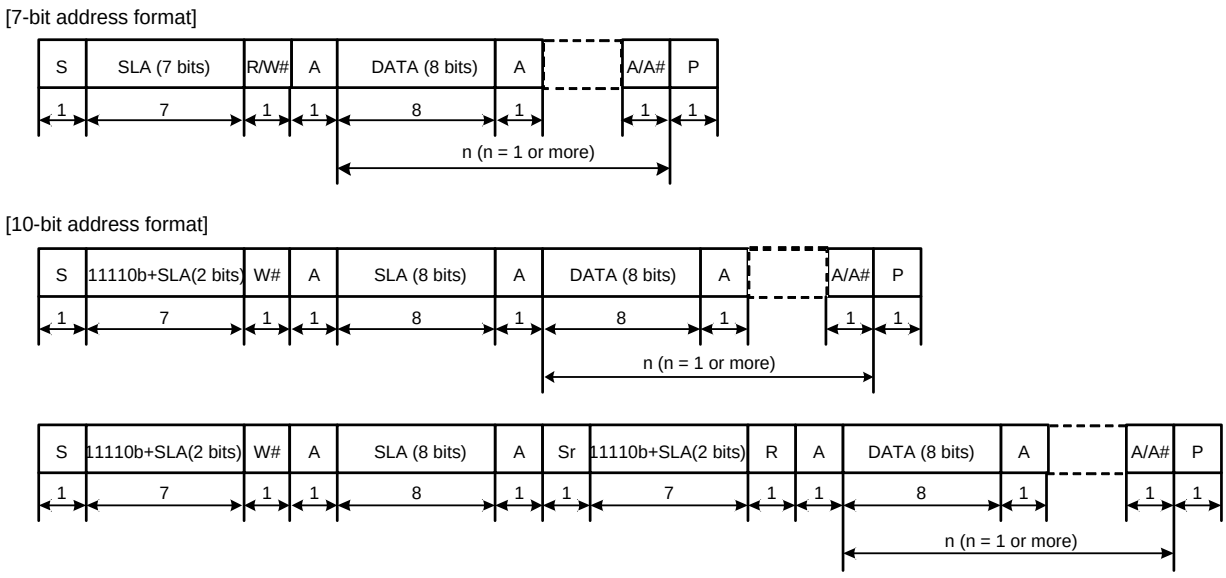
7.8.4.2.3 I²C/I3C Protocol

(1) Communication Protocol

(a) I²C Communication Data Format

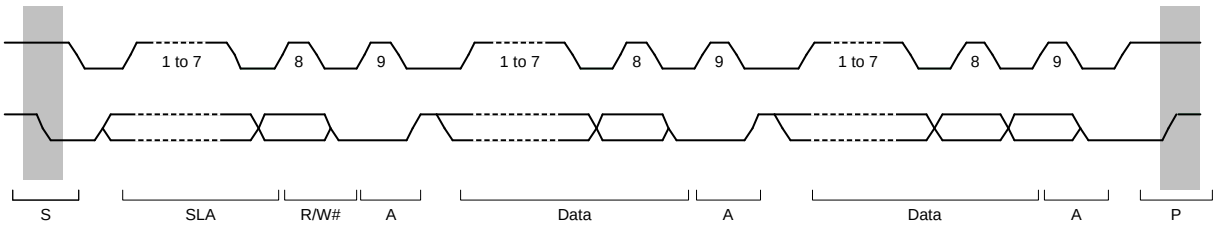
The I²C bus format consists of 8-bit data and 1-bit acknowledge. The frame following a START condition or Repeated START condition is an address frame used to specify a slave device with which the master device communicates. The specified slave is valid until a new slave is specified or a STOP condition is issued.

Figure 7.8-48 shows the I²C bus format, and **Figure 7.8-49** shows the I²C bus timing.



Note: n: Number of transmit DATA bytes

Figure 7.8-48 I²C bus format



S: START condition. The master device drives the SDA line low from high level while the SCL line is at a high level.

SLA: Slave address, by which the master device selects a slave device.

R/W#: Indicates the direction of data transfer: from the slave device to the master device when R/W = 1, or from the master device to the slave device when R/W = 0.

A: Acknowledge. The receive device drives the SDA line low. (In master transmit mode, the slave device returns acknowledge. In master receive mode, the master device returns acknowledge.)

A#: Not Acknowledge. The receive device drives the SDA line high.

Sr: Repeated START condition. The master device drives the SDA line low from the high level after the setup time has elapsed with the SCL line at the high level.

DATA: Transmitted or received data

P: STOP condition. The master device drives the SDA line high from low level while the SCL line is at a high level.

Figure 7.8-49 I^2C bus timing (SLA = 7 bits)

(b) I3C Communication Data Format

Figure 7.8-50 through **Figure 7.8-53** illustrate a typical communication for each of the four I3C Protocols. While these diagrams do not exhaustively illustrate all possible I3C communications, they do serve as useful introductions to the signaling and transmission formatting used in each I3C Protocol.

Figure 7.8-50 illustrates example communication using I3C Single Data Rate (SDR) coding with Broadcast (7Eh). It shows the Master reading a byte of data from the Slave at Address 2Bh in SDR Mode. From the Bus Free Condition, the Master issues a START by driving the SDA line Low while keeping the SCL line High. It then issues the Broadcast Address (7Eh) followed by RnW (0 for Write). Then the Master turns on a pull-up resistor and goes to Open Drain.

All Slaves ACK by pulling the SDA line Low (in the Figure, pink fill means the Slave is in control of the SDA line at this time). The Master then issues a Repeated START, then the Address of the Slave (2Bh) it wants to read followed by RnW (1 for Read). The Master then turns on a pull-up resistor and goes to Open Drain, allowing the Slave to acknowledge by pulling the SDA line Low. At this point, the Master continues to toggle the SCL line and release the SDA line, allowing the Slave to drive SDA to send one byte of data (4Ah) followed by T. T = 1b informs the Master that there is additional data, whereas T = 0b signals the end. Here there is additional data, so the Slave drives SDA High until SCL goes High, at which time it releases SDA. The Master has the option of holding SDA High with a weak pullup, which signals to the Slave that the Master allows another byte to be transmitted, or to pull SDA Low (while SCL is High – hence a Repeated START), which would signal to the Slave that the Master has terminated the Read and is taking over.

SDR Mode is backwards compatible with Legacy I^2C Devices, because the High time of an SCL pulse is always less than 50ns and therefore SCL will always appear to be Low because of the I^2C 50ns Spike Filter.

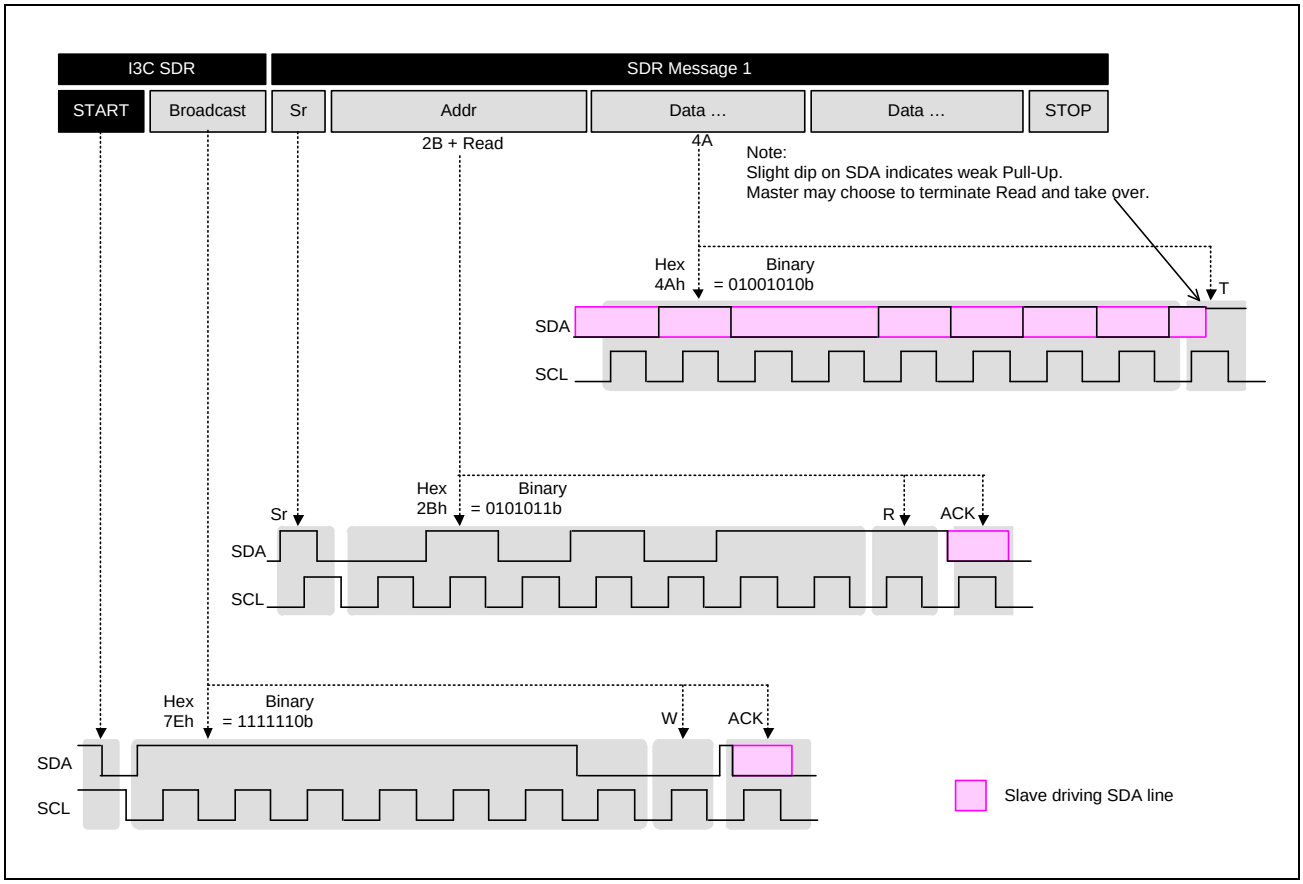


Figure 7.8-50 Example communication using I3C coding SDR with broadcast (7Eh)

Figure 7.8-51 illustrates example communication using I3C Single Data Rate (SDR) coding without Broadcast (7Eh). It shows the Master reading a byte of data from the Slave at Address 2Bh in SDR Mode. From the Bus Free Condition, The Master then issues a START, then the Address of the Slave (2Bh) it wants to read followed by RnW (1 for Read).

The Master then turns on a pull-up resistor and goes to Open Drain, allowing the Slave to acknowledge by pulling the SDA line Low. At this point, the Master continues to toggle the SCL line and release the SDA line, allowing the Slave to drive SDA to send one byte of data (4Ah) followed by T. T = 1b informs the Master that there is additional data, whereas T = 0b signals the end. Here there is additional data, so the Slave drives SDA High until SCL goes High, at which time it releases SDA. The Master has the option of holding SDA High with a weak pull-up, which signals to the Slave that the Master allows another byte to be transmitted, or to pull SDA Low (while SCL is High – hence a Repeated START), which would signal to the Slave that the Master has terminated the Read and is taking over.

SDR Mode is backwards compatible with Legacy I^2C Devices, because the High time of an SCL pulse is always less than 50ns and therefore SCL will always appear to be Low because of the I^2C 50ns Spike Filter.

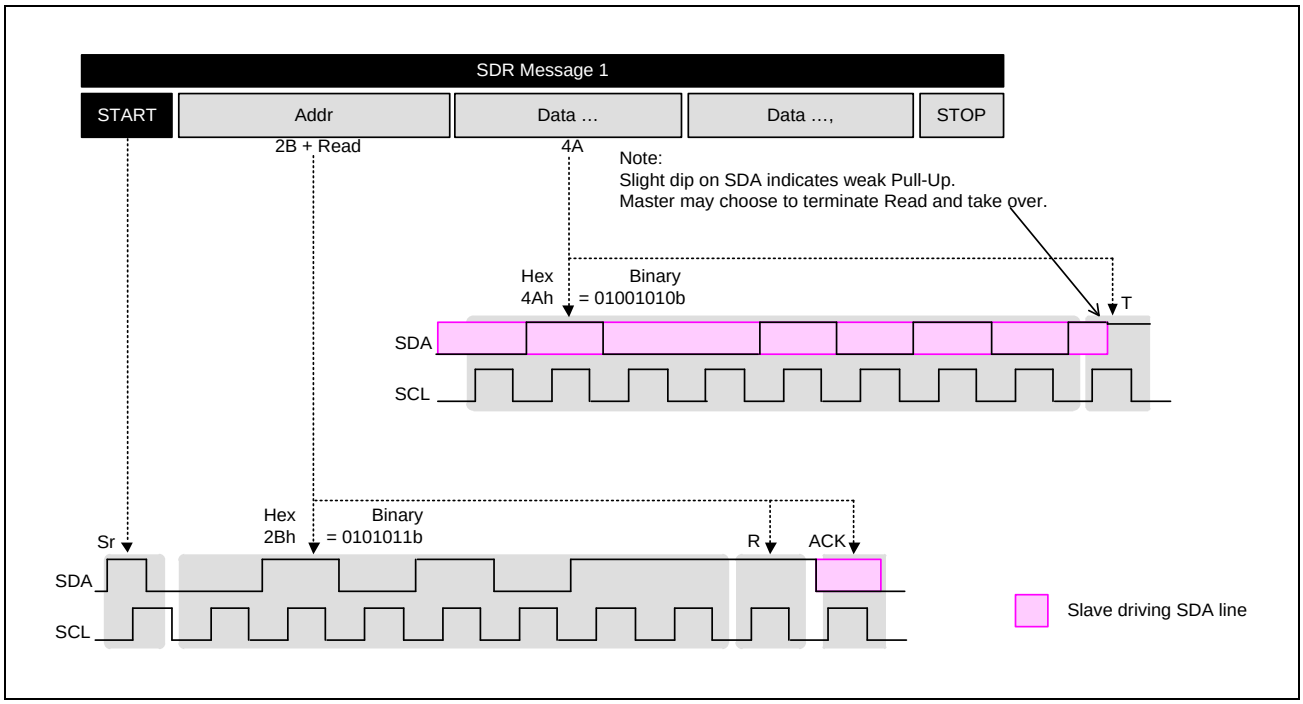


Figure 7.8-51 Example communication using I3C coding SDR without broadcast (7Eh)

Figure 7.8-52 shows the Master issuing a CCC Direct Command to a single Slave. This particular command (GETPID) reads the Provisional ID of a Slave.

From the Bus Free Condition, the Master issues a START by driving the SDA line Low while keeping the SCL line High. It then issues the Broadcast Address (7Eh) followed by RnW (0h for Write). Then the Master turns on a pull-up resistor and goes to Open Drain. All Slaves ACK by pulling SDA Low (in the Figure, pink fill means the Slaves are in control of SDA at this time). The Master then issues the Direct Common Command Code for GETPID (8Ch) followed by parity bit T (odd parity = 0 for 8Ch) then the 7-bit Dynamic Address of the Slave (chosen arbitrarily here to be 2Bh) followed by a RnW bit (1 for Read). Then the Master turns on a pull-up resistor and goes to Open Drain, allowing the Slave at Address 2Bh to ACK by pulling SDA Low, which tells the Master that the Slave Acknowledges the command and will comply. (Alternatively, the Slave may NACK by not pulling SDA Low, which would inform the Master that the Slave will not comply – in this case, that an error occurred.) Following the ACK the Slave outputs its 48-bit PID one byte at a time, and then the Master issues a Repeated START (this part of the waveform sequence is not shown in the Figure).

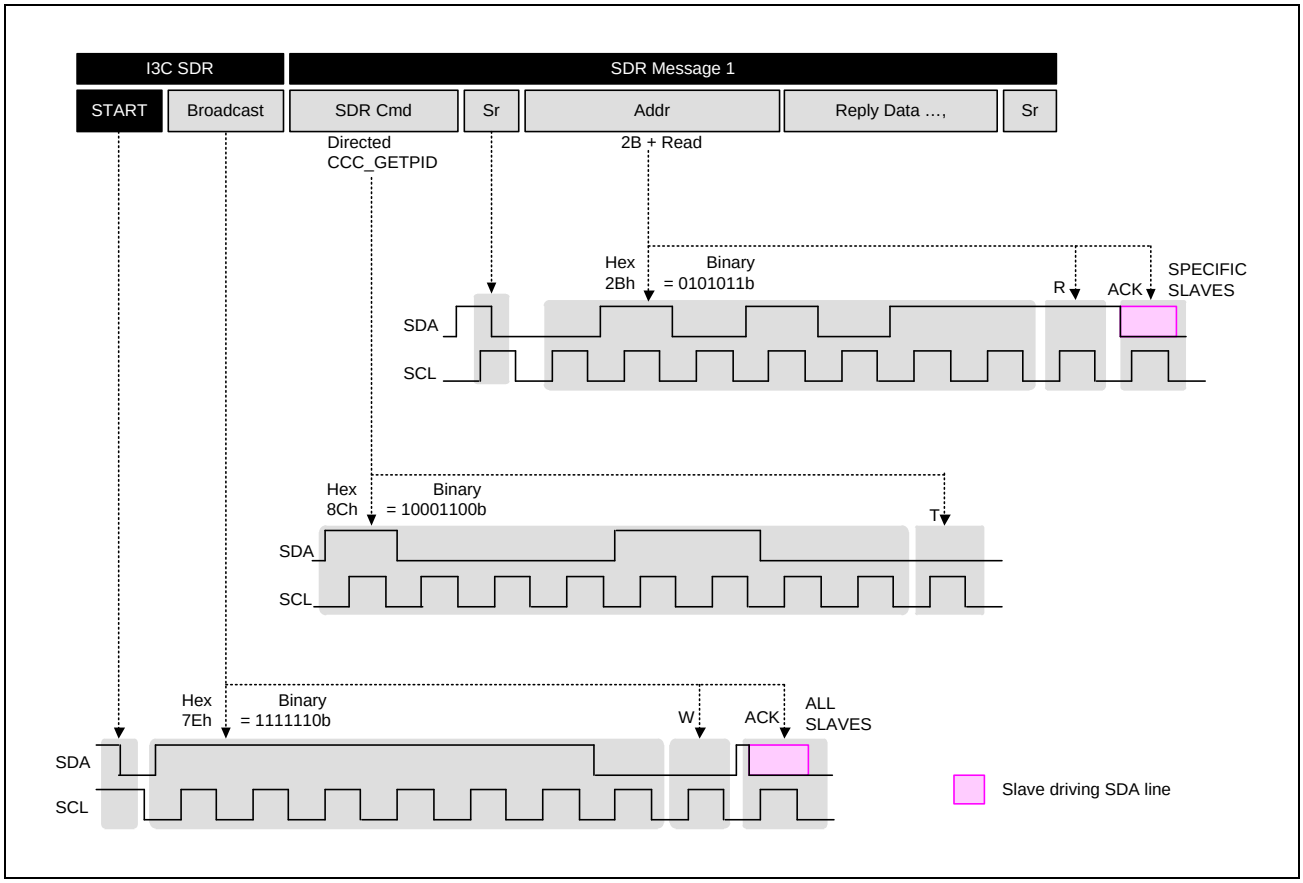


Figure 7.8-52 Example communication using I3C coding SDR with CCC direct addressing

Figure 7.8-53 illustrates example SDR communication with a CCC Broadcast command. The command used in this example sets the Maximum Read Length of all Slaves to 43 bytes (002Bh).

From the Bus Free Condition, the Master issues a START by driving the SDA line Low while keeping the SCL line High. It then issues the Broadcast Address (7Eh) followed by RnW (0h for Write). Then the Master turns on a pull-up resistor and goes to Open Drain. All Slaves ACK by pulling SDA Low (in the Figure, pink fill means the Slaves are in control of SDA at this time). The Master then issues the Broadcast Common Command Code for SETMRL (09h) followed by parity bit T (odd parity = 1 for 09h), and then 2 data bytes (MSB first) to define the maximum number of bytes which can be read from a Slave in a single read operation. Each data byte is followed by a T bit (parity bit – odd parity). After this the Master issues a Repeated START.

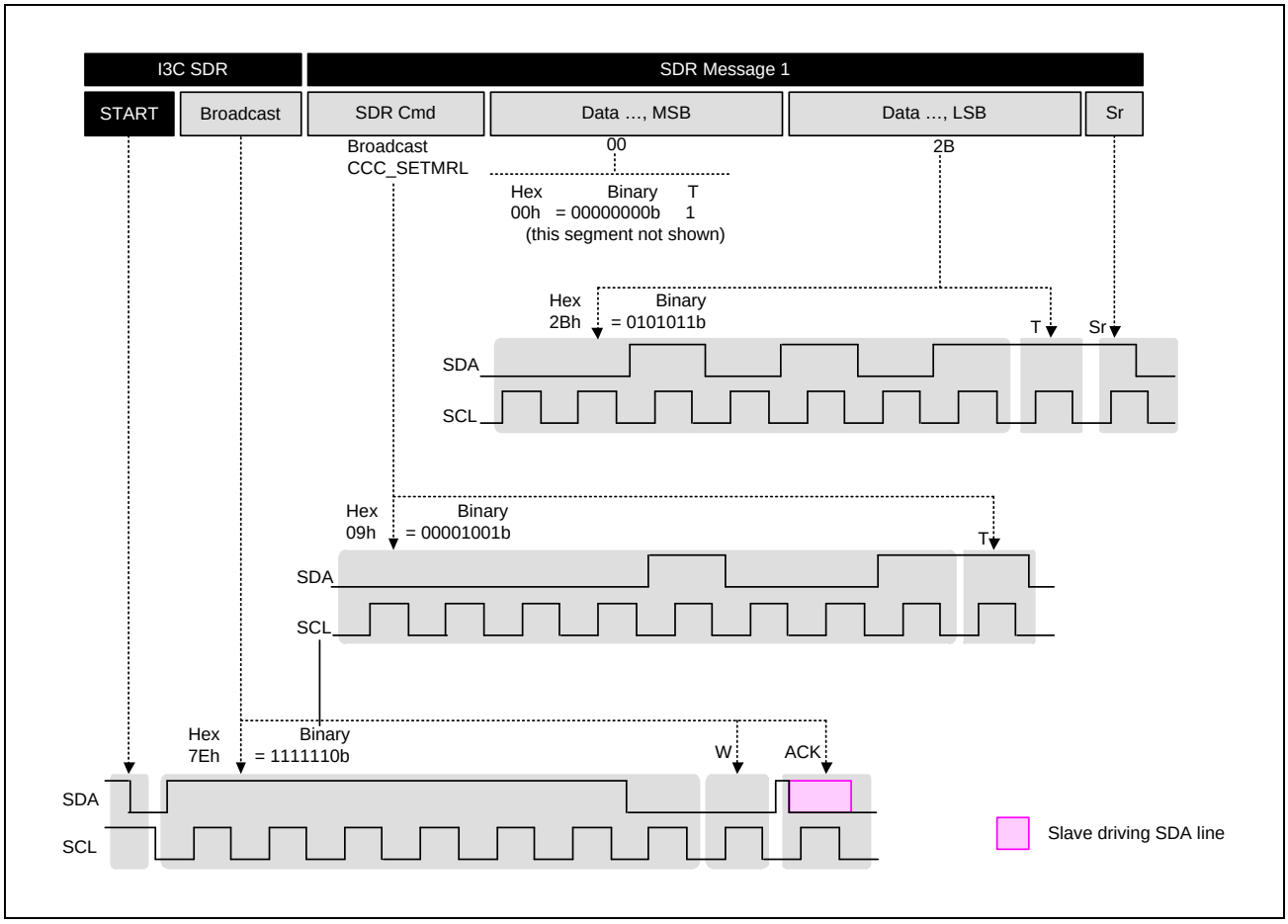


Figure 7.8-53 Example communication using I3C coding SDR with CCC broadcast

(2) Bus Conditions

I3C defines three distinct conditions in which the I3C Bus shall be considered inactive: Bus Free, Bus Available, and Bus Idle (see **Figure 7.8-54**).

(a) Bus Free Condition

State on the I3C Bus where both the SCL line and the SDA line are High for at least the period set by BFRECDT.FRECYC[8:0] bit.

(b) Bus Available Condition [I3C mode]

State on the I3C Bus where both the SCL line and the SDA line are High for at least the period set by BAVLCDT.AVLCYC[8:0] bit.

A Slave may only issue a START Request (For example, for an In-Band Interrupt, or for a Master Handoff Request) after a Bus Available Condition.

(c) Bus Idle Condition [I3C mode]

State on the I3C Bus where both the SCL line and the SDA line are High for at least the period set by BIDLCDT.IDLCYC[17:0] bit.

A Slave may only issue a START Request (For example, for a Hot-Join) after a Bus Idle Condition. Specifications are as follows. IDLE needs to be the largest.

BFRECDT.FRECYC[8:0] < BAVLCDT.AVLCYC[8:0] < BIDLCDT.IDLCYC[17:0]

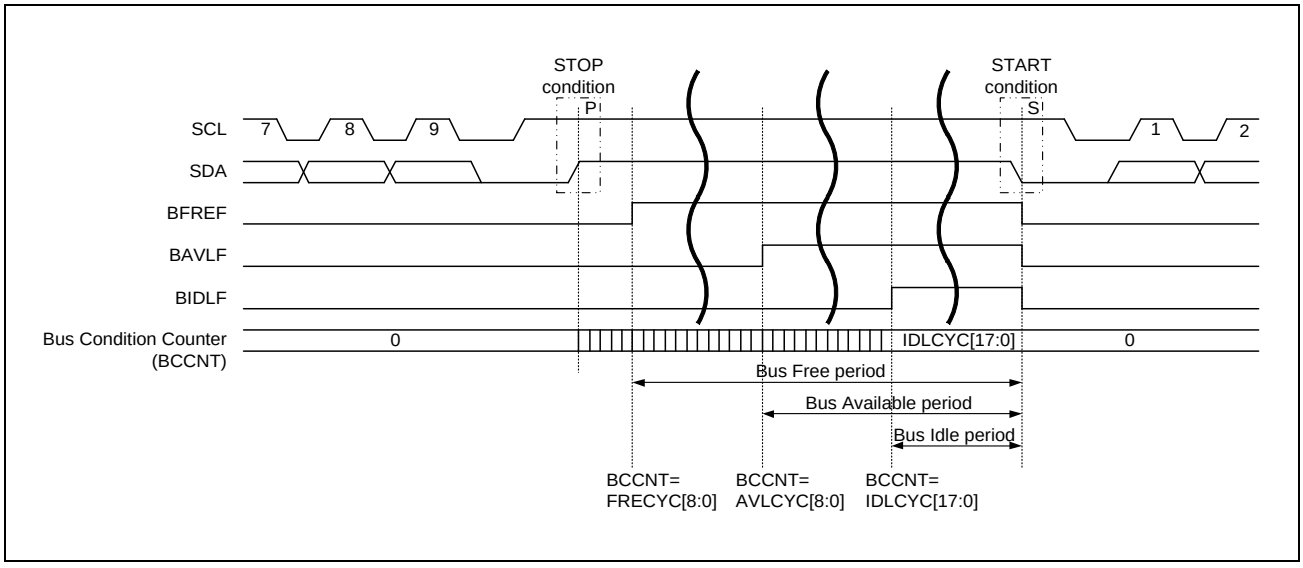


Figure 7.8-54 Bus conditions

(3) START Condition / Repeated START Condition / STOP Condition Issuing Function

(a) Issuing a START Condition

I3C issues a START condition when the CNDCTL.STCND bit is set to 1.

Set the STCND bit to 1b (START condition issuance request) when the BCST.BFREF flag is set to 1b (bus free state).

I3C issues a START condition.

When a START condition is issued normally, I3C automatically shifts to the master transmit mode. A START condition is issued in the following sequence.

[START condition issuance]

- Drive the SDA line low (high level to low level).
- Ensure the time set in STDBR.SBRHO[7:0] and the START condition hold time.
- Drive the SCL line low (high level to low level).
- Detect low level of the SCL line and ensure the low-level period of SCL line set in STDBR.SBRLO[7:0].

(b) Issuing a Repeated START Condition

I3C issues a Repeated START condition when the CNDCTL.SRCND bit is set to 1.

When the SRCND bit is set to 1b, a Repeated START condition issuance request is made and I3C issues a Repeated START condition when the BCST.BFREF flag = 0b (bus busy state) and the PRSST.CRMS bit = 1b (master mode).

A Repeated START condition is issued in the following sequence. [Repeated START condition issuance]

- Release the SDA line.
- Ensure the low-level period of SCL line set in STDBR.SBRLO[7:0] or EXTBR.EBRLO[7:0].
- Release the SCL line (low level to high level).
- Detect a high level of the SCL line and ensure the time set in STDBR.SBRLO[7:0] or EXTBR.EBRLO[7:0] and the Repeated START condition setup time.
- Drive the SDA line low (high level to low level).
- Ensure the time set in STDBR.SBRHO[7:0] or EXTBR.EBRHO[7:0] and the Repeated START condition hold time.
- Drive the SCL line low (high level to low level).
- Detect a low level of the SCL line and ensure the low-level period of SCL line set in STDBR.SBRLO[7:0] or EXTBR.EBRLO[7:0].

NOTE

When issuing Repeated START conditions request, write the slave address to NTDTBP0 after confirming CNDCTL.SRCND = 0. Data written in the period of CNDCTL.SRCND = 1 is not forwarded because retransmission condition before the occurrence.

To issue a Repeated START condition in Hs-mode, follow the steps below.

1) Wait until PRSTDBG.SCOLV = 0b.
2) Set EXTBR.EBRHO[7:0] to satisfy the hold time of the Repeated START condition.
3) Set the CNDCTL.SRCND bit to 1b.
4) After confirming CNDCTL.SRCND = 0b, wait until PRSTDBG.SCOLV = 0b.
5) Set EXTBR.EBRHO[7:0] according to the High period of the SCL clock in Hs-mode.
6) Write the slave address to NTDTBP0.

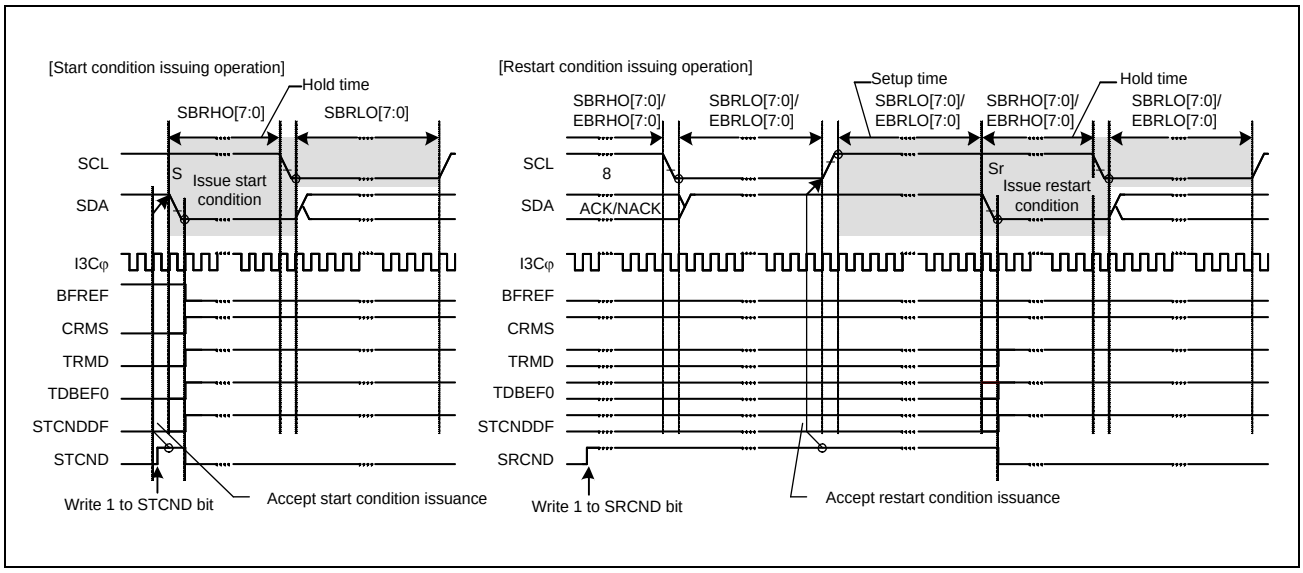


Figure 7.8-55 START condition / repeated START condition issue timing (STCND and SRCND bits)

Figure 7.8-56 shows the operation to issue a Repeated START condition after the master transmission. [Repeated START condition issuance after the master transmission]

- Initial setting. For details, see **7.8.4.3.1 Initial Setting Flow**.
- Read the BFREF flag in BCST to check that the bus is open, and then set the STCND bit in CNDCTL to 1 (START condition issuance request). Upon receiving the request, I3C issues a START condition. At the same time, the BFREF flag is automatically set to 0b and the STCNDDF flag in BST is automatically set to 1b and the STCND bit is automatically set to 0. At this time, if the START condition is detected and the internal levels for the SDA output state and the levels on the SDA line have matched while the STCND bit = 1b, I3C recognizes that issuing of the START condition as requested by the STCND bit has been successfully completed, and CRMS and TRMD bits in PRSST is automatically set to 1b, placing I3C in master transmit mode. The NTST.TDBEF0 flag is also automatically set to 1b in response to setting of the TRMD bit to 1b.
- Check that the NTST.TDBEF0 flag = 1b, and then write the value for transmission (the slave address and the R/W# bit) to NTDTBP0. Once the data for transmission are written to NTDTBP0, the TDBEF0 flag is automatically set to 0b, the data are transferred from NTDTBP0, and the TDBEF0 flag is again set to 1b. After the byte containing the slave address and R/W# bit has been transmitted, the value of the TRMD bit is automatically updated to select master transmit or master receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 0, I3C continues in master transmit mode. Since the BST.NACKDF flag being 1b at this time indicates that no slave device recognized the address or there was an error in communications, write 1b to CNDCTL.SPCND bit to issue a STOP condition. For data transmission with an address in the 10-bit format, start by writing 1111 0, the 2 higher-order bits of the slave address, and W to NTDTBP0 as the first address transmission. Then, as the second address transmission, write the 8 lower-order bits of the slave address to NTDTBP0.
- After confirming that the NTST.TDBEF0 flag = 1b, write the data for transmission to the NTDTBP0 register. I3C automatically holds the SCL line low until the data for transmission are ready, a Repeated START condition is issued or a STOP condition is issued.
- After all bytes of data for transmission have been written to the NTDTBP0 register, wait until the value of the BST.TENDF flag returns to 1b, and then, after check that the BST.STCNDDF flag = 1b, set the BST.STCNDDF flag to 0b.
- Set the SRCND bit in CNDCTL to 1b (Repeated START condition issuance request). Upon receiving the request, I3C issues a Repeated START condition.

- After check that the BST.STCNDDF flag = 1, write the value for transmission (the slave address and the R/W# bit) to NTDTBP0.

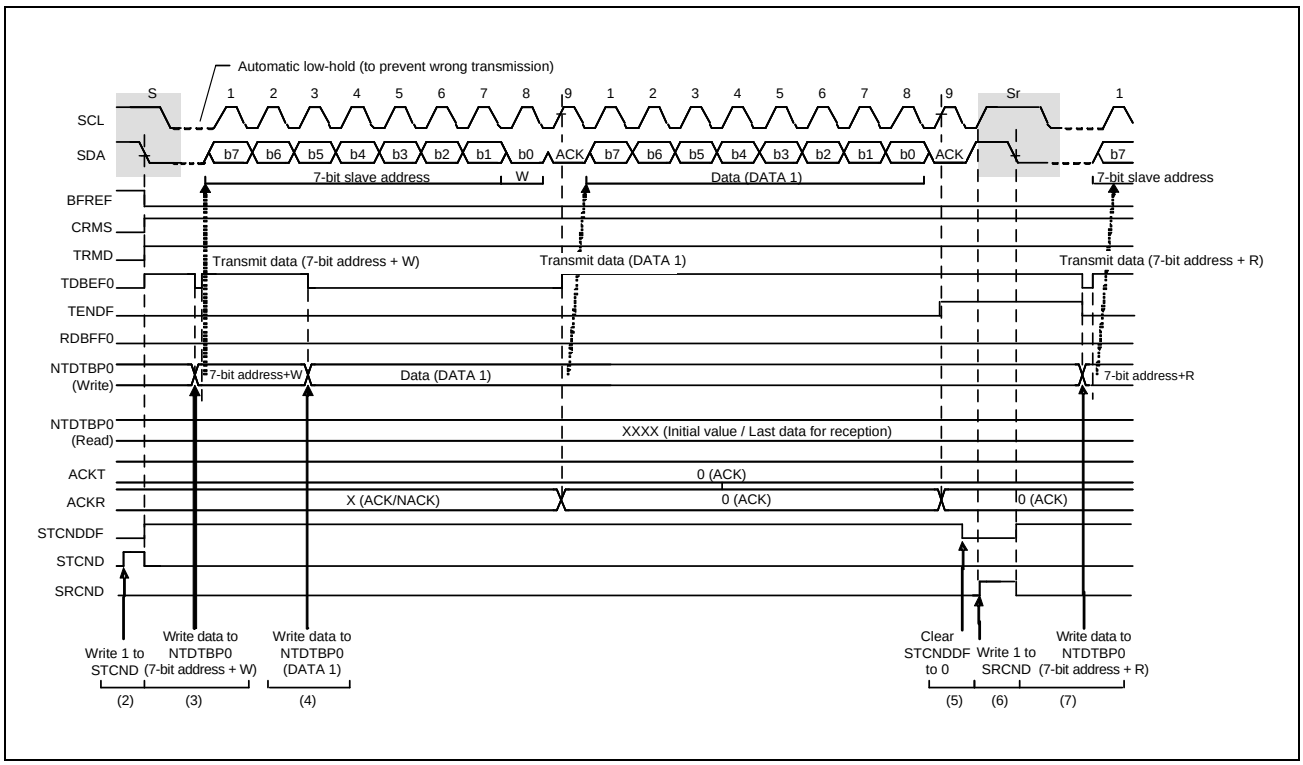


Figure 7.8-56 Repeated START condition issuance after the master transmission timing

(c) Issuing a STOP Condition

I3C issues a STOP condition when the SPCND bit in CNDCTL is set to 1.

When the SPCND bit is set to 1b, a STOP condition issuance request is made and I3C issues a STOP condition when the BCST.BFREF flag = 0b (bus busy state) and the PRSST.MST bit = 1b (master mode).

A STOP condition is issued in the following sequence.

[STOP condition issuance]

- Drive the SDA line low (high level to low level).
- Ensure the low-level period of SCL line set in STDBR.SBRLO[7:0] or EXTBR.EBRLO[7:0].
- Release the SCL line (low level to high level).
- Detect a high level of the SCL line and ensure the time set in STDBR.SBRHO[7:0] or EXTBR.EBRHO[7:0] and the STOP condition setup time.
- Release the SDA line (low level to high level).
- Ensure the time set in BFRECDT.FRECYC[8:0] and the bus free time.
- Set the BFREF flag to 1b (to release the bus mastership).

NOTE

To issue a STOP condition in Hs-mode, follow the steps below.

1) Wait until PRSTDBG.SCOLV=0b.
2) Set EXTBR.EBRHO[7:0] to satisfy the setup time for the STOP condition.
3) Set the CNDCTL.SPCND bit to 1b.
4) Wait until CNDCTL.SPCND=0b.
5) Set EXTBR.EBRHO [7: 0] according to the High period of the SCL clock in Hs-mode.

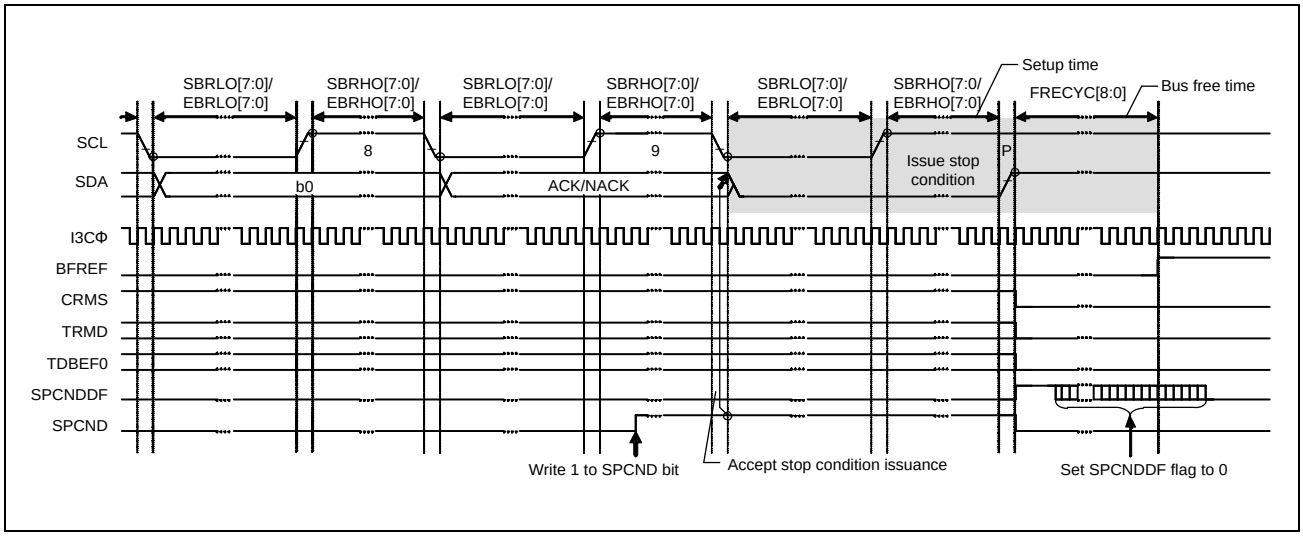


Figure 7.8-57 STOP condition issue timing (SPCND bit)

(4) Address Match Detection

I3C can set three unique slave addresses in addition to the general call address and host address, and also can set 7- bit or 10- bit slave addresses.

(a) Slave-Address Match Detection [I²C mode]

I3C can set three unique slave addresses, and has a slave address detection function for each unique slave address.

When the SVCTL.SVAEn bit (n = 0 to 2) is set to 1b, the slave addresses set in the SVDVADn register (n = 0 to 2) can be detected.

When I3C detects a match of the set slave address, the corresponding SVST.SVAFn flag (n = 0 to 2) is set to 1b at the rising edge of the ninth SCL clock cycle, and the NTST.RDBFF0 flag or the NTST.TDBEF0 flag is set to 1b by the following R/W# bit. This causes a receive data full interrupt (INT_ri3c_rx_n) or transmit data empty interrupt (INT_ri3c_tx_n) to be generated. The SVAFn flag is used to identify which slave address has been specified.

Figure 7.8-58 to **Figure 7.8-60** show the SVAFn flag set timing in three cases.

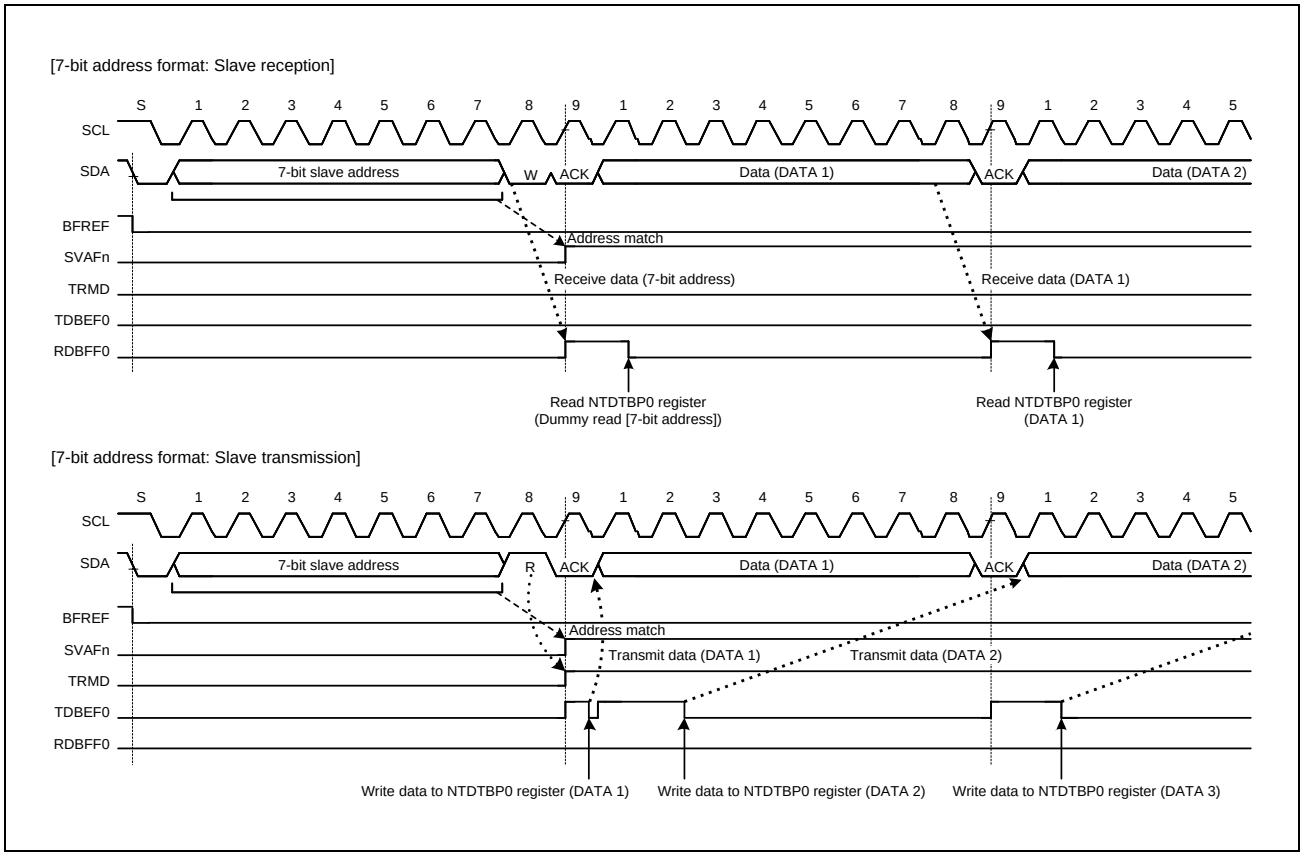


Figure 7.8-58 SVAFn flag set timing with 7-bit address format selected

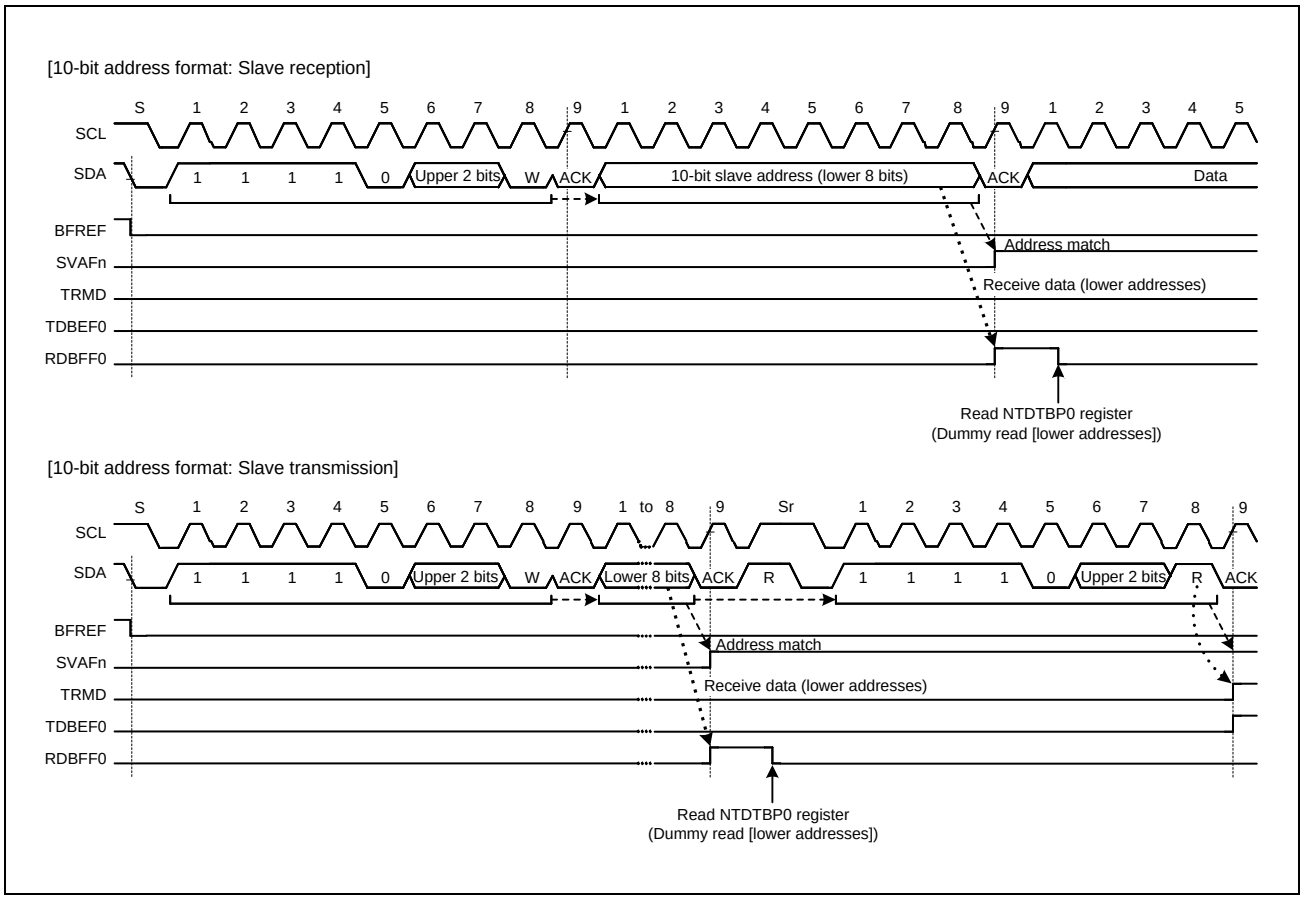


Figure 7.8-59 SVAFn flag set timing with 10-bit address format selected

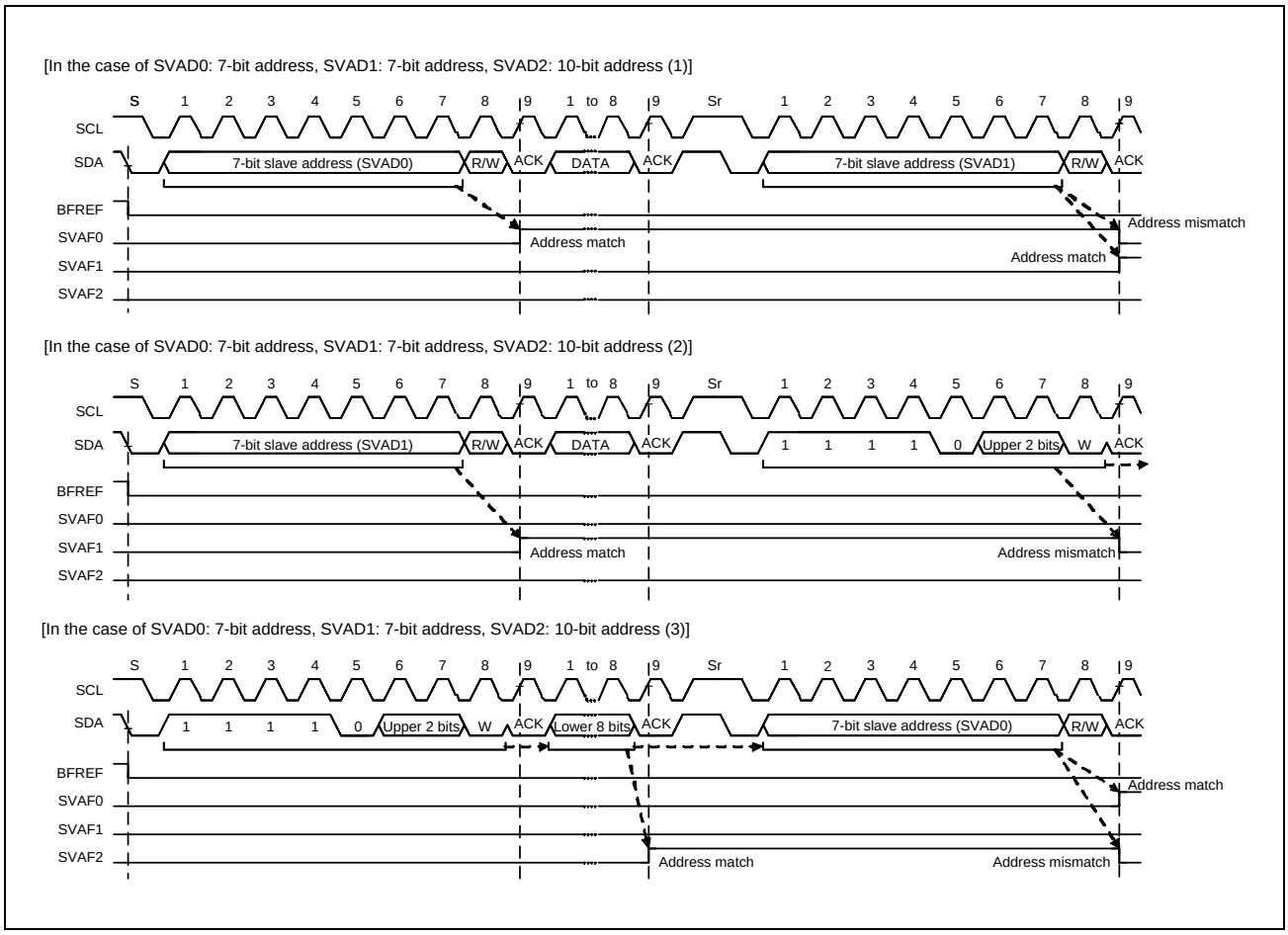


Figure 7.8-60 SVAFn flag set/clear timing with 7-bit/10-bit address formats mixed

(b) Detection of the General Call Address [I²C mode]

I3C has a facility for detecting the general call address (0000 000 + 0 (write)). This is enabled by setting the SVCTL.GCAE bit to 1b.

If the address received after a START or Repeated START condition is issued is 0000 000 + 1 (read) (start byte), I3C recognizes this as the address of a slave device with an all-zero address but not as the general call address.

When I3C detects the general call address, both the SVST.GCAF flag and the NTST.RDBFF0 flag are set to 1b on the rising edge of the ninth cycle of SCL clock. This leads to the generation of a receive data full interrupt (INT_ri3c_rx_n). The value of the GCAF flag can be confirmed to recognize that the general call address has been transmitted.

Operation after detection of the general call address is the same as normal slave receive operation.

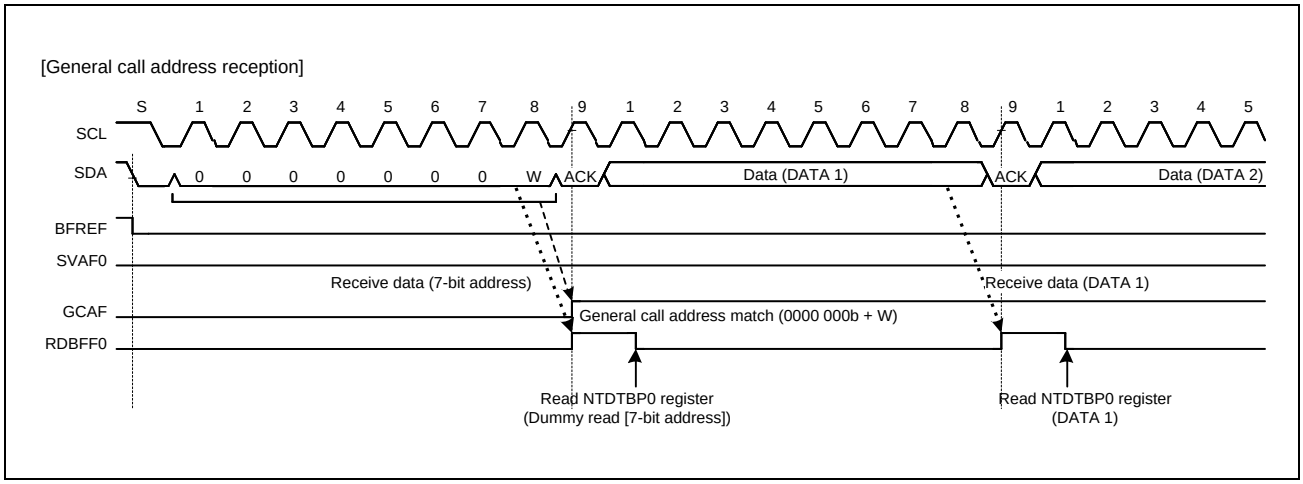


Figure 7.8-61 Timing of GCAF flag setting during reception of general call address

(c) Device-ID Address Detection [I²C mode]

I3C module has a facility for detecting device-ID addresses conformant with the I²C-bus specification (Rev.03). When I3C receives 1111 100b as the first byte after a START condition or Repeated START condition was issued with the SVCTL.DVIDE bit set to 1b, I3C recognizes the address as a device ID, sets the SVST.DVIDF flag to 1b on the rising edge of the ninth SCL clock cycle when the following R/W# bit = 0b, and then compares the second and subsequent bytes with its own slave address. If the address matches the value in the slave address register, I3C sets the corresponding SVST.SVAFn flag (n = 0 to 2) to 1b.

After that, when the first byte received after a START or Repeated START condition is issued matches the device ID address (1111 100b) again and the following R/W# bit = 1, I3C does not compare the second and subsequent bytes and sets the NTST.TDBEF0 flag to 1b.

In the device-ID address detection function, I3C sets the DVIDF flag to 0 if a match with I3C's own slave address is not obtained or a match with the device ID address is not obtained after a match with I3C's own slave address and the detection of a Repeated START condition. If the first byte after detection of a START or Repeated START condition matches the device ID address (1111 100b) and the R/W# bit = 0b, I3C sets the DVIDF flag to 1b and compares the second and subsequent bytes with I3C's slave address. If the R/W# bit = 1b, the DVIDF flag holds the previous value and I3C does not compare the second and subsequent bytes. Therefore, the reception of a device-ID address can be checked by reading the DVIDF flag after confirming that TDBEF0 flag = 1b.

Furthermore, prepare the device-ID fields (3 bytes: 12 bits indicating the manufacturer + 9 bits identifying the part + 3 bits indicating the revision) that must be sent to the host after reception of a continuous device-ID field as normal data for transmission. For details of the information that must be included in device-ID fields, contact NXP Semiconductors.

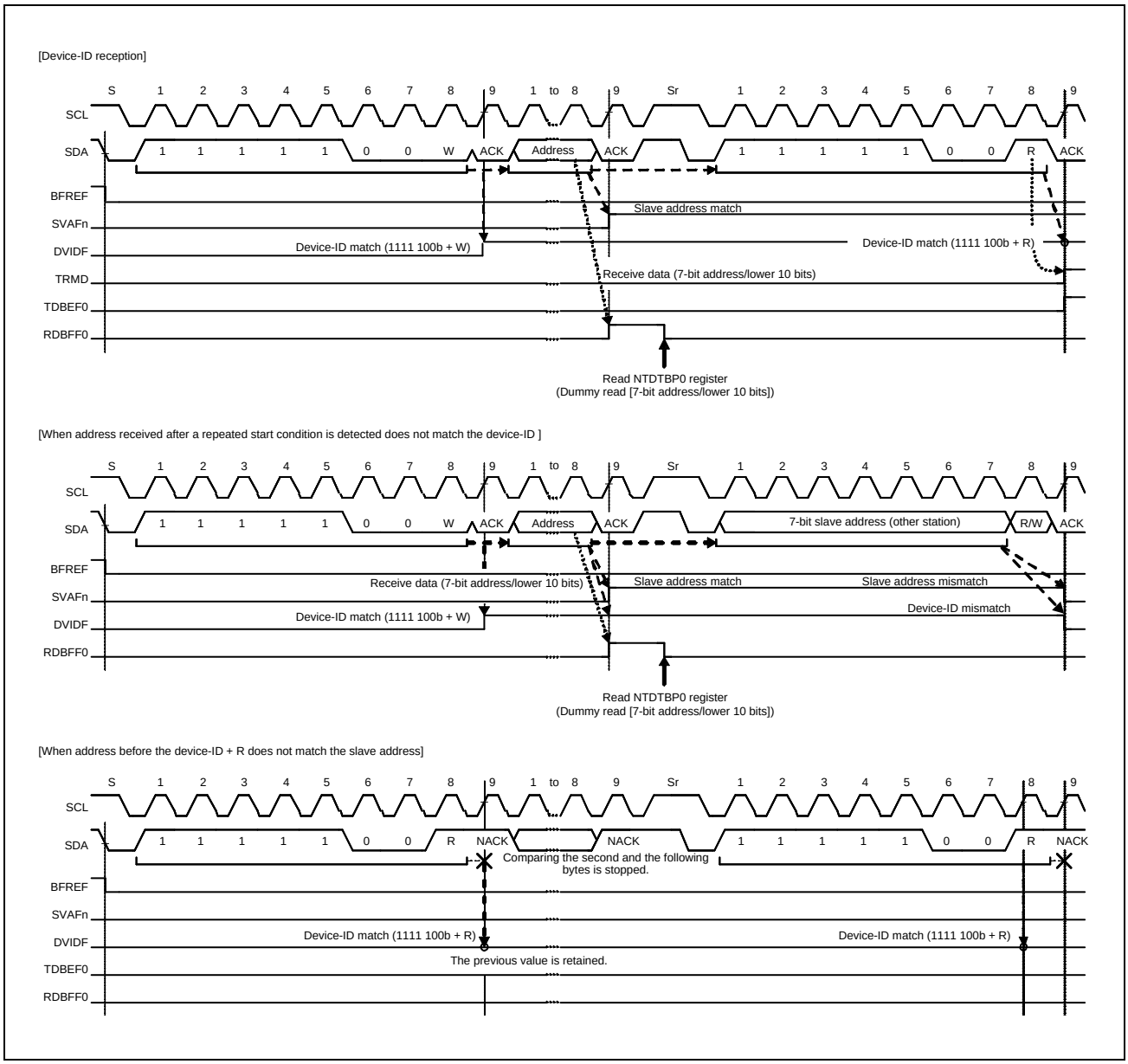


Figure 7.8-62 SVAFn/DVIDF flag set/clear timing during reception of device-ID

(d) Host Address Detection [I²C mode]

I3C has a function to detect the host address while the SMBus is operating. When the SVCTL.HOAE bit is set to 1b while the BFCTL.SMBS bit = 1b, I3C can detect the host address (0001 000) in slave receive mode (bits CRMS and TRMD in the PRSST register = 00b).

When I3C detects the host address, the SVST.HOAF flag is set to 1 at the rising edge of the ninth SCL clock cycle, and at the same time, the NTST.RDBFF0 flag is set to 1b when the R/W# bit = 0b (Wr bit). This causes a receive data full interrupt (INT_ri3c_rx_n) to be generated. The HOAF flag is used to recognize that the host address was sent from the smart battery or other devices.

If the bit following the host address (0001 000b) is an Rd bit (R/W# bit = 1b), I3C can also detect the host address. After the host address is detected, I3C operates in the same manner as normal slave operation.

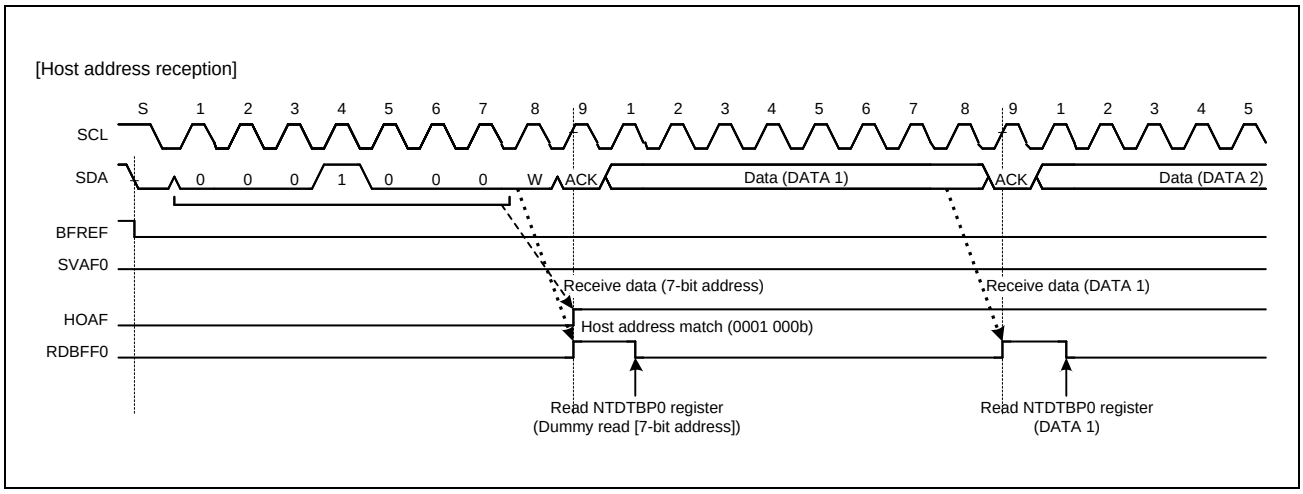


Figure 7.8-63 HOAF flag set timing during reception of host address

(e) Hs-mode master code Detection [I²C mode]

IC has a facility for detecting the Hs-mode master code (0000 1XXXb). When IIC receives the Hs-mode master code (0000 1XXXb) as the first byte after a START condition was issued with the SVCTL.HSMCE bit set to 1b, this module recognizes the address as the Hs-mode master code, sets the SVST.HSMCF flag to 1b on the rising edge of the ninth SCL clock cycle. The first byte after Repeated START after NACK response to Hs-mode master code is recognized as a slave address and compared with the slave address set by SVDVADn.SVAD[9:0] (n = 0 to 2). When IIC detects a match of the set slave address, the corresponding SVST.SVAFn flag (n = 0 to 2) is set to 1 at the rising edge of the ninth SCL clock cycle, and the NTST.RDBFF0 flag or the NTST.TDBEF0 flag is set to 1 by the following R/W# bit. This causes a receive data full interrupt (INT_ri3c_rx_n) or transmit data empty interrupt (INT_ri3c_tx_n) to be generated. The SVAFy flag is used to identify which slave address has been specified. The SVST.HSMCF flag is cleared to 0b when the STOP condition is detected.

NOTE

If the Hs-mode master code (0000 1XXXb) is received with the SVCTL.HSMCE bit set to 0b, other patterns are ignored until the STOP condition is detected.

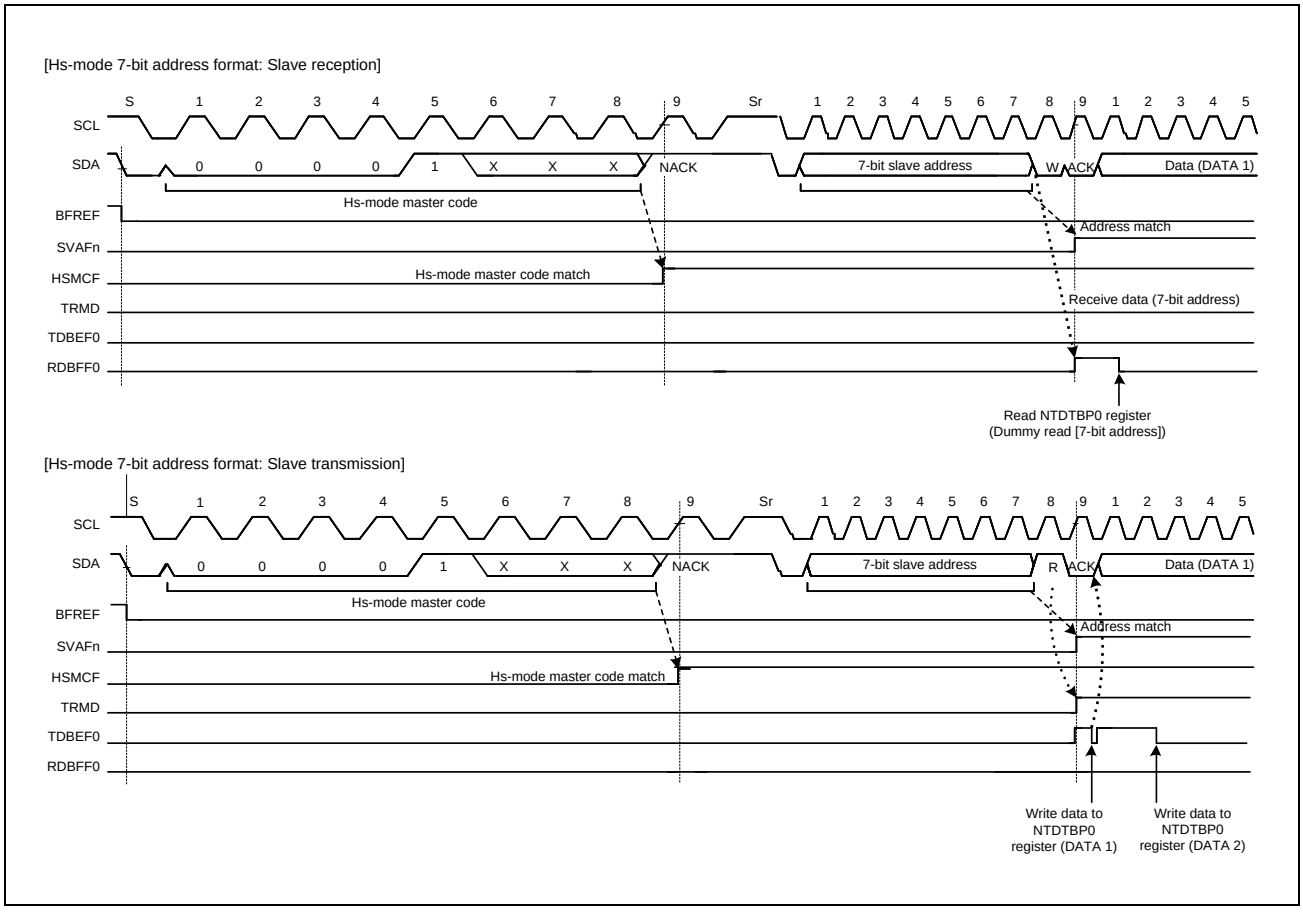


Figure 7.8-64 SVAFn/HSMCF Flag Set Timing during Reception of Hs-mode master code

(f) CCC detection function [I3C mode]

◆ **In case of Broadcast CCC**

1. It receives Broadcast Address (7Eh) and R/W# = 0 after START condition or Repeated START.
2. Respond to ACK.
3. Receive Common Command Code (CCC).
4. In accordance with the CCC, the following data is stored. (Storage destination: see **Table 7.8-12**)
5. Store the Receive Status Descriptor into the Receive Status Queue.

◆ **In case of Broadcast CCC (ENTDAA)**

1. It receives Broadcast Address (7Eh) and R/W# = 0b after START condition.
2. Respond to ACK.
3. Receive ENTDAA.
4. If receives Broadcast Address (7Eh) and R/W# = 1b after Repeated START.
5. When the Dynamic Address is not assigned, ACK response is done.
6. This Provisional ID (SDCTPIDH[31:0], SDCTPIDL[15:0]), BCR (SVDCT.TBCR[7:0]) and DCR (SVDCT.TDCR[7:0]) are transmitted.
7. When winning the arbitration in a transmission of the above Step 6, the dynamic address following that is received. When losing arbitration in a transmission of the above Step 6, processing of Step 6 is repeated from Step 4.
8. When parity of the Dynamic Address is valid, ACK response is done.
9. When parity of the Dynamic Address is invalid, NACK replies, and repeat the process from Steps 4 to 7.
10. SDATBAS0.SDDYAD[6:0] is renewed and the SVDVAD0.SDYADV bit is set to 1b.
11. Upon detecting the STOP condition, Store the Receive Status Descriptor into the Receive Status Queue.

◆ **In case of Direct Write CCC**

1. It receives Broadcast Address (7Eh) and R/W# = 0b after START condition or Repeated START.
2. Respond to ACK.
3. Receive Common Command Code (CCC).
4. Receive Dynamic Address and R/W# = 0b after Repeated START.
5. Compare the received Dynamic Address with the assigned Dynamic Address, and if it matches, I3C responds with ACK.
   If they do not match, it responds with NACK and waits for Repeated START or STOP.
6. In accordance with the CCC, the following data is stored. (Storage destination: see **Table 7.8-12**)
7. Store the Receive Status Descriptor into the Receive Status Queue.

◆ **In case of Direct Read CCC**

1. It receives Broadcast Address (7Eh) and R/W# = 1b after START condition or Repeated START.
2. Respond to ACK.
3. Receive Common Command Code (CCC).

4. Receive Dynamic Address and R/W# = 1b after Repeated START.
5. Compare the received Dynamic Address with the assigned Dynamic Address, and if it matches, I3C responds with ACK.
   If they do not match, it responds with NACK and waits for Repeated START or STOP.
6. Respond from SFR according to CCC. (Responding CCC: see **Table 7.8-12**)
7. Store the Receive Status Descriptor into the Receive Status Queue.

Table 7.8-12 Common command code operation

Command Code	CCC Type	Command Name	With Data	Auto Response	Storage
00h	Broadcast	ENEC	Yes	—	SFR
01h	Broadcast	DISEC	Yes	—	SFR
02h	Broadcast	ENTAS0	No	—	SFR
03h	Broadcast	ENTAS1	No	—	SFR
04h	Broadcast	ENTAS2	No	—	SFR
05h	Broadcast	ENTAS3	No	—	SFR
06h	Broadcast	RSTDAA	No	—	SFR
07h	Broadcast	ENTDAA	Yes	Yes	SFR
08h	Broadcast	DEFSLVS	Yes	—	FIFO
09h	Broadcast	SETMWL	Yes	—	SFR
0Ah	Broadcast	SETMRL	Yes	—	SFR
0Bh	Broadcast	ENTTM	Yes	—	SFR
28h	Broadcast	SETXTIME	Yes	—	FIFO
29h	Broadcast	SETAASA	No	—	SFR
80h	Direct Write	ENEC	Yes	—	SFR
81h	Direct Write	DISEC	Yes	—	SFR
82h	Direct Write	ENTAS0	No	—	SFR
83h	Direct Write	ENTAS1	No	—	SFR
84h	Direct Write	ENTAS2	No	—	SFR
85h	Direct Write	ENTAS3	No	—	SFR
86h	Direct Write	RSTDAA	No	—	SFR
87h	Direct Write	SETDASA	Yes	—	SFR
88h	Direct Write	SETNEWDA	Yes	—	SFR
89h	Direct Write	SETMWL	Yes	—	SFR
8Ah	Direct Write	SETMRL	Yes	—	SFR
8Bh	Direct Read	GETMWL	—	Yes	SFR
8Ch	Direct Read	GETMRL	—	Yes	SFR
8Dh	Direct Read	GETPID	—	Yes	SFR
8Eh	Direct Read	GETBCR	—	Yes	SFR
8Fh	Direct Read	GETDCR	—	Yes	SFR
90h	Direct Read	GETSTATUS	—	Yes	SFR
91h	Direct Read	GETACCMST	—	Yes	SFR
94h	Direct Read	GETMXDS	—	Yes	SFR
98h	Direct Write	SETXTIME	Yes	—	FIFO
99h	Direct Read	GETXTIME	—	Yes	SFR

(5) Arbitration-Lost Detection [I²C mode]

In addition to the normal arbitration-lost detection function defined by the I²C-bus specification, the I3C has functions to prevent double-issue of a start condition, to detect arbitration-lost during transmission of NACK, and to detect arbitration- lost in slave transmit mode.

(a) Master Arbitration-Lost Detection (MALE Bit)

The I3C drives the SDA line low to issue a start condition. However, if the SDA line has already been driven low by another master device issuing a START condition, this module causes arbitration to be lost, so priority is given to transfer by the other master device. Similarly, if the CNDCTL.STCND bit is set to 1 while the BCST.BFREF flag is 0 (bus busy state), arbitration is lost, so priority is given to transfer by the other master device. No start condition is issued in this case.

When a start condition is issued successfully, if the data for transmission including the address bits (the internal SDA output level) and the level on the SDA line do not match (the high output as the internal SDA output, that is, the SDA30 pin is in the high-impedance state) and the low level is detected on the SDA line, the I3C loses in arbitration.

I3C detects master arbitration-lost when the following conditions are met while the BSTE.ALE bit = 1b and the BFCTL.MALE bit = 1b (master arbitration-lost detection enabled).

If arbitration of mastership is lost, I3C immediately enters slave receive mode.

If a slave address (including the general call address) matches its own address at this time, I3C continues in slave operation.

[Conditions for master arbitration-lost]

- Non-matching of the internal level for output on SDA and the level on the SDA line after a START condition was issued by setting the CNDCTL.STCND bit to 1b while the BCST.BFREF flag was set to 1b (erroneous issuing of a START condition)
- Setting of the CNDCTL.STCND bit to 1b (START condition double-issue error) while the BFREF flag is set to 0

Remark I3C does not issue a START condition.

- When the transmit data excluding acknowledge (internal SDA output level) does not match the level on the SDA line in master transmit mode (bits CRMS and TRMD in the PRSST register = 11)

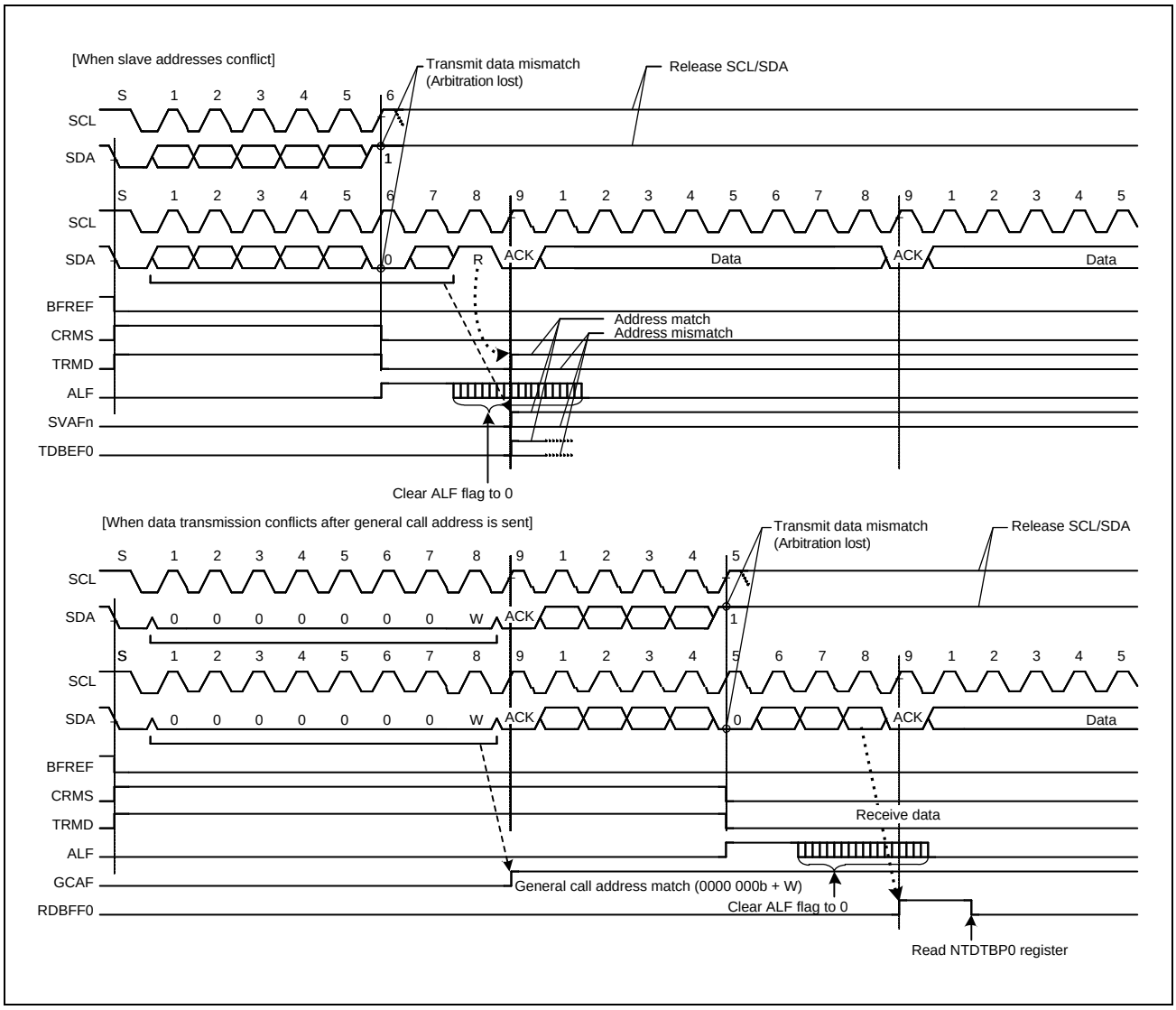


Figure 7.8-65 Examples of master arbitration-lost detection (MALE = 1b)

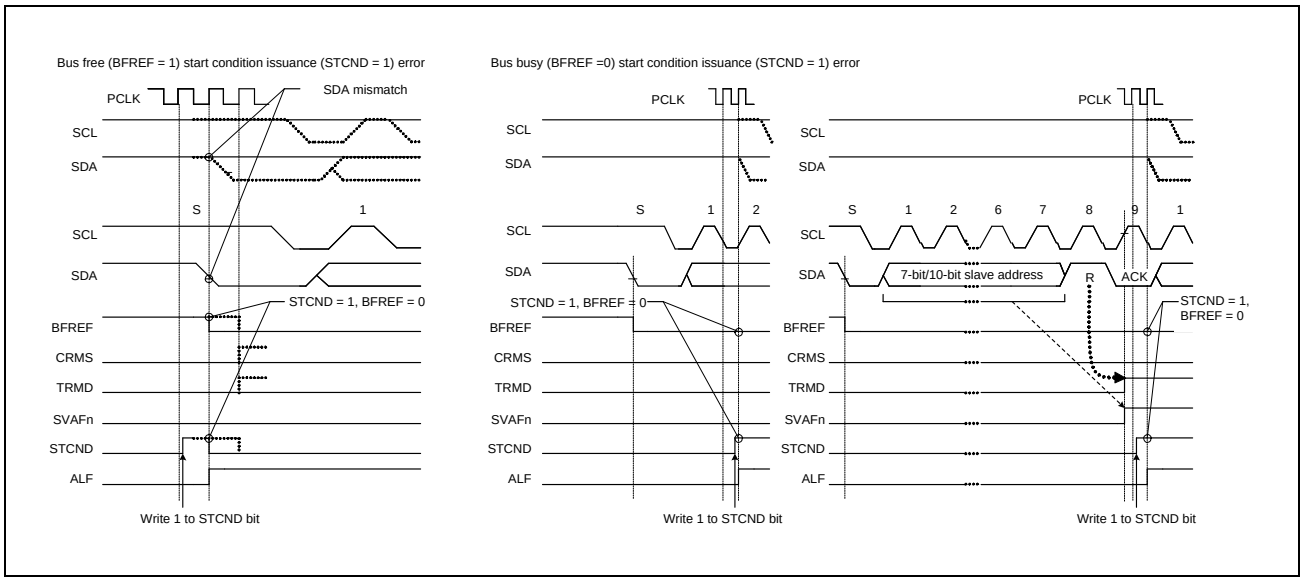


Figure 7.8-66 Arbitration-lost detection when a START condition is issued (MALE = 1b)

(b) Arbitration-Lost Detection during NACK Transmission (NALE Bit)

The I3C has a function to cause arbitration to be lost if the internal SDA output level does not match the level on the SDA line (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state) and the low level is detected on the SDA line during transmission of NACK in receive mode. Arbitration is lost due to a conflict of NACK transmission and ACK transmission when two or more master devices receive data from the same slave device simultaneously in a multi-master system. Such conflict occurs when multiple master devices send/receive the same information through a single slave device.

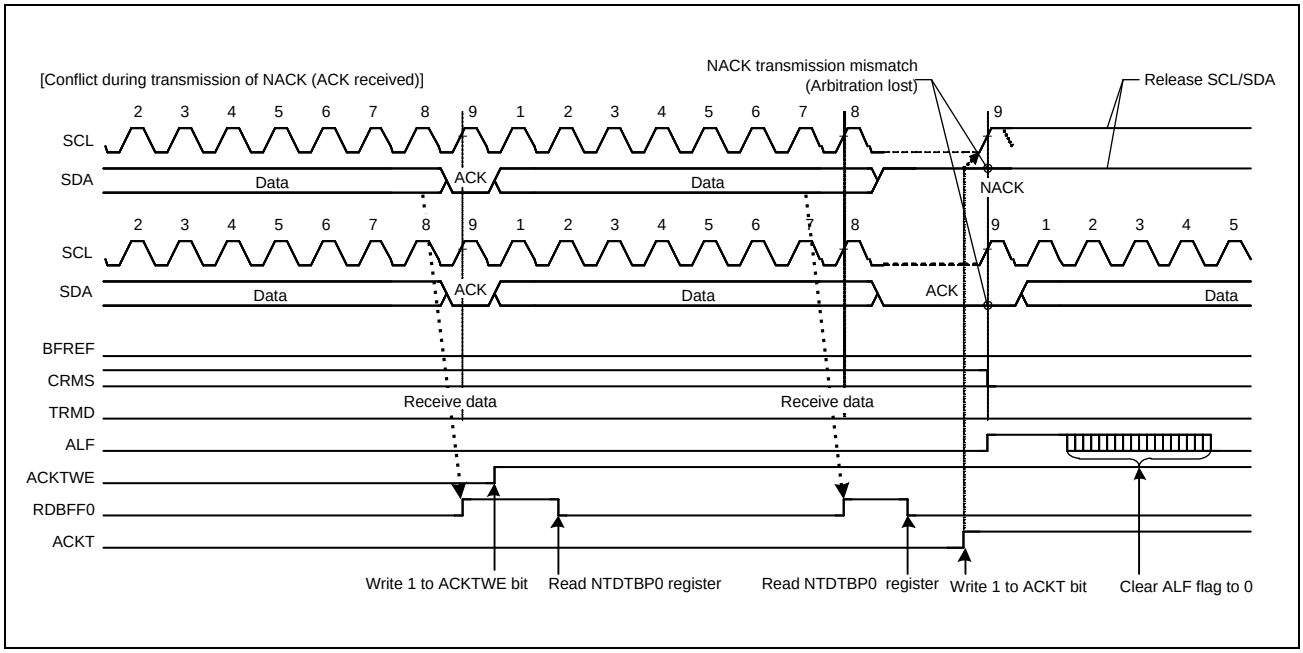


Figure 7.8-67 Example of arbitration-lost detection during transmission of NACK (NALE = 1b)

The following section explains arbitration-lost detection using an example where two master devices (master A and master B) and a single slave device are connected through the bus. In this example, master A receives 2 bytes of data from the slave device, and master B receives 4 bytes of data from the slave device.

If master A and master B access the slave device simultaneously, because the slave address is identical, arbitration is not lost in both master A and master B during access to the slave device. Therefore, both master A and master B recognize that they have obtained the bus mastership and operate as such. In this example, master A sends NACK when it has received 2 final bytes of data from the slave device. Meanwhile, master B sends ACK because it has not received the necessary 4 bytes of data. At this time, the NACK transmission from master A and the ACK transmission from master B conflict. In general, if a conflict like this occurs, master A cannot detect ACK transmitted by master B and issues a stop condition. Therefore, the issuance of the stop condition conflicts with the SCL clock output of master B, which disturbs communication.

When this module receives ACK during transmission of NACK, it detects a defeat in conflict with other master devices and causes arbitration to be lost.

If arbitration is lost during transmission of NACK, this module is immediately released from the slave-matched state and enters slave receive mode. This prevents a stop condition from being issued, preventing a communication failure on the bus.

Similarly, in the ARP command processing of SMBus, the function to detect loss of arbitration during transmission of NACK is also available for eliminating the extra clock cycle processing (such as FFh transmission processing)

necessary if the UDID (Unique Device Identifier) of assign address does not match in the Get UDID (general) processing after the Assign Address command.

The I3C detects arbitration-lost during transmission of NACK when the following condition is met while the BSTE.ALE bit = 1b and the BFCTL.NALE bit = 1b (arbitration-lost detection during NACK transmission enabled).

[Condition for arbitration-lost during NACK transmission]

- When the internal SDA output level does not match the SDA line (ACK is received) during transmission of NACK (ACKCTL.ACKT bit = 1b)

(c) Slave Arbitration-Lost Detection (SALE Bit)

The I3C has a function to cause arbitration to be lost if the data for transmission (the internal SDA output level) and the level on the SDA line do not match (the high output as the internal SDA output, that is, the SDA pin is in the high-impedance state and the low level is detected on the SDA line in slave transmit mode). This arbitration-lost detection function is mainly used when transmitting a UDID (Unique Device Identifier) over an SMBus.

If arbitration is lost during transmission of DATA, this module is immediately released from the slave-matched state and enters slave receive mode. This function can detect conflicts of data during transmission of UDIDs over an SMBus and eliminate subsequent redundant processing (processing for the transmission of FFh).

The I3C detects slave arbitration-lost when the following condition is met while the BSTE.ALE bit = 1b and the BFCTL.SALE bit = 1b (slave arbitration-lost detection enabled).

[Condition for slave arbitration-lost]

- When the transmit data excluding acknowledge (internal SDA output level) does not match the level on the SDA line in slave transmit mode (bits CRMS and TRMD in the PRSST register = 01b).

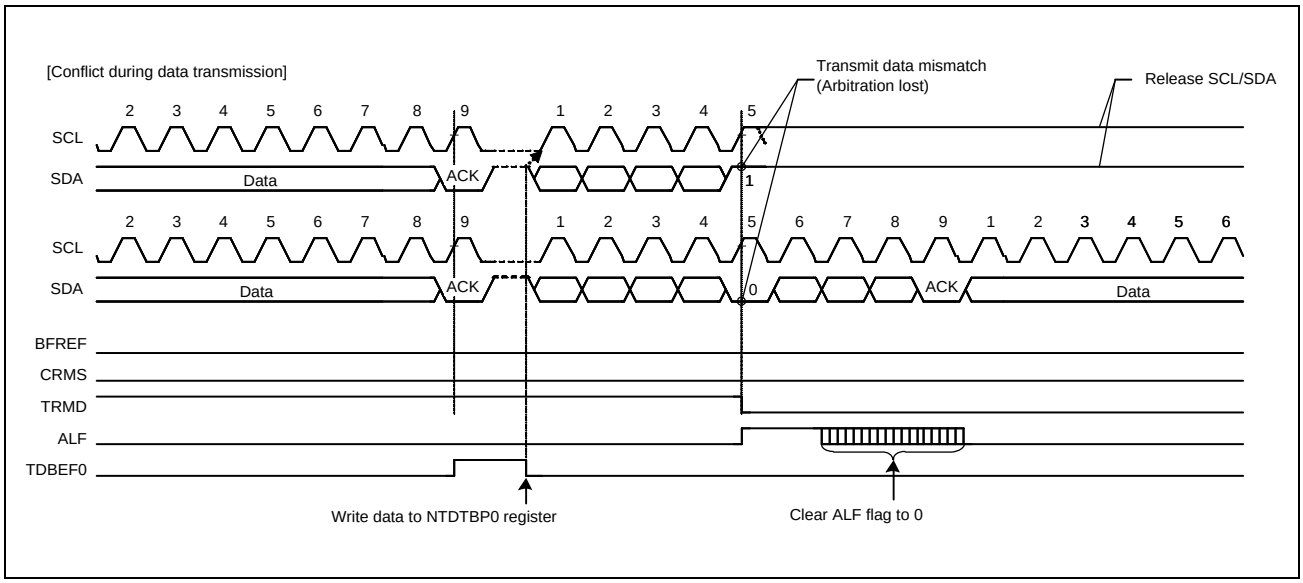


Figure 7.8-68 Example of slave arbitration-lost detection (SALE = 1)

(6) Clock Stretching [I²C mode]

(a) Function to Prevent Wrong Transmission of Transmit Data

When data have not been written to the I²C bus transmit data register (NTDTBP0) with I3C in transmission mode (PRSST.TRMD = 1b), the SCL line is automatically held at the low level over the intervals shown below. This low-hold period is extended until data for transmission have been written, which prevents the unintended transmission of erroneous data.

Master transmit mode

- Low-level interval after a START condition or Repeated START condition is issued
- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

Slave transmit mode

- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

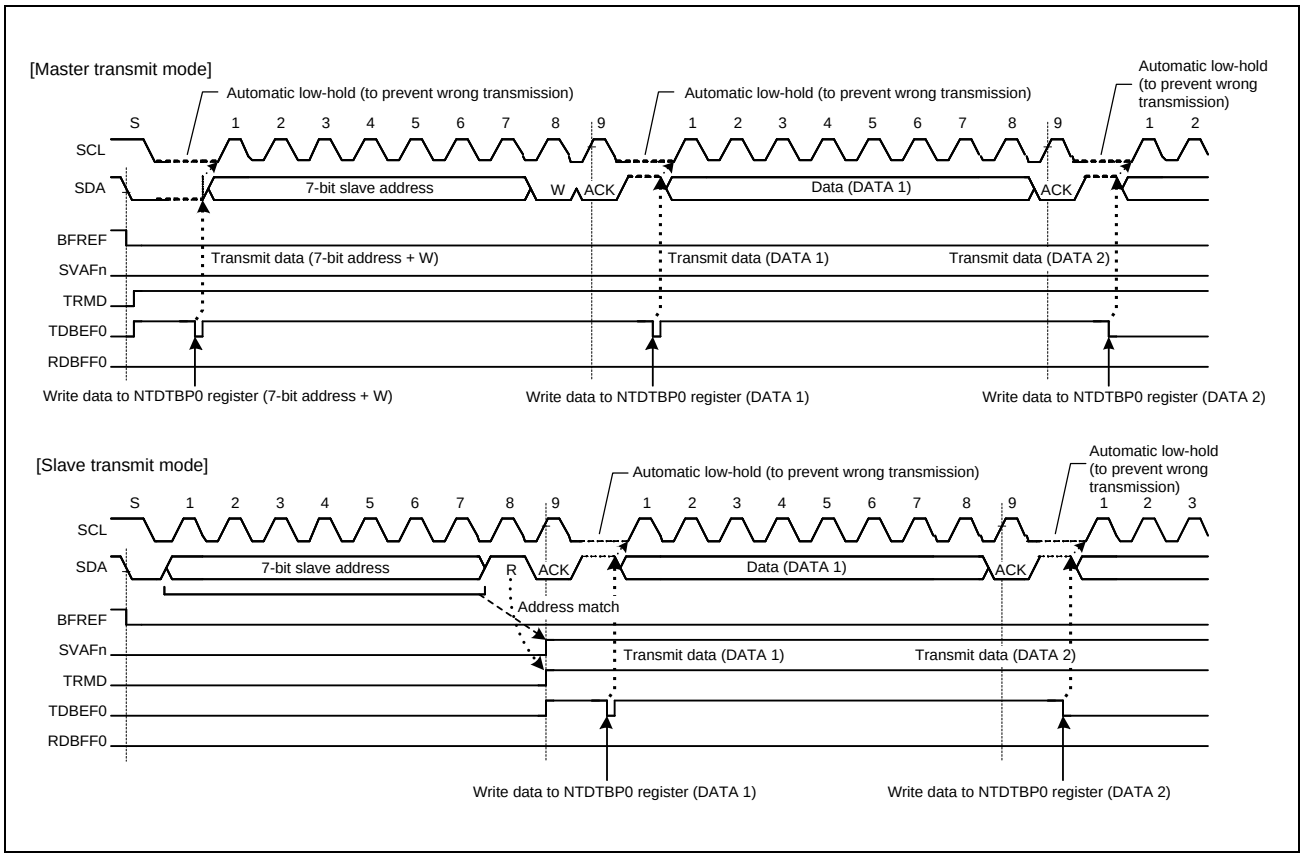


Figure 7.8-69 Automatic low-hold operation in transmit mode

(b) NACK Reception Transfer Abort Function

I3C has a function to abort transfer operation when NACK is received in transmit mode (PRSST.TRMD = 1b). This function is enabled when the BSTE.NACKDE bit is set to 1b (transfer abort enabled). If the next transmit data has already been written (NTST.TDBEF0 = 0b) when NACK is received, next data transmission at the falling edge of the ninth SCL clock cycle is automatically aborted. This prevents the SDA line output level from being held low when the MSB of the next transmit data is 0b.

If the transfer operation is aborted by this function (BST.NACKDF = 1b), transmit operation and receive operation are discontinued. To restore transmit/receive operation, be sure to set the NACKDF flag to 0b. In master transmit mode, restore operation using either of the methods below:

- After issuing a Repeated START condition, set the NACKDF flag to 0b
- After issuing a STOP condition, set the NACKDF flag to 0b and then issue a START condition

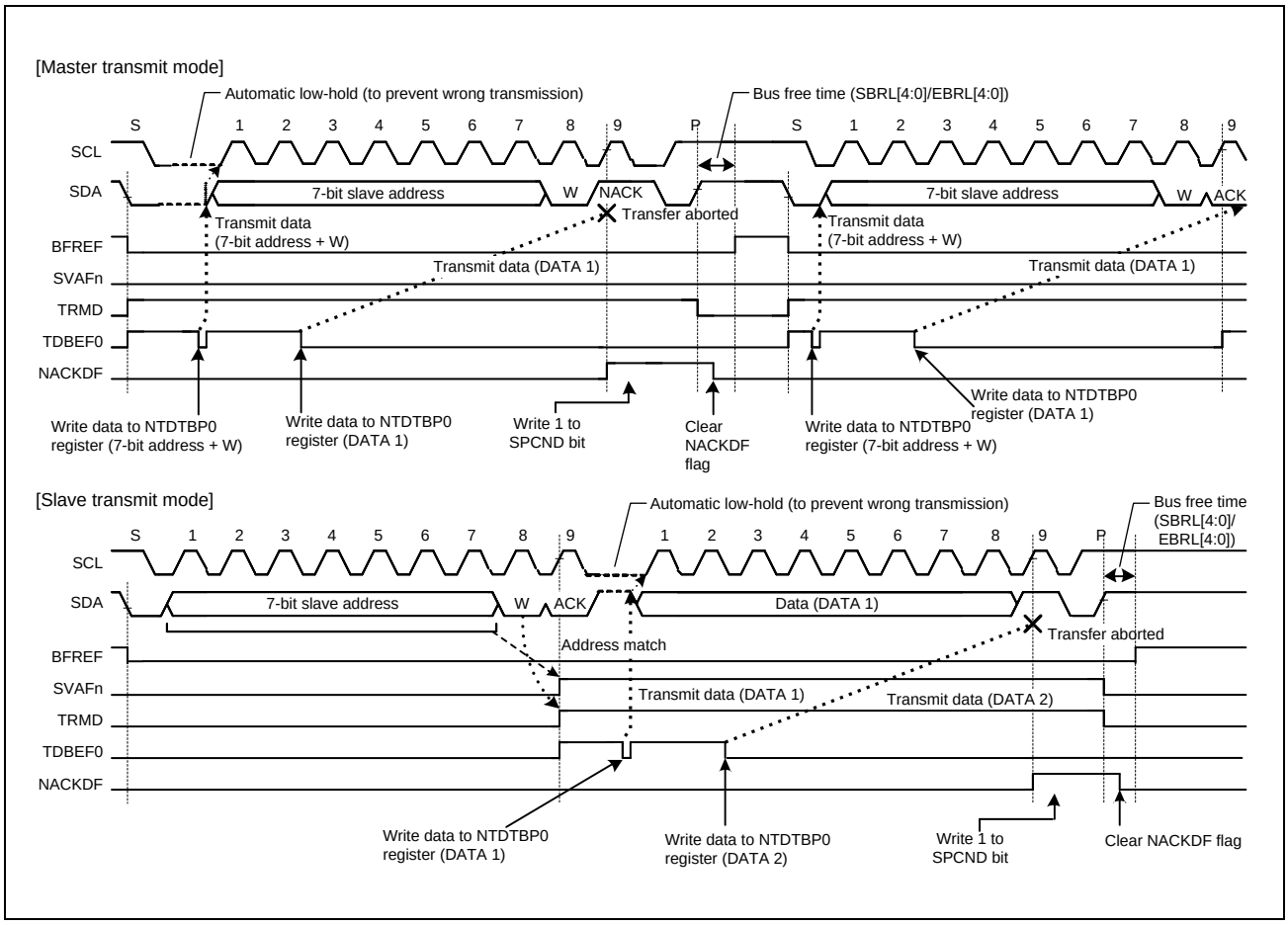


Figure 7.8-70 Abort of data transfer when NACK is received (NACKE = 1b)

(b-1) Function to Prevent Failure to Receive Data

If response processing is delayed when receive data (NTDTBP0) read is delayed for a period of one transfer frame or more with receive data full (NTST.RDBFF0 = 1b) in receive mode (PRSST.TRMD = 0b), I3C holds the SCL line low automatically immediately before the next data is received to prevent failure to receive data.

This function to prevent failure to receive data using the automatic low-hold function is also enabled even if the read processing of the final receive data is delayed and, in the meantime, I3C's own slave address or another slave address is received after a STOP condition is issued.

Sections in which the SCL line is held low can be selected with a combination of the RWE and ACKTWE bits in SCSTRCTL.

(b-2) 1-Byte Receive Operation and Automatic Low-Hold Function Using the RWE Bit

When the SCSTRCTL.RWE bit is set to 1b, I3C performs 1-byte receive operation using the RWE bit function.

Furthermore, when the SCSTRCTL.ACKTWE bit = 0b, I3C automatically sends the ACKCTL.ACKT bit value for the acknowledge bit in the period from the falling edge of the eighth SCL clock cycle to the falling edge of the ninth SCL clock cycle, and automatically holds the SCL line low at the falling edge of the ninth SCL clock cycle using the RWE bit function. This low-hold is released by reading data from NTDTBP0, which enables bytewise receive operation.

The RWE bit function is enabled for receive frames after a match with I3C's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

(b-3) 1-Byte Receive Operation (ACK/NACK Transmission Control) and Automatic Low-Hold Function Using the ACKTWE Bit

When the SCSTRCTL.ACKTWE bit is set to 1b, I3C performs 1-byte receive operation using the ACKTWE bit function.

When the ACKTWE bit is set to 1b, the NTST.RDBFF0 flag (receive data full) is set to 1b at the rising edge of the eighth SCL clock cycle, and the SCL line is automatically held low at the falling edge of the eighth SCL clock cycle. This low-hold is released by writing a value to the ACKCTL.ACKT bit, but cannot be released by reading data from NTDTBP0, which enables receive operation by the ACK/NACK transmission control according to the data received in byte units.

The ACKTWE bit function is enabled for receive frames after a match with I3C's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

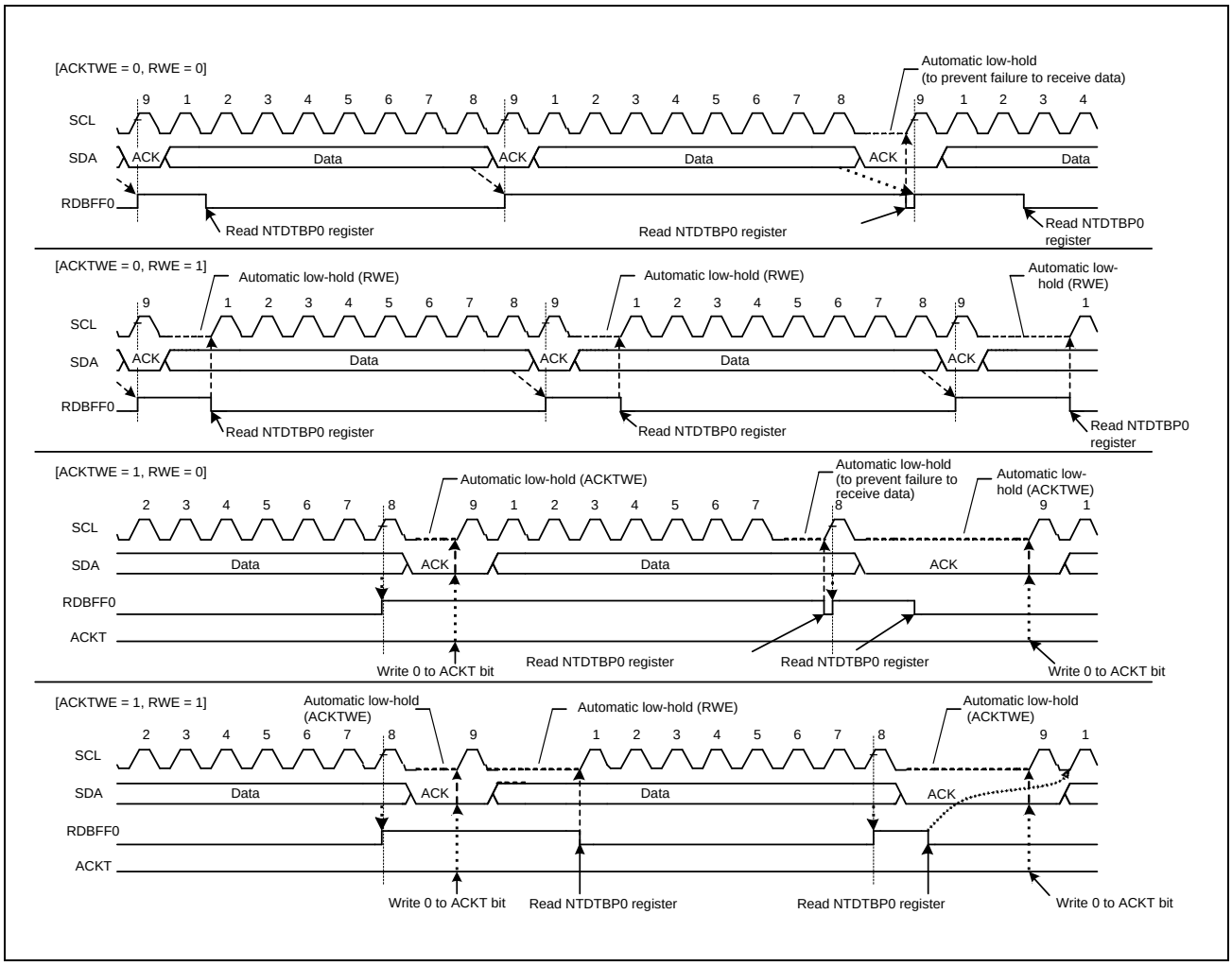


Figure 7.8-71 Automatic low-hold operation in receive mode (using ACKTWE and RWE bits)

(7) Clock Stalling [I3C mode]

I3C has the function of stalling the SCL during the SCL Low period.

The SCL stall control is described in the table below.

Table 7.8-13 I3C clock stalling

Clock stalling condition	Clock stalling control	Clock stalling period
I3C Transfer, ACK/NACK Phase	SCSTLCTL.ACKPE bit setting	During the count period of SCSTLCTL.STLCYC [15: 0] value
	Transmit Data FIFO Empty	Until data is written to the Transmit FIFO
	Receive Data FIFO Full	Until data is read from the Receive FIFO
I3C Write Data Transfer, Parity Bit	SCSTLCTL.PARPE bit setting	During the count period of SCSTLCTL.STLCYC [15: 0] value
	Transmit Data FIFO Empty	Until data is written to the Transmit FIFO
I3C Read Transfer, Transition Bit	Receive Data FIFO Full	Until data is read from the Receive FIFO
Assigned Address Phase	SCSTLCTL.AAPE bit setting	During the count period of SCSTLCTL.STLCYC [15: 0] value

The following figure shows the stalling timing of each Condition.

(a) I3C Transfer, ACK/NACK Phase

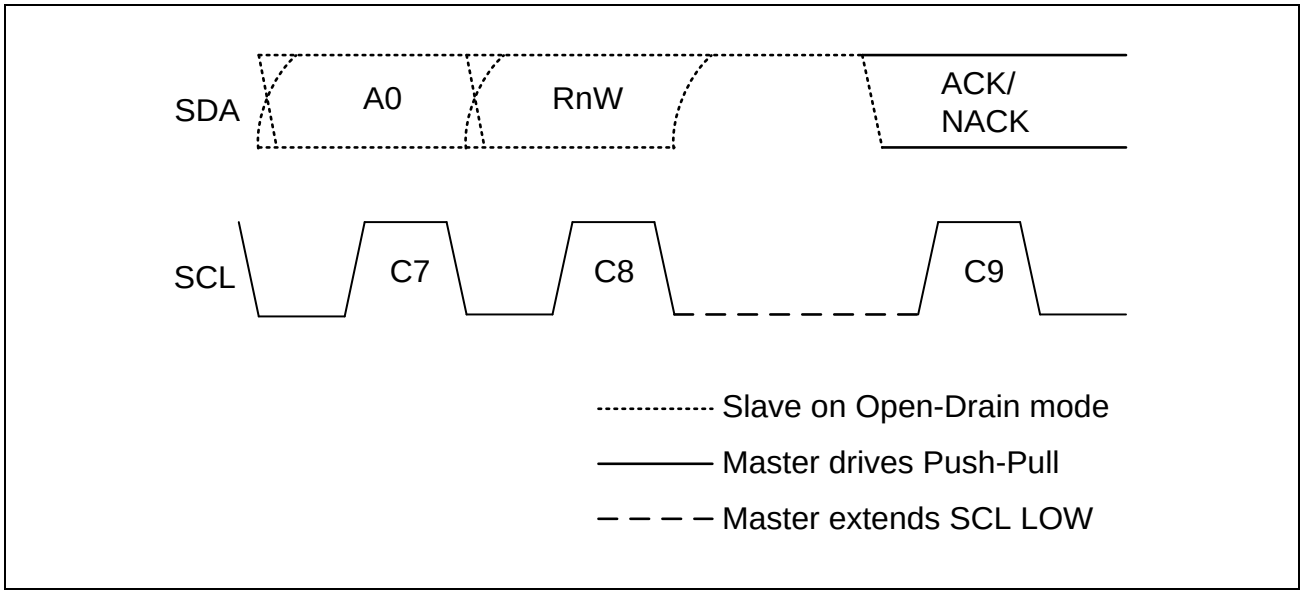


Figure 7.8-72 Master clock stalling in ACK phase

(b) I3C Write Data Transfer, Parity Bit

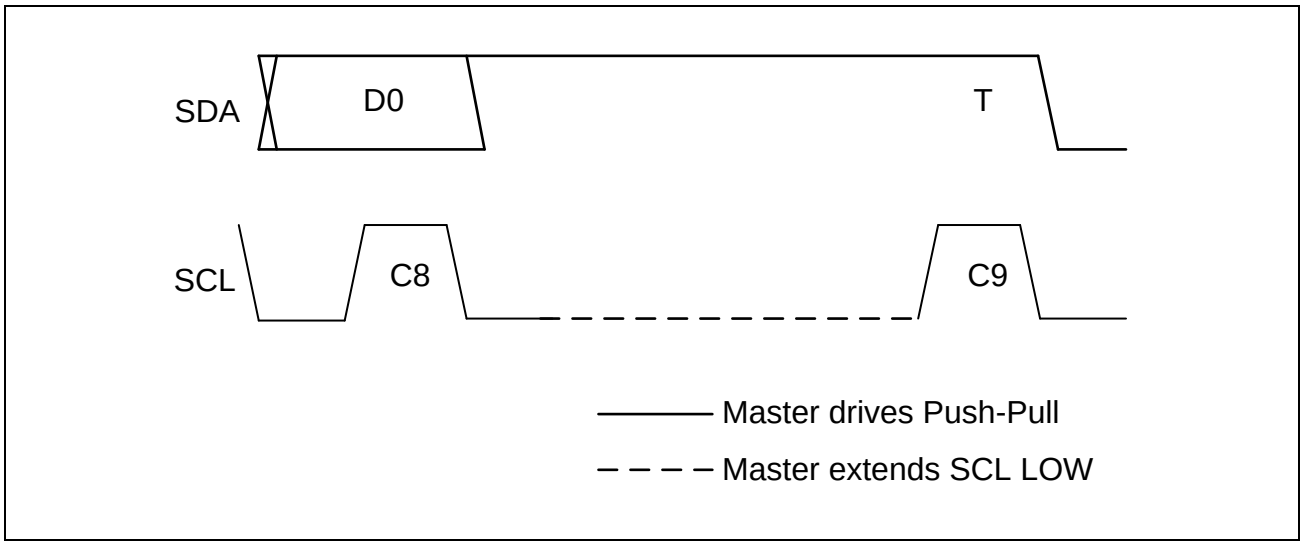


Figure 7.8-73 Master clock stalling in write parity bit

(c) I3C Read Transfer, Transition Bit

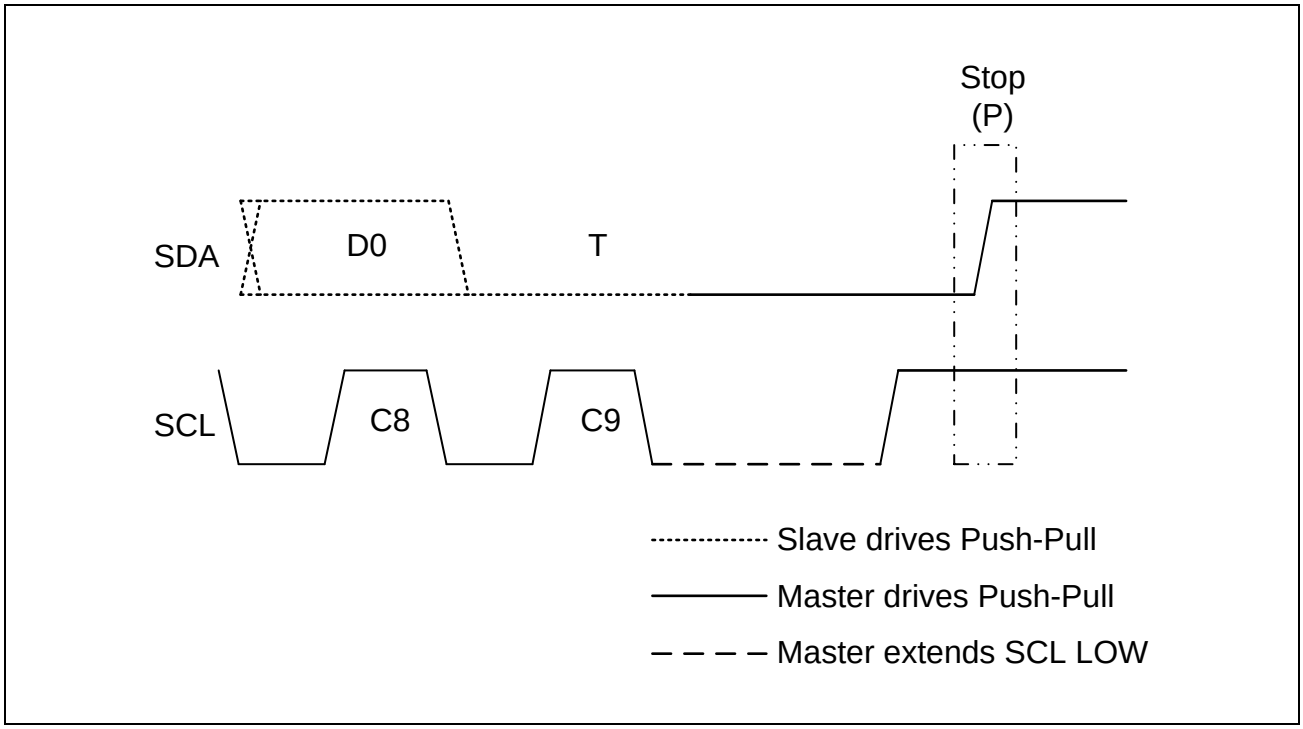


Figure 7.8-74 Master clock stalling in T-bit before next read data

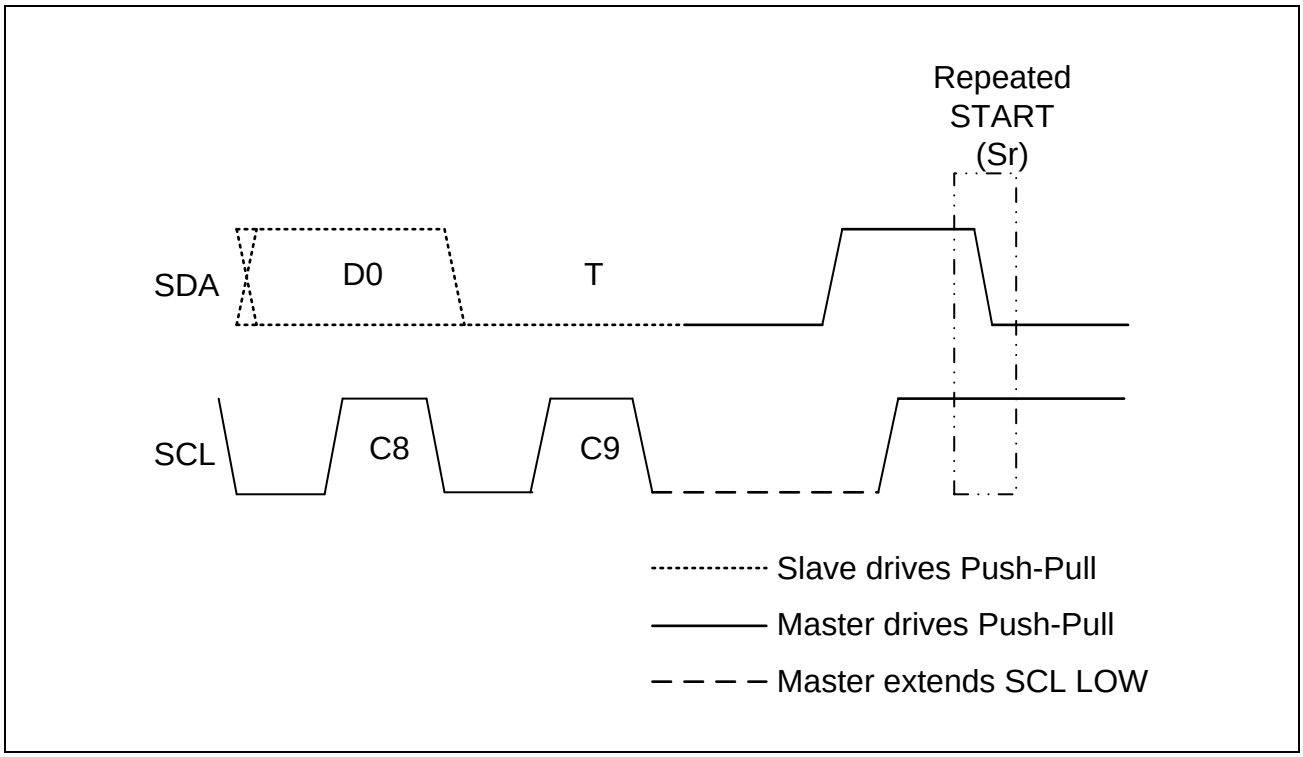


Figure 7.8-75 Master clock stalling in T-bit before STOP

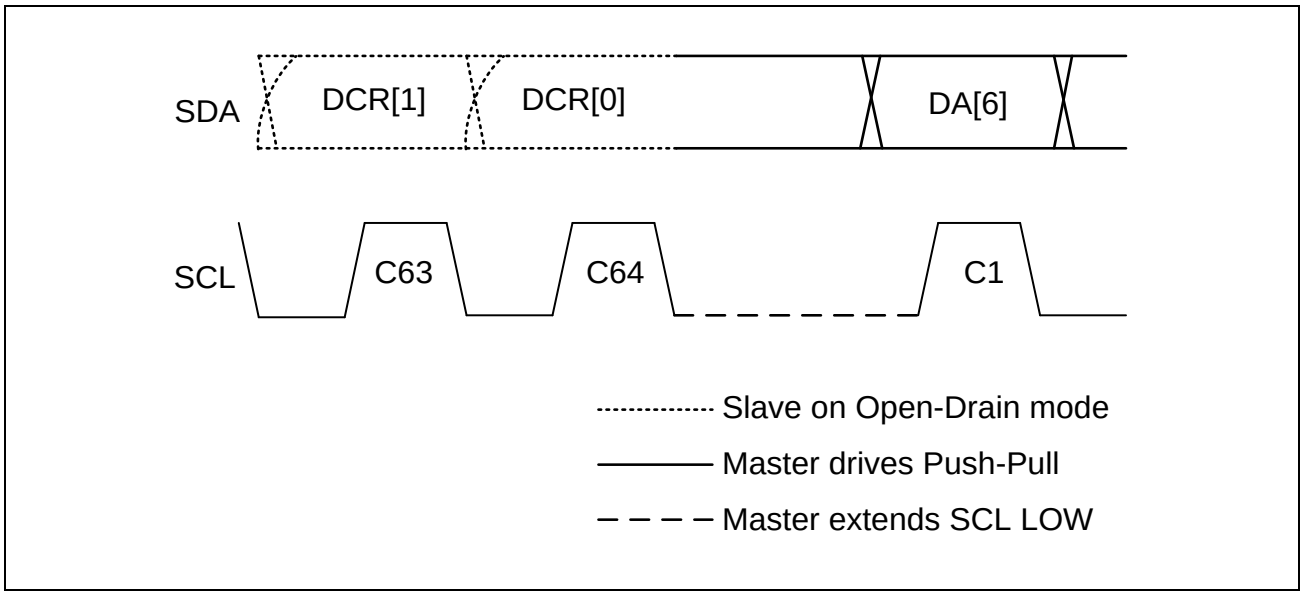


Figure 7.8-76 Master clock stalling in low T-bit before repeated START

(d) Dynamic Address Assignment, First Bit of Assigned Address

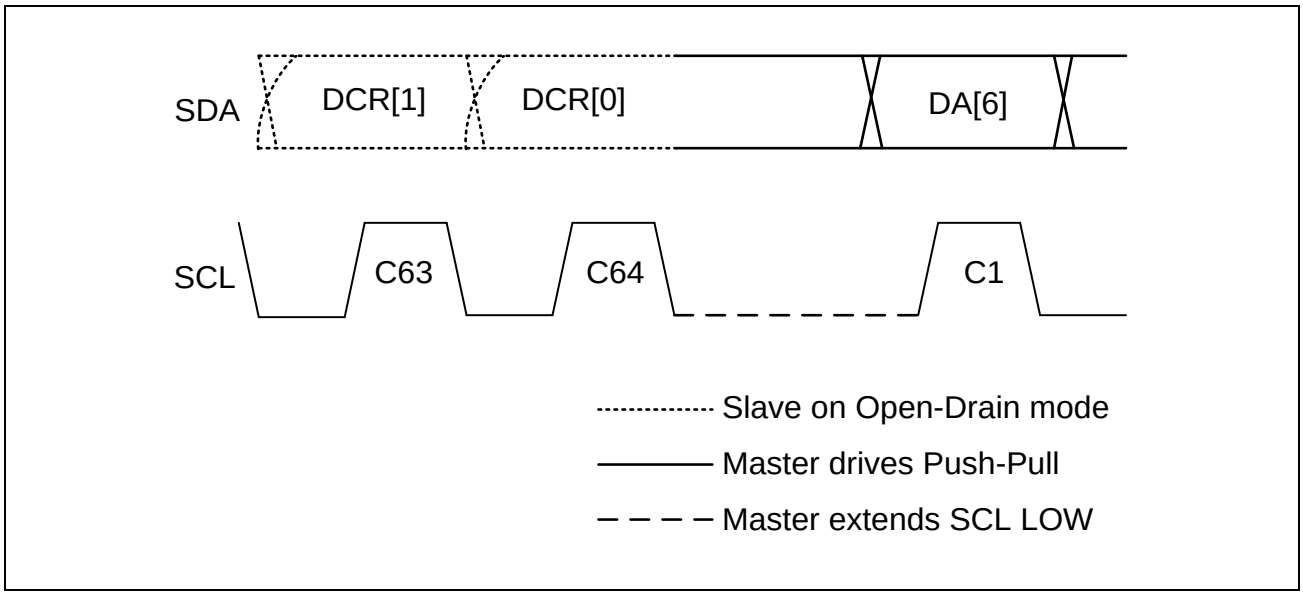


Figure 7.8-77 Master clock stalling in dynamic address first bit

(8) In-Band Interrupt [I3C mode]

I3C detects In-Band Interrupt in the arbitrated Address Header following a START condition (but not following a Repeated START). If START Request (SDA Low Drive) is issued from Slave Device, I3C drives SCL low and completes START condition. After that, it supplies SCL and receives In-Band Interrupt Request.

The In-Band Interrupt to be detected is classified into the following three types.

- Slave Interrupt Request
- Mastership Request
- Hot-Join Event

The operation when detecting each In-Band Interrupt is described below.

(a) Slave Interrupt Request

1. Detect Slave Address with RnW bit High in Address Header.
2. Compare the detected Slave Address with the DVDYAD[7:0] in each DAT (DATBASm register).
3. When it does not match DAT.DVDYAD[7:0]:
   Responds NACK, then issues the STOP condition. When it matches the DAT.DVDYAD[7:0] bits and the DAT.DVSIRRJ bit = 1b:
   It operates in the following order:
   (a) Responds NACK.
   (b) Issues Repeated START condition, then automatically issues Direct DISEC CCC to the detected Slave.
   (c) Issues the STOP condition.
   When it matches the DAT.DVDYAD[7:0] bits and the DAT.DVSIRRJ bit = 0b:
   Responds ACK.
4. When DAT.DVIBIPL = 0b:
   Issues the STOP condition.
   When DAT.DVIBIPL = 1b:
   Drives the SCL to receive the IBI Data from the Slave following the ACK response and receives IBI Data.
   It stores the received IBI Data into the IBI Data Queue.
   Each time IBI Data of the size set by the NQTHCTL.IBIDSSZ[7:0] bits is received, the IBI Status Descriptor is stored in the IBI Status Queue.
5. After detection of Low of T-bit following IBI Data, issues STOP condition.
6. After issues of STOP condition NACK response:
   – If IBINCTL.NRSIRCTL = 0b, the IBI Status Descriptor is not stored into the IBI Status Queue.
   – If IBINCTL.NRSIRCTL = 1b, the IBI Status Descriptor is stored into the IBI Status Queue.
   ACK response:
   Stores the IBI Status Descriptor into the IBI Status Queue.

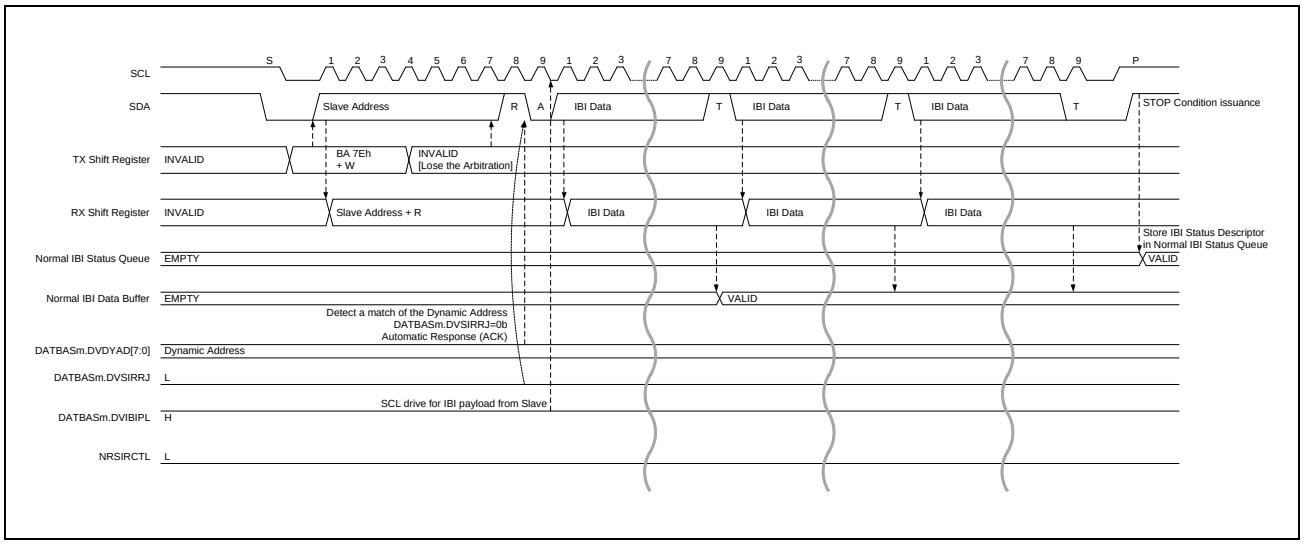


Figure 7.8-78 Slave interrupt request: ACK and DVIBIPL = 1

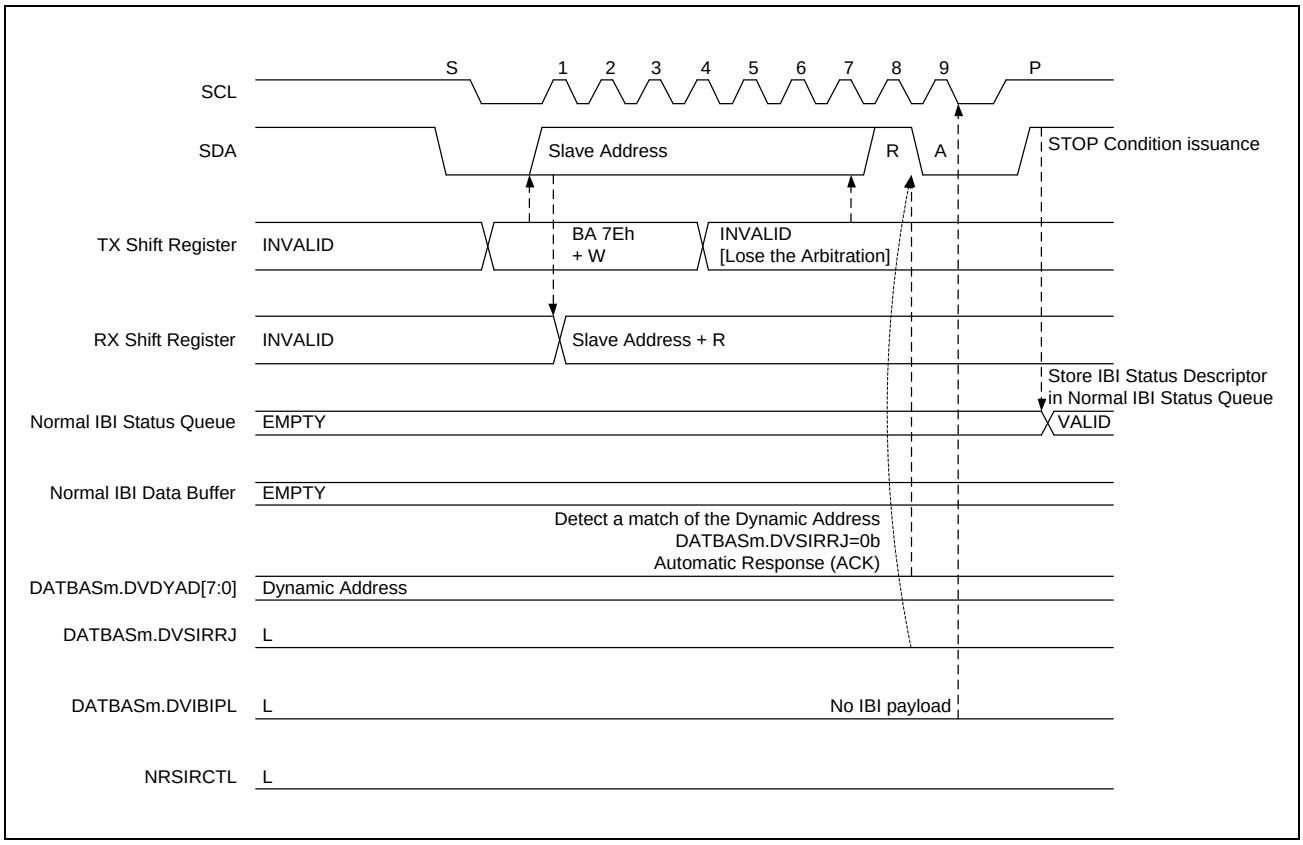


Figure 7.8-79 Slave interrupt request: ACK and DVIBIPL = 0

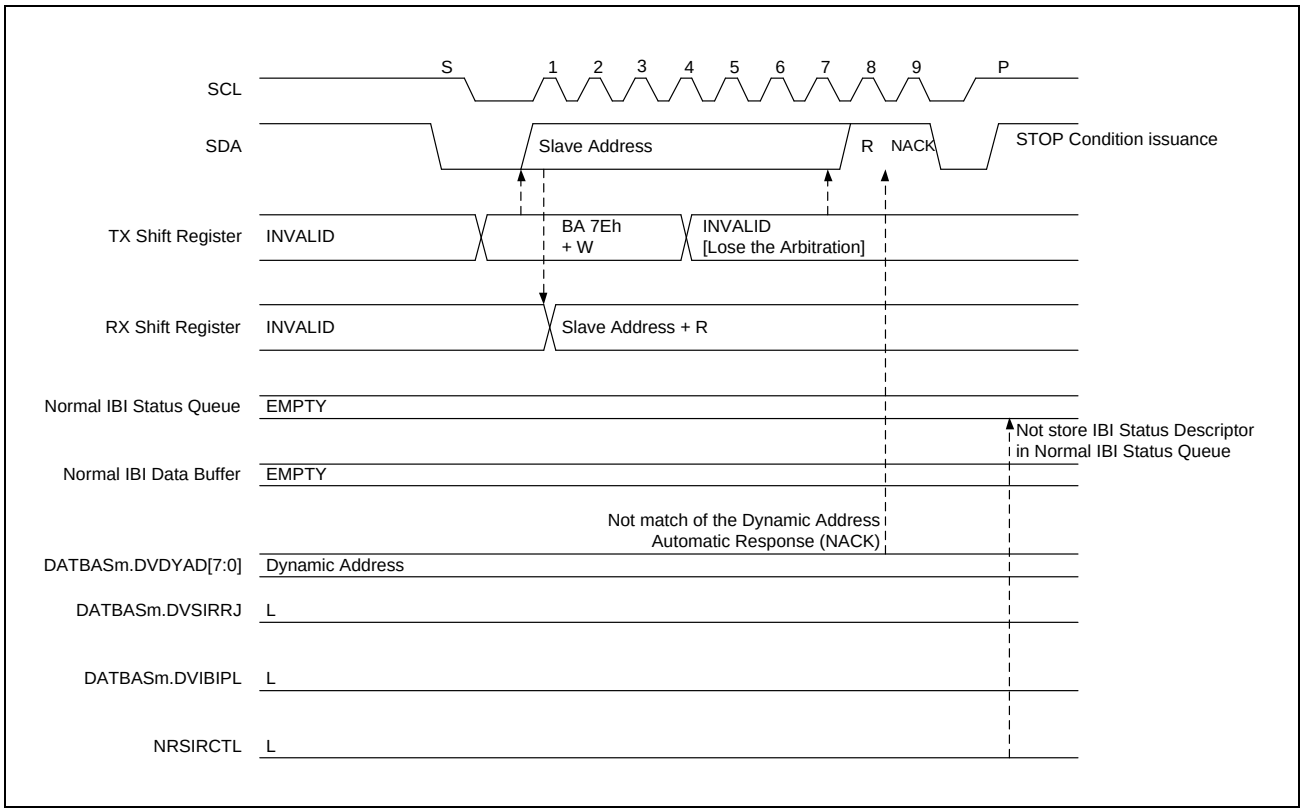


Figure 7.8-80 Slave interrupt request: NACK (not match the SDATBAS0.SDDYAD[6:0]) and NRSIRCTL = 0

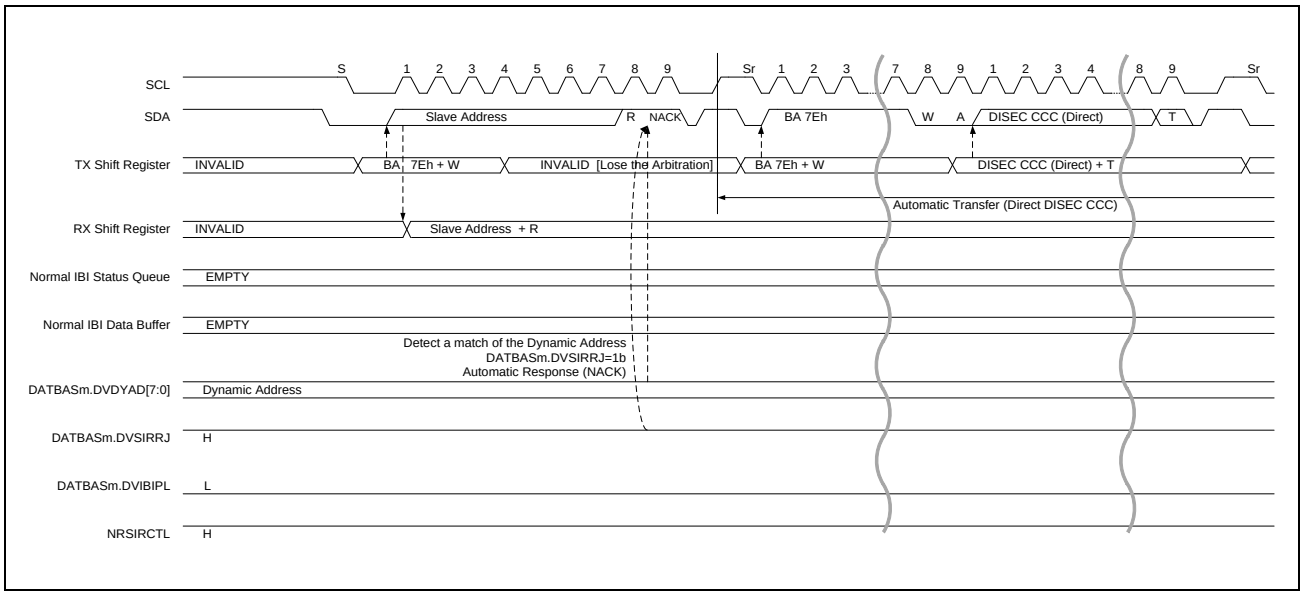


Figure 7.8-81 Slave interrupt request: NACK (DVSIRRJ = 1) and NRSIRCTL = 1 (1/2)

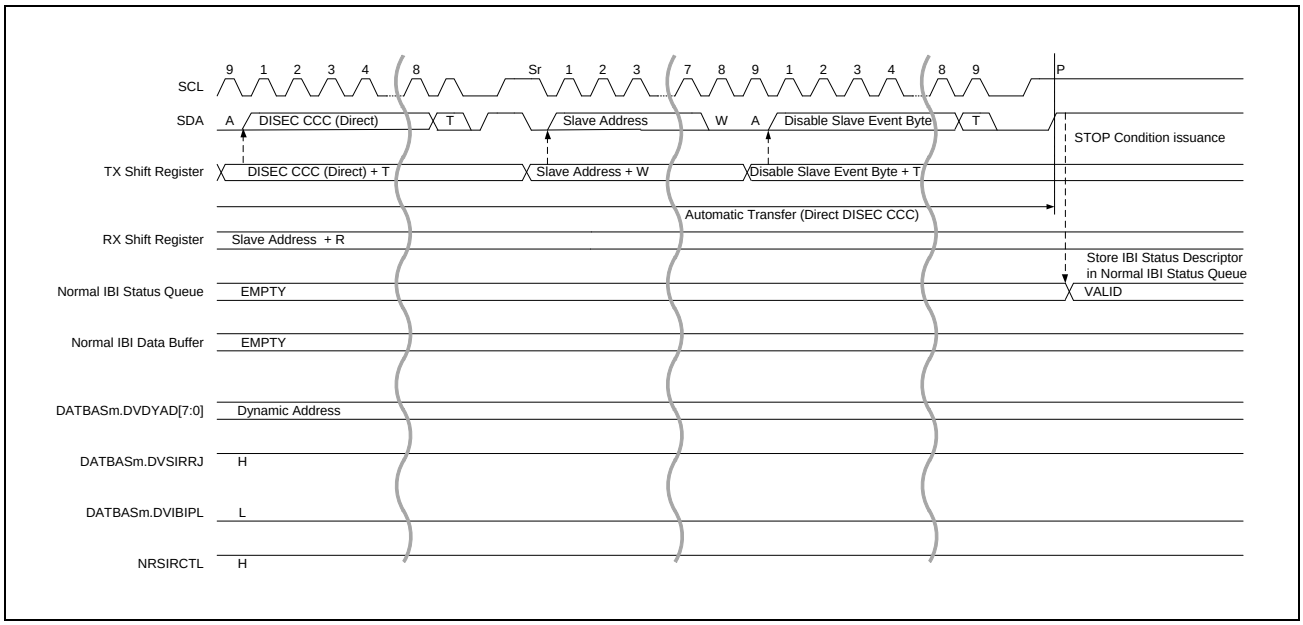


Figure 7.8-82 Slave interrupt request: NACK (DVSIRRJ = 1) and NRSIRCTL = 1 (2/2)

(b) Mastership Request

1. Detect Slave Address with RnW bit Low in Address Header.
2. Compare the detected Slave Address with the DVDYAD[7:0] in each DAT (DATBASm register).
3. When it does not match DAT.DVDYAD[7:0]:
   Responds NACK, then issues the STOP condition.
   When it matches the DAT.DVDYAD[7:0] bits and Device Role[1:0] in RBCR (MSDCTm) is other than I3C Master (01):
   Responds NACK, then issues the STOP condition.
   When it matches the DAT.DVDYAD[7:0] bits and Device Role[1:0] in RBCR (MSDCTm) is I3C Master (01):
   - When DAT.DVMRRJ = 1b
     It operates in the following order.
     (a) Responds NACK.
     (b) Issued Repeated START condition and automatically issues Direct DISEC CCC to the detected Slave.
     (c) Issues the STOP condition.
   - When DAT.DVMRRJ = 0b
     Responds ACK, then issues STOP condition.
4. After issues of STOP condition,
   NACK response:
   – If IBINCTL.NRMRCTL = 0b, the IBI Status Descriptor is not stored into the IBI Status Queue.
   – If IBINCTL.NRMRCTL = 1b, the IBI Status Descriptor is stored into the IBI Status Queue.
   ACK response:
   Stores the IBI Status Descriptor into the IBI Status Queue.

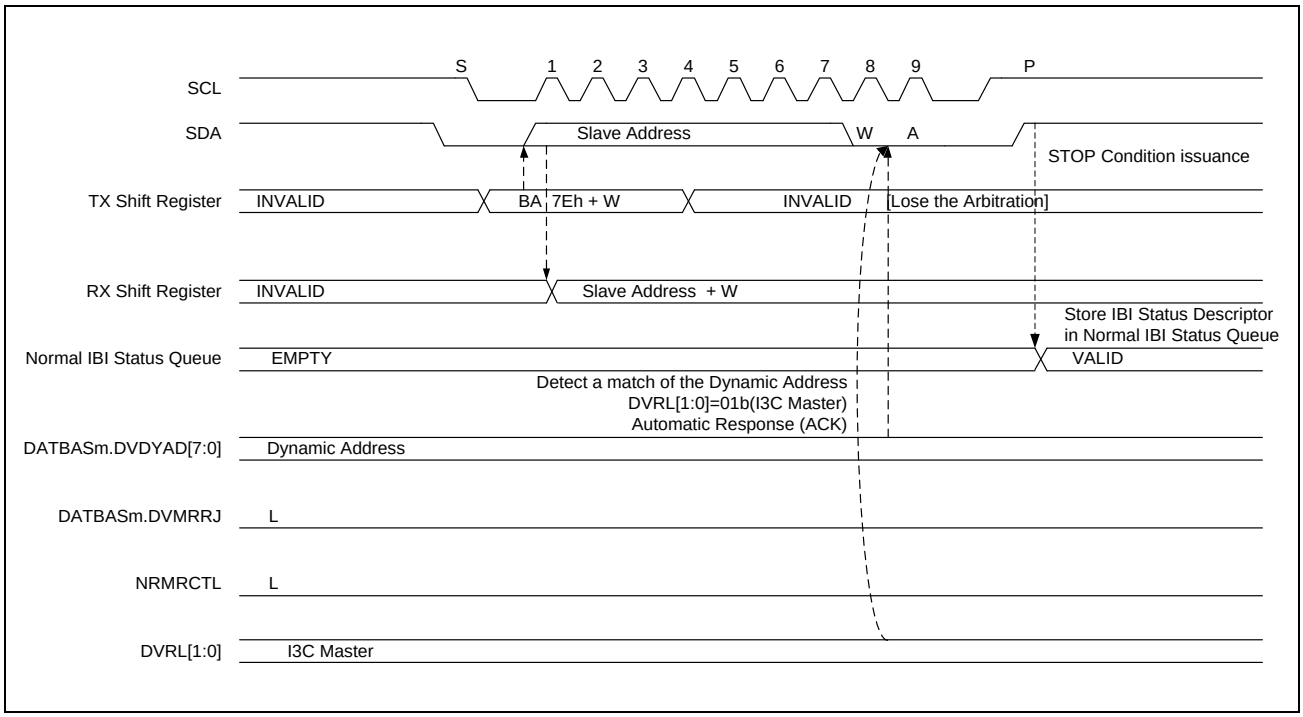


Figure 7.8-83 Mastership request: ACK

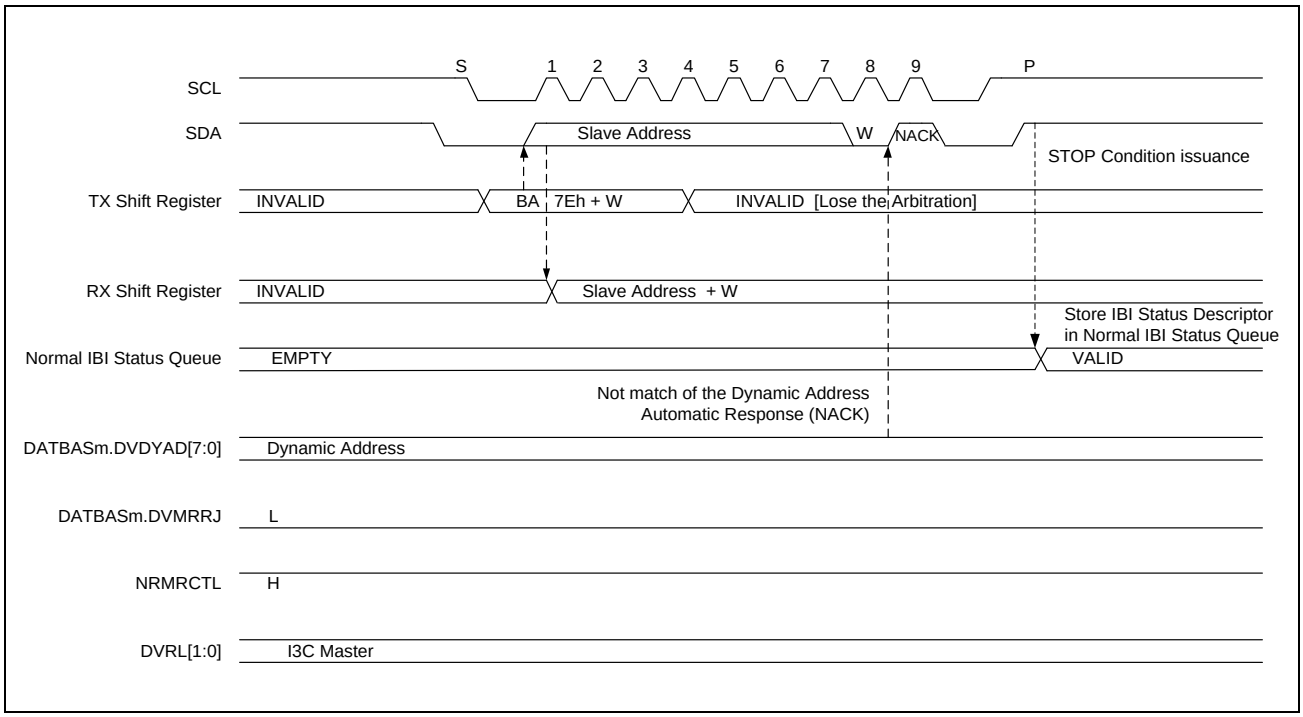


Figure 7.8-84 Mastership request: NACK (not match the DATBASm.DVDYAD[7:0]) and NRMRCTL = 1

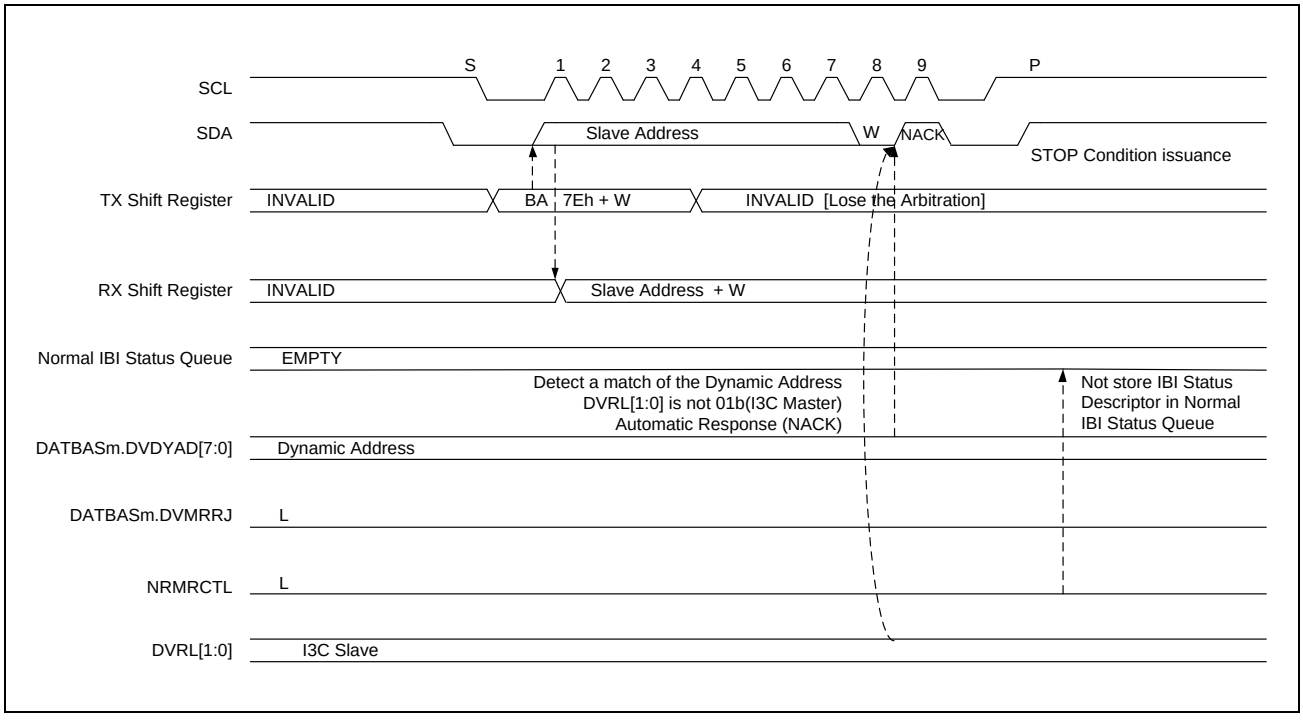


Figure 7.8-85 Mastership request: NACK (Device Role[1:0] is not 01 (I3C master)) and NRMRCTL = 0

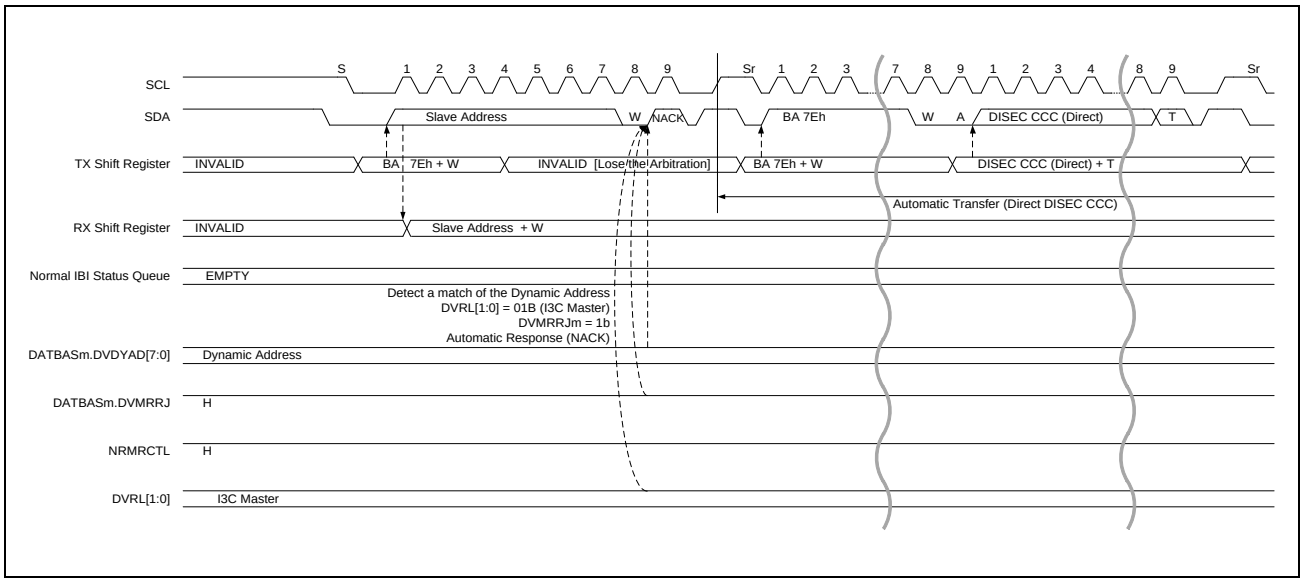


Figure 7.8-86 Mastership request: NACK (DVMRRJ = 1) and NRMRCTL = 1 (1/2)

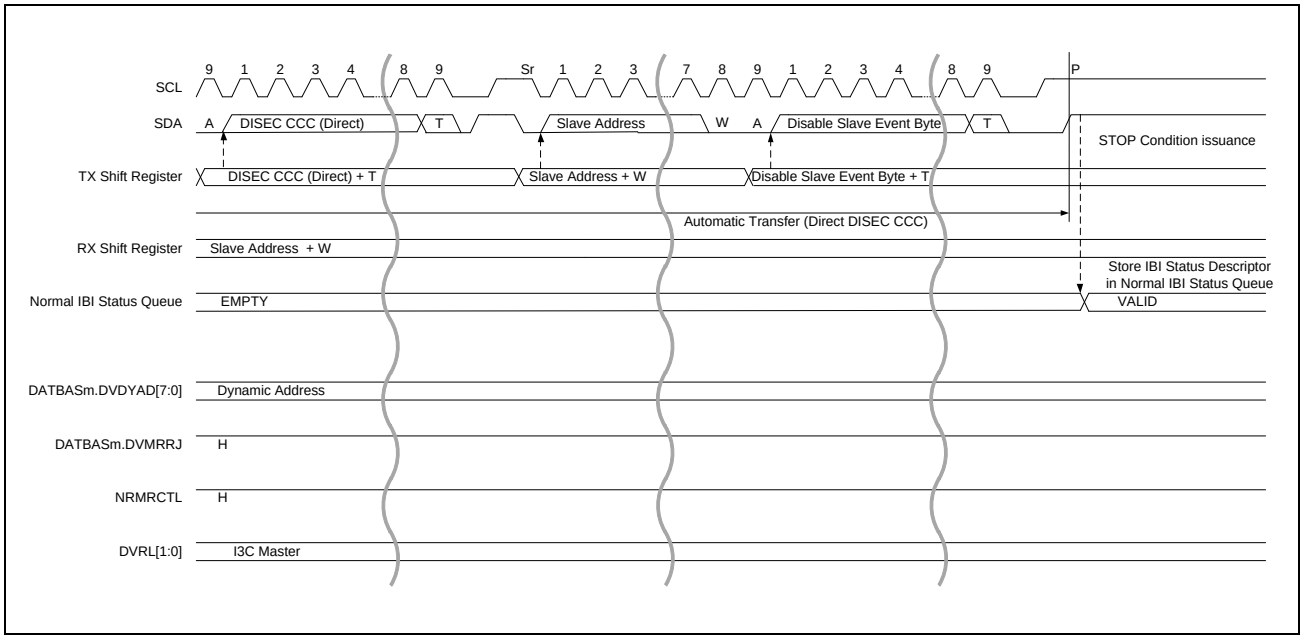


Figure 7.8-87 Mastership request: NACK (DVMRRJ = 1) and NRMRCTL = 1 (2/2)

(c) Hot-Join Event

1. Detect the Hot-Join Address (02h) with RnW bit Low in the Address Header.
2. When BCTL.HJACKCTL = 1b,
   It operates in the following order.
   (a) Responds NACK.
   (b) Issues Repeated START condition and automatically issues Broadcast DISEC CCC.
   (c) Issues the STOP condition.
   When BCTL.HJACKCTL = 0b,
   Responds ACK, then issues STOP condition.
3. After issues of STOP condition, NACK response:
   – If IBINCTL.NRHJCTL = 0b, the IBI Status Descriptor is not stored into the IBI Status Queue.
   – If IBINCTL.NRHJCTL = 1b, store the IBI Status Descriptor into the IBI Status Queue.
   ACK response:
   Stores the IBI Status Descriptor into the IBI Status Queue.

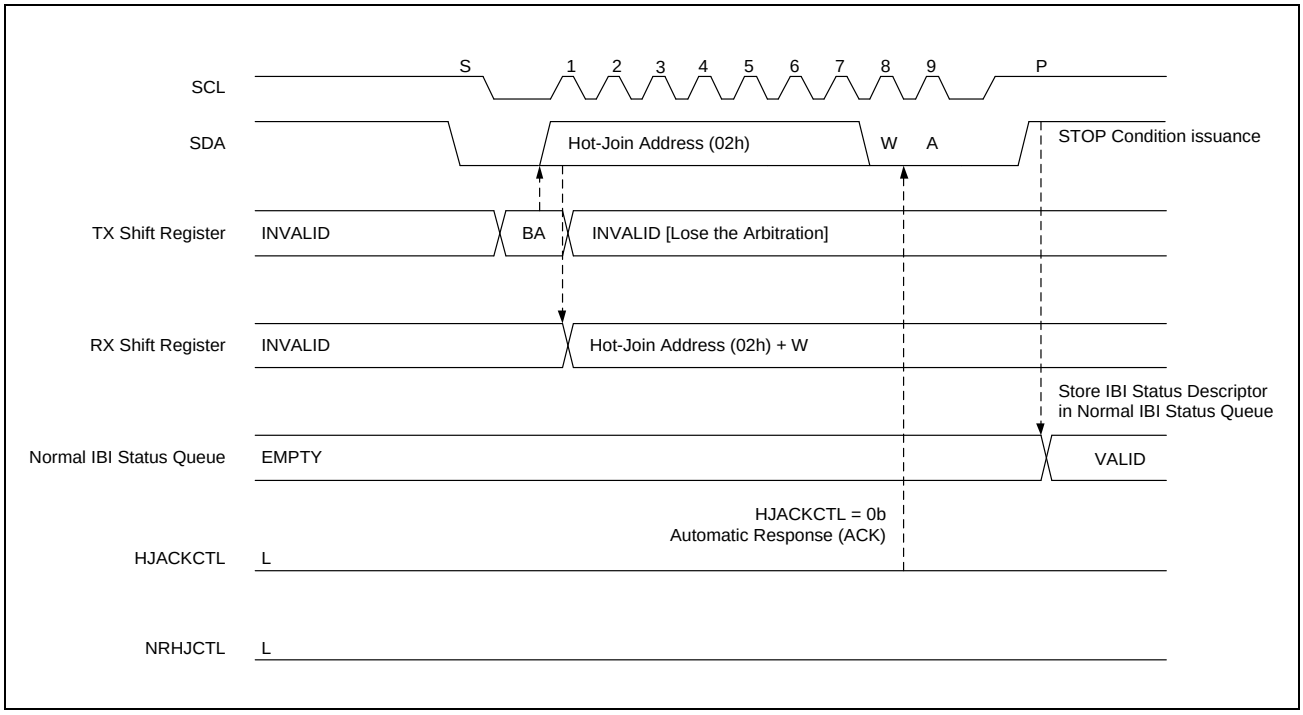


Figure 7.8-88 Hot-join event: ACK

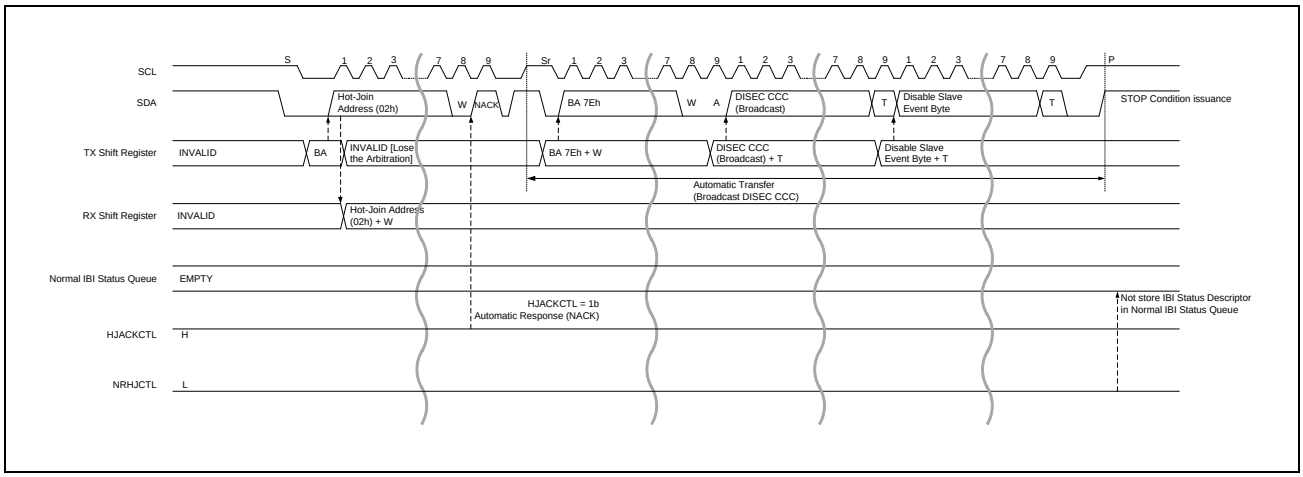


Figure 7.8-89 Hot-join event: NACK (HJACKCTL = 1) and NRHJCTL = 0

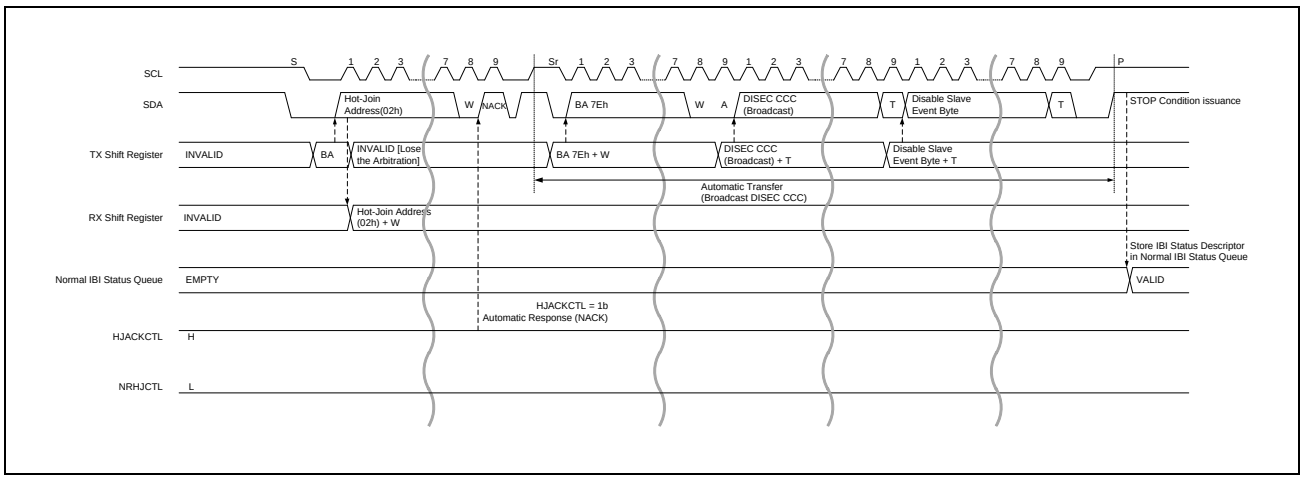


Figure 7.8-90 Hot-join event: NACK (HJACKCTL = 1) and NRHJCTL = 1

(9) Timing Control [I3C mode]

Timing Control is a function that enables Master to efficiently read data from Slave by controlling and grasping the timing at which the Slave Device samples the sensor value.

For details, refer to *5.1.8 Timing Control* of *MIPI I3C Specification v1.0*.

In I3C, Timing Control supports the following three modes.

- Sync Mode
- Async Mode 0 (Asynchronous Basic Mode)
- Async Mode 1 (Asynchronous Advanced Mode)

The resources for realizing Timing Control in each mode are described below.

(a) Sync Mode

1. **<I3C Master>**
   When STCTL.STOE is set to 1, when the master sends an ST message (SETXTIME CCC with ST subcommand), there is a function to issue an event (ri3c_elcstev) under the START condition of the ST message.
   While measuring the T_ph period with an external timer, the start of T_ph and the Delay Time [DT] of the ST message can be measured by capturing the count value with the ri3c_elcstev event.
   The measured value of Delay Time is sent as a DT message (SETXTIME CCC with DT subcommand) following the ST message.
2. **<I3C Slave>**
   When STCTL.STOE is set to 1, there is a function to issue an event (ri3c_elcstev) for each START condition.
   STCTL.STOE is cleared when an ST message is received (SETXTIME CCC using the ST subcommand).
   Check the reception of the ST message with the Receive Status Descriptor, and use an external timer to correct the T_ph period based on the count value captured in the ri3c_elcstev event and the Delay Time obtained from the DT message.
   While measuring the T_ph period with an external timer, the start of T_ph and the Delay Time [DT] of the ST message can be measured by capturing the count value with the ri3c_elcstev event.
   The sampling timing is recalculated by the corrected T_ph.

(b) Async Mode 0 (Asynchronous Basic Mode)
For timing control in Async Mode 0, set the ATCTL register if necessary.

1. **<I3C Master>**
   I3C has counters of MREF (32bit) and MC2 (16bit) for Async Mode 0.

   <MREF Counter>
   When ATCCNTE.ATCE is enabled, the counter starts counting.
   Its value is captured as MREF on the SCL rise edge next to ACK for the IBI transmitted from the I3C Slave.

   <MC2 Counter>
   After enabling ATCCNTE.ATCE, the counter counts up from the SCL rise edge next to ACK for the IBI transmitted from the I3C Slave to the SCL rise edge next to the Tbit after Mandatory Byte, and its value is captured as MC2.
   The MREF and MC2 capture values are stored next to the IBI Status Descriptor when IBI is received from the I3C Slave with the DATBASm.DVIBITS bit set to 1.
   The MREF counter implemented in I3C is a 32-bit counter.

However, if the 32-bit counter is insufficient due to system requirements, I3C has events (ri3c_mrefovf, ri3c_mrefcpt) for expansion. These events are enabled by setting ATCTL.MREFOE to 1b.
ri3c_mrefovf is output when the internal MREF counter overflows.
The MREF counter can be extended by using it as a count event for an external timer. ri3c_mrefcpt is output at the same timing as the capture timing of the internal MREF counter. By using it as the capture timing of the external timer, it can be used as an MREF counter concatenated with the value stored in the IBI Data buffer.

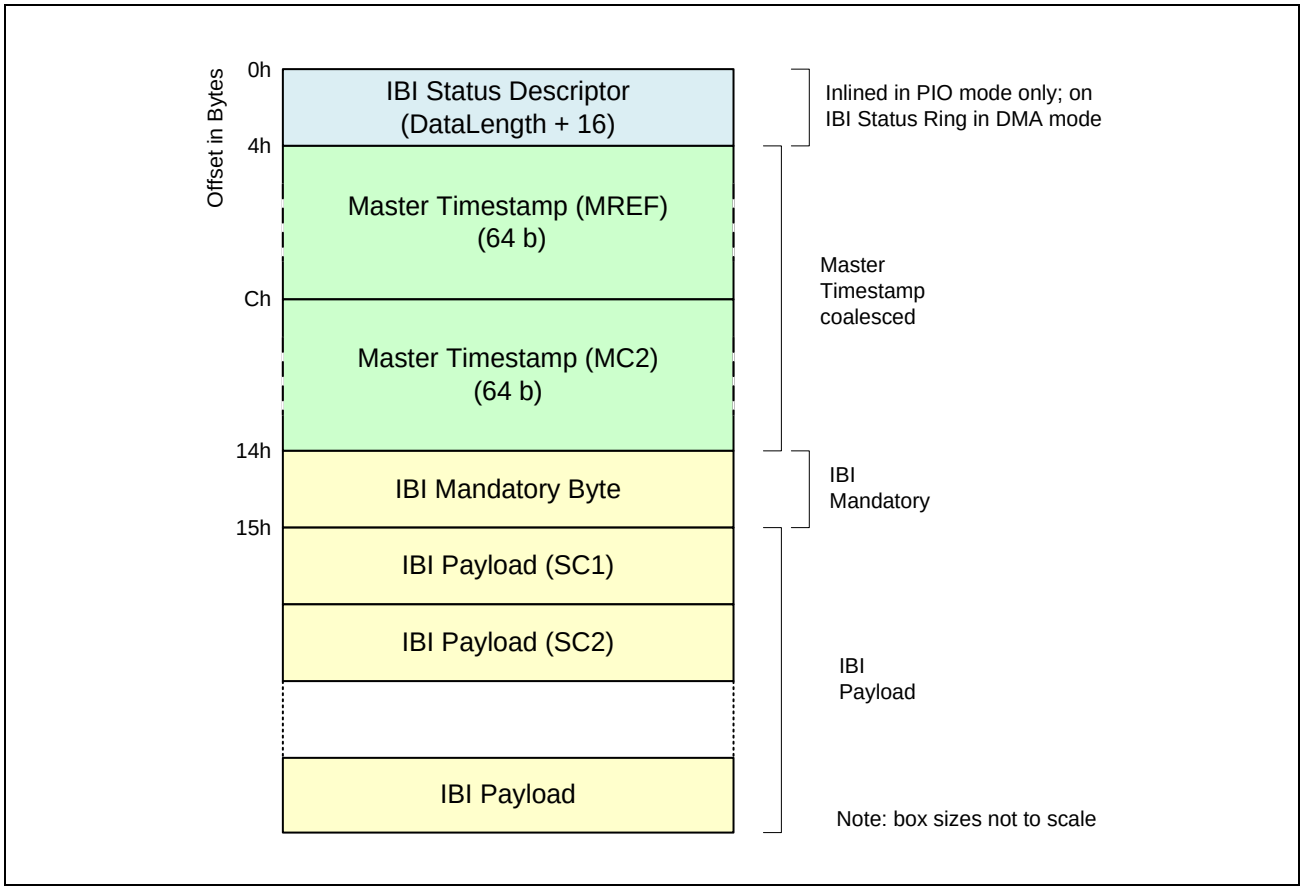


Figure 7.8-91 Master Timestamp Counters for IBI Event

Remark Please evaluate the Sensor Event time of I3C Slave according to the calculation formula of the MIPI I3C specification v1.0 document.

2. **<I3C Slave>**

I3C has counters of SC1 (16bit) and SC2 (8bit) for Async Mode 0.

<SC1 Counter>

After enabling ATCCNTE.ATCE, the counter counts up from SC1 count trigger (ATTRG.ATSTRG bit) to SCL rise edge next to ACK for the IBI, and its value is captured as SC1.

<SC2 Counter>

After enabling ATCCNTE.ATCE, the counter counts up from SCL rise edge next to ACK for the IBI transmitted from I3C Slave to SCL rise edge next to Tbit after Mandatory Byte, and its value is captured as SC2.
When the CETSS.ASYNE [0] bit = 1b and the ITS bit in Command Descriptor for issuing IBI is 1b, the SC1 and SC2 capture values are transmitted following the IBI Mandatory Byte as shown in the following figure.

If the SC1 and SC2 counters overflow, FFFFh and FFh are captured and CETSS.ICOVF is set to 1b.
The DATA_LENGTH[15:0] bits value of the Command Descriptor sets a value obtained by adding the number of data of SC1 and SC2 to the number of transmission data.

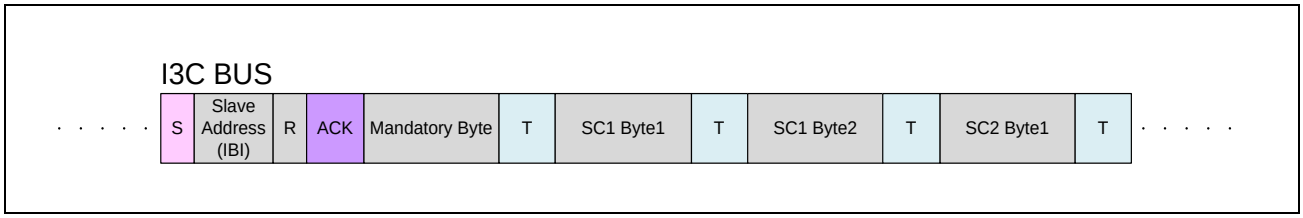


Figure 7.8-92 Example of Asynchronous Mode 0 Timestamp Data Transfer

(c) Async Mode 1 (Asynchronous Advanced Mode)
For timing control in Async Mode 1, set the ATCTL register if necessary.

1. **<I3C Master>**

I3C has counters of MREF (32bit), MSyncCNT (32bit) and MC2 (16bit) for Async Mode 1.

<MREF Counter>
When ATCCNTE.ATCE is enabled, the counter starts counting.
Its value is captured as MREF at the SCL rise edge next to ACK for the IBI transmitted from the I3C Slave.
The MREF extension method described in “1. I3C Master” of “(b) Async Mode 0” is also possible in Async Mode 1.

<MSyncCNT Counter>
When ATCCNTE.ATCE is enabled, the counter starts counting.
Its value is captured as MSyncCNT for each aME (SDA falling edge of START condition), and stored in the capture register.

<MC2 Counter>
After enabling ATCCNTE.ATCE, the counter counts up from SCL rise edge next to ACK for the IBI transmitted from I3C Slave to SCL rise edge next to Tbit after Mandatory Byte, and its value is captured as MC2.
The MREF and MC2 capture values are stored next to the IBI Status Descriptor when the IBI is received from the I3C Slave with the DATBASm.DVIBITS bit set to 1b. (Same as Async Mode 0)
When ATCTL.AMEOE is enabled, an aME Event (ri3c_elcamev) is issued for each aME. Use that event as a trigger to read the MSyncCNT capture value from the MRCCPT register and hold it in an external memory.

2. **<I3C Slave>**

I3C has counters of SC1 (16bit), SC2 (8bit) and aME_TICK (8bit) for Async Mode 1.

<SC1 Counter>
After enabling ATCCNTE.ATCE, the counter counts up from SC1 count trigger*1 to the first aME, and its value is captured as SC1.

<SC2 Counter>
After enabling ATCCNTE.ATCE, the counter counts up from SCL rise edge next to ACK for the IBI transmitted from I3C Slave to SCL rise edge next to Tbit after Mandatory Byte, and its value is captured as SC2.

<aME_TICK Counter>

After enabling ATCCNTE.ATCE, the counter counts every aME, and its value is captured as aME_TICK at the SCL rise edge next to ACK for the IBI.
The aME_TICK counter is cleared on the first aME after the SC1 count trigger.

When the CETSS.ASYNE[1] bit = 1b and the ITS bit in Command Descriptor for issuing IBI is 1b, the SC1, SC2 and aME_TICK capture values are transmitted following the IBI Mandatory Byte as shown in the following figure.
If the SC1 and SC2 counters overflow, FFFFh and FFh are captured and CETSS.ICOVF is set to 1b.
The DATA_LENGTH[15:0] bits value of the Command Descriptor sets a value obtained by adding the number of data of SC1, SC2 and aME_TICK to the number of transmission data.

Note 1. SW or external trigger can be selected by selection bits.
The SC1 count trigger select method described in “2. I3C Slave” of “(b) Async Mode 0” is also possible in Async Mode 1

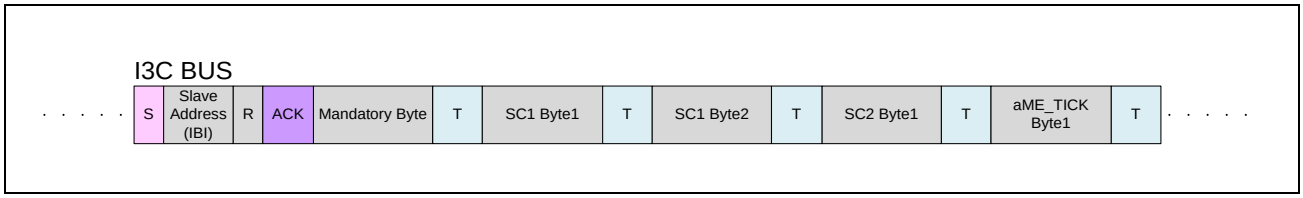


Figure 7.8-93 Example of Asynchronous Mode 1 Timestamp Data Transfer

(10) Port Control

(a) Extra SCL Clock Cycle Output Function

In master mode, I3C module has a facility for the output of extra SCL clock cycles to release the SDA line of the slave device from being held at the low level due to the master being out of synchronization with the slave device.

This function is mainly used in master mode to release the SDA line of the slave device from the state of being fixed to the low level by including extra cycles of SCL output from I3C with single cycles of the SCL clock as the unit in the case of a bus error where I3C cannot issue a Repeated START condition or a STOP condition because the slave device is holding the SDA line at the low level. Do not use this facility in normal situations. Using it when communications are proceeding correctly will lead to malfunctions.

When the OUTCTL.EXCYC bit is set to 1b, an additional clock pulse at the frequency set by the REFCKCTL.IREFCKS[2:0] bits and the STDBR.SBRHO[7:0] and STDBR.SBRLO[7:0] registers is output from the SCL pin. After output of this clock pulse, the EXCYC bit automatically becomes 0b. After confirming that the EXCYC bit is 0, wait for the setup time of the Repeated START condition or STOP condition, and then confirm the detection of the Repeated START condition or STOP condition. If the Repeated START condition or STOP condition is not detected, consecutive additional clock pulses can be output by writing 1b to the EXCYC bit again.

When I3C module is in master mode and the slave device is holding the SDA line at the low level because synchronization with the slave device has been lost due to the effects of noise, etc., the output of a Repeated START condition or a STOP condition is not possible. The facility for output of an extra cycle of the SCL clock can be used to output extra cycles of SCL one by one to make the slave device release the SDA line from being held at the low level, thus recovering the bus from an unusable state. Release of the SDA line by the slave device can be monitored by reading the SDILV bit in PRSTDBG. After the SDA line has been released by the slave device, the preset of a Repeated START condition or a STOP condition is issued.

Use this function with the BFCTL.MALE bit set to 0b (master arbitration-lost detection is disabled).

[Output conditions for using the EXCYC bit in OUTCTL]

- When the bus is free (BFREF flag in BCST = 1b) or in master mode (CRMS bit = 1b in PRSST and BFREF flag = 0b in BCST)
- When the communication device does not hold the SCL line low

Figure 7.8-94 shows the operation timing of the extra SCL clock cycle output function (EXCYC bit).

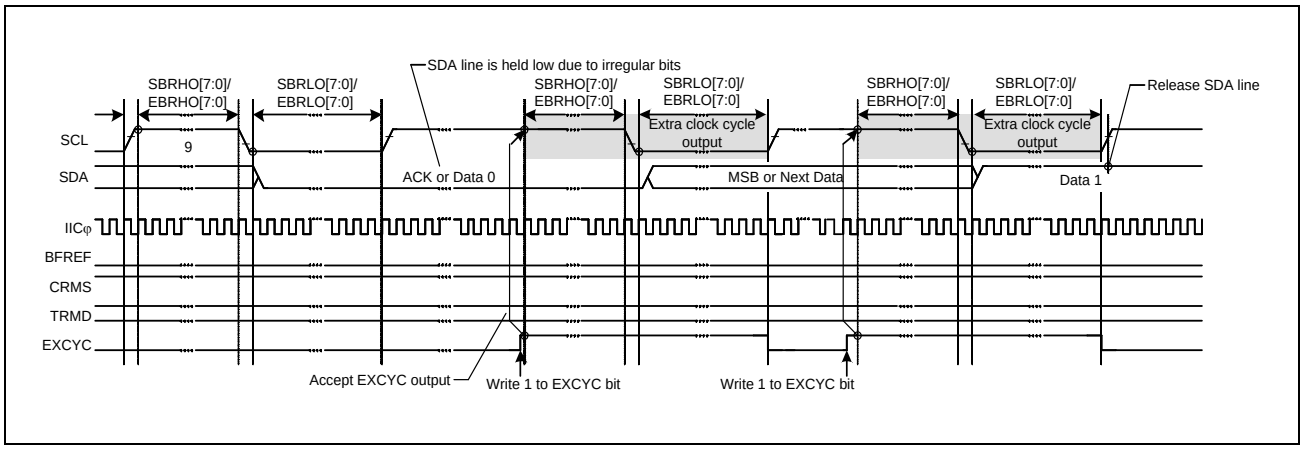


Figure 7.8-94 Extra SCL clock cycle output function (EXCYC bit)

(11) SMBus Operation [I²C mode]

I3C is available for data communication conforming to the SMBus (Version 2.0). To perform SMBus communication, set the BFCTL.SMBS bit to 1b. To use the transfer rate within a range of 10 kbps to 100 kbps of the SMBus specification, set the REFCKCTL.IREFCKS[2:0] bits, the STDBR.SBRHO[7:0] bits, and the STDBR.SBRLO[7:0] bits. In addition, determine the values of the OUTCTL.SDODCS bit and the OUTCTL.SDOD[2:0] bits to meet the data hold time specification of 300 ns or more. If I3C is used only as an I²C slave device, the transfer rate setting is not necessary, whereas the STDBR.SBRLO[7:0] bits needs to be set to a value longer than the data setup time (250 ns).

For the SMBus device default address (1100 001b), use one of the slave device address table basic registers 0 to 2 (SDATBASn.SDSTAD[6:0] bits (n = 0 to 2)), and set the corresponding SDATBASn.SDADLS bit (7-bit/10-bit address format select) (n = 0 to 2) to 0 (7-bit address format).

When transmitting the UDID (Unique Device Identifier), set the BFCTL.SALE bit to 1b to enable the slave arbitration-lost detection function.

(a) SMBus Timeout Measurement

(a-1) Measuring timeout of slave device

The following period (timeout interval: $T_{LOW:SEXT}$) must be measured for slave devices in SMBus communication.

- From START condition to STOP condition

To measure timeout for slave devices, measure the period from START condition detection to STOP condition detection with the GPT timer using a START condition detection interrupt (INT_ri3c_st_n) and STOP condition detection interrupt (INT_ri3c_sp_n) of I3C. The measured timeout period must be within the total clock low-level period [slave device] $T_{LOW:SEXT}$: 25 ms (max.) of the SMBus specification.

If the time measured with the GPT exceeds the clock low-level detection timeout TTIMEOUT: 25 ms (min.) of the SMBus specification, the slave device must release the bus by writing 1b to the RSTCTL.INTLRST bit to issue an internal reset of I3C. When an internal reset is issued, I3C stops driving the bus for the SCL pin and SDA pin and make the SCL/SDA pin outputs high-impedance, which releases the bus.

(a-2) Measuring timeout of master device

The following periods (timeout interval: $T_{LOW:MEXT}$) must be measured for master devices in SMBus communication.

- From START condition to acknowledge bit
- Between acknowledge bits
- From acknowledge bit to STOP condition

To measure timeout for master devices, measure these periods with the GPT timer using a START condition detection interrupt (INT_ri3c_st_n), STOP condition detection interrupt (INT_ri3c_sp_n), and transmit end interrupt (I3Cn_TEND) or receive data buffer full interrupt (INT_ri3c_rx_n) of I3C. The measured timeout period must be within the total clock low-level extended period (master device) $T_{LOW:MEXT}$: 10 ms (max.) of the SMBus specification, and the total of all $T_{LOW:MEXT}$ from START condition to STOP condition must be within $T_{LOW:SEXT}$: 25 ms (max.).

For the ACK receive timing (rising edge of the ninth SCL clock cycle), monitor the BST.TENDF flag in master transmit mode (master transmitter) and the NTST.RDBFF0 flag in master receive mode (master receiver). For this reason, perform bytewise transmit operation in master transmit mode, and hold the SCSTRCTL.ACKTWE bit 0 until the byte just before reception of the final byte in master receive mode. While the ACKTWE bit = 0b, the RDBFF0 flag is set to 1b at the rising edge of the ninth SCL clock cycle.

If the period measured with the GPT exceeds the total clock low-level extended period (master device) $T_{LOW:MEXT}$: 10 ms (max.) of the SMBus specification or the total of measured periods exceeds the clock low-level detection timeout

TTIMEOUT: 25 ms (min.) of the SMBus specification, the master device must stop the transaction by issuing a STOP condition. In master transmit mode, immediately stop the transmit operation (writing data to NTDTBP0).

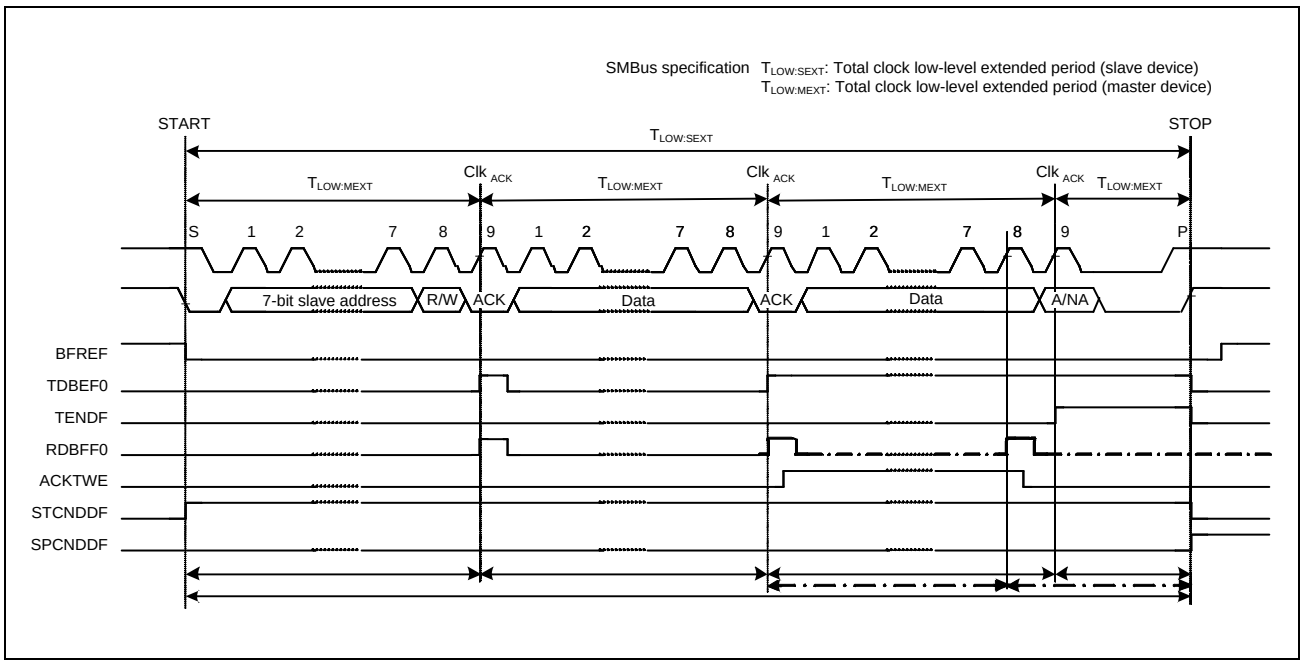


Figure 7.8-95 SMBus timeout measurement

(b) Packet Error Code (PEC)

This product incorporates a CRC calculator. The CRC calculator enables transmission of a packet error code (PEC) or checking the received data of the SMBus in data communication of I3C. For the CRC generating polynomials of the CRC calculator, see **7.6 CRC Operation Unit (CRC)**.

The PEC data in master transmit mode can be generated by writing all transmit data to the CRC data input register (CRCDIR) in the CRC calculator.

The PEC data in master receive mode can be checked by writing all receive data to CRCDIR in the CRC calculator and comparing the obtained value in the CRC data output register (CRCDOR) with the received PEC data.

To send ACK or NACK according to the match or mismatch result when the final byte is received as a result of the PEC code check, set the SCSTRCTL.ACKTWE bit to 1 before the rising edge of the eighth SCL clock cycle during reception of the final byte, and hold the SCL line low at the falling edge of the eighth clock cycle.

(c) SMBus Host Notification Protocol (Notify ARP Master Command)

In communications over an SMBus, a slave device can temporarily act as a master device to notify the SMBus host (or ARP master) of its own slave address or to request its own slave address from the SMBus host.

For this product to operate as an SMBus host (or ARP master), the host address (0001 000b) sent from the slave device must be detected as a slave address, so I3C has a function for detecting the host address. To detect the host address as a slave address, set the BFCTL.SMBS bit and the SVCTL.HOAE bit to 1b. Operation after the host address has been detected is the same as normal slave operation.

(12) Common Command Codes (CCC) [I3C mode]

For the common command code (CCC), refer to *5.1.9 Common Command Codes (CCC)* in *MIPI I3C Specification v1.0*. This I3C is based on *Table 15 I3C Common Command Codes* in *5.1.9.3 Common Command Definitions* of *MIPI I3C Specification v1.0*.

The MIPI Reserved area and Vendor Extension area of Command Code are described below.

- I3C Master mode:
  When sending CCCs in the MIPI Reserved area and Vendor Extension area from the I3C Master, only Broadcast/Direct SET CCCs using the Immediate Transfer Command can be sent.
  Sending Direct GET CCC is not supported.
- I3C Slave mode:
  Only Broadcast/Direct SET CCC can be received for CCC in MIPI Reserved area and Vendor Extension area.
  Receiving Direct GET CCC is not supported.

7.8.4.2.4 Error Detection

(1) SDR Error Detection and Recovery Methods for I3C Slave Devices [I3C mode]

The seven error types summarized in **Table 7.8-14** are supported for all I3C slave devices. Each error type is further explained below the table.

Table 7.8-14 SDR slave error types

Error type	Description	Error detection method
S0	Broadcast address/W (= 7Eh/W) or Dynamic address/RW	Detect any of the following: 3Eh/W 5Eh/W 6Eh/W 76h/W 7Ah/W 7Ch/W 7Fh/W 7Eh/R
S1	CCC code	Parity check, using T-Bit
S2	Write data	Parity check, using T-Bit
S3	Assigned address during Dynamic address arbitration	Parity check, using PAR Bit
S4	7Eh/R after Sr during Dynamic address arbitration	Detect any value other than 7Eh/R after Sr during Dynamic Address Arbitration
S5	Transaction after detecting CCC	Detect illegally formatted CCC
S6 (optional)	Monitoring error	Slave detects (through monitoring) that transmitted Data differs from what it intended to transmit (Does not apply during Dynamic address arbitration)

(2) SDR Error Detection and Recovery Methods for I3C Master Devices [I3C mode]

The two error types summarized in **Table 7.8-15** are supported for all I3C master devices. Each error type is further explained below the table.

Table 7.8-15 SDR master error types

Error type	Description	Error detection method
M0	Transaction after sending CCC	Detect illegally formatted CCC
M1 (optional)	Monitoring error	Master detects (through monitoring) transmitted data different from what it intended to transmit (Does not apply during Dynamic address arbitration)
M2	No response to Broadcast address (7Eh)	Master detects NACK after Broadcast address (7Eh) transmission

(3) Timeout Error Detection

I3C includes a timeout function for detecting when the SCL line has been stuck longer than the predetermined time. I3C can detect an abnormal bus state by monitoring that the SCL line is stuck low or high for a predetermined time.

The timeout function monitors the SCL line state and counts the low-level period or high-level period using the internal counter. The timeout function resets the internal counter each time the SCL line changes (rising or falling), but continues to count unless the SCL line changes. If the internal counter overflows due to no SCL line change, I3C can detect the timeout and report the bus hung state.

This timeout function is enabled when BSTE.TODE = 1b. It detects a hung state that the SCL line is stuck low or high during the following conditions: (When TMOCTL.TOMDS[1:0] = 00b)

- The bus is busy (BCST.BFREF = 0b) in master mode (PRSST.CRMS = 1b).
- I3C's own slave address is detected (SVST register is not 0000h) and the bus is busy (BCST.BFREF = 0b) in slave mode (PRSST.CRMS = 0b).
- The bus is free (BCST.BFREF = 1b) while generation of a START condition is requested (CNDCTL.STCND = 1b).

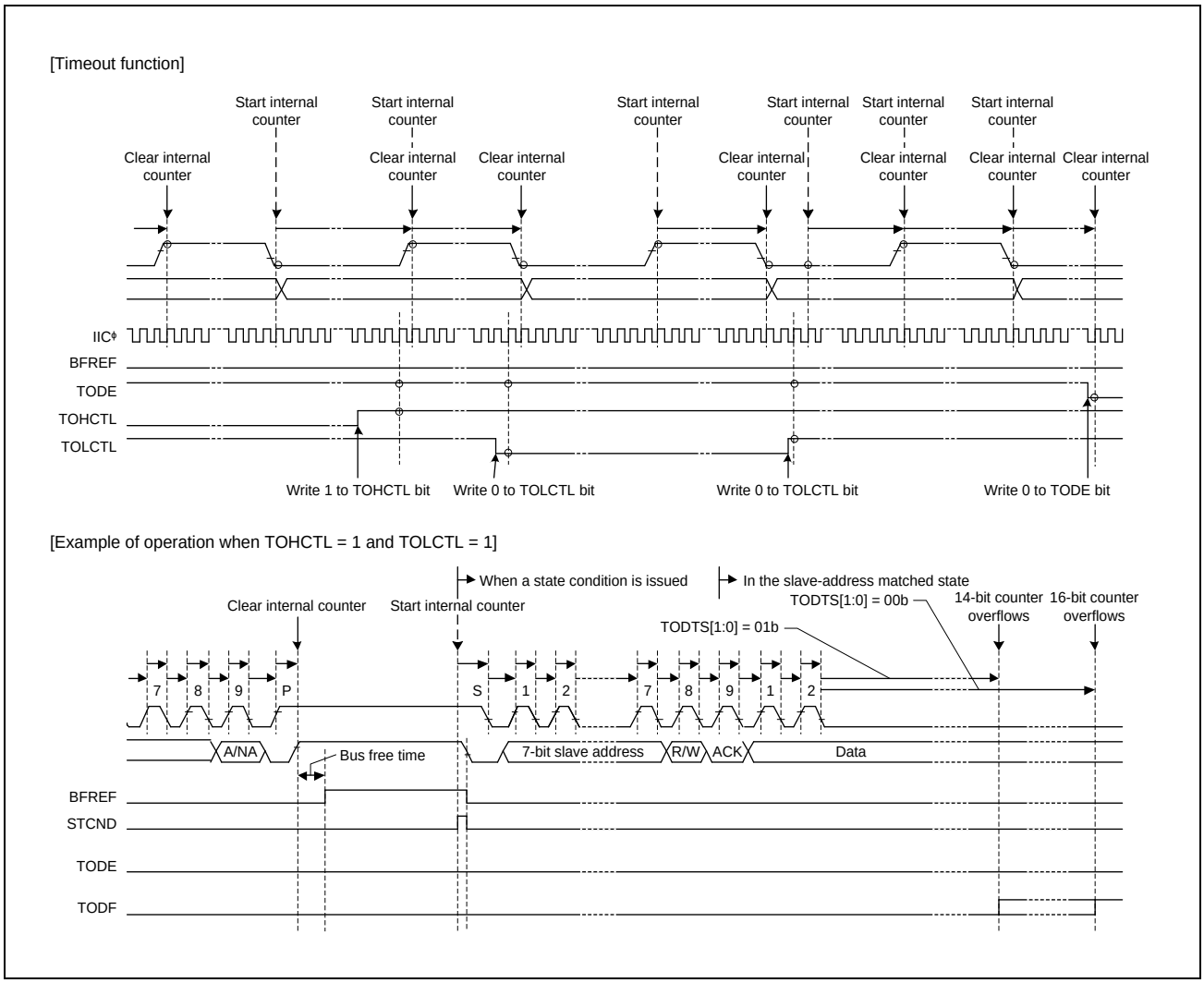


Figure 7.8-96 Timeout error detection (TODE, TODTS[1:0], TOHCTL, and TOLCTL bits)

(4) Resume Operation [I3C mode]

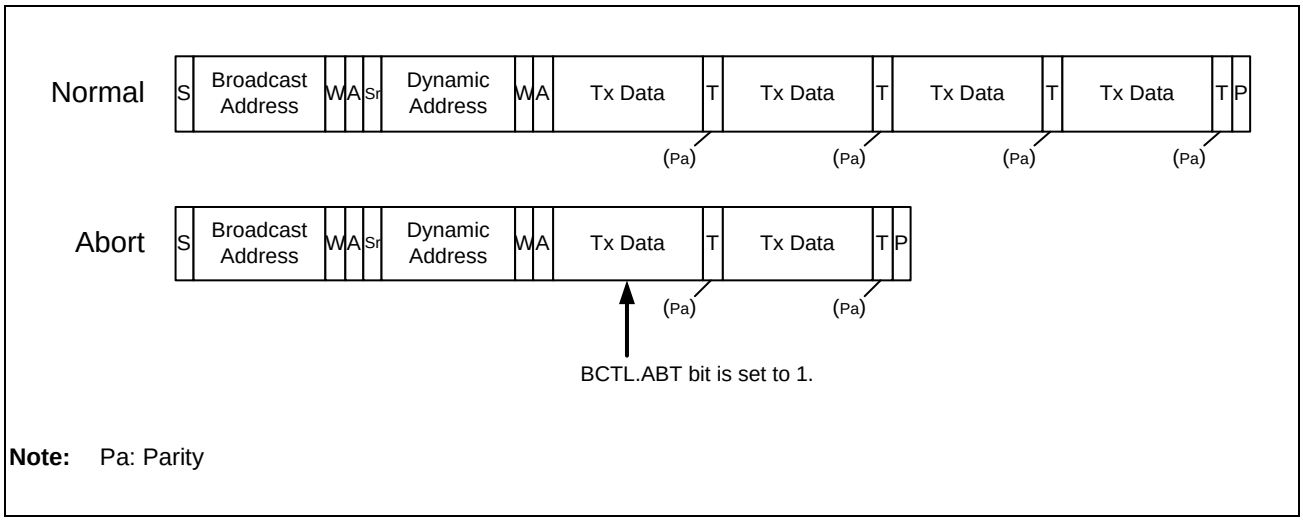
I3C enters the Halt state as a result of any type of error occurring in a transfer.

The error type is indicated by the field ERR_STATUS in Response Descriptor or Receive Status Descriptor. After I3C has entered the Halt state, the user must write the value 1 to the RSM bit to resume operation. I3C shall auto-clear the RSM bit once it has initiated the next Command transfer or detected the START condition.

(5) Abort Operation [I3C mode]

When the BCTL.ABT bit is set to 1b, I3C relinquish control of the bus before completing the currently issued transfer. In response to an abort request, I3C issues the STOP condition on the bus after the complete data byte is transferred or received. After I3C has aborted, the user shall clear the BCTL.ABT bit to allow operation on the bus.

Remark For Read transaction, when BCTL.ABT is set to 1b, that receive data is stored in Receive data buffer.



Note: Pa: Parity

Figure 7.8-97 Abort operation of SDR write transfer

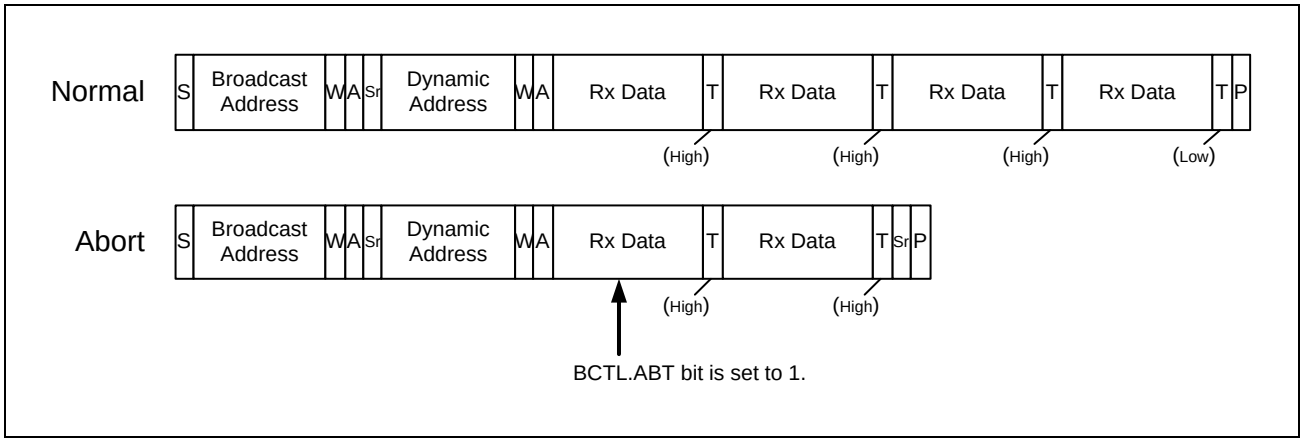


Figure 7.8-98 Abort operation of SDR read transfer

(6) Error Recovery Operation [I3C mode]

(a) Error Recovery Operation

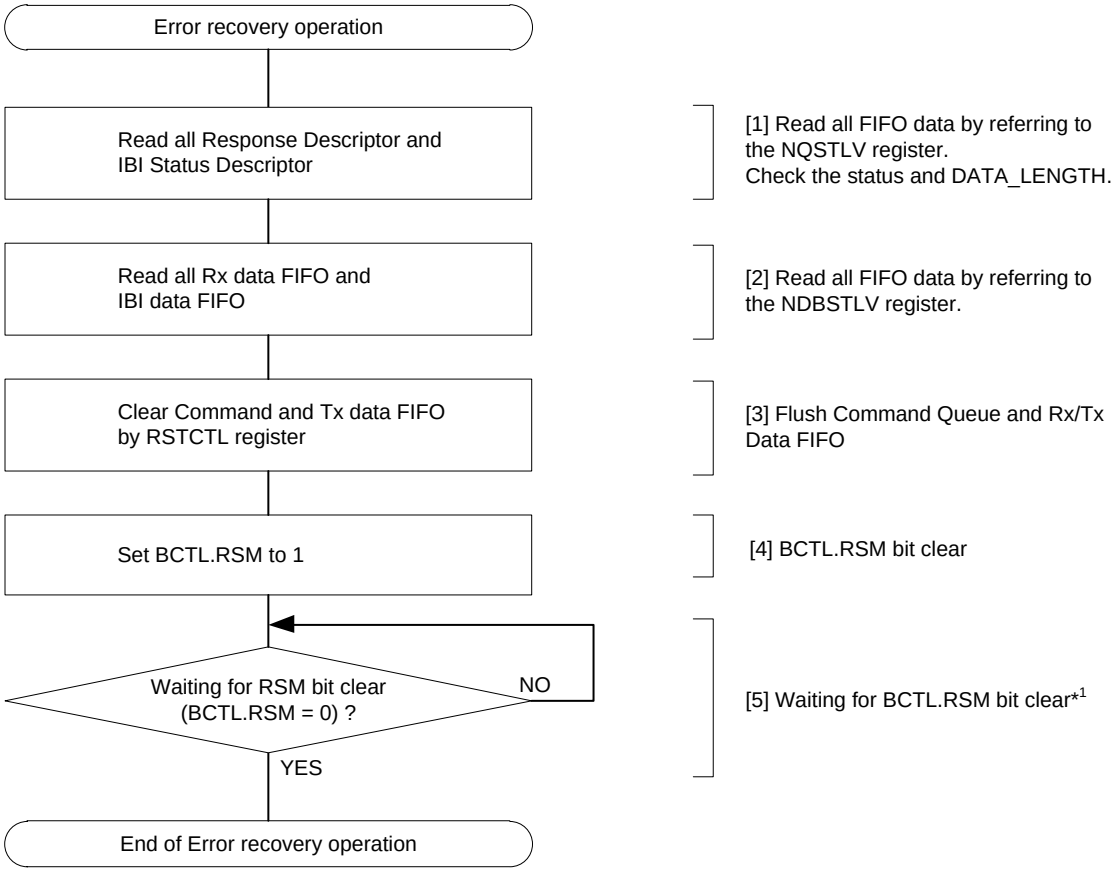
When an error occurs, the flags of INST.INEF, NTST.TEF, NTST.TABTF, HTST.TEF and HTST.TABTF are set to 1b according to the cause of the error.

Or the interrupts INT_ri3c_ierr_n, INT_ri3c_terr_n and INT_ri3c_abort_n associated with each flags are asserted. (When detection and interrupts are enabled.)

There is a possibility of communication error or I3C internal error.

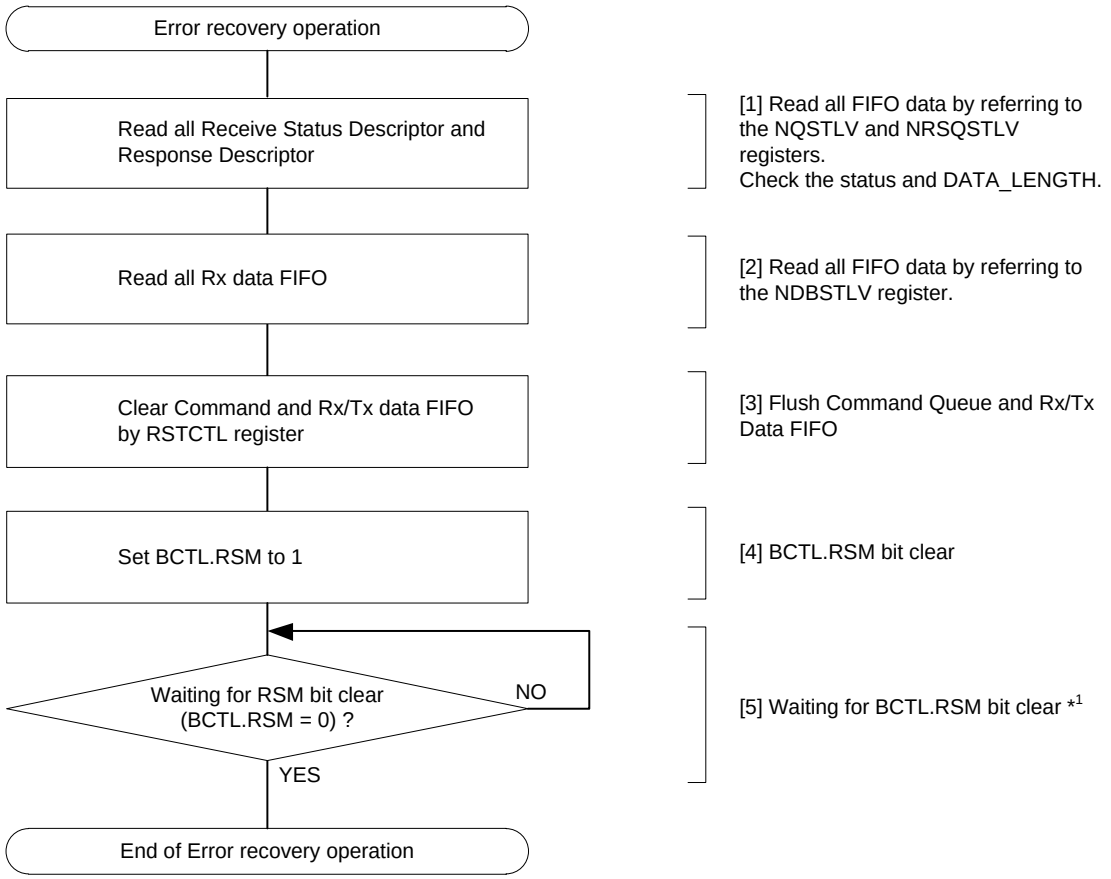
If an error occurs, I3C will be suspended. (BCTL.RSM becomes 1b.) After I3C is suspended, the application must write the value 1 to the BCTL.RSM bit to resume I3C operation and recover from the suspended state.

The error recovery flow is shown in **Figure 7.8-99** and **Figure 7.8-100**.



Note 1. The Command Descriptor can be written to the Command Queue via writing to the Command Queue Port (NCMDQP) before the RSM bit is cleared.

Figure 7.8-99 Example of Error Recovery Operation Flowchart for I3C Master



Note 1. The Command Descriptor for issuing IBI can be written to the Command Queue via writing to the Command Queue Port (NCMDQP) before the RSM bit is cleared.

Figure 7.8-100 Example of Error Recovery Operation Flowchart for I3C Slave

When I3C Slave recovers from an error according to the error recovery flow, after setting BCTL.RSM to 1b, BCTL.RSM becomes 0b after detecting a state in which Bus Available period communication is not performed on I3C Bus.

If communication occurs on the I3C bus within the Bus Available period, BCTL.RSM will not be set to 0b and error recovery will not be completed, and a NACK response will be transmitted.

(b) Master Error Detection and Escalation Handling

If the Master does not receive an ACK of a transmitted private Message to a Slave and Steps 1 and 2 described in Chapter 5.1.10.2.4 of MIPI I3C Spec v1.0 fail, the processing flow of Step 3 shown in **Figure 7.8-101** and **Figure 7.8-102** is followed.

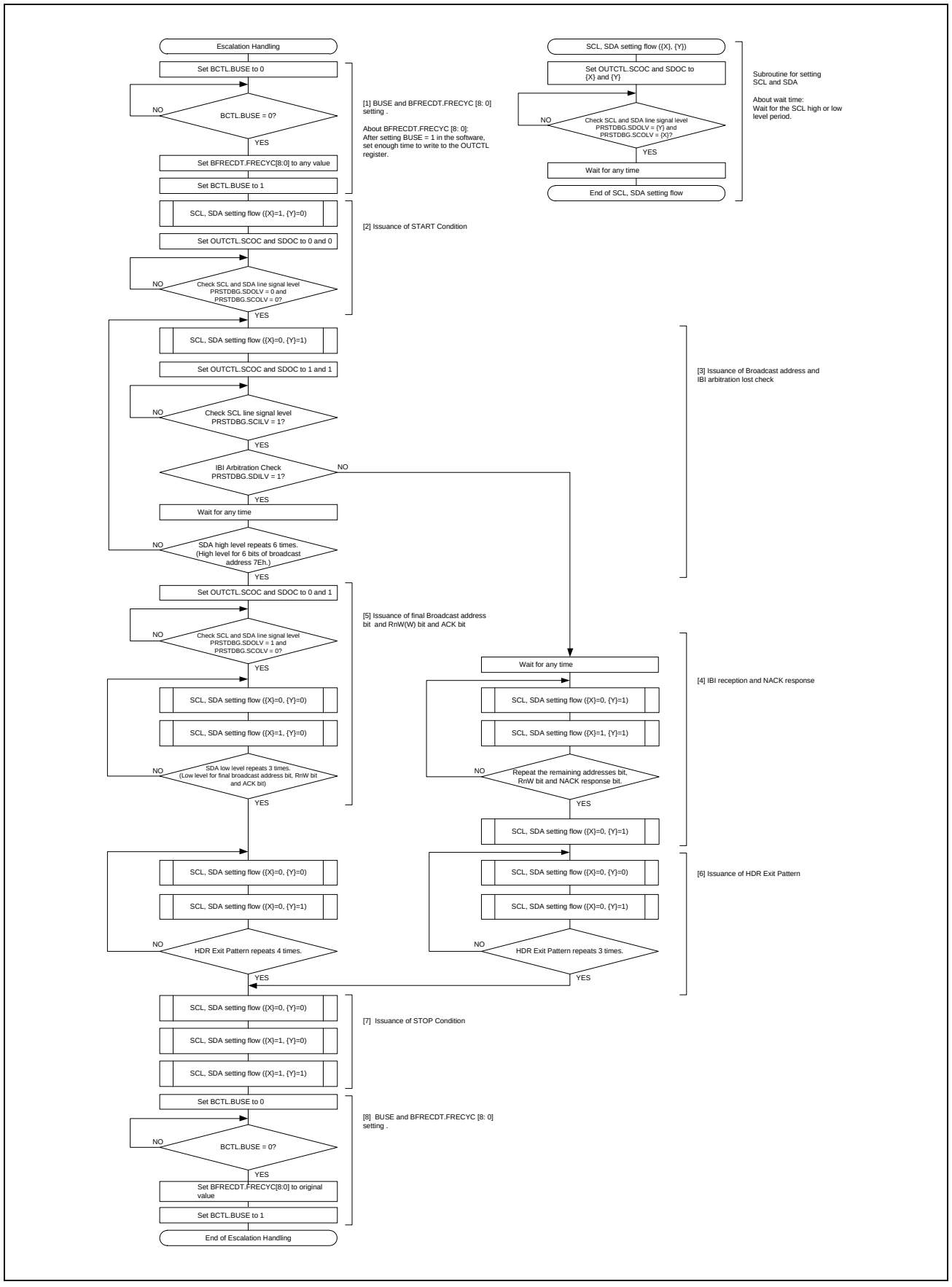


Figure 7.8-101 Escalation Handling Flowchart for I3C Master

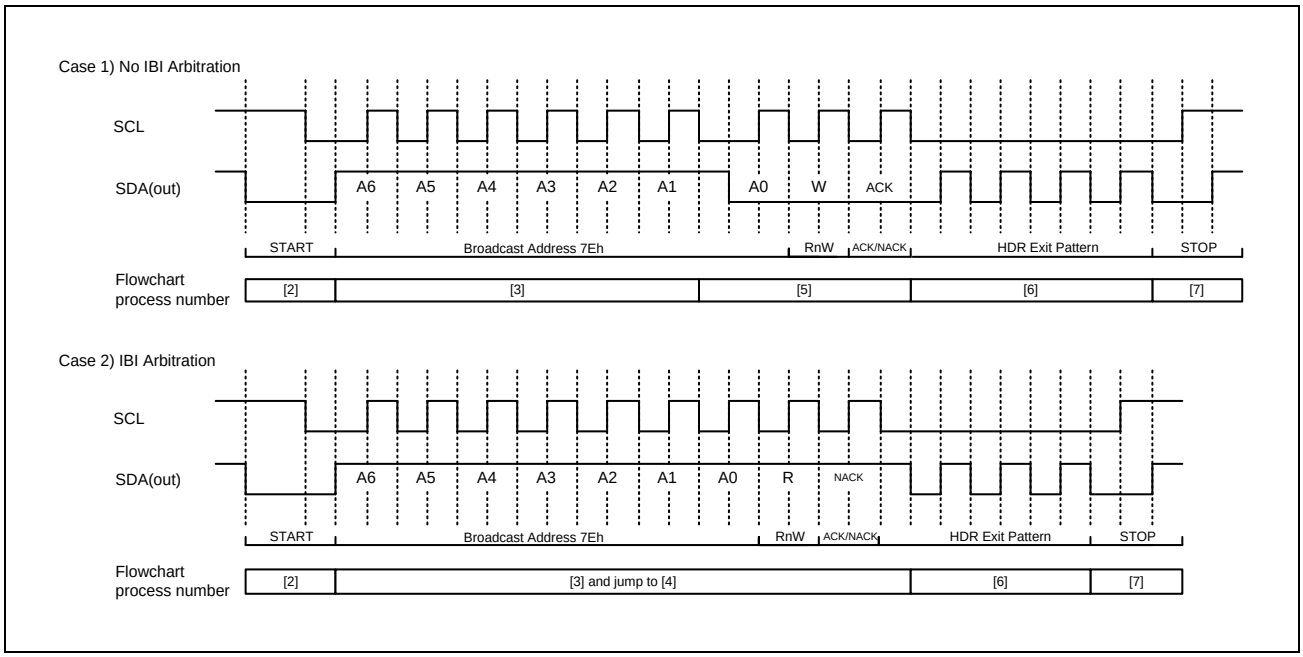


Figure 7.8-102 Escalation Handling timing chart for I3C Master

7.8.4.2.5 Low Power Function [I²C/I3C common]

(1) Wake-Up Function [I²C mode]

This module is equipped with the Wake-up function that causes the microcomputer to transition from low power consumption mode with system clock is stopped (software standby mode, snooze, etc.) to the normal operation. The Wake-up function is used to generate a Wake-Up interrupt signal when the received data matches the address set to Wake-Up interrupt factor also receives data in a state where the operating clock (I3C_0_PCLK/I3C_0_TCLK) is stopped (I3C_0_PCLK/I3C_0_TCLK asynchronous operation). This wake-up interrupt signal causes the microcomputer to transition to the normal operation. After Wake-Up interrupt occurs, switch I3C to I3C_0_PCLK/I3C_0_TCLK synchronous operation, it will be able to continue the communication operation.

The Wake-up function has four wake-up operation modes (normal WU mode 1, normal WU mode 2, command recovery mode, and EEP response mode). The table below describes the behavior in these four wake-up operation modes.

Table 7.8-16 Wake-Up Operation Mode

Mode	ACK Response Timing	ACK Type Responded before Recovery to I3C_0_PCLK/I3C_0_TCLK synchronous operation	SCL State before Recovery to I3C_0_PCLK/I3C_0_TCLK synchronous operation
Normal WU mode 1	Before recovery to I3C_0_PCLK/I3C_0_TCLK synchronous operation*1	ACK	Fixed to “L”
Normal WU mode 2	After recovery to I3C_0_PCLK/I3C_0_TCLK synchronous operation*2	Before recovery: no response (NACK level retained) After recovery: ACK response	Fixed to “L”
Command recovery mode	Before recovery to I3C_0_PCLK/I3C_0_TCLK synchronous operation*1	ACK	Open
EEP response mode	Before recovery to I3C_0_PCLK/I3C_0_TCLK synchronous operation*1	NACK	Open

Note 1. Switching timing from I3C_0_PCLK/I3C_0_TCLK asynchronous operation to I3C_0_PCLK/I3C_0_TCLK Synchronous operation is the fall of the 9th clock of SCL.

Note 2. Switching timing from I3C_0_PCLK/I3C_0_TCLK asynchronous operation to I3C_0_PCLK/I3C_0_TCLK Synchronous operation is the fall of the 8th clock of SCL.

The following can be selected as Wake-Up interrupt factor.

- Host address detection (valid when SVCTL.HOAE=1b)
- General call address detection (valid when SVCTL.GCAE=1b)
- Slave address 0*1 detection (valid when SVCTL.SVAE0=1b)
- Slave address 1*1 detection (valid when SVCTL.SVAE1=1b)
- Slave address 2*1 detection (valid when SVCTL.SVAE2=1b)

Note 1. Only 7-bit address can be set. Please set SDADLS bit to “0b” in SDATBASn.

(a) Normal Wake-Up mode 1

This section describes the behavior, the timing, and a use case of normal WU mode 1.

1. A wake-up interrupt triggered by the match of the slave address makes the transition to the normal operation in the manner described below.

   <Before Wake-Up recovery>
   ACK is sent in response to the data received with its own slave address.

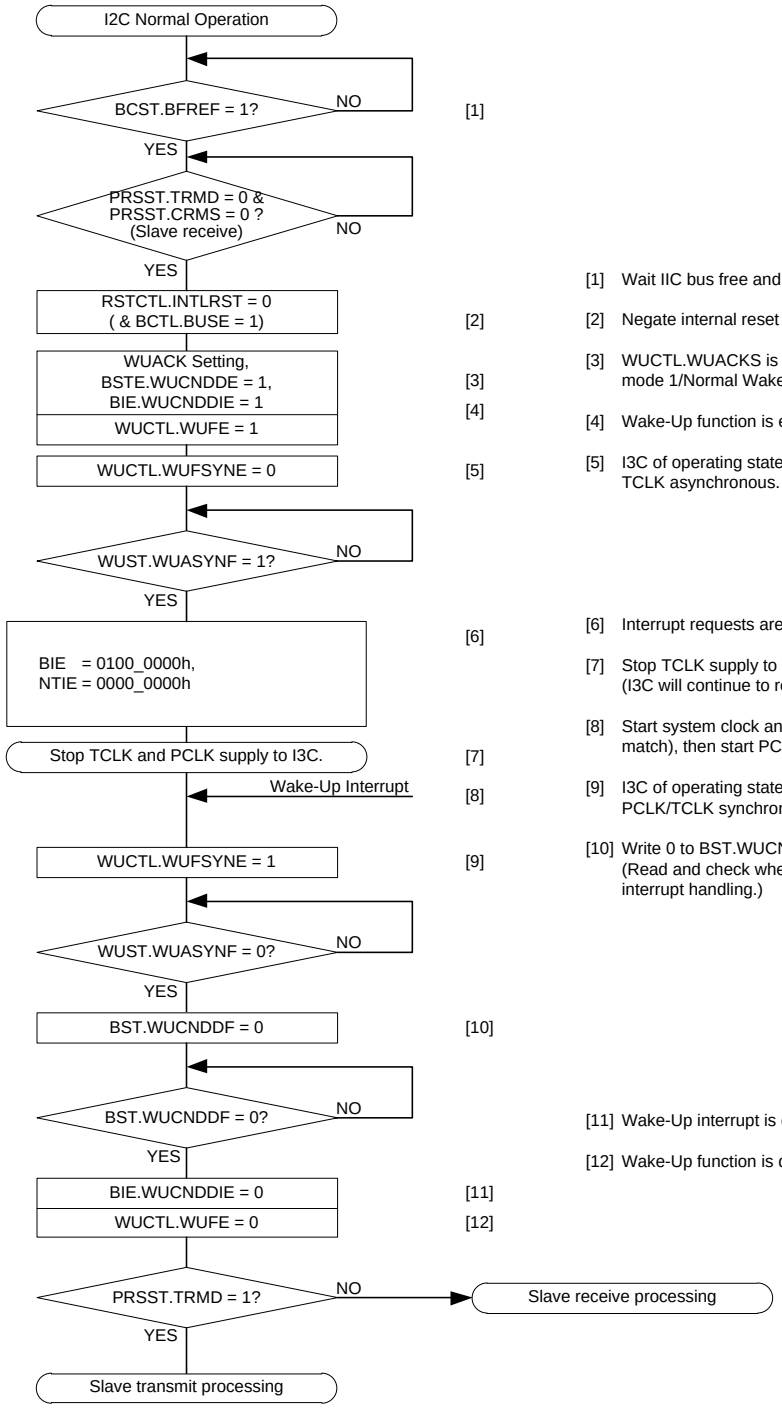
   <During Wake-Up recovery>
   ACK response is made at the 9th clock cycle of SCL, and the SCL is held low afterwards.*1

   <After Wake-Up recovery>
   Normal operation continues.

   If the slave address does not match, the SCL line is not held low after the fall of the 9th clock cycle of SCL, and the slave operation continues. See **Figure 7.8-103** below for a use case.

2. A wake-up interrupt is not generated at the transition to the normal operation if the transition is triggered by a cause (other recovery causes (IRQ)) other than a wake-up interrupt signal generated by a slave address match. BST.WUCNDDF is not set in this case. Carry out the following processing according to **Figure 7.8-104**.

Note 1. Between ninth clock cycle and first clock cycle during Wake-Up recovery, SCSTRCTL.RWE=1 does not work.



[1] Wait IIC bus free and I3C stay in the standby state.

[2] Negate internal reset if assert the internal reset.

[3] WUCTL.WUACKS is set up according to each mode. (Normal Wake-Up mode 1/Normal Wake-Up mode 2) Wake-Up Interrupt is enabled.

[4] Wake-Up function is enabled.

[5] I3C of operating state is changed from PCLK/TCLK synchronous to PCLK/TCLK asynchronous.

[6] Interrupt requests are disabled except WUI.

[7] Stop TCLK supply to I3C, then stop PCLK supply to I3C. (I3C will continue to receive)

[8] Start system clock and TCLK supply to I3C by Wake-Up interrupt (address match), then start PCLK supply to I3C.

[9] I3C of operating state is changed from PCLK/TCLK asynchronous to PCLK/TCLK synchronous.

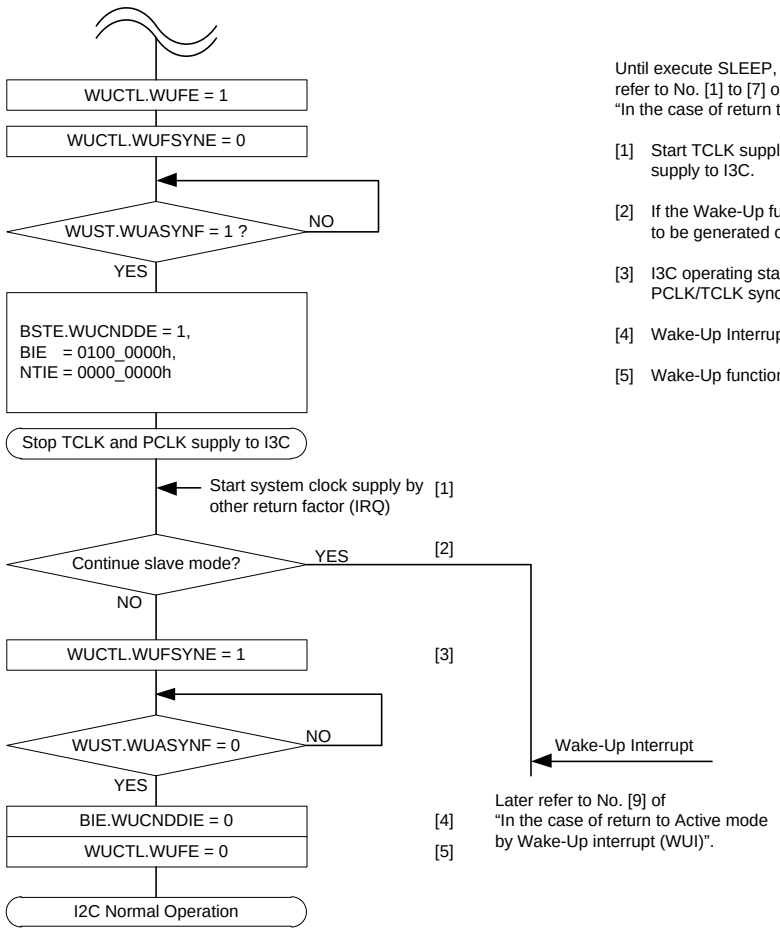
[10] Write 0 to BST.WUCNDDF.
(Read and check whether clearing has been completed befor return from interrupt handling.)

[11] Wake-Up interrupt is disabled.

[12] Wake-Up function is disabled.

Note: TCLK = I3C_0_TCLK
PCLK = I3C_0_PCLK

Figure 7.8-103 Use Case of Normal WU Mode 1 (Wake-up recovery by a wake-up interrupt triggered by the match of the slave address)



Until execute SLEEP, it is the same method,
refer to No. [1] to [7] of
“In the case of return to Active mode by Wake-Up interrupt (WUI)”.

[1] Start TCLK supply to I3C by other return factor (IRQ), then start PCLK supply to I3C.

[2] If the Wake-Up function is used continuously by the slave mode, wait to be generated of the Wake-Up interrupt request.

[3] I3C operating state is changed from PCLK/TCLK asynchronous to PCLK/TCLK synchronous.

[4] Wake-Up Interrupt is disabled.

[5] Wake-Up function is disabled.

Note: TCLK = I3C_0_TCLK
PCLK = I3C_0_PCLK

Figure 7.8-104 Use Case of Normal WU Modes 1 and 2 (Wake-up recovery by other recovery causes (IRQ))

(b) Normal Wake Up Mode2

This section describes the behavior, the timing, and a use case of normal WU mode 2.

1. A wake-up interrupt triggered by the match of the slave address makes the transition to the normal operation in the manner described below.

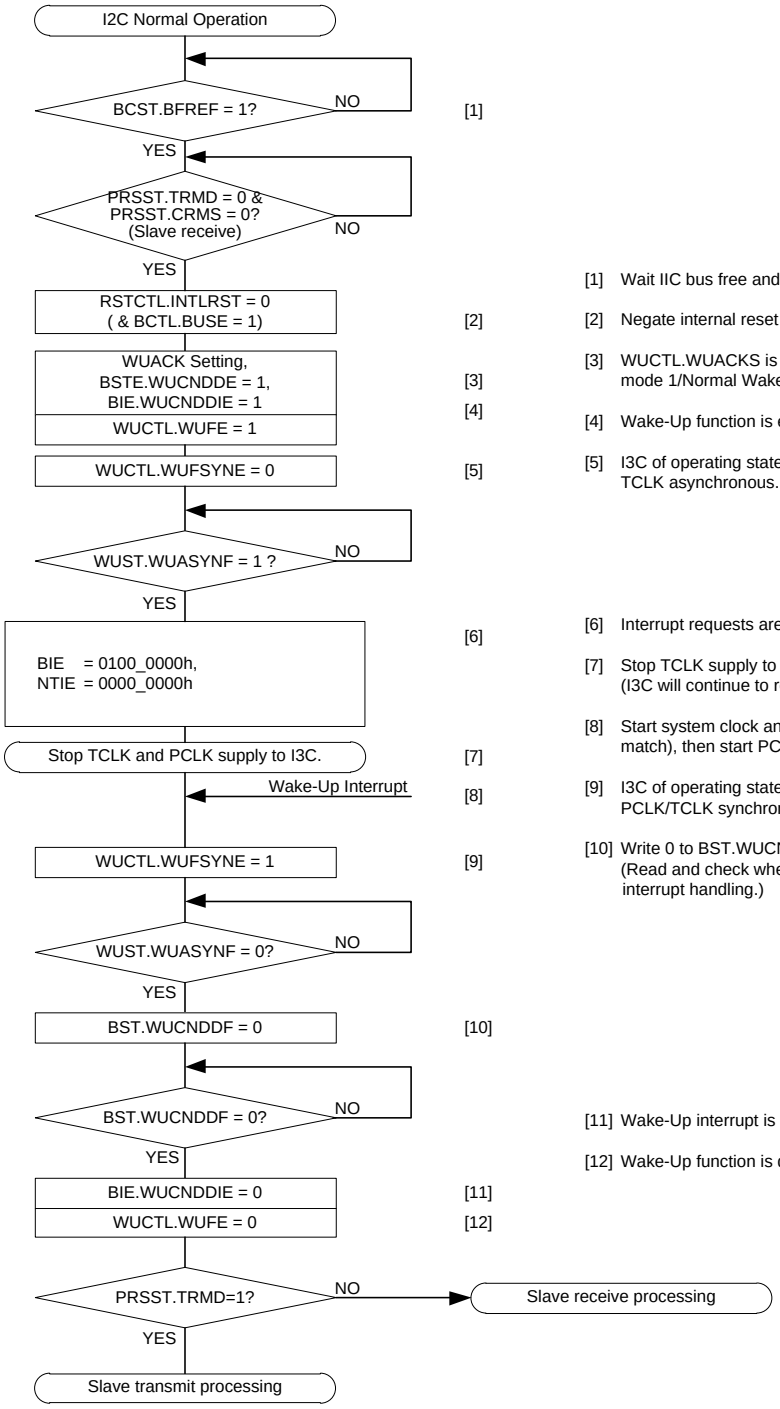
   <Before Wake-Up recovery>
   No response to the data received with its own slave address (until 8th SCL cycle end)

   <During Wake-Up recovery>
   Holding the SCL line low during the 8th and 9th clock cycles

   <After Wake-Up recovery>
   Returning ACK at the 9th clock cycle of SCL, and continuing the normal operation

   If the slave address does not match, the SCL line is not held low after the fall of the 8th SCL clock cycle. The slave operation continues. See **Figure 7.8-105** below for a use case.

2. A wake-up interrupt is not generated at the transition to the normal operation if the transition is triggered by a cause (other recovery causes (IRQ)) other than a wake-up interrupt signal generated by a slave address match. BST.WUCNDDF is not set in this case. Carry out the following processing according to **Figure 7.8-104**.



[1] Wait IIC bus free and I3C stay in the standby state.

[2] Negate internal reset if assert the internal reset.

[3] WUCTL.WUACKS is set up according to each mode. (Normal Wake-Up mode 1/Normal Wake-Up mode 2) Wake-Up Interrupt is enabled.

[4] Wake-Up function is enabled.

[5] I3C of operating state is changed from PCLK/TCLK synchronous to PCLK/TCLK asynchronous.

[6] Interrupt requests are disabled except WUI.

[7] Stop TCLK supply to I3C, then stop PCLK supply to I3C. (I3C will continue to receive)

[8] Start system clock and TCLK supply to I3C by Wake-Up interrupt (address match), then start PCLK supply to I3C.

[9] I3C of operating state is changed from PCLK/TCLK asynchronous to PCLK/TCLK synchronous.

[10] Write 0 to BST.WUCNDDF.
(Read and check whether clearing has been completed befor return from interrupt handling.)

[11] Wake-Up interrupt is disabled.

[12] Wake-Up function is disabled.

Note: TCLK = I3C_0_TCLK
PCLK = I3C_0_PCLK

Figure 7.8-105 Use Case of Normal WU Mode 2 (Wake-up recovery by a wake-up interrupt triggered by the match of the slave address)

(c) Command recovery mode/ EEP response mode (Special Wake Up mode)

In the command recovery mode and EEP response mode, the SCL line is not held low during the wake-up recovery period (after the rise of the 9th clock cycle of SCL), so other I^2C/I3C devices can use the I^2C bus during this period. This section describes the behavior, the timing, and use cases of the command recovery mode and the EEP response mode.

1. A wake-up interrupt triggered by the match of the slave address makes the transition to the normal operation in the manner described below.

   <Before Wake-Up recovery>
   In response to the data received with its own slave address, ACK (command recovery mode) or NACK (EEP response mode) is returned.

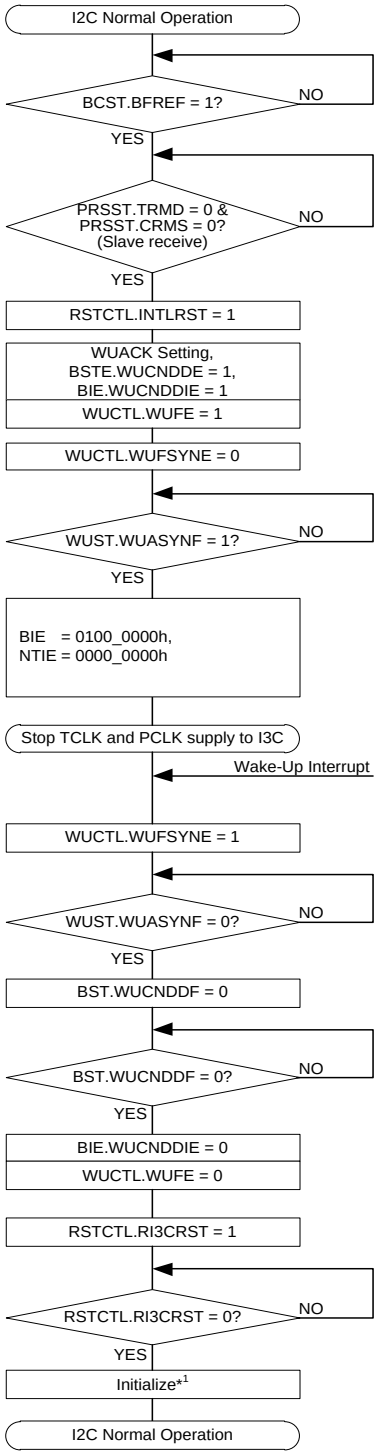
   <During Wake-Up recovery>
   The SCL line is not held low.

   <After Wake-Up recovery>
   Normal operation continues after I3C initial setting. If the slave address does not match, the slave operation continues. See **Figure 7.8-106** below for a use case.

Remark 1. Because the SCL line is not held low during wake-up recovery, the transmission/reception of the data that follows the slave address is not possible.

Remark 2. The command recovery mode and the EEP response mode are internal reset (RSTCTL.INTLRST= 1b) states. Therefore, the match of the slave address does not set the SVST flags (HOAF, GCAF, and SVAF2, SVAF1, SVAF0).

2. A wake-up interrupt is not generated at the transition to the normal operation if the transition is triggered by a cause (other recovery causes (IRQ)) other than a wake-up interrupt signal generated by a slave address match. BST.WUCNDDF is not set in this case. Carry out the following processing according to **Figure 7.8-107**.



[1] Waite IIC bus free and I3C stay in the standby state.

[2] Assert the internal reset.

[3] WUCTL.WUACKS is set up according to each mode. (Command return mode/EEP response mode) Wake-Up Interrupt is enabled.

[4] Wake-Up function is enabled.

[5] I3C of operating state is changed from PCLK/TCLK synchronous to PCLK/TCLK asynchronous.

[6] Interrupt requests are disabled except WUI.

[7] Stop TCLK supply to I3C, then stop PCLK supply to I3C. (I3C will continue to receive)

[8] Start system clock and TCLK supply to I3C by Wake-Up interrupt (address match), then start PCLK supply to I3C.

[9] I3C of operating state is changed from PCLK/TCLK asynchronous to PCLK/TCLK synchronous.

[10] Write 0 to BST.WUCNDDF. (Read and check whether clearing has been completed befor return from interrupt handling.)

[11] Wake-Up interrupt is disabled.

[12] Wake-Up function is disabled.

[13] I3C software reset.

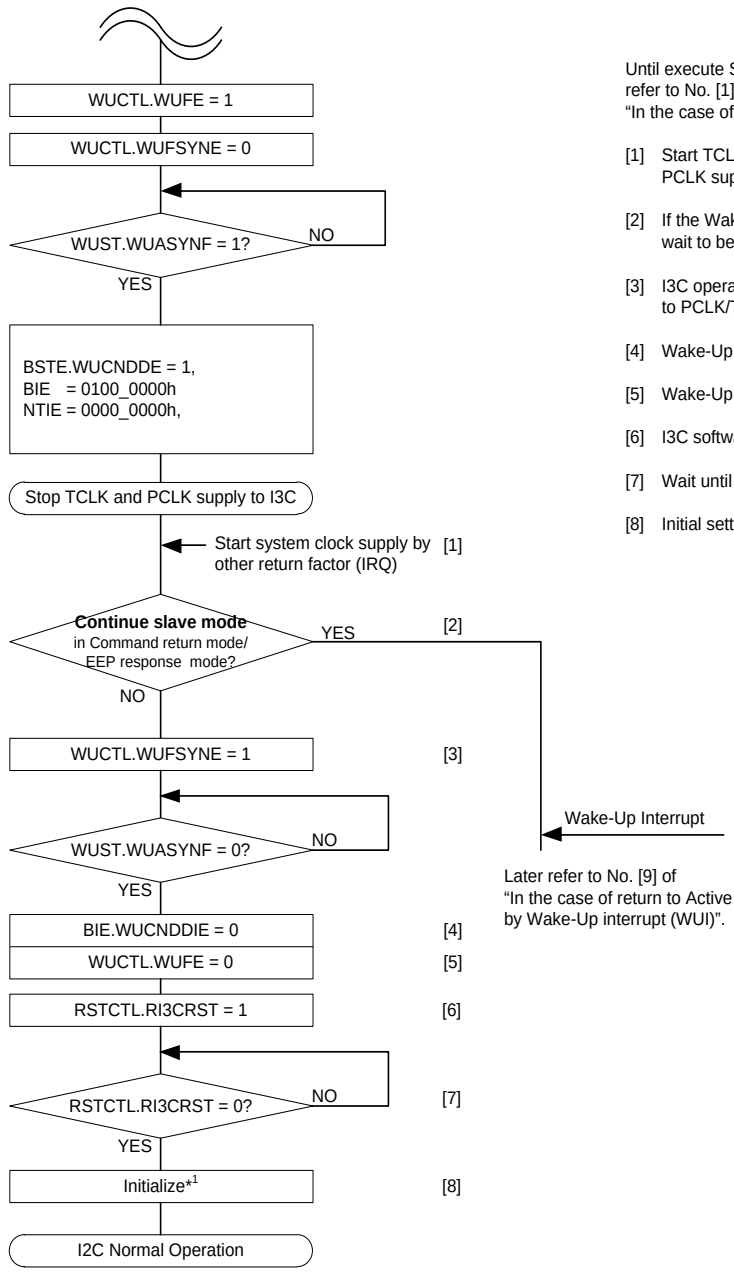
[14] Wait until RSTCTL.RI3CRST becomes 0.

[15] Initial settings

Note: TCLK = I3C_0_TCLK
PCLK = I3C_0_PCLK

Note 1. Refer to **7.8.4.3.1 Initial Setting Flow** for details of initialization.

Figure 7.8-106 Use Case of Command Recover Mode and EEP Response Mode (Wake-up recovery by a wake-up interrupt triggered by the match of the slave address)



Until execute SLEEP, it is the same method,
refer to No. [1] to [7] of
“In the case of return to Active mode by Wake-Up interrupt (WUI)”.

[1] Start TCLK supply to I3C by other return factor (IRQ), then start PCLK supply to R-I3C.

[2] If the Wake-Up function is used continuously by the slave mode, wait to be generated of the Wake-Up interrupt request.

[3] I3C operating state is changed from PCLK/TCLK asynchronous to PCLK/TCLK synchronous.

[4] Wake-Up Interrupt is disabled.

[5] Wake-Up function is disabled.

[6] I3C software reset.

[7] Wait until RSTCTL.RI3CRST becomes 0.

[8] Initial settings

Note: TCLK = I3C_0_TCLK
PCLK = I3C_0_PCLK

Note 1. Refer to **7.8.4.3.1 Initial Setting Flow** for details of initialization.

Figure 7.8-107 Use Case of Command Recover Mode and EEP Response Mode (Wake-up recovery by other recovery causes (IRQ))

(d) Precautions on the use of the Wake-Up function.

Precautions on the use of the Wake-Up function are shown below.

1. Do not change the registers in the I3C except the WUCTL.WUFSYNE bit while the WUST.WUASYNF flag = 1b (during I3C_0_PCLK/I3C_0_TCLK asynchronous operation).
2. Set WUCTL.WUFE = BSTE.WUCNDDE = BIE.WUCNDDIE = 1b and PRSST.CRMS = PRSST.TRMD = 0b (slave reception mode) before switching I3C_0_PCLK/I3C_0_TCLK asynchronous mode.
3. You cannot select the device ID and the 10-bit slave address for Wake-Up interrupt factor. Set the DVIDE bit in SVCTL and SDADLS bit in SDATBASn to 0b.
4. Set all bits in BIE(TENDIE, NACKDIE, SPCNDDIE, STCNDDIE, ALIE, TODIE) and TDBEIE and RDBFIE bits in HTIE to 0b (Interrupt disabled) before switching the asynchronous operation.
5. Do not use the timeout function while the Wake-up function is enabled (WUCTL.WUFE=1b).
6. Wake-up interrupt is generated during I3C_0_PCLK/I3C_0_TCLK asynchronous operation (when WUST.WUASYNF=1b). In case of detecting slave address matching in I3C_0_PCLK/I3C_0_TCLK synchronous mode (WUST.WUASYNF=0), Wake-Up interrupt does not occur, and BST.WUCNDDF flag will not be set.
7. In I3C_0_PCLK/I3C_0_TCLK synchronous operation mode, the I3C may initiate the next reception upon detecting a conflict between the timing of writing “0b” to the WUCTL.WUFSYNE bit and the START condition. In this case, when data communication ends and a STOP condition is detected, the WUST.WUASYNF flag becomes 1b (I3C_0_PCLK/I3C_0_TCLK asynchronous mode transition) and Wake-Up event detection starts.
8. If you want to switch from I3C_0_PCLK/I3C_0_TCLK asynchronous operation to I3C_0_PCLK/I3C_0_TCLK synchronous operation without address match detection, it will switch on the STOP condition detection. When the WUCTL.WUFSYNE bit is set to 1 in a bus free state, I3C_0_PCLK/I3C_0_TCLK asynchronous operation continues (Reception operation: waiting communication frame). WUST.WUASYNF flag becomes to 0b when I3C detects the STOP condition of the next communication frame, and I3C switches to I3C_0_PCLK/I3C_0_TCLK synchronous operation.
9. After writing 0 to WUFSYNE bit in WUCTL, do not change I3C operation mode setting register (BFCTL, SCSTRCTL, ACKCTL, INCTL, SVCTL, SDATBASn) until switched to the I3C_0_PCLK/I3C_0_TCLK asynchronous operation from I3C_0_PCLK/I3C_0_TCLK synchronous operation (while WUST.WUASYNF flag = 1b). If register value changes by the interrupt processing etc. in this period, the I3C might malfunction without succeeding to the setting to the asynchronous operation.
10. During I3C_0_PCLK/I3C_0_TCLK asynchronous operation (WUST.WUASYNF = 1b), do not refer to each flag of SVST, BST, HTST register and BCST.BFREF flag.
11. Do not set ACKCTL.ACKT = 1b in order to make an ACK response in the synchronization unit when Wake-Up is performed by slave address match in Normal Wake-Up Mode 2.

(2) Wake-Up Function [I3C mode]

(a) I3C Master Wake-Up

Wake-Up interrupt causes of I3C Master are shown below.

- SDA Low detection (IBI request from I3C Slave)

See below for details on the master Wake-Up operation.

1. **<Before Wake-Up recovery>**
   SDA Low Drive is detected and the INTWU interrupt is asserted.

2. **<During Wake-Up recovery>**
   Keep SCL Line High.

3. **<After Wake-Up recovery>**
   Drive SCL Low and complete START condition. SCL is supplied on the I3C Bus and IBI from I3C Slave is received.

Remark 1. If active mode (normal operation) is entered due to other factors, disable the Wake-Up function as necessary.

Remark 2. After confirming PRSTDBG.SDILV = 1b, set WUCTL.WUFE = 0b.

Remark 3. Do not use the timeout function while the Wake-up function is enabled (WUCTL.WUFE=1b).

(b) I3C Slave Wake-Up

Wake-Up interrupt causes of I3C Slave are shown below.

- Broadcast Address (7Eh) and detection of its own Slave Address match

See below for details on the slave Wake-Up operation.

1. **<Before Wake-Up recovery>**
   STEP 1:
   If I3C detects BA (7Eh/W) following a START (or Repeated START) Condition, then I3C shall generates ACK (after 7Eh/W).

   STEP 2:
   If I3C detects its own Dynamic Address after a Repeated START condition following Step1,
   then I3C shall generates NACK (after its own Dynamic Address) and then issues a INTWU interrupt.

   <During Wake-Up recovery>
   I3C always generates NACK.

   <After Wake-Up recovery>
   Normal operation continues.

Remark 1. If active mode (normal operation) is entered due to other factors, disable the Wake-Up function as necessary.

Remark 2. Do not use the timeout function while the Wake-up function is enabled (WUCTL.WUFE=1b).

7.8.4.2.6 Other

(1) SCL Synchronization Circuit [I²C mode]

This function is enabled while the PRTS.PRTMD bit is set to 1.

In generation of the SCL clock, I3C starts counting out the value for width at high level specified in STDBR.SBRHO[7:0] when it detects a rising edge on the SCL line and drives the SCL line low once counting of the width at high level is complete.

When I3C detects the falling edge of the SCL line, it starts counting out the width at low level period specified in STDBR.SBRLO[7:0], and then stops driving the SCL line (releases the line) once counting of the width at low level is complete. The SCL clock is thus generated.

If multiple master devices are connected to the I²C bus, a collision of SCL signals may arise due to contention with another master device. In such cases, the master devices have to synchronize their SCL signals. Since this synchronization of SCL signals must be bit by bit, I3C is equipped with a facility (the SCL synchronization circuit) to obtain bit-by-bit synchronization of the SCL clock signals by monitoring the SCL line while in master mode.

When I3C has detected a rising edge on the SCL line and thus started counting out the width at high level specified in STDBR.SBRHO[7:0], and the level on the SCL line falls because an SCL signal is being generated by another master device, I3C stops counting when it detects the falling edge, drives the level on the SCL line low, and starts counting out the width at low level specified in STDBR.SBRLO[7:0]. When I3C finishes counting out the width at low level, it stops driving the SCL line to the low level (releases the line). At this time, if the width at low level of the SCL clock signal from the other master device is longer than the width at low level set in this module, the width at low level of the SCL signal will be extended. Once the width at low level for the other master device has ended, the SCL signal rises because the SCL line has been released. When I3C finishes outputting the low-level period of the SCL clock, the SCL line is released and the SCL clock rises. That is, in cases of contention of SCL signals from more than one master, the width at high level of the SCL signal is synchronized with that of the clock having the narrower width, and the width at low level of the SCL signal is synchronized with that of the clock having the broader width. However, such synchronization of the SCL signal is only enabled when the SCSYNE bit in BFCTL is set to 1b.

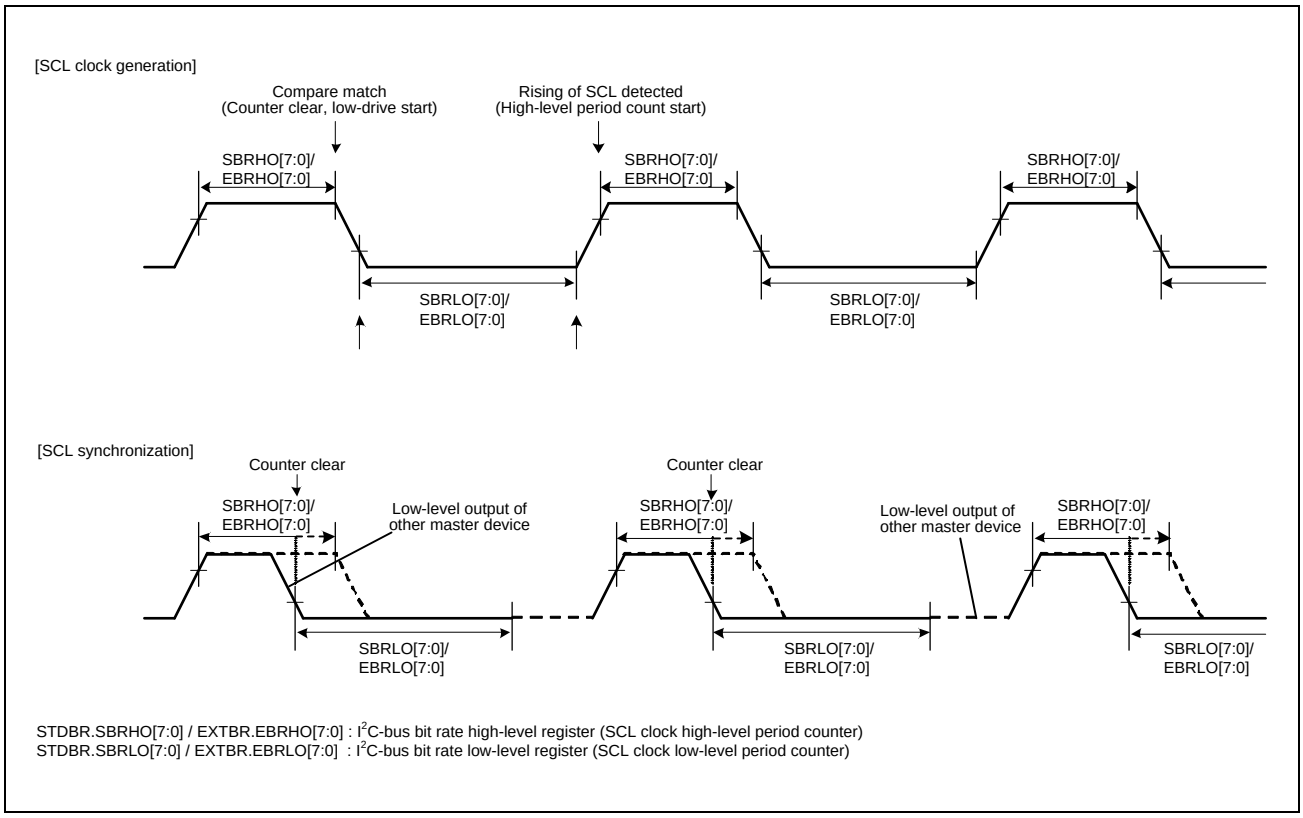


Figure 7.8-108 Generation and synchronization of the SCL signal

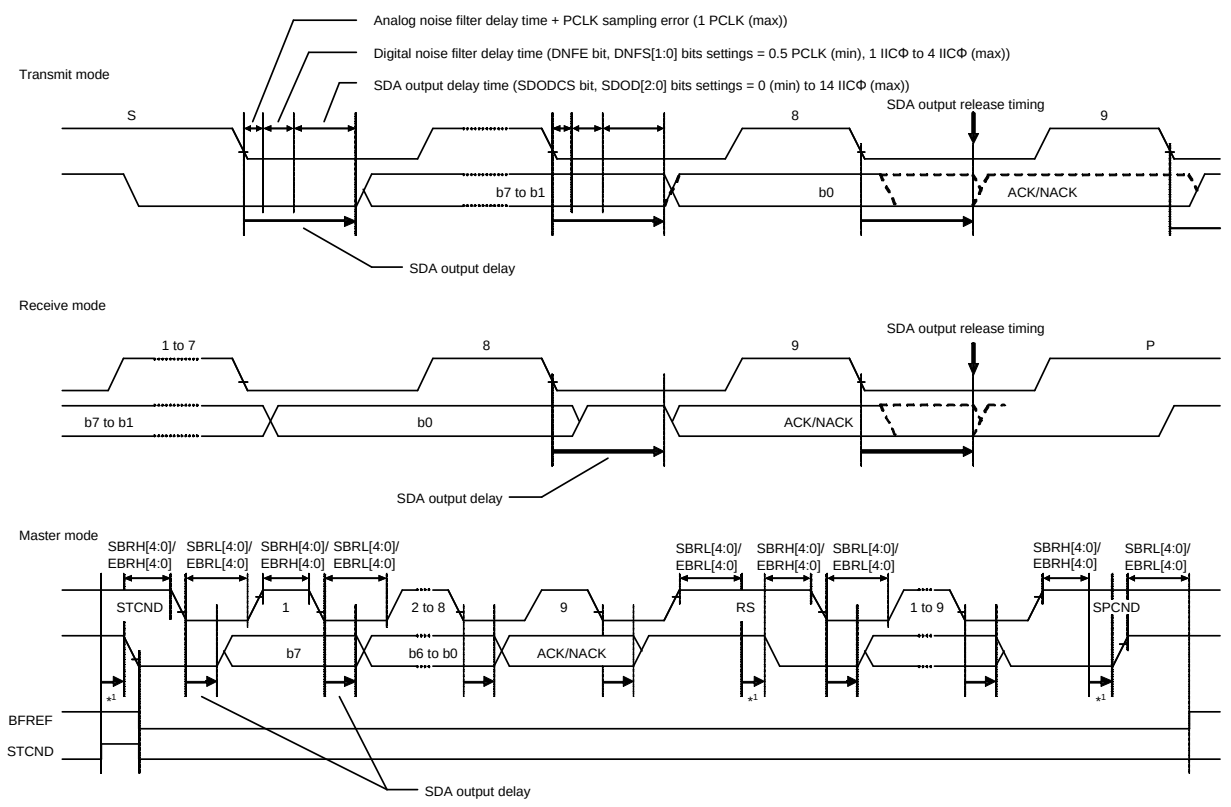
(2) Facility for Delaying SDA Output [I²C mode]

I3C module incorporates a facility for delaying output on the SDA line. The delay can be applied to all output (issuing of the START, Repeated START, and STOP conditions, data, and the ACK and NACK signals) on the SDA line.

With the SDA output delay facility, SDA output is delayed from detection of a falling edge of the SCL signal to ensure that the SDA signal is output within the interval over which the SCL clock is at the low level. Doing this leads to usage with the aim of preventing erroneous operation of communications devices, with the aim of satisfying the 300 ns (minimum) data- hold time requirement of the SMBus specification.

The output delay facility is enabled by setting the SDOD[2:0] bits in OUTCTL to any value other than 000b, and disabled by setting the same bits to 000b.

While the SDA output delay facility is enabled (while the SDOD[2:0] bits in OUTCTL are set to any value other than 000b), the SDODCS bit in OUTCTL selects the clock source for counting by the SDA output delay counter as the internal base clock (I3Cϕ) for I3C module or as a clock signal derived by dividing the frequency of the internal base clock by two (I3Cϕ/2). The counter counts the number of cycles set in the SDOD[2:0] bits in OUTCTL. After counting of the set number of cycles of delay is completed, I3C module places the required output (START, Repeated START, or STOP condition, data, or an ACK or NACK signal) on the SDA line.



Note: PCLK = I3C_0_PCLK

Note 1. The output delay function is set by the SDODCS and SDOD[2:0] bits when a start (STCND), repeated start (SRCND), or stop (SPCND) condition is issued.

Figure 7.8-109 SDA output delay facility

(3) Digital Noise-Filter Circuits [I²C mode]

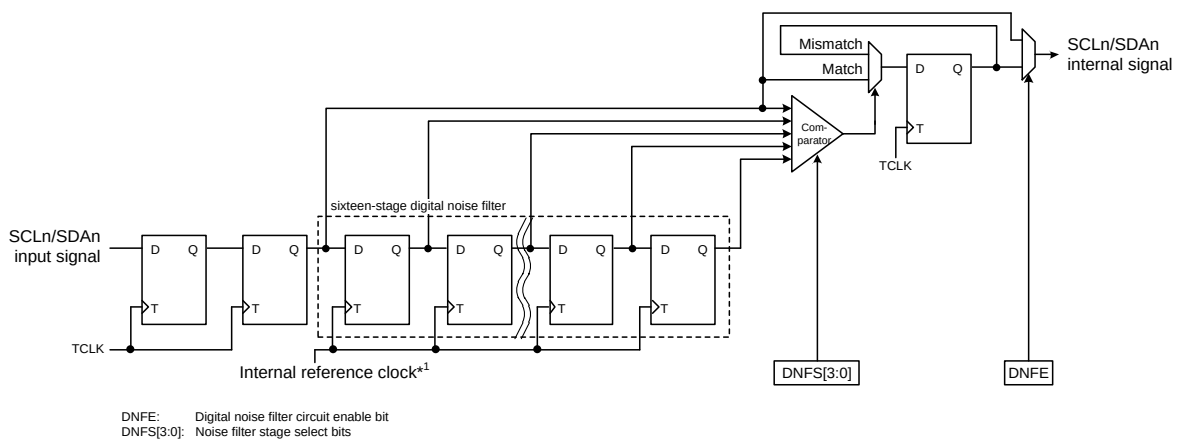
The states of the SCL and SDA pins are conveyed to the internal circuitry through digital noise-filter circuits. **Figure 7.8-110** is a block diagram of the digital noise-filter circuit.

The on-chip digital noise-filter circuit of I3C consists of 16 flip-flop circuit stages connected in series and a match detection circuit. During the Hs-mode transfer, only the first four flip-flop circuits stages are enabled.

The number of effective stages in the digital noise filter is selected by the INCTL.DNFS[3:0] bits. The selected number of effective stages determines the noise-filtering capability as a period from one to sixteen I3Cϕ cycles.

The input signal to the SCL pin (or SDA pin) is sampled on rising edges of the I3Cϕ signal. When the input signal level matches the output level of the number of effective flip-flop circuit stages as selected by the INCTL.DNFS[3:0] bits, the signal level is conveyed to the subsequent stage. If the signal levels do not match, the previous value is retained.

If the ratio between the frequency of the internal operating clock (I3C_0_TCLK) and the transfer rate is small (For example, data transfer at 400 kbps with I3C_0_TCLK = 4 MHz), the characteristics of the digital noise filter may lead to the elimination of needed signals as noise.



Note: TCLK = I3C_0_TCLK

Note 1. Internal reference clock = I3Cϕ

Figure 7.8-110 Block diagram of digital noise filter circuit

7.8.4.3 Operation

7.8.4.3.1 Initial Setting Flow

(1) I^2C Initial Setting Flow (Single Buffer Transfer)

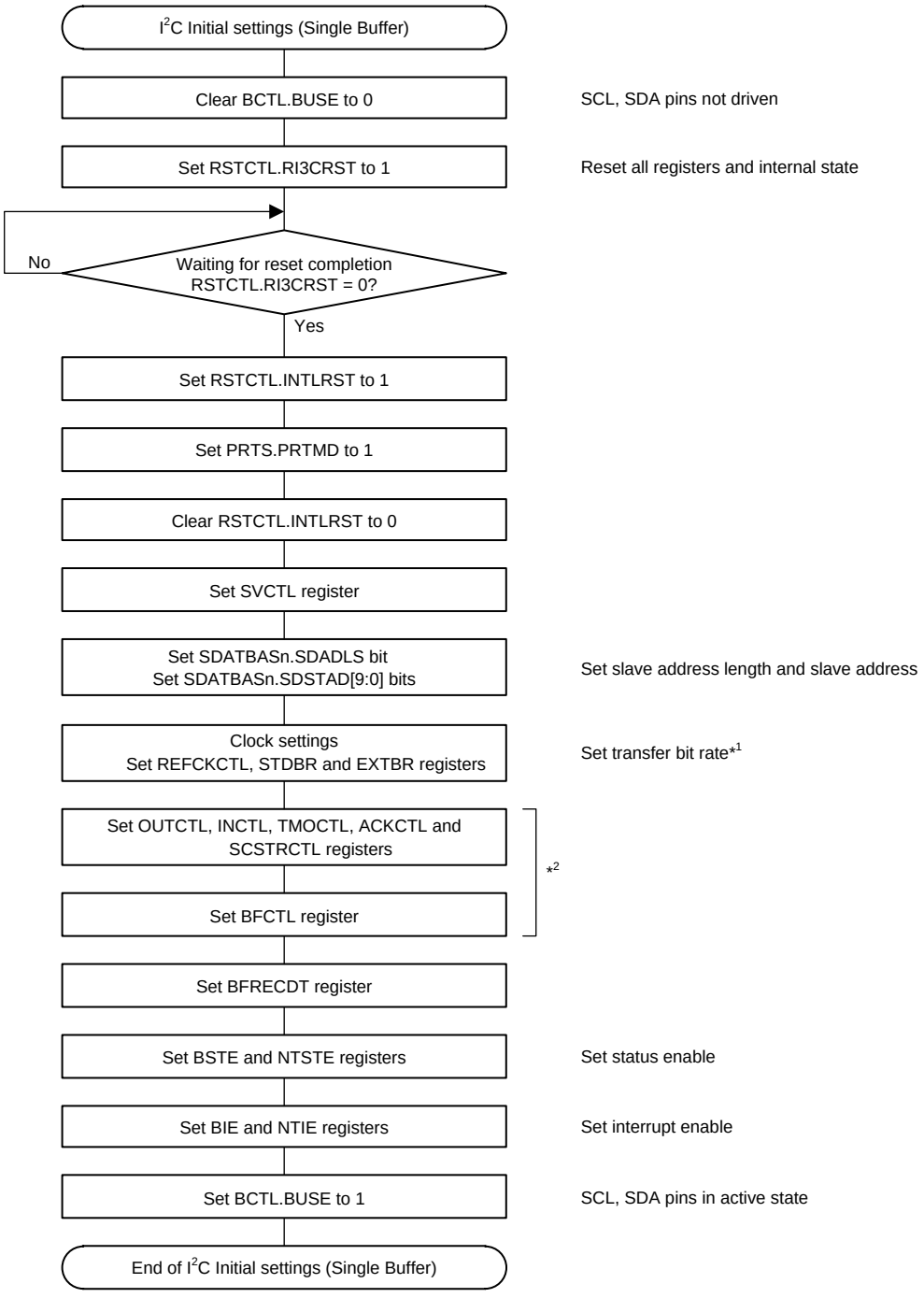
Before starting data transmission and reception, initialize I3C according to the procedure in **Figure 7.8-111**.

First, set the BCTL.BUSE bit to 0b (SCL, SDA pins not driven).

Next, set the RSTCTL.RI3CRST bit to 1b (I3C reset). This initializes all registers and the internal state. Then, waits for RI3CRST to become 0b.

This initializes the various flags and some registers. See **7.8.7 Reset Descriptions**.

After that, set registers SDATBASn.SDADLS, SDATBASn.SDSTAD[9:0], STDBR, INCTL, OUTCTL, TMOCTL, SCSTRCTL, ACKCTL, and BFCTL, then set the other registers as necessary (for initial settings of I3C, see **Figure 7.8-111**).



Note: n = 0 to 2

Note 1. When the I3C is used only in slave mode, set the STDBR register to a value longer than the data setup time.

Note 2. Set these registers as necessary.

Figure 7.8-111 Example of I²C Initialization Flowchart (Single Buffer Transfer)

(2) I3C Initial Setting Flow

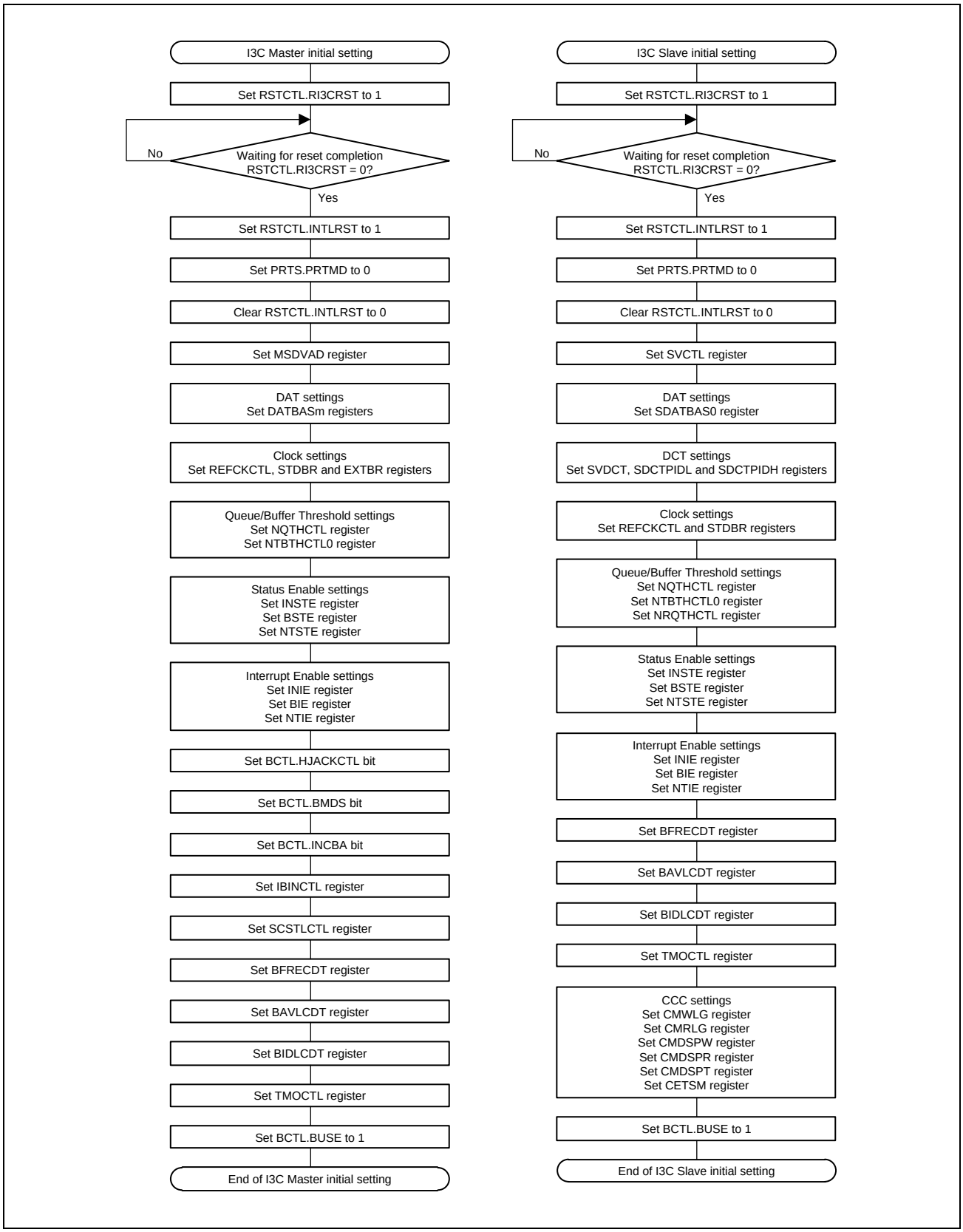


Figure 7.8-112 Example of I3C initialization flowchart

7.8.4.3.2 I3C Communication Flow

Figure 7.8-113 illustrates how I3C communication is initiated:

- All I3C communication occurs within a frame. The frame begins with a START, followed by one or more transfers, and a STOP.

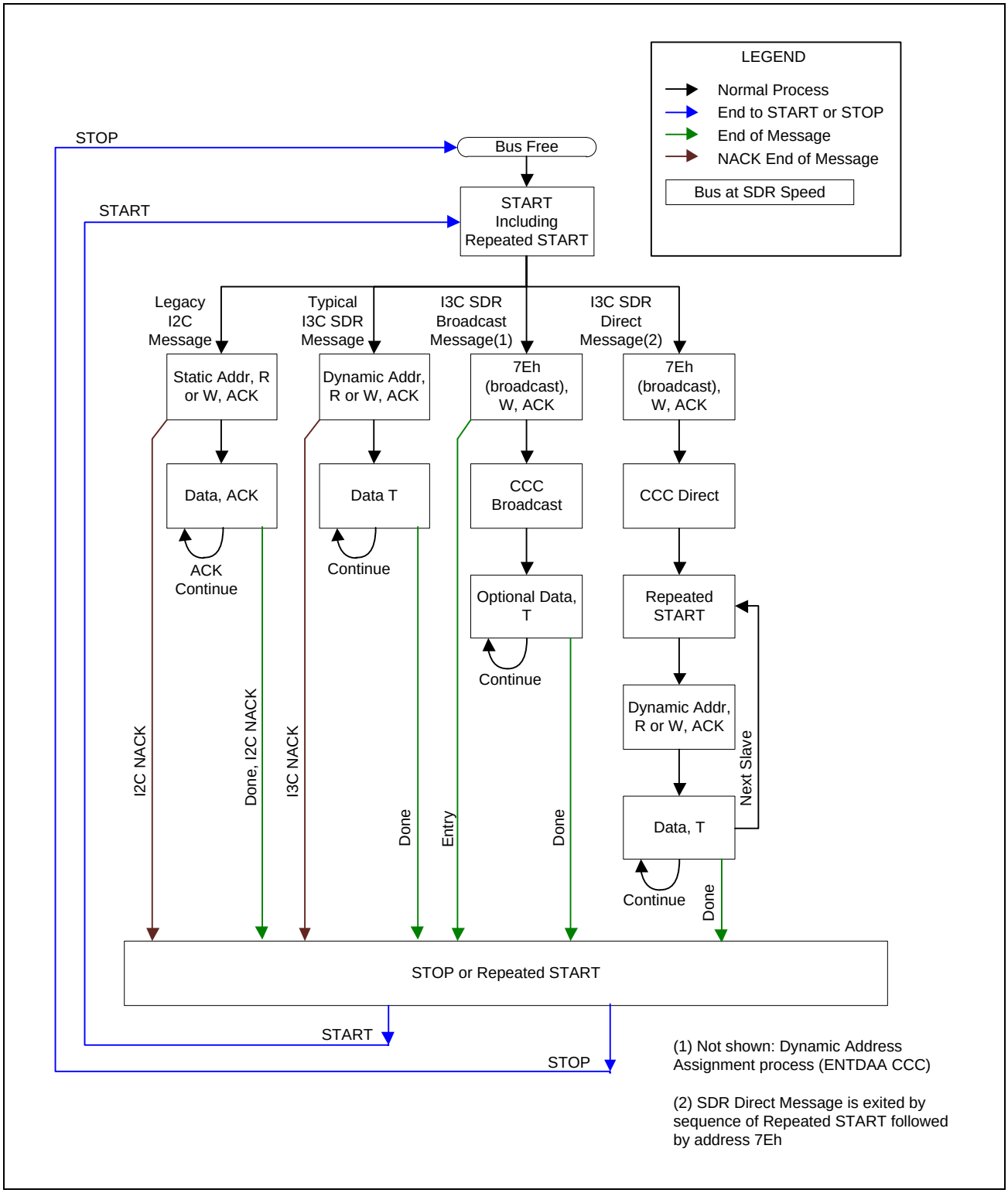


Figure 7.8-113 I3C communication flow

I3C is based on a frame encapsulation approach. A frame includes a data payload. The transfer protocol for the data payload is either SDR. Frames are bordered by I^2C-like bus management.

The I3C frame always includes at least the START, the Header, the Data, and the STOP. The Header following a START allows for Bus Arbitration. The Master uses the Header to address Slave device (s). Slave devices (s) may use the Header Arbitration for multiple purposes: for In-Band Interrupt, for Hot-Join, and for Secondary Master functionality.

I3C allows only one Master to have control of the I3C bus at a time. Mechanisms for handoff of the Master role from one device to another device are provided.

7.8.4.3.3 Master Mode Communication Flow

(1) I²C Master Transmission Flow (Single Buffer Transfer)

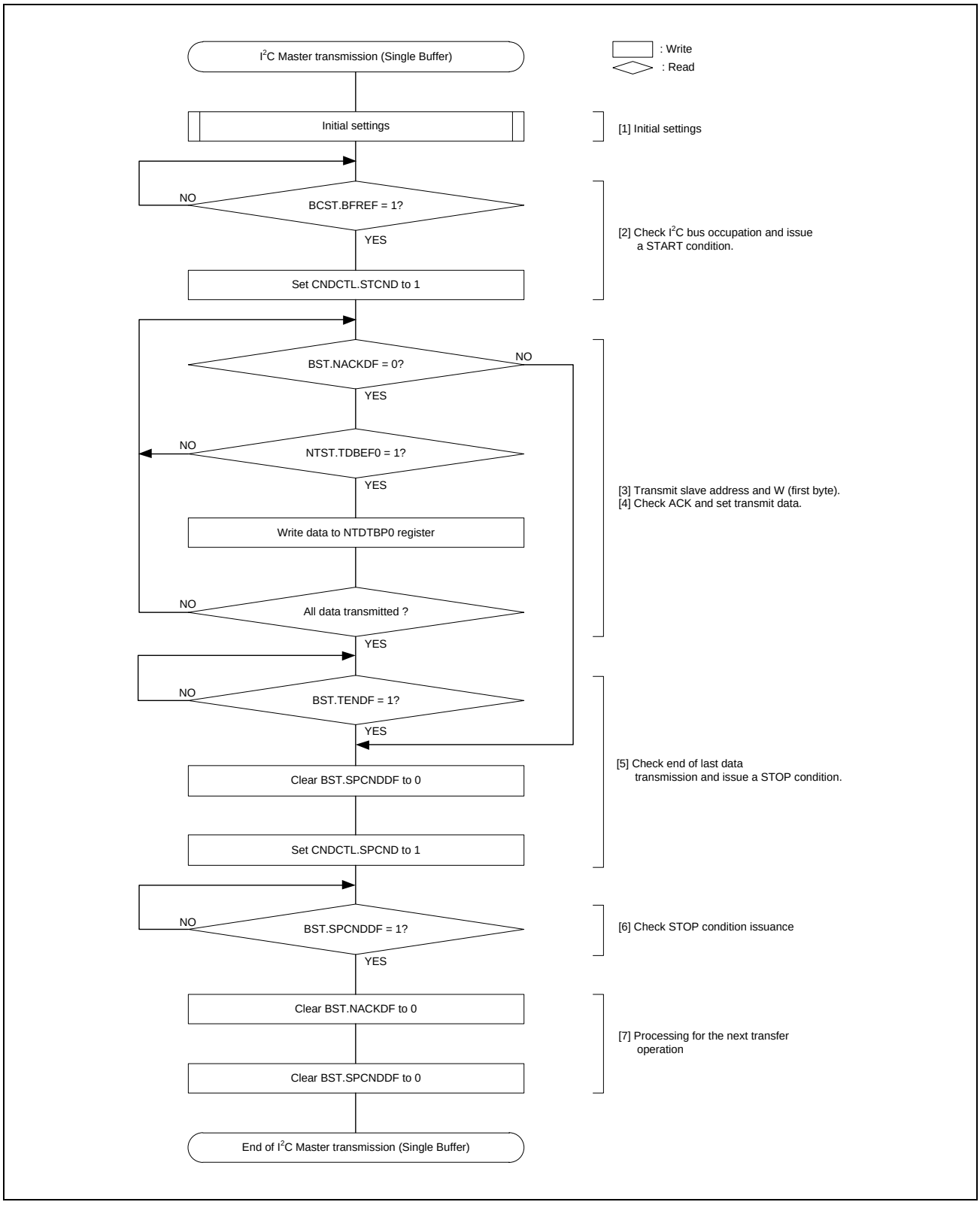


Figure 7.8-114 Example of I²C master transmission flowchart (single buffer transfer)

(2) I²C Master Reception Flow (Single Buffer Transfer)

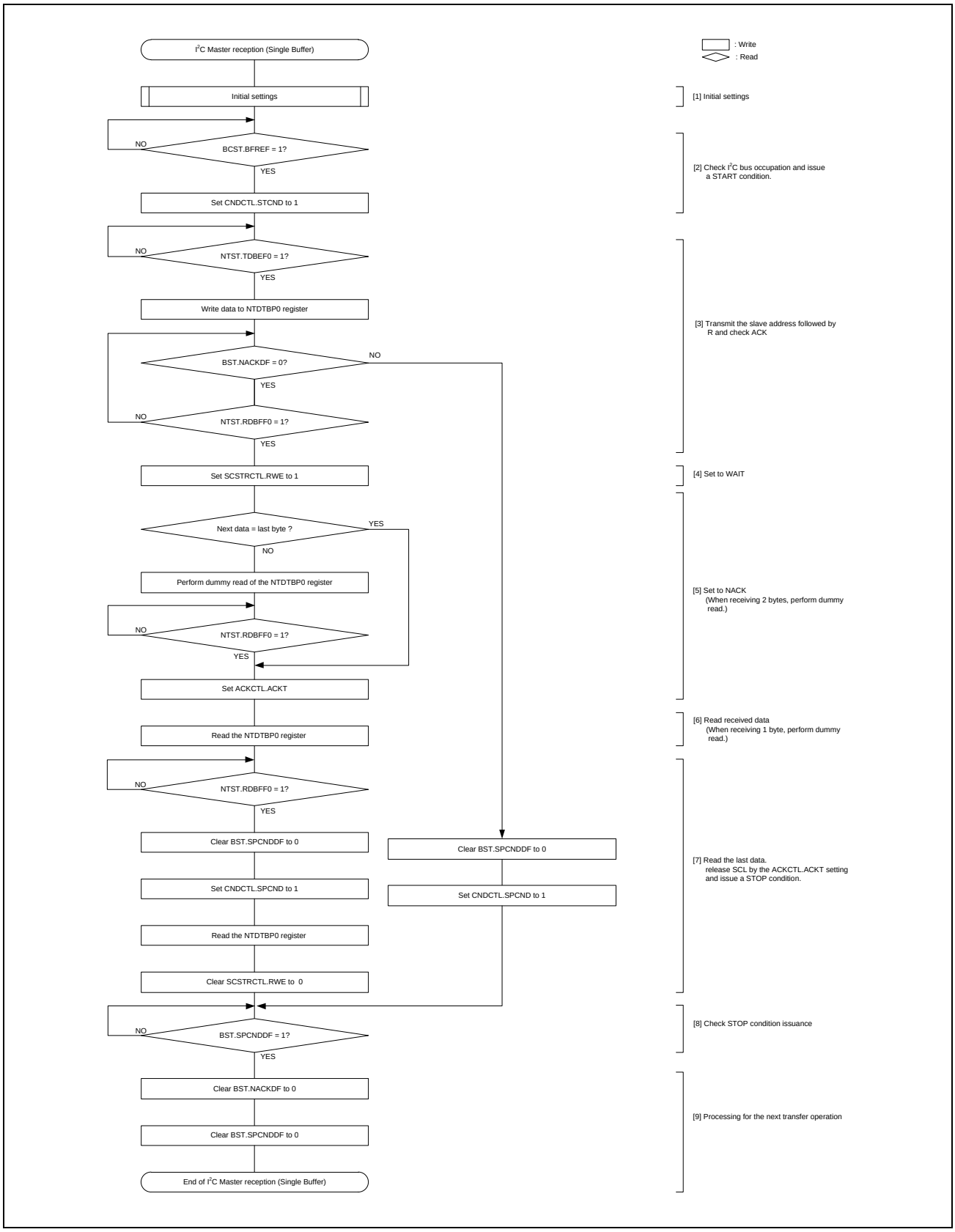


Figure 7.8-115 Example of I²C master reception flowchart (7-bit address format, 1 or 2 bytes)

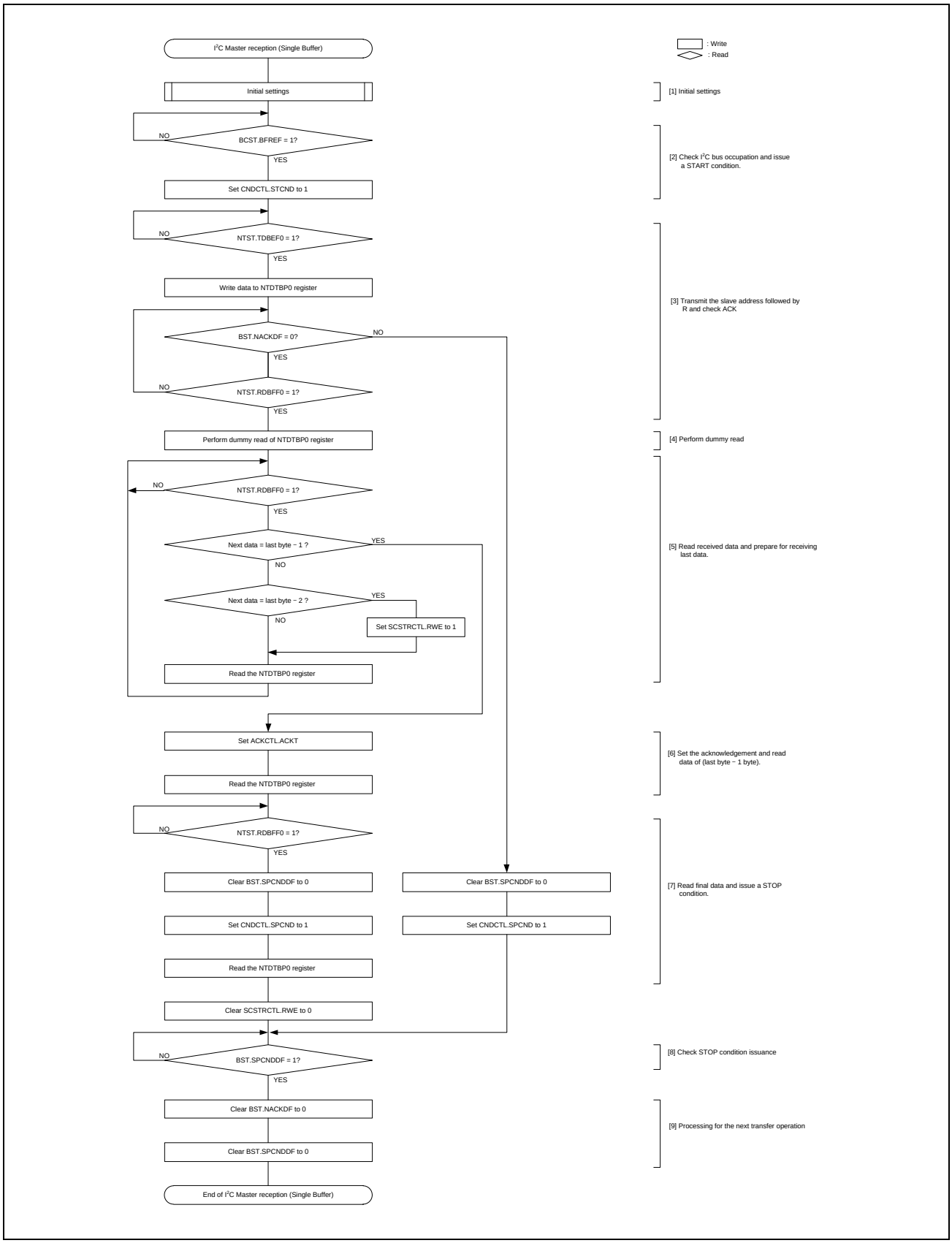
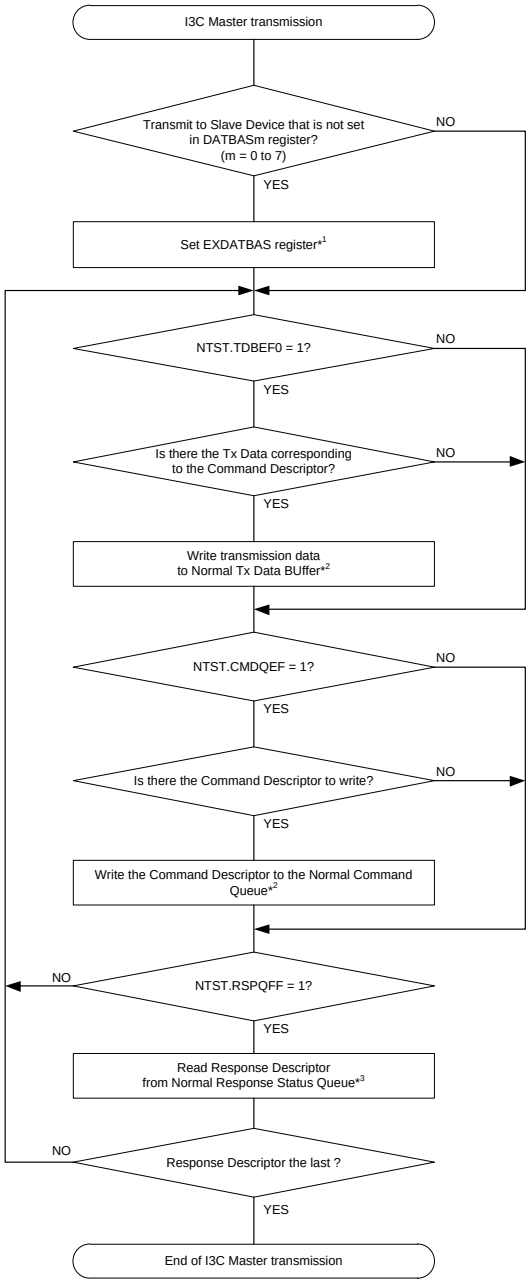


Figure 7.8-116 Example of I²C master reception flowchart (7-bit address format, 3 bytes or more)

(3) I3C Master Transmission Flow (Normal FIFO Buffer Transfer)

Master transmission flow in I3C normal FIFO buffer transfer is common to Legacy I^2C and SDR (Private Transfer, Broadcast CCC, Direct CCC).



Note 1. Changing the EXDATBAS register is prohibited until all response descriptors of command for the Slave device set in the EXDATBAS register are stored.

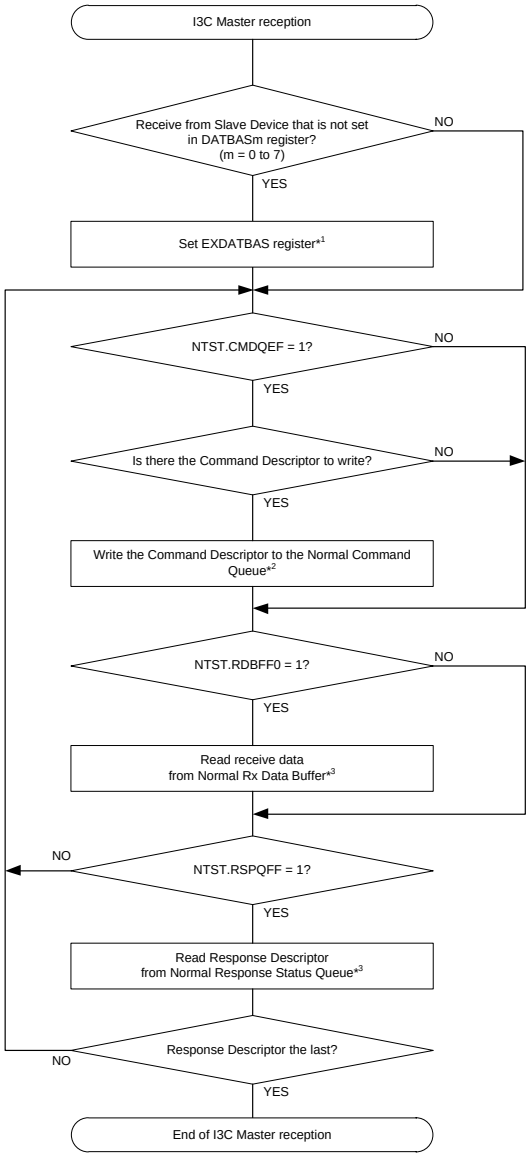
Note 2. In the case the register is written by a way other than DMAC, clear corresponding flags after write access.

Note 3. In the case the register is read by a way other than DMAC, clear corresponding flags after read access.

Figure 7.8-117 Example of I3C master transmission flowchart (normal FIFO buffer transfer)

(4) I3C Master Reception Flow (Normal FIFO Buffer Transfer)

Master reception flow in I3C normal FIFO buffer transfer is common to Legacy I^2C and SDR (Private Transfer, Broadcast CCC, Direct CCC).



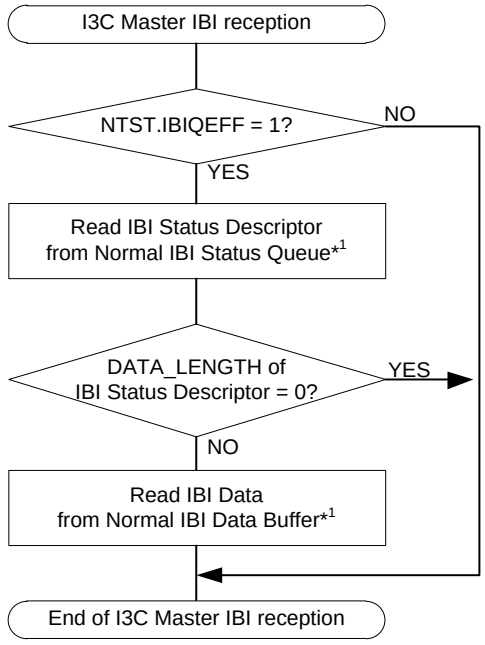
Note 1. Changing the EXDATBAS register is prohibited until all response descriptors of command for the Slave device set in the EXDATBAS register are stored.

Note 2. In the case the register is written by a way other than DMAC, clear corresponding flags after write access.

Note 3. In the case the register is read by a way other than DMAC, clear corresponding flags after read access.

Figure 7.8-118 Example of I3C master reception flowchart (normal FIFO buffer transfer)

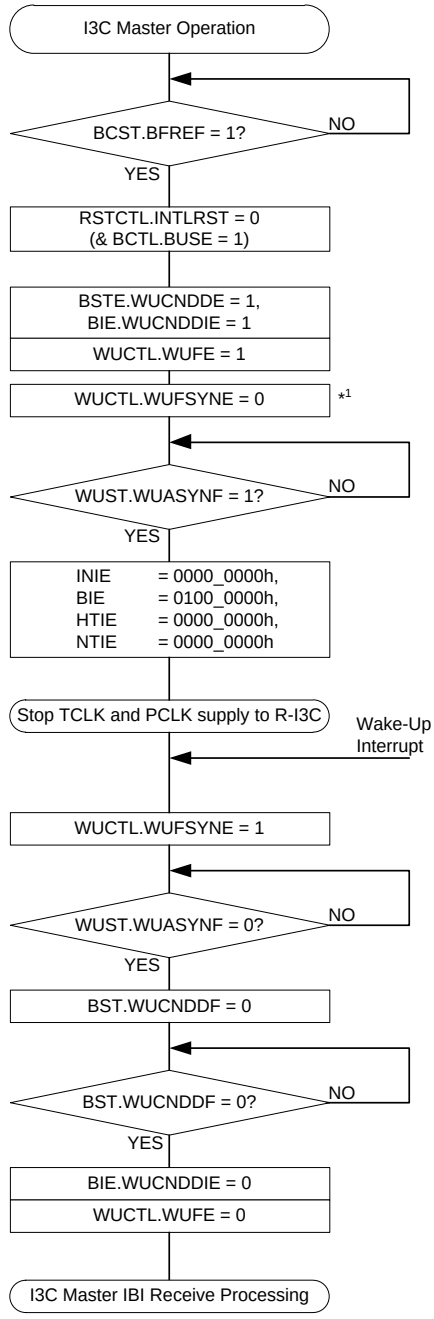
(5) I3C Master IBI Reception Flow



Note 1. In the case the register is read by a way other than DMAC, clear corresponding flags after read access.

Figure 7.8-119 Example of I3C master IBI reception flowchart

(6) I3C Master Wake-Up Flow



[1] [1] Wait I3C bus free and I3C stay in the standby state.

[2] [2] Negate internal reset if assert the internal reset.

[3] [3] Wake-Up Interrupt is enabled.

[4] [4] Wake-Up function is enabled.

[5] [5] R-I3C of operating state is changed from I3C_0_PCLK / I3C_0_TCLK synchronous to I3C_0_PCLK / I3C_0_TCLK asynchronous.

[6] [6] Interrupt requests are disabled except WUI.

[7] [7] Stop I3C_0_TCLK supply to R-I3C, then stop I3C_0_PCLK supply to I3C. (I3C will continue to receive)

[8] [8] Start system clock and I3C_0_TCLK supply to R-I3C by Wake-Up interrupt,then start I3C_0_PCLK supply to I3C.

[9] [9] R-I3C of operating state is changed from I3C_0_PCLK / I3C_0_TCLK asynchronous to I3C_0_PCLK / I3C_0_TCLK synchronous.

[10] [10] Write 0 to BST.WUCNDDF.
(Read and check whether clearing has been completed befor return from interrupt handling.)

[11] [11] Wake-Up interrupt is disabled.

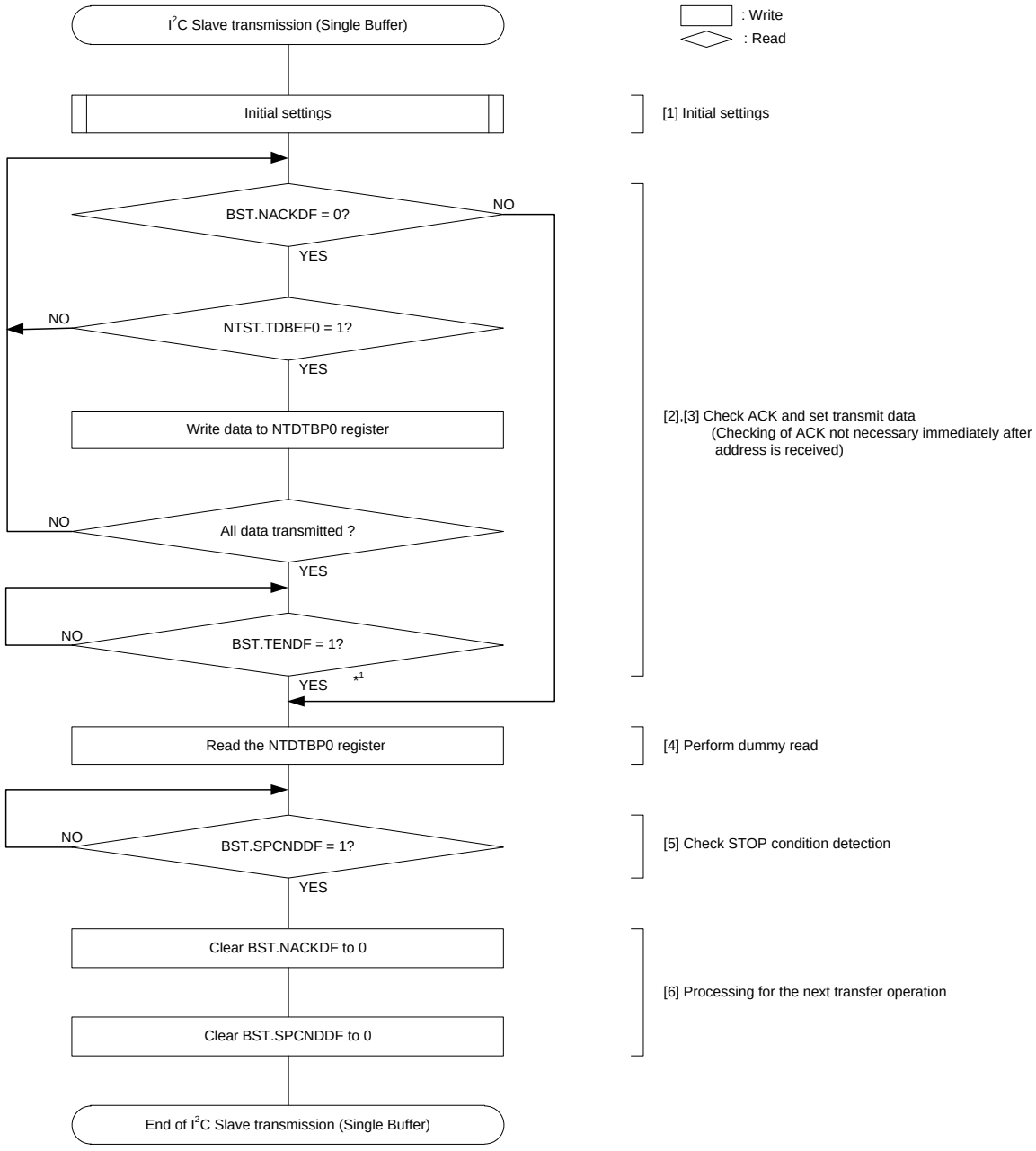
[12] [12] Wake-Up function is disabled.

Note 1. After switching the SDA pin to the general I/O port (Input) in the I/O port function setting, set WUCTL.WUFSYNE to 1. Then wait until WUST.WUASYNF = 1, and switch the SDA pin to the peripheral function in the I/O port function setting.

Figure 7.8-120 Use Case of I3C Master Wake-Up

7.8.4.3.4 Slave Mode Communication Flow

(1) I²C Slave Transmission Flow (Single Buffer Transfer)



Note 1. Normally, Master responds to the last byte with NACK, so when BST.TENDF=1, BST.NACKDF will also 1.
If BST.NACKDF is 0 after BST.TENDF=1, the master may be hung up, so reset I3C with RSTCTL.INTLRST.

Figure 7.8-121 Example of I²C slave transmission flowchart (single buffer transfer)

(2) I²C Slave Reception Flow (Single Buffer Transfer)

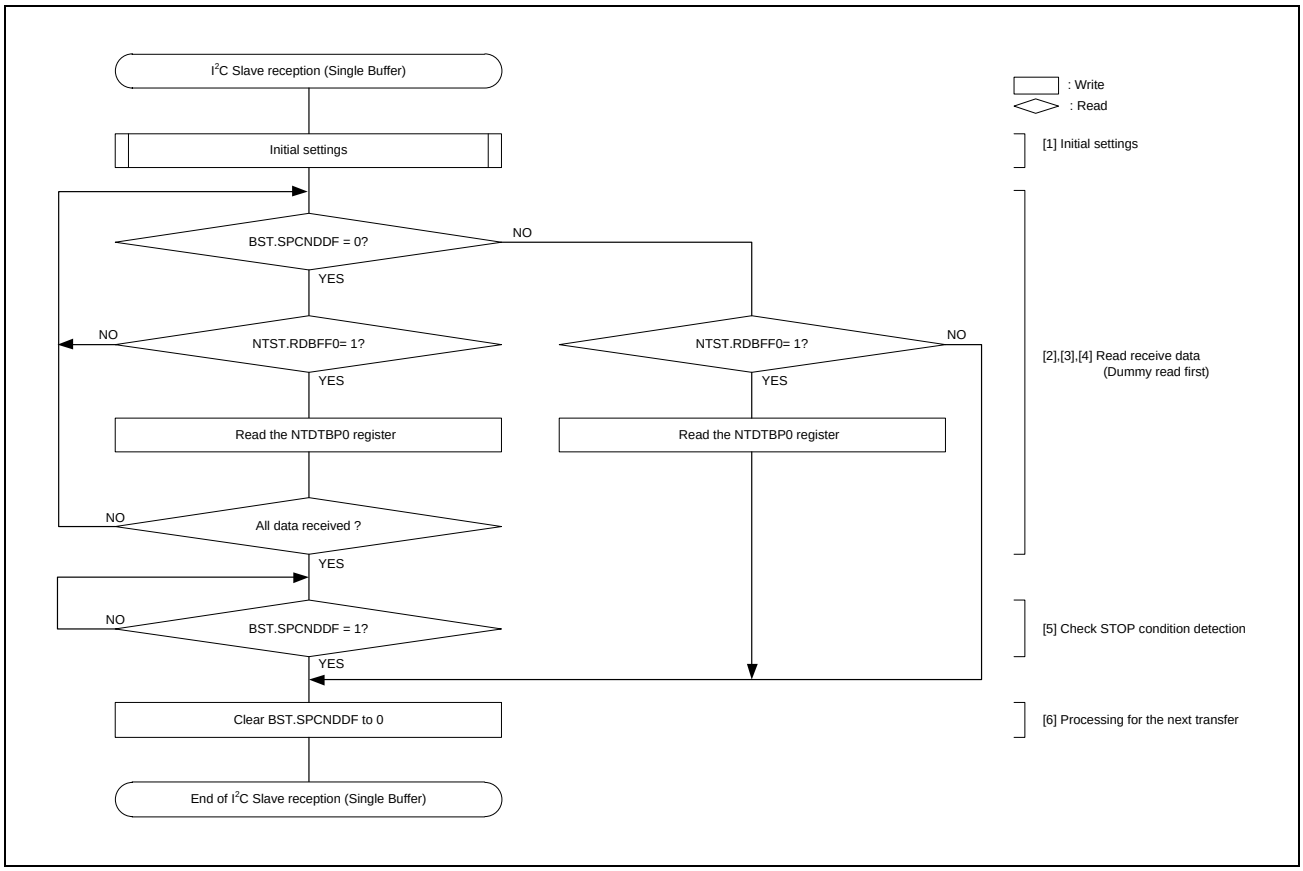
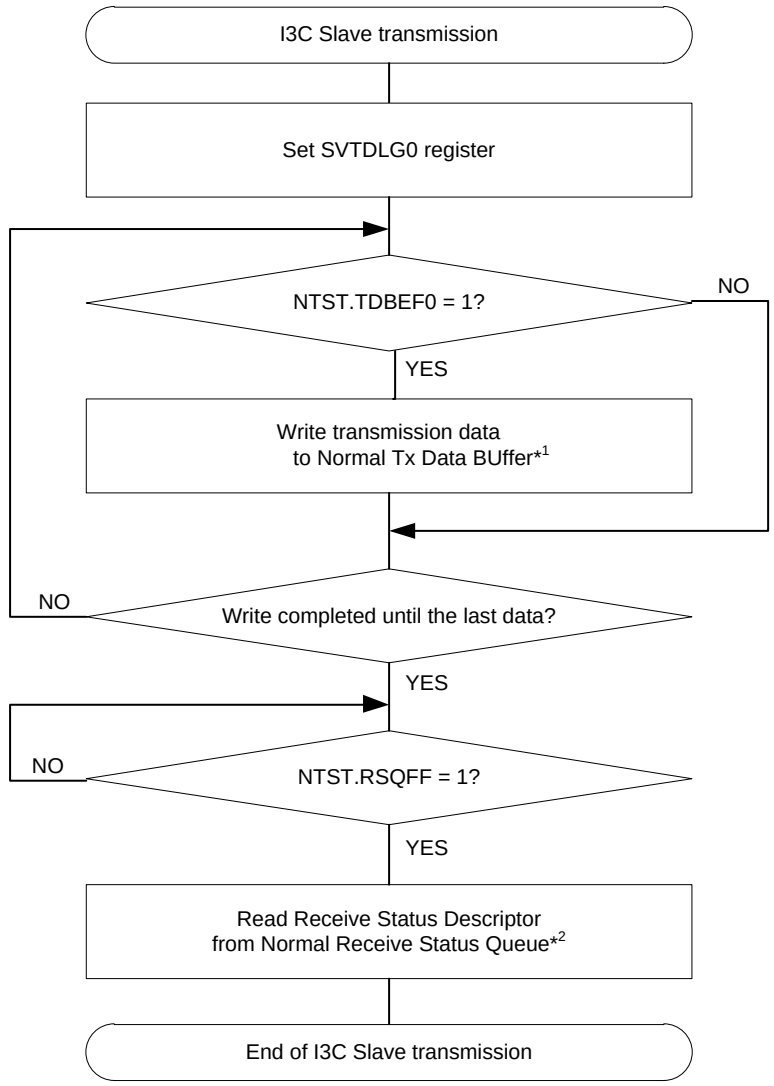


Figure 7.8-122 Example of I²C slave reception flowchart (single buffer transfer)

(3) I3C Slave Transmission Flow (Normal FIFO Buffer Transfer)

Slave Transmission Flow in I3C normal FIFO buffer transfer is common to Legacy I^2C, SDR (Private Transfer, Broadcast CCC, Direct CCC).



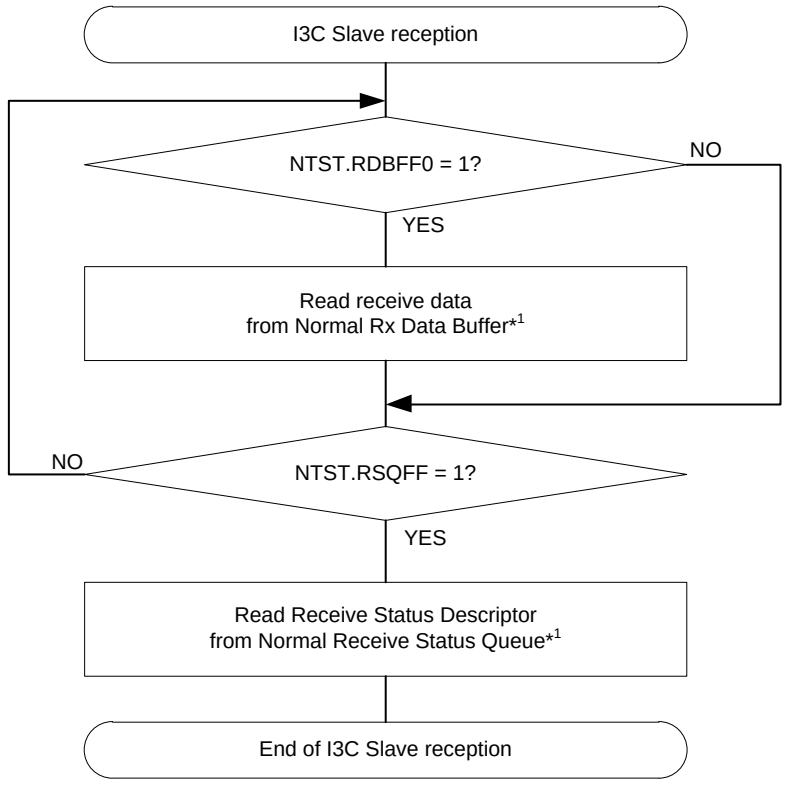
Note 1. In the case the register is written by a way other than DMAC, clear corresponding flags after write access.

Note 2. In the case the register is read by a way other than DMAC, clear corresponding flags after read access.

Figure 7.8-123 Example of I3C slave transmission flowchart (normal FIFO buffer transfer)

(4) I3C Slave Reception Flow (Normal FIFO Buffer Transfer)

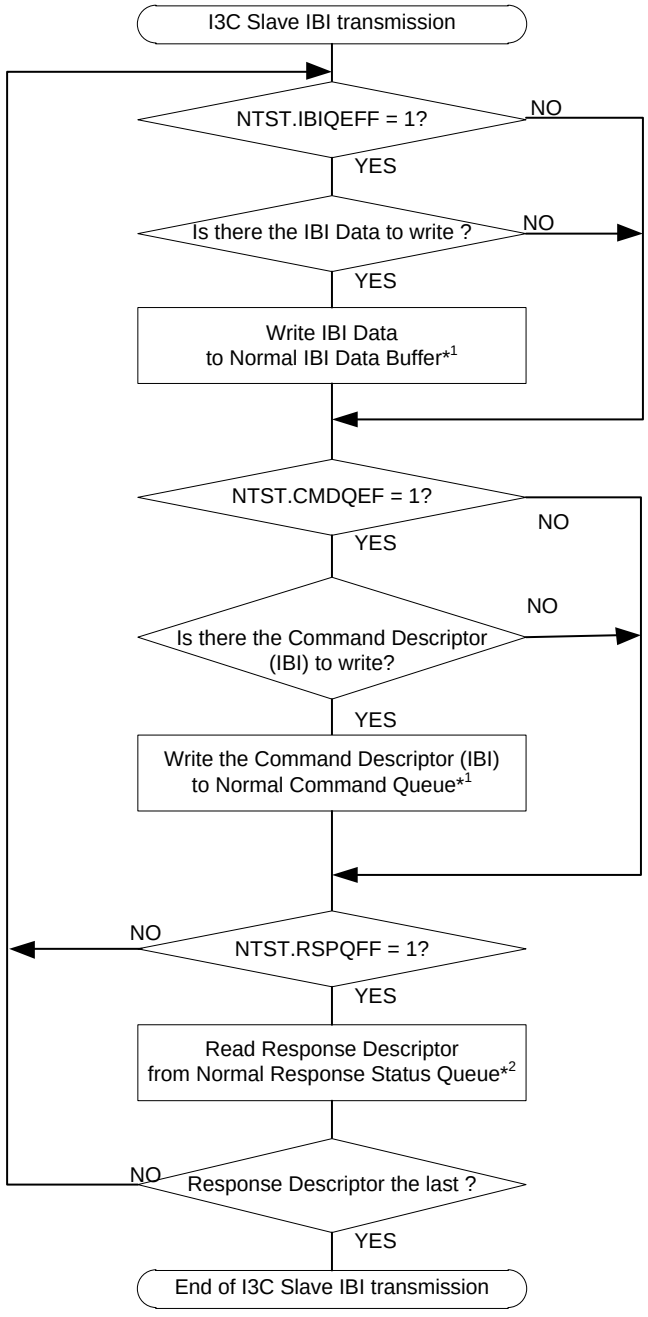
Slave Reception Flow in I3C normal FIFO buffer transfer is common to Legacy I^2C, SDR (Private Transfer, Broadcast CCC, Direct CCC).



Note 1. In the case the register is read by a way other than DMAC, clear corresponding flags after read access.

Figure 7.8-124 Example of I3C slave reception flowchart (normal FIFO buffer transfer)

(5) I3C Slave IBI Transmission Flow



Note 1. In the case the register is written by a way other than DMAC, clear corresponding flags after write access.

Note 2. In the case the register is read by a way other than DMAC, clear corresponding flags after read access.

Figure 7.8-125 Example of I3C slave IBI transmission flowchart

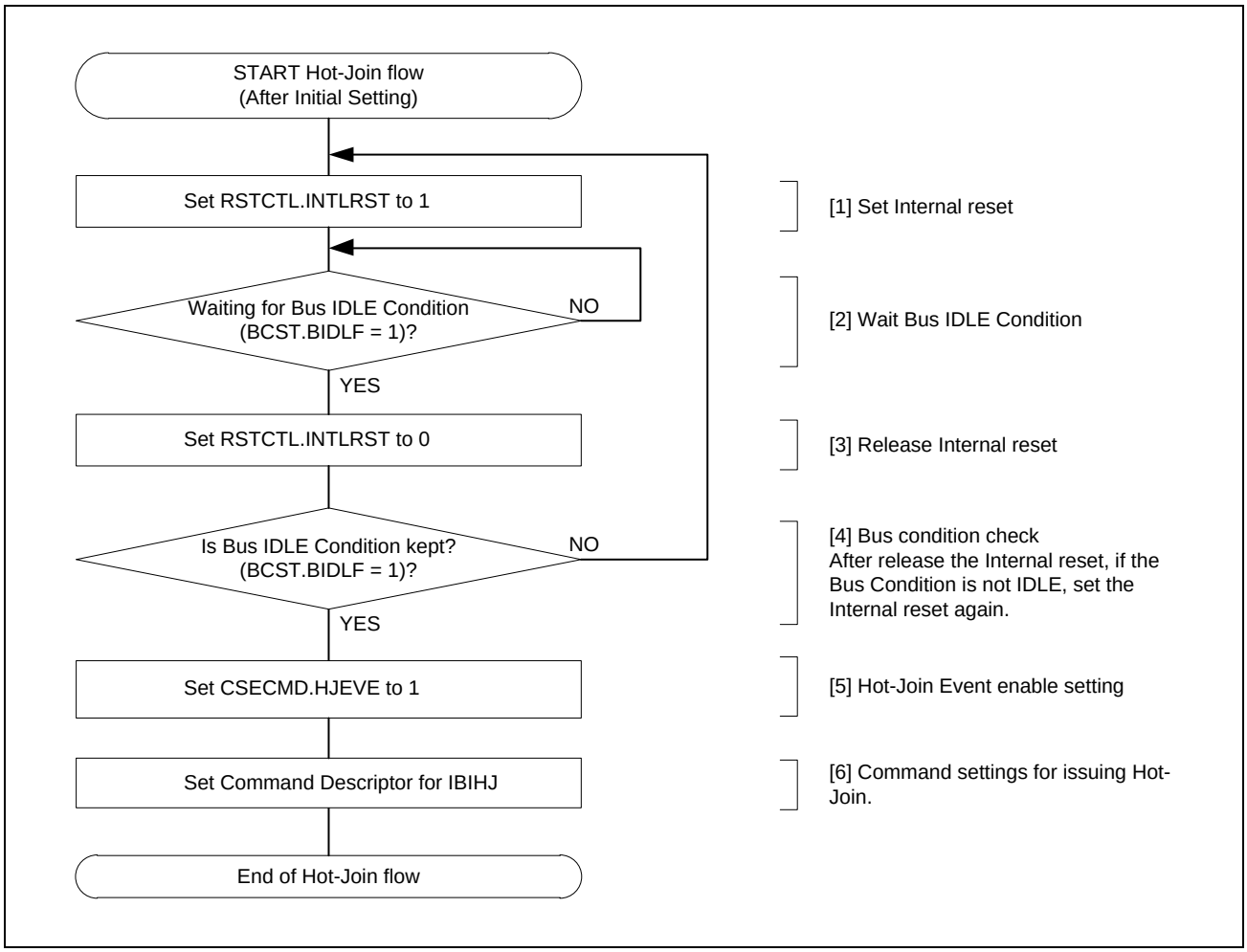


Figure 7.8-126 Hot-Join Flowchart after the I3C Bus has already been configured

(6) I3C Slave Wake-Up Flow

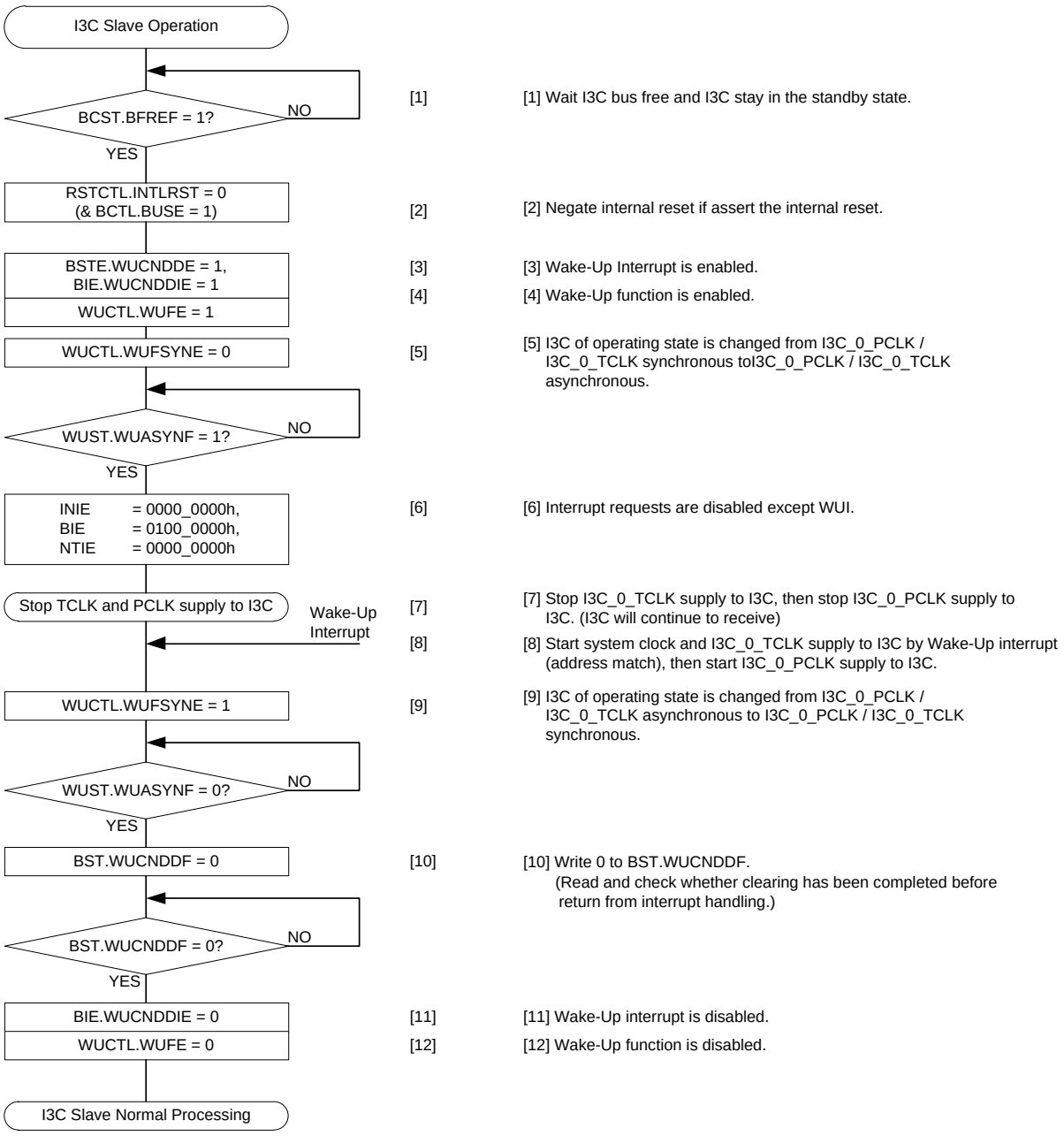


Figure 7.8-127 Use Case of I3C Slave Wake-Up (Wake-up recovery by a wake-up interrupt triggered by the match of the slave address)

7.8.5 Interrupt Sources

I3C can generate the following interrupt requests:

7.8.5.1 Overview

The I3C has the interrupt factors shown in **Table 7.8-17**.

The interrupt indicated by Possible in the DMAC Activation column are capable of activating data transfer by the DMAC.

Table 7.8-17 Interrupt Generation

Symbol	Interrupt source	Interrupt flag	Support				DMAC Activation
			I^2C	I3CM	I3C2M	I3CS	
INT_ri3c_resp_n	Normal response status buffer full	NTST.RSPQFF	—	✓	✓	✓	Not possible
INT_ri3c_cmd_n	Normal command buffer empty	NTST.CMDQEF	—	✓	✓	✓	Not possible
INT_ri3c_ibi_n	Normal IBI status buffer empty/full	NTST.IBIQEFF	—	✓	✓	✓	Not possible
INT_ri3c_rx_n	Normal receive data buffer full	NTST.RDBFF0	✓	✓	✓	✓	Possible
INT_ri3c_tx_n	Normal transmit data buffer empty	NTST.TDBEF0	✓	✓	✓	✓	Possible
INT_ri3c_rcv_n	Normal receive status buffer full	NTST.RSQFF	—	—	✓	✓	Not possible
INT_ri3c_tend_n	Transmit end	BST.TENDF	✓	—	—	—	Not possible
INT_ri3c_st_n	Start condition detection	BST.STCNDDF	✓	✓	✓	✓	Not possible
INT_ri3c_sp_n	STOP condition detection	BST.SPCNDDF	✓	✓	✓	✓	Not possible
INT_ri3c_nack_n	NACK detection	BST.NACKDF	✓	—	—	—	Not possible
INT_ri3c_al_n	Arbitration lost	BST.ALF	✓	—	—	—	Not possible
INT_ri3c_tmo_n	Timeout detection	BST.TODF	✓	✓	✓	✓	Not possible
INT_ri3c_ierr_n	Non-recoverable internal error	INST.INEF	—	✓	✓	✓	Not possible
INT_ri3c_terr_n	Transfer Error	NTST.TEF	—	✓	✓	✓	Not possible
INT_ri3c_abort_n	Transfer Abort	NTST.TABTF	—	✓	✓	✓	Not possible
INT_ri3c_wu_n	Wake-up condition detection	BST.WUCNDDF	✓	✓	✓	✓	Not possible

Note: ✓: Support
— : Not support

Note: I^2C: I^2C Master/Slave (Single Buffer)
I3CM: I3C Master
I3C2M: I3C Secondary Master
I3CS: I3C Slave

7.8.6 Event Link Output

I3C handles event output for the event link controller (ELC) corresponding to the following sources.

(1) Communication event

When a Communication event (arbitration-lost detection, detection of NACK, detection of timeout, detection of a START condition, or detection of a STOP condition) occurs, the corresponding event signal can be output for another module via the ELC.

(2) Receive data full

When a receive data register becomes full, the corresponding event signal can be output for another module via the ELC.

(3) Transmit data empty

When a transmit data register becomes empty, the corresponding event signal can be output for another module via the ELC.

(4) Transmit end

On completion of transfer, the corresponding event signal can be output for another module via the ELC.

All event signals from I3C to ELC are shown in **Table 7.8-18**.

Table 7.8-18 Event Signals

Signal Name	Description	Support			
		I^2C	I3CM	I3C2M	I3CS
ri3c_elccommu	Communication event	✓	✓	✓	✓
ri3c_elcresp	Normal response status buffer full event	—	✓	✓	✓
ri3c_elccmd	Normal command buffer empty event	—	✓	✓	✓
ri3c_elcibi	Normal IBI status buffer empty/full event	—	✓	✓	✓
ri3c_elcrx	Normal receive data buffer full event	✓	✓	✓	✓
ri3c_elctx	Normal transmit data buffer empty event	✓	✓	✓	✓
ri3c_elcrcv	Normal receive status buffer full event	—	—	✓	✓
ri3c_elctend	Transmit end event	✓	—	—	—
ri3c_elcstev	Synchronous timing event	—	✓	✓	✓
ri3c_mrefovf	MREF counter overflow event	—	✓	✓	—
ri3c_mrefcpt	MREF capture event	—	✓	✓	—
ri3c_elcamev	Additional master-initiated bus event	—	✓	✓	—

Note: ✓: Support
— : Not support

Note: I^2C: I^2C Master/Slave (Single Buffer)
I3CM: I3C Master
I3C2M: I3C Secondary Master
I3CS: I3C Slave

7.8.6.1 Interrupt Handling and Event Linking

I3C module produces four kinds of interrupt: communication event (arbitration-lost detection, detection of NACK, detection of timeout, detection of a START condition, or detection of a STOP condition), receive data full, transmit data empty, and transmit end interrupts. Each of these has an enable bit to control enabling and disabling of the interrupt signal. An interrupt request signal is output for the CPU when an interrupt source condition is satisfied while the setting of the corresponding enable bit is enabled.

The corresponding event link output signals are sent to other modules as event signals via the ELC when the interrupt source conditions are satisfied, regardless of the settings of the interrupt enable bits. For details on interrupt sources, see **7.8.5.1 Overview**.

7.8.7 Reset Descriptions

Table 7.8-19 Register states when issuing each condition (1)

Register symbol	Register bit name	System reset	RSTCTL Register							
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST
PRTS	PRTMD	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BCTL	BUSE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSM	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	ABT	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HJACKCTL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	INCBA	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
MSDVAD	MDYADV	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MDYAD[6:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
RSTCTL	INTLRST	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSQRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	IBIQRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	RDBRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	TDBRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	RSPQRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	CMDQRST	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	RI3CRST	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved	Saved
PRSST	PRSSTWP	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	TRMD	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	CRMS	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
INST	INEF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
INSTE	INEE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
INIE	INEIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
INSTFC	INEFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
DVCT	IDX[4:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
IBINCTL	NRSIRCTL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NRMRCTL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NRHJCTL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BFCTL	HSME	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	FMPE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SMBS	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SCSYNE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SALE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NALE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MALE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Note: In reset: To be reset (The FIFO corresponding to this register is cleared)

Table 7.8-20 Register states when issuing each condition (2) (1/2)

Register symbol	Register bit name	System reset	RSTCTL Register							
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST
SVCTL	SVAE2	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SVAE1	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SVAE0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HOAE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVIDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	HSMCE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	GCAE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
REFCKCTL	IREFCKS[2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
STDBR	DSBRPO	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SBRHP[5:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SBRLP[5:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SBRHO[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SBRLO[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
EXTBR	EBRHP[5:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EBRLP[5:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EBRHO[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EBRLO[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BFRECDT	FRECYC[8:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BAVLCDT	AVLCYC[8:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BIDLCDT	IDLCYC[17:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
OUTCTL	SDODCS	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SDOD[2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EXCYC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SOCWP	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	SCOC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SDOC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
INCTL	DNFE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DNFS[3:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
TMOCTL	TOMDS[1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TOHCTL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TOLCTL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TODTS[1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
WUCTL	WUFE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	WUFSYNE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	WUANFS	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	WUACKS	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
ACKCTL	ACKTWP	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	ACKT	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	ACKR	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
SCSTRCTL	RWE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ACKTWE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ACKPE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Table 7.8-20 Register states when issuing each condition (2) (2/2)

Register symbol	Register bit name	System reset	RSTCTL Register							
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST
SCSTRCTL	PARPE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	AAPE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	STLCYC[15:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SVTDLG0	STDLG[15:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
STCTL	STOE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
ATCTL	CDIV[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	AMEOE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MREFOE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
ATTRG	ATSTRG	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
ATCCNTE	ATCE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Note: In reset: To be reset (The FIFO corresponding to this register is cleared)

Table 7.8-21 Register states when issuing each condition (3)

Register symbol	Register bit name	System reset	RSTCTL Register							
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST
CNDCTL	SPCND	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	SRCND	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	STCND	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
NCMDQP	NCMDQP[31:0]	In reset	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved
NRSPQP	NRSPQP[31:0]	In reset	In reset	In reset	Saved	In reset	Saved	Saved	Saved	Saved
NTDTBP0	NTDTBP0[31:0]	In reset	In reset	In reset	Saved	Saved	In reset	In reset	Saved	Saved
NIBIQP	NIBIQP[31:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	In reset	Saved
NRSQP	NRSQP[31:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	In reset
NQTHCTL	IBIQTH[2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	IBIDSSZ[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSPQTH[1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	CMDQTH[1: 0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NTBTHCTL0	RXSTTH[2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TXSTTH[2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RXDBTH[2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TXDBTH[2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NRQTHCTL	RSQTH	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BST	WUCNDDF	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TODF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	ALF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	TENDF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	NACKDF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	SPCNDDF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	STCNDDF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
BSTE	WUCNDDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TODE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ALE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TENDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NACKDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SPCNDDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	STCNDDE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Note: In reset: To be reset (The FIFO corresponding to this register is cleared)

Table 7.8-22 Register states when issuing each condition (4)

Register symbol	Register bit name	System reset	RSTCTL Register							
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST
BIE	WUCNDDIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TODIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ALIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TENDIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NACKDIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SPCNDDIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	STCNDDIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BSTFC	WUCNDDFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TODFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ALFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TENDFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	NACKDFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SPCNDDFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	STCNDDFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NTST	RSQFF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	In reset
	TEF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	TABTF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	RSPQFF	In reset	In reset	In reset	Saved	In reset	Saved	Saved	Saved	Saved
	CMDQEF	In reset	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved
	IBIQEFF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	In reset	Saved
	RDBFF0	In reset	In reset	In reset	Saved	Saved	Saved	In reset	Saved	Saved
	TDBEF0	In reset	In reset	In reset	Saved	Saved	In reset	Saved	Saved	Saved
NTSTE	RSQFE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TEE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TABTE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSPQFE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	CMDQEE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	IBIQEFE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RDBFE0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TDBEE0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
NTIE	RSQFIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TEIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TABTIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSPQFIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	CMDQEIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	IBIQEFIE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RDBFIE0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TDBEIE0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Note: In reset: To be reset (The FIFO corresponding to this register is cleared)

Table 7.8-23 Register states when issuing each condition (5)

Register symbol	Register bit name	System reset	RSTCTL Register							
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST
NTSTFC	RSQFFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TEFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TABTFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RSPQFFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	CMDQEFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	IBIQEFFC	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	RDBFFC0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TDBEFC0	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Note: In reset: To be reset (The FIFO corresponding to this register is cleared)

Table 7.8-24 Register states when issuing each condition (6)

Register symbol	Register bit name	System reset	RSTCTL Register							
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST
BCST	BIDLF	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	BAVLF	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	BFREF	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SVST	SVAF2	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	SVAF1	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	SVAF0	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	HOAF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	DVIDF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	HSMCF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	GCAF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
WUST	WUASYNF	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
MRCCPT	MRCCPT[31:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
DATBASm (m = 0 to 7)	DVTYP	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVNACK[1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVDYAD[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVIBITS	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVMRRJ	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVSIRRJ	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVIBIPL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	DVSTAD[6:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Note: In reset: To be reset (The FIFO corresponding to this register is cleared)

Table 7.8-25 Register states when issuing each condition (7)

Register symbol	Register bit name	System reset	RSTCTL Register							
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST
EXDATBAS	EDTYP	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EDNACK[1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EDDYAD[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	EDSTAD[6:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SDATBASn (n = 0 to 2)	SDDYAD[6:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SDIBIPL	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SDADLS	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SDSTAD[9:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
MSDCTm (m = 0 to 7)	RBCR[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SVDCT	TBCR[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	TDCR[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Note: In reset: To be reset (The FIFO corresponding to this register is cleared)

Table 7.8-26 Register states when issuing each condition (8)

Register symbol	Register bit name	System reset	RSTCTL Register							
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST
SDCTPIDL	SDCTPIDL[31:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SDCTPIDH	SDCTPIDH[31:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SVDVADn (n = 0 to 2)	SDYADV	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SSTADV	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SADLG	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SVAD[9:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CSECMD	HJEVE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MSRQE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SVIRQE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CEACTST	ACTST[3:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CMWLG	MWLG[15:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CMRLG	IBIPSZ[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MRLG[15:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CETSTMD	TSTMD[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CGDVST	VDRSV[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	ACTMD[1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	PRTE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	PNDINT[3:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
CMDSPW	MSWDR[2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CMDSPR	CDTTIM[2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MSRDR[2:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CMDSPT	MRTE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	MRTTIM[23:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved

Note: In reset: To be reset (The FIFO corresponding to this register is cleared)

Table 7.8-27 Register states when issuing each condition (9)

Register symbol	Register bit name	System reset	RSTCTL Register							
			RI3CRST	INTLRST	CMDQRST	RSPQRST	TDBRST	RDBRST	IBIQRST	RSQRST
CETSM	INAC[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	FREQ[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SPTASYN[1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SPTSYN	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
CETSS	ICOVF	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
	ASYNE[1:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SYNE	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
BITCNT	BCNT[4:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
NQSTLV	IBISCNT[4:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	In reset	Saved
	IBIQLV[7:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	In reset	Saved
	RSPQLV[7:0]	In reset	In reset	In reset	Saved	In reset	Saved	Saved	Saved	Saved
	CMDQFLV[7:0]	In reset	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved
NDBSTLV0	RDBLV[7:0]	In reset	In reset	In reset	Saved	Saved	Saved	In reset	Saved	Saved
	TDBFLV[7:0]	In reset	In reset	In reset	Saved	Saved	In reset	Saved	Saved	Saved
NRSQSTLV	RSQLV[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
PRSTDBG	SDOLV	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SCOLV	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SDILV	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
	SCILV	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
MSERRCNT	M2ECNT[7:0]	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved	Saved
SC1CPT	SC1C[15:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved
SC2CPT	SC2C[15:0]	In reset	In reset	In reset	Saved	Saved	Saved	Saved	Saved	Saved

Note: In reset: To be reset (The FIFO corresponding to this register is cleared)

SECTION 7 LOW-SPEED INTERFACE

7.9 CAN-FD Interface (CANFD)

7.9.1 Overview

The LSI has a 6-channel CAN-FD module (CANFD) that complies with ISO 11898-1 (2015) Standards. CANFD transmits and receives both formats of messages, namely the standard identifier (11 bits) (identifier is hereafter referred to as ID) and extended ID (29 bits). **Table 7.9-1** lists the specifications of the CANFD, **Figure 7.9-1** shows a block diagram of the CANFD, and **Table 7.9-2** lists the I/O pins.

Table 7.9-1 CANFD Specifications (1/2)

Parameter	Description
Number of channels	Six channels
Protocol	CAN-FD ISO 11898-1 (2015)
Communication speed	• Classical CAN mode: 1 Mbps • CAN FD mode*1 Nominal bit rate: Max. 1 Mbps Data bit rate: Max. 8 Mbps
Buffer	Total 320 buffers (when frame size is 76 bytes) • Individual buffers: 192 buffers (32 buffers × 6 channels) Transmit buffer: 32 buffers per channel Transmit queue: 4 queues per channel (shared with transmit buffer, up to 16 buffers allocatable) • Shared buffers: 384 buffers for all channels Receive buffer: 0 to 96 buffers Receive FIFO buffer: 8 FIFO buffers (up to 384 buffers allocatable to each) Transmit/receive FIFO buffer: 3 FIFO buffers per channel (up to 384 buffers allocatable to each)
Reception function	• Receives data frames and remote frames • Selects ID format (standard ID, extended ID, or both IDs) to be received • Sets interrupt enable/disable for each FIFO • Mirror function (reception of messages transmitted from the CAN node itself) • Timestamp function (to record message reception time as a 16-bit timer value)
Reception filter function	• Selects receive message according to 768 receive rules • Sets the number of receive rules (0 to 127) for each channel • Acceptance filter processing: Sets ID and mask for each receive rule • DLC filter processing: Enables DLC filter check for each acceptance rule
Receive message transfer function	• Routing function Transfers receive messages to arbitrary destinations (can be transferred to up to 8 buffers) Transfer destination: Receive buffer, receive FIFO buffer, and/or transmit/receive FIFO buffer • Label addition function Stores label information together with a message in a receive buffer and FIFO buffer
Transmission function	• Transmits data frames and remote frames • Selects ID format (standard ID, extended ID, or both IDs) to be transmitted • Sets interrupt enable/disable for each transmit buffer and transmit/receive FIFO buffer • Selects ID priority transmission or transmit buffer number priority transmission • Transmit request can be aborted (possible to confirm with a flag) • One-shot transmission function
Interval transmission function	Transmit message at configurable intervals (transmit mode or gateway mode of transmit/receive FIFO buffers)

Table 7.9-1 CANFD Specifications (2/2)

Parameter	Description
Transmit queue function	Transmits all stored messages according to the ID priority
Transmit history function	Stores the history information of transmission-completed messages Adds timestamp (recording message transmission time as a 16-bit timer value) to the history information
Gateway function	Transmits a received message automatically
Bus off recovery mode selection	Selects the method for returning from bus-off state. • ISO 11898-1 compliant • Automatic entry to channel halt mode at bus-off entry • Automatic entry to channel halt mode at bus-off end • Transition to channel halt mode by program request • Transition to the error-active state by program request (forcible return from the bus-off state)
Error status monitoring	• Monitors CAN protocol errors (stuff error, from error, ACK error, CRC error, bit error, ACK delimiter error, and bus dominant lock) • Defects error status transitions (error warning, error passive, bus-off entry, and bus-off recovery) • Reads the error counter • Monitors DLC errors
Interrupt source	8 interrupt sources • Global interrupts (2 sources) RX FIFO interrupt Global error interrupt • Channel interrupts (3 sources/channel) Channel transmit interrupt Channel error interrupt Common RX FIFO interrupt 14 DMA requests • 8 for RX FIFO • 6 for the first common FIFO (1 source/channel)
CAN clock source	CANFD_0_clkc (80 MHz)
Test function	Test function for user evaluation • Listen-only mode • Self-test mode 0 (external loopback) • Self-test mode 1 (internal loopback) • Restricted operation mode • RAM test (read/write test) • Inter-channel communication test (CRC error test enabled)

Note 1. 176 LQFP and 128 LQFP products do not support CAN-FD mode.

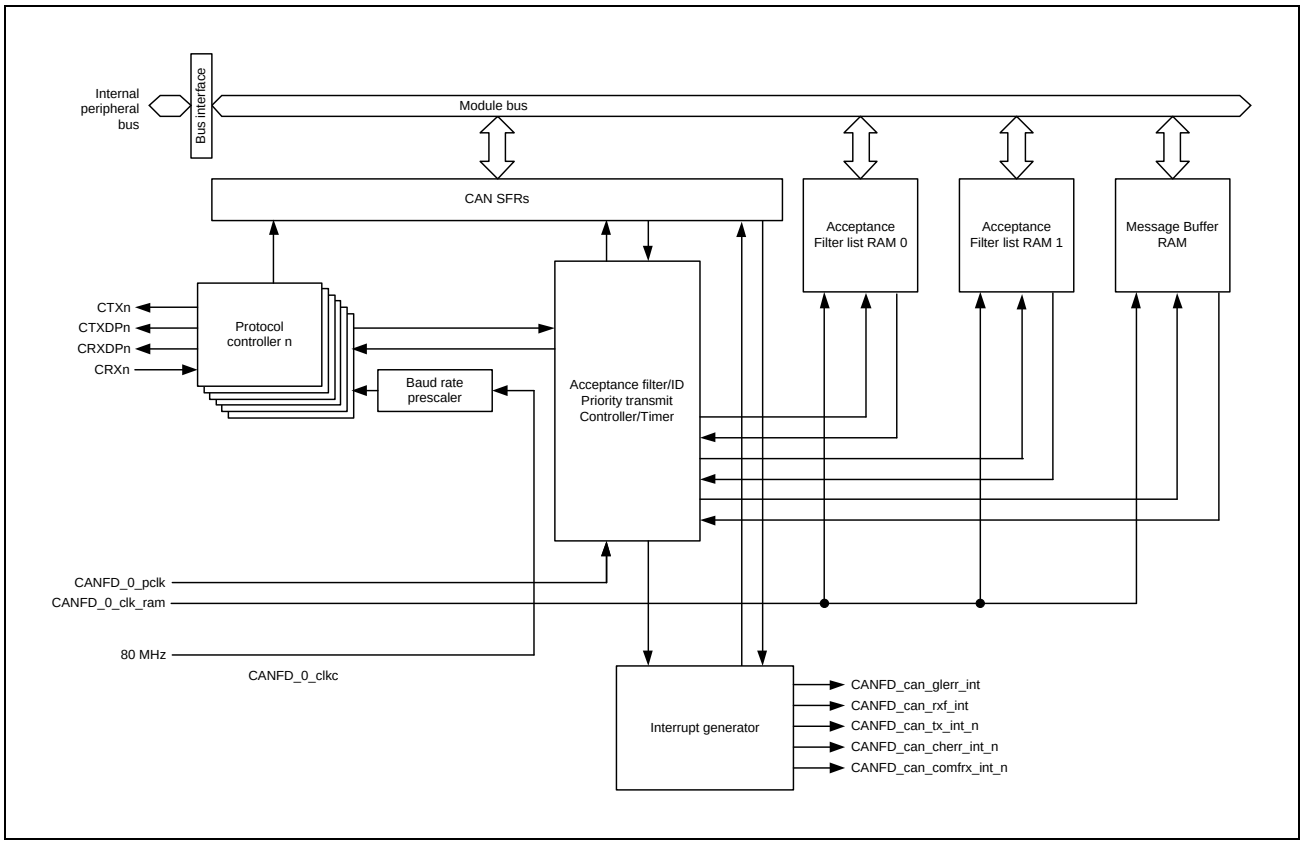


Figure 7.9-1 Block Diagram of CANFD

Table 7.9-2 lists the input/output pins of the CANFD module.

Table 7.9-2 Pin Configuration of CANFD

Channel	Pin Name	Input/Output	Function
CANFDn	CRXn	Input	CANFDn receive data input
	CTXn	Output	CANFDn transmit data output
	CRXDPn	Output	CANFDn receive data phase output This signal indicates the received data phase of a CAN-FD frame (active high).
	CTXDPn	Output	CANFDn transmit data phase output This signal indicates the transmission data phase of a CAN-FD frame (active high).

Note: n = 0 to 5

Table 7.9-3 CANFD Interrupt Sources

Name	Interrupt Sources	DMAC Activation
CANFD_can_rxf_int	RX FIFO interrupt	Not possible
CANFD_can_glerr_int	Global error interrupt	Not possible
CANFD_can_rf_dmareg_m	RX FIFO m DMA request	Possible
CANFD_can_tx_int_n	Channel n TX interrupt	Not possible
CANFD_can_cherr_int_n	Channel n CAN error interrupt	Not possible
CANFD_nan_comfrx_int_n	Channel n Common RX FIFO or TXQ interrupt	Not possible
CANFD_can_cf_dmareg_n	Channel n First common FIFO DMA request	Possible

Note: m = 0 to 7, n = 0 to 5

7.9.2 CANFD Registers

The base address for the CANFD is as follows.

Table 7.9-4 Register Base Address

Base Register Name	Base Address
<CFD_base>	0_1244_0000h (5244_0000h*1, 4244_0000h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

7.9.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
Channel 0 Nominal Bit Rate Configuration Register	CFDC0NCFG	0000_0000h	0000h	8, 16, 32
Channel 0 Control Register	CFDC0CTR	0000_0005h	0004h	8, 16, 32
Channel 0 Status Register	CFDC0STS	0000_0005h	0008h	8, 16, 32
Channel 0 Error Flag Register	CFDC0ERFL	0000_0000h	000Ch	8, 16, 32
Channel 1 Nominal Bit Rate Configuration Register	CFDC1NCFG	0000_0000h	0010h	8, 16, 32
Channel 1 Control Register	CFDC1CTR	0000_0005h	0014h	8, 16, 32
Channel 1 Status Register	CFDC1STS	0000_0005h	0018h	8, 16, 32
Channel 1 Error Flag Register	CFDC1ERFL	0000_0000h	001Ch	8, 16, 32
Channel 2 Nominal Bit Rate Configuration Register	CFDC2NCFG	0000_0000h	0020h	8, 16, 32
Channel 2 Control Register	CFDC2CTR	0000_0005h	0024h	8, 16, 32
Channel 2 Status Register	CFDC2STS	0000_0005h	0028h	8, 16, 32
Channel 2 Error Flag Register	CFDC2ERFL	0000_0000h	002Ch	8, 16, 32
Channel 3 Nominal Bit Rate Configuration Register	CFDC3NCFG	0000_0000h	0030h	8, 16, 32
Channel 3 Control Register	CFDC3CTR	0000_0005h	0034h	8, 16, 32
Channel 3 Status Register	CFDC3STS	0000_0005h	0038h	8, 16, 32
Channel 3 Error Flag Register	CFDC3ERFL	0000_0000h	003Ch	8, 16, 32
Channel 4 Nominal Bit Rate Configuration Register	CFDC4NCFG	0000_0000h	0040h	8, 16, 32
Channel 4 Control Register	CFDC4CTR	0000_0005h	0044h	8, 16, 32
Channel 4 Status Register	CFDC4STS	0000_0005h	0048h	8, 16, 32
Channel 4 Error Flag Register	CFDC4ERFL	0000_0000h	004Ch	8, 16, 32
Channel 5 Nominal Bit Rate Configuration Register	CFDC5NCFG	0000_0000h	0050h	8, 16, 32
Channel 5 Control Register	CFDC5CTR	0000_0005h	0054h	8, 16, 32
Channel 5 Status Register	CFDC5STS	0000_0005h	0058h	8, 16, 32
Channel 5 Error Flag Register	CFDC5ERFL	0000_0000h	005Ch	8, 16, 32
Reserve	-	-	0060h to 007Fh	-
Global IP Version Register	CFDGIPV	1124_0043h	0080h	8, 16, 32
Global Configuration Register	CFDGCFG	0000_0000h	0084h	8, 16, 32
Global Control Register	CFDGCTR	0000_0005h	0088h	8, 16, 32
Global Status Register	CFDGSTS	0000_000Dh	008Ch	8, 16, 32
Global Error Flag Register	CFDGERFL	0000_0000h	0090h	8, 16, 32
Global Timestamp Counter Register	CFDGTSC	0000_0000h	0094h	16, 32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
Global Acceptance Filter List Entry Control Register	CFDGAFLECTR	0000_0000h	0098h	8, 16, 32
Global Acceptance Filter List Configuration Register 0	CFDGAFLCFG0	0000_0000h	009Ch	8, 16, 32
Global Acceptance Filter List Configuration Register 1	CFDGAFLCFG1	0000_0000h	00A0h	8, 16, 32
Global Acceptance Filter List Configuration Register 2	CFDGAFLCFG2	0000_0000h	00A4h	8, 16, 32
Reserve	-	-	00A8h to 00ABh	-
RX Message Buffer Number Register	CFDRMNB	0000_0000h	00ACh	8, 16, 32
RX Message Buffer New Data Register t	CFDRMNDt	0000_0000h	00B0h + t x 0004h	8, 16, 32
Reserve	-	-	00BCh to 00BFh	-
RX FIFO Configuration/Control Register 0	CFDRFCC0	0000_0000h	00C0h	8, 16, 32
RX FIFO Configuration/Control Register 1	CFDRFCC1	0000_0000h	00C4h	8, 16, 32
RX FIFO Configuration/Control Register 2	CFDRFCC2	0000_0000h	00C8h	8, 16, 32
RX FIFO Configuration/Control Register 3	CFDRFCC3	0000_0000h	00CCh	8, 16, 32
RX FIFO Configuration/Control Register 4	CFDRFCC4	0000_0000h	00D0h	8, 16, 32
RX FIFO Configuration/Control Register 5	CFDRFCC5	0000_0000h	00D4h	8, 16, 32
RX FIFO Configuration/Control Register 6	CFDRFCC6	0000_0000h	00D8h	8, 16, 32
RX FIFO Configuration/Control Register 7	CFDRFCC7	0000_0000h	00DCh	8, 16, 32
RX FIFO Status Register 0	CFDRFSTS0	0000_0001h	00E0h	8, 16, 32
RX FIFO Status Register 1	CFDRFSTS1	0000_0001h	00E4h	8, 16, 32
RX FIFO Status Register 2	CFDRFSTS2	0000_0001h	00E8h	8, 16, 32
RX FIFO Status Register 3	CFDRFSTS3	0000_0001h	00ECh	8, 16, 32
RX FIFO Status Register 4	CFDRFSTS4	0000_0001h	00F0h	8, 16, 32
RX FIFO Status Register 5	CFDRFSTS5	0000_0001h	00F4h	8, 16, 32
RX FIFO Status Register 6	CFDRFSTS6	0000_0001h	00F8h	8, 16, 32
RX FIFO Status Register 7	CFDRFSTS7	0000_0001h	00FCh	8, 16, 32
RX FIFO Pointer Control Register 0	CFDRFPCTR0	0000_0000h	0100h	8, 16, 32
RX FIFO Pointer Control Register 1	CFDRFPCTR1	0000_0000h	0104h	8, 16, 32
RX FIFO Pointer Control Register 2	CFDRFPCTR2	0000_0000h	0108h	8, 16, 32
RX FIFO Pointer Control Register 3	CFDRFPCTR3	0000_0000h	010Ch	8, 16, 32
RX FIFO Pointer Control Register 4	CFDRFPCTR4	0000_0000h	0110h	8, 16, 32
RX FIFO Pointer Control Register 5	CFDRFPCTR5	0000_0000h	0114h	8, 16, 32
RX FIFO Pointer Control Register 6	CFDRFPCTR6	0000_0000h	0118h	8, 16, 32
RX FIFO Pointer Control Register 7	CFDRFPCTR7	0000_0000h	011Ch	8, 16, 32
Common FIFO Configuration/Control Register n	CFDCFCCn	0000_0000h	0120h + n x 0004h	8, 16, 32
Reserve	-	-	016Ch to 017Fh	-
Common FIFO Configuration/Control Enhancement Register n	CFDCFCCEn	0000_0000h	0180h + n x 0004h	8, 16, 32
Reserve	-	-	01CCh to 01DFh	-
Common FIFO Status Register n	CFDCFSTSn	0000_0001h	01E0h + n x 0004h	8, 16, 32
Reserve	-	-	022Ch to 023Fh	-
Common FIFO Pointer Control Register n	CFDCFPCTRn	0000_0000h	0240h + n x 0004h	8, 16, 32
Reserve	-	-	028Ch to 029Fh	-
FIFO Empty Status Register	CFDFESTS	0000_3FFFh	02A0h	8, 16, 32
FIFO Full Status Register	CFDFFSTS	0000_0000h	02A4h	8, 16, 32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
FIFO Message Lost Status Register	CFDFMSTS	0000_0000h	02A8h	8, 16, 32
RX FIFO Interrupt Flag Status Register	CFDRFISTS	0000_0000h	02ACh	8, 16, 32
Common FIFO RX Interrupt Flag Status Register	CFDCFRISTS	0000_0000h	02B0h	8, 16, 32
Common FIFO TX Interrupt Flag Status Register	CFDCFTISTS	0000_0000h	02B4h	8, 16, 32
Common FIFO One Frame RX Interrupt Flag Status Register	CFDCFOFRISTS	0000_0000h	02B8h	8, 16, 32
Common FIFO One Frame TX Interrupt Flag Status Register	CFDCFOFTISTS	0000_0000h	02BCh	8, 16, 32
Common FIFO Message Overwrite Status Register	CFDCFMOWSTS	0000_0000h	02C0h	8, 16, 32
FIFO FDC Full Status Register	CFDFFFSTS	0000_0000h	02C4h	8, 16, 32
Reserve	-	-	02C8h to 02CFh	-
TX Message Buffer Control Register n	CFDTMCn	00h	02D0h + n x 0001h	8
Reserve	-	-	0450h to 07CFh	-
TX Message Buffer Status Register n	CFDTMSTSn	00h	07D0h + n x 0001h	8
Reserve	-	-	0950h to 0CCFh	-
TX Message Buffer Transmission Request Status Register f	CFDTMTRSTSf	0000_0000h	0CD0h + f x 0004h	8, 16, 32
Reserve	-	-	0D00h to 0D6Fh	-
TX Message Buffer Transmission Abort Request Status Register f	CFDTMTARSTSf	0000_0000h	0D70h + f x 0004h	8, 16, 32
Reserve	-	-	0DA0h to 0E0Fh	-
TX Message Buffer Transmission Completion Status Register f	CFDTMTCSTSf	0000_0000h	0E10h + f x 0004h	8, 16, 32
Reserve	-	-	0E40h to 0EAFh	-
TX Message Buffer Transmission Abort Status Register f	CFDTMTASTSf	0000_0000h	0EB0h + f x 0004h	8, 16, 32
Reserve	-	-	0D0h to 0F4Fh	-
TX Message Buffer Transmission Interrupt Enable Register f	CFDTMIECf	0000_0000h	0F50h + f x 0004h	8, 16, 32
Reserve	-	-	0F80h to 0FFFh	-
TX Queue Configuration/Control Register 00	CFDTXQCC00	0000_0000h	1000h	8, 16, 32
TX Queue Configuration/Control Register 01	CFDTXQCC01	0000_0000h	1004h	8, 16, 32
TX Queue Configuration/Control Register 02	CFDTXQCC02	0000_0000h	1008h	8, 16, 32
TX Queue Configuration/Control Register 03	CFDTXQCC03	0000_0000h	100Ch	8, 16, 32
TX Queue Configuration/Control Register 04	CFDTXQCC04	0000_0000h	1010h	8, 16, 32
TX Queue Configuration/Control Register 05	CFDTXQCC05	0000_0000h	1014h	8, 16, 32
Reserve	-	-	1018h to 101Fh	-
TX Queue Status Register 00	CFDTXQSTS00	0000_0001h	1020h	8, 16, 32
TX Queue Status Register 01	CFDTXQSTS01	0000_0001h	1024h	8, 16, 32
TX Queue Status Register 02	CFDTXQSTS02	0000_0001h	1028h	8, 16, 32
TX Queue Status Register 03	CFDTXQSTS03	0000_0001h	102Ch	8, 16, 32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
TX Queue Status Register 04	CFDTXQSTS04	0000_0001h	1030h	8, 16, 32
TX Queue Status Register 05	CFDTXQSTS05	0000_0001h	1034h	8, 16, 32
Reserve	-	-	1038h to 104Fh	-
TX Queue Pointer Control Register 00	CFDTXQPCTR00	0000_0000h	1040h	8, 16, 32
TX Queue Pointer Control Register 01	CFDTXQPCTR01	0000_0000h	1044h	8, 16, 32
TX Queue Pointer Control Register 02	CFDTXQPCTR02	0000_0000h	1048h	8, 16, 32
TX Queue Pointer Control Register 03	CFDTXQPCTR03	0000_0000h	104Ch	8, 16, 32
TX Queue Pointer Control Register 04	CFDTXQPCTR04	0000_0000h	1050h	8, 16, 32
TX Queue Pointer Control Register 05	CFDTXQPCTR05	0000_0000h	1054h	8, 16, 32
Reserve	-	-	1058h to 105Fh	-
TX Queue Configuration/Control Register 10	CFDTXQCC10	0000_0000h	1060h	8, 16, 32
TX Queue Configuration/Control Register 11	CFDTXQCC11	0000_0000h	1064h	8, 16, 32
TX Queue Configuration/Control Register 12	CFDTXQCC12	0000_0000h	1068h	8, 16, 32
TX Queue Configuration/Control Register 13	CFDTXQCC13	0000_0000h	106Ch	8, 16, 32
TX Queue Configuration/Control Register 14	CFDTXQCC14	0000_0000h	1070h	8, 16, 32
TX Queue Configuration/Control Register 15	CFDTXQCC15	0000_0000h	1074h	8, 16, 32
Reserve	-	-	1078h to 107Fh	-
TX Queue Status Register 10	CFDTXQSTS10	0000_0001h	1080h	8, 16, 32
TX Queue Status Register 11	CFDTXQSTS11	0000_0001h	1084h	8, 16, 32
TX Queue Status Register 12	CFDTXQSTS12	0000_0001h	1088h	8, 16, 32
TX Queue Status Register 13	CFDTXQSTS13	0000_0001h	108Ch	8, 16, 32
TX Queue Status Register 14	CFDTXQSTS14	0000_0001h	1090h	8, 16, 32
TX Queue Status Register 15	CFDTXQSTS15	0000_0001h	1094h	8, 16, 32
Reserve	-	-	1098h to 109Fh	-
TX Queue Pointer Control Register 10	CFDTXQPCTR10	0000_0000h	10A0h	8, 16, 32
TX Queue Pointer Control Register 11	CFDTXQPCTR11	0000_0000h	10A4h	8, 16, 32
TX Queue Pointer Control Register 12	CFDTXQPCTR12	0000_0000h	10A8h	8, 16, 32
TX Queue Pointer Control Register 13	CFDTXQPCTR13	0000_0000h	10ACh	8, 16, 32
TX Queue Pointer Control Register 14	CFDTXQPCTR14	0000_0000h	10B0h	8, 16, 32
TX Queue Pointer Control Register 15	CFDTXQPCTR15	0000_0000h	10B4h	8, 16, 32
Reserve	-	-	10B8h to 10BFh	-
TX Queue Configuration/Control Register 20	CFDTXQCC20	0000_0000h	10C0h	8, 16, 32
TX Queue Configuration/Control Register 21	CFDTXQCC21	0000_0000h	10C4h	8, 16, 32
TX Queue Configuration/Control Register 22	CFDTXQCC22	0000_0000h	10C8h	8, 16, 32
TX Queue Configuration/Control Register 23	CFDTXQCC23	0000_0000h	10CCh	8, 16, 32
TX Queue Configuration/Control Register 24	CFDTXQCC24	0000_0000h	10D0h	8, 16, 32
TX Queue Configuration/Control Register 25	CFDTXQCC25	0000_0000h	10D4h	8, 16, 32
Reserve	-	-	10D8h to 10DFh	-
TX Queue Status Register 20	CFDTXQSTS20	0000_0001h	10E0h	8, 16, 32
TX Queue Status Register 21	CFDTXQSTS21	0000_0001h	10E4h	8, 16, 32
TX Queue Status Register 22	CFDTXQSTS22	0000_0001h	10E8h	8, 16, 32
TX Queue Status Register 23	CFDTXQSTS23	0000_0001h	10ECh	8, 16, 32
TX Queue Status Register 24	CFDTXQSTS24	0000_0001h	10F0h	8, 16, 32
TX Queue Status Register 25	CFDTXQSTS25	0000_0001h	10F4h	8, 16, 32
Reserve	-	-	10F8h to 10FFh	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
TX Queue Pointer Control Register 20	CFDTXQPCTR20	0000_0000h	1100h	8, 16, 32
TX Queue Pointer Control Register 21	CFDTXQPCTR21	0000_0000h	1104h	8, 16, 32
TX Queue Pointer Control Register 22	CFDTXQPCTR22	0000_0000h	1108h	8, 16, 32
TX Queue Pointer Control Register 23	CFDTXQPCTR23	0000_0000h	110Ch	8, 16, 32
TX Queue Pointer Control Register 24	CFDTXQPCTR24	0000_0000h	1110h	8, 16, 32
TX Queue Pointer Control Register 25	CFDTXQPCTR25	0000_0000h	1114h	8, 16, 32
Reserve	-	-	1118h to 111Fh	-
TX Queue Configuration/Control Register 30	CFDTXQCC30	0000_0000h	1120h	8, 16, 32
TX Queue Configuration/Control Register 31	CFDTXQCC31	0000_0000h	1124h	8, 16, 32
TX Queue Configuration/Control Register 32	CFDTXQCC32	0000_0000h	1128h	8, 16, 32
TX Queue Configuration/Control Register 33	CFDTXQCC33	0000_0000h	112Ch	8, 16, 32
TX Queue Configuration/Control Register 34	CFDTXQCC34	0000_0000h	1130h	8, 16, 32
TX Queue Configuration/Control Register 35	CFDTXQCC35	0000_0000h	1134h	8, 16, 32
Reserve	-	-	1138h to 113Fh	-
TX Queue Status Register 30	CFDTXQSTS30	0000_0001h	1140h	8, 16, 32
TX Queue Status Register 31	CFDTXQSTS31	0000_0001h	1144h	8, 16, 32
TX Queue Status Register 32	CFDTXQSTS32	0000_0001h	1148h	8, 16, 32
TX Queue Status Register 33	CFDTXQSTS33	0000_0001h	114Ch	8, 16, 32
TX Queue Status Register 34	CFDTXQSTS34	0000_0001h	1150h	8, 16, 32
TX Queue Status Register 35	CFDTXQSTS35	0000_0001h	1154h	8, 16, 32
Reserve	-	-	1158h to 115Fh	-
TX Queue Pointer Control Register 30	CFDTXQPCTR30	0000_0000h	1160h	8, 16, 32
TX Queue Pointer Control Register 31	CFDTXQPCTR31	0000_0000h	1164h	8, 16, 32
TX Queue Pointer Control Register 32	CFDTXQPCTR32	0000_0000h	1168h	8, 16, 32
TX Queue Pointer Control Register 33	CFDTXQPCTR33	0000_0000h	116Ch	8, 16, 32
TX Queue Pointer Control Register 34	CFDTXQPCTR34	0000_0000h	1170h	8, 16, 32
TX Queue Pointer Control Register 35	CFDTXQPCTR35	0000_0000h	1174h	8, 16, 32
Reserve	-	-	1178h to 117Fh	-
TX Queue Empty Status Register	CFDTXQESTS	0000_00FFh	1180h	8, 16, 32
TX Queue Full Interrupt Status Register	CFDTXQFISTS	0000_0000h	1184h	8, 16, 32
TX Queue Message Lost Status Register	CFDTXQMSTS	0000_0000h	1188h	8, 16, 32
Reserve	-	-	118Ch to 118Fh	-
TX Queue Interrupt Status Register	CFDTXQISTS	0000_0000h	1190h	8, 16, 32
TX Queue One Frame TX Interrupt Status Register	CFDTXQOFTISTS	0000_0000h	1194h	8, 16, 32
TX Queue One Frame RX Interrupt Status Register	CFDTXQOFRISTS	0000_0000h	1198h	8, 16, 32
TX Queue Full Status Register	CFDTXQFSTS	0000_0000h	119Ch	8, 16, 32
Reserve	-	-	11A0h to 11FFh	-
TX History List Configuration/Control Register 0	CFDTHLCC0	0000_0000h	1200h	8, 16, 32
TX History List Configuration/Control Register 1	CFDTHLCC1	0000_0000h	1204h	8, 16, 32
TX History List Configuration/Control Register 2	CFDTHLCC2	0000_0000h	1208h	8, 16, 32
TX History List Configuration/Control Register 3	CFDTHLCC3	0000_0000h	120Ch	8, 16, 32
TX History List Configuration/Control Register 4	CFDTHLCC4	0000_0000h	1210h	8, 16, 32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
TX History List Configuration/Control Register 5	CFDTHLCC5	0000_0000h	1214h	8, 16, 32
Reserve	-	-	1218h to 121Fh	-
TX History List Status Register 0	CFDTHLSTS0	0000_0001h	1220h	8, 16, 32
TX History List Status Register 1	CFDTHLSTS1	0000_0001h	1224h	8, 16, 32
TX History List Status Register 2	CFDTHLSTS2	0000_0001h	1228h	8, 16, 32
TX History List Status Register 3	CFDTHLSTS3	0000_0001h	122Ch	8, 16, 32
TX History List Status Register 4	CFDTHLSTS4	0000_0001h	1230h	8, 16, 32
TX History List Status Register 5	CFDTHLSTS5	0000_0001h	1234h	8, 16, 32
Reserve	-	-	1238h to 123Fh	-
TX History List Pointer Control Register 0	CFDTHLPCTR0	0000_0000h	1240h	8, 16, 32
TX History List Pointer Control Register 1	CFDTHLPCTR1	0000_0000h	1244h	8, 16, 32
TX History List Pointer Control Register 2	CFDTHLPCTR2	0000_0000h	1248h	8, 16, 32
TX History List Pointer Control Register 3	CFDTHLPCTR3	0000_0000h	124Ch	8, 16, 32
TX History List Pointer Control Register 4	CFDTHLPCTR4	0000_0000h	1250h	8, 16, 32
TX History List Pointer Control Register 5	CFDTHLPCTR5	0000_0000h	1254h	8, 16, 32
Reserve	-	-	1258h to 12FFh	-
Global TX Interrupt Status Register 0	CFDGTINTSTS0	0000_0000h	1300h	8, 16, 32
Global TX Interrupt Status Register 1	CFDGTINTSTS1	0000_0000h	1304h	8, 16, 32
Global Test Configuration Register	CFDGTSTCFG	0000_0000h	1308h	8, 16, 32
Global Test Control Register	CFDGTSTCTR	0000_0000h	130Ch	8, 16, 32
Reserve	-	-	1310h to 1313h	-
Global FD Configuration Register	CFDGFDCFG	0000_0000h	1314h	8, 16, 32
Reserve	-	-	1318h to 131Bh	-
Global Lock Key Register	CFDGLOCKK	0000_0000h	131Ch	16, 32
Reserve	-	-	1320h to 132Fh	-
DMA Transfer Control Register	CFDCDTCT	0000_0000h	1330h	8, 16, 32
DMA Transfer Status Register	CFDCDTSTS	0000_0000h	1334h	8, 16, 32
Reserve	-	-	1338h to 133Fh	-
DMA TX Transfer Control Register	CFDCDTTCT	0000_0000h	1340h	8, 16, 32
DMA TX Transfer Status Register	CFDCDTTSTS	0000_0000h	1344h	8, 16, 32
Reserve	-	-	1348h to 134Fh	-
Global RX Interrupt Status Register 0	CFDGRINTSTS0	0000_0000h	1350h	8, 16, 32
Global RX Interrupt Status Register 1	CFDGRINTSTS1	0000_0000h	1354h	8, 16, 32
Global RX Interrupt Status Register 2	CFDGRINTSTS2	0000_0000h	1358h	8, 16, 32
Global RX Interrupt Status Register 3	CFDGRINTSTS3	0000_0000h	135Ch	8, 16, 32
Global RX Interrupt Status Register 4	CFDGRINTSTS4	0000_0000h	1360h	8, 16, 32
Global RX Interrupt Status Register 5	CFDGRINTSTS5	0000_0000h	1364h	8, 16, 32
Reserve	-	-	1368h to 137Fh	-
Global Reset Control Register	CFDGRSTC	0000_0000h	1380h	16, 32
Global Flexible CAN Mode Configuration Register	CFDGFCMC	0000_0000h	1384h	8, 16, 32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
Reserve	-	-	1388h to 138Bh	-
Global Flexible Transmission Buffer Assignment Configuration Register	CFDGFTBAC	0000_0000h	138Ch	8, 16, 32
Reserve	-	-	1390h to 13FFh	-
Channel 0 Data Bit Rate Configuration Register	CFDC0DCFG	0000_0000h	1400h	8, 16, 32
Channel 0 CAN-FD Configuration Register	CFDC0FDCFG	0000_0000h	1404h	8, 16, 32
Channel 0 CAN-FD Control Register	CFDC0FDCTR	0000_0000h	1408h	8, 16, 32
Channel 0 CAN-FD Status Register	CFDC0FDSTS	0000_0000h	140Ch	8, 16, 32
Channel 0 CAN-FD CRC Register	CFDC0FDCRC	0000_0000h	1410h	8, 16, 32
Reserve	-	-	1414h to 1417h	-
Channel 0 Bus Load Control Register	CFDC0BLCT	0000_0000h	1418h	8, 16, 32
Channel 0 Bus Load Status Register	CFDC0BLSTS	0000_0000h	141Ch	8, 16, 32
Channel 1 Data Bit Rate Configuration Register	CFDC1DCFG	0000_0000h	1420h	8, 16, 32
Channel 1 CAN-FD Configuration Register	CFDC1FDCFG	0000_0000h	1424h	8, 16, 32
Channel 1 CAN-FD Control Register	CFDC1FDCTR	0000_0000h	1428h	8, 16, 32
Channel 1 CAN-FD Status Register	CFDC1FDSTS	0000_0000h	142Ch	8, 16, 32
Channel 1 CAN-FD CRC Register	CFDC1FDCRC	0000_0000h	1430h	8, 16, 32
Reserve	-	-	1434h to 1437h	-
Channel 1 Bus Load Control Register	CFDC1BLCT	0000_0000h	1438h	8, 16, 32
Channel 1 Bus Load Status Register	CFDC1BLSTS	0000_0000h	143Ch	8, 16, 32
Channel 2 Data Bit Rate Configuration Register	CFDC2DCFG	0000_0000h	1440h	8, 16, 32
Channel 2 CAN-FD Configuration Register	CFDC2FDCFG	0000_0000h	1444h	8, 16, 32
Channel 2 CAN-FD Control Register	CFDC2FDCTR	0000_0000h	1448h	8, 16, 32
Channel 2 CAN-FD Status Register	CFDC2FDSTS	0000_0000h	144Ch	8, 16, 32
Channel 2 CAN-FD CRC Register	CFDC2FDCRC	0000_0000h	1450h	8, 16, 32
Reserve	-	-	1454h to 1457h	-
Channel 2 Bus Load Control Register	CFDC2BLCT	0000_0000h	1458h	8, 16, 32
Channel 2 Bus Load Status Register	CFDC2BLSTS	0000_0000h	145Ch	8, 16, 32
Channel 3 Data Bit Rate Configuration Register	CFDC3DCFG	0000_0000h	1460h	8, 16, 32
Channel 3 CAN-FD Configuration Register	CFDC3FDCFG	0000_0000h	1464h	8, 16, 32
Channel 3 CAN-FD Control Register	CFDC3FDCTR	0000_0000h	1468h	8, 16, 32
Channel 3 CAN-FD Status Register	CFDC3FDSTS	0000_0000h	146Ch	8, 16, 32
Channel 3 CAN-FD CRC Register	CFDC3FDCRC	0000_0000h	1470h	8, 16, 32
Reserve	-	-	1474h to 1477h	-
Channel 3 Bus Load Control Register	CFDC3BLCT	0000_0000h	1478h	8, 16, 32
Channel 3 Bus Load Status Register	CFDC3BLSTS	0000_0000h	147Ch	8, 16, 32
Channel 4 Data Bit Rate Configuration Register	CFDC4DCFG	0000_0000h	1480h	8, 16, 32
Channel 4 CAN-FD Configuration Register	CFDC4FDCFG	0000_0000h	1484h	8, 16, 32
Channel 4 CAN-FD Control Register	CFDC4FDCTR	0000_0000h	1488h	8, 16, 32
Channel 4 CAN-FD Status Register	CFDC4FDSTS	0000_0000h	148Ch	8, 16, 32
Channel 4 CAN-FD CRC Register	CFDC4FDCRC	0000_0000h	1490h	8, 16, 32
Reserve	-	-	1494h to 1497h	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
Channel 4 Bus Load Control Register	CFDC4BLCT	0000_0000h	1498h	8, 16, 32
Channel 4 Bus Load Status Register	CFDC4BLSTS	0000_0000h	149Ch	8, 16, 32
Channel 5 Data Bit Rate Configuration Register	CFDC5DCFG	0000_0000h	14A0h	8, 16, 32
Channel 5 CAN-FD Configuration Register	CFDC5FDCFG	0000_0000h	14A4h	8, 16, 32
Channel 5 CAN-FD Control Register	CFDC5FDCTR	0000_0000h	14A8h	8, 16, 32
Channel 5 CAN-FD Status Register	CFDC5FDSTS	0000_0000h	14ACh	8, 16, 32
Channel 5 CAN-FD CRC Register	CFDC5FDCRC	0000_0000h	14B0h	8, 16, 32
Reserve	-	-	14B4h to 14B7h	-
Channel 5 Bus Load Control Register	CFDC5BLCT	0000_0000h	14B8h	8, 16, 32
Channel 5 Bus Load Status Register	CFDC5BLSTS	0000_0000h	14BCh	8, 16, 32
Reserve	-	-	14C0h to 17FFh	-
Global Acceptance Filter List ID Register n	CFDGAFLIDn	0000_0000h	1800h + (n-1) x 0010h	8, 16, 32
Global Acceptance Filter List Mask Register n	CFDGAFLMn	0000_0000h	1804h + (n-1) x 0010h	8, 16, 32
Global Acceptance Filter List Pointer 0 Register n	CFDGAFLP0n	0000_0000h	1808h + (n-1) x 0010h	8, 16, 32
Global Acceptance Filter List Pointer 1 Register n	CFDGAFLP1n	0000_0000h	180Ch + (n-1) x 0010h	8, 16, 32
Reserve	-	-	1900h to 1FFFh	-
RX Message Buffer ID Register n	CFDRMIDn	0000_0000h	2000h + n x 0080h	8, 16, 32
RX Message Buffer Pointer Register	CFDRMPTRn	0000_0000h	2004h + n x 0080h	8, 16, 32
RX Message Buffer CAN-FD Status Register	CFDRMFDSTSn	0000_0000h	2008h + n x 0080h	8, 16, 32
RX Message Buffer Data Field p Register n	CFDRMDFp_n	0000_0000h	200Ch + p x 0004h + n x 0080h	8, 16, 32
RX FIFO Access ID Register n	CFDRFIDn	0000_0000h	6000h + n x 0080h	8, 16, 32
RX FIFO Access Pointer Register n	CFDRFPTRn	0000_0000h	6004h + n x 0080h	8, 16, 32
RX FIFO Access CAN-FD Status Register n	CFDRFFDSTSn	0000_0000h	6008h + n x 0080h	8, 16, 32
RX FIFO Access Data Field p Register n	CFDRFDFpn	0000_0000h	600Ch + p x 0004h + n x 0080h	8, 16, 32
Common FIFO Access ID Register n	CFDCFIDn	0000_0000h	6400h + n x 0080h	8, 16, 32
Common FIFO Access Pointer Register n	CFDCFPTRn	0000_0000h	6404h + n x 0080h	8, 16, 32
Common FIFO Access CAN-FD Control/Status Register n	CFDCFFDCSTSn	0000_0000h	6408h + n x 0080h	8, 16, 32
Common FIFO Access Data Field p Register n	CFDCFDFpn	0000_0000h	640Ch + p x 0004h + n x 0080h	8, 16, 32
Channel 0 TX History List Access Register 0	CFDTHLACC00	0000_0000h	8000h	8, 16, 32
Channel 0 TX History List Access Register 1	CFDTHLACC10	0000_0000h	8004h	8, 16, 32
Channel 1 TX History List Access Register 0	CFDTHLACC01	0000_0000h	8008h	8, 16, 32
Channel 1 TX History List Access Register 1	CFDTHLACC11	0000_0000h	800Ch	8, 16, 32
Channel 2 TX History List Access Register 0	CFDTHLACC02	0000_0000h	8010h	8, 16, 32
Channel 2 TX History List Access Register 1	CFDTHLACC12	0000_0000h	8014h	8, 16, 32
Channel 3 TX History List Access Register 0	CFDTHLACC03	0000_0000h	8018h	8, 16, 32
Channel 3 TX History List Access Register 1	CFDTHLACC13	0000_0000h	801Ch	8, 16, 32
Channel 4 TX History List Access Register 0	CFDTHLACC04	0000_0000h	8020h	8, 16, 32
Channel 4 TX History List Access Register 1	CFDTHLACC14	0000_0000h	8024h	8, 16, 32
Channel 5 TX History List Access Register 0	CFDTHLACC05	0000_0000h	8028h	8, 16, 32
Channel 5 TX History List Access Register 1	CFDTHLACC15	0000_0000h	802Ch	8, 16, 32
Reserve	-	-	8030h to 83FFh	-
RAM Test Page Access Register n	CFDRPGACCn	0000_0000h	8400h + n x 0004h	8, 16, 32
Reserve	-	-	8500h to FFFFh	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
TX Message Buffer ID Register n	CFDTMIDn	0000_0000h	1_0000h + n x 0080h	8, 16, 32
TX Message Buffer Pointer Register n	CFDTMPTRn	0000_0000h	1_0004h + n x 0080h	8, 16, 32
TX Message Buffer CAN-FD Control Register n	CFDTMFDCTRn	0000_0000h	1_0008h + n x 0080h	8, 16, 32
TX Message Buffer Data Field p Register n	CFDTMDFp_n	0000_0000h	1_000Ch + p x 0004h + n x 0080h	8, 16, 32

Note 1. The read access size is fixed at 32 bits.

7.9.2.2 CANFD Register Description

7.9.2.2.1 Channel n Nominal Bit Rate Configuration Register (CFDCnNCFG) (n = 0 to 5)

The CFDCnNCFG register is used to configure the transmission/reception nominal bit rate parameters of the channels. Do not write this register in CH_OPERATION or CH_SLEEP mode. Set this register in CH_RESET or CH_HALT mode.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 0000h + n x 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NTSEG2[6:0]							NTSEG1[7:0]								NSJW
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NSJW[5:0]						NBRP[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	NTSEG2[6:0]	0h	RW	Nominal Bit Rate Time Segment 2 Control 00h: Setting prohibited 01h: 2 Tq ⋮ 7Eh: 127 Tq 7Fh: 128 Tq
24 to 17	NTSEG1[7:0]	0h	RW	Nominal Bit Rate Time Segment 1 Control 00h: Setting prohibited 01h: 2 Tq 02h: 3 Tq ⋮ FEh: 255 Tq FFh: 256 Tq
16 to 10	NSJW[6:0]	0h	RW	Nominal Bit Rate Resynchronization Jump Width Control 00h: 1 Tq 01h: 2 Tq ⋮ 7Eh: 127 Tq 7Fh: 128 Tq
9 to 0	NBRP[9:0]	0h	RW	Nominal Bit Rate Prescaler Nominal bit rate prescaler division ratio setting

NBRP[9:0] bits (Nominal Bit Rate Prescaler)

The NBRP[9:0] bits are used to define the peripheral bus clock periods contained in a time quantum.

NSJW[6:0] bits (Nominal Bit Rate Resynchronization Jump Width Control)

The NSJW[6:0] bits set the synchronization jump width. A value from 1 to 128 time quanta can be set.

NTSEG1[7:0] bits (Nominal Bit Rate Time Segment 1 Control)

The NTSEG1[7:0] bits set the segment TSEG1 to compensate for edges on the CAN bus with a positive phase error. These bits contain the propagation segment. Configure a Tq value only between 2 and 256, inclusive. See **7.9.4.1.2 CAN Bit Timing** for more details.

NTSEG2[6:0] bits (Nominal Bit Rate Time Segment 2 Control)

The NTSEG2[6:0] bits set the segment TSEG2 to compensate for edges on the CAN bus with a negative phase error. Configure a Tq value only between 2 and 128, inclusive.

7.9.2.2.2 Channel n Control Register (CFDCnCTR) (n = 0 to 5)

The CFDCnCTR register controls the modes of the related channel. It is used to enable generation of interrupts if errors are detected on the CAN bus connected to this channel. It is also used to configure the channel in test mode.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 0004h + n x 0010h

Initial Value : 0000_0005h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ROM	CRCT	-	-	-	CTMS[1:0]		CTME	ERRD	BOM[1:0]		-	TDCVFIE	SOCOIE	EOCOIE	TAIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE	-	-	-	-	RTBO	CSLPR	CHMDC[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	ROM*[1]	0h	RW	Restricted Operation Mode Enable 0b: Restricted operation mode disabled 1b: Restricted operation mode enabled
30	CRCT	0h	RW	CRC Error Test Enable 0b: The first bit of reception ID field is not inverted 1b: The first bit of reception ID field is inverted
29 to 27	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
26, 25	CTMS[1:0]	0h	RW	Communication Test Mode Select 00b: Standard test mode 01b: Listen-only mode 10b: Self-test mode 0 (External loopback mode) 11b: Self-test mode 1 (Internal loopback mode)
24	CTME	0h	RW	Communication Test Mode Enable 0b: Communication test mode is disabled 1b: Communication test mode is enabled
23	ERRD	0h	RW	Error Display Mode Select 0b: Error flags are displayed only for the first error information after bits [14:8] in the CFDCnERFL register are all cleared 1b: Error flag for all error information are displayed
22, 21	BOM[1:0]	0h	RW	Bus-Off Recovery Mode Select 00b: ISO11898-1 compliant 01b: Entry to Channel Halt mode automatically at bus-off entry 10b: Entry to Channel Halt mode automatically at bus-off end 11b: Entry to Channel Halt mode (in bus-off state) by program request
20	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
19	TDCVFIE*[1]	0h	RW	Transceiver Delay Compensation Violation Interrupt Enable 0b: Transceiver delay compensation violation interrupt disabled 1b: Transceiver delay compensation violation interrupt enabled
18	SOCOIE	0h	RW	Successful Occurrence Counter Overflow Interrupt Enable 0b: Successful occurrence counter overflow interrupt disabled 1b: Successful occurrence counter overflow interrupt enabled
17	EOCOIE	0h	RW	Error Occurrence Counter Overflow Interrupt Enable 0b: Error occurrence counter overflow interrupt disabled 1b: Error occurrence counter overflow interrupt enabled
16	TAIE	0h	RW	Transmission Abort Interrupt Enable 0b: Transmission abort interrupt disabled 1b: Transmission abort interrupt enabled
15	ALIE	0h	RW	Arbitration Lost Interrupt Enable 0b: Arbitration lost interrupt disabled 1b: Arbitration lost interrupt enabled

Bit	Bit Name	Initial Value	R/W	Description
14	BLIE	0h	RW	Bus Lock Interrupt Enable 0b: Bus lock interrupt disabled 1b: Bus lock interrupt enabled
13	OLIE	0h	RW	Overload Interrupt Enable 0b: Overload interrupt disabled 1b: Overload interrupt enabled
12	BORIE	0h	RW	Bus-Off Recovery Interrupt Enable 0b: Bus-off recovery interrupt disabled 1b: Bus-off recovery interrupt enabled
11	BOEIE	0h	RW	Bus-Off Entry Interrupt Enable 0b: Bus-off entry interrupt disabled 1b: Bus-off entry interrupt enabled
10	EPIE	0h	RW	Error Passive Interrupt Enable 0b: Error passive interrupt disabled 1b: Error passive interrupt enabled
9	EWIE	0h	RW	Error Warning Interrupt Enable 0b: Error warning interrupt disabled 1b: Error warning interrupt enabled
8	BEIE	0h	RW	Bus Error Interrupt Enable 0b: Bus error interrupt disabled 1b: Bus error interrupt enabled
7 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3	RTBO	0h	RW	Forcible Return from Bus-Off When this bit is set to 1b, forcible return from the bus-off state is made. This bit is always read as 0b.
2	CSLPR	1h	RW	Channel Stop Mode 0b: Other than Channel Stop mode 1b: Channel Stop mode
1, 0	CHMDC[1:0]	1h	RW	Mode Select 00b: Channel Communication mode 01b: Channel Reset mode 10b: Channel Halt mode 11b: Setting prohibited

Note 1. These bits are not available in the classical CAN function.

CHMDC[1:0] bits (Mode Select)

The CHMDC[1:0] bits can be used to configure modes of the CAN channel.

CAN mode transitions are described in more details in **7.9.3.3 Channel Modes**.

Setting CHMDC[1:0] bits to 11b has no effect. When the CAN-FD module is in GL_HALT mode, these bits can only be set to 10b or 01b. These bits cannot be set in CH_SLEEP mode.

These bits can change automatically when transitioning to Halt mode by the CFDCnCTR.BOM settings.

If CPU write access to CFDCnCTR.CHMDC occurs at the same time when the CAN channel enters Halt mode (at the start of bus-off when CFDCnCTR.BOM[1:0] = 01b, or at the end of bus-off when CFDCnCTR.BOM[1:0] = 10b), then the CPU write access has the highest priority.

The CAN channel changes the value of CFDCnCTR.CHMDC within the Channel Control Registers for the specified cases only if the CFDCnCTR.CHMDC[1:0] value is 00b (Operation mode).

CSLPR bit (Channel Stop Mode)

When the CSLPR bit is 1b, a Sleep mode request is generated for the corresponding CAN channel.

When this bit is 0b, a request to exit Sleep mode is generated for the related CAN-FD channel.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_SLEEP mode.

RTBO bit (Forcible Return from Bus-Off)

When the protocol controller of the CAN channel enters bus-off state, you can force a recovery from bus-off state by setting the RTBO bit in the Channel Control Register to 1b.

The error state changes from bus-off state to integrating with a maximum delay of 1 CAN bit time.

When the RTBO bit is set to 1b, the REC and TEC registers are initialized and the Bus-Off Status bit (Channel Bus-off Status, CFDCnSTS.BOSTS) is set to 0b.

Registers other than the REC and TEC registers are not initialized by this command. Even if CFDCnCTR.BORIE is set, a bus-off recovery interrupt is not generated by this recovery from the bus-off state.

The RTBO bit cannot be set in CH_SLEEP mode. Setting this bit in any state other than bus-off state has no effect and the bit is cleared immediately. The read value is always 0b.

Return from the Bus-Off command should be used only when CFDCnCTR.BOM[1:0] is set to 00b.

Only write to this bit when the related CAN-FD channel is in CH_OPERATION mode. This bit is automatically cleared when set by software.

BEIE bit (Bus Error Interrupt Enable)

When the BEIE and the CFDCnERFL.BEF bits are both 1b, an error interrupt request is generated.

This bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

EWIE bit (Error Warning Interrupt Enable)

When the EWIE and the CFDCnERFL.EWF bits are both 1b, an error interrupt request is generated.

The EWIE bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

EPIE bit (Error Passive Interrupt Enable)

An error interrupt request is generated when the EPIE bit and the CFDCnERFL.EPF are both 1b.

The EPIE bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

BOEIE bit (Bus-Off Entry Interrupt Enable)

When the BOEIE and the CFDCnERFL.BOEF bits are both 1b, an error interrupt request is generated.

The BOEIE bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

BORIE bit (Bus-Off Recovery Interrupt Enable)

When the BORIE and the CFDCnERFL.BORF bits are both 1b, an error interrupt request is generated.

The BORIE bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

OLIE bit (Overload Interrupt Enable)

When the OLIE and the CFDCnERFL.OVLF bits are both 1b, an error interrupt request is generated.

Do not write to this bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

BLIE bit (Bus Lock Interrupt Enable)

When the BLIE and the CFDCnERFL.BLF bits are both 1b, an error interrupt request is generated.

Do not write to this bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

ALIE bit (Arbitration Lost Interrupt Enable)

When the ALIE and the CFDCnERFL.ALF bits are both 1b, an error interrupt request is generated.

Do not write to this bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

TAIE bit (Transmission Abort Interrupt Enable)

When the TAIE bit is 1b and a transmission is successfully aborted from the transmit buffer, an interrupt request is generated.

Do not write to this bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

EOCOIE bit (Error Occurrence Counter Overflow Interrupt Enable)

When the EOCOIE bit is 1b and the CFDCnFDSTS.EOCO bit belonging to the corresponding CAN channel is 1b, an error interrupt request is generated.

The EOCOIE bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

SOCOIE bit (Successful Occurrence Counter Overflow Interrupt Enable)

When the SOCOIE bit is 1b and the CFDCnFDSTS.SOCO bit belonging to the corresponding CAN channel is 1b, an error interrupt request is generated.

The SOCOIE bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

TDCVFIE bit (Transceiver Delay Compensation Violation Interrupt Enable)

When the TDCVFIE bit is 1b and the CFDCnFDSTS.TDCVF bit belonging to the corresponding CAN channel is 1b, an error interrupt request is generated.

The TDCVFIE bit cannot be set in CH_SLEEP mode.

Only write to this bit when the related CAN-FD channel is in CH_RESET mode. Do not set this bit when in Classical-only mode.

Note: This bit is not available in the classical CAN function.

BOM[1:0] bits (Bus-Off Recovery Mode Select)

The BOM[1:0] bits control the timing of the recovery from Bus-Off mode of the CAN-FD Channel.

Do not write to these bits in CH_SLEEP mode. Only write to these bits when the related CAN-FD channel is in CH_RESET mode.

ERRD bit (Error Display Mode Select)

The ERRD bit controls the display mode of the error flag bits [14:8] in the Channel Error Flag Register (CFDCnERFL).

If the ERRD bit is 0b and more than one errors occur at the same time, the error flag bits are set for all the errors that occurred at the same time. No further errors are flagged until the error flag bits [14:8] are cleared.

Do not write to the ERRD bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

CTME bit (Communication Test Mode Enable)

The CTME bit enables the channel test modes.

Do not write to this bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_HALT mode.

CTMS[1:0] bits (Communication Test Mode Select)

The CTMS[1:0] bits are used to select the required test mode.

Do not write to these bits in CH_SLEEP or CH_RESET mode. Only write to these bits when the related CAN-FD channel is in CH_HALT mode.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

CRCT bit (CRC Error Test Enable)

The CRCT bit checks the internal CRC generator logic of the protocol controller.

This bit inverts the first bit (ID bit) of the CAN message data stream that is being received, so that the internal generated CRC result does not match the received CRC value of the frame. Refer to the bit stuffing rule when using this feature for the possibility of receiving a stuff error (due to the inversion) rather than a CRC error.

The internal generated CRC value is always observed in the following registers:

- CFDCnERFL.CRCREG (classical CAN frames)
- CFDCnFDCRC.CRCREG (CAN-FD frames)

Some limitations exist when using this bit:

- It is not possible to use this feature with CAN nodes connected to the device externally, only with nodes connected to the internal CAN bus communication can be used
- One CAN node can send a reference message and the receiver node can invert one bit of the incoming bit stream

Note: The transmitter and receiver modes share the same CRC generator, therefore it is not necessary to consider the modes separately when testing this limitation.

The CRC Error Test mode is enabled if the CRCT (new control signal that inverts the first bit of the bit stream) and CTME bits are both 1b.

Do not write to the CRCT bit in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_HALT mode.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

ROM bit (Restricted Operation Mode Enable)

When the ROM and CTME bits are both 1b, the restricted operation mode is enabled. This mode should only be used in basic test mode (CFDCnCTR.CTMS[1:0] = 00b).

The ROM bit cannot be set in CH_SLEEP mode. Only write to this bit when the related CAN-FD channel is in CH_HALT mode.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. Do not set this bit when in Classical-only mode.

Note: This bit is not available in the classical CAN function.

7.9.2.2.3 Channel n Status Register (CFDCnSTS) (n = 0 to 5)

The CFDCnSTS register shows the mode, error, and transmission/reception status of the related channel together with its reception and transmission error count values.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 0008h + n x 0010h
Initial Value : 0000_0005h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEC[7:0]								REC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	ESIF	COMSTS	RECSTS	TRMSTS	BOSTS	EPSTS	CSLPSTS	CHLTSTS	CRSTSTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TEC[7:0]	0h	R	Transmission Error Count The transmit error counter (TEC) can be read.
23 to 16	REC[7:0]	0h	R	Reception Error Count The receive error counter (REC) can be read.
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8	ESIF*1	0h	RW	Error State Indication Flag 0b: No CANFD message whose ESI bit is recessive has been received 1b: At least one CANFD message whose ESI bit is recessive has been received
7	COMSTS	0h	R	Communication Status Flag 0b: Communication is not ready 1b: Communication is ready
6	RECSTS	0h	R	Receive Status Flag 0b: Bus idle, in transmission or bus off state 1b: In reception
5	TRMSTS	0h	R	Transmit Status Flag 0b: Bus idle or in reception 1b: In transmission or bus off state
4	BOSTS	0h	R	Bus-Off Status Flag 0b: Not in bus-off state 1b: In bus-off state
3	EPSTS	0h	R	Error Passive Status Flag 0b: Not in error passive state 1b: In error passive state
2	CSLPSTS	1h	R	Channel Stop Status Flag 0b: Not in Channel Sleep mode 1b: In Channel Sleep mode
1	CHLTSTS	0h	R	Channel Halt Status Flag 0b: Not in Channel Halt mode 1b: In Channel Halt mode
0	CRSTSTS	1h	R	Channel Reset Status Flag 0b: Not in Channel Reset mode 1b: In Channel Reset mode

Note 1. This bit is not available in the classical CAN function.

CRSTSTS bit (Channel Reset Status Flag)

The CRSTSTS bit indicates whether the related CAN channel is in Reset mode.

This bit is set automatically when the related CAN channel enters Channel Reset mode. When the mode is changed from Reset mode to Sleep mode, the CRSTSTS bit remains 1b.

This bit is cleared automatically when the related CAN channel exits the Channel Reset mode, except when changing to Sleep mode.

CHLTSTS bit (Channel Halt Status Flag)

The CHLTSTS bit indicates whether the related CAN channel is in Halt mode.

This bit is set automatically when the related CAN module enters Halt mode, and is cleared automatically when the related CAN module exits Halt mode.

CSLPSTS bit (Channel Stop Status Flag)

The CSLPSTS bit indicates whether the related CAN channel is in Sleep mode.

This bit is set automatically when the related CAN-FD channel enters Sleep mode, and is cleared automatically when the related CAN-FD channel exits Sleep mode.

EPSTS bit (Error Passive Status Flag)

The EPSTS bit indicates whether the related CAN-FD channel has entered the error passive state.

This bit is set automatically when the value of the CAN Transmission or Reception Counter Register exceeds the value of 7Fh.

This bit is cleared automatically when the related CAN-FD channel exits the error passive state or enters Reset mode.

BOSTS bit (Bus-Off Status Flag)

The BOSTS bit indicates whether the related CAN-FD channel has entered the error bus-off state.

This bit is set automatically when the value of the related CAN Transmission Error Count Register exceeds FFh and the related CAN-FD channel is in the bus-off state (CAN Transmission Error Count Register > FFh).

This bit is cleared automatically when the related CAN-FD channel exits bus-off state.

TRMSTS bit (Transmit Status Flag)

The TRMSTS bit indicates whether the related CAN-FD channel is transmitting a message.

This bit is set automatically when the related CAN-FD channel is operating as a transmitter node or is in the bus-off state.

This bit is cleared automatically when the related CAN-FD channel is in the bus-idle state or starts operating as a receiver node.

RECSTS bit (Receive Status Flag)

The RECSTS bit indicates whether the related CAN-FD channel is receiving a message.

This bit is set automatically when the related CAN-FD channel is operating as a receiver node.

This bit is cleared automatically when the related CAN-FD channel is in the bus-idle state or starts operating as a transmitter node.

COMSTS bit (Communication Status Flag)

The COMSTS bit indicates whether the related CAN-FD channel is ready for communication.

This bit is set automatically when the related CAN-FD channel is ready to perform communication following the detection of 11 consecutive recessive bits after exiting the Reset or Halt mode.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

Note: This bit is 1b during bus-off state.

ESIF bit (Error State Indication Flag)

The ESIF bit is set when the ESI bit is sampled recessive for a reception CAN message without any error. When in Loopback or Mirror mode, the self-transmitted messages are considered reception messages.

If a set from the CAN-FD channel occurs simultaneously with a clear by a write access, then the bit is set.

This bit is cleared by writing 0b to it. This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

Note: This bit is not available in the classical CAN function.

REC[7:0] bits (Reception Error Count)

The REC[7:0] bits increment or decrement the counter value according to error status of the CAN-FD channel during reception, and display the value of the REC error counter.

The value in bus-off state is indeterminate.

These bits are cleared automatically when the CAN-FD module enters GL_RESET or the CAN-FD channel is in CH_RESET mode.

TEC[7:0] bits (Transmission Error Count)

The TEC[7:0] bits increment or decrement the counter value according to error status of the CAN-FD channel during transmission, and display the value of the TEC error counter.

Only write to these bits when in test mode and CAN-FD channel is in CH_HALT mode.

Do not read data from this register while CFDCnCTR.TRWE is set.

These bits are cleared automatically when CAN-FD module is in GL_RESET or CAN-FD channel is in CH_RESET mode.

7.9.2.2.4 Channel n Error Flag Register (CFDCnERFL) (n = 0 to 5)

The CFDCnERFL register shows the status of various error conditions detectable regardless of the setting of the related CAN Channel Error Interrupt Enable Register. It also shows the status of the various bus errors detectable by the CAN channel. Refer to the CAN specification (ISO 11898-1) for a description of error occurrence conditions.

Only write to this register when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 000Ch + n x 0010h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	CRCREG[14:0]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR	ALF	BLF	OVLF	BORF	BOEF	EPF	EWF	BEF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
30 to 16	CRCREG[14:0]	0h	R	CRC Calculation Data (CRC length: 15 bits) These bits show the CRC value calculated based on the transmit message or receive message.
15	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
14	ADERR	0h	RW*1	Acknowledge Delimiter Error Flag 0b: Channel acknowledge delimiter error not detected 1b: Channel acknowledge delimiter error detected
13	B0ERR	0h	RW*1	Dominant Bit Error Flag 0b: Channel dominant bit error not detected 1b: Channel dominant bit error detected
12	B1ERR	0h	RW*1	Recessive Bit Error Flag 0b: Channel recessive bit error not detected 1b: Channel recessive bit error detected
11	CERR	0h	RW*1	CRC Error Flag 0b: Channel CRC error not detected 1b: Channel CRC error detected
10	AERR	0h	RW*1	Acknowledge Error Flag 0b: Channel acknowledge error not detected 1b: Channel acknowledge error detected
9	FERR	0h	RW*1	Form Error Flag 0b: Channel form error not detected 1b: Channel form error detected
8	SERR	0h	RW*1	Stuff Error Flag 0b: Channel stuff error not detected 1b: Channel stuff error detected
7	ALF	0h	RW*1	Arbitration Lost Flag 0b: Channel arbitration-lost not detected 1b: Channel arbitration-lost detected
6	BLF	0h	RW*1	Bus Lock Flag 0b: Channel bus lock not detected 1b: Channel bus lock detected
5	OVLF	0h	RW*1	Overload Flag 0b: Channel overload not detected 1b: Channel overload detected

Bit	Bit Name	Initial Value	R/W	Description
4	BORF	0h	RW*1	Bus-Off Recovery Flag 0b: Channel bus-off recovery not detected 1b: Channel bus-off recovery detected
3	BOEF	0h	RW*1	Bus-Off Entry Flag 0b: Channel bus-off entry not detected 1b: Channel bus-off entry detected
2	EPF	0h	RW*1	Error Passive Flag 0b: Channel error passive not detected 1b: Channel error passive detected
1	EWF	0h	RW*1	Error Warning Flag 0b: Channel error warning not detected 1b: Channel error warning detected
0	BEF	0h	RW*1	Bus Error Flag 0b: Channel bus error not detected 1b: Channel bus error detected

Note 1. To clear each flag of this register, software must write 0b to the corresponding bit. These flags cannot be set to 1 by software.

BEF bit (Bus Error Flag)

The BEF bit indicates a detection of a CAN channel bus error state, flagged by bits [14:8] in this register.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

This bit is set automatically when a bus error is detected, and is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

If a set from the CAN channel occurs simultaneously with a clear by a write access, then the bit is set.

EWF bit (Error Warning Flag)

The EWF bit indicates whether an error warning condition has been detected for the CAN channel.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

This bit is set automatically when either TEC or REC exceeds 5Fh.

The setting of this bit only occurs when the TEC or REC initially exceeds 5Fh. Therefore, if the TEC or REC remains > 5Fh and the EWF bit is cleared by software, it is not set again until both the TEC and REC go below 60h and either TEC or REC crosses over again from a value 5Fh to a value > 5Fh.

If a set condition occurs simultaneously with a clear condition, then the bit is set. It is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

EPF bit (Error Passive Flag)

The EPF bit indicates a detection of a CAN channel error passive state.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

This bit is set automatically when the CAN error state becomes error passive state.

The setting of this bit only occurs when the TEC or REC initially exceeds 7Fh. Therefore, if the TEC or REC remains > 7Fh and the bit is cleared by software, it is not set again until both the TEC and REC go below 80h and either TEC or REC crosses over again from a value ≤ 7Fh to a value > 7Fh.

If a set condition occurs simultaneously with a clear condition, then the bit is set. It is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

BOEF bit (Bus-Off Entry Flag)

The BOEF bit indicates a detection of a CAN channel bus-off entry state.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

This bit is set automatically when the CAN error state enters the bus-off state.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If a set condition occurs simultaneously with a clear condition, then the bit is set.

BORF bit (Bus-Off Recovery Flag)

The BORF bit indicates a detection of a CAN channel bus-off recovery state.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

This bit is set automatically if CAN channel recovers from bus-off state in the following conditions:

- When CFDCnCTR.BOM[1:0] is 00b and normal recovery (11 consecutive recessive bits × 128 times detected) occurs
- When CFDCnCTR.BOM[1:0] is 10b and normal recovery (11 consecutive recessive bits × 128 times detected) occurs
- When CFDCnCTR.BOM[1:0] is 11b and normal recovery (11 consecutive recessive bits × 128 times detected) occurs.

The bit is not set if CAN channel recovers from bus-off state in the following conditions:

- When CAN Reset mode is requested
- When CFDCnCTR.RTBO is set to 1b (the CAN channel returns to error active)
- When CFDCnCTR.BOM[1:0] is 01b
- When CFDCnCTR.BOM[1:0] is 11b and a halt request is asserted before the CAN channel reaches the end of the bus- off state.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If a set condition occurs simultaneously with a clear condition, the flag is set.

OVLF bit (Overload Flag)

The OVLF flag indicates a detection of a CAN channel overload state.

The OVLF bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

This bit is set automatically when an overload condition is detected. If a set condition occurs simultaneously with a clear condition, then the bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

BLF bit (Bus Lock Flag)

The BLF bit indicates a detection of a CAN channel bus lock condition.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

This bit is set automatically when 32 consecutive dominant bits are detected on the CAN bus while the CAN channel is in Operation mode.

If a set condition occurs simultaneously with a clear condition, then the bit is set. It is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

It is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

ALF bit (Arbitration Lost Flag)

The ALF bit indicates a detection of a CAN channel bus arbitration-lost condition.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

The bit is set automatically when an arbitration-lost condition is detected on the CAN bus while the CAN channel is in Operation mode.

If a set condition occurs simultaneously with a clear condition, then the bit is set. It is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

SERR bit (Stuff Error Flag)

The SERR bit indicates a detection of a CAN stuff error.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when a stuff error is detected. If CFDCnCTR.ERRD bit is 1b and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0b and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 000_0000b.

FERR bit (Form Error Flag)

The FERR bit indicates a detection of a CAN form error.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when a form error is detected. If CFDCnCTR.ERRD bit is 1b and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0b and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 000_0000b.

AERR bit (Acknowledge Error Flag)

The AERR bit indicates a detection of a CAN acknowledge error.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.

2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when an acknowledge error is detected. If CFDCnCTR.ERRD bit is 1b and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0b and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 000_0000b.

CERR bit (CRC Error Flag)

The CERR bit indicates a detection of a CAN CRC error.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when a CRC error is detected. If CFDCnCTR.ERRD bit is 1b and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0b and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 000_0000b.

B1ERR bit (Recessive Bit Error Flag)

The B1ERR bit indicates a detection of a recessive bit error.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when a recessive bit error (expected recessive bit, sampled as dominant bit) is detected. If CFDCnCTR.ERRD bit is 1b and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0b and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 000_0001b.

B0ERR bit (Dominant Bit Error Flag)

The B0ERR bit indicates a detection of a dominant bit error.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when a dominant bit error (expected dominant bit, sampled as recessive bit) is detected. If CFDCnCTR.ERRD bit is 1b and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0b and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 000_0000b.

ADERR bit (Acknowledge Delimiter Error Flag)

The ADERR bit indicates a detection of an acknowledge delimiter bit error.

This bit is cleared by writing 0b to it, and can only be set by CAN-FD module logic. Writing 1b has no effect.

To clear this bit, use the following sequence:

1. Clear the corresponding flag bit.
2. Read if the flag bit is cleared.
3. If yes, continue, else go back to step 1.

This bit is set automatically when a form error is detected during the acknowledge delimiter state of frame transmission. If CFDCnCTR.ERRD bit is 1b and if the set and clear conditions occur at the same time for this bit, then this bit is set.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode. If CFDCnCTR.ERRD bit is 0b and the set and clear conditions occur at the same time for this bit, then it is cleared if a bit at the error flag bits [14:8] in the CFDCnERFL register is already set. Otherwise, this bit is set if the error flag bits [14:8] in the CFDCnERFL register is 000_0000b.

CRCREG[14:0] bits (CRC Calculation Data (CRC length: 15 bits))

The CRCREG[14:0] bits read the calculated CRC value when CFDCnCTR.CTME bit is 1b for the channel.

If CFDCnCTR.CTME bit is 0b, then these bits are always read as 0b.

These bits show the CAN2.0 CRC value calculated by the CAN-FD channel logic when the CTME bit is enabled.

The CFDCnERFL.CRCREG value is updated in the first bit of the CRC field of the CAN frame (reception and transmission).

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

7.9.2.2.5 Channel n Data Bit Rate Configuration Register (CFDCnDCFG) (n = 0 to 5)

This register is not available in the classical CAN function.

The CFDCnDCFG register configures the transmission/reception data bit rate parameters of the channels.

The channel of Classical-only mode does not perform configuration of this register.

Do not write to this register in CH_OPERATION or CH_SLEEP mode. Only write to this register in CH_RESET or CH_HALT mode.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1400h + n x 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	DSJW[3:0]				-	-	-	-	DTSEG2[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	DTSEG1[4:0]					DBRP[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
27 to 24	DSJW[3:0]	0h	RW	Data Bit Rate Resynchronization Jump Width Control 0h: 1 Tq 1h: 2 Tq ⋮ Eh: 15 Tq Fh: 16 Tq
23 to 20	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
19 to 16	DTSEG2[3:0]	0h	RW	Data Bit Rate Time Segment 2 Control 0h: Setting prohibited 1h: 2 Tq 2h: 3 Tq 3h: 4 Tq ⋮ Eh: 15 Tq Fh: 16 Tq
15 to 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12 to 8	DTSEG1[4:0]	0h	RW	Data Bit Rate Time Segment 1 Control 00h: Setting prohibited 01h: 2 Tq 02h: 3 Tq 03h: 4 Tq ⋮ 1Eh: 31 Tq 1Fh: 32 Tq
7 to 0	DBRP[7:0]	0h	RW	Data Bit Rate Prescaler Division Ratio Setting When the set value = P (0 to 255), the data bit rate prescaler divides fCAN by (P + 1).

DBRP[7:0] bits (Data Bit Rate Prescaler Division Ratio Setting)

The DBRP[7:0] bits define the peripheral bus clock periods contained in a time quantum.

DTSEG1[4:0] bits (Data Bit Rate Time Segment 1 Control)

The DTSEG1[4:0] bits set the segment TSEG1 to compensate for edges on the CAN bus with a positive phase error. A value from 2 to 32 time quanta can be set.

The DTSEG1[4:0] bits are also used to set the propagation segment.

Do not write any other value to these bits. See **7.9.4.1.2 CAN Bit Timing** for more details.

DTSEG2[3:0] bits (Data Bit Rate Time Segment 2 Control)

The DTSEG2[3:0] bits set the segment TSEG2 to compensate for edges on the CAN bus with a negative phase error. A value from 2 to 16 time quanta can be set.

Do not write any other value to these bits.

DSJW[3:0] bits (Data Bit Rate Resynchronization Jump Width Control)

The DSJW[3:0] bits set the synchronization jump width. A value from 1 to 16 time quanta can be set.

7.9.2.2.6 Channel n CAN-FD Configuration Register (CFDCnFDCFG) (n = 0 to 5)

The CFDCnFDCFG register configures which communication direction (transmitter/receiver) errors are counted.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1404h + n x 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDTE	CLOE	REFE	FDOE	-	GWBRS	GWFDF	GWEN	TDCO[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	ESIC	TDCE	TDCOC	-	-	-	-	-	EOCCFG[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CFDTE	0h	RW	CAN-FD Frame Distinction Enable 0b: CAN-FD frame distinction disabled 1b: CAN-FD frame distinction enabled
30	CLOE*2	0h	RW	Classical CAN-Only Enable*1 0b: Classical-only mode disabled 1b: Classical-only mode enabled
29	REFE	0h	RW	RX Edge Filter Enable 0b: RX edge filter disabled 1b: RX edge filter enabled
28	FDOE*2	0h	RW	FD-Only Enable 0b: FD-only mode disabled 1b: FD-only mode enabled
27	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
26	GWBRS*2	0h	RW	Gateway BRS Configuration Bit 0b: Gateway frame is transmitted with BRS = 0 1b: Gateway frame is transmitted with BRS = 1
25	GWFDF*2	0h	RW	Gateway FDF Configuration Bit 0b: Gateway frame is transmitted as Classical CAN frame 1b: Gateway frame is transmitted as CAN-FD frame
24	GWEN*2	0h	RW	CAN2.0, CAN-FD Multi Gateway Enable 0b: Multi gateway disabled 1b: Multi gateway enabled
23 to 16	TDCO*2[7:0]	0h	RW	Transceiver Delay Compensation Offset These bits are set to the transmitter delay compensation offset value.
15 to 11	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10	ESIC*2	0h	RW	Error State Indication Configuration 0b: The ESI bit in the frame represents the error state of the node itself 1b: The ESI bit in the frame represents the error state of the message buffer if the node itself is not in error passive. If the node is in error passive, then the ESI bit is driven by the node itself.
9	TDCE*2	0h	RW	Transceiver Delay Compensation Enable 0b: Transceiver delay compensation disabled 1b: Transceiver delay compensation enabled
8	TDCOC*2	0h	RW	Transmitter Delay Compensation Offset Configuration 0b: Measurement and offset 1b: Offset-only
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	EOCCFG[2:0]	0h	RW	Error Occurrence Counter Configuration 000b: All transmit messages and receive messages 001b: All transmit messages 010b: All receive messages 011b: Setting prohibited 100b: Only data phase of transmitted or received CANFD message 101b: Only data phase of transmitted CANFD message 110b: Only data phase of received CANFD message 111b: Setting prohibited

Note 1. This bit is set to 1b for this LSI with CANFD disabled.

Note 2. These bits are not available in the classical CAN function.

EOCCFG[2:0] bits (Error Occurrence Counter Configuration)

The EOCCFG[2:0] bits select which type of CAN frame configuration and direction, including protocol errors are counted.

Do not write to these bits in CH_OPERATION or CH_SLEEP mode.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

TDCOC bit (Transmitter Delay Compensation Offset Configuration)

The TDCOC bit selects which offset is used when defining the position of the secondary sample point (SSP) for the CAN- FD channel. If the bit is set to 0b, the position of the SSP is the measured transceiver delay plus the fixed offset. If the bit is 1b, the position of the SSP is defined only by the offset.

Do not write to this bit in CH_OPERATION or CH_SLEEP mode.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode. Do not set this bit when in Classical-only mode.

This bit is not available in the classical CAN function.

TDCE bit (Transceiver Delay Compensation Enable)

The TDCE bit enables the transceiver delay compensation for the CAN-FD channel.

Do not write to this bit in CH_OPERATION or CH_SLEEP mode.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode. Do not set this bit when in Classical-only mode.

This bit is not available in the classical CAN function.

ESIC bit (Error State Indication Configuration)

Bus controllers that are used as CAN-to-CAN gateway support that in every forwarded CAN-FD message. The ESI flag does not change to reflect the status of the gateway, bridge, or router but instead the flag is sent as it was in the original message.

The ESIC bit controls the transmission of either the ESI flag information or the message of ESI flag information (CFDCFFDCSTSn.CFESI or CFDTMFDCTRn.TMESI).

Do not write to this bit in CH_OPERATION or CH_SLEEP mode.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode. Do not set this bit when in Classical-only mode.

This bit is not available in the classical CAN function.

TDCO[7:0] bits (Transceiver Delay Compensation Offset)

The TDCO[7:0] bits set the secondary sample point offset. How this value is used, depends on the CFDCnFDCFG.TDCOC setting.

If CFDCnFDCFG.TDCOC = 0b, the transceiver delay compensation result is equal to the Trv_Delay (measured delay) + the value in CFDCnFDCFG.TDCO, rounded down to the nearest integer number of time quanta. Otherwise, the result is equal to the value in CFDCnFDCFG.TDCO. See **7.9.4.1.5 Transmitter Delay Compensation** for details on how CFDCnFDCFG.TDCO is used.

The actual offset value is interpreted as TDCO + 1. For example, if 4 is set in TDCO, the offset is 5 clock cycles. Clock cycle is 1 cycle of CAN channel DLL clock.

Do not write to the TDCO[7:0] bits in CH_OPERATION or CH_SLEEP mode.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode. Do not set to this bit when in Classical-only mode.

This bit is not available in the classical CAN function.

GWEN bit (CAN2.0, CAN-FD Multi Gateway Enable)

When the GWEN bit is enabled, a multi gateway is enabled. Message received on one node can be routed to another node using the COM FIFO when they are configured as gateway FIFO (CFDCFCCn.CFM[1:0] = 10b). Furthermore, when TX Queue is set as Gateway mode, the message received on one node can be stored in TX Queue and can be sent to another node.

The FDF and BRS bits of the routed message can be changed by the configuration value of the CFDCnFDCFG.GWFDF and CFDCnFDCFG.GWBRS bits. By this, the transmitted value of these bits can be replaced.

Example)

CFDCnFDCFG.GWEN = 1b on channel y
CFDCnFDCFG.GWFDF = 1b

If a Classical CAN frame is received on channel x and routed to a gateway FIFO or TX Queue of channel y, then this CAN frame is sent on channel y as a CAN-FD frame because of the CFDCnFDCFG.GWFDF bit.

Table 7.9-5 shows how the message information is changed depending on the received and configured data.

Table 7.9-5 Modified Message Information by Received and Configured Data

Routed CAN Frame	Routed Received DLC	CAN BRS Bit	Configured CFDCnFDCFG.GWFDF Bit	Gateway Message DLC	Gateway Message BRS Bit	Gateway Message Frame Type
CAN2.0	≤ 8	N/A	1	≤ 8	Based on configuration of CFDCnFDCFG.GWBRS	CAN-FD
CAN2.0	> 8	N/A	1	= 8	Based on configuration of CFDCnFDCFG.GWBRS	CAN-FD
CAN-FD	≤ 8	None	1	≤ 8	Based on configuration of CFDCnFDCFG.GWBRS	CAN-FD
CAN-FD	> 8	None	1	> 8	Based on configuration of CFDCnFDCFG.GWBRS	CAN-FD
CAN2.0	≤ 8	N/A	0	≤ 8	N/A	CAN2.0
CAN2.0	> 8	N/A	0	> 8	N/A	CAN2.0
CAN-FD	≤ 8	None	0	≤ 8	N/A	CAN2.0
CAN-FD	> 8	None	0	= 8	N/A	CAN2.0

Note: This gateway is limited to an 8-byte data payload for different frame type. If routing and target frame type is the same, the data length code (DLC) value remains the same. If the source frame is a CAN-FD with more than 8 data bytes, then on classical destination node, the data payload is reduced to 8 bytes. Only the first 8 bytes of data perform gateway transmission.

Note: Transmission buffers other than the gateway FIFO are not affected by this feature.

Do not route remote frames through the gateway when CFDCnFDCFG.GWEN is set. When a destination node is CFDCnFDCFG.FDOE = 1b, set CFDCnFDCFG.GWEN and CFDCnFDCFG.GWFDF to 1b. When a destination node is CFDCnFDCFG.CLOE = 1b, set CFDCnFDCFG.GWEN = 1b and CFDCnFDCFG.GWFDF = 0b.

Do not write to this bit in CH_OPERATION or CH_SLEEP mode.

Only write to this bit when the related CAN-FD channel is in CH_RESET mode.

This bit is not available in the classical CAN function.

GWFDF bit (Gateway FDF Configuration Bit)

When the GWEN bit is set to 1b, the FDF bit of the transmitting gateway frame is replaced by the value of the GWFDF bit.

Do not write to this bit in CH_OPERATION or CH_SLEEP mode.

Only write to these bits when the CAN-FD module is in CH_RESET mode. Do not set this bit when in Classical-only mode.

This bit is not available in the classical CAN function.

GWBRS bit (Gateway BRS Configuration Bit)

When the GWEN bit is set to 1b, the BRS bit of the transmitting gateway frame is replaced by the value of CFDCnFDCFG.GWBRS.

In classical CAN frames, the GWBRS bit is invalid.

Do not write to this bit in CH_OPERATION or CH_SLEEP mode.

Only write to this bit when the related CAN-FD channel is in CH_RESET mode. Do not set this bit when in Classical-only mode.

This bit is not available in the classical CAN function.

FDOE bit (FD-Only Enable)

The FDOE bit enables the reception and transmission of CAN-FD-only frames. If enabled, communication in Classical CAN frame format is disabled. Transmission of Classical CAN frames is not possible because the FDF bit of the message buffer is a don’t care (CFDCFFDCSTSn.CFFDF/CFDTMFDCTRn.TMFDF).

If messages with Classical CAN frame format are received, the protocol controller treats them as invalid frames and response with error frames. When a Classical CAN frame is configured for transmitting, the FDF bit is sent as recessive, therefore an FD frame is sent. If the data length code (DLC) is configured of greater than 8 bytes, the remaining data bytes are padded with CCh.

The FDOE bit cannot be written in CH_OPERATION, CH_HALT or CH_SLEEP mode.

Do not set CFDCnFDCFG.FDOE and CFDCnFDCFG.CLOE simultaneously.

This bit is not available in the classical CAN function.

REFE bit (RX Edge Filter Enable)

The REFE bit enables the RX edge filter during the IDLE detection (bus integration). When the bit is enabled, two consecutive dominant time quanta are required to detect a synchronization edge.

The REFE bit cannot be written in CH_OPERATION, CH_HALT and CH_SLEEP mode. Do not set this bit when in Classical-only mode and when CFDCnFDCFG.CFDTE = 0b (disabled CAN-FD frame distinction).

CLOE bit (Classical CAN-Only Enable)

The CLOE bit enables the Classical CAN-only mode. If this bit is 1b, the protocol controller can only send classical frames and response with a form or CRC error on FD frames.

Do not set CFDCnFDCFG.CLOE and CFDCnFDCFG.FDOE simultaneously.

CFDCnFDCFG.CLOE	CFDCnFDCFG.FDOE	Channel mode
0	0	CAN-FD mode (Dual mode)
0	1	FD-only mode
1	0	Classical CAN-only mode
1	1	Reserved

Do not write to this bit in CH_OPERATION, CH_HALT or CH_SLEEP mode.

Only write to these bits when the CAN-FD channel is in CH_RESET mode.

This bit is not available in the classical CAN function.

CFDTE bit (CAN-FD Frame Distinction Enable)

The CFDTE bit enables the CAN-FD frame distinction function. The CFDTE bit is required for Classical CAN-only mode (CFDCnFDCFG.CLOE = 1b).

If this bit is 0b, the protocol controller can only send Classical frames and response with a Form or CRC error on FD frames.

If this bit is 1b, the protocol controller behaves according to the ISO 11898-1 (DIS 2015) specification. If the FDF bit is detected recessive, then the protocol controller enters the protocol exception state, and attempts to integrate back to the CAN communication.

Do not write this bit in CH_OPERATION, CH_HALT or CH_SLEEP mode.

Only write to these bits when the related CAN-FD channel is in CH_RESET mode.

7.9.2.2.7 Channel n CAN-FD Control Register (CFDCnFDCTR) (n = 0 to 5)

The CFDCnFDCTR register (n = 0 to 5) controls the error and successful occurrence counters.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1408h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SOCCLR	EOCCLR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	SOCCLR	0h	RW	Successful Occurrence Counter Clear 0b: No successful occurrence counter clear 1b: Clear successful occurrence counter
0	EOCCLR	0h	RW	Error Occurrence Counter Clear 0b: No error occurrence counter clear 1b: Clear error occurrence counter

EOCCLR bit (Error Occurrence Counter Clear)

The EOCCLR bit is used to clear the error occurrence counter.

Do not write to this bit in CH_SLEEP or CH_RESET mode. The read value is always 0b.

This bit is cleared automatically by the CAN-FD logic and when the related CAN-FD channel is in CH_RESET mode.

SOCCLR bit (Successful Occurrence Counter Clear)

The SOCCLR bit is used to clear the successful occurrence counter.

Do not write to this bit in CH_SLEEP or CH_RESET mode. The read value is always 0b.

This bit is cleared automatically by the CAN-FD logic and when the related CAN-FD channel is in CH_RESET mode.

7.9.2.2.8 Channel n CAN-FD Status Register (CFDCnFDSTS) (n = 0 to 5)

The CFDCnFDSTS register (n = 0 to 5) indicates the transceiver compensation delay result and its related FIFO message lost status.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 140Ch + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SOC[7:0]								EOC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TDCVF	-	-	-	-	-	SOCO	EOCO	TDCR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	SOC[7:0]	0h	R	Successful Occurrence Counter These bits show the successful occurrence counter value.
23 to 16	EOC[7:0]	0h	R	Error Occurrence Counter These bits show the error occurrence counter value.
15	TDCVF	0h	RW	Transceiver Delay Compensation Violation Flag 0b: Transceiver delay compensation violation has not occurred 1b: Transceiver delay compensation violation has occurred
14 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	SOCO	0h	RW	Successful Occurrence Counter Overflow Flag 0b: Successful occurrence counter has not overflowed 1b: Successful occurrence counter has overflowed
8	EOCO	0h	RW	Error Occurrence Counter Overflow Flag 0b: Error occurrence counter has not overflowed 1b: Error occurrence counter has overflowed
7 to 0	TDCR[7:0]	0h	R	Transceiver Delay Compensation Result

TDCR[7:0] bits (Transceiver Delay Compensation Result)

The TDCR[7:0] bits are set when the transceiver delay has been measured.

The measured delay is a multiple of the CAN channel DLL clock. The result depends on the CFDCnFDCFG.TDCOC configuration and the offset value in CFDCnFDCFG.TDCO. See **7.9.4.1.5 Transmitter Delay Compensation** for details on how this value is derived.

The TDCR[7:0] bits are updated at the falling edge between the FDF bit and the RES bit when CFDCnFDCFG.TDCOC = 0b and the transceiver delay compensation is enabled (CFDCnFDCFG.TDCE = 1b).

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

EOCO bit (Error Occurrence Counter Overflow Flag)

The EOCO bit indicates whether the related CAN channel error occurrence counter has overflowed. This bit is cleared by writing 0b to it. Writing 1b has no effect.

This bit is set automatically when CFDCnFDSTS.EOC is FFh and a CAN bus error is detected based on the configuration defined in CFDCnFDCFG.EOCCFG.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Do not use the bit clear instruction to clear this bit.

SOCO bit (Successful Occurrence Counter Overflow Flag)

The SOCO bit indicates whether the related CAN channel successful occurrence counter has overflowed. This bit is cleared by writing 0b to it. Writing 1b has no effect.

This bit is set automatically when CFDCnFDSTS.SOC is FFh and a successful message reception or successful message transmission occurs.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Do not use the bit clear instruction to clear this bit.

TDCVF bit (Transceiver Delay Compensation Violation Flag)

The CAN-FD module captures internally the transmitted data bit-by-bit. This data is then compared against the received CAN bus level which is delayed by the transceiver loop delay.

The transceiver delay has some variations depending on the physical parameters such as temperature. The result bit CFDCnFDSTS.TDCR is updated by each message. However, temporary maximum delay violation could be missed. Therefore, the TDCVF bit captures this violation.

This bit is cleared by writing 0b to it. Writing 1b has no effect.

This bit is set automatically when the transceiver delay compensation is greater than the maximum delay compensation (6 data bit times - 2 clk_dlc) and the internal bit is overrun.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Do not use the bit clear instruction to clear this bit.

EOC[7:0] bits (Error Occurrence Counter)

The EOC[7:0] bits are used together with the SOC[7:0] bits to support an option for host-controlled fall-back to payload bit rate identical to arbitration bit rate when messages utilizing the reduced payload bit length have significant higher error rates compared to other messages.

This higher error rate can be detected depending on the configuration of the CFDCnFDCFG.EOCCFG bits.

The EOC[7:0] bits are set only by CAN-FD module logic. These bits are cleared by writing 1b to CFDCnFDCTR.EOCCLR. Writing any other value has no effect.

These bits are updated when an error occurs, according to the configuration of the CFDCnFDCFG.EOCCFG bits. When the counter reaches the value of FFh, the update stops.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

SOC[7:0] bits (Successful Occurrence Counter)

The SOC[7:0] bits are used together with the EOC[7:0] bits to support an option for host-controlled fall-back to payload bit rate identical to arbitration bit rate when messages utilizing the reduced payload bit length have significant higher error rates compared to other messages.

The SOC[7:0] bits are set only by CAN-FD module logic. Writing any other value has no effect.

These bits are updated when the occurrence of any error-free messages on the bus is detected through reception or transmission. When the counter reaches the value of FFh, the update stops. In Loopback mode, the counter is incremented twice.

These bits are cleared by writing 1b to CFDCnFDCTR.SOCCLR.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

7.9.2.2.9 Channel n CAN-FD CRC Register (CFDCnFDCRC) (n = 0 to 5)

The CFDCnFDCRC register (n = 0 to 5) holds the CRC value calculated for the CAN-FD frame.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1410h + n x 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	SCNT[3:0]				-	-	-	-	CRCREG[20:16]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRCREG[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read.
28 to 25	SCNT[3:0]	0h	R	Stuff Bit Count These bits shows the stuff bit count (mod 8) for the CAN-FD frame.
24 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read.
20 to 0	CRCREG[20:0]	0h	R	CRC Register Value These bits show the CRC value calculated for the CAN-FD frame.

CRCREG[20:0] bits (CRC Register Value)

The CRCREG[20:0] bits contain the CRC value calculated by the CAN-FD channel logic when the CFDCnCTR.CTME bit is enabled.

The CFDCnFDCRC.CRCREG value is updated in the first bit of the CRC field of the CAN-FD frame (reception and transmission).

When the CFDCnCTR.CTME bit is 0b, the CRCREG[20:0] bits are always read as 0b.

When bit 17th of the CRC field is used, CRCREG[20:17] are always read as 0b.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

SCNT[3:0] bits (Stuff Bit Count)

The SCNT[3:0] bits contain the stuff count value of the CAN-FD frame. These bits indicate the number of inserted stuff bits (modulo 8, Gray-coded) for a CAN-FD frame when the CFDCnCTR.CTME bit is enabled in CFDCnFDCRC.SCNT[3:1]. SCNT[0] is the parity bit.

When the CFDCnCTR.CTME bit is 0, the SCNT[3:0] bits are always read as 0b.

The SCNT value is updated in the first bit of CRC field of the CAN-FD frame (reception and transmission).

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

7.9.2.2.10 Global IP Version Register (CFDGIPV)

The CFDGIPV register shows the release version of the CAN-FD module.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 0080h
Initial Value : 1124_0043h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	PSI[13:0]													
Initial Value	0	0	0	1	0	0	0	1	0	0	1	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	IPT[1:0]		IPV[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read.
29 to 16	PSI[13:0]	1124h	R	Parameter Status Information These bits show the IP configuration. Bits Description [29:27] Number of channel (010b: 6 channels) [26:24] Number of TXMB (001b: 32/channel) [23:21] Number of AFL entry (001b: 128/channel) [20:17] Number of pool buffer (0010b: 64/channel) [16] Same ID overwrite function of TXQ (0b: Support)
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read.
9, 8	IPT[1:0]	0h*1	R	IP Type These bits show the IP type used in the product. 00b: CAN IP 01b: CAN-FD IP 1xb: Reserved
7 to 0	IPV[7:0]	43h	R	IP Version These bits show the main release number.

Note 1. This value depends on the package type.

7.9.2.2.11 Global Configuration Register (CFDGCFG)

The CFDGCFG register is used to select the transmission priority to be used for all the TX message buffers and the clock source for the CAN protocol engine of all CAN channels. The CFDGCFG register is also used to select the source for the timestamp clock and to configure the frequency for the timestamp clock and interval timer reference clock.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 0084h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ITRCP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSBTCS[2:0]			TSSS	TSP[3:0]				-	-	CMPOC	DCS	MME	DRE	DCE	TPRI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	ITRCP[15:0]	0h	RW	Interval Timer Reference Clock Prescaler FIFO interval timer prescaler value
15 to 13	TSBTCS[2:0]	0h	RW	Timestamp Bit Time Channel Select 000b: Select clock from channel 0 001b: Select clock from channel 1 010b: select clock from Channel 2 011b: select clock from Channel 3 100b: select clock from Channel 4 101b: select clock from Channel 5 Others: Setting prohibited
12	TSSS	0h	RW	Timestamp Source Select 0b: Source clock for timestamp counter is peripheral clock 1b: Source clock for timestamp counter is bit time clock
11 to 8	TSP[3:0]	0h	RW	Timestamp Prescaler 0h: Timestamp prescaler = 1 1h: Timestamp prescaler = 2 2h: Timestamp prescaler = 4 3h: Timestamp prescaler = 8 ⋮ Dh: Timestamp prescaler = 8192 Eh: Timestamp prescaler = 16384 Fh: Timestamp prescaler = 32768
7, 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	CMPOC	0h	RW	CAN-FD Message Payload Overflow Configuration 0b: Message is rejected 1b: Message payload is cut to fit to configured message size
4	DCS	0h	RW	Data Link Controller Clock Select 0b: CANFD_0_clkc 1b: Setting prohibited
3	MME	0h	RW	Mirror Mode Enable 0b: Mirror mode disabled 1b: Mirror mode enabled
2	DRE	0h	RW	DLC Replacement Enable 0b: DLC replacement disabled 1b: DLC replacement enabled
1	DCE	0h	RW	DLC Check Enable 0b: DLC check disabled 1b: DLC check enabled

Bit	Bit Name	Initial Value	R/W	Description
0	TPRI	0h	RW	Transmission Priority 0b: ID priority 1b: Message buffer number priority

TPRI bit (Transmission Priority)

The TPRI bit selects the transmission priority for all CAN channels.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

Message buffer number priority should not be used together with TX queue transmission.

DCE bit (DLC Check Enable)

The DCE bit enables data length code (DLC) check for all CAN channels.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

DRE bit (DLC Replacement Enable)

When the DRE bit is 1b and the DCE is 1b, the CAN-FD stores the configured value (CFDGAFLP0n.GAFLDLC) of the DLC in the destination RX message buffer or FIFO buffer if the DLC check passes. Otherwise, the DLC value in the destination RX message buffer or FIFO buffer is unchanged.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

MME bit (Mirror Mode Enable)

The MME bit enables the Mirror mode for all CAN channels.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

DCS bit (Data Link Controller Clock Select)

The DCS bit selects the clock source for CAN communication.

Do not write to this bit in GL_SLEEP or GL_OPERATION mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

CMPOC bit (CAN-FD Message Payload Overflow Configuration)

The CMPOC bit controls the message payload acceptance mechanism when the received payload is higher than the message buffer payload size CFDRMNB.RMPLS, CFDRFCCn.RFPLS, and CFDCFCCn.CFPLS. The received message payload is always compared with the available message payload size in the message buffer.

Do not write to this bit in GL_SLEEP or GL_OPERATION mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

When this bit is set and payload overflow occurs, the DLC value is stored in the RX message buffer or FIFO buffer unchanged.

TSP[3:0] bits (Timestamp Prescaler)

The value configured in the TSP[3:0] bits defines the period of the clock source used for the timestamp counter.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

TSSS bit (Timestamp Source Select)

The TSSS bit allows the selection of the clock source for the timestamp counter.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode. Additionally, do not set this bit to 1b when CAN-FD communication is used.

Note: The bit time clock varies depending on the nominal and data rate bit configuration.

TSBTCS[2:0] bits (Timestamp Bit Time Channel Select)

The TSBTCS[2:0] bits allow the selection of the bit time clock of a particular channel for the timestamp counter.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

ITRCP[15:0] bits (Interval Timer Reference Clock Prescaler)

The ITRCP[15:0] bits allow the definition of a reference clock for the FIFO interval timer source clock.

When these bits are 0000h, the timer is disabled.

Do not write to this bit in GL_SLEEP mode. Only write to this bit when CAN-FD module is in GL_RESET mode.

7.9.2.2.12 Global Control Register (CFDGCTR)

The CFDGCTR register controls the global mode of the CAN-FD module and the timestamp function. The register also enables and disables the global error interrupts.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 0088h
Initial Value : 0000_0005h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TSRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MOWEIE	QMEIE	-	QOWEIE	CMPOFIE	THLEIE	MEIE	DEIE	-	-	-	-	-	GSLPR	GMDC[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	TSRST	0h	RW	Timestamp Reset 0b: Timestamp not reset 1b: Timestamp reset
15	MOWEIE	0h	RW	Message Lost Error Interrupt Enable 0b: GW FIFO message lost error interrupt disabled 1b: GW FIFO message lost error interrupt enabled
14	QMEIE	0h	RW	TXQ Message Lost Error Interrupt Enable 0b: TXQ message lost error interrupt disabled 1b: TXQ message lost error interrupt enabled
13	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12	QOWEIE	0h	RW	TXQ Message Overwrite Error Interrupt Enable 0b: TXQ message overwrite error interrupt disabled 1b: TXQ message overwrite error interrupt enabled
11	CMPOFIE	0h	RW	CAN-FD Message Payload Overflow Flag Interrupt Enable 0b: CAN-FD message payload overflow flag interrupt disabled 1b: CAN-FD message payload overflow flag interrupt enabled
10	THLEIE	0h	RW	TX History List Entry Lost Interrupt Enable 0b: TX history list entry lost interrupt disabled 1b: TX history list entry lost interrupt enabled
9	MEIE	0h	RW	Message Lost Error Interrupt Enable 0b: Message lost error interrupt disabled 1b: Message lost error interrupt enabled
8	DEIE	0h	RW	DLC Check Interrupt Enable 0b: DLC check interrupt disabled 1b: DLC check interrupt enabled
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	GSLPR	1h	RW	Global Sleep Request 0b: Global sleep request disabled 1b: Global sleep request enabled
1, 0	GMDC[1:0]	1h	RW	Global Mode Control 00b: Global operation mode request 01b: Global reset mode request 10b: Global halt mode request 11b: Keep current value

GMDC[1:0] bits (Global Mode Control)

The GMDC[1:0] bits can be used to configure the modes for the CAN-FD module. Additionally, if CFDGCTR.GSLPR bit is 1b when the CAN-FD module is in Reset mode, the CAN-FD module transits to Global Sleep mode. Additionally, if CFDGCTR.GSLPR is 1b when the CAN-FD module is in Reset Mode, then the CAN-FD module transits to Global Sleep Mode.

Setting the GMDC[1:0] bits to 11b has no effect. Mode transition is described in detail in **7.9.3.2 Global Modes**.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

GSLPR bit (Global Sleep Request)

The GSLPR bit globally selects the sleep request for CAN-FD module including all CAN channels. Channel sleep request is set automatically for all channels.

Only write to this bit when the CAN-FD module is in GL_RESET or GL_SLEEP mode.

DEIE bit (DLC Check Interrupt Enable)

When the DEIE bit is 1b, an interrupt is generated if a DLC error is detected in the received frames.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

MEIE bit (Message Lost Error Interrupt Enable)

When the MEIE bit is 1b, an interrupt is generated if a message lost condition occurs.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

THLEIE bit (TX History List Entry Lost Interrupt Enable)

When the THLEIE bit is 1b, an interrupt is generated if a TX history list entry lost condition occurs.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

CMPOFIE bit (CAN-FD Message Payload Overflow Flag Interrupt Enable)

When the CMPOFIE bit is 1b, an interrupt is generated when a CAN-FD message payload overflow condition occurs.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

QOWEIE bit (TXQ Message Overwrite Error Interrupt Enable)

When the QOWEIE bit is 1b, an interrupt is generated when a TXQ message overwrite condition occurs.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

QMEIE bit (TXQ Message Lost Error Interrupt Enable)

When the QMEIE bit is 1b, an interrupt is generated when a TXQ message lost condition occurs.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

MOWEIE bit (Message Lost Error Interrupt Enable)

When the MOWEIE bit is 1b, an interrupt is generated in GW mode and a GW FIFO message overwrite condition occurs.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

TSRST bit (Timestamp Reset)

When the TSRST bit is 1b, the Global Timestamp Register is reset to 0000h.

Do not write to this bit when the CAN-FD module is in GL_SLEEP or GL_RESET mode.

Read value is always 0b.

This bit is cleared automatically by the CAN-FD logic.

7.9.2.2.13 Global Status Register (CFDGSTS)

The CFDGSTS register indicates the global status of the CAN-FD module.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 008Ch
Initial Value : 0000_000Dh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	GRAMINIT	GSLPSTS	GHLTSTS	GRSTSTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read.
3	GRAMINIT	1h	R	Global RAM Initialization 0b: RAM initialization is complete 1b: RAM initialization is ongoing
2	GSLPSTS	1h	R	Global Sleep Status 0b: Not in Sleep mode 1b: In Sleep mode
1	GHLTSTS	0h	R	Global Halt Status 0b: Not in Halt mode 1b: In Halt mode
0	GRSTSTS	1h	R	Global Reset Status 0b: Not in Reset mode 1b: In Reset mode

GRSTSTS bit (Global Reset Status)

The GRSTSTS bit indicates the status of Global CAN-FD module Reset mode.

This bit is set automatically when the CAN-FD module enters GL_RESET mode. When the mode changes from GL_RESET mode to GL_SLEEP mode, this bit remains set.

This bit is cleared automatically when the CAN-FD module exits the GL_RESET mode.

GHLTSTS bit (Global Halt Status)

The GHLTSTS bit indicates the status of Global CAN-FD module Halt mode.

This bit is set automatically when the CAN-FD module enters GL_HALT mode.

This bit is cleared automatically when the CAN-FD module exits the GL_HALT mode.

GSLPSTS bit (Global Sleep Status)

The GSLPSTS bit indicates the status of Global CAN-FD module Sleep mode.

This bit is set automatically when the CAN-FD module enters GL_SLEEP mode.

This bit is cleared automatically when the CAN-FD module exits the GL_SLEEP mode.

GRAMINIT bit (Global RAM Initialization)

The GRAMINIT bit indicates the status of Global CAN-FD module RAM initialization.

This bit is set automatically when the CAN-FD module enters GL_SLEEP mode after a hardware reset.

This bit is cleared automatically when the CAN-FD module completed RAM initialization.

7.9.2.2.14 Global Error Flag Register (CFDGERFL)

The CFDGERFL register indicates the detection of global errors.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 0090h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	EEF5	EEF4	EEF3	EEF2	EEF1	EEF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	MOWES	QMES	-	QOWES	CMPOF	THLES	MES	DEF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	RW	R	RW	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
21 to 16	EEFn	0h	RW	ECC Error Flag for Channel n (n = 0 to 5) 0b: ECC error not detected during TX-SCAN 1b: ECC error detected during TX-SCAN
15 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7	MOWES	0h	R	Message Overwrite Error Status 0b: Message overwrite error not detected 1b: Message overwrite error detected
6	QMES	0h	R	TXQ Message Lost Error Status 0b: TXQ message lost error not detected 1b: TXQ message lost error detected
5	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	QOWES	0h	R	TXQ Message Overwrite Error Status 0b: TXQ message overwrite error not detected 1b: TXQ message overwrite error detected
3	CMPOF	0h	RW	CAN-FD Message Payload Overflow Flag 0b: CAN-FD message payload overflow not detected 1b: CAN-FD message payload overflow detected
2	THLES	0h	R	TX History List Entry Lost Error Status 0b: TX history list entry lost error not detected 1b: TX history list entry lost error detected
1	MES	0h	R	Message Lost Error Status 0b: Message lost error not detected 1b: Message lost error detected
0	DEF	0h	RW	DLC Error Flag 0b: DLC error not detected 1b: DLC error detected

DEF bit (DLC Error Flag)

The DEF bit indicates the error status of the DLC.

Do not write to this bit when the CAN-FD module is in GL_SLEEP or GL_RESET mode. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

This bit is set automatically when a DLC error is detected in a received frame.

The bit is cleared by writing 0b to it.

This bit is cleared automatically in GL_RESET mode.

MES bit (Message Lost Error Status)

The MES bit indicates status of the message lost error.

This bit is set automatically when a FIFO message lost error is detected.

This bit is cleared automatically when:

- All FIFO message lost flags are cleared
- The CAN-FD module is in GL_RESET mode.

THLES bit (TX History List Entry Lost Error Status)

The THLES bit indicates status of the TX history list entry lost error.

This bit is set automatically when a TX history list entry lost error is detected.

This bit is cleared automatically when:

- All TX history list entry lost flags are cleared
- The CAN-FD module is in GL_RESET mode.

CMPOF bit (CAN-FD Message Payload Overflow Flag)

The CMPOF bit is set automatically when a CAN-FD message payload overflow is detected on at least one channel.

Do not write to this bit when the CAN-FD module is in GL_SLEEP or GL_RESET mode.

This bit is cleared by writing 0b to it. Writing 1b to this bit has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

This bit is cleared automatically in GL_RESET mode.

QOWES bit (TXQ Message Overwrite Error Status)

The QOWES bit is set automatically when a TXQ message overwrite error is detected.

This bit is cleared automatically when all TXQ message overwrite flags are cleared.

This bit is cleared automatically in GL_RESET mode.

QMES bit (TXQ Message Lost Error Status)

The QMES bit is set automatically when a TXQ message lost error is detected.

This bit is cleared automatically when all TXQ message lost flags are cleared.

This bit is cleared automatically in GL_RESET mode.

MOWES bit (Message Overwrite Error Status)

The MOWES bit is set automatically when GW mode and COMFIFO message overwrite error is detected.

This bit is cleared automatically when all COMFIFO Message Overwrite flags are cleared.

This bit is cleared automatically in GL_RESET mode.

EEFn bit (ECC Error Flag for Channel n) (n = 0 to 5)

The EEFn bit specifies whether an ECC error has occurred on Channel n.

Do not write to this bit when the CAN-FD module is in GL_SLEEP or GL_RESET mode. Writing 1b to this bit has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

The bit is cleared by writing 0b to it. This bit is cleared automatically in GL_RESET mode.

7.9.2.2.15 Global TX Interrupt Status Register 0 (CFDGTINTSTS0)

The CFDGTINTSTS0 register indicates the detection of transmit specific interrupts.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1300h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	CFOTIF3	TQOFIF3	THIF3	CFTIF3	TQIF3	TAIF3	TSIF3	-	CFOTIF2	TQOFIF2	THIF2	CFTIF2	TQIF2	TAIF2	TSIF2
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	CFOTIF1	TQOFIF1	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1	-	CFOTIF0	TQOFIF0	THIF0	CFTIF0	TQIF0	TAIF0	TSIF0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read.
30	CFOTIF3	0h	R	COM FIFO One Frame Transmission Interrupt Flag Channel 3 0b: Channel 3 COM FIFO One Frame Transmission Interrupt flag not set 1b: Channel 3 COM FIFO One Frame Transmission Interrupt flag set
29	TQOFIF3	0h	R	TX Queue One Frame Transmission Interrupt Flag Channel 3 0b: Channel 3 TX Queue One Frame Transmission Interrupt flag not set 1b: Channel 3 TX Queue One Frame Transmission Interrupt flag set
28	THIF3	0h	R	TX History List Interrupt Channel 3 0b: Channel 3 TX History List Interrupt flag not set 1b: Channel 3 TX History List Interrupt flag set
27	CFTIF3	0h	R	COM FIFO TX/GW Mode Interrupt Flag Channel 3 0b: Channel 3 COM FIFO TX/GW Mode Interrupt flag not set 1b: Channel 3 COM FIFO TX/GW Mode Interrupt flag set
26	TQIF3	0h	R	TX Queue Interrupt Flag Channel 3 0b: Channel 3 TX Queue Interrupt flag not set 1b: Channel 3 TX Queue Interrupt flag set
25	TAIF3	0h	R	TX Abort Interrupt Flag Channel 3 0b: Channel 3 TX Abort Interrupt flag not set 1b: Channel 3 TX Abort Interrupt flag set
24	TSIF3	0h	R	TX Successful Interrupt Flag Channel 3 0b: Channel 3 TX Successful Interrupt flag not set 1b: Channel 3 TX Successful Interrupt flag set
23	-	0h	R	Reserved Whenever it is read, 0b is read.
22	CFOTIF2	0h	R	COM FIFO One Frame Transmission Interrupt Flag Channel 2 0b: Channel 2 COM FIFO One Frame Transmission Interrupt flag not set 1b: Channel 2 COM FIFO One Frame Transmission Interrupt flag set
21	TQOFIF2	0h	R	TX Queue One Frame Transmission Interrupt Flag Channel 2 0b: Channel 2 TX Queue One Frame Transmission Interrupt flag not set 1b: Channel 2 TX Queue One Frame Transmission Interrupt flag set
20	THIF2	0h	R	TX History List Interrupt Channel 2 0b: Channel 2 TX History List Interrupt flag not set 1b: Channel 2 TX History List Interrupt flag set
19	CFTIF2	0h	R	COM FIFO TX/GW Mode Interrupt Flag Channel 2 0b: Channel 2 COM FIFO TX/GW Mode Interrupt flag not set 1b: Channel 2 COM FIFO TX/GW Mode Interrupt flag set
18	TQIF2	0h	R	TX Queue Interrupt Flag Channel 2 0b: Channel 2 TX Queue Interrupt flag not set 1b: Channel 2 TX Queue Interrupt flag set
17	TAIF2	0h	R	TX Abort Interrupt Flag Channel 2 0b: Channel 2 TX Abort Interrupt flag not set 1b: Channel 2 TX Abort Interrupt flag set

Bit	Bit Name	Initial Value	R/W	Description
16	TSIF2	0h	R	TX Successful Interrupt Flag Channel 2 0b: Channel 2 TX Successful Interrupt flag not set 1b: Channel 2 TX Successful Interrupt flag set
15	-	0h	R	Reserved Whenever it is read, 0b is read.
14	CFOTIF1	0h	R	COM FIFO One Frame Transmission Interrupt Flag Channel 1 0b: Channel 1 COM FIFO One Frame Transmission Interrupt flag not set 1b: Channel 1 COM FIFO One Frame Transmission Interrupt flag set
13	TQOFIF1	0h	R	TX Queue One Frame Transmission Interrupt Flag Channel 1 0b: Channel 1 TX Queue One Frame Transmission Interrupt flag not set 1b: Channel 1 TX Queue One Frame Transmission Interrupt flag set
12	THIF1	0h	R	TX History List Interrupt Channel 1 0b: Channel 1 TX History List Interrupt flag not set 1b: Channel 1 TX History List Interrupt flag set
11	CFTIF1	0h	R	COM FIFO TX/GW Mode Interrupt Flag Channel 1 0b: Channel 1 COM FIFO TX/GW Mode Interrupt flag not set 1b: Channel 1 COM FIFO TX/GW Mode Interrupt flag set
10	TQIF1	0h	R	TX Queue Interrupt Flag Channel 1 0b: Channel 1 TX Queue Interrupt flag not set 1b: Channel 1 TX Queue Interrupt flag set
9	TAIF1	0h	R	TX Abort Interrupt Flag Channel 1 0b: Channel 1 TX Abort Interrupt flag not set 1b: Channel 1 TX Abort Interrupt flag set
8	TSIF1	0h	R	TX Successful Interrupt Flag Channel 1 0b: Channel 1 TX Successful Interrupt flag not set 1b: Channel 1 TX Successful Interrupt flag set
7	-	0h	R	Reserved Whenever it is read, 0b is read.
6	CFOTIF0	0h	R	COM FIFO One Frame Transmission Interrupt Flag Channel 0 0b: Channel 0 COM FIFO One Frame Transmission Interrupt flag not set 1b: Channel 0 COM FIFO One Frame Transmission Interrupt flag set
5	TQOFIF0	0h	R	TX Queue One Frame Transmission Interrupt Flag Channel 0 0b: Channel 0 TX Queue One Frame Transmission Interrupt flag not set 1b: Channel 0 TX Queue One Frame Transmission Interrupt flag set
4	THIF0	0h	R	TX History List Interrupt Channel 0 0b: Channel 0 TX History List Interrupt flag not set 1b: Channel 0 TX History List Interrupt flag set
3	CFTIF0	0h	R	COM FIFO TX/GW Mode Interrupt Flag Channel 0 0b: Channel 0 COM FIFO TX/GW Mode Interrupt flag not set 1b: Channel 0 COM FIFO TX/GW Mode Interrupt flag set
2	TQIF0	0h	R	TX Queue Interrupt Flag Channel 0 0b: Channel 0 TX Queue Interrupt flag not set 1b: Channel 0 TX Queue Interrupt flag set
1	TAIF0	0h	R	TX Abort Interrupt Flag Channel 0 0b: Channel 0 TX Abort Interrupt flag not set 1b: Channel 0 TX Abort Interrupt flag set
0	TSIF0	0h	R	TX Successful Interrupt Flag Channel 0 0b: Channel 0 TX Successful Interrupt flag not set 1b: Channel 0 TX Successful Interrupt flag set

TSIFn bit (TX Successful Interrupt Flag Channel n (n = 0 to 3))

The CFDGTINTSTS0.TSIFn bit is set to 1b when the TX Successful Interrupt flag of the related channel is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX MB Result Status bits are cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

TAIFn bit (TX Abort Interrupt Flag Channel n (n = 0 to 3))

The CFDGTINTSTS0.TAIFn bit is set to 1b when the TX Abort Interrupt flag of the related channel is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX MB Result Status bits are cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

TQIFn bit (TX Queue Interrupt Flag Channel n (n = 0 to 3))

The CFDGTINTSTS0.TQIFn bit is set to 1b when the TX Queue Interrupt flag of the related channel is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX Queue Interrupt flag is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

CFTIFn bit (COM FIFO in TX/GW Mode Interrupt Flag Channel n (n = 0 to 3))

The CFDGTINTSTS0.CFTIFn bit is set to 1b when the related COM TX/GW FIFO Mode Interrupt flag (CFDCFSTSn.CFTXIF) is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related COM TX/GW FIFO Mode Interrupt flag (CFDCFSTSn.CFTXIF) is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

THIFn bit (TX History List Interrupt Flag Channel n (n = 0 to 3))

The CFDGTINTSTS0.THIFn bit is set to 1b when the related TX History List Interrupt flag (CFDTHLSTSn.THLIF) is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX History List Interrupt flag (CFDTHLSTSn.THLIF) is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

TQOFIFn bit (TX Queue One Frame Transmission Interrupt Flag Channel n (n = 0 to 3))

The CFDGTINTSTS0.TQOFIFn bit is set to 1b when the TX Queue One Frame Transmission Interrupt flag of the related channel is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX Queue One Frame Transmission Interrupt flag is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

CFOTIFn bit (COM FIFO One Frame Transmission Interrupt Flag Channel n (n = 0 to 3))

The CFDGTINTSTS0.CFOTIFn bit is set to 1b when the related COM FIFO One Frame Transmission Interrupt flag (CFDCFSTSn.CFOFTXIF) is set (when the Interrupt is enabled).

This bit is cleared automatically:

- When the related COM FIFO One Frame Transmission Interrupt flag (CFDCFSTSn.CFOFTXIF) is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

7.9.2.2.16 Global TX Interrupt Status Register 1 (CFDGTINTSTS1)

The CFDGTINTSTS1 register indicates the detection of transmit specific interrupts.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1304h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	CFOTIF5	TQOFIF5	THIF5	CFTIF5	TQIF5	TAIF5	TSIF5	-	CFOTIF4	TQOFIF4	THIF4	CFTIF4	TQIF4	TAIF4	TSIF4
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	-	All 0	R	Reserved Whenever it is read, 0b is read.
14	CFOTIF5	0h	R	COM FIFO One Frame Transmission Interrupt Flag Channel 5 0b: Channel 5 COM FIFO One Frame Transmission Interrupt flag not set 1b: Channel 5 COM FIFO One Frame Transmission Interrupt flag set
13	TQOFIF5	0h	R	TX Queue One Frame Transmission Interrupt Flag Channel 5 0b: Channel 5 TX Queue One Frame Transmission Interrupt flag not set 1b: Channel 5 TX Queue One Frame Transmission Interrupt flag set
12	THIF5	0h	R	TX History List Interrupt Channel 5 0b: Channel 5 TX History List Interrupt flag not set 1b: Channel 5 TX History List Interrupt flag set
11	CFTIF5	0h	R	COM FIFO TX/GW Mode Interrupt Flag Channel 5 0b: Channel 5 COM FIFO TX/GW Mode Interrupt flag not set 1b: Channel 5 COM FIFO TX/GW Mode Interrupt flag set
10	TQIF5	0h	R	TX Queue Interrupt Flag Channel 5 0b: Channel 5 TX Queue Interrupt flag not set 1b: Channel 5 TX Queue Interrupt flag set
9	TAIF5	0h	R	TX Abort Interrupt Flag Channel 5 0b: Channel 5 TX Abort Interrupt flag not set 1b: Channel 5 TX Abort Interrupt flag set
8	TSIF5	0h	R	TX Successful Interrupt Flag Channel 5 0b: Channel 5 TX Successful Interrupt flag not set 1b: Channel 5 TX Successful Interrupt flag set
7	-	0h	RW	Reserved Whenever it is read, 0b is read.
6	CFOTIF4	0h	R	COM FIFO One Frame Transmission Interrupt Flag Channel 4 0b: Channel 4 COM FIFO One Frame Transmission Interrupt flag not set 1b: Channel 4 COM FIFO One Frame Transmission Interrupt flag set
5	TQOFIF4	0h	R	TX Queue One Frame Transmission Interrupt Flag Channel 4 0b: Channel 4 TX Queue One Frame Transmission Interrupt flag not set 1b: Channel 4 TX Queue One Frame Transmission Interrupt flag set
4	THIF4	0h	R	TX History List Interrupt Channel 4 0b: Channel 4 TX History List Interrupt flag not set 1b: Channel 4 TX History List Interrupt flag set
3	CFTIF4	0h	R	COM FIFO TX/GW Mode Interrupt Flag Channel 4 0b: Channel 4 COM FIFO TX/GW Mode Interrupt flag not set 1b: Channel 4 COM FIFO TX/GW Mode Interrupt flag set
2	TQIF4	0h	R	TX Queue Interrupt Flag Channel 4 0b: Channel 4 TX Queue Interrupt flag not set 1b: Channel 4 TX Queue Interrupt flag set
1	TAIF4	0h	R	TX Abort Interrupt Flag Channel 4 0b: Channel 4 TX Abort Interrupt flag not set 1b: Channel 4 TX Abort Interrupt flag set

Bit	Bit Name	Initial Value	R/W	Description
0	TSIF4	0h	R	TX Successful Interrupt Flag Channel 4 0b: Channel 4 TX Successful Interrupt flag not set 1b: Channel 4 TX Successful Interrupt flag set

TSIFn bit (TX Successful Interrupt Flag Channel n (n = 4, 5))

The CFDGTINTSTS0.TSIFn bit is set to 1b when the TX Successful Interrupt flag of the related channel is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX MB Result Status bits are cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

TAIFn bit (TX Abort Interrupt Flag Channel n (n = 4, 5))

The CFDGTINTSTS0.TAIFn bit is set to 1b when the TX Abort Interrupt flag of the related channel is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX MB Result Status bits are cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

TQIFn bit (TX Queue Interrupt Flag Channel n (n = 4, 5))

The CFDGTINTSTS0.TQIFn bit is set to 1b when the TX Queue Interrupt flag of the related channel is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX Queue Interrupt flag is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

CFTIFn bit (COM FIFO in TX/GW Mode Interrupt Flag Channel n (n = 4, 5))

The CFDGTINTSTS0.CFTIFn bit is set to 1b when the related COM TX/GW FIFO Mode Interrupt flag (CFDCFSTSn.CFTXIF) is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related COM TX/GW FIFO Mode Interrupt flag (CFDCFSTSn.CFTXIF) is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

THIFn bit (TX History List Interrupt Flag Channel n (n = 4, 5))

The CFDGTINTSTS0.THIFn bit is set to 1b when the related TX History List Interrupt flag (CFDTHLSTSn.THLIF) is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX History List Interrupt flag (CFDTHLSTSn.THLIF) is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

TQOFIFn bit (TX Queue One Frame Transmission Interrupt Flag Channel n (n = 4, 5))

The CFDGTINTSTS0.TQOFIFn bit is set to 1b when the TX Queue One Frame Transmission Interrupt flag of the related channel is set (when the interrupt is enabled).

This bit is cleared automatically:

- When the related TX Queue One Frame Transmission Interrupt flag is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

CFOTIFn bit (COM FIFO One Frame Transmission Interrupt Flag Channel n (n = 4, 5))

The CFDGTINTSTS0.CFOTIFn bit is set to 1b when the related COM FIFO One Frame Transmission Interrupt flag (CFDCFSTSn.CFOFTXIF) is set (when the Interrupt is enabled).

This bit is cleared automatically:

- When the related COM FIFO One Frame Transmission Interrupt flag (CFDCFSTSn.CFOFTXIF) is cleared (when the interrupt enable is disabled)
- When in GL_RESET or CH_RESET mode.

7.9.2.2.17 Global Timestamp Counter Register (CFDGTSC)

The CFDGTSC register stores the timestamp based on the selected configuration.

Access Size : 16, 32 bits
Address : <CFD_base> + 0094h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read.
15 to 0	TS[15:0]	0h	R	Timestamp value

TS[15:0] bits (Timestamp value)

The Timestamp value is stored in the Global Timestamp Counter register based on the configuration of TSSS, TSBTCS, and TSP. The proper incrementing of the timestamp counter cannot be guaranteed when transitioning to halt state.

Do not write to bits TS[15:0] when the CAN-FD module is in GL_RESET or GL_SLEEP mode.

The TS[15:0] bits are cleared automatically in GL_RESET mode.

7.9.2.2.18 Global Acceptance Filter List Entry Control Register (CFDGAFLECTR)

The CFDGAFLECTR register is used to select the Global Acceptance Filter List page for reading or writing entries into the Global Acceptance Filter List.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 0098h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	AFLDAE	-	-	AFLPN[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8	AFLDAE	0h	RW	Acceptance Filter List Data Access Enable 0b: Acceptance Filter List data access disabled 1b: Acceptance Filter List data access enabled
7 to 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5 to 0	AFLPN[5:0]	0h	RW	Acceptance Filter List Page Number Select an Acceptance Filter List page (Set from 0 to 47.)

AFLPN[5:0] bits (Acceptance Filter List Page Number)

The AFLPN[5:0] bits select the page number to access the desired RAM area of the Acceptance Filter List. One Acceptance Filter List page consists of 16 Acceptance Filter List entries.

Read/write accesses to the Acceptance Filter List can only be performed through a fixed window.

Do not write to these bits when the CAN-FD module is in GL_SLEEP mode. Enter only the values between 0h and 2Fh, inclusive.

AFLDAE bit (Acceptance Filter List Data Access Enable)

The AFLDAE bit prevents write access to the Acceptance Filter List when cleared after configuration of the Acceptance Filter List.

Data can be read from the Acceptance Filter List independent of the status of this bit.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode. Set this bit to enable write access for the Acceptance Filter List.

7.9.2.2.19 Global Acceptance Filter List Configuration Register w (CFDGAFLCFGw) (w = 0, 1, 2)

The CFDGAFLCFGw register is used to define the number of rules for entries in the Acceptance Filter List, applicable for channels 0 to 5.

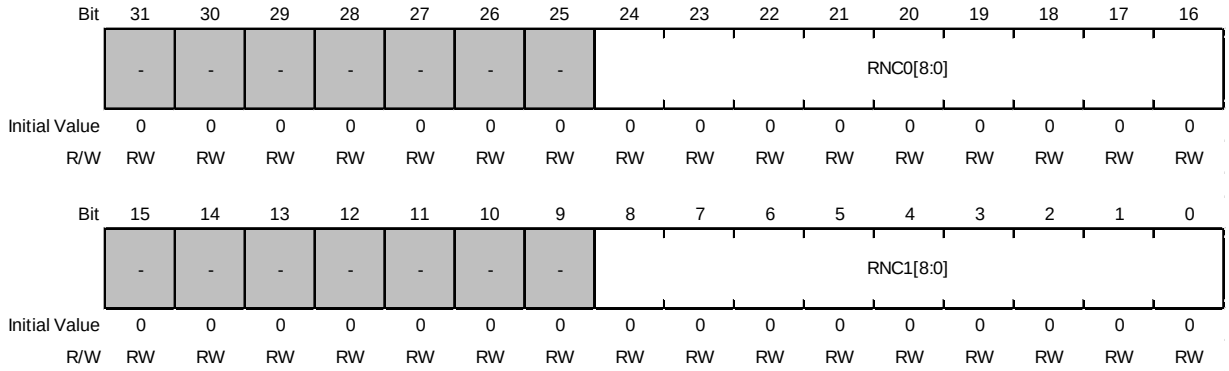
The total number of available entries in the Acceptance Filter List is 256 for 2 CAN channels. However, the filters can be allocated flexibly to the different channels depending on requirements as long as both of the following conditions are satisfied:

- The maximum number of acceptance filter per channel is 256
- The total number of rules defined for all channels is not exceeding the number of available entries in the Acceptance Filter List.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 009Ch + w x 0004h

Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
24 to 16	RNC0[8:0]	0h	RW	Rule Number for Channel 0 Number of rules dedicated to channel 0
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0 is read. The written value should always be 0b.
8 to 0	RNC1[8:0]	0h	RW	Rule Number for Channel 1 Number of rules dedicated to channel 1

RNCn[8:0] bits (Rule Number for Channel n (n = 0 to 5))

The RNCn[8:0] bits define the number of rules in the Acceptance Filter List for channel n.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

7.9.2.2.20 Global Acceptance Filter List ID Register n (CFDGAFLIDn) (n = 1 to 16)

The CFDGAFLIDn register (n = 1 to 16) is used to configure the ID field for the rules of entries in the Global Acceptance Filter List.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1800h + (n - 1) x 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDE	GAFLRTR	GAFLLB	GAFLID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	GAFLIDE	0h	RW	Global Acceptance Filter List Entry IDE Field 0b: Standard identifier of rule entry ID is valid for acceptance filtering 1b: Extended identifier of rule entry ID is valid for acceptance filtering
30	GAFLRTR	0h	RW	Global Acceptance Filter List Entry RTR Field 0b: Data frame 1b: Remote frame
29	GAFLLB	0h	RW	Global Acceptance Filter List Entry Loopback Configuration 0b: Global Acceptance Filter List entry ID for acceptance filtering with attribute RX 1b: Global Acceptance Filter List entry ID for acceptance filtering with attribute TX
28 to 0	GAFLID[28:0]	0h	RW	Global Acceptance Filter List Entry ID Field ID part of the Global Acceptance Filter List entry

GAFLID[28:0] bits (Global Acceptance Filter List Entry ID Field)

The GAFLID[28:0] bits represent the CAN identifier (ID) field of each entry in the Global Acceptance Filter List.

The acceptance filter process compares this field against the ID of a received CAN message. For alignment of these bits in standard and extended frame formats, see **7.9.2.2.89 Identifier Bits Alignment**.

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLLB bit (Global Acceptance Filter List Entry Loopback Configuration)

The GAFLLB bit selects whether entry in the Global Acceptance Filter List gets the attribute RX or TX.

This attribute determines the validity of the entry in Mirror mode, Loopback test mode, and during standard (non-loopback) reception. See **7.9.5.5 Loopback Modes** for detailed description of the validity of the Global Acceptance Filter List entry depending on transmitter/receiver case, the type of loopback mode, and RX/TX attribute.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLRTR bit (Global Acceptance Filter List Entry RTR Field)

The GAFLRTR bit allows the configuration of the specified frame format (data frame or remote frame) for each entry of the Global Acceptance Filter List. For each rule entry in a CAN channel, the acceptance filter process compares this bit against the RTR bit of the received CAN message.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLIDE bit (Global Acceptance Filter List Entry IDE Field)

The GAFLIDE bit allows the configuration of the ID format (standard ID or extended ID) for each entry in the Global Acceptance Filter List. For each rule entry in a CAN channel, the acceptance filter process compares this bit against the IDE bit of the received CAN message.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

7.9.2.2.21 Global Acceptance Filter List Mask Register n (CFDGAFLMn) (n = 1 to 16)

The CFDGAFLMn register (n = 1 to 16) is used to configure the Mask field of each rule for entries in the Global Acceptance Filter List.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1804h + (n - 1) x 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDEM	GAFLRTRM	GAFLIFL1	GAFLIDM[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLIDM[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	GAFLIDEM	0h	RW	Global Acceptance Filter List IDE Mask 0b: IDE bit is not used for ID matching 1b: IDE bit is used for ID matching
30	GAFLRTRM	0h	RW	Global Acceptance Filter List Entry RTR Mask 0b: RTR bit is not used for ID matching 1b: RTR bit is used for ID matching
29	GAFLIFL1	0h	RW	Global Acceptance Filter List Information Label 1 Global Acceptance Filter List information label bit 1
28 to 0	GAFLIDM[28:0]	0h	RW	Global Acceptance Filter List ID Mask Field Global Acceptance Filter List Mask field bits for ID field

GAFLIDM[28:0] bits (Global Acceptance Filter List ID Mask Field)

GAFLIDM[28:0] bits are the filter mask bits for the related bits in the CAN Identifier field of each Global Acceptance Filter List entry.

0: Corresponding STD-ID/EXT-ID bit is not used for ID matching
1: Corresponding STD-ID/EXT-ID bit is used for ID matching

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0b.
Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLIFL1 bit (Global Acceptance Filter List Information Label 1)

The GAFLIFL1 bit allows the configuration of a 2-bit information label to be attached to a received message accepted by the associated entry in the Global Acceptance Filter List. This bit is a MSB bit of an information label.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

This bit is stored in the Information Label Field [1] (CFDRMFDSTS.RMIFL[1], CFDRFFDSTS.RFIFL[1], CFDCFFDCSTSn.CFIFL[1]) of the storage location of an incoming message.

Note: This bit is stored in CFDTHLACC1n.TIFL[1] when CFDTHLCCn.THLDGE = 1b is set up using Gateway function.

GAFLRTRM bit (Global Acceptance Filter List Entry RTR Mask)

The GAFLRTRM bit allows the configuration of the RTR mask bit for each entry in the Global Acceptance Filter List.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLIDEM bit (Global Acceptance Filter List IDE Mask)

The GAFLIDEM bit allows the configuration of the IDE mask bit for each entry in the Global Acceptance Filter List.

When the IDE mask bit is 0b, the ID comparison depends on the received IDE bit.

If the received IDE bit is 0b, the STD-ID comparison takes place.

If the received IDE bit is 1b, the EXT-ID comparison takes place.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to this bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

7.9.2.2.22 Global Acceptance Filter List Pointer 0 Register n (CFDGAFLP0n) (n = 1 to 16)

The CFDGAFLP0n register (n = 1 to 16) is used to configure the data length code (DLC), software pointer, single message buffer select, and message buffer direction pointer for each rule entry in the Global Acceptance Filter List.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1808h + (n - 1) x 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLPTR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLRMV	-	-	GAFLRMDP[4:0]					GAFLIFL0	GAFLSRD2	GAFLSRD1	GAFLSRD0	GAFLDLC[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	GAFLPTR [15:0]	0h	RW	Global Acceptance Filter List Pointer
15	GAFLRMV	0h	RW	Global Acceptance Filter List RX Message Buffer Valid 0b: Single message buffer direction pointer is invalid 1b: Single message buffer direction pointer is valid
14, 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12 to 8	GAFLRMDP [4:0]	0h	RW	Global Acceptance Filter List RX Message Buffer Direction Pointer RX message buffer number for storage of received messages
7	GAFLIFL0	0h	RW	Global Acceptance Filter List Information Label 0
6	GAFLSRD2	0h	RW	Global Acceptance Filter List Select Routing Destination 2 0b: Routing target is CFIFO2 1b: Routing target is TX Queue 2 instead of CFIFO2
5	GAFLSRD1	0h	RW	Global Acceptance Filter List Select Routing Destination 1 0b: Routing target is CFIFO1 1b: Routing target is TX Queue 1 instead of CFIFO1
4	GAFLSRD0	0h	RW	Global Acceptance Filter List Select Routing Destination 0 0b: Routing target is CFIFO0 1b: Routing target is TX Queue 0 instead of CFIFO0
3 to 0	GAFLDLC[3:0]	0h	RW	Global Acceptance Filter List DLC Field Minimum number of data bytes in a data frame required for acceptance

GAFLDLC[3:0] bits (Global Acceptance Filter List DLC Field)

The GAFLDLC[3:0] bits allow the configuration of a minimum data length code (DLC) value for a message to be accepted by the associated entry in the Global Acceptance Filter List (automatic DLC filter function).

DLC filter process is only passed if the DLC value of the message accepted by an entry in the Global Acceptance Filter List is equal to or higher than the DLC value configured for this associated Global Acceptance Filter List entry. Automatic DLC filter function is disabled for the corresponding rule entry when this field is set to 0b.

Table 7.9-6 shows DLC value that can be configured.

Table 7.9-6 Configuration of DLC Value

Format	DLC[3:0]	Description
CAN and CAN-FD	0000b	DLC of received message = 0 or more (DLC filter check is disabled)
CAN and CAN-FD	0001b	DLC of received message = 1 or more
CAN and CAN-FD	0010b	DLC of received message = 2 or more
CAN and CAN-FD	0011b	DLC of received message = 3 or more
CAN and CAN-FD	0100b	DLC of received message = 4 or more
CAN and CAN-FD	0101b	DLC of received message = 5 or more
CAN and CAN-FD	0110b	DLC of received message = 6 or more
CAN and CAN-FD	0111b	DLC of received message = 7 or more
CAN	1xxxb	DLC of received message = 8 or more
CAN-FD	1000b	DLC of received message = 8 or more*1
CAN-FD	1001b	DLC of received message = 12 or more*1
CAN-FD	1010b	DLC of received message = 16 or more*1
CAN-FD	1011b	DLC of received message = 20 or more*1
CAN-FD	1100b	DLC of received message = 24 or more*1
CAN-FD	1101b	DLC of received message = 32 or more*1
CAN-FD	1110b	DLC of received message = 48 or more*1
CAN-FD	1111b	DLC of received message = 64*1

Note 1. This setting is not available in the classical CAN function.

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLSRD0 bit (Global Acceptance Filter List Select Routing Destination 0)

The GAFLSRD0 bit changes a copy destination to CFIFO0 or TXQ0 by routing.

If this bit is set as 1b, the preset value of CFDGAFLP1n.GAFLFDP selects TXQ0.

If this bit is set to 0b, the preset value of CFDGAFLP1n.GAFLFDP selects CFIFO0.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to the bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLSRD1 bit (Global Acceptance Filter List Select Routing Destination 1)

The GAFLSRD1 bit changes a copy destination to CFIFO1 or TXQ1 by routing.

If this bit is set to 1b, the preset value of CFDGAFLP1n.GAFLFDP selects TXQ1.

If this bit is set to 0b, the preset value of CFDGAFLP1n.GAFLFDP selects CFIFO1.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to the bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLSRD2 bit (Global Acceptance Filter List Select Routing Destination 2)

The GAFLSRD2 bit changes a copy destination to CFIFO2 or TXQ2 by routing.

If this bit is set to 1b, the preset value of CFDGAFLP1n.GAFLFDP selects TXQ2.

If this bit is set to 0b, the preset value of CFDGAFLP1n.GAFLFDP selects CFIFO2.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to the bit when the CAN-FD module is in CH_RESET or CH_HALT mode.

GAFLIFL0 bit (Global Acceptance Filter List Information Label 0)

The GAFLIFL0 bit allows the configuration of a 2-bit information label that can be attached to a received message accepted by the related Global Acceptance Filter List entry. This bit is a LSB bit of an information label.

Do not write to this bit when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to the bit when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

This bit is stored in Information Label Field[0] (CFDRMFDSTS.RMIFL[0], CFDRFFDSTS.RFIFL[0], CFDCFFDCSTSn.CFIFL[0]) of the storage location of an incoming message.

This bit is stored in CFDTHLACC1n.TIFL[0] when CFDTHLCCn.THLDGE = 1b is set up using the gateway function.

GAFLRMDP[4:0] bits (Global Acceptance Filter List RX Message Buffer Direction Pointer)

The GAFLRMDP[4:0] bits allow the configuration of a single reception message buffer as the destination target for a received message that passes the acceptance check of the related Global Acceptance Filter List entry. The value entered is the single destination message buffer number.

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

CFDRMNB.NRXMB[7:0] is the value entered in the RX Message Buffer Number Register to configure the number of RX message buffers. The value to be entered in CFDGAFLP0n.GAFLRMDP[4:0] bits should only be between 00h and CFDRMNB.NRXMB[7:0] to 1 less.

If CFDRMNB.NRXMB[7:0] = 00h, the GAFLRMV bit should be configured as 0b.

GAFLRMV bit (Global Acceptance Filter List RX Message Buffer Valid)

The GAFLRMV bit allows the enabling or disabling of a single reception message buffer as the target for a received message that passes the acceptance check of the related Global Acceptance Filter List entry.

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

GAFLPTR[15:0] bits (Global Acceptance Filter List Pointer)

The GAFLPTR[15:0] bits allow the configuration of a 16-bit pointer to be attached to a received message accepted by the related Global Acceptance Filter List entry. The pointer is added during message storage in the Message Buffer area and can be used by the application as a support function. The pointer information can be used for example, to support PDU Identifier allocation for the received message in AUTOSAR systems.

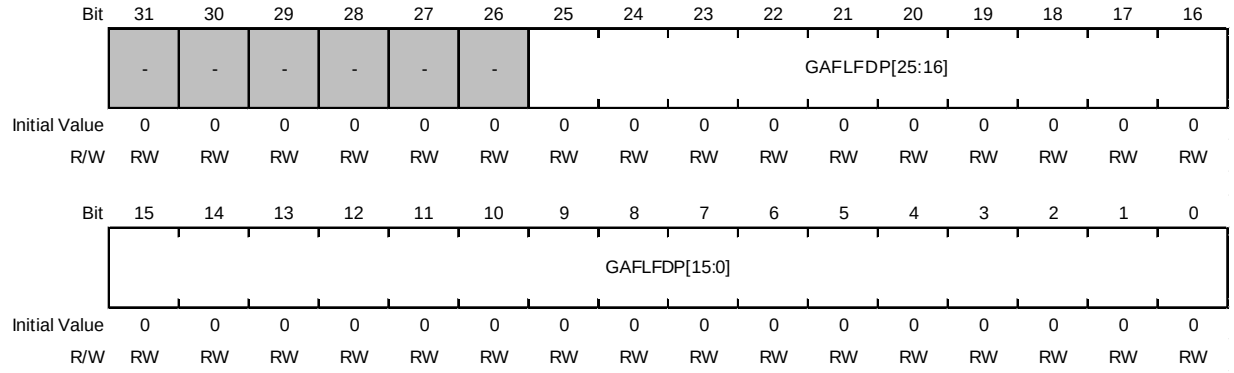
Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0b.

Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

7.9.2.2.23 Global Acceptance Filter List Pointer 1 Register n (CFDGAFLP1n) (n = 1 to 16)

The CFDGAFLP1n register (n = 1 to 16) is used to configure the FIFO direction pointer fields in each Rule Entry of the Global Acceptance Filter List.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 180Ch + (n - 1) x 0010h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
25 to 0	GAFLFDP [25:0]	0h	RW	Global Acceptance Filter List FIFO Direction Pointer FIFO direction pointer bits for received message storage

GAFLFDP[25:0] bits (Global Acceptance Filter List FIFO Direction Pointer)

The GAFLFDP[25:0] bits allow the configuration of FIFO buffers as the target for a received message that passes the acceptance check of the related Global Acceptance Filter List entry. Each bit of the CFDGAFLP1n.GAFLFDP[25:0] is configured as dedicated FIFO.

Bit	Value (binary)	Function
0	0	Disable RX FIFO 0 as target for reception
	1	Enable RX FIFO 0 as target for reception
1	0	Disable RX FIFO 1 as target for reception
	1	Enable RX FIFO 1 as target for reception
2	0	Disable RX FIFO 2 as target for reception
	1	Enable RX FIFO 2 as target for reception
3	0	Disable RX FIFO 3 as target for reception
	1	Enable RX FIFO 3 as target for reception
4	0	Disable RX FIFO 4 as target for reception
	1	Enable RX FIFO 4 as target for reception
5	0	Disable RX FIFO 5 as target for reception
	1	Enable RX FIFO 5 as target for reception
6	0	Disable RX FIFO 6 as target for reception
	1	Enable RX FIFO 6 as target for reception
7	0	Disable RX FIFO 7 as target for reception
	1	Enable RX FIFO 7 as target for reception
8	0	Disable Common FIFO 0 and Channel 0 TX Queue 0 as target for reception
	1	GAFLSRD0 = 0b: Enable Common FIFO 0 as target for reception GAFLSRD0 = 1b: Enable Channel 0 TX Queue 0 as target for reception

Bit	Value (binary)	Function
9	0	Disable Common FIFO 1 and Channel 0 TX Queue 1 as target for reception
	1	GAFLSRD1 = 0b: Enable Common FIFO 1 as target for reception GAFLSRD1 = 1b: Enable Channel 0 TX Queue 1 as target for reception
10	0	Disable Common FIFO 2 and Channel 0 TX Queue 2 as target for reception
	1	GAFLSRD2 = 0b: Enable Common FIFO 2 as target for reception GAFLSRD2 = 1b: Enable Channel 0 TX Queue 2 as target for reception
11	0	Disable Common FIFO 3 and Channel 1 TX Queue 0 as target for reception
	1	GAFLSRD0 = 0b: Enable Common FIFO 3 as target for reception GAFLSRD0 = 1b: Enable Channel 1 TX Queue 0 as target for reception
12	0	Disable Common FIFO 4 and Channel 1 TX Queue 1 as target for reception
	1	GAFLSRD1 = 0b: Enable Common FIFO 4 as target for reception GAFLSRD1 = 1b: Enable Channel 1 TX Queue 1 as target for reception
13	0	Disable Common FIFO 5 and Channel 1 TX Queue 2 as target for reception
	1	GAFLSRD2 = 0b: Enable Common FIFO 5 as target for reception GAFLSRD2 = 1b: Enable Channel 1 TX Queue 2 as target for reception
14	0	Disable Common FIFO 6 and Channel 2 TX Queue 0 as target for reception
	1	GAFLSRD0 = 0b: Enable Common FIFO 6 as target for reception GAFLSRD0 = 1b: Enable Channel 2 TX Queue 0 as target for reception
15	0	Disable Common FIFO 7 and Channel 2 TX Queue 1 as target for reception
	1	GAFLSRD1 = 0b: Enable Common FIFO 7 as target for reception GAFLSRD1 = 1b: Enable Channel 2 TX Queue 1 as target for reception
16	0	Disable Common FIFO 8 and Channel 2 TX Queue 2 as target for reception
	1	GAFLSRD2 = 0b: Enable Common FIFO 8 as target for reception GAFLSRD2 = 1b: Enable Channel 2 TX Queue 2 as target for reception
17	0	Disable Common FIFO 9 and Channel 3 TX Queue 0 as target for reception
	1	GAFLSRD0 = 0b: Enable Common FIFO 9 as target for reception GAFLSRD0 = 1b: Enable Channel 3 TX Queue 0 as target for reception
18	0	Disable Common FIFO 10 and Channel 3 TX Queue 1 as target for reception
	1	GAFLSRD1 = 0b: Enable Common FIFO 10 as target for reception GAFLSRD1 = 1b: Enable Channel 3 TX Queue 1 as target for reception
19	0	Disable Common FIFO 11 and Channel 3 TX Queue 2 as target for reception
	1	GAFLSRD2 = 0b: Enable Common FIFO 11 as target for reception GAFLSRD2 = 1b: Enable Channel 3 TX Queue 2 as target for reception
20	0	Disable Common FIFO 12 and Channel 4 TX Queue 0 as target for reception
	1	GAFLSRD0 = 0b: Enable Common FIFO 12 as target for reception GAFLSRD0 = 1b: Enable Channel 4 TX Queue 0 as target for reception
21	0	Disable Common FIFO 13 and Channel 4 TX Queue 1 as target for reception
	1	GAFLSRD1 = 0b: Enable Common FIFO 13 as target for reception GAFLSRD1 = 1b: Enable Channel 4 TX Queue 1 as target for reception
22	0	Disable Common FIFO 14 and Channel 4 TX Queue 2 as target for reception
	1	GAFLSRD2 = 0b: Enable Common FIFO 14 as target for reception GAFLSRD2 = 1b: Enable Channel 4 TX Queue 2 as target for reception
23	0	Disable Common FIFO 15 and Channel 5 TX Queue 0 as target for reception
	1	GAFLSRD0 = 0b: Enable Common FIFO 15 as target for reception GAFLSRD0 = 1b: Enable Channel 5 TX Queue 0 as target for reception
24	0	Disable Common FIFO 16 and Channel 5 TX Queue 1 as target for reception
	1	GAFLSRD1 = 0b: Enable Common FIFO 16 as target for reception GAFLSRD1 = 1b: Enable Channel 5 TX Queue 1 as target for reception

Bit	Value (binary)	Function
25	0	Disable Common FIFO 17 and Channel 5 TX Queue 2 as target for reception
	1	GAFLSRD2 = 0b: Enable Common FIFO 17 as target for reception GAFLSRD2 = 1b: Enable Channel 5 TX Queue 2 as target for reception

Do not write to these bits when CFDGAFLECTR.AFLDAE bit is 0b. Only write to these bits when the related CAN-FD channel is in CH_RESET or CH_HALT mode.

For storage in TX queue, TX queue buffers of a target that is in GW mode is possible.

Only one of the following configurations is valid:

- Up to 8 destination FIFO buffers
- 7 destination FIFO buffers plus one RX message buffer
- 8 destination TX queue buffers
- 7 destination TX queue buffers plus one RX message buffer
- A maximum of 8 destinations in all at FIFO buffer and TX queue buffer.

7.9.2.2.24 RX Message Buffer Number Register (CFDRMNB)

The CFDRMNB register is used to configure the total number of RX message buffers allocated to all channels.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 00ACh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	RMPLS[2:0]			NRXMB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10 to 8	RMPLS*[1][2:0]	0h	RW	Reception Message Buffer Payload Data Size 000b: 8 bytes 001b: 12 bytes 010b: 16 bytes 011b: 20 bytes 100b: 24 bytes 101b: 32 bytes 110b: 48 bytes 111b: 64 bytes
7 to 0	NRXMB[7:0]	0h	RW	Number of RX Message Buffers

Note 1. These bits are not available in the classical CAN function.

NRXMB[7:0] bits (Number of RX Message Buffers)

The NRXMB[7:0] bits are used to configure the number of RX message buffers.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

Enter only values between 0 and 32 inclusive, with 00h indicating that no RX message buffer is allocated.

RMPLS[2:0] bits (Reception Message Buffer Payload Data Size)

The RMPLS[2:0] bits are used to configure the message buffer payload data size.

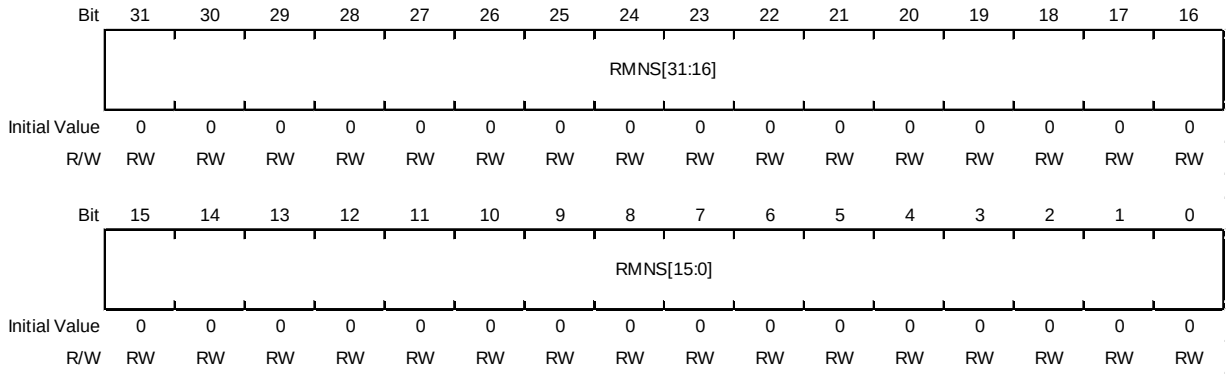
Only write to these bits when the CAN-FD module is in GL_RESET mode.

Note: These bits are not available in the classical CAN function.

7.9.2.2.25 RX Message Buffer New Data Register t (CFDRMNDt) (t = 0 to 2)

The CFDRMNDt specifies the new data storage status of the RX message buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 00B0h + t x 0004h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RMNS[31:0]	0h	RW	RX Message Buffer New Data Status 0b: New data not stored in corresponding RX message buffer 1b: New data stored in corresponding RX message buffer

RMNS[31:0] bits (RX Message Buffer New Data Status)

The RMNSu[31:0] bits indicate the status of new data for the corresponding RX message buffer. RMNS bit [0] corresponds to RX message buffer [0] and so on.

no_of_channels = 6
no_of_CFDRMBCPs_per_channel = (number of RX message buffer components per channel) = 16
no_of_CFDRMBCPs = no_of_channels × no_of_CFDRMBCPs_per_channel (6 × 16 = 96)
no_of_bits_per_register = 32
no_of_CFDRMNDs (number of CFDRMND Registers) = no_of_CFDRMBCPs/no_of_bits_per_register (96/32 = 3)
t = [0 ... no_of_CFDRMNDs − 1]
u = [t × 32 ... (no_of_CFDRMBCPs − ((no_of_CFDRMNDs − 1 − t) × 32) − 1)]

t can be calculated from the target of the New Data Status flag (u) using the formula t = floor (u/32)

Bit position can be calculated using the formula (u − (t × 32))

Do not write to these bits when the CAN-FD module is in GL_RESET or GL_SLEEP mode. Writing 1b has no effect.

These bits cannot be cleared when message storage in the corresponding RX message buffer is in progress.

Do not use the bit clear instruction to clear these bits. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

These bits are set automatically when storage of new messages are in the corresponding RX message buffer. These bits are cleared by writing 0b. These bits are cleared automatically when the CAN-FD module is in GL_RESET mode.

When CFDRMNB.RMPLS = 000b (maximum 8 bytes payload), the duration of message storage is 6 CANFD_0_pclk cycles.

When CFDRMNB.RMPLS > 000b, the duration of message storage is 6 CANFD_0_pclk cycles + 1 for each 4 bytes (maximum of 20 CANFD_0_pclk cycles for 64 bytes).

Note: This feature is not available in the classical CAN function.

7.9.2.2.26 RX FIFO Configuration/Control Register n (CFDRFCCn) (n = 0 to 7)

The CFDRFCCn registers (n = 0 to 7) are used to configure and control the eight RX FIFOs.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 00C0h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RFFIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFIGCV[2:0]			RFIM	-	RFDC[2:0]			-	RFPLS[2:0]			-	-	RFIE	RFE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	RFFIE	0h	RW	RX FIFO Full Interrupt Enable 0b: FIFO interrupt generation disabled 1b: FIFO interrupt generation enabled
15 to 13	RFIGCV[2:0]	0h	RW	RX FIFO Interrupt Generation Counter Value 000b: Interrupt generated when FIFO is 1/8th full 001b: Interrupt generated when FIFO is 1/4th full 010b: Interrupt generated when FIFO is 3/8th full 011b: Interrupt generated when FIFO is 1/2 full 100b: Interrupt generated when FIFO is 5/8th full 101b: Interrupt generated when FIFO is 3/4th full 110b: Interrupt generated when FIFO is 7/8th full 111b: Interrupt generated when FIFO is full
12	RFIM	0h	RW	RX FIFO Interrupt Mode 0b: Interrupt generated when RX FIFO counter reaches RFIGCV value from values smaller than RFIGCV 1b: Interrupt generated at the end of every received message storage
11	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
10 to 8	RFDC[2:0]	0h	RW	RX FIFO Depth Configuration 000b: FIFO Depth = 0 messages 001b: FIFO Depth = 4 messages 010b: FIFO Depth = 8 messages 011b: FIFO Depth = 16 messages 100b: FIFO Depth = 32 messages 101b: FIFO Depth = 48 messages 110b: FIFO Depth = 64 messages 111b: FIFO Depth = 128 messages
7	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6 to 4	RFPLS*1[2:0]	0h	RW	Rx FIFO Payload Data Size Configuration 000b: 8 bytes 001b: 12 bytes 010b: 16 bytes 011b: 20 bytes 100b: 24 bytes 101b: 32 bytes 110b: 48 bytes 111b: 64 bytes
3, 2	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
1	RFIE	0h	RW	RX FIFO Interrupt Enable 0b: FIFO interrupt generation disabled 1b: FIFO interrupt generation enabled
0	RFE	0h	RW	RX FIFO Enable 0b: FIFO disabled 1b: FIFO enabled

Note 1. These bits are not available in the classical CAN function.

RFE bit (RX FIFO Enable)

The RFE bit enables the FIFO. When this bit is set to 0b, the RX FIFO is cleared to empty.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode.

This bit can only be set if the configured FIFO depth is greater than 000b (CFDRFCCn.RFDC[2:0] > 000b).

Set the RFE bit with a separate write access to the CFDRFCCn register, after all the other bits in the CFDRFCCn register are set.

This bit is cleared automatically when the CAN-FD module is in GL_RESET mode.

RFIE bit (RX FIFO Interrupt Enable)

The RFIE bit enables generation of the FIFO interrupt.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

RFPLS[2:0] bits (Rx FIFO Payload Data Size Configuration)

The RFPLS[2:0] bits define the message data payload allocation in the RAM.

This is the maximum number of bytes which can be received by this FIFO.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

Note: These bits are not available in the classical CAN function.

RFDC[2:0] bits (RX FIFO Depth Configuration)

The RFDC[2:0] bits select the depth of the FIFO in terms of the number of messages. If the FIFO depth is configured to 0 messages, the FIFO cannot be used.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

RFIM bit (RX FIFO Interrupt Mode)

The RFIM bit selects the interrupt generation condition for the FIFO.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

RFIGCV[2:0] bits (RX FIFO Interrupt Generation Counter Value)

The RFIGCV[2:0] bits select the counter value of the FIFO for generation of FIFO interrupts. These values represent fractions of the FIFO depth for which an interrupt is generated.

Do not write to these bits when the CAN-FD module is in GL_SLEEP mode.

The setting of the RFIGCV[2:0] bits should be synchronized with the RFDC[2:0] bits.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

RFFIE bit (RX FIFO Full Interrupt Enable)

The RFFIE bit enables generation of the RX FIFO full interrupt. Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

The following content shows examples of interruptions:

- Interruption output in number of arbitrary stages (CFDRFCCn.RFIGCV)
- Interruption output in FIFO full state.

Note: Management of the receiving data of FIFO can be performed by these notices of interruption.

7.9.2.2.27 RX FIFO Status Register n (CFDRFSTSn) (n = 0 to 7)

The CFDRFSTSn register (n = 0 to 7) shows the status of messages stored in the corresponding FIFO buffers.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 00E0h + n x 0004h

Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RFFIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMC[7:0]								-	-	-	-	RFIF	RFMLT	RFFLL	RFEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	RFFIF	0h	RW	RX FIFO Full Interrupt Flag 0b: FIFO full interrupt condition not satisfied 1b: FIFO full interrupt condition satisfied
15 to 8	RFMC[7:0]	0h	R	RX FIFO Message Count Number of messages stored in FIFO
7 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3	RFIF	0h	RW	RX FIFO Interrupt Flag 0b: FIFO interrupt condition not satisfied 1b: FIFO interrupt condition satisfied
2	RFMLT	0h	RW	RX FIFO Message Lost 0b: No message lost in FIFO 1b: FIFO message lost
1	RFFLL	0h	R	RX FIFO Full 0b: FIFO not full 1b: FIFO full
0	RFEMP	1h	R	RX FIFO Empty 0b: FIFO not empty 1b: FIFO empty

RFEMP bit (RX FIFO Empty)

The RFEMP bit is set automatically when:

- The RFMC bit is 0b
- RX FIFO is disabled by setting the CFDRFCCn.RFE bit to 0b
- The CAN-FD module is in GL_RESET mode.

The RFEMP bit is cleared automatically when the first message is stored in the RX FIFO buffer.

RFFLL bit (RX FIFO Full)

The RFFLL bit is set automatically when the number of CAN messages stored in the FIFO buffer matches the configured FIFO depth.

The RFFLL is cleared automatically when:

- The number of CAN messages stored in the FIFO buffer is less than the configured FIFO depth

- RX FIFO is disabled by setting the CFDRFCCn.RFE bit to 0b
- The CAN-FD module is in GL_RESET mode.

RFMLT bit (RX FIFO Message Lost)

Only write to the RFMLT bit when CAN-FD module is in GL_HALT or GL_OPERATION mode. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

This bit is set automatically whenever a message is lost due to attempted storage when the FIFO buffer is already full. If a set from the CAN channel occurs simultaneously with a clear by a write access, then the bit is set.

The bit is cleared:

- By writing 0b to it
- When the CAN-FD module is in GL_RESET mode.

RFIF bit (RX FIFO Interrupt Flag)

The RFIF bit is set automatically when the configured interrupt condition is satisfied. This bit is not automatically cleared when the RX FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

If a set from the CAN channel occurs simultaneously with a clear by a write access, then this bit is set.

The bit is cleared by writing 0b to it. The bit is also cleared when CAN-FD module is in GL_RESET mode.

RFMC[7:0] bits (RX FIFO Message Count)

The RFMC[7:0] bits indicate the number of CAN messages stored in the RX FIFO buffer that can be read by the CPU.

These bits are cleared automatically when the FIFO is disabled and when the CAN-FD module is in GL_RESET mode.

RFFIF bit (RX FIFO Full Interrupt Flag)

The RFFIF bit is not cleared automatically when the RX FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

This bit is set automatically when the FIFO full interrupt condition is satisfied. If a set from the CAN channel occurs simultaneously with a clear by a write access, then the bit is set.

The bit is cleared by writing 0b to it.

The bit is also cleared when the CAN-FD module is in GL_RESET mode.

7.9.2.2.28 RX FIFO Pointer Control Register n (CFDRFPCTRn) (n = 0 to 7)

The CFDRFPCTRn register (n = 0 to 7) can be used to increment the read pointer of the corresponding RX FIFO buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 0100h + n x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RFPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 0	RFPC*[1][7:0]	0h	RW	RX FIFO Pointer Control Increments read pointer of the corresponding RX FIFO buffers

Note 1. These bits are not available in the classical CAN function.

RFPC[7:0] bits (RX FIFO Pointer Control)

When the value FFh is written to the RFPC bits, the pointer of the corresponding RX FIFO buffer is moved to the next FIFO entry. Only write FFh to these registers when the corresponding RX FIFO buffer is enabled and not empty.

The read value from these bits is always 00h.

Only write to these bits when the CAN-FD module is in GL_HALT or GL_OPERATION mode.

Do not write to the RX FIFO Pointer Control registers when DMA is enabled.

7.9.2.2.29 Common FIFO Configuration/Control Register n (CFDCFCCn) (n = 0 to 17)

The CFDCFCCn register (n = 0 to 17) is used to configure the Common FIFOs.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 0120h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFITT[7:0]								CFDC[2:0]			CFTML[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFIGCV[2:0]			CFIM	CFITR	CFITSS	CFM[1:0]		-	CFPLS[2:0]			-	CFTXIE	CFRXIE	CFE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CFITT[7:0]	0h	RW	Common FIFO Interval Transmission Time Delay the start of transmission from the FIFO if configured in TX or GW mode, delay is a multiple of basic Interval Timer Clock Source unit
23 to 21	CFDC[2:0]	0h	RW	Common FIFO Depth Configuration 000b: FIFO Depth = 0 messages 001b: FIFO Depth = 4 messages 010b: FIFO Depth = 8 messages 011b: FIFO Depth = 16 messages 100b: FIFO Depth = 32 messages 101b: FIFO Depth = 48 messages 110b: FIFO Depth = 64 messages 111b: FIFO Depth = 128 messages
20 to 16	CFTML[4:0]	0h	RW	Common FIFO TX Message Buffer Link Transmission scan link position of the corresponding channel
15 to 13	CFIGCV[2:0]	0h	RW	Common FIFO Interrupt Generation Counter Value 000b: Interrupt generated when FIFO is 1/8th full 001b: Interrupt generated when FIFO is 1/4th full 010b: Interrupt generated when FIFO is 3/8th full 011b: Interrupt generated when FIFO is 1/2 full 100b: Interrupt generated when FIFO is 5/8th full 101b: Interrupt generated when FIFO is 3/4th full 110b: Interrupt generated when FIFO is 7/8th full 111b: Interrupt generated when FIFO is full
12	CFIM	0h	RW	Common FIFO Interrupt Mode 0b: RX FIFO mode: RX interrupt generated when Common FIFO counter reaches CFIGCV value from a lower value TX FIFO mode: TX interrupt generated when Common FIFO transmits the last message successfully GW FIFO mode: For RX interrupt flag: Interrupt generated when FIFO counter increments and reaches the value configured in CFIGCV For TX interrupt flag: Interrupt generated when FIFO transmits the last message successfully 1b: RX FIFO mode: RX interrupt generated at the end of every received message storage TX FIFO mode: TX interrupt generated for every successfully transmitted message GW FIFO mode: For RX interrupt flag: Interrupt generated when a message is stored in the FIFO For TX interrupt flag: Interrupt generated when a message is successfully transmitted from the FIFO
11	CFITR	0h	RW	Common FIFO Interval Timer Resolution 0b: Reference clock period × 1 1b: Reference clock period × 10

Bit	Bit Name	Initial Value	R/W	Description
10	CFITSS	0h	RW	Common FIFO Interval Timer Source Select 0b: Reference clock (× 1 / × 10 period) 1b: Bit time clock of related channel (FIFO is linked to fixed channel)
9, 8	CFM[1:0]	0h	RW	Common FIFO Mode 00b: RX FIFO mode 01b: TX FIFO mode 10b: CAN – CAN GW FIFO mode 11b: Reserved
7	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
6 to 4	CFPLS*1[2:0]	0h	RW	Common FIFO Payload Data Size Configuration 000b: 8 bytes 001b: 12 bytes 010b: 16 bytes 011b: 20 bytes 100b: 24 bytes 101b: 32 bytes 110b: 48 bytes 111b: 64 bytes
3	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	CFTXIE	0h	RW	Common FIFO TX Interrupt Enable 0b: FIFO interrupt generation disabled for Frame TX 1b: FIFO interrupt generation enabled for Frame TX
1	CFRXIE	0h	RW	Common FIFO RX Interrupt Enable 0b: FIFO interrupt generation disabled for Frame RX 1b: FIFO interrupt generation enabled for Frame RX
0	CFE	0h	RW	Common FIFO Enable 0b: FIFO disabled 1b: FIFO enabled

Note 1. These bits are not available in the classical CAN function.

CFE bit (Common FIFO Enable)

The CFE bit enables the FIFO when set. FIFO is disabled when this bit is cleared.

This bit can also be used, by clearing it, to abort transmission from Common FIFO when configured in TX mode or GW mode, or to stop reception into the Common FIFO in RX mode.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFOs configured as TX or GW FIFO.

This bit can only be set if the configured FIFO depth is greater than 0 (CFDC bit > 0).

Set the CFE bit with a separate write access to the CFDCFCCn register, after all the other bits in this register are set.

This bit is cleared automatically when the CAN-FD module is in GL_RESET mode.

This bit is also cleared automatically when the related channel is in CH_RESET mode if the FIFO is configured in TX or GW mode.

CFRXIE bit (Common FIFO RX Interrupt Enable)

The CFRXIE bit enables generation of FIFO interrupts when the interrupt flag is set after reception of a frame in the corresponding FIFO buffer.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

CFTXIE bit (Common FIFO TX Interrupt Enable)

The CFTXIE bit enables generation of common FIFO interrupts when the interrupt flag is set after transmission of a frame from the corresponding FIFO buffer.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

CFPLS[2:0] bits (Common FIFO Payload Data Size Configuration)

The CFPLS[2:0] bits define the message data payload allocation in the RAM. This is the maximum number of bytes which can be received or transmitted by the FIFO buffer.

For details, see **7.9.6 FIFO Buffers and Normal Message Buffer Configuration**.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

Note: These bits are not available in the classical CAN function.

CFM[1:0] bits (Common FIFO Mode)

The CFM[1:0] bits select the mode of the FIFO. When a hardware reset is applied, all the Common FIFO buffers are configured in RX FIFO mode. Do not configure these bits to 11b.

Do not write to these bits in GL_OPERATION or GL_SLEEP mode.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

CFITSS bit (Common FIFO Interval Timer Source Select)

The CFITSS bit selects the basic clock source for the Interval Transmission Timer.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode. In addition, do not write to this bit when the CFE bit is set to 1b.

Do not write 1b to this bit when CAN-FD communication is used.*[1]

Note: The bit time clock can vary depending on the nominal and data rate bit configuration.

Note 1. This feature is not available in the classical CAN function.

CFITR bit (Common FIFO Interval Timer Resolution)

The CFITR bit selects the resolution of the reference clock for the Interval Transmission Timer (peripheral clock is the source for the reference clock).

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode. Also, do not write to this bit when the CFE bit is set to 1b.

CFIM bit (Common FIFO Interrupt Mode)

The CFIM bit selects the interrupt generation condition for the FIFO buffer.

Do not write to this bit in GL_SLEEP mode.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

CFIGCV[2:0] bits (Common FIFO Interrupt Generation Counter Value)

The CFIGCV[2:0] bits select the message counter value for the generation of FIFO interrupts. These values represent fractions of the FIFO depth at which the interrupt is to be generated.

Do not write to these bits when the CAN-FD module is in GL_SLEEP mode.

The setting of these bits should be synchronized with the CFDC[2:0] bits.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

CFTML[4:0] bits (Common FIFO TX Message Buffer Link)

The CFTML[4:0] bits select the normal transmit message buffer position where the TX or GW FIFO is linked to, for transmission scanning.

Do not write to these bits in GL_OPERATION or GL_SLEEP mode.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

CFDC[2:0] bits (Common FIFO Depth Configuration)

The CFDC[2:0] bits select the depth of the common FIFO in terms of the number of messages. If the FIFO depth is configured to 0 message, the FIFO cannot be used.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

CFITT[7:0] bits (Common FIFO Interval Transmission Time)

The CFITT[7:0] bits select the delay in the start of transmission for all messages transmitted from this FIFO buffer when configured in TX or GW mode. The delay is a multiple of the basic interval timer clock source period (reference clock × 1, reference clock × 10, or bit time clock of the related CAN channel).

Do not write to these bits when the CAN-FD module is in GL_SLEEP mode.

Do not write to these bits when the CFE bit is set to 1b.

When CFDGCFG.ITRCP[15:0] = 0000h, set the CFITT[7:0] bits to 0000h.

7.9.2.2.30 Common FIFO Configuration/Control Enhancement Register n (CFDCFCCEn) (n = 0 to 17)

The CFDCFCCEn register (n = 0 to 17) is used to configure the Common FIFOs.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 0180h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CFBME
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CFMOWM	-	-	-	-	-	CFOFTXIE	CFOFRXIE	CFFIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
16	CFBME	0h	RW	Common FIFO Buffering Mode Enable 0b: Transmission from Common FIFO 1b: Transmission halt from Common FIFO
15 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8	CFMOWM	0h	RW	Common FIFO Message Overwrite Mode 0b: Message discarded mode 1b: Message overwrite mode
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	CFOFTXIE	0h	RW	Common FIFO One Frame Transmission Interrupt Enable 0b: One Frame TX interrupt generation disabled 1b: One Frame TX interrupt generation enabled
1	CFOFRXIE	0h	RW	Common FIFO One Frame Reception Interrupt Enable 0b: One Frame RX interrupt generation disabled 1b: One Frame RX interrupt generation enabled
0	CFFIE	0h	RW	Common FIFO Full Interrupt Enable 0b: FIFO Interrupt generation disabled 1b: FIFO Interrupt generation enabled

CFFIE bit (Common FIFO Full Interrupt Enable)

The CFFIE bit enables generation of the FIFO full interrupt when the interrupt flag is set after reception of a frame in the corresponding FIFO buffer.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

The following content shows examples of interruptions:

1. Interruption output in number of arbitrary stages (CFDCFCCn.CFIGCV).
2. Interruption output in FIFO full state.

Management of the receiving data of FIFO can be performed by these notices of interruption.

CFOFRXIE bit (Common FIFO One Frame Reception Interrupt Enable)

The CFOFRXIE bit enables generation of the one frame reception interrupt when the interrupt flag is set after reception of a frame in the corresponding FIFO buffer.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

CFOFTXIE bit (Common FIFO One Frame Transmission Interrupt Enable)

The CFOFTXIE bit enables generation of the one frame transmission interrupt when the interrupt flag is set after transmission of a frame in the corresponding FIFO buffer.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

CFMOWM bit (Common FIFO Message Overwrite Mode)

When the CFMOWM bit is 0b, a receiving message is discarded and FIFO is full. When the CFMOWM bit is 1b, a receiving message is overwritten and FIFO is full.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode.

Only write 1b to this bit when the common FIFO is in GW mode.

Do not write 1b to this bit when the CFE bit is 1b.

CFBME bit (Common FIFO Buffering Mode Enable)

When the CFBME bit is 0b, messages are transmitted from FIFO. When the CFBME bit is 1b, messages are not transmitted from FIFO.

Do not write to this bit when the CAN-FD module is in GL_SLEEP mode. Additionally, do not write 1b to this bit when the CFE bit is 1b.

7.9.2.2.31 Common FIFO Status Register n (CFDCFSTSn) (n = 0 to 17)

The CFDCFSTSn register (n = 0 to 17) shows the status of messages stored in the corresponding FIFO buffers.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 01E0h + n x 0004h

Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	CFMOW	-	-	-	-	-	CFOFTXIF	CFOFRXIF	CFFIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFMC[7:0]								-	-	-	CFTXIF	CFRXIF	CFMLT	CFFLL	CFEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
24	CFMOW	0h	RW	Common FIFO Message Overwrite 0b: No message overwrite occurred in FIFO 1b: Message overwrite occurred in FIFO
23 to 19	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18	CFOFTXIF	0h	RW	Common FIFO One Frame Transmission Interrupt Flag For each FIFO that transmits a frame, a corresponding interrupt is set.
17	CFOFRXIF	0h	RW	Common FIFO One Frame Reception Interrupt Flag For each FIFO that receives a frame, a corresponding interrupt is set.
16	CFFIF	0h	RW	Common FIFO Full Interrupt Flag 0b: Interrupt condition not satisfied for FIFO full interrupt 1b: Interrupt condition satisfied for FIFO full interrupt
15 to 8	CFMC[7:0]	0h	R	Common FIFO Message Count Number of messages stored in FIFO
7 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	CFTXIF	0h	RW	Common TX FIFO Interrupt Flag 0b: FIFO interrupt condition not satisfied after frame transmission 1b: FIFO Interrupt condition satisfied after frame transmission
3	CFRXIF	0h	RW	Common RX FIFO Interrupt Flag 0b: FIFO interrupt condition not satisfied after frame reception 1b: FIFO interrupt condition satisfied after frame reception
2	CFMLT	0h	RW	Common FIFO Message Lost 0b: FIFO message not lost 1b: FIFO message lost
1	CFFLL	0h	R	Common FIFO Full 0b: FIFO not full 1b: FIFO full
0	CFEMP	1h	R	Common FIFO Empty 0b: FIFO not empty 1b: FIFO empty

CFEMP bit (Common FIFO Empty)

The CFEMP bit is set automatically when:

- The CPU has read all messages from the FIFO configured in RX mode
- All messages have been transmitted from the FIFO configured in TX or GW mode
- The FIFO is disabled by setting the CFE bit to 0b

- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET when FIFO configured in TX or GW mode.

The CFEMP bit is cleared automatically when:

- The first reception message is stored in the FIFO buffer when configured in RX mode
- The first message to be transmitted is stored in the FIFO buffer when configured in TX or GW mode.

CFFLL bit (Common FIFO Full)

The CFFLL bit is set automatically when the number of CAN messages stored in the FIFO matches the configured FIFO depth.

The CFFLL bit is cleared automatically when:

- The number of CAN messages stored in the FIFO is less than the configured FIFO depth
- The FIFO is disabled by setting the CFE bit to 0b
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode when FIFO buffer is configured in TX or GW mode.

CFMLT bit (Common FIFO Message Lost)

The CFMLT bit is set automatically whenever a message is lost due to attempted storage of a new message when FIFO is already full in RX or GW mode.

If a set from the CAN channel occurs simultaneously with a clear by a write access, then this bit is set.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFO configured as TX or GW FIFO. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

The CFMLT bit is cleared:

- By writing 0b to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in TX or GW mode.

CFRXIF bit (Common RX FIFO Interrupt Flag)

The CFRXIF bit is not cleared automatically if the Common FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFO configured as TX or GW FIFO. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

This bit is set automatically when the configured interrupt condition is satisfied for Common FIFO buffers when configured in GW mode or RX mode.

The CFRXIF bit is cleared:

- By writing 0b to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in GW mode.

CFTXIF bit (Common TX FIFO Interrupt Flag)

The CFTXIF bit is not cleared automatically if the Common FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFO buffer configured as TX or GW FIFO. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

This bit is set automatically when the configured interrupt condition is satisfied for Common FIFO buffers configured in GW or TX mode.

The CFTXIF bit is cleared:

- By writing 0b to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in TX or GW mode.

CFMC[7:0] bits (Common FIFO Message Count)

The CFMC[7:0] bits indicate the following:

- Number of CAN messages stored by the CPU in the FIFO buffer configured in TX mode pending for transmission
- Number of CAN messages stored in the FIFO buffer configured in RX mode by CAN-FD to be read by the CPU
- Number of CAN messages stored by the CAN-FD in the GW FIFO pending for transmission.

The CFMC[7:0] bits are cleared automatically when:

- The FIFO is disabled
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in TX or GW mode.

CFFIF bit (Common FIFO Full Interrupt Flag)

The CFFIF bit is not cleared automatically if the Common FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFO configured as TX or GW FIFO. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

This bit is set automatically when the FIFO Full Interrupt condition is satisfied for Common FIFO buffers when configured in GW or RX mode.

If a set from the CAN channel occurs simultaneously with a clear by a write access, then the bit is set.

The CFFIF bit is cleared:

- By writing 0b to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in TX or GW mode.

CFOFRXIF bit (Common FIFO One Frame Reception Interrupt Flag)

The CFOFRXIF bit is not cleared automatically if the Common FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for the FIFO buffer configured as TX or GW FIFO. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

This bit is set automatically when the One Frame Reception Interrupt condition is satisfied for the Common FIFO buffers when configured in GW or RX mode.

If a set from the CAN channel occurs simultaneously with a clear by a write access, then the bit is set.

The CFOFRXIF bit is cleared:

- By writing 0b to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in TX or GW mode.

This bit is not affected by the CFIM bit of CFDCFCCn.

CFOFTXIF bit (Common FIFO One Frame Transmission Interrupt Flag)

The CFOFTXIF bit is not cleared automatically if the Common FIFO buffer is disabled.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFOs configured as TX or GW FIFO. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

This bit is set automatically when the One Frame Transmission Interrupt condition is satisfied for Common FIFO buffers configured in GW mode or TX mode.

If a set from the CAN channel occurs simultaneously with a clear by a write access, then the bit is set.

The bit is cleared:

- By writing 0b to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO is configured in TX or GW mode.

This bit is not affected by the CFIM bit of CFDCFCCn.

CFMOW bit (Common FIFO Message Overwrite)

The CFMOW bit is set automatically whenever a message is an overwrite storage of a new message when CFDCFCCEn.CFMOWM = 1b and FIFO is already full in GW mode.

If the set from the CAN channel occurs simultaneously with the clear by a write access, the bit is set.

Only write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode and the related CAN-FD channel is not in CH_RESET mode for FIFO buffer configured as TX or GW FIFO. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

The CFMOW bit is cleared:

- By writing 0b to it
- When the CAN-FD module is in GL_RESET mode
- When the related CAN-FD channel is in CH_RESET mode if the FIFO buffer is configured in TX or GW mode.

7.9.2.2.32 Common FIFO Pointer Control Register n (CFDCFPCTRn) (n = 0 to 17)

The CFDCFPCTRn register (n = 0 to 17) can be used to increment the read or write pointer of the corresponding Common FIFO buffer.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 0240h + n x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CFPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	W	Reserved The written value should always be 0b.
7 to 0	CFPC[7:0]	0h	W	Common FIFO Pointer Control Increments read or write pointer of the corresponding Common FIFO buffers depending on the mode configuration.

CFPC[7:0] bits (Common FIFO Pointer Control)

When the value FFh is written into the CFPC[7:0] bits, the read pointer of the corresponding Common FIFO buffer (when configured in RX mode), or the write pointer of the corresponding Common FIFO buffer (when configured in TX mode) moves to the next FIFO entry.

The read value from these bits is always 00h.

Only write to these bits when the CAN-FD module is in GL_HALT or GL_OPERATION mode.

Only write FFh to this register when:

- The Common FIFO buffer is enabled and is not empty if configured in RX mode
- The Common FIFO buffer is enabled and is not full if configured in TX mode
- The Common FIFO buffer is enabled and is not configured in GW mode.

Do not write to the Common FIFO Pointer Control registers when DMA is enabled.

7.9.2.2.33 FIFO Empty Status Register (CFDFESTS)

The CFDFESTS register shows status of the empty bits of the FIFO buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 02A0h
Initial Value : 0000_3FFFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	CFXEMP[17:8]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFXEMP[7:0]								RFXEMP[7:0]							
Initial Value	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read.
25 to 8	CFXEMP[17:0]	3Fh	R	Common FIFO Empty Status 0b: Corresponding FIFO not empty 1b: Corresponding FIFO empty
7 to 0	RFXEMP[7:0]	FFh	R	RX FIFO Empty Status 0b: Corresponding FIFO not empty 1b: Corresponding FIFO empty

RFXEMP[7:0] bits (RX FIFO Empty Status)

Bit [7] (RFXEMP[7]) is associated with FIFO index 7 and bit [0] (RFXEMP[0]) is associated with FIFO index 0.

The RFXEMP[7:0] bits are set when the CAN-FD module is in GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the RX FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the RX FIFO Status Registers.

CFXEMP[17:0] bits (Common FIFO Empty Status)

Bit [25] (CFXEMP[17]) is associated with common FIFO index 17 and bit [8] (CFXEMP[0]) is associated with common FIFO index 0.

The CFXEMP[17:0] bits are set when the CAN-FD module is in GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

7.9.2.2.34 FIFO Full Status Register (CFDFFSTS)

The CFDFFSTS register shows status of the full bits of the FIFO buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 02A4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	CFXFLL[17:8]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFXFLL[7:0]								RFXFLL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read.
25 to 8	CFXFLL[17:0]	0h	R	Common FIFO Full Status 0b: Corresponding FIFO not full 1b: Corresponding FIFO full
7 to 0	RFXFLL[7:0]	0h	R	RX FIFO Full Status 0b: Corresponding FIFO not full 1b: Corresponding FIFO full

RFXFLL[7:0] bits (RX FIFO Full Status)

The RFXFLL[7:0] bits are cleared when CAN-FD module is in GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the RX FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the RX FIFO Status Registers.

CFXFLL[17:0] bits (Common FIFO Full Status)

The CFXFLL[17:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

7.9.2.2.35 FIFO Message Lost Status Register (CFDFMSTS)

The CFDFMSTS register shows status of the message lost bits of the FIFO buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 02A8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	CFXMLT[17:8]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFXMLT[7:0]								RFXMLT[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read.
25 to 8	CFXMLT[17:0]	0h	R	Common FIFO Message Lost Status 0b: Corresponding FIFO Message Lost flag not set 1b: Corresponding FIFO Message Lost flag set
7 to 0	RFXMLT[7:0]	0h	R	RX FIFO Message Lost Status 0b: Corresponding FIFO Message Lost flag not set 1b: Corresponding FIFO Message Lost flag set

RFXMLT[7:0] bits (RX FIFO Message Lost Status)

The RFXMLT[7:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the RX FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the RX FIFO Status Registers.

CFXMLT[17:0] bits (Common FIFO Message Lost Status)

The CFXMLT[17:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

7.9.2.2.36 RX FIFO Interrupt Flag Status Register (CFDRFISTS)

The CFDRFISTS register shows status of the interrupt flag bits of the RX FIFO buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 02ACh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	RFXFFLL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RFXIF[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read.
23 to 16	RFXFFLL[7:0]	0h	R	RX FIFO[x] Interrupt Full Flag Status 0b: Corresponding RX FIFO Interrupt Full flag not set 1b: Corresponding RX FIFO Interrupt Full flag set
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read.
7 to 0	RFXIF[7:0]	0h	R	RX FIFO[x] Interrupt Flag Status 0b: Corresponding RX FIFO Interrupt flag not set 1b: Corresponding RX FIFO Interrupt flag set

RFXIF[7:0] bits (RX FIFO[x] Interrupt Flag Status)

Each bit is set automatically when the corresponding interrupt flag bit is set in the RX FIFO Status Registers.

The RFXIF[7:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is cleared automatically when the corresponding interrupt flag bit is cleared in the RX FIFO Status Registers.

RFXFFLL[7:0] bits (RX FIFO[x] Interrupt Full Flag Status)

Each bit is set automatically when the corresponding interrupt full flag bit is set in the RX FIFO Status Registers.

The RFXFFLL[7:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is cleared automatically when the corresponding interrupt full flag bit is cleared in the RX FIFO Status Registers.

7.9.2.2.37 Common FIFO RX Interrupt Flag Status Register (CFDCFRISTS)

The CFDCFRISTS register shows status of the interrupt flag bits of the Common FIFO buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 02B0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CFXRXIF[17:16]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFXRXIF[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read.
17 to 0	CFXRXIF[17:0]	0h	R	Common FIFO RX Interrupt Flag Status 0b: Corresponding Common FIFO RX Interrupt flag not set 1b: Corresponding Common FIFO RX Interrupt flag set

CFXRXIF[17:0] bits (Common FIFO RX Interrupt Flag Status)

Each bit is set automatically when the corresponding RX interrupt flag bit is set in the Common FIFO Status Registers.

The CFXRXIF[17:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is cleared automatically when the corresponding RX interrupt flag bit is cleared in the Common FIFO Status Registers.

7.9.2.2.38 Common FIFO TX Interrupt Flag Status Register (CFDCFTISTS)

The CFDCFTISTS register shows status of the interrupt flag bits of the Common FIFO buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 02B4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CFXTXIF[17:16]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFXTXIF[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read.
17 to 0	CFXTXIF[17:0]	0h	R	Common FIFO TX Interrupt Flag Status 0b: Corresponding Common FIFO TX Interrupt flag not set 1b: Corresponding Common FIFO TX Interrupt flag set

CFXTXIF[17:0] bits (Common FIFO TX Interrupt Flag Status)

Each bit is set automatically when the corresponding TX interrupt flag bit is set in the Common FIFO Status Registers.

The CFXTXIF[17:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is cleared automatically when the corresponding TX interrupt flag bit is cleared in the Common FIFO Status Registers.

7.9.2.2.39 FIFO FDC Full Status Register (CFDFFFSTS)

The CFDFFFSTS register shows status of the full interrupt flag bits of the FIFO buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 02C4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	CFXFFLL[17:8]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFXFFLL[7:0]								RFXFFLL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read.
25 to 8	CFXFFLL[17:0]	0h	R	COMMON FIFO FDC Level Full Status 0b: Corresponding FIFO full interrupt not set 1b: Corresponding FIFO full interrupt set
7 to 0	RFXFFLL[7:0]	0h	R	RX FIFO FDC Level Full Status 0b: Corresponding FIFO full interrupt not set 1b: Corresponding FIFO full interrupt set

RFXFFLL[7:0] bits (RX FIFO FDC Level Full Status)

Each bit is set automatically when the corresponding bit is set in the RX FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the RX FIFO Status Registers.

The RFXFFLL[7:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

CFXFFLL[17:0] bits (COMMON FIFO FDC Level Full Status)

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

The CFXFFLL[17:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

7.9.2.2.40 Common FIFO Message Overwrite Status Register (CFDCFMOWSTS)

The CFDCFMOWSTS register shows the status of the Interrupt Flag bits of the Common FIFO Buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 02C0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CFXMOW[17:16]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFXMOW[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read.
17 to 0	CFXMOW[17:0]	0h	R	Common FIFO Massage Overwrite Status 0b: Corresponding FIFO Overwrite flag not set 1b: Corresponding FIFO Overwrite flag set

CFXMOW[17:0] bits (Common FIFO Massage Overwrite Status)

The CFXMOW[17:0] bits are cleared when the CAN-FD module is in GL_RESET mode. This register is only valid in GW mode.

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

7.9.2.2.41 Common FIFO One Frame RX Interrupt Flag Status Register (CFDCFOFRISTS)

The CFDCFOFRISTS register shows status of the interrupt flag bits of the Common FIFO buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 02B8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CFXOFRXIF [17:16]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFXOFRXIF[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read.
17 to 0	CFXOFRXIF [17:0]	0h	R	Common FIFO One Frame RX Interrupt Flag Status 0b: Corresponding Common FIFO One Frame RX Interrupt flag not set 1b: Corresponding Common FIFO One Frame RX Interrupt flag set

CFXOFRXIF[17:0] bits (Common FIFO One Frame RX Interrupt Flag Status)

Each bit is set automatically when the corresponding One Frame RX Interrupt flag bit is set in the Common FIFO Status Registers.

The CFXOFRXIF[17:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is cleared automatically when the corresponding One Frame RX Interrupt flag bit is cleared in the Common FIFO Status Registers.

7.9.2.2.42 Common FIFO One Frame TX Interrupt Flag Status Register (CFDCFOFTISTS)

The CFDCFOFTISTS register shows status of the interrupt flag bits of the Common FIFO buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 02BCh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CFXOFTXIF [17:16]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFXOFTXIF[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read.
17 to 0	CFXOFTXIF [17:0]	0h	R	Common FIFO One Frame TX Interrupt Flag Status 0b: Corresponding Common FIFO One Frame TX Interrupt flag not set 1b: Corresponding Common FIFO One Frame TX Interrupt flag set

CFXOFTXIF[17:0] bits (Common FIFO One Frame TX Interrupt Flag Status)

Each bit is set automatically when the corresponding One Frame TX Interrupt flag bit is set in the Common FIFO Status Registers.

The CFXOFTXIF[17:0] bits are cleared when the CAN-FD module is in GL_RESET mode.

Each bit is cleared automatically when the corresponding One Frame TX Interrupt flag bit is cleared in the Common FIFO Status Registers.

7.9.2.2.43 DMA Transfer Control Register (CFDCDTCT)

The CFDCDTCT register controls the start and stop of DMA transfer operation.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1330h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	CFDMAE5	CFDMAE4	CFDMAE3	CFDMAE2	CFDMAE1	CFDMAE0	RFDMAE7	RFDMAE6	RFDMAE5	RFDMAE4	RFDMAE3	RFDMAE2	RFDMAE1	RFDMAE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
13	CFDMAE5	0h	RW	DMA Transfer Enable for Common FIFO 0 of Channel 5 0b: DMA transfer request disabled 1b: DMA transfer request enabled
12	CFDMAE4	0h	RW	DMA Transfer Enable for Common FIFO 0 of Channel 4 0b: DMA transfer request disabled 1b: DMA transfer request enabled
11	CFDMAE3	0h	RW	DMA Transfer Enable for Common FIFO 0 of Channel 3 0b: DMA transfer request disabled 1b: DMA transfer request enabled
10	CFDMAE2	0h	RW	DMA Transfer Enable for Common FIFO 0 of Channel 2 0b: DMA transfer request disabled 1b: DMA transfer request enabled
9	CFDMAE1	0h	RW	DMA Transfer Enable for Common FIFO 0 of Channel 1 0b: DMA transfer request disabled 1b: DMA transfer request enabled
8	CFDMAE0	0h	RW	DMA Transfer Enable for Common FIFO 0 of Channel 0 0b: DMA transfer request disabled 1b: DMA transfer request enabled
7	RFDMAE7	0h	RW	DMA Transfer Enable for RX FIFO 7 0b: DMA transfer request disabled 1b: DMA transfer request enabled
6	RFDMAE6	0h	RW	DMA Transfer Enable for RX FIFO 6 0b: DMA transfer request disabled 1b: DMA transfer request enabled
5	RFDMAE5	0h	RW	DMA Transfer Enable for RX FIFO 5 0b: DMA transfer request disabled 1b: DMA transfer request enabled
4	RFDMAE4	0h	RW	DMA Transfer Enable for RX FIFO 4 0b: DMA transfer request disabled 1b: DMA transfer request enabled
3	RFDMAE3	0h	RW	DMA Transfer Enable for RX FIFO 3 0b: DMA transfer request disabled 1b: DMA transfer request enabled
2	RFDMAE2	0h	RW	DMA Transfer Enable for RX FIFO 2 0b: DMA transfer request disabled 1b: DMA transfer request enabled
1	RFDMAE1	0h	RW	DMA Transfer Enable for RX FIFO 1 0b: DMA transfer request disabled 1b: DMA transfer request enabled

Bit	Bit Name	Initial Value	R/W	Description
0	RFDMAE0	0h	RW	DMA Transfer Enable for RX FIFO 0 0b: DMA transfer request disabled 1b: DMA transfer request enabled

RFDMAEn (n = 0 to 7) bit (DMA Transfer Enable for RX FIFOn)

Number of RX FIFOs = 8

The RFDMAEn bit cannot be set in GL_SLEEP or GL_RESET mode.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

CFDMAEn (n = 0 to 5) bit (DMA Transfer Enable for Common FIFO 0 of Channels 0 to 5)

The CFDMAEn bit enables or disables DMA transfer request for common FIFO 0 of channels 0 to 5. Only Common FIFO 0 can be linked to a DMA channel with this bit. Common FIFO 1 cannot be linked to a DMA channel.

To link Common FIFO 2, see bit CFDCDTTCT.CFDMAEn in **7.9.2.2.45 DMA TX Transfer Control Register (CFDCDTTCT)**.

The CFDMAEn bit cannot be set in GL_SLEEP or GL_RESET mode.

Do not enable a DMA transfer for a Common FIFO that is configured as TX or GW FIFO.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

7.9.2.2.44 DMA Transfer Status Register (CFDCDTSTS)

The CFDCDTSTS register shows the status of the DMA transfer.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1334h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	CFDM ASTS5	CFDM ASTS4	CFDM ASTS3	CFDM ASTS2	CFDM ASTS1	CFDM ASTS0	RFDM ASTS7	RFDM ASTS6	RFDM ASTS5	RFDM ASTS4	RFDM ASTS3	RFDM ASTS2	RFDM ASTS1	RFDM ASTS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read.
13	CFDMASTS5	0h	R	DMA Transfer Status only for Common FIFO 0 of Channel 5 0b: DMA transfer stopped 1b: DMA transfer ongoing
12	CFDMASTS4	0h	R	DMA Transfer Status only for Common FIFO 0 of Channel 4 0b: DMA transfer stopped 1b: DMA transfer ongoing
11	CFDMASTS3	0h	R	DMA Transfer Status only for Common FIFO 0 of Channel 3 0b: DMA transfer stopped 1b: DMA transfer ongoing
10	CFDMASTS2	0h	R	DMA Transfer Status only for Common FIFO 0 of Channel 2 0b: DMA transfer stopped 1b: DMA transfer ongoing
9	CFDMASTS1	0h	R	DMA Transfer Status only for Common FIFO 0 of Channel 1 0b: DMA transfer stopped 1b: DMA transfer ongoing
8	CFDMASTS0	0h	R	DMA Transfer Status only for Common FIFO 0 of Channel 0 0b: DMA transfer stopped 1b: DMA transfer ongoing
7	RFDMASTS7	0h	R	DMA Transfer Status for RX FIFO 7 0b: DMA transfer stopped 1b: DMA transfer ongoing
6	RFDMASTS6	0h	R	DMA Transfer Status for RX FIFO 6 0b: DMA transfer stopped 1b: DMA transfer ongoing
5	RFDMASTS5	0h	R	DMA Transfer Status for RX FIFO 5 0b: DMA transfer stopped 1b: DMA transfer ongoing
4	RFDMASTS4	0h	R	DMA Transfer Status for RX FIFO 4 0b: DMA transfer stopped 1b: DMA transfer ongoing
3	RFDMASTS3	0h	R	DMA Transfer Status for RX FIFO 3 0b: DMA transfer stopped 1b: DMA transfer ongoing
2	RFDMASTS2	0h	R	DMA Transfer Status for RX FIFO 2 0b: DMA transfer stopped 1b: DMA transfer ongoing
1	RFDMASTS1	0h	R	DMA Transfer Status for RX FIFO 1 0b: DMA transfer stopped 1b: DMA transfer ongoing

Bit	Bit Name	Initial Value	R/W	Description
0	RFDMASTS0	0h	R	DMA Transfer Status for RX FIFO 0 0b: DMA transfer stopped 1b: DMA transfer ongoing

RFDMASTSn (n = 0 to 7) bit (DMA Transfer Status for RX FIFO n)

Number of RX FIFOs = 8

Each bit is set automatically when the corresponding DMA enable bit is set and the corresponding DMA FIFO is not empty.

Each bit is cleared automatically when the DMA transfer stops either because the DMA is disabled or the DMA FIFO is empty.

When CFDCDTCT.RFDMAEn (see CFDCDTCT.RFDMAEn bit in **7.9.2.2.43 DMA Transfer Control Register (CFDCDTCT)**) is set to 0b while DMA transfer for the corresponding FIFO is ongoing, the RFDMASTSn bit becomes 0b when the DMA transfer is complete.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

CFDMASTSn bit (DMA Transfer Status only for Common FIFO 0 of Channels 0 to 5)

Each bit is set automatically when the corresponding DMA enable bit is set and the corresponding DMA FIFO is not empty.

Each bit is cleared automatically when the DMA transfer stops either because the DMA is disabled or the DMA FIFO is empty.

When CFDCDTCT.CFDMAEn (see CFDCDTCT.CFDMAEn bit in **7.9.2.2.43 DMA Transfer Control Register (CFDCDTCT)**) is set to 0b while DMA transfer for the corresponding FIFO is ongoing, the CFDMASTSn bit becomes 0b when the DMA transfer is complete.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

7.9.2.2.45 DMA TX Transfer Control Register (CFDCDTTCT)

The CFDCDTTCT register controls the start and stop of DMA transfer operation.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1340h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	CFDMAE5	CFDMAE4	CFDMAE3	CFDMAE2	CFDMAE1	CFDMAE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	TQ3DMAE5	TQ3DMAE4	TQ3DMAE3	TQ3DMAE2	TQ3DMAE1	TQ3DMAE0	-	-	TQ0DMAE5	TQ0DMAE4	TQ0DMAE3	TQ0DMAE2	TQ0DMAE1	TQ0DMAE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
21	CFDMAE5	0h	RW	DMA TX Transfer Enable for Common FIFO 2 of Channel 5 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
20	CFDMAE4	0h	RW	DMA TX Transfer Enable for Common FIFO 2 of Channel 4 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
19	CFDMAE3	0h	RW	DMA TX Transfer Enable for Common FIFO 2 of Channel 3 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
18	CFDMAE2	0h	RW	DMA TX Transfer Enable for Common FIFO 2 of Channel 2 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
17	CFDMAE1	0h	RW	DMA TX Transfer Enable for Common FIFO 2 of Channel 1 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
16	CFDMAE0	0h	RW	DMA TX Transfer Enable for Common FIFO 2 of Channel 0 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
15 to 14	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
13	TQ3DMAE5	0h	RW	DMA TX Transfer Enable for TXQ 3 of Channel 5 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
12	TQ3DMAE4	0h	RW	DMA TX Transfer Enable for TXQ 3 of Channel 4 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
11	TQ3DMAE3	0h	RW	DMA TX Transfer Enable for TXQ 3 of Channel 3 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
10	TQ3DMAE2	0h	RW	DMA TX Transfer Enable for TXQ 3 of Channel 2 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
9	TQ3DMAE1	0h	RW	DMA TX Transfer Enable for TXQ 3 of Channel 1 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
8	TQ3DMAE0	0h	RW	DMA TX Transfer Enable for TXQ 3 of Channel 0 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
7 to 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

Bit	Bit Name	Initial Value	R/W	Description
5	TQ0DMAE5	0h	RW	DMA TX Transfer Enable for TXQ 0 of Channel 5 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
4	TQ0DMAE4	0h	RW	DMA TX Transfer Enable for TXQ 0 of Channel 4 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
3	TQ0DMAE3	0h	RW	DMA TX Transfer Enable for TXQ 0 of Channel 3 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
2	TQ0DMAE2	0h	RW	DMA TX Transfer Enable for TXQ 0 of Channel 2 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
1	TQ0DMAE1	0h	RW	DMA TX Transfer Enable for TXQ 0 of Channel 1 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled
0	TQ0DMAE0	0h	RW	DMA TX Transfer Enable for TXQ 0 of Channel 0 0b: DMA TX transfer request disabled 1b: DMA TX transfer request enabled

TQ0DMAEn (n = 0 to 5) bit (DMA TX Transfer Enable for TXQ 0 of Channel n)

The TQ0DMAEn bit cannot be set in GL_SLEEP or GL_RESET mode.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

TQ3DMAEn (n = 0 to 5) bit (DMA TX Transfer Enable for TXQ 3 of Channel n)

The TQ3DMAEn bit cannot be set in GL_SLEEP or GL_RESET mode.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

CFDMAEn (n = 0 to 5) bit (DMA TX Transfer Enable for Common FIFO 2 of Channel n)

The CFDMAEn bit cannot be set in GL_SLEEP or GL_RESET mode.

Only common FIFO 2 can be linked to a DMA channel with this bit. Common FIFO 1 cannot be linked to a DMA channel. To link Common FIFO 0, see CFDCDTCT.CFDMAEn bit in **7.9.2.2.43 DMA Transfer Control Register (CFDCDTCT)**.

Do not enable a DMA transfer for a Common FIFO that is configured as RX or GW FIFO.

The CFDMAEn bit is cleared when the CAN-FD module is in GL_RESET mode.

7.9.2.2.46 DMA TX Transfer Status Register (CFDCDTTSTS)

The CFDCDTTSTS register shows the status of the DMA TX transfer.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1344h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	CFDMASTS5	CFDMASTS4	CFDMASTS3	CFDMASTS2	CFDMASTS1	CFDMASTS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	TQ3DMASTS5	TQ3DMASTS4	TQ3DMASTS3	TQ3DMASTS2	TQ3DMASTS1	TQ3DMASTS0	-	-	TQ0DMASTS5	TQ0DMASTS4	TQ0DMASTS3	TQ0DMASTS2	TQ0DMASTS1	TQ0DMASTS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	-	All 0	R	Reserved Whenever it is read, 0b is read.
21	CFDMASTS5	0h	R	DMA TX Transfer Status for Common FIFO 2 of Channel 5 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
20	CFDMASTS4	0h	R	DMA TX Transfer Status for Common FIFO 2 of Channel 4 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
19	CFDMASTS3	0h	R	DMA TX Transfer Status for Common FIFO 2 of Channel 3 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
18	CFDMASTS2	0h	R	DMA TX Transfer Status for Common FIFO 2 of Channel 2 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
17	CFDMASTS1	0h	R	DMA TX Transfer Status for Common FIFO 2 of Channel 1 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
16	CFDMASTS0	0h	R	DMA TX Transfer Status for Common FIFO 2 of Channel 0 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
15 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read.
13	TQ3DMASTS5	0h	R	DMA TX Transfer Status for TXQ3 of Channel 5 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
12	TQ3DMASTS4	0h	R	DMA TX Transfer Status for TXQ3 of Channel 4 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
11	TQ3DMASTS3	0h	R	DMA TX Transfer Status for TXQ3 of Channel 3 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
10	TQ3DMASTS2	0h	R	DMA TX Transfer Status for TXQ3 of Channel 2 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
9	TQ3DMASTS1	0h	R	DMA TX Transfer Status for TXQ3 of Channel 1 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
8	TQ3DMASTS0	0h	R	DMA TX Transfer Status for TXQ3 of Channel 0 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
7 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read.

Bit	Bit Name	Initial Value	R/W	Description
5	TQ0DMASTS5	0h	R	DMA TX Transfer Status for TXQ0 of Channel 5 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
4	TQ0DMASTS4	0h	R	DMA TX Transfer Status for TXQ0 of Channel 4 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
3	TQ0DMASTS3	0h	R	DMA TX Transfer Status for TXQ0 of Channel 3 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
2	TQ0DMASTS2	0h	R	DMA TX Transfer Status for TXQ0 of Channel 2 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
1	TQ0DMASTS1	0h	R	DMA TX Transfer Status for TXQ0 of Channel 1 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled
0	TQ0DMASTS0	0h	R	DMA TX Transfer Status for TXQ0 of Channel 0 0b: DMA TX transfer stopped 1b: DMA TX transfer enabled

TQ0DMASTSn (n = 0 to 5) bit (DMA TX Transfer Status for TXQ 0 of Channel n)

The TQ0DMASTSn bit is set when the CFDCDTTCT.TQ0DMAEn bit in the corresponding CFDCDTTCT register is set (see **7.9.2.2.45 DMA TX Transfer Control Register (CFDCDTTCT)**).

This bit is cleared when:

- The CFDCDTTCT.TQ0DMAEn bit in the corresponding CFDCDTTCT register is cleared
- The CAN-FD module is in GL_RESET mode.

TQ3DMASTSn (n = 0 to 5) bit (DMA TX Transfer Status for TXQ 3 of Channel n)

The TQ3DMASTSn bit is set when the CFDCDTTCT.TQ3DMAEn bit in the corresponding CFDCDTTCT register is set (see **7.9.2.2.45 DMA TX Transfer Control Register (CFDCDTTCT)**).

This bit is cleared when:

- The CFDCDTTCT.TQ3DMAEn bit in the corresponding CFDCDTTCT register is cleared
- The CAN-FD module is in GL_RESET mode.

CFDMASTSn (n = 0 to 5) bit (DMA TX Transfer Status only for Common FIFO 2 of Channel n)

The CFDMASTSn bit is set when the CFDCDTTCT.CFDMAEn bit in the corresponding CFDCDTTCT register is set (see **7.9.2.2.45 DMA TX Transfer Control Register (CFDCDTTCT)**).

This bit is cleared when:

- The CFDCDTTCT.CFDMAEn bit in the corresponding CFDCDTTCT register is cleared
- The CAN-FD module is in GL_RESET mode.

7.9.2.2.47 Global RX Interrupt Status Register n (CFDGRINTSTSn) (n = 0 to 5)

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1350h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	CFOFRIF[2:0]			-	CFRFIF[2:0]			-	-	-	-	-	CFRIF[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	BQOFRIF[1:0]		-	QOFRIF[2:0]			-	-	BQFIF[1:0]		-	QFIF[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read.
30 to 28	CFOFRIF[2:0]	0h	R	Common FIFO One Frame RX Interrupt Flag Channel n (n = 0 to 5) 0b: Corresponding Common FIFO One Frame RX Interrupt flag is not set 1b: Corresponding Common FIFO One Frame RX Interrupt flag is set
27	-	0h	R	Reserved Whenever it is read, 0b is read.
26 to 24	CFRFIF[2:0]	0h	R	Common FIFO FDC Level Full Interrupt Flag Channel n (n = 0 to 5) 0b: Corresponding Common FIFO Full Interrupt flag is not set 1b: Corresponding Common FIFO Full Interrupt flag is set
23 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read.
18 to 16	CFRIF[2:0]	0h	R	Common FIFO RX Interrupt Flag Channel n (n = 0 to 5) 0b: Corresponding Common FIFO RX Interrupt flag is not set 1b: Corresponding Common FIFO RX Interrupt flag is set
15 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read.
13 to 12	BQOFRIF[1:0]	0h	R	Borrowed TXQ One Frame RX Interrupt Flag Channel n (n = 0 to 5) 0b: Corresponding TXQ One Frame RX Interrupt flag is not set 1b: Corresponding TXQ One Frame RX Interrupt flag is set
11	-	0h	R	Reserved Whenever it is read, 0b is read.
10 to 8	QOFRIF[2:0]	0h	R	TXQ One Frame RX Interrupt Flag Channel n (n = 0 to 5) 0b: Corresponding TXQ One Frame RX Interrupt flag is not set 1b: Corresponding TXQ One Frame RX Interrupt flag is set
7 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read.
5 to 4	BQFIF[1:0]	0h	R	Borrowed TXQ Full Interrupt Flag Channel n (n = 0 to 5) 0b: Corresponding TXQ Full Interrupt flag is not set 1b: Corresponding TXQ Full Interrupt flag is set
3	-	0h	R	Reserved Whenever it is read, 0b is read.
2 to 0	QFIF[2:0]	0h	R	TXQ Full Interrupt Flag Channel n (n = 0 to 5) 0b: Corresponding TXQ Full Interrupt flag is not set 1b: Corresponding TXQ Full Interrupt flag is set

QFIF[2:0] bits (TXQ Full Interrupt Flag Channel n (n = 0 to 5))

The QFIF[2:0] bits are set automatically when the TXQ Full Interrupt flag of the related channel is set when the interrupt is enabled.

The QFIF[2:0] bits are cleared automatically when:

- The related TXQ result status bits are cleared or the interrupt enable is disabled

- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

BQFIF[1:0] bits (Borrowed TXQ Full Interrupt Flag Channel n (n = 0 to 5))

The BQFIF[1:0] bits are set when a flexible transmission buffer assignment function is used and the borrowed TXQ is in full status. Operation is the same as CFDGRINTSTSn.QFIF.

The bit of the channel that lends TXMB is a reserved bit.

QOFRIF[2:0] bits (TXQ One Frame RX Interrupt Flag Channel n (n = 0 to 5))

The QOFRIF[2:0] bits are set automatically when the TXQ One Frame RX Interrupt flag of the related channel is set when the interrupt is enabled.

The QOFRIF[2:0] bits are cleared automatically when:

- The related TXQ result status bits are cleared or the interrupt enable is disabled
- The CAN-FD module is in GL_RESET or CH_RESET mode.

BQOFRIF[1:0] bits (Borrowed TXQ One Frame RX Interrupt Flag Channel n (n = 0 to 5))

The BQOFRIF[1:0] bits are set when a flexible transmission buffer assignment function is used and the borrowed TXQ receives one frame. Operation is the same as CFDGRINTSTSn.QOFRIF.

The bit of the channel that lends TXMB is a reserved bit.

CFRIF[2:0] bits (Common FIFO RX Interrupt Flag Channel n (n = 0 to 5))

The CFRIF[2:0] bits are set automatically when the Common FIFO RX Interrupt flag of the related channel is set when the interrupt is enabled.

The CFRIF[2:0] bits are cleared automatically when:

- The related Common FIFO RX result status bits are cleared or the interrupt enable is disabled
- The CAN-FD module is in GL_RESET or CH_RESET mode.

CFRFIF[2:0] bits (Common FIFO FDC Level Full Interrupt Flag Channel n (n = 0 to 5))

The CFRFIF[2:0] bits are set automatically when the Common FIFO Full Interrupt flag of the related channel is set when the interrupt is enabled.

The CFRFIF[2:0] bits are cleared automatically when:

- The related Common FIFO RX result status bits are cleared or the interrupt enable is disabled
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

CFOFRIF[2:0] bits (Common FIFO One Frame RX Interrupt Flag Channel n (n = 0 to 5))

The CFOFRIF[2:0] bits are set automatically when the Common FIFO One Frame RX Interrupt flag of the related channel is set when the interrupt is enabled.

The CFOFRIF[2:0] bits are cleared automatically when:

- The related Common FIFO RX result status bits are cleared or the interrupt enable is disabled
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

7.9.2.2.48 TX Message Buffer Control Register n (CFDTMCn) (n = 0 to 383)

The CFDTMCn register (n = 0 to 383) configures the TX message buffer functions.

Remark n = 16 to 31, 48 to 63, 80 to 95, 112 to 127, 144 to 159, 176 to 191, 208 to 223, 240 to 255, 272 to 287, 304 to 319, 336 to 351, 368 to 383 are unused.

Access Size : 8 bits
Address : <CFD_base> + 02D0h + n x 0001h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	TMOM	TMTAR	TMTR
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	TMOM	0h	RW	TX Message Buffer One-shot Mode 0b: TX message buffer not configured in one-shot mode 1b: TX message buffer configured in one-shot mode
1	TMTAR	0h	RW	TX Message Buffer Transmission Abort Request 0b: TX message buffer transmission request abort not requested 1b: TX message buffer transmission request abort requested
0	TMTR	0h	RW	TX Message Buffer Transmission Request 0b: TX Message buffer transmission not requested 1b: TX message buffer transmission requested

TMTR bit (TX Message Buffer Transmission Request)

When the TMTR bit is set, the CAN-FD module logic tries to transmit the message stored in the corresponding message buffer.

Only write to this bit when the related CAN-FD module is in CH_HALT or CH_OPERATION mode.

Do not set this bit if the corresponding TX message buffer is linked to a COM FIFO in TX or GW mode or is a part of TX Queue.

This bit cannot be directly cleared by a CPU write access.

This bit can only be set when the Transmission Result flag bits (CFDTMSTSn.TMTRF[1:0]) in the CFDTMSTSn register corresponding to the message buffer are cleared to 00b.

The TMTR bit is automatically cleared by the:

- CAN-FD module logic at the end of a successful transmission
- CAN-FD module logic at the end of a transmission abort, requested by the corresponding CFDTMCn.TMTAR bit
- CAN-FD module logic when there is a detection of a CAN bus error or arbitration loss if CFDTMCn.TMOM bit is set for the message buffer
- CAN-FD module logic when the CAN-FD module is in GL_RESET mode or the related channel is in CH_RESET mode.

TMTAR bit (TX Message Buffer Transmission Abort Request)

When the TMTAR bit is set, the CAN-FD module logic tries to abort the transmission of the frame stored in the corresponding message buffer.

In most cases, transmission cannot be aborted if the internal scan for transmission is complete and the message buffer has already been selected for transmission. In this case, frame may be transmitted successfully from the message buffer. The message buffer selection is released by entering CH_HALT mode.

However, message buffer selected for transmission can be aborted by an abort request when the CAN node detects a new message on the bus (RX pin) before it starts transmission from the selected message buffer.

Only write to the TMTAR bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode. This bit can only be set when the related transmit request TMTR bit is set.

The TMTAR bit cannot be cleared by a CPU write access. Clearing of this bit by CAN-FD has priority over setting by a CPU write access.

The TMTAR bit is automatically cleared by:

- The CAN-FD module logic at the end of a successful transmission
- The CAN-FD module logic at the end of a transmission abort
- The CAN-FD module logic when there is detection of a CAN bus error or arbitration loss
- The CAN-FD module logic when the CAN-FD module is in GL_RESET mode or the related channel enters CH_RESET mode.

TMOM bit (TX Message Buffer One-shot Mode)

When the TMOM bit is set, the CAN-FD module logic tries to transmit the message only once.

If the transmission is successful, the CFDTMSTSn.TMTRF[1:0] bits are set to 10b or 11b. Otherwise, the transmission is automatically aborted and CFDTMSTSn.TMTRF[1:0] bits are set to 01b due to a bus error or a bus arbitration lost.

The TMOM bit remains set if the transmission has completed successfully or aborted due to an error or a loss of arbitration.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Set this bit at the same time as the TMTR bit. Clear this bit with a write access.

If a message has already been requested for transmission, do not write to this bit until the message has been successfully transmitted or transmission has been aborted.

The TMOM bit is automatically cleared by the CAN-FD module logic when the CAN-FD module is in GL_RESET mode or the related channel is in CH_RESET mode.

7.9.2.2.49 TX Message Buffer Status Register n (CFDTMSTSn) (n = 0 to 383)

The CFDTMSTSn register (n = 0 to 383) shows status of the transmission and transmission abort for the corresponding message buffers.

Remark n = 16 to 31, 48 to 63, 80 to 95,112 to 127, 144 to 159, 176 to 191, 208 to 223, 240 to 255, 272 to 287, 304 to 319, 336 to 351, 368 to 383 are unused.

Access Size : 8 bits
Address : <CFD_base> + 07D0h + n x 0001h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	TMTARM	TMTRM	TMTRF[1:0]		TMTSTS
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	R	R	RW	RW	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4	TMTARM	0h	R	TX Message Buffer Transmission Abort Request Mirrored 0b: TX message buffer transmission request abort not requested 1b: TX message buffer transmission request abort requested
3	TMTRM	0h	R	TX Message Buffer Transmission Request Mirrored 0b: TX message buffer transmission not requested 1b: TX message buffer transmission requested
2, 1	TMTRF[1:0]	0h	RW	TX Message Buffer Transmission Result Flag 00b: No result 01b: Transmission aborted from the TX message buffer 10b: Transmission successful from the TX message buffer and transmission abort was not requested 11b: Transmission successful from the TX message buffer and transmission abort was requested
0	TMTSTS	0h	R	TX Message Buffer Transmission Status 0b: No on-going transmission 1b: On-going transmission

TMTSTS bit (TX Message Buffer Transmission Status)

The TMTSTS bit is set automatically at the start of the transmission from the corresponding TX message buffer.

This bit is cleared automatically when:

- Transmission stops
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

TMTRF[1:0] bits (TX Message Buffer Transmission Result Flag)

The TMTRF[1:0] bits show the result for the corresponding TX message buffer. The status is as follows:

- 00b: Transmission in progress or has not been requested
- 01b: Transmission has been aborted from the corresponding TX message buffer
- 10b: Transmission was successful from the corresponding TX message buffer and the CFDTMCn.TMTAR bit was not set for this TX message buffer
- 11b: Transmission was successful from the corresponding TX message buffer, but the CFDTMCn.TMTAR bit was set for this TX message buffer.

Only write to these bits when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

The TMTRF[1:0] bits are cleared automatically when the CAN-FD module is in GL_RESET mode or the related channel is in CH_RESET mode.

TMTRM bit (TX Message Buffer Transmission Request Mirrored)

The TMTRM bit is set when the CFDTMCn.TMTR bit in the corresponding CFDTMCn register is set.

This bit is cleared when the CFDTMCn.TMTR bit in the corresponding CFDTMCn register is cleared.

TMTARM bit (TX Message Buffer Transmission Abort Request Mirrored)

The TMTARM bit is set when the CFDTMCn.TMTAR bit in the corresponding CFDTMCn register is set.

This bit is cleared when the CFDTMCn.TMTAR bit in the corresponding CFDTMCn register is cleared.

7.9.2.2.50 TX Message Buffer Transmission Request Status Register f (CFDTMTRSTSf) (f = 0 to 11)

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 0CD0h + f x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTRSTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read.
15 to 0	TMTRSTS[15:0]	0h	R	TX Message Buffer Transmission Request Status 0b: Transmission not requested for corresponding TX message buffer 1b: Transmission requested for corresponding TX message buffer

TMTRSTS[15:0] bits (TX Message Buffer Transmission Request Status)

The TMTRSTS[15:0] bits show status of the CFDTMCn.TMTR bits of the TX Message Buffer Control Registers.

Alignment of the TMTRSTS[15:0] bits is shown in **Table 7.9-7**.

Table 7.9-7 Alignment of TMTRSTS[15:0] Mirror Bits

Bit Position	TX Message Buffer Number
n × 64 − fmin × 32	n × 64 + 0
n × 64 + 1 − fmin × 32	n × 64 + 1
⋮	⋮
n × 64 + 15 − fmin × 32	n × 64 + 15
n × 64 + 32 − fmax × 32	n × 64 + 32
⋮	⋮
n ×64 + 46 − fmax × 32	n × 64 + 46
n × 64 + 47 − fmax × 32	n × 64 + 47

Note: When n = 0, fmin = 0, fmax = 1
When n = 1, fmin = 2, fmax = 3
When n = 2, fmin = 4, fmax = 5
When n = 3, fmin = 6, fmax = 7
When n = 4, fmin = 8, fmax = 9
When n = 5, fmin = 10, fmax = 11

Each bit is set automatically when the corresponding bit is set in the TX Message Buffer Control Registers (CFDTMCn), and only when the message buffer does not belong to a TX Queue.

Each bit is cleared automatically when:

- The corresponding bit is cleared in the TX Message Buffer Control Registers
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

7.9.2.2.51 TX Message Buffer Transmission Abort Request Status Register f (CFDTMTARSTSf) (f = 0 to 11)

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 0D70h + f x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTARSTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read.
15 to 0	TMTARSTS [15:0]	0h	R	TX Message Buffer Transmission Abort Request Status 0b: Transmission abort not requested for corresponding TX message buffer 1b: Transmission abort requested for corresponding TX message buffer

TMTARSTS[15:0] bits (TX Message Buffer Transmission Abort Request Status)

The TMTARSTS[15:0] bits show status of the CFDTMCn.TMTAR bits of the TX Message Buffer Control Registers. Alignment of the TMTARSTS[15:0] bits is shown in **Table 7.9-8**.

Table 7.9-8 Alignment of TMTARSTS[15:0] Mirror Bits

Bit Position	TX Message Buffer Number
n × 64 − fmin × 32	n × 64 + 0
n × 64 + 1 − fmin × 32	n × 64 + 1
⋮	⋮
n × 64 + 15 − fmin × 32	n × 64 + 15
n × 64 + 32 − fmax × 32	n × 64 + 32
⋮	⋮
n × 64 + 46 − fmax × 32	n × 64 + 46
n × 64 + 47 − fmax × 32	n × 64 + 47

Note: When n = 0, fmin = 0, fmax = 1
When n = 1, fmin = 2, fmax = 3
When n = 2, fmin = 4, fmax = 5
When n = 3, fmin = 6, fmax = 7
When n = 4, fmin = 8, fmax = 9
When n = 5, fmin = 10, fmax = 11

Each bit is set automatically when the corresponding bit is set in the TX Message Buffer Control Registers, and when the message buffer belongs to a TX Queue.

Each bit is cleared automatically when:

- The corresponding bit is cleared in the TX Message Buffer Control Registers
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

7.9.2.2.52 TX Message Buffer Transmission Completion Status Register f (CFDTMTCSTSf) (f = 0 to 11)

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 0E10h + f x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTCSTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read.
15 to 0	TMTCSTS [15:0]	0h	R	TX Message Buffer Transmission Completion Status 0b: Transmission not completed for corresponding TX message buffer 1b: Transmission completed for corresponding TX message buffer

TMTCSTS[15:0] bits (TX Message Buffer Transmission Completion Status)

The TMTCSTS[15:0] bits show status of successful completion of the TX Message Buffer Status Registers.

Alignment of the TMTCSTS[15:0] bits is shown in **Table 7.9-9**.

Table 7.9-9 Alignment of TMTCSTS[15:0] Mirror Bits

Bit Position	TX Message Buffer Number
n × 64 − fmin × 32	n × 64 + 0
n × 64 + 1 − fmin × 32	n × 64 + 1
⋮	⋮
n × 64 + 15 − fmin × 32	n × 64 + 15
n × 64 + 32 − fmax × 32	n × 64 + 32
⋮	⋮
n × 64 + 46 − fmax × 32	n × 64 + 46
n × 64 + 47 − fmax × 32	n × 64 + 47

Note: When n = 0, fmin = 0, fmax = 1
When n = 1, fmin = 2, fmax = 3
When n = 2, fmin = 4, fmax = 5
When n = 3, fmin = 6, fmax = 7
When n = 4, fmin = 8, fmax = 9
When n = 5, fmin = 10, fmax = 11

Each bit is set automatically when the corresponding bit is set in the TX Message Buffer Status Registers.

Each bit is cleared automatically when:

- The corresponding bit is cleared in the TX Message Buffer Status Registers
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

If a CAN channel enters CH_RESET mode, then the bits related to that channel are cleared.

7.9.2.2.53 TX Message Buffer Transmission Abort Status Register f (CFDTMTASTSf) (f = 0 to 11)

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 0EB0h + f x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTASTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read.
15 to 0	TMTASTS [15:0]	0h	R	TX Message Buffer Transmission Abort Status 0b: Transmission not aborted for corresponding TX message buffer 1b: Transmission aborted for corresponding TX message buffer

TMTASTS[15:0] bits (TX Message Buffer Transmission Abort Status)

The TMTASTS[15:0] bits show status of the successful transmission abort of the corresponding TX message buffer.

Alignment of the TMTASTS[15:0] bits is shown in **Table 7.9-10**.

Table 7.9-10 Alignment of TMTASTS[15:0] Mirror Bits

Bit Position	TX Message Buffer Number
n × 64 − fmin × 32	n × 64 + 0
n × 64 + 1 − fmin × 32	n × 64 + 1
⋮	⋮
n × 64 + 15 − fmin × 32	n × 64 + 15
n × 64 + 32 − fmax × 32	n × 64 + 32
⋮	⋮
n × 64 + 46 − fmax × 32	n × 64 + 46
n × 64 + 47 − fmax × 32	n × 64 + 47

Note: When n = 0, fmin = 0, fmax = 1
When n = 1, fmin = 2, fmax = 3
When n = 2, fmin = 4, fmax = 5
When n = 3, fmin = 6, fmax = 7
When n = 4, fmin = 8, fmax = 9
When n = 5, fmin = 10, fmax = 11

Each bit is set automatically when the CFDTMSTSn.TMTRF[1:0] bits are set to 01b in the corresponding TX Message Buffer Status Register.

Each bit is cleared automatically when:

- The CFDTMSTSn.TMTRF[1:0] bits are cleared in the corresponding TX Message Buffer Status Register
- The CAN-FD module is in GL_RESET mode
- The related CAN-FD channel is in CH_RESET mode.

7.9.2.2.54 TX Message Buffer Transmission Interrupt Enable Register f (CFDTMIECf) (f = 0 to 11)

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 0F50h + f x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMIE[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 0	TMIE[15:0]	0h	RW	TX Message Buffer Interrupt Enable 0b: TX message buffer interrupt disabled for corresponding TX message buffer 1b: TX message buffer interrupt enabled for corresponding TX message buffer

TMIE[15:0] bits (TX Message Buffer Interrupt Enable)

If the TMIE[15:0] bits are set, an interrupt is generated at the end of a successful transmission from the corresponding message buffer.

See **7.9.7 Interrupt and DMA** for TX Message Buffer Interrupt specification.

Alignment of the TMIE[15:0] bits is shown in **Table 7.9-11**.

Table 7.9-11 Alignment of TMIE[15:0] bits

Bit Position	TX Message Buffer Number
n × 64 − fmin × 32	n × 64 + 0
n × 64 + 1 − fmin × 32	n × 64 + 1
⋮	⋮
n × 64 + 15 − fmin × 32	n × 64 + 15
n × 64 + 32 − fmax × 32	n × 64 + 32
⋮	⋮
n × 64 + 46 − fmax × 32	n × 64 + 46
n × 64 + 47 − fmax × 32	n × 64 + 47

Note: When n = 0, fmin = 0, fmax = 1
When n = 1, fmin = 2, fmax = 3
When n = 2, fmin = 4, fmax = 5
When n = 3, fmin = 6, fmax = 7
When n = 4, fmin = 8, fmax = 9
When n = 5, fmin = 10, fmax = 11

Do not write to the TMIE[15:0] bits when:

- The CAN-FD module is in GL_SLEEP mode
- The related CAN-FD channel is in CH_SLEEP mode
- The corresponding TX message buffer is part of a TX Queue
- The corresponding TX message buffer is linked to a Common FIFO with the CFDCFCCn.CFTML[4:0] bits.

7.9.2.2.55 TX Queue Configuration/Control Register 0n (CFDTXQCC0n) (n = 0 to 5)

The CFDTXQCC0n register (n = 0 to 5) is used to configure the TX Queue transmission.

TXQ0 is composed of TXMB0 to TXMB15 (at the maximum) when TXQE is enabled.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1000h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	TXQOFTXIE	TXQOFRXIE	TXQFIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	TXQDC[4:0]					TXQIM	-	TXQTXIE	-	-	TXQOWE	TXQGWE	TXQE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18	TXQOFTXIE	0h	RW	TXQ One Frame Transmission Interrupt Enable 0b: One Frame TX interrupt generation disabled 1b: One Frame TX interrupt generation enabled
17	TXQOFRXIE	0h	RW	TXQ One Frame Reception Interrupt Enable 0b: One Frame RX interrupt generation disabled 1b: One Frame RX interrupt generation enabled
16	TXQFIE	0h	RW	TXQ Full Interrupt Enable 0b: TX Queue full interrupt generation disabled 1b: TX Queue full interrupt generation enabled
15 to 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12 to 8	TXQDC[4:0]	0h	RW	TX Queue Depth Configuration 00h: 0 messages 01h: Reserved 02h: 3 messages 03h: 4 messages ⋮ 0Fh: 16 messages Others: Setting prohibited
7	TXQIM	0h	RW	TX Queue Interrupt Mode 0b: When the last message is successfully transmitted 1b: At every successful transmission
6	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	TXQTXIE	0h	RW	TX Queue TX Interrupt Enable 0b: TX Queue TX interrupt disabled 1b: TX Queue TX interrupt enabled
4, 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	TXQOWE	0h	RW	TX Queue Overwrite Mode Enable 0b: TX Queue OW mode disabled 1b: TX Queue OW mode enabled
1	TXQGWE	0h	RW	TX Queue Gateway Mode Enable 0b: TX Queue GW mode disabled 1b: TX Queue GW mode enabled
0	TXQE	0h	RW	TX Queue Enable 0b: TX Queue disabled 1b: TX Queue enabled

TXQE bit (TX Queue Enable)

The TXQE bit cannot be set if the configured TX Queue depth is 00h (CFDTXQCC0n.TXQDC[4:0] = 00h).

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_RESET or CH_SLEEP mode.

The TXQE bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

TXQGWE bit (TX Queue Gateway Mode Enable)

When the TXQGWE bit is set, the TX Queue is in TX Queue GW mode.

When this bit is set, CPU must not access the TX Queue.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQOWE bit (TX Queue Overwrite Mode Enable)

When the TXQOWE bit is set, the TX Queue is in TX Queue Overwrite mode.

An overwrite function is valid when the same ID such as ID of the data written in from the gateway or CPU is in TX Queue. For example, when a frame is received and is stored into the TX Queue, if a message with the same ID is stored in the TX Queue, the old message is overwritten by the new message.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

When using the function in GW mode, the depth of TXQ (CFDTXQCC0n.TXQDC) should be configured to the value which is the various number of ID that is used in the TX Queue plus 3.

The function is valid for the standard ID frame but is invalid for the extended ID frame.

Do not modify this bit when the CFDTXQCC0n.TXQE bit is 1b.

TXQTXIE bit (TX Queue TX Interrupt Enable)

When the TXQTXIE bit is set, an interrupt is generated based on the setting of the TXQIM bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

TXQIM bit (TX Queue Interrupt Mode)

The TXQIM bit selects the interrupt generation condition for the TX Queue.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP

- CH_HALT
- CH_OPERATION.

TXQDC[4:0] bits (TX Queue Depth Configuration)

The TXQDC[4:0] bits select the depth of the transmission queue. The message buffer selection starts from MB[0] up to MB[15] depending on the configured depth.

When using TXQ1 and TXQ0 simultaneously, the total depth of TXQ1 and TXQ0 should be 16 or less.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQFIE bit (TXQ Full Interrupt Enable)

When the TXQFIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS0n.TXQFIF bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

Only write 1b to this bit in Gateway mode (CFDTXQCC0n.TXQGWE = 1b).

TXQOFRXIE bit (TXQ One Frame Reception Interrupt Enable)

When the TXQOFRXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS0n.TXQOFRXIF bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

Only write 1b to this bit in GW mode (CFDTXQCC0n.TXQGWE = 1b).

TXQOFTXIE bit (TXQ One Frame Transmission Interrupt Enable)

When the TXQOFTXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS0n.TXQOFTXIF bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

7.9.2.2.56 TX Queue Configuration/Control Register 1n (CFDTXQCC1n) (n = 0 to 5)

The CFDTXQCC1n register (n = 0 to 5) is used to configure the TX Queue transmission.

TXQ1 is composed of TXMB15 to TXMB0 (at the maximum) when TXQE is enabled.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1060h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	TXQOFTXIE	TXQOFRXIE	TXQFIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	TXQDC[4:0]					TXQIM	-	TXQTXIE	-	-	TXQOWE	TXQGWE	TXQE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18	TXQOFTXIE	0h	RW	TXQ One Frame Transmission Interrupt Enable 0b: One Frame TX interrupt generation disabled 1b: One Frame TX interrupt generation enabled
17	TXQOFRXIE	0h	RW	TXQ One Frame Reception Interrupt Enable 0b: One Frame RX interrupt generation disabled 1b: One Frame RX interrupt generation enabled
16	TXQFIE	0h	RW	TXQ Full Interrupt Enable 0b: TX Queue full interrupt generation disabled 1b: TX Queue full interrupt generation enabled
15 to 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12 to 8	TXQDC[4:0]	0h	RW	TX Queue Depth Configuration 00h: 0 messages 01h: Reserved 02h: 3 messages 03h: 4 messages ⋮ 0Fh: 16 messages Others: Setting prohibited
7	TXQIM	0h	RW	TX Queue Interrupt Mode 0b: When the last message is successfully transmitted 1b: At every successful transmission
6	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	TXQTXIE	0h	RW	TX Queue TX Interrupt Enable 0b: TX Queue TX interrupt disabled 1b: TX Queue TX interrupt enabled
4, 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	TXQOWE	0h	RW	TX Queue Overwrite Mode Enable 0b: TX Queue OW mode disabled 1b: TX Queue OW mode enabled
1	TXQGWE	0h	RW	TX Queue Gateway Mode Enable 0b: TX Queue GW mode disabled 1b: TX Queue GW mode enabled
0	TXQE	0h	RW	TX Queue Enable 0b: TX Queue disabled 1b: TX Queue enabled

TXQE bit (TX Queue Enable)

The TXQE bit cannot be set if the configured TX Queue depth is 00h (CFDTXQCC1n.TXQDC[4:0] = 00h).

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do write to this bit when the related CAN-FD channel is in CH_RESET or CH_SLEEP mode.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

TXQGWE bit (TX Queue Gateway Mode Enable)

When the TXQGWE bit is set, the TX Queue is in TX Queue Gateway mode.

When this bit is set, CPU must not access the TX Queue.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQOWE bit (TX Queue Overwrite Mode Enable)

When the TXQOWE bit is set, the TX Queue is in TX Queue Overwrite mode.

An overwrite function is valid when the same ID such as ID of the data written in from the gateway or CPU is in TX Queue. For example, when a frame is received and is stored into the TX Queue, if a message with the same ID is stored in the TX Queue, the old message is overwritten by the new message.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

When using the function in GW mode, the depth of TXQ (CFDTXQCC1n.TXQDC) should be configured to the value which is the various number of ID that is used in the TX Queue plus 3.

The function is valid for the standard ID frame but is invalid for the extended ID frame.

Do not modify this bit when the CFDTXQCC1n.TXQE bit is 1.

TXQTXIE bit (TX Queue TX Interrupt Enable)

When the TXQTXIE bit is set, an interrupt is generated based on the setting of the TXQIM bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

TXQIM bit (TX Queue Interrupt Mode)

The TXQIM bit selects the interrupt generation condition for the TX Queue.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP

- CH_HALT
- CH_OPERATION.

TXQDC[4:0] bits (TX Queue Depth Configuration)

The TXQDC[4:0] bits select the depth of the transmission queue. The message buffer selection starts from MB[15] down to MB[0] depending on the configured depth.

When using TXQ1 and TXQ0 simultaneously, the total depth of TXQ1 and TXQ0 should be 16 or less.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQFIE bit (TXQ Full Interrupt Enable)

When the TXQFIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS1n.TXQFIF bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

Only write 1b to this bit when in Gateway mode (CFDTXQCC1n.TXQGWE = 1b).

TXQOFRXIE bit (TXQ One Frame Reception Interrupt Enable)

When the TXQOFRXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS1n.TXQOFRXIF bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

Only write 1b to this bit when in Gateway mode (CFDTXQCC1n.TXQGWE = 1b).

TXQOFTXIE bit (TXQ One Frame Transmission Interrupt Enable)

When the TXQOFTXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS1n.TXQOFTXIF bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

7.9.2.2.57 TX Queue Configuration/Control Register 2n (CFDTXQCC2n) (n = 0 to 5)

The CFDTXQCC2n register (n = 0 to 5) is used to configure the TX Queue transmission.

TXQ2 is composed of TXMB32 to TXMB47 (at the maximum) when TXQE is enabled.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 10C0h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	TXQOFTXIE	TXQOFRXIE	TXQFIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	TXQDC[4:0]					TXQIM	-	TXQTXIE	-	-	TXQOWE	TXQGWE	TXQE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	RW	Description
31 to 19	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18	TXQOFTXIE	0h	RW	TXQ One Frame Transmission Interrupt Enable 0b: One Frame TX interrupt generation disabled 1b: One Frame TX interrupt generation enabled
17	TXQOFRXIE	0h	RW	TXQ One Frame Reception Interrupt Enable 0b: One Frame RX interrupt generation disabled 1b: One Frame RX interrupt generation enabled
16	TXQFIE	0h	RW	TXQ Full Interrupt Enable 0b: TX Queue full interrupt generation disabled 1b: TX Queue full interrupt generation enabled
15 to 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12 to 8	TXQDC[4:0]	0h	RW	TX Queue Depth Configuration 00h: 0 messages 01h: Reserved 02h: 3 messages 03h: 4 messages ⋮ 0Fh: 16 messages Others: Setting prohibited
7	TXQIM	0h	RW	TX Queue Interrupt Mode 0b: When the last message is successfully transmitted 1b: At every successful transmission
6	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	TXQTXIE	0h	RW	TX Queue TX Interrupt Enable 0b: TX Queue TX interrupt disabled 1b: TX Queue TX interrupt enabled
4, 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	TXQOWE	0h	RW	TX Queue Overwrite Mode Enable 0b: TX Queue OW mode disabled 1b: TX Queue OW mode enabled
1	TXQGWE	0h	RW	TX Queue Gateway Mode Enable 0b: TX Queue GW mode disabled 1b: TX Queue GW mode enabled
0	TXQE	0h	RW	TX Queue Enable 0b: TX Queue disabled 1b: TX Queue enabled

TXQE bit (TX Queue Enable)

The TXQE bit cannot be set if the configured TX Queue depth is 00h (CFDTXQCC2n.TXQDC[4:0] = 00h).

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_RESET or CH_SLEEP mode.

TXQGWE bit (TX Queue Gateway Mode Enable)

When the TXQGWE bit is set, the TX Queue is in TX Queue Gateway mode. When this bit is set, CPU must not access the TX Queue.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQOWE bit (TX Queue Overwrite Mode Enable)

When the TXQOWE bit is set, the TX Queue is in TX Queue Overwrite mode.

An overwrite function is valid when the same ID such as ID of the data written in from the gateway or CPU is in TX Queue. For example, when a frame is received and is stored into the TX Queue, if a message with the same ID is stored in the TX Queue, the old message is overwritten by the new message.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

When using the function in GW mode, the depth of TXQ (CFDTXQCC2n.TXQDC) should be configured to the value which is the various number of ID that is used in the TX Queue plus 3.

The function is valid for the standard ID frame but is invalid for the extended ID frame.

Do not modify this bit when the CFDTXQCC2n.TXQE bit is 1b.

TXQTXIE bit (TX Queue TX Interrupt Enable)

When the TXQTXIE bit is set, an interrupt is generated based on the setting of the TXQIM bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

TXQIM bit (TX Queue Interrupt Mode)

The TXQIM bit selects the interrupt generation condition for the TX Queue.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP

- CH_HALT
- CH_OPERATION.

TXQDC[4:0] bits (TX Queue Depth Configuration)

The TXQDC[4:0] bits select the depth of the transmission queue. The message buffer selection starts from MB[32] up to MB[47] depending on the configured depth.

When using TXQ3 and TXQ2 simultaneously, the total depth of TXQ3 and TXQ2 should be 16 or less.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION

TXQFIE bit (TXQ Full Interrupt Enable)

When the TXQFIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS2n.TXQFIF bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

Only write 1b to this bit when in GW mode (CFDTXQCC2n.TXQGWE = 1b).

TXQOFRXIE bit (TXQ One Frame Reception Interrupt Enable)

When the TXQOFRXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS2n.TXQOFRXIF bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

Only write 1b to this bit when in GW mode (CFDTXQCC2n.TXQGWE = 1b).

TXQOFTXIE bit (TXQ One Frame Transmission Interrupt Enable)

When the TXQOFTXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS2n.TXQOFTXIF bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

7.9.2.2.58 TX Queue Configuration/Control Register 3n (CFDTXQCC3n) (n = 0 to 5)

The CFDTXQCC3n register (n = 0 to 5) is used to configure the TX Queue transmission.

TXQ3 is composed of TXMB47 to TXMB32 (at the maximum) when TXQE is enabled.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1120h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	TXQOFTXIE	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	TXQDC[4:0]					TXQIM	-	TXQTXIE	-	-	TXQOWE	-	TXQE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18	TXQOFTXIE	0h	RW	TXQ One Frame Transmission Interrupt Enable 0b: One Frame TX interrupt generation disabled 1b: One Frame TX interrupt generation enabled
17 to 13	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
12 to 8	TXQDC[4:0]	0h	RW	TX Queue Depth Configuration 00h: 0 messages 01h: Reserved 02h: 3 messages 03h: 4 messages ⋮ 0Fh: 16 messages Others: Setting prohibited
7	TXQIM	0h	RW	TX Queue Interrupt Mode 0b: When the last message is successfully transmitted 1b: At every successful transmission
6	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	TXQTXIE	0h	RW	TX Queue TX Interrupt Enable 0b: TX Queue TX interrupt disabled 1b: TX Queue TX interrupt enabled
4, 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	TXQOWE	0h	RW	TX Queue Overwrite Mode Enable 0b: TX Queue OW mode disabled 1b: TX Queue OW mode enabled
1	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	TXQE	0h	RW	TX Queue Enable 0b: TX Queue disabled 1b: TX Queue enabled

TXQE bit (TX Queue Enable)

The TXQE bit cannot be set if the configured TX Queue depth is 00h (CFDTXQCC3n.TXQDC[4:0] = 00h).

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_RESET or CH_SLEEP mode.

TXQOWE bit (TX Queue Overwrite Mode Enable)

When the TXQOWE bit is set, the TX Queue is in TX Queue Overwrite mode.

An overwrite function is valid when the same ID such as ID of the data written in from the gateway or CPU is in TX Queue. For example, when a frame is received and is stored into the TX Queue, if a message with the same ID is stored in the TX Queue, the old message is overwritten by the new message.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

When using the function in GW mode, the depth of TXQ (CFDTXQCC3n.TXQDC) should be configured to the value which is the various number of ID that is used in the TX Queue plus 3.

The function is valid for the standard ID frame but is invalid for the extended ID frame.

Do not modify this bit when the CFDTXQCC3n.TXQE bit is 1b.

TXQTXIE bit (TX Queue TX Interrupt Enable)

When the TXQTXIE bit is set, an interrupt is generated based on the setting of the TXQIM bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

TXQIM bit (TX Queue Interrupt Mode)

The TXQIM bit selects the interrupt generation condition for the TX Queue.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQDC[4:0] bits (TX Queue Depth Configuration)

The TXQDC[4:0] bits select the depth of the transmission queue. The message buffer selection starts from MB[47] down to MB[32] depending on the configured depth.

When using TXQ3 and TXQ2 simultaneously, the total depth of TXQ3 and TXQ2 should be 16 or less.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in any of the following modes:

- CH_SLEEP
- CH_HALT
- CH_OPERATION.

TXQOFTXIE bit (TXQ One Frame Transmission Interrupt Enable)

When the TXQOFTXIE bit is set, an interrupt is generated based on the setting of the CFDTXQSTS3n.TXQOFTXIF bit.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the related CAN-FD channel is in CH_SLEEP mode.

7.9.2.2.59 TX Queue Status Register 0n (CFDTXQSTS0n) (n = 0 to 5)

The CFDTXQSTS0n register (n = 0 to 5) shows the status of the TX Queue of corresponding CAN channel.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1020h + n x 0004h

Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	TXQMOW	TXQMLT	TXQOFTXIF	TXQOFRXIF	TXQFIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	TXQMC[5:0]						-	-	-	-	-	TXQTXIF	TXQFLL	TXQEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	RW	RW	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20	TXQMOW	0h	RW	TXQ Message Overwrite 0b: No message overwrite in TXQ 1b: Message overwrite in TXQ
19	TXQMLT	0h	RW	TXQ Message Lost 0b: No message lost in TXQ 1b: TXQ message lost
18	TXQOFTXIF	0h	RW	TXQ One Frame Transmission Interrupt Flag When one frame is transmitted from TXQ, an interrupt is set.
17	TXQOFRXIF	0h	RW	TXQ One Frame Reception Interrupt Flag When TXQ receives one frame, an interrupt is set.
16	TXQFIF	0h	RW	TXQ Full Interrupt Flag When TXQ is in full status, an interrupt is set.
15, 14	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
13 to 8	TXQMC[5:0]	0h	R	TX Queue Message Count Number of messages in the TX Queue.
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	TXQTXIF	0h	RW	TX Queue TX Interrupt Flag 0b: TX Queue interrupt condition not satisfied after a frame TX 1b: TX Queue interrupt condition satisfied after a frame TX
1	TXQFLL	0h	R	TX Queue Full 0b: TX Queue not full 1b: TX Queue full
0	TXQEMP	1h	R	TX Queue Empty 0b: TX Queue not empty 1b: TX Queue empty

TXQEMP bit (TX Queue Empty)

The TXQEMP bit is set automatically when the TX Queue is disabled or no messages are stored in the TX Queue.

This bit is set automatically when:

- The last message is transmitted from the TX Queue
- The related CAN-FD channel is in CH_RESET mode.

The bit is cleared automatically when the first message to be transmitted is stored in the TX Queue.

TXQFLL bit (TX Queue Full)

The TXQFLL bit is set automatically when the number of CAN messages stored in the TX Queue matches the configured TX Queue depth.

This bit is cleared automatically when:

- The number of CAN messages stored in the TX Queue is less than the configured TX Queue depth
- The related CAN-FD channel is in CH_RESET mode.

TXQTXIF bit (TX Queue TX Interrupt Flag)

The TXQTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

This bit is set automatically when the configured interrupt condition is satisfied for the TX Queue.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMC[5:0] bits (TX Queue Message Count)

The TXQMC[5:0] bits show the number of CAN messages in the TX Queue.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

TXQFIF bit (TXQ Full Interrupt Flag)

The TXQFIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of the TX Queue.

Only in Gateway mode (CFDTXQCC0n.TXQGWE = 1b) that this bit is set automatically when the TX Queue transits to a buffer full status.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQOFRXIF bit (TXQ One Frame Reception Interrupt Flag)

The TXQOFRXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When receiving data is stored in the TX Queue in Gateway mode, this bit is set automatically.

This function can only be used in Gateway mode of the TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQOFTXIF bit (TXQ One Frame Transmission Interrupt Flag)

The TXQOFTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When transmission is successful in the TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMLT bit (TXQ Message Lost)

The TXQMLT bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When a message lost occurs in Gateway mode of the TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMOW bit (TXQ Message Overwrite)

The TXQMOW bit is not cleared automatically if the TX Queue is disabled.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When CFDTXQCC0n.TXQOWE = 1b and message overwrite occurs in TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

7.9.2.2.60 TX Queue Status Register 1n (CFDTXQSTS1n) (n = 0 to 5)

The CFDTXQSTS1n register (n = 0 to 5) shows the status of the TX Queue of corresponding CAN channel.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1080h + n x 0004h

Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	TXQMOW	TXQMLT	TXQOFTXIF	TXQOFRXIF	TXQFIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	TXQMC[5:0]						-	-	-	-	-	TXQTXIF	TXQFLL	TXQEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	RW	RW	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20	TXQMOW	0h	RW	TXQ Message Overwrite 0b: No message overwrite in TXQ 1b: Message overwrite in TXQ
19	TXQMLT	0h	RW	TXQ Message Lost 0b: No message lost in TXQ 1b: TXQ message lost
18	TXQOFTXIF	0h	RW	TXQ One Frame Transmission Interrupt Flag When one frame is transmitted from TXQ, an interrupt is set.
17	TXQOFRXIF	0h	RW	TXQ One Frame Reception Interrupt Flag When TXQ receives one frame, an interrupt is set.
16	TXQFIF	0h	RW	TXQ Full Interrupt Flag When TXQ is in full status, an interrupt is set.
15, 14	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
13 to 8	TXQMC[5:0]	0h	R	TX Queue Message Count Number of messages in the TX Queue
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	TXQTXIF	0h	RW	TX Queue TX Interrupt Flag 0b: TX Queue interrupt condition not satisfied after a frame TX 1b: TX Queue interrupt condition satisfied after a frame TX
1	TXQFLL	0h	R	TX Queue Full 0b: TX Queue not full 1b: TX Queue full
0	TXQEMP	1h	R	TX Queue Empty 0b: TX Queue not empty 1b: TX Queue empty

TXQEMP bit (TX Queue Empty)

The TXQEMP bit is set automatically when the TX Queue is disabled or no messages are stored in the TX Queue.

The bit is cleared automatically when the first message to be transmitted is stored in the TX Queue.

This bit is set automatically when:

- The last message is transmitted from the TX Queue
- The related CAN-FD channel is in CH_RESET mode.

TXQFLL bit (TX Queue Full)

The TXQFLL bit is set automatically when the number of CAN messages stored in the TX Queue matches the configured TX Queue depth.

This bit is cleared automatically when:

- The number of CAN messages stored in the TX Queue is less than the configured TX Queue depth
- The related CAN-FD channel is in CH_RESET mode.

TXQTXIF bit (TX Queue TX Interrupt Flag)

The TXQTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

This bit is set automatically when the configured interrupt condition is satisfied for the TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMC[5:0] bits (TX Queue Message Count)

The TXQMC[5:0] bits show the number of CAN messages in the TX Queue.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

TXQFIF bit (TXQ Full Interrupt Flag)

The TXQFIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

Only in Gateway mode (CFDTXQCC1n.TXQGWE = 1b) that this bit is set automatically when TX Queue transits to a buffer full status.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQOFRXIF bit (TXQ One Frame Reception Interrupt Flag)

The TXQOFRXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When receiving data is stored in the TX Queue in Gateway mode, this bit is set automatically.

This function can only be used in Gateway mode of the TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQOFTXIF bit (TXQ One Frame Transmission Interrupt Flag)

The TXQOFTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When transmission is successful in the TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMLT bit (TXQ Message Lost)

The TXQMLT bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When a message lost occurs in Gateway mode of the TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMOW bit (TXQ Message Overwrite)

The TXQMOW bit is not cleared automatically if the TX Queue is disabled.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When CFDTXQCC1n.TXQOWE = 1b and message overwrite occurs in TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

7.9.2.2.61 TX Queue Status Register 2n (CFDTXQSTS2n) (n = 0to 5)

The CFDTXQSTS2n register (n = 0 to 5) shows the status of the TX Queue of corresponding CAN Channel.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 10E0h + n x 0004h

Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	TXQMOW	TXQMLT	TXQOFTXIF	TXQOFRXIF	TXQFIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	TXQMC[5:0]						-	-	-	-	-	TXQTXIF	TXQFLL	TXQEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	RW	RW	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20	TXQMOW	0h	RW	TXQ Message Overwrite 0b: No message overwrite in TXQ 1b: Message overwrite in TXQ
19	TXQMLT	0h	RW	TXQ Message Lost 0b: No message lost in TXQ 1b: TXQ message lost
18	TXQOFTXIF	0h	RW	TXQ One Frame Transmission Interrupt Flag When one frame is transmitted from TXQ, an interrupt is set.
17	TXQOFRXIF	0h	RW	TXQ One Frame Reception Interrupt Flag When TXQ receives one frame, an interrupt is set.
16	TXQFIF	0h	RW	TXQ Full Interrupt Flag When TXQ is in full status, an interrupt is set.
15, 14	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
13 to 8	TXQMC[5:0]	0h	R	TX Queue Message Count Number of messages in the TX Queue.
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	TXQTXIF	0h	RW	TX Queue TX Interrupt Flag 0b: TX Queue interrupt condition not satisfied after a frame TX 1b: TX Queue interrupt condition satisfied after a frame TX
1	TXQFLL	0h	R	TX Queue Full 0b: TX Queue not full 1b: TX Queue full
0	TXQEMP	1h	R	TX Queue Empty 0b: TX Queue not empty 1b: TX Queue empty

TXQEMP bit (TX Queue Empty)

The TXQEMP bit is set automatically when the TX Queue is disabled or no messages are stored in the TX Queue.

This bit is cleared automatically when the first message to be transmitted is stored in the TX Queue.

This bit is set automatically when:

- The last message is transmitted from the TX Queue
- The related CAN-FD channel is in CH_RESET mode.

TXQFLL bit (TX Queue Full)

The TXQFLL bit is set automatically when the number of CAN messages stored in the TX Queue matches the configured TX Queue depth.

This bit is cleared automatically when:

- The number of CAN messages stored in the TX Queue is less than the configured TX Queue depth
- The related CAN-FD channel is in CH_RESET mode.

TXQTXIF bit (TX Queue TX Interrupt Flag)

The TXQTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of the TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

This bit is set automatically when the configured interrupt condition is satisfied for the TX Queue.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMC[5:0] bits (TX Queue Message Count)

The TXQMC[5:0] bits show the number of CAN messages in the TX Queue.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

TXQFIF bit (TXQ Full Interrupt Flag)

The TXQFIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of the TX Queue.

Only in Gateway mode (CFDTXQCC2n.TXQGWE = 1b) that this bit is set automatically when the TX Queue transits to a buffer full status.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQOFRXIF bit (TXQ One Frame Reception Interrupt Flag)

The TXQOFRXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When receiving data is stored in the TX Queue in Gateway mode, this bit is set automatically.

This function can only be used in Gateway mode of the TX queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQOFTXIF bit (TXQ One Frame Transmission Interrupt Flag)

The TXQOFTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When transmission is successful in the TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

The bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMLT bit (TXQ Message Lost)

The TXQMLT bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When a message lost occurs in Gateway mode of the TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMOW bit (TXQ Message Overwrite)

The TXQMOW bit is not cleared automatically if the TX Queue is disabled.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When CFDTXQCC2n.TXQOWE = 1b and message overwrite occurs in TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

7.9.2.2.62 TX Queue Status Register 3n (CFDTXQSTS3n) (n = 0 to 5)

The CFDTXQSTS3n register (n = 0 to 5) shows the status of the TX Queue of corresponding CAN Channel.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1140h + n x 0004h

Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	TXQMOW	-	TXQOFTXIF	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	TXQMC[5:0]						-	-	-	-	-	TXQTXIF	TXQFLL	TXQEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	RW	RW	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
20	TXQMOW	0h	RW	TXQ Message Overwrite 0b: No message overwrite in TXQ 1b: Message overwrite in TXQ
19	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
18	TXQOFTXIF	0h	RW	TXQ One Frame Transmission Interrupt Flag When one frame is transmitted from TXQ, an interrupt is set.
17 to 14	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
13 to 8	TXQMC[5:0]	0h	R	TX Queue Message Count Number of messages in the TX Queue
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	TXQTXIF	0h	RW	TX Queue TX Interrupt Flag 0b: TX Queue interrupt condition not satisfied after a frame TX 1b: TX Queue interrupt condition satisfied after a frame TX
1	TXQFLL	0h	R	TX Queue Full 0b: TX Queue not full 1b: TX Queue full
0	TXQEMP	1h	R	TX Queue Empty 0b: TX Queue not empty 1b: TX Queue empty

TXQEMP bit (TX Queue Empty)

The TXQEMP bit is set automatically when the TX Queue is disabled or no messages are stored in the TX Queue.

This bit is set automatically when:

- The last message is transmitted from the TX Queue
- The related CAN-FD channel is in CH_RESET mode.

This bit is cleared automatically when the first message to be transmitted is stored in the TX Queue.

TXQFLL bit (TX Queue Full)

The TXQFLL bit is set automatically when the number of CAN messages stored in the TX Queue matches the configured TX Queue depth.

This bit is cleared automatically when:

- The number of CAN messages stored in the TX Queue is less than the configured TX Queue depth
- The related CAN-FD channel is in CH_RESET mode.

TXQTXIF bit (TX Queue TX Interrupt Flag)

The TXQTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of the TX Queue.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is set automatically when the configured interrupt condition is satisfied for the TX Queue.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMC[5:0] bits (TX Queue Message Count)

The TXQMC[5:0] bits show the number of CAN messages in the TX Queue.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

TXQOFTXIF bit (TXQ One Frame Transmission Interrupt Flag)

The TXQOFTXIF bit is not cleared automatically if the TX Queue is disabled.

When stopping the TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue. When transmission is successful in the TX Queue, this bit is set automatically.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

TXQMOW bit (TXQ Message Overwrite)

The TXQMOW bit is not cleared automatically if the TX Queue is disabled.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When CFDTXQCC3n.TXQOWE = 1b and message overwrite occurs in TX Queue, this bit is set automatically.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b. Writing 1b has no effect.

You cannot write to this bit when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

7.9.2.2.63 TX Queue Pointer Control Register 0n (CFDTXQPCTR0n) (n = 0 to 5)

The CFDTXQPCTR0n register (n = 0 to 5) is used to confirm storage of a full message in the corresponding TX Queue buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1040h + n x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TXQPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	W	Reserved The written value should always be 0b.
7 to 0	TXQPC[7:0]	0h	W	TX Queue Pointer Control Increments the write pointer to the TX Queue buffer in the corresponding channel

TXQPC[7:0] bits (TX Queue Pointer Control)

When the value FFh is written to the TXQPC[7:0] bits, the write pointer of the corresponding TX Queue buffer is updated and a transmit request is initiated for this message.

The read value from these bits is always 00h. Do not write to the FIFO control registers when DMA is enabled.

You cannot write to these bits when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Only write FFh to this register when:

- The corresponding TX Queue is enabled and not full
- The Common FIFO is enabled and is not configured in GW mode.

7.9.2.2.64 TX Queue Pointer Control Register 1n (CFDTXQPCTR1n) (n = 0 to 5)

The CFDTXQPCTR1n register (n = 0 to 5) is used to confirm storage of a full message in the corresponding TX Queue buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 10A0h + n x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TXQPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	W	Reserved The written value should always be 0b.
7 to 0	TXQPC[7:0]	0h	W	TX Queue Pointer Control Increments the write pointer to the TX Queue buffer in the corresponding channel

TXQPC[7:0] bits (TX Queue Pointer Control)

When the value FFh is written to the TXQPC[7:0] bits, the write pointer of the corresponding TX Queue buffer is updated and a transmit request is initiated for this message.

The read value from these bits is always 00h. You cannot write to the FIFO control registers when DMA is enabled.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Only write FFh to this register when:

- The corresponding TX Queue is enabled and not full
- The Common FIFO is enabled and is not configured in GW mode.

7.9.2.2.65 TX Queue Pointer Control Register 2n (CFDTXQPCTR2n) (n = 0 to 5)

The CFDTXQPCTR2n register (n = 0 to 5) is used to confirm storage of a full message in the corresponding TX Queue buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1100h + n x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TXQPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	W	Reserved The written value should always be 0b.
7 to 0	TXQPC[7:0]	0h	W	TX Queue Pointer Control Increments the write pointer to the TX Queue buffer in the corresponding channel

TXQPC[7:0] bits (TX Queue Pointer Control)

When the value FFh is written to the TXQPC[7:0] bits, the write pointer of the corresponding TX Queue buffer is updated and a transmit request is initiated for this message.

The read value from these bits is always 00h. You cannot write to the FIFO control registers when DMA is enabled.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Only write FFh to this register when:

- The corresponding TX Queue is enabled and not full
- The Common FIFO is enabled and is not configured in GW mode.

7.9.2.2.66 TX Queue Pointer Control Register 3n (CFDTXQPCTR3n) (n = 0 to 5)

The CFDTXQPCTR3n register (n = 0 to 5) is used to confirm storage of a full message in the corresponding TX Queue buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1160h + n x 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TXQPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	W	Reserved The written value should always be 0b.
7 to 0	TXQPC[7:0]	0h	W	TX Queue Pointer Control Increments the write pointer to the TX Queue buffer in the corresponding channel

TXQPC[7:0] bits (TX Queue Pointer Control)

When the value FFh is written to the TXQPC[7:0] bits, the write pointer of the corresponding TX Queue buffer is updated and a transmit request is initiated for this message.

The read value from these bits is always 00h. Do not write to the FIFO control registers when DMA is enabled.

You cannot write to these bits when the related CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Only write FFh to this register when:

- The corresponding TX Queue is enabled and not full
- The Common FIFO is enabled and is not configured in Gateway mode.

7.9.2.2.67 TX Queue Empty Status Register (CFDTXQESTS)

The CFDTXQESTS register shows the status of the empty bits of the TXQ buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1180h
Initial Value : 0000_00FFh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	TXQxEMP[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXQxEMP[15:0]															
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read.
23 to 0	TXQxEMP[23:0]	FFh	R	TXQ Empty Status 0b: TXQ not empty 1b: TXQ empty

TXQxEMP[23:0] bits (TXQ Empty Status)

Each bit is set automatically when the corresponding bit is set in the TX Queue Empty Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue Empty Status Register.

This bit is set when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Channel 0 TX Queue 3
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Channel 1 TX Queue 3
8	Channel 2 TX Queue 0
9	Channel 2 TX Queue 1
10	Channel 2 TX Queue 2
11	Channel 2 TX Queue 3
12	Channel 3 TX Queue 0
13	Channel 3 TX Queue 1
14	Channel 3 TX Queue 2
15	Channel 3 TX Queue 3
16	Channel 4 TX Queue 0
17	Channel 4 TX Queue 1
18	Channel 4 TX Queue 2

Bit Position	Corresponding TX Queue
19	Channel 4 TX Queue 3
20	Channel 5 TX Queue 0
21	Channel 5 TX Queue 1
22	Channel 5 TX Queue 2
23	Channel 5 TX Queue 3

7.9.2.2.68 TX Queue Full Interrupt Status Register (CFDTXQFISTS)

The CFDTXQFISTS register shows the status of the full interrupt bits of the TXQ buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1184h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	TXQ5FULL[2:0]			-	TXQ4FULL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	TXQ3FULL[2:0]			-	TXQ2FULL[2:0]			-	TXQ1FULL[2:0]			-	TXQ0FULL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	-	All 0	R	Reserved Whenever it is read, 0b is read.
22 to 20	TXQ5FULL[2:0]	0h	R	TXQ Full Interrupt Status Flag for Channel 5 0b: TXQ full interrupt not set 1b: TXQ full interrupt set
19	-	0h	R	Reserved Whenever it is read, 0b is read.
18 to 16	TXQ4FULL[2:0]	0h	R	TXQ Full Interrupt Status Flag for Channel 4 0b: TXQ full interrupt not set 1b: TXQ full interrupt set
15	-	0h	R	Reserved Whenever it is read, 0b is read.
14 to 12	TXQ3FULL[2:0]	0h	R	TXQ Full Interrupt Status Flag for Channel 3 0b: TXQ full interrupt not set 1b: TXQ full interrupt set
11	-	0h	R	Reserved Whenever it is read, 0b is read.
10 to 8	TXQ2FULL[2:0]	0h	R	TXQ Full Interrupt Status Flag for Channel 2 0b: TXQ full interrupt not set 1b: TXQ full interrupt set
7	-	0h	R	Reserved Whenever it is read, 0b is read.
6 to 4	TXQ1FULL[2:0]	0h	R	TXQ Full Interrupt Status Flag for Channel 1 0b: TXQ full interrupt not set 1b: TXQ full interrupt set
3	-	0h	R	Reserved Whenever it is read, 0b is read.
2 to 0	TXQ0FULL[2:0]	0h	R	TXQ Full Interrupt Status Flag for Channel 0 0b: TXQ full interrupt not set 1b: TXQ full interrupt set

TXQnFULL[2:0] (n = 0 to 5) bits (TXQ Full Interrupt Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue Full Interrupt Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue Full Interrupt Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Reserved
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Reserved
8	Channel 2 TX Queue 0
9	Channel 2 TX Queue 1
10	Channel 2 TX Queue 2
11	Reserved
12	Channel 3 TX Queue 0
13	Channel 3 TX Queue 1
14	Channel 3 TX Queue 2
15	Reserved
16	Channel 4 TX Queue 0
17	Channel 4 TX Queue 1
18	Channel 4 TX Queue 2
19	Reserved
20	Channel 5 TX Queue 0
21	Channel 5 TX Queue 1
22	Channel 5 TX Queue 2
23	Reserved

7.9.2.2.69 TX Queue Message Lost Status Register (CFDTXQMSTS)

The CFDTXQMSTS register shows the status of the message lost bits of the TXQ buffers.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1188h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	TXQ5ML[2:0]			-	TXQ4ML[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	TXQ3ML[2:0]			-	TXQ2ML[2:0]			-	TXQ1ML[2:0]			-	TXQ0ML[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	-	All 0	R	Reserved Whenever it is read, 0b is read.
22 to 20	TXQ5ML[2:0]	0h	R	TXQ Message Lost Status Flag for Channel 5 0b: TXQ message lost flag not set 1b: TXQ message lost flag set
19	-	0h	R	Reserved Whenever it is read, 0b is read.
18 to 16	TXQ4ML[2:0]	0h	R	TXQ Message Lost Status Flag for Channel 4 0b: TXQ message lost flag not set 1b: TXQ message lost flag set
15	-	0h	R	Reserved Whenever it is read, 0b is read.
14 to 12	TXQ3ML[2:0]	0h	R	TXQ Message Lost Status Flag for Channel 3 0b: TXQ message lost flag not set 1b: TXQ message lost flag set
11	-	0h	R	Reserved Whenever it is read, 0b is read.
10 to 8	TXQ2ML[2:0]	0h	R	TXQ Message Lost Status Flag for Channel 2 0b: TXQ message lost flag not set 1b: TXQ message lost flag set
7	-	0h	R	Reserved Whenever it is read, 0b is read.
6 to 4	TXQ1ML[2:0]	0h	R	TXQ Message Lost Status Flag for Channel 1 0b: TXQ message lost flag not set 1b: TXQ message lost flag set
3	-	0h	R	Reserved Whenever it is read, 0b is read.
2 to 0	TXQ0ML[2:0]	0h	R	TXQ Message Lost Status Flag for Channel 0 0b: TXQ message lost flag not set 1b: TXQ message lost flag set

TXQnML[2:0] (n = 0 to 5) bits (TXQ Message Lost Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue Message Lost Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue Message Lost Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Reserved
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Reserved
8	Channel 2 TX Queue 0
9	Channel 2 TX Queue 1
10	Channel 2 TX Queue 2
11	Reserved
12	Channel 3 TX Queue 0
13	Channel 3 TX Queue 1
14	Channel 3 TX Queue 2
15	Reserved
16	Channel 4 TX Queue 0
17	Channel 4 TX Queue 1
18	Channel 4 TX Queue 2
19	Reserved
20	Channel 5 TX Queue 0
21	Channel 5 TX Queue 1
22	Channel 5 TX Queue 2
23	Reserved

7.9.2.2.70 TX Queue Interrupt Status Register (CFDTXQISTS)

The CFDTXQISTS register shows the status of the interrupt flag of the TXQ buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1190h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	TXQ5ISF[3:0]				TXQ4ISF[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXQ3ISF[3:0]				TXQ2ISF[3:0]				TXQ1ISF[3:0]				TXQ0ISF[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read.
23 to 0	TXQnISF[3:0]	0h	R	TXQ Interrupt Status Flag for Channel n (n = 0 to 5) 0b: TXQ Interrupt flag not set 1b: TXQ Interrupt flag set

TXQnISF[3:0] (n = 0 to 5) bits (TXQ Interrupt Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue Interrupt Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue Interrupt Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Channel 0 TX Queue 3
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Channel 1 TX Queue 3
8	Channel 2 TX Queue 0
9	Channel 2 TX Queue 1
10	Channel 2 TX Queue 2
11	Channel 2 TX Queue 3
12	Channel 3 TX Queue 0
13	Channel 3 TX Queue 1
14	Channel 3 TX Queue 2
15	Channel 3 TX Queue 3
16	Channel 4 TX Queue 0
17	Channel 4 TX Queue 1
18	Channel 4 TX Queue 2

Bit Position	Corresponding TX Queue
19	Channel 4 TX Queue 3
20	Channel 5 TX Queue 0
21	Channel 5 TX Queue 1
22	Channel 5 TX Queue 2
23	Channel 5 TX Queue 3

7.9.2.2.71 TX Queue One Frame TX Interrupt Status Register (CFDTXQOFTISTS)

The CFDTXQOFTISTS register shows the status of the One Frame TX Interrupt flag of the TXQ buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1194h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	TXQ5OFTISF[3:0]				TXQ4OFTISF[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXQ3OFTISF[3:0]				TXQ2OFTISF[3:0]				TXQ1OFTISF[3:0]				TXQ0OFTISF[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read.
23 to 0	TXQnOFTISF[3:0]	0h	R	TXQ One Frame TX Interrupt Status Flag for Channel n (n = 0 to 5) 0b: TXQ One Frame TX Interrupt flag not set 1b: TXQ One Frame TX Interrupt flag set

TXQnOFTISF[3:0] (n = 0 to 5) bits (TXQ One Frame TX Interrupt Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue One Frame TX Interrupt Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue One Frame TX Interrupt Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Channel 0 TX Queue 3
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Channel 1 TX Queue 3
8	Channel 2 TX Queue 0
9	Channel 2 TX Queue 1
10	Channel 2 TX Queue 2
11	Channel 2 TX Queue 3
12	Channel 3 TX Queue 0
13	Channel 3 TX Queue 1
14	Channel 3 TX Queue 2
15	Channel 3 TX Queue 3
16	Channel 4 TX Queue 0

Bit Position	Corresponding TX Queue
17	Channel 4 TX Queue 1
18	Channel 4 TX Queue 2
19	Channel 4 TX Queue 3
20	Channel 5 TX Queue 0
21	Channel 5 TX Queue 1
22	Channel 5 TX Queue 2
23	Channel 5 TX Queue 3

7.9.2.2.72 TX Queue One Frame RX Interrupt Status Register (CFDTXQOFRISTS)

The CFDTXQOFRISTS register shows the status of the One Frame RX Interrupt flag of the TXQ buffers.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1198h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	TXQ5OFRISF[2:0]			-	TXQ4OFRISF[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	TXQ3OFRISF[2:0]			-	TXQ2OFRISF[2:0]			-	TXQ1OFRISF[2:0]			-	TXQ0OFRISF[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	-	All 0	R	Reserved Whenever it is read, 0b is read.
22 to 20	TXQ5OFRISF [2:0]	0h	R	TXQ One Frame RX Interrupt Status Flag for Channel 5 0b: TXQ One Frame RX Interrupt flag not set 1b: TXQ One Frame RX Interrupt flag set
19	-	0h	R	Reserved Whenever it is read, 0b is read.
18 to 16	TXQ4OFRISF [2:0]	0h	R	TXQ One Frame RX Interrupt Status Flag for Channel 4 0b: TXQ One Frame RX Interrupt flag not set 1b: TXQ One Frame RX Interrupt flag set
15	-	0h	R	Reserved Whenever it is read, 0b is read.
14 to 12	TXQ3OFRISF [2:0]	0h	R	TXQ One Frame RX Interrupt Status Flag for Channel 3 0b: TXQ One Frame RX Interrupt flag not set 1b: TXQ One Frame RX Interrupt flag set
11	-	0h	R	Reserved Whenever it is read, 0b is read.
10 to 8	TXQ2OFRISF [2:0]	0h	R	TXQ One Frame RX Interrupt Status Flag for Channel 2 0b: TXQ One Frame RX Interrupt flag not set 1b: TXQ One Frame RX Interrupt flag set
7	-	0h	R	Reserved Whenever it is read, 0b is read.
6 to 4	TXQ1OFRISF [2:0]	0h	R	TXQ One Frame RX Interrupt Status Flag for Channel 1 0b: TXQ One Frame RX Interrupt flag not set 1b: TXQ One Frame RX Interrupt flag set
3	-	0h	R	Reserved Whenever it is read, 0b is read.
2 to 0	TXQ0OFRISF [2:0]	0h	R	TXQ One Frame RX Interrupt Status Flag for Channel 0 0b: TXQ One Frame RX Interrupt flag not set 1b: TXQ One Frame RX Interrupt flag set

TXQnOFRISF[2:0] (n = 0 to 5) bits (TXQ One Frame RX Interrupt Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue One Frame RX Interrupt Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue One Frame RX Interrupt Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Reserved
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Reserved
8	Channel 2 TX Queue 0
9	Channel 2 TX Queue 1
10	Channel 2 TX Queue 2
11	Reserved
12	Channel 3 TX Queue 0
13	Channel 3 TX Queue 1
14	Channel 3 TX Queue 2
15	Reserved
16	Channel 4 TX Queue 0
17	Channel 4 TX Queue 1
18	Channel 4 TX Queue 2
19	Reserved
20	Channel 5 TX Queue 0
21	Channel 5 TX Queue 1
22	Channel 5 TX Queue 2
23	Reserved

7.9.2.2.73 TX Queue Full Status Register (CFDTXQFSTS)

The CFDTXQFSTS register shows the status of the Full Status flag bits of the TXQ buffers.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 119Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	TXQ5FSF[3:0]				TXQ4FSF[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXQ3FSF[3:0]				TXQ2FSF[3:0]				TXQ1FSF[3:0]				TXQ0FSF[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0 is read.
23 to 0	TXQnFSF[3:0]	0h	R	TXQ Full Status Flag for Channel n (n = 0 to 5) 0b: TXQ Full flag not set 1b: TXQ Full flag set

TXQnFSF[3:0] (n = 0 to 5) bits (TXQ Full Status Flag)

Each bit is set automatically when the corresponding bit is set in the TX Queue Full Status Register.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Queue Full Status Register.

This bit is cleared when the CAN-FD module is in GL_RESET mode.

Bit Position	Corresponding TX Queue
0	Channel 0 TX Queue 0
1	Channel 0 TX Queue 1
2	Channel 0 TX Queue 2
3	Channel 0 TX Queue 3
4	Channel 1 TX Queue 0
5	Channel 1 TX Queue 1
6	Channel 1 TX Queue 2
7	Channel 1 TX Queue 3
8	Channel 2 TX Queue 0
9	Channel 2 TX Queue 1
10	Channel 2 TX Queue 2
11	Channel 2 TX Queue 3
12	Channel 3 TX Queue 0
13	Channel 3 TX Queue 1
14	Channel 3 TX Queue 2
15	Channel 3 TX Queue 3
16	Channel 4 TX Queue 0
17	Channel 4 TX Queue 1
18	Channel 4 TX Queue 2

Bit Position	Corresponding TX Queue
19	Channel 4 TX Queue 3
20	Channel 5 TX Queue 0
21	Channel 5 TX Queue 1
22	Channel 5 TX Queue 2
23	Channel 5 TX Queue 3

7.9.2.2.74 TX History List Configuration/Control Register n (CFDTHLCCn) (n = 0 to 5)

The CFDTHLCCn register (n = 0 to 5) configures the TX History List functions.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1200h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	THLDGE	THLDTE	THLIM	THLIE	-	-	-	-	-	-	-	THLE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
11	THLDGE	0h	RW	TX History List Dedicated Gateway Enable 0b: Not dedicated Gateway FIFO + Gateway TX Queue 1b: Dedicated Gateway FIFO + Gateway TX Queue
10	THLDTE	0h	RW	TX History List Dedicated TX Enable 0b: TX FIFO + TX Queue 1b: Flat TX MB + TX FIFO + TX Queue
9	THLIM	0h	RW	TX History List Interrupt Mode 0b: Interrupt generated if TX History List level reaches ¾ of the TX History List depth 1b: Interrupt generated for every successfully stored entry
8	THLIE	0h	RW	TX History List Interrupt Enable 0b: TX History List Interrupt disabled 1b: TX History List Interrupt enabled
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	THLE	0h	RW	TX History List Enable 0b: TX History List disabled 1b: TX History List enabled

THLE bit (TX History List Enable)

The THLE bit enables the TX History List buffer when it is set.

You cannot write to this bit when the related CAN-FD channel is in CH_RESET or CH_SLEEP mode.

This bit is cleared automatically when the related CAN-FD channel is in CH_RESET mode.

THLIE bit (TX History List Interrupt Enable)

The THLIE bit enables the generation of the TX History List interrupt when it is set.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

THLIM bit (TX History List Interrupt Mode)

The THLIM bit selects the interrupt generation condition for the FIFO.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode.

THLDTE bit (TX History List Dedicated TX Enable)

The THLDTE bit selects the condition for storing an entry in the TX History List after successful transmission.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode.

THLDGE bit (TX History List Dedicated Gateway Enable)

The THLDGE bit selects the condition for storing an entry in the TX History List after successful transmission.

You cannot write to this bit when the CAN-FD module is in GL_SLEEP mode.

Do not write to this bit when the CAN-FD module is in GL_HALT or GL_OPERATION mode.

7.9.2.2.75 TX History List Status Register n (CFDTHLSTSn) (n = 0 to 5)

The CFDTHLSTSn register (n = 0 to 5) shows the status of data stored in the TX History List buffer.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1220h + n x 0004h

Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	THLMC[5:0]						-	-	-	-	THLIF	THLELT	THLFLL	THLEMP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	RW	RW	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
13 to 8	THLMC[5:0]	0h	R	TX History List Message Count Number of messages stored in TX History List
7 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3	THLIF	0h	RW	TX History List Interrupt Flag 0b: TX History List interrupt condition not satisfied 1b: TX History List interrupt condition satisfied
2	THLELT	0h	RW	TX History List Entry Lost 0b: No entry lost in TX History List 1b: TX History List entry Lost
1	THLFLL	0h	R	TX History List Full 0b: TX History List not full 1b: TX History List full
0	THLEMP	1h	R	TX History List Empty 0b: TX History List not empty 1b: TX History List empty

THLEMP bit (TX History List Empty)

The THLEMP bit is set automatically when the CPU has read all the entries from the TX History List buffer.

This bit is cleared automatically when the first entry is stored to the TX History List.

This bit is set automatically when:

- TX History List is disabled
- The related CAN-FD channel is in CH_RESET mode.

THLFLL bit (TX History List Full)

The THLFLL bit is set automatically when the number of entries in the TX History List buffer matches the TX History List depth.

Each TX History List can store up to 32 entries (each channel has a dedicated TX History List).

This bit is cleared automatically when:

- The number of entries in the TX History List buffer is less than the TX History List depth

- The TX History List is disabled
- The related CAN-FD channel is in CH_RESET mode.

THLELT bit (TX History List Entry Lost)

The THLELT bit is set when a new entry cannot be stored because the related TX History List buffer is already full.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

THLIF bit (TX History List Interrupt Flag)

The THLIF bit is set when the configured interrupt condition is satisfied.

Only write to this bit when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode. Writing 1b has no effect.

Do not use the bit clear instruction to clear this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Other bits remain 1b.

This bit is cleared:

- By writing 0b to it
- When the related CAN-FD channel is in CH_RESET mode.

This bit is automatically cleared in CH_RESET mode.

THLMC[5:0] bits (TX History List Message Count)

The THLMC[5:0] bits show the number of transmitted messages stored in the TX History List.

These bits are cleared automatically when the related CAN-FD channel is in CH_RESET mode.

7.9.2.2.76 Channel n TX History List Access Register 0 (CFDTHLACC0n) (n = 0 to 5)

The CFDTHLACC0n register (n = 0 to 5) provides access to the entry in the TX History List based on the read timestamp value.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 8000h + n x 0008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TGW	-	-	-	-	-	BN[6:0]							BT[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMTS[15:0]	0h	R	Transmit Timestamp Transmit timestamp value for software drivers
15	TGW	0h	R	Transmit Gateway Buffer Indication 0b: No transmission from gateway 1b: Transmission from gateway
14 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read.
9 to 3	BN[6:0]	0h	R	Buffer Number Number of the message buffer
2 to 0	BT[2:0]	0h	R	Buffer Type 001b: Flat TX message buffer 010b: TX FIFO message buffer number and gateway FIFO message number 100b: TX Queue message buffer number Others: Setting prohibited

BT[2:0] bits (Buffer Type)

The BT[2:0] bits indicate whether data has been stored following a transmission from a FIFO buffer, a TX Queue or a TX message buffer.

BN[6:0] bits (Buffer Number)

The BN[6:0] bits show the message buffer from which transmission was successfully completed. If a message from a Common FIFO is transmitted, then these bits show the message buffer that is linked to the Common FIFO for transmission.

TGW bit (Transmit Gateway Buffer Indication)

The TGW bit is automatically set to 1b when transmission is completed in GW mode.

TMTS[15:0] bits (Transmit Timestamp)

The TMTS[15:0] bits indicate the timestamp for use by software drivers.

7.9.2.2.77 Channel n TX History List Access Register 1 (CFDTHLACC1n) (n = 0 to 5)

The CFDTHLACC1n register (n = 0 to 5) provides access to entry in the TX History List based on the read pointer value.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 8004h + n x 0008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TIFL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read.
17, 16	TIFL[1:0]	0h	R	Transmit Information Label These bits indicate that message buffer information label, TX FIFO information label, or AFL information label is stored for software drivers.
15 to 0	TID[15:0]	0h	R	Transmit ID These bits indicate that message buffer reference ID, TX FIFO reference ID, or AFL pointer field is stored for software drivers.

TID[15:0] bits (Transmit ID)

The TID[15:0] bits indicate whether the message buffer reference ID (CFDTMFDCTRn.TMPTR) or the TX FIFO reference ID (CFDCFFDCSTSn.CFPTR) is for use by software drivers.

When transmission in Gateway mode, these bits indicate the AFL pointer field (CFDGAFLP0n.GAFLPTR) instead of the message buffer reference ID (CFDTMFDCTRn.TMPTR).

TIFL[1:0] bits (Transmit Information Label)

The TIFL[1:0] bits indicate whether the message buffer information label (CFDTMFDCTRn.TMIFL) or the TX FIFO information label (CFDCFFDCSTSn.CFIFL) is for use by software drivers.

When transmission in Gateway mode, these bits indicate the AFL pointer field (CFDGAFLMn.GAFLIFL1 and CFDGAFLP0n.GAFLIFL0) instead of the MB information label (CFDTMFDCTRn.TMIFL).

7.9.2.2.78 TX History List Pointer Control Register n (CFDTHLPCTRn) (n = 0 to 5)

The CFDTHLPCTRn register (n = 0 to 5) is used to increment the read pointer of the TX History List.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1240h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	THLPC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	W	Reserved The written value should always be 0b.
7 to 0	THLPC[7:0]	0h	W	TX History List Pointer Control Increments the write pointer to the TX History List in the corresponding channel

THLPC[7:0] bits (TX History List Pointer Control)

When FFh is written to the THLPC[7:0] bits, the read pointer of the TX History List is moved to the next TX History List entry address.

The read value from these bits is always 00h. Only write to these bits when the related CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Only write FFh to these registers when the corresponding TX History List is enabled and not empty.

7.9.2.2.79 Global Reset Control Register (CFDGRSTC)

Access Size : 16, 32 bits
Address : <CFD_base> + 1380h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	KEY[7:0]								-	-	-	-	-	-	-	SRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 8	KEY[7:0]	All 0	W	Key Code These bits control the validity of rewriting of the SRST bit.
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	SRST	0h	RW	Software Reset 0b: Normal state 1b: Software reset state

SRST bit (Software Reset)

When the SRST bit is set, the CAN-FD module is in the same state as hardware reset. When a reset is required, write 1b then write 0b to this bit.

This bit is cleared when the CAN-FD module is in GL_SLEEP mode.

When this bit is cleared, the RAM initialization sequence does not operate. The configuration of RAM is performed by software.

The RAM is not initialized when software reset is performed during the initialization of RAM. Software must perform the initialization of RAM.

KEY[7:0] bits (Key Code)

When C4h is written in the KEY[7:0] bits, a write to the SRST bit is valid.

The read value from these bits is always 00h.

7.9.2.2.80 Global Flexible CAN Mode Configuration Register (CFDGFCMC)

Flexible CAN mode configured in the CFDGFCMC register and Flexible transmission buffer assignment configured in the CFDGFTBAC register should not be used simultaneously.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1384h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	FLXC2	FLXC1	FLXC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	FLXC2	0h	RW	Flexible CAN mode between Channel 4 and Channel 5 0b: Normal mode 1b: Flexible CAN mode
1	FLXC1	0h	RW	Flexible CAN mode between Channel 2 and Channel 3 0b: Normal mode 1b: Flexible CAN mode
0	FLXC0	0h	RW	Flexible CAN mode between Channel 0 and Channel 1 0b: Normal mode 1b: Flexible CAN mode

FLXC0 bit (Flexible CAN Mode between Channel 0 and Channel 1)

When the FLXC0 bit is set, Channel 0 and Channel 1 of a CAN-FD module are in Flexible CAN mode.

Channel 1 uses TX/RX terminal of Channel 0. The TX/RX terminal of Channel 1 cannot be used.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

FLXC1 bit (Flexible CAN Mode between Channel 2 and Channel 3)

When the FLXC1 bit is set, Channel 2 and Channel 3 of a CAN-FD module are in Flexible CAN mode.

Channel 3 uses TX/RX terminal of Channel 2. The TX/RX terminal of Channel 3 cannot be used.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

FLXC2 bit (Flexible CAN Mode between Channel 4 and Channel 5)

When the FLXC2 bit is set, Channel 4 and Channel 5 of a CAN-FD module are in Flexible CAN mode.

Channel 5 uses TX/RX terminal of Channel 4. The TX/RX terminal of Channel 5 cannot be used.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

7.9.2.2.81 Global Flexible Transmission Buffer Assignment Configuration Register (CFDGFTBAC)

Flexible transmission buffer assignment configured in the CFDGFTBAC register and Flexible CAN mode configured in the CFDGFCMC register should not be used simultaneously.

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 138Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	FLXMB3[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	FLXMB1[3:0]				-	-	-	-	FLXMB0[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
19 to 16	FLXMB3[3:0]	0h	RW	Flexible transmission buffer assignment between Channel 4 and Channel 5 By setting these bits the even channel can use the configured number of TX mailboxes of the odd channel 0000b: 0 0001b: 4 0010b: 8 0011b: 12 0100b: 16 0101b: 20 0110b: 24 0111b: 28 1000b: 32
15 to 12	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
11 to 8	FLXMB1[3:0]	0h	RW	Flexible transmission buffer assignment between Channel 2 and Channel 3 By setting these bits the even channel can use the configured number of TX mailboxes of the odd channel 0000b: 0 0001b: 4 0010b: 8 0011b: 12 0100b: 16 0101b: 20 0110b: 24 0111b: 28 1000b: 32
7 to 4	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
3 to 0	FLXMB0[3:0]	0h	RW	Flexible transmission buffer assignment between Channel 0 and Channel 1 By setting these bits the even channel can use the configured number of TX mailboxes of the odd channel 0000b: 0 0001b: 4 0010b: 8 0011b: 12 0100b: 16 0101b: 20 0110b: 24 0111b: 28 1000b: 32

FLXMB0[3:0] bits (Flexible Transmission Buffer Assignment between Channel 0 and Channel 1)

Channel 0 can use the number TXMB of channel 1 from 0 to 32 by the configuration of these bits.

Only write to this bit when the CAN-FD module is in GL_RESET mode.

7.9.2.2.82 Global Test Configuration Register (CFDGTSTCFG)

The CFDGTSTCFG register is used to configure the CAN channels joining the internal CAN bus communication test mode and the RAM test mode page.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1308h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	RTMPS[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	C5ICBCE	C4ICBCE	C3ICBCE	C2ICBCE	C1ICBCE	C0ICBCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
25 to 16	RTMPS[9:0]	0h	RW	RAM Test Mode Page Select Select a RAM test mode page
15 to 6	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5 to 0	CnICBCE	0h	RW	Channel n Internal CAN Bus Communication Test Mode Enable (n = 0 to 5) 0b: Channel n internal CAN bus communication disabled 1b: Channel n internal CAN bus communication enabled

CnICBCE (n = 0 to 5) bits (Channel n Internal CAN Bus Communication Test Mode Enable)

When the CnICBCE bits are set and CAN-FD module is configured in the internal CAN bus communication test mode, then CAN channel n joins the internal CAN bus communication test mode operation.

Do not write to these bits when the CAN-FD module is in GL_RESET or GL_SLEEP mode.

Only write to these bits when the CAN-FD module is in GL_HALT mode.

These bits are cleared automatically when the CAN-FD module is in GL_RESET mode.

RTMPS[9:0] bits (RAM Test Mode Page Select)

The RTMPS[9:0] bits select the RAM page mode for CPU read/write access when the CAN-FD module is configured in RAM test mode.

See **7.9.9.2.1 RAM Test Mode** for the RAM test mode specification.

Do not write to these bits when the CAN-FD module is in GL_RESET or GL_SLEEP mode.

Only enter values from 0 to 0 to 47 (02Fh) for the AFL RAM and 48 to 228 (0E4h) for the message buffer RAM.

The setting range of these bits depends on the combination of parameters.

Only write to these bits when the CAN-FD module is in GL_HALT mode.

These bits are cleared automatically when the related CAN-FD channel is in GL_RESET mode.

7.9.2.2.83 Global Test Control Register (CFDGTSTCTR)

The CFDGTSTCTR register is used to control the global test modes of the CAN-FD module.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 130Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	RTME	-	ICBCTME
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	RTME	0h	RW	RAM Test Mode Enable 0b: RAM test mode disabled 1b: RAM test mode enabled
1	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	ICBCTME	0h	RW	Internal CAN Bus Communication Test Mode Enable 0b: Internal CAN Bus Communication test mode disabled 1b: Internal CAN Bus Communication test mode enabled

ICBCTME bit (Internal CAN Bus Communication Test Mode Enable)

When the ICBCTME bit is set, internal CAN bus communication is enabled for the CAN channels that are configured for internal CAN bus communication participation. See **7.9.9.2.2 Internal CAN Bus Communication Mode** for the specification of internal CAN bus communication test mode.

Only write to this bit when the CAN-FD module is in GL_HALT mode.

Clear this bit when the CAN-FD module is in GL_HALT mode.

This bit is cleared automatically when the CAN-FD module is in GL_RESET mode.

RTME bit (RAM Test Mode Enable)

When the RTME bit is set, the CAN-FD module is configured in RAM test mode. See **7.9.9.2.1 RAM Test Mode** for RAM test mode specification.

Only write to this bit when the CAN-FD module is in GL_HALT mode.

Clear this bit when the CAN-FD module is in GL_HALT mode.

This bit is cleared automatically when the CAN-FD module is in GL_RESET mode.

7.9.2.2.84 Global FD Configuration Register (CFDGFDCFG)

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1314h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	TSCCFG[1:0]		-	-	-	-	-	-	-	RPED
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9, 8	TSCCFG[1:0]	0h	RW	Timestamp Capture Configuration 00b: Timestamp capture at the sample point of SOF (start of frame) 01b: Timestamp capture at frame valid indication 10b: Timestamp capture at the sample point of RES bit 11b: Reserved
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	RPED	0h	RW	RES Bit Protocol Exception Disable 0b: Protocol exception event detection enabled 1b: Protocol exception event detection disabled

RPED bit (RES Bit Protocol Exception Disable)

The RPED bit configures the protocol exception event handling according to ISO 11898-1.

When this bit is enabled, the protocol exception event detection is disabled, and the protocol controller transmits an error frame when the protocol exception event is detected (RES bit is sampled recessive).

Only write to this bit when the CAN-FD module is in GL_RESET mode.

TSCCFG[1:0] bits (Timestamp Capture Configuration)

The TSCCFG[1:0] bits configure the different capture points of the timestamp for transmission and reception.

When CFDGFDCFG.TSCCFG[1:0] = 10b, the timestamp capture is performed for CAN-FD frames at RES bit and for Classical frames at the start of frame.

Only write to these bits when the CAN-FD module is in GL_RESET mode.

7.9.2.2.85 Global Lock Key Register (CFDGLOCKK)

The CFDGLOCKK register is a write-only register that is used to unlock the protection for special test bits. See **7.9.9.2 Global Test Modes** for Lock key specification.

Access Size : 16, 32 bits
Address : <CFD_base> + 131Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCK[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	W	Reserved The written value should always be 0b.
15 to 0	LOCK[15:0]	0h	W	Lock Key Key bits for unlocking the protection of test modes

LOCK[15:0] bits (Lock Key)

The unlock key sequence must be written in the LOCK[15:0] bits to configure the CAN-FD module in RAM test modes.

The read value from these bits is always 0000h.

You cannot write to these bits when the CAN-FD module is in GL_SLEEP or GL_RESET mode.

Do not write to these bits when the CAN-FD module is in GL_OPERATION mode.

7.9.2.2.86 RAM Test Page Access Register n (CFDRPGACCn) (n = 0 to 63)

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 8400h + n x 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDTA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDTA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDTA[31:0]	0h	RW	RAM Data Test Access RAM data bytes

RDTA[31:0] bits (RAM Data Test Access)

Data can be read from or written into the RDTA[31:0] bits when the CAN-FD module is configured in RAM test mode.

Only write to this bit when the CAN-FD module is in GL_HALT mode and RAM test mode is enabled.

If data is read when RAM test mode is not enabled, then it is always read as 0000_0000h.

Software data should be read/written in the RAM Test Page Access registers during RAM test mode.

7.9.2.2.87 Channel n Bus Load Control Register (CFDCnBLCT) (n = 0 to 5)

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 1418h + n x 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	BLCLD	-	-	-	-	-	-	-	BLCE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
8	BLCLD	0h	W	Bus Load Counter Load When CFDCnBLCT.BLCLD is set, it is reset after a bus load counter value is loaded.
7 to 1	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	BLCE	0h	RW	Bus Load Counter Enable 0b: Bus load counter disabled 1b: Bus load counter enabled

BLCE bit (Bus Load Counter Enable)

The BLCE bit indicates the enabling signal of a bus load counter. Write to this bit after setting up the CFDCnNCFG register.

When this bit is 0b, the bus load counter stops but the counter is not clear.

The bit is cleared when the related CAN-FD channel is in CH_RESET mode.

Do not write to this bit in CH_SLEEP mode.

BLCLD bit (Bus Load Counter Load)

When the BLCLD bit is set, the bus load counter value is loaded into CFDCnBLSTS.BLC and the bus load counter is reset.

The read value is always 0b.

7.9.2.2.88 Channel n Bus Load Status Register (CFDCnBLSTS) (n = 0 to 5)

Access Size : 8, 16, 32 bits

Address : <CFD_base> + 141Ch + n x 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BLC[28:13]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BLC[12:0]													-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	BLC[28:0]	0h	R	Bus Load Counter Status These bits indicate the bus load counter value.
2 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read.

BLC[28:0] bits (Bus Load Counter Status)

When the CFDCnBLCT.BLCLD bit is set, the bus load counter value is loaded to BLC[28:0] bits and the bus load counter is reset.

These bits are set only by the CAN-FD module. Writing any value has no effect.

7.9.2.2.89 Identifier Bits Alignment

Standard Identifier (11 bits) format: ID28 – ID18 is aligned to b10 to b0, b11 to b28 should be 0b.

Extended Identifier (29 bits) format: ID28 – ID0 is aligned to b28 – b0

Table 7.9-12 Standard Identifier (11-Bit Format)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
IDE = 0b	RTR	—	0	0	0	0	0	0	0	0	0	0	0	0	0

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0	0	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21	ID20	ID19	ID18

Table 7.9-13 Extended Identifier (29-Bit Format)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
IDE = 1b	RTR	—	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21	ID20	ID19	ID18	ID17	ID16

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0

7.9.2.3 Message Buffer Component Structure

7.9.2.3.1 Start Addresses

The start address for each of the Message Buffer component is calculated using the number of related Message Buffer components and the number of channels.

- number_of_channels: 6
- number_of_RMBCPs_per_channel: 16
- number_of_RFMBCPs: 8
- number_of_CFMBCPs_per_channel: 3
- number_of_TMBCPs_per_channel: 64

The start addresses for each register in the Message Buffer component are depicted in **Table 7.9-14**.

Table 7.9-14 Message Buffer Component Register Start Addresses

b = Message Buffer Component Index	MBCP	Register	p	Start Address (n = 0 to 5)
0 to number_of_RMBCPs_per_channel - 1	RMBCPb[i]	RMID	x	2000h + b × 0080h + n × 800h
		RMPTR	x	2004h + b × 0080h + n × 800h
		RMFDSTS	x	2008h + b × 0080h + n × 800h
		RMDFp	0 to 15	200Ch + p × 0004h + b × 0080h + n × 800h
0 to number_of_RFMBCPs - 1	RFMBCPb[i]	RFID	x	6000h + b × 0080h
		RFPTR	x	6004h + b × 0080h
		RFFDSTS	x	6008h + b × 0080h
		RFDFp	0 to 15	600Ch + p × 0004h + b × 0080h
0 to number_of_CFMBCPs_per_channel - 1	CFMBCPb[i]	CFID	x	6400h + b × 0080h + n × 180h
		CFPTR	x	6404h + b × 0080h + n × 180h
		CFFDCSTS	x	6408h + b × 0080h + n × 180h
		CFDFp	0 to 15	640Ch + p × 0004h + b × 0080h + n × 180h
0 to number_of_TMBCPs_per_channel - 1	TMBCPb[i]	TMID	x	10000h + b × 0080h + n × 2000h
		TMPTR	x	10004h + b × 0080h + n × 2000h
		TMFDCTR	x	10008h + b × 0080h + n × 2000h
		TMDFp	0 to 15	1000Ch + p × 0004h + b × 0080h + n × 2000h

The message buffer configuration consists of four types of Message Buffer components:

- RX Message Buffer Component (CFDRMBCPb[i])
- RX FIFO Access Message Buffer Component (CFDRFMBCPb[i])
- Common FIFO Access Message Buffer Component (CFDCFMBCPb[i])
- TX Message Buffer Component (CFDTMBCPb[i]).

Where b = the Message Buffer component index having a range that varies based on the type of Message Buffer component and i = channel index that has a range from 0 to n.

For a summary of this configuration, see **Figure 7.9-28**. For a detailed description of the number of and the different types of message buffers, see **7.9.6 FIFO Buffers and Normal Message Buffer Configuration**.

As described in **7.9.2 CANFD Registers**, each Message Buffer component consists of the following registers:

- Identifier (ID)
- Pointer (PTR)
- Data Field (DFp).

p is the Data Field register index having a range that varies based on the type of Message Buffer component.

Rc is the Message Buffer Component register where c = Message Buffer Component register index having a range that varies based on the type of Message Buffer component.

A description of the registers, their associated bits and their accessibility are shown below the summary and detailed figures of each component.

In each of the figures, a cell that contains '-' means reserved and has the same behavior as reserved bits for registers in **7.9.2.3 Message Buffer Component Structure**.

(1) RX Message Buffer ID Register n (CFDRMIDn) (n = 0 to 95)

The CFDRMIDn register (n = 0 to 95) stores the ID field, IDE bit, and RTR bit of the received message.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 2000h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMIDE	RMRTR	-	RMID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RMIDE	0h	R	RX Message Buffer IDE Bit 0b: STD-ID is stored 1b: EXT-ID is stored
30	RMRTR	0h	R	RX Message Buffer RTR Bit 0b: Data frame 1b: Remote frame
29	-	0h	R	Reserved Whenever it is read, 0b is read.
28 to 0	RMID[28:0]	0h	R	RX Message Buffer ID Field STD-ID/EXT-ID fields

RMID[28:0] bits (RX Message Buffer ID Field)

The RMID[28:0] are the bits of the STD-ID/EXT-ID fields of the message stored in the RX message buffer.

For alignment of these bits in standard and extended frame format, see **7.9.2.2.89 Identifier Bits Alignment**.

See **7.9.2.3.1 Start Addresses** for details on how to interpret the structure of this buffer component.

RMRTR bit (RX Message Buffer RTR Bit)

The RMRTR bit shows whether a data frame or a remote frame was stored in the RX message buffer.

Note: There are no remote frames in CAN-FD format. When a CAN-FD frame is received, the register reflects the state of the received value (the RRS bit in FD frame format).

RMIDE bit (RX Message Buffer IDE Bit)

The RMIDE bit shows whether message with Standard Identifier or Extended Identifier was stored in the RX message buffer.

(2) RX Message Buffer Pointer Register (CFDRMPTRn) (n = 0 to 95)

The CFDRMPTRn register (n = 0 to 95) stores the DLC and Timestamp fields for the received message.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 2004h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDLC[3:0]				-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	RMDLC[3:0]	0h	RW	RX Message Buffer DLC Field Number of data bytes received in a CAN frame.
27 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 0	RMTS[15:0]	0h	RW	RX Message Buffer Timestamp Field Timestamp value stored for the message in the RX message buffer

RMTS[15:0] bits (RX Message Buffer Timestamp Field)

The RMTS[15:0] bits store the timestamp value taken at the capture point as configured by CFDGFDCFG.TSCCFG of the received message.

RMDLC[3:0] bits (RX Message Buffer DLC Field)

The RMDLC[3:0] bits store the number of data bytes that were received in the RX message buffer.

See Table 5 in ISO 11898-1 (2015) Specification for details in defining the number of data bytes that were received.

Note: The maximum capacity of the buffer belongs to CFDRMNB.RMPLS and this is not available in the classical CAN function.

(3) RX Message Buffer CAN-FD Status Register (CFDRMFDSTSn) (n = 0 to 95)

The CFDRMFDSTSn register (n = 0 to 95) shows the status of the FDF, BRS, and ESI bits, and pointer of the received CAN-FD frame.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 2008h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMPTR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	RMIFL[1:0]		-	-	-	-	-	RMFDF	RMBRS	RMESI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	RMPTR[15:0]	0h	RW	RX Message Buffer Pointer Field
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9, 8	RMIFL[1:0]	0h	RW	RX Message Buffer Information Label Field
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	RMFDF*[1]	0h	RW	CAN FD Format bit 0b: No CAN-FD frame received 1b: CAN-FD frame received
1	RMBRS*[1]	0h	RW	Bit Rate Switch bit 0b: CAN-FD frame received with no bit rate switch 1b: CAN-FD frame received with bit rate switch
0	RMESI*[1]	0h	RW	Error State Indicator bit 0b: CAN-FD frame received from error active node 1b: CAN-FD frame received from error passive node

Note 1. This bit is not available in the classical CAN function.

RMESI bit (Error State Indicator bit)

The RMESI bit has the same value as the ESI bit of the received CAN-FD frame.

When the received FDF bit is 0b, this means a CAN2.0 frame is received and 0b is stored to this bit.

Note: This bit is not available in the classical CAN function.

RMBRS bit (Bit Rate Switch bit)

The RMBRS bit has the same value as the BRS bit of the received CAN-FD frame.

When the received FDF bit is 0b, this means a CAN2.0 frame is received and 0b is stored to this bit.

Note: This bit is not available in the classical CAN function.

RMFDF bit (CAN FD Format bit)

The RMFDF bit has the same value as the FDF bit of the received CAN-FD frame.

Note: This bit is not available in the classical CAN function.

RMIFL[1:0] bits (RX Message Buffer Information Label Field)

The RMIFL[1:0] bits store the information label value from the related Global Acceptance Filter List entry.

RMPTR[15:0] bits (RX Message Buffer Pointer Field)

The RMPTR[15:0] bits store the pointer value from the related Global Acceptance Filter List entry.

(4) RX Message Buffer Data Field p Register n (CFDRMDFp_n) (p = 0 to 15, n = 0 to 95)

The CFDRMDFp_n register (p = 0 to 15, n = 0 to 95) stores the data bytes (4 × p) to data bytes (4 × p + 3) of the received message.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 200Ch + p x 0004h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB_HH[7:0]								RMDB_HL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB_LH[7:0]								RMDB_LL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RMDB_HH*[1] [7:0]	0h	RW	RX Message Buffer Data Byte (4 × p + 3)
23 to 16	RMDB_HL[7:0]	0h	RW	RX Message Buffer Data Byte (4 × p + 2)
15 to 8	RMDB_LH[7:0]	0h	RW	RX Message Buffer Data Byte (4 × p + 1)
7 to 0	RMDB_LL[7:0]	0h	RW	RX Message Buffer Data Byte (4 × p)

Note 1. These bits are not available in the classical CAN function.

RMDB_LL[7:0] bits (RX Message Buffer Data Byte (4 × p))

The RMDB_LL[7:0] bits store data bytes (4 × p) of the message in the RX message buffer. Unused data bytes are filled with 00h.

RMDB_LH[7:0] bits (RX Message Buffer Data Byte (4 × p + 1))

The RMDB_LH[7:0] bits store data bytes (4 × p + 1) of the message in the RX message buffer. Unused Data Bytes are filled with 00h.

RMDB_HL[7:0] bits (RX Message Buffer Data Byte (4 × p + 2))

The RMDB_HL[7:0] bits store data bytes (4 × p + 2) of the message in the RX message buffer. Unused data bytes are filled with 00h.

RMDB_HH[7:0] bits (RX Message Buffer Data Byte (4 × p + 3))

The RMDB_HH[7:0] bits store data bytes (4 × p + 3) of the message in the RX message buffer. Unused data bytes are filled with 00h.

These bits are not available in the classical CAN function.

(5) RX FIFO Access ID Register n (CFDRFIDn) (n = 0 to 7)

The CFDRFIDn register (n = 0 to 7) stores the ID field, IDE bit, and RTR bit of the message.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 6000h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFIDE	RFRTR	-	RFID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RFIDE	0h	R	RX FIFO Buffer IDE bit 0b: STD-ID has been received 1b: EXT-ID has been received
30	RFRTR	0h	R	RX FIFO Buffer RTR bit 0b: Data frame 1b: Remote frame
29	-	0h	R	Reserved Whenever it is read, 0b is read.
28 to 0	RFID[28:0]	0h	R	RX FIFO Buffer ID Field STD-ID/EXT-ID fields

RFID[28:0] bits (RX FIFO Buffer ID Field)

The RFID[28:0] bits are the bits of the STD-ID/EXT-ID fields of the message in the FIFO buffer.

For alignment of these bits in standard and extended frame format, see **7.9.2.2.89 Identifier Bits Alignment**.

RFRTR bit (RX FIFO Buffer RTR bit)

The RFRTR bit shows whether a data frame or a remote frame was stored in the FIFO buffer.

Note: There are no remote frames in CAN-FD format. When a CAN-FD frame was received, the register reflects the state of the received value (RRS bit in FD frame format).

RFIDE bit (RX FIFO Buffer IDE bit)

The RFIDE bit shows whether message with the Standard Identifier or Extended Identifier was received in the FIFO buffer.

(6) RX FIFO Access Pointer Register n (CFDRFPTRn) (n = 0 to 7)

The CFDRFPTRn register (n = 0 to 7) stores the DLC and Timestamp fields for the received message.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 6004h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDLC[3:0]				-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	RFDLC[3:0]	0h	R	RX FIFO Buffer DLC Field Number of data bytes received in a CAN frame
27 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read.
15 to 0	RFTS[15:0]	0h	R	RX FIFO Timestamp Value Timestamp value of the received CAN frame

RFTS[15:0] bits (RX FIFO Timestamp Value)

The RFTS[15:0] bits store the timestamp value taken at the capture point as configured by the CFDGFDCFG.TSCCFG bit of the received message.

RFDLC[3:0] bits (RX FIFO Buffer DLC Field)

The RFDLC[3:0] bits store the number of data bytes that were received in the RX FIFO buffer.

See Table 5 in ISO 11898-1 (2015) Specification for details in defining the number of data bytes that were received.

(7) RX FIFO Access CAN-FD Status Register n (CFDRFFDSTSn) (n = 0 to 7)

The CFDRFFDSTSn register (n = 0 to 7) shows the status of the FDF, BRS, and ESI bits, including the pointer of the received CAN-FD frame.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 6008h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDRFPTR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	RFIFL[1:0]		-	-	-	-	-	RFFDF	RFBRS	RFESI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	CFDRFPTR [15:0]	0h	R	RX FIFO Buffer Pointer Field
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read.
9, 8	RFIFL[1:0]	0h	R	RX FIFO Buffer Information Label Field
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read.
2	RFFDF*1	0h	R	CAN FD Format bit 0b: No CAN-FD frame received 1b: CAN-FD frame received
1	RFBRS*1	0h	R	Bit Rate Switch bit 0b: CAN-FD frame received with no bit rate switch 1b: CAN-FD frame received with bit rate switch
0	RFESI*1	0h	R	Error State Indicator bit 0b: CAN-FD frame received from error active node 1b: CAN-FD frame received from error passive node

Note 1. This bit is not available in the classical CAN function.

RFESI bit (Error State Indicator bit)

The RFESI bit has the same value as the ESI bit of the received CAN-FD frame.

When the received FDF bit is 0b, this means a CAN2.0 frame is received and 0b is stored to this bit.

Note: This bit is not available in the classical CAN function.

RFBRS bit (Bit Rate Switch bit)

The RFBRS bit has the same value as the BRS bit of the received CAN-FD frame.

When the received FDF bit is 0b, this means a CAN2.0 frame is received and 0b is stored to this bit.

Note: This bit is not available in the classical CAN function.

RFFDF bit (CAN FD Format bit)

The RFFDF bit has the same value as the FDF bit of the received CAN-FD frame.

Note: This bit is not available in the classical CAN function.

RFIFL[1:0] bits (RX FIFO Buffer Information Label Field)

The RFIFL[1:0] bits store the information label value from the related Global Acceptance Filter List entry.

CFDRFPTR[15:0] bits (RX FIFO Buffer Pointer Field)

The CFDRFPTR[15:0] bits store the pointer value from the related Global Acceptance Filter List entry.

(8) RX FIFO Access Data Field p Register n (CFDRFDFpn) (p = 0 to 15, n = 0 to 7)

The CFDRFDFpn register (p = 0 to 15, n = 0 to 7) stores data bytes (4 × p) to data bytes (4 × p + 3) of the received message.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 600Ch + p x 0004h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB_HH[7:0]								RFDB_HL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB_LH[7:0]								RFDB_LL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	RFDB_HH[7:0]	0h	R	RX FIFO Buffer Data Byte (4 × p + 3)
23 to 16	RFDB_HL[7:0]	0h	R	RX FIFO Buffer Data Byte (4 × p + 2)
15 to 8	RFDB_LH[7:0]	0h	R	RX FIFO Buffer Data Byte (4 × p + 1)
7 to 0	RFDB_LL[7:0]	0h	R	RX FIFO Buffer Data Byte (4 × p)

RFDB_LL[7:0] bits (RX FIFO Buffer Data Byte (4 × p))

The RFDB_LL[7:0] bits store data bytes (4 × p) of the message present in the FIFO buffer. Unused data bytes are filled with 00h according to the configured data payload size CFDRFCCn.RFPLS.

RFDB_LH[7:0] bits (RX FIFO Buffer Data Byte (4 × p + 1))

The RFDB_LH[7:0] bits store data bytes (4 × p + 1) of the message present in the FIFO buffer. Unused data bytes are filled with 00h.

RFDB_HL[7:0] bits (RX FIFO Buffer Data Byte (4 × p + 2))

The RFDB_HL[7:0] bits store data bytes (4 × p + 2) of the message present in the FIFO buffer. Unused data bytes are filled with 00h.

RFDB_HH[7:0] bits (RX FIFO Buffer Data Byte (4 × p + 3))

The RFDB_HH[7:0] bits store data bytes (4 × p + 3) of the message present in the FIFO buffer. Unused data bytes are filled with 00h.

(9) Common FIFO Access ID Register n (CFDCFIDn) (n = 0 to 17)

The CFDCFIDn register (n = 0 to 17) stores the ID field, IDE bit, and RTR bit of the message.

Remark In TX mode, you can read data from the FIFO, only for the current entry based on the write pointer value, and not for the other entries.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 6400h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFIDE	CFRTR	THLEN	CFID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CFIDE	0h	RW	Common FIFO Buffer IDE bit 0b: STD-ID is to be transmitted or has been received 1b: EXT-ID is to be transmitted or has been received
30	CFRTR	0h	RW	Common FIFO Buffer RTR bit 0b: Data frame 1b: Remote frame
29	THLEN	0h	RW	THL Entry Enable TX FIFO mode: 0b: Entry is not to be stored in THL after successful TX 1b: Entry is to be stored in THL after successful TX RX FIFO mode: Reserved, This bit is read as 0b.
28 to 0	CFID[28:0]	0h	RW	Common FIFO Buffer ID Field STD-ID/EXT-ID fields

CFID[28:0] bits (Common FIFO Buffer ID Field)

The CFID[28:0] bits are the bits of the STD-ID/EXT-ID fields of the message in the FIFO buffer.

For alignment of these bits in standard and extended frame format, see **7.9.2.2.89 Identifier Bits Alignment**.

In TX mode, you can write and read data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

THLEN bit (THL Entry Enable)

The THLEN bit controls the storage of an entry corresponding to the transmitted message in the TX History list at the end of a successful transmission.

In TX mode, you can write and read data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

CFRTR bit (Common FIFO Buffer RTR bit)

The CFRTR bit selects whether a data frame or a remote frame is to be transmitted from or was received in the FIFO buffer.

Note: There are no remote frames in CAN FD format. When a CAN-FD frame is received (RX mode), the register reflects the state of the received value (RRS bit in FD frame format). When CAN-FD transmission (TX or GW mode CFDCFID.CFFDF = 1b), the bit is always transmitted dominant (data frame).

In TX mode, you can write and read data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

CFIDE bit (Common FIFO Buffer IDE bit)

The CFIDE bit selects whether a message with EXT-ID or STD-ID is to be transmitted from or was received in the FIFO buffer.

In TX mode, you can write and read data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

(10) Common FIFO Access Pointer Register n (CFDCFPTRn) (n = 0 to 17)

The CFDCFPTRn register (n = 0 to 17) stores the DLC and Timestamp fields.

Remark In TX mode, you can read data from the FIFO buffer, only for the current entry based on the write pointer value, and not for the other entries.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 6404h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDLC[3:0]				-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFTS[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	CFDLC[3:0]	0h	RW	Common FIFO Buffer DLC Field Number of data bytes received in a CAN frame, or transmitted in a CAN frame.
27 to 16	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
15 to 0	CFTS[15:0]	0h	RW	Common FIFO Timestamp Value Timestamp value of the received CAN frame (FIFO in RX mode).

CFTS[15:0] bits (Common FIFO Timestamp Value)

The CFTS[15:0] bits store the timestamp value taken at the capture point as configured by the CFDGFDCFG.TSCCFG bit of the received message (if FIFO is configured in RX mode).

In TX mode, you can read and write data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

CFDLC[3:0] bits (Common FIFO Buffer DLC Field)

The CFDLC[3:0] bits store the number of data bytes that were received in the FIFO buffer or are to be transmitted.

See *Table 5* in *ISO 11898-1 (2015) Specification* for details in defining the number of data bytes.

In TX mode, you can read and write data from the FIFO buffers. Do not read data for the other entries in the FIFO when configured in TX mode.

In RX mode, you can only read data from the FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

(11) Common FIFO Access CAN-FD Control/Status Register n (CFDCFFDCSTSn) (n = 0 to 17)

The CFDCFFDCSTSn register (n = 0 to 17) shows the status of the FDF, BRS, and ESI bits, including the pointer of the received CAN-FD frame or the CAN-FD frame to transmit.

Remark In TX mode, you can read data from the FIFO, only for the current entry based on the write pointer value, and not for the other entries.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 6408h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFPTR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	CFIFL[1:0]		-	-	-	-	-	CFFDF	CFBRS	CFESI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	CFPTR[15:0]	0h	RW	Common FIFO Buffer Pointer Field
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9, 8	CFIFL[1:0]	0h	RW	COMMON FIFO Buffer Information Label Field
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	CFFDF*[1]	0h	RW	CAN FD Format bit 0b: No CAN-FD frame received or to transmit 1b: CAN-FD frame received or to transmit
1	CFBRS*[1]	0h	RW	Bit Rate Switch bit 0b: CAN-FD frame received or to transmit with no bit rate switch 1b: CAN-FD frame received or to transmit with bit rate switch
0	CFESI*[1]	0h	RW	Error State Indicator bit 0b: CAN-FD frame received or to transmit by error active node 1b: CAN-FD frame received or to transmit by error passive node

Note 1. This bit is not available in the classical CAN function.

CFESI bit (Error State Indicator bit)

In TX mode, you can read and write data from FIFO buffers. In this mode, when the CAN-FD module is not in error passive, the CFESI bit equals the write value. Otherwise, it is a don't care and the bit is transmitted as 1 on the CAN bus, indicating this is an error passive node.

In RX mode, you can only read data from FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

In RX or GW mode, the CFESI bit is updated with the ESI bit value of the CAN-FD frame when it has been received, indicating the error state of the transmitting node. In RX or GW mode, 0b is stored to this bit when the received FDF bit is 0b, this means a CAN 2.0 frame is received.

Note: This bit is not available in the classical CAN function.

CFBRS bit (Bit Rate Switch bit)

In TX mode, you can read and write data from FIFO buffers. In this mode, the CAN-FD module either transmits a 0b to indicate no bit rate switch in the frame to be transmitted or a 1 to indicate a bit rate switch in the frame to be transmitted.

In RX mode, you can only read data from FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

In RX or GW mode, the CFBRS bit is updated with the BRS bit value of the CAN-FD frame when it has been received, indicating whether there is a bit rate switch (1) or (0) on the CAN-FD frame.

In RX or GW mode, 0b is stored to the CFBRS bit when the received FDF bit is 0b, this means a CAN 2.0 frame is received.

Note: This bit is not available in the classical CAN function.

CFFDF bit (CAN FD Format bit)

In TX mode, you can read and write data from FIFO buffers. In this mode, the CAN-FD module either transmits a 0b to indicate a CAN 2.0 frame is to be transmitted or a 1 to indicate a CAN-FD frame is to be transmitted.

In RX mode, you can only read data from FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

In RX or GW mode, the CFFDF bit is updated with the FDF bit value of the CAN frame when it has been received, indicating whether it is a CAN 2.0 frame (0) or a CAN-FD frame (1).

This bit is not available in the classical CAN function.

CFIFL[1:0] bits (COMMON FIFO Buffer Information Label Field)

If the Common FIFO is configured in TX mode, the value programmed in CFDCFFDCSTSn.CFIFL[1:0] is stored together with additional message information, to the TX History List after successful transmission of the message.

The information label value from the related Global Acceptance Filter List entry is stored in these bits (if FIFO is configured in either RX or GW mode).

In TX mode, you can read and write data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

CFPTR[15:0] bits (Common FIFO Buffer Pointer Field)

If the Common FIFO is configured in TX mode, the value programmed in CFDCFFDCSTSn.CFPTR[15:0] is stored together with additional message information, to the TX History List after successful transmission of the message.

The pointer value from the related Global Acceptance Filter List entry is stored in these bits (if FIFO is configured in either RX or GW mode).

In TX mode, you can read and write data from FIFO buffers.

In RX mode, you can only read data from FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

(12) Common FIFO Access Data Field p Register n (CFDCFDFpn) (p = 0 to 15, n = 0 to 17)

The CFDCFDFpn register (p = 0 to 15, n = 0 to 17) stores data bytes (4 × p) to data bytes (4 × p + 3) of the message.

Remark In TX mode, you can read data from the FIFO, only for the current entry based on the write pointer value, and not for the other entries.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 640Ch + p x 0004h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB_HH[7:0]								CFDB_HL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB_LH[7:0]								CFDB_LL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CFDB_HH[7:0]	0h	RW	Common FIFO Buffer Data Byte (4 × p + 3)
23 to 16	CFDB_HL[7:0]	0h	RW	Common FIFO Buffer Data Byte (4 × p + 2)
15 to 8	CFDB_LH[7:0]	0h	RW	Common FIFO Buffer Data Byte (4 × p + 1)
7 to 0	CFDB_LL[7:0]	0h	RW	Common FIFO Buffer Data Byte (4 × p)

CFDB_LL[7:0] bits (Common FIFO Buffer Data Byte (4 × p))

The CFDB_LL[7:0] bits store data bytes (4 × p) of the message present in the FIFO buffer.

In TX mode, you can read and write data from the FIFO buffers.

In RX mode, you can only read data from the FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

In RX or GW mode, unused data bytes are filled with 00h, according to their configured data payload size CFDCFCCn.CFPLS.*[1]

CFDB_LH[7:0] bits (Common FIFO Buffer Data Byte (4 × p + 1))

The CFDB_LH[7:0] bits store data bytes (4 × p + 1) of the message present in the FIFO buffer.

In TX mode, you can read and write data from the FIFO buffers.

In RX mode, you can only read data from the FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

In RX or GW mode, unused data bytes are filled with 00h, according to their configured data payload size CFDCFCCn.CFPLS.*[1]

CFDB_HL[7:0] bits (Common FIFO Buffer Data Byte (4 × p + 2))

The CFDB_HL[7:0] bits store data bytes (4 × p + 2) of the message present in the FIFO buffer.

In TX mode, you can read and write data from the FIFO buffers.

In RX mode, you can only read data from the FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

In RX or GW mode, unused data bytes are filled with 00h, according to their configured data payload size CFDCFCCn.CFPLS.*[1]

CFDB_HH[7:0] bits (Common FIFO Buffer Data Byte (4 × p + 3))

The CFDB_HH[7:0] bits store data bytes (4 × p + 3) of the message present in the FIFO buffer.

In TX mode, you can read and write data from the FIFO buffers.

In RX mode, you can only read data from the FIFO buffers.

In GW mode, you cannot write data to the FIFO buffers.

In RX or GW mode, unused data bytes are filled with 00h, according to their configured data payload size CFDCFCCn.CFPLS.*[1]

Note 1. In RX or GW mode, unused data bytes are filled with 00h according to the configured data payload size CFDCFCCn.CFPLS, which is a CAN-FD feature not found in classical CAN.

(13) TX Message Buffer ID Register n (CFDTMIDn) (n = 0 to 383)

The CFDTMIDn register (n = 0 to 383) stores the ID, IDE, RTR fields and history configuration of the message to be transmitted from the associated buffer.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1_0000h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIDE	TMRTR	THLEN	TMID[28:16]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMID[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	TMIDE	0h	RW	TX Message Buffer IDE bit 0b: STD-ID is transmitted 1b: EXT-ID is transmitted
30	TMRTR	0h	RW	TX Message Buffer RTR bit 0b: Data frame 1b: Remote frame
29	THLEN	0h	RW	Tx History List Entry 0b: Entry not stored in THL after successful TX 1b: Entry stored in THL after successful TX
28 to 0	TMID[28:0]	0h	RW	TX Message Buffer ID Field STD-ID/EXT-ID fields

TMID[28:0] bits (TX Message Buffer ID Field)

The TMID[28:0] bits are bits of the STD-ID/EXT-ID fields of the message stored in this TX message buffer.

For alignment of these bits in standard and extended frame format, see **7.9.2.2.89 Identifier Bits Alignment**.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

THLEN bit (Tx History List Entry)

The THLEN bit controls the storage of an entry corresponding to the transmitted message in the TX History list at the end of a successful transmission.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

TMRTR bit (TX Message Buffer RTR bit)

The TMRTR bit selects whether a data frame or remote frame is to be transmitted from this TX message buffer.

Note: There are no remote frames in CAN-FD format. For a CAN-FD transmission (CFDTMFDCTR.CFFDF = 1b), this bit is always transmitted dominant (data frame).

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

TMIDE bit (TX Message Buffer IDE bit)

The TMIDE bit selects whether a message with EXT-ID or STD-ID is to be transmitted from this TX message buffer.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

(14) TX Message Buffer Pointer Register n (CFDTMPTRn) (n = 0 to 383)

The CFDTMPTRn register (n = 0 to 383) stores the DLC fields of the message to transmit from the associated buffer.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1_0004h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDLC[3:0]				-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	TMDLC[3:0]	0h	RW	TX Message Buffer DLC Field Number of data bytes to be transmitted in a CAN frame.
27 to 0	-	All 0	RW	Reserved Whenever it is read, undefined value is read. The written value should always be 0b.

TMDLC[3:0] bits (TX Message Buffer DLC Field)

The TMDLC[3:0] bits select the number of data bytes to be transmitted from this TX message buffer when the corresponding TMRTR bit is configured as 0b.

See Table 5 in ISO 11898-1 (2015) Specification for details in defining the number of data bytes to be transmitted.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

(15) TX Message Buffer CAN-FD Control Register n (CFDTMFDCTRn) (n = 0 to 383)

The CFDTMFDCTRn register (n = 0 to 383) shows the status of the FDF, BRS, and ESI bits, including the pointer fields of the CAN-FD frame to be transmitted.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1_0008h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMPTR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	TMIFL[1:0]		-	-	-	-	-	TMFDF	TMBRS	TMESI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TMPTR[15:0]	0h	RW	TX Message Buffer Pointer Field
15 to 10	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9, 8	TMIFL[1:0]	0h	RW	TX Message Buffer Information Label Field
7 to 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	TMFDF*[1]	0h	RW	CAN FD Format bit 0b: No CAN-FD frame to transmit 1b: CAN-FD frame to transmit
1	TMBRS*[1]	0h	RW	Bit Rate Switch bit 0b: CAN-FD frame to transmit with no bit rate switch 1b: CAN-FD frame to transmit with bit rate switch
0	TMESI*[1]	0h	RW	Error State Indicator bit 0b: CAN-FD frame to transmit by error active node 1b: CAN-FD frame to transmit by error passive node

Note 1. This bit is not available in the classical CAN function.

TMESI bit (Error State Indicator bit)

If the channel is not in error passive, then the TMESI bit equals the write value, otherwise it is a don’t care and the bit is transmitted as 1 on the CAN bus, indicating this is an error passive node.

Do not write to the TMESI bit when the related CAN-FD channel is in CH_SLEEP mode.

Note: This bit is not available in the classical CAN function.

TMBRS bit (Bit Rate Switch bit)

Do not write to the TMBRS bit when the related CAN-FD channel is in CH_SLEEP mode.

Note: This bit is not available in the classical CAN function.

TMFDF bit (CAN FD Format bit)

Do not write to the TMFDF bit when the related CAN-FD channel is in CH_SLEEP mode.

Note: This bit is not available in the classical CAN function.

TMIFL[1:0] bits (TX Message Buffer Information Label Field)

The TMIFL[1:0] bits store the information label value to be copied, together with additional message information, in the TX History List after successful transmission of the message.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

TMPTR[15:0] bits (TX Message Buffer Pointer Field)

The TMPTR[15:0] bits store the pointer value to be copied, together with additional message information in the TX History List after successful transmission of the message.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

(16) TX Message Buffer Data Field p Register n (CFDTMDFp_n) (p = 0 to 15, n = 0 to 383)

The CFDTMDFp_n register (p = 0 to 15, n = 0 to 383) stores data bytes (4 × p) to data bytes (4 × p + 3) of the message to transmit from the associated buffer.

Access Size : 8, 16, 32 bits
Address : <CFD_base> + 1_000Ch + p x 0004h + n x 0080h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB_HH[7:0]								TMDB_HL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB_LH[7:0]								TMDB_LL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TMDB_HH[7:0]	0h	RW	TX Message Buffer Data Byte (4 × p + 3)
23 to 16	TMDB_HL[7:0]	0h	RW	TX Message Buffer Data Byte (4 × p + 2)
15 to 8	TMDB_LH[7:0]	0h	RW	TX Message Buffer Data Byte (4 × p + 1)
7 to 0	TMDB_LL[7:0]	0h	RW	TX Message Buffer Data Byte (4 × p)

TMDB_LL[7:0] bits (TX Message Buffer Data Byte (4 × p))

Data bytes (4 × p)[7:0] of the message stored in the TX message buffer.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

TMDB_LH[7:0] bits (TX Message Buffer Data Byte (4 × p + 1))

Data bytes (4 × p + 1)[7:0] of the message stored in the TX message buffer.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

TMDB_HL[7:0] bits (TX Message Buffer Data Byte (4 × p + 2))

Data bytes (4 × p + 2)[7:0] of the message stored in the TX message buffer.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

TMDB_HH[7:0] bits (TX Message Buffer Data Byte (4 × p + 3))

Data bytes (4 × p + 3)[7:0] of the message stored in the TX message buffer.

Do not write to these bits when the related CAN-FD channel is in CH_SLEEP mode.

7.9.3 Operation

7.9.3.1 Overview

The modes of the CAN-FD module can be classified into 2 groups:

- Global modes
- Channel modes

7.9.3.2 Global Modes

Global modes are applicable for the complete CAN-FD module. The global modes of the CAN-FD module are:

- Global Sleep
- Global Reset
- Global Halt
- Global Operation

Figure 7.9-2 shows the possible transitions between the Global modes.

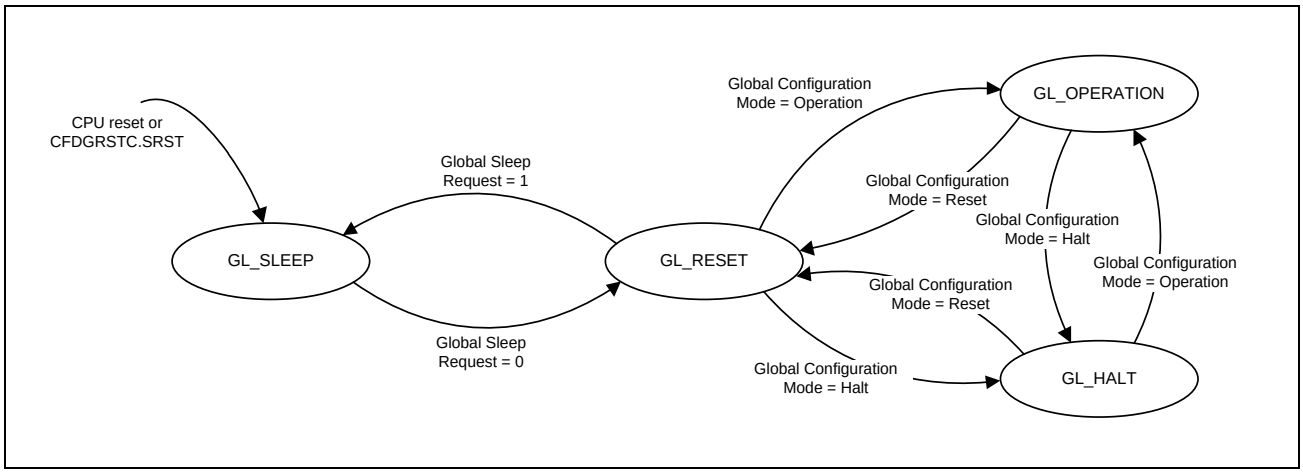


Figure 7.9-2 Transition between CAN-FD Global Modes

Changes in the Global mode can affect the Channel mode. **Table 7.9-15** shows the effect of a Global mode transition on a Channel mode.

Table 7.9-15 Possible CAN-FD Channel Modes and Global Modes

Current Global Mode	Target Global Mode			
	Sleep	Reset	Halt	Operation
Sleep	—	Ch-Sleep: Keep Ch-Reset: N/A Ch-Halt: N/A Ch-Oper: N/A	—	—
Reset	Ch-Sleep: Keep Ch-Reset: → Ch-Sleep Ch-Halt: N/A Ch-Oper: N/A	—	Ch-Sleep: Keep Ch-Reset: Keep Ch-Halt: N/A Ch-Oper: N/A	Ch-Sleep: Keep Ch-Reset: Keep Ch-Halt: N/A Ch-Oper: N/A
Halt	—	Ch-Sleep: Keep Ch-Reset: Keep Ch-Halt: → Ch-Reset Ch-Oper: N/A	—	Ch-Sleep: Keep Ch-Reset: Keep Ch-Halt: Keep Ch-Oper: N/A
Operation	—	Ch-Sleep: Keep Ch-Reset: Keep Ch-Halt: → Ch-Reset Ch-Oper: → Ch-Reset	Ch-Sleep: Keep Ch-Reset: Keep Ch-Halt: Keep Ch-Oper: → Ch-Halt	—

7.9.3.2.1 Global Sleep Mode

After the release of a hardware reset or after setting and clearing the CFDGRSTC.SRST bit, the CAN-FD module automatically enters Global Sleep mode.

The CAN-FD module also enters the Global Sleep mode when the Global Sleep Request bit is set while it is in Global Reset mode. This control bit cannot be set in Global Halt mode or Global Operation mode.

Setting the Global Sleep Request bit sets all Channel Sleep Request bits and forces all channels into the Channel Sleep mode. Sleep mode is used for power saving purpose. When CAN-FD module is in Global Sleep mode, only the clock for CPU write access to the Global Sleep Mode Request bit is active. All other clocks are stopped and all other functions of the CAN-FD module are suspended.

Read access from all registers is still possible and all register values are preserved.

In Global Sleep mode, RAM access is prohibited because the logic which generates a RAM address does not operate.

After setting the Global Sleep Request bit, it is necessary to confirm that the Global Sleep status has been updated, indicating successful transition to Global Sleep mode before the Global Sleep Request bit can be cleared again.

Figure 7.9-3 shows the procedure for entering Global Sleep mode and **Figure 7.9-4** shows the procedure for exiting Global Sleep mode.

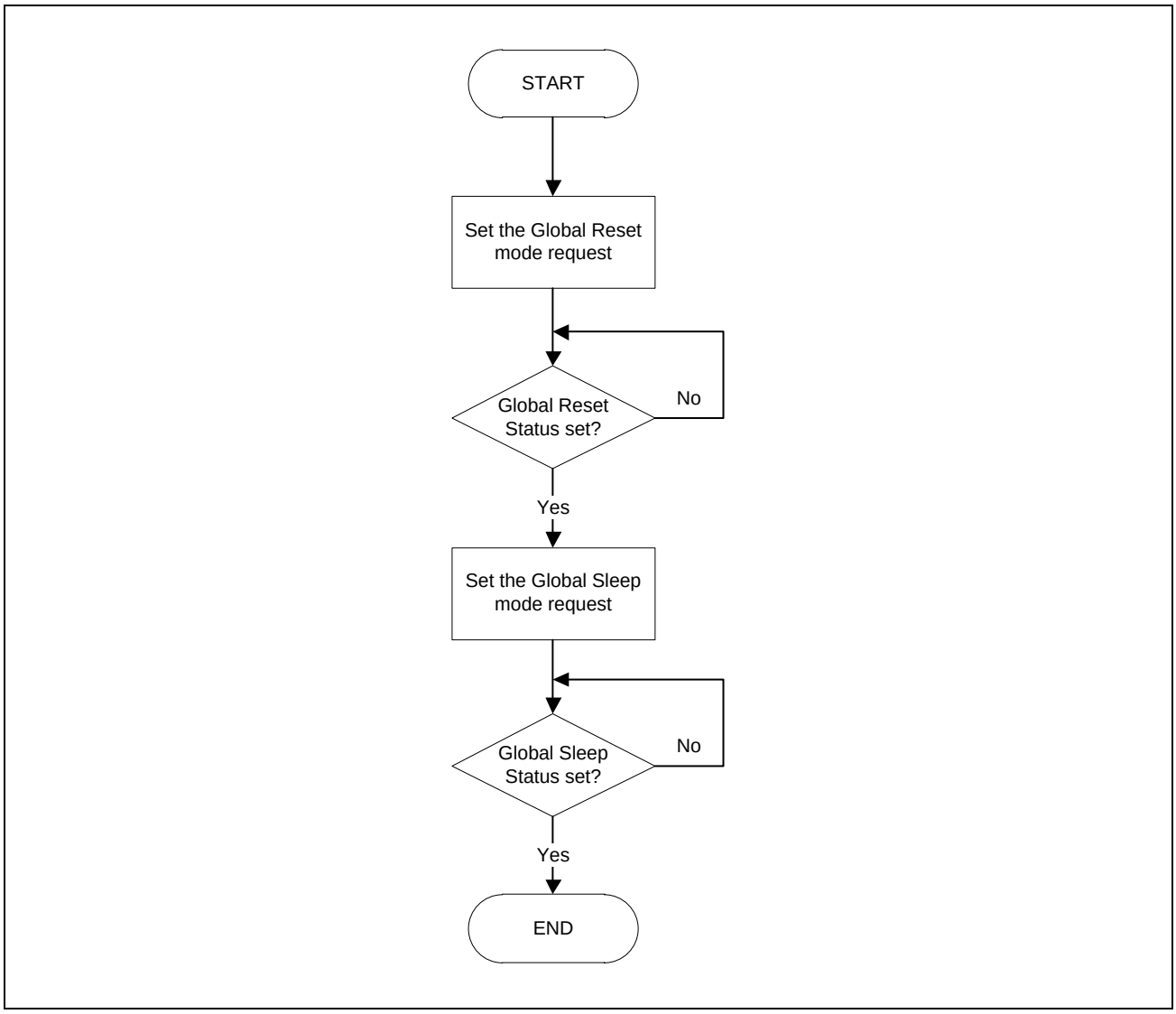


Figure 7.9-3 Procedure for Entering Global Sleep Mode

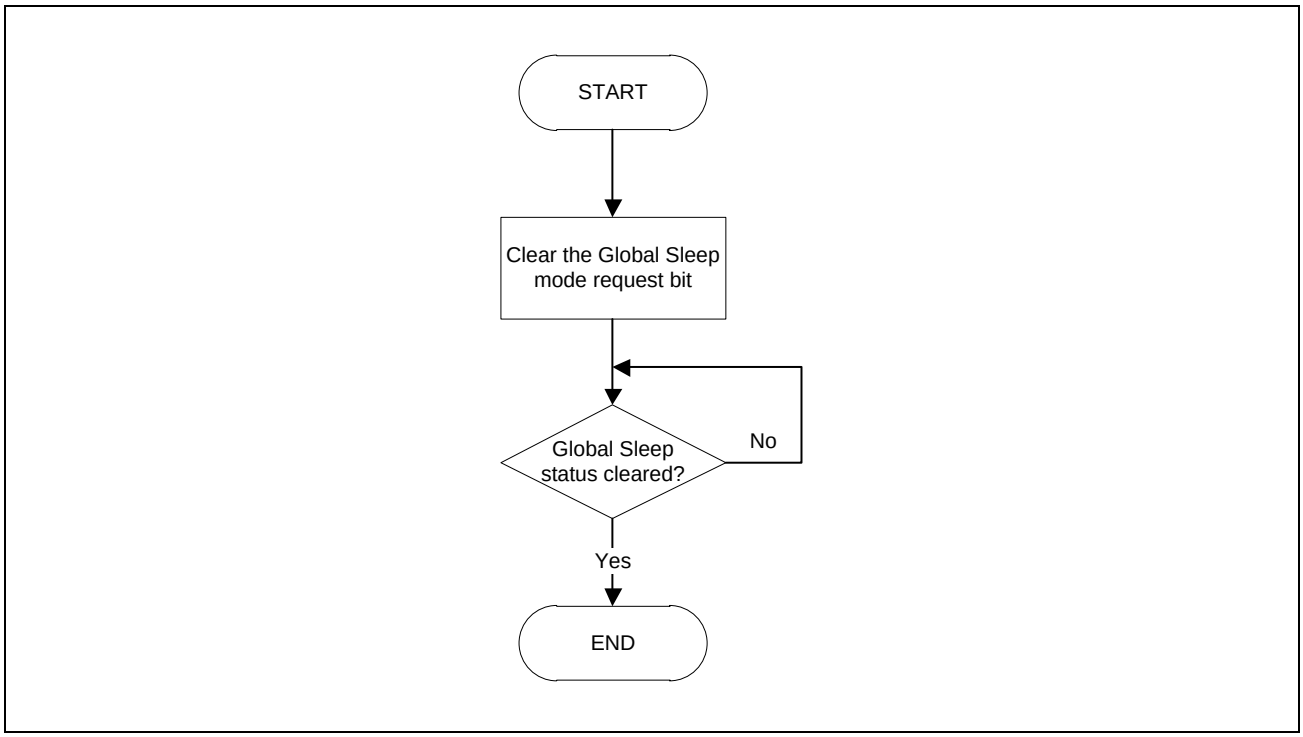


Figure 7.9-4 Procedure for Exiting Global Sleep Mode

7.9.3.2.2 Global Reset Mode

The CAN-FD module enters the Global Reset mode in the following ways:

- Global Mode Control bits CFDGCTR.GMDC[1:0] in the Global Control Register is configured for Global Reset mode while the CAN-FD module is in Global Halt or Global Operation mode.
- Global Sleep Mode Request bit is cleared while CAN-FD module is in Global Sleep mode.

In Global Reset mode, all CAN-FD module functions are suspended and all status and flag registers are initialized.

Additionally all FIFOs and all channel TX Queues are disabled and transmission control bits are cleared.

Configuration registers (except the test mode registers) are not initialized in this mode to their device reset values and the CAN-FD module can be configured.

See **7.9.3.4 Global Mode and Channel Mode Transition Interactions** for a detailed description of the behavior of all registers when transition to Global Reset mode.

Setting the Global mode to Reset by setting the Global Mode Control bits CFDGCTR.GMDC[1:0] in the Global Control Register to 01b sets all Channel Mode Control bits CFDCnCTR.CHMDC[1:0] in the Channel Control Registers to 01b and forces all channels into the Channel Reset mode.

For channels that are already in Channel Reset mode or Channel Sleep mode, this automatic transition is not performed (CFDCnCTR.CHMDC[1:0] of related channel already set to 01b).

After setting Global Mode Control bits CFDGCTR.GMDC[1:0] to Reset mode, it is necessary to confirm that the Reset Mode Status bit CFDGSTS.GRSTSTS in the Global Status Register has been updated, indicating successful transition to Global Reset mode before CFDGCTR.GMDC can be changed again.

Figure 7.9-5 shows the procedure for entering Global Reset mode and **Figure 7.9-6** shows the procedure for exiting Global Reset mode.

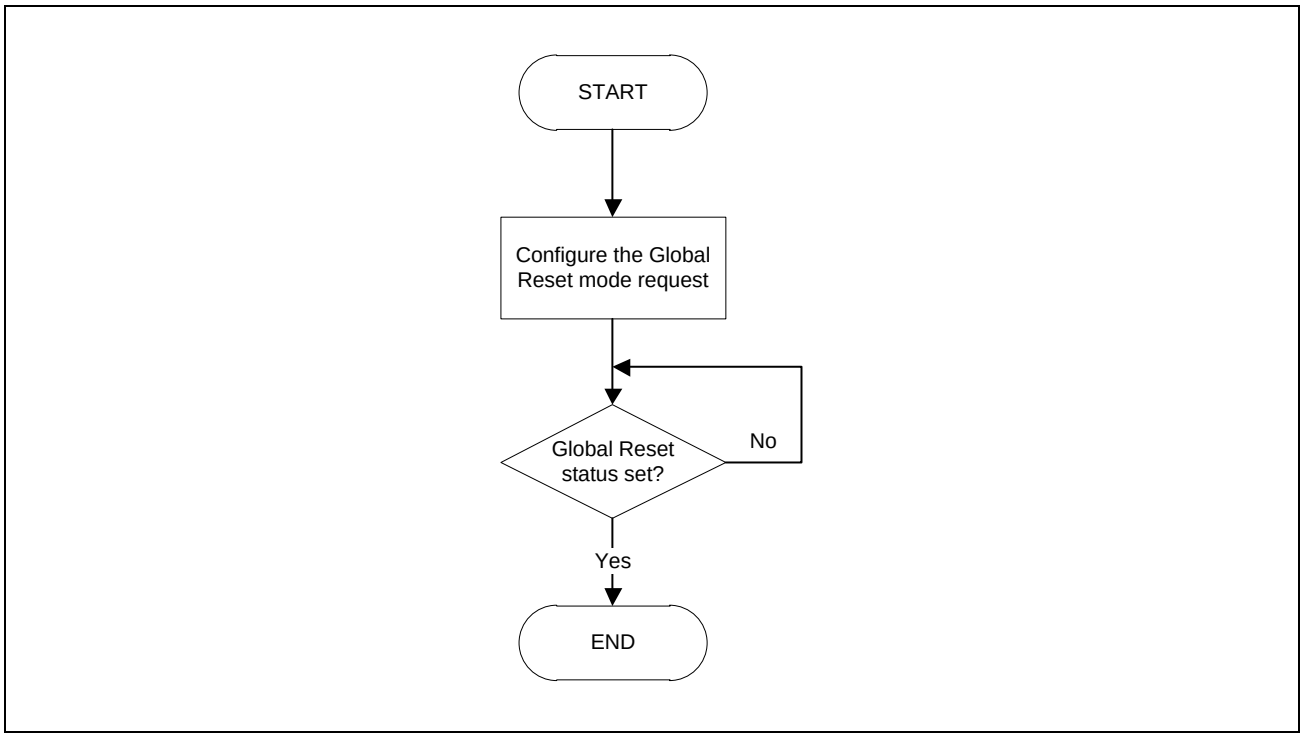


Figure 7.9-5 Procedure for Entering Global Reset Mode

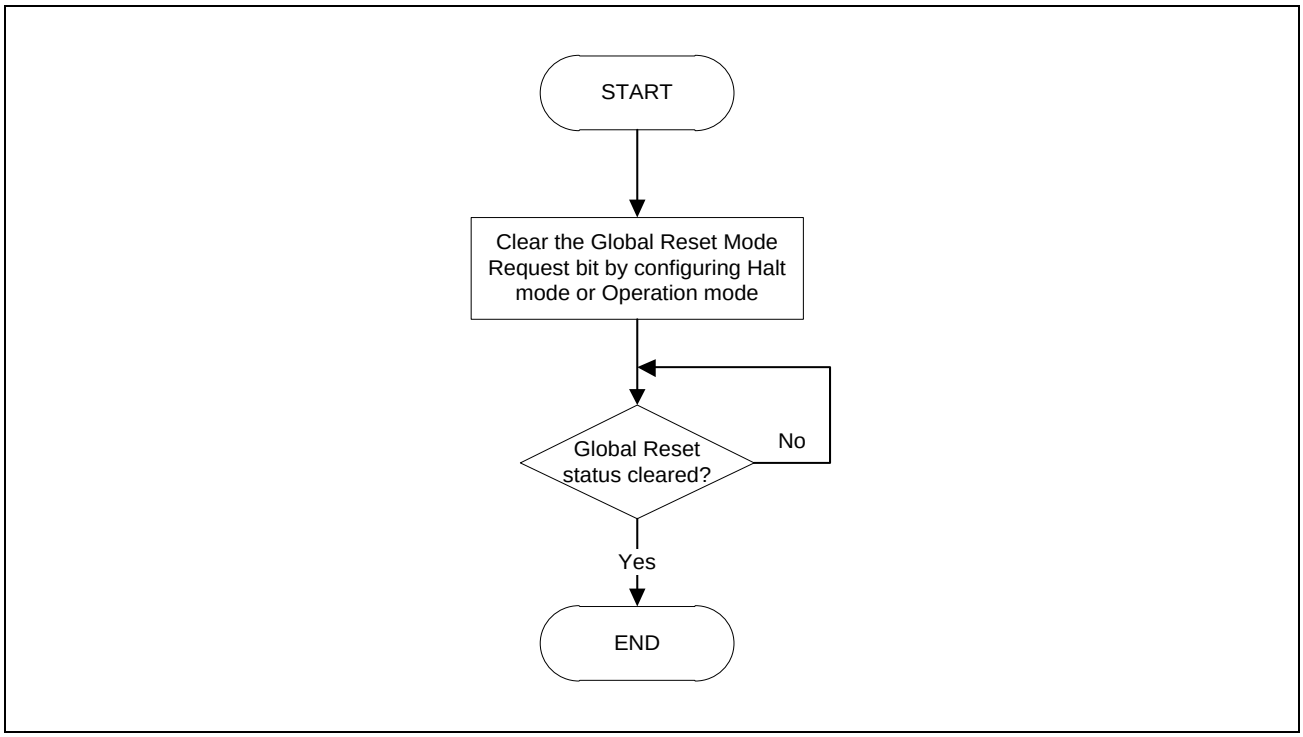


Figure 7.9-6 Procedure for Exiting Global Reset Mode

7.9.3.2.3 Global Halt Mode

The CAN-FD module enters the Global Halt mode in the following ways:

- Global Mode Control bits CFDGCTR.GMDC[1:0] in the Global Control Register is configured for Global Halt mode while the CAN-FD module is in Global Reset mode:
  - The channel is in either Channel Reset or Channel Sleep mode and remains in this mode.
- Global Mode Control bits CFDGCTR.GMDC[1:0] in the Global Control Register is configured for Global Halt mode while the CAN-FD module is in Global Operation mode:
  - All channels in Channel Reset, Channel Halt, or Channel Sleep mode remain in this mode
  - All channels in Channel Operation mode transit to Channel Halt mode
  - Global Halt Mode Status bit is set when all channels have left Channel Operation mode.

If a transmission or reception is ongoing for a channel, the transition to Channel Halt mode is delayed until completion of the communication.

Similarly, if a channel is in bus-off state, the full bus-off recovery sequence may be delayed depending on the channel configuration.

In Global Halt mode, all communications are suspended and CAN-FD logic does not cause any change to the Status and Flag registers (only when a channel is in the bus-off state that its REC and TEC values are cleared). Additionally, the test mode configuration and control registers are not initialized in this mode.

The Global Halt mode should be used to configure global module test modes.

See **7.9.3.4 Global Mode and Channel Mode Transition Interactions** for a detailed description of the behavior of all registers when transition to Global Halt mode is performed.

Setting the Global mode to Halt mode by setting the Global Mode Control bits CFDGCTR.GMDC[1:0] in the Global Control Register to 10b sets all Channel Mode Control bits CFDCnCTR.CHMDC[1:0] in the Channel Control Registers to 10b for the channels that are in Channel Operation mode and forces these channels into the Channel Halt mode.

For channels that are already in Channel Reset, Channel Halt, or Channel Sleep mode, this automatic transition is not performed.

Therefore, the Global Halt mode request can be used to shut down all CAN-FD channel communications without loss of messages and disruption on the related CAN bus (no interruption of reception/transmission processes on the channels).

After setting the Global Mode Control bits CFDGCTR.GMDC[1:0] to Halt mode, it is necessary to confirm that the Halt Mode Status bit CFDGSTS.GHLTSTS in the Global Status Register has been updated to indicate a successful transition to Global Halt mode. Do not specify any other SFR setting until confirming CFDGSTS.GHLTSTS is set.

Figure 7.9-7 shows the procedure for entering Global Halt mode and **Figure 7.9-8** shows the procedure for exiting Global Halt mode.

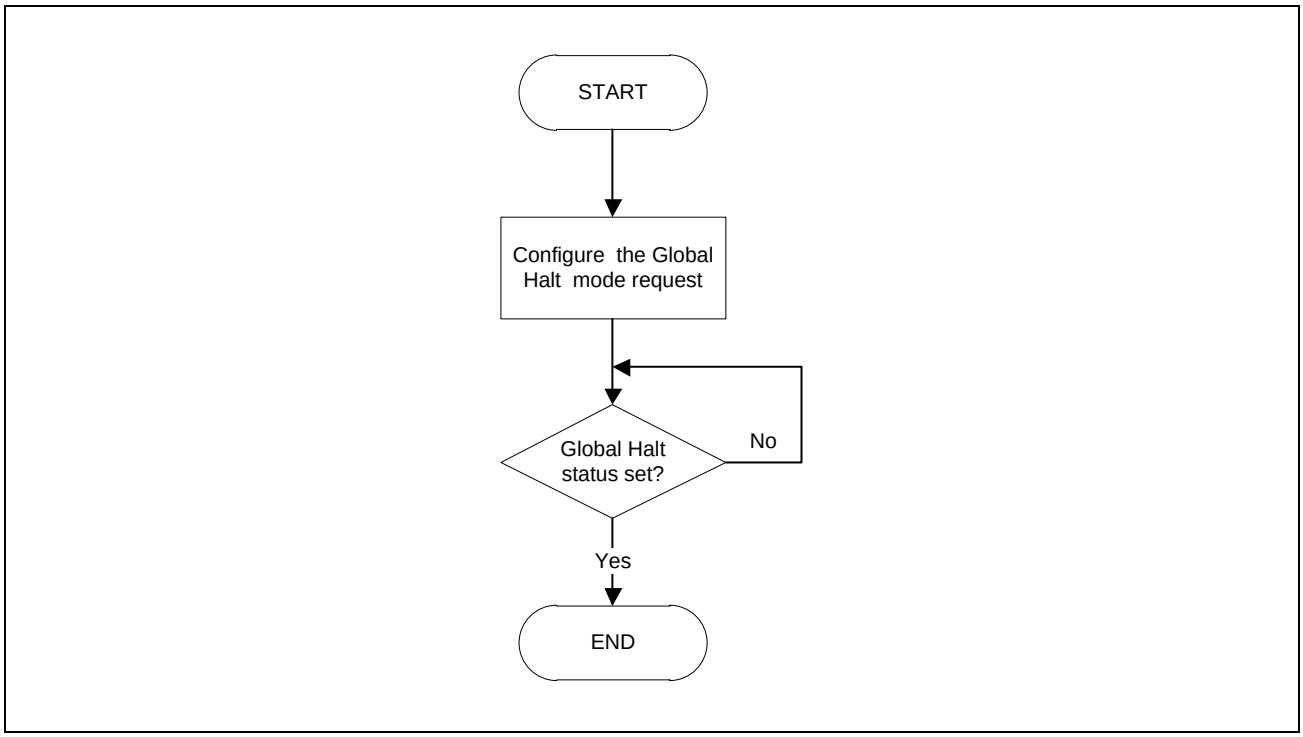


Figure 7.9-7 Procedure for Entering Global Halt Mode

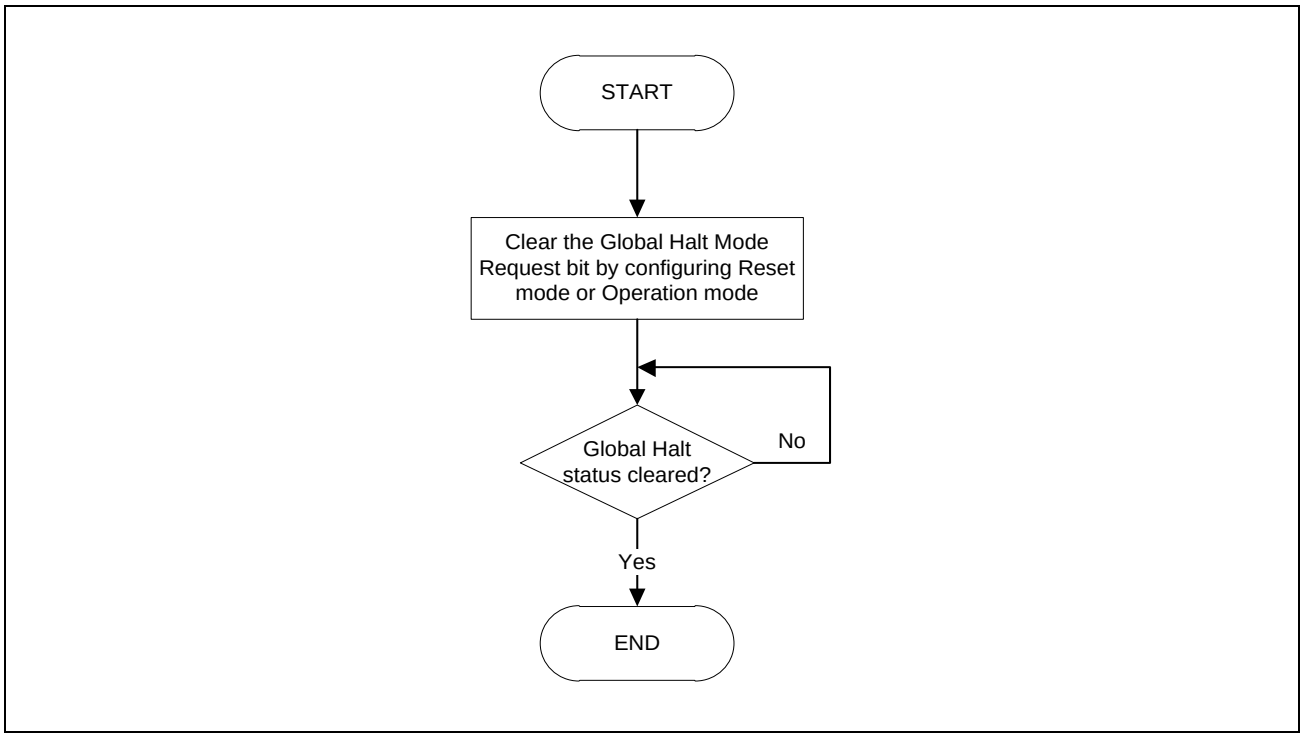


Figure 7.9-8 Procedure for Exiting Global Halt Mode

7.9.3.2.4 Global Operation Mode

The CAN-FD module enters the Global Operation mode when the Global Mode Configuration bits are set to Global Operation mode.

The CAN-FD channels can only be set to Channel Operation mode and start CAN communication when CAN-FD is in Global Operation mode.

After setting the Global Mode Control bits CFDGCTR.GMDC[1:0] to Global Operation mode, it is necessary to confirm that the Global Reset Mode Status bit CFDGSTS.GRSTSTS and the Global Halt Mode Status bit CFDGSTS.GHLTSTS in the Global Status Register have been cleared to indicate a successful transition to Global Operation mode before CFDGCTR.GMDC can be modified again.

Figure 7.9-9 shows the procedure for entering Global Operation mode and **Figure 7.9-10** shows the procedure for exiting Global Operation mode.

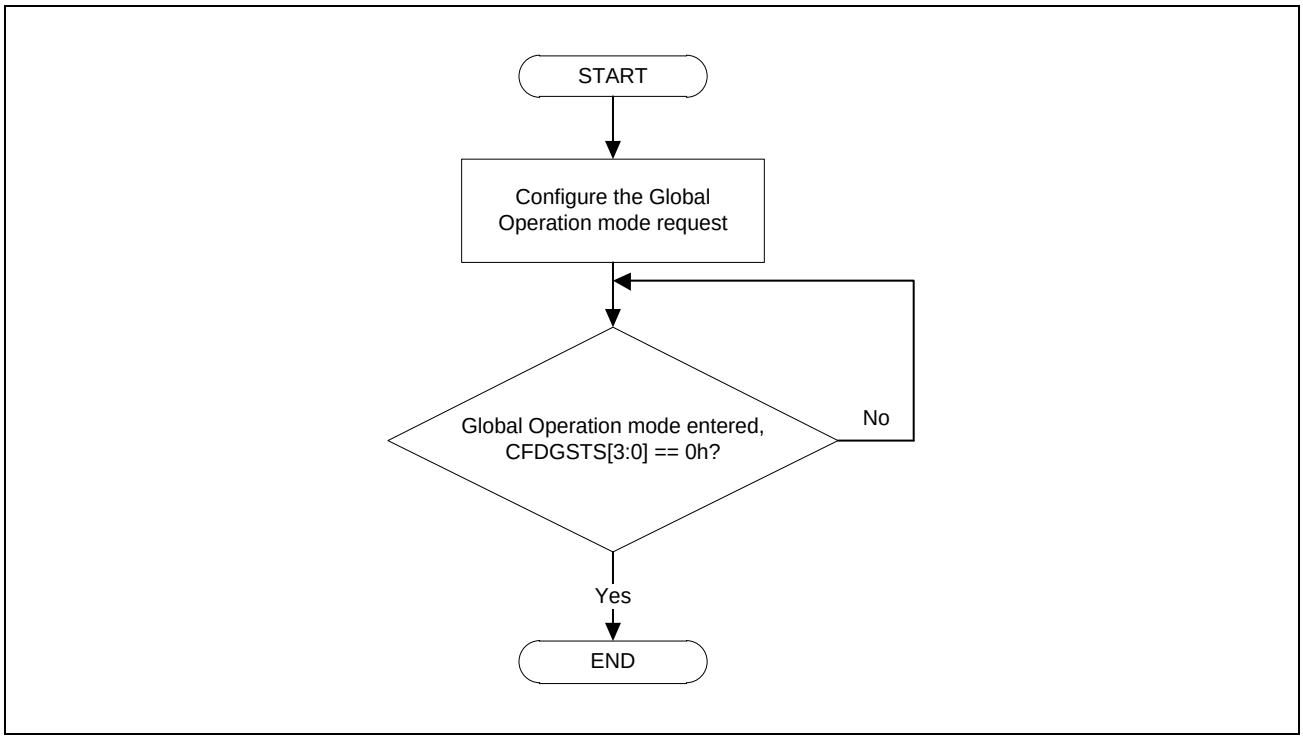


Figure 7.9-9 Procedure for Entering Global Operation Mode

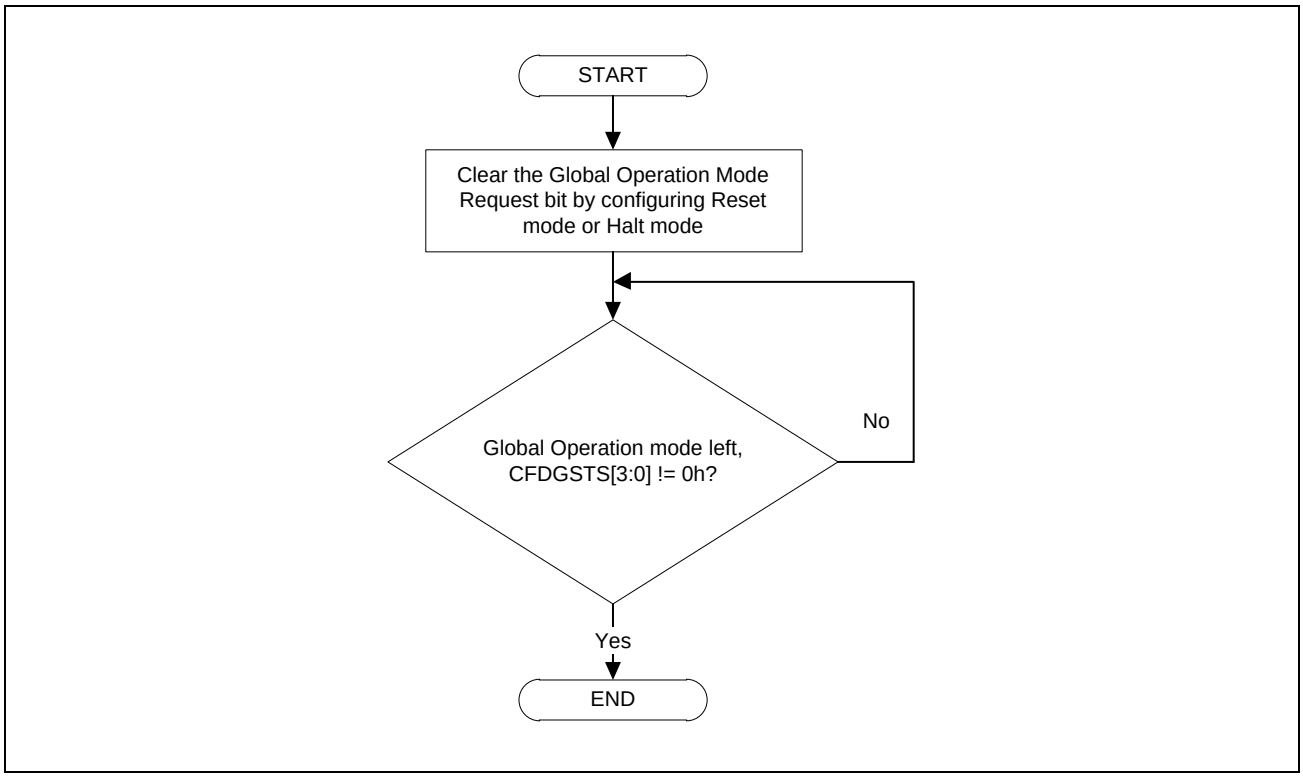


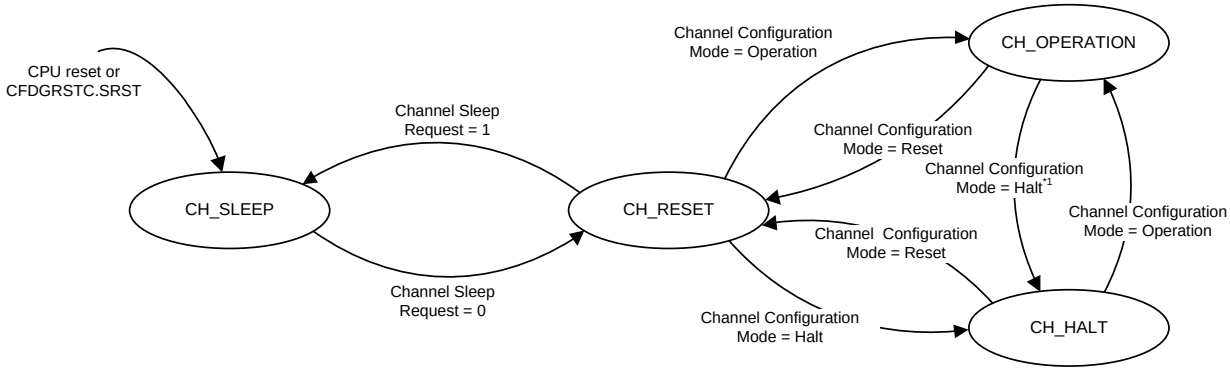
Figure 7.9-10 Procedure for Exiting Global Operation Mode

7.9.3.3 Channel Modes

Each CAN channel can be in one of the following four channel modes:

- Reset
- Halt
- Operation
- Sleep

Figure 7.9-11 shows possible transitions between the channel modes.



Note 1. The transition timing depends on BOM[1:0] setting in the Channel Control Register (CFDCnCTR):
When BOM[1:0] = 01b, the state transition timing is immediate after entering bus-off state
When BOM[1:0] = 10b, the state transition timing is at the end of bus-off state
When BOM[1:0] = 11b, the state transition timing is matching with the setting of CHMDC to HALT mode

Figure 7.9-11 Transition between CAN Channel Modes

7.9.3.3.1 CAN Channel Sleep Mode

After the release of a hardware reset or after setting and clearing the CFDGRSTC.SRST bit, each CAN channel of the CAN- FD module automatically enters Channel Sleep mode.

Each CAN channel also enters Channel Sleep mode when the related Channel Sleep Mode Request bit is set while the CAN channel is in Channel Reset mode. Do not set this control bit in Channel Halt mode or Channel Operation mode.

Entering the CAN Channel Sleep mode instantly stops the clock supplied to the CAN channel unit and therefore reduces power consumption.

After setting the Channel Sleep Mode Request bit, it is necessary to confirm that the Channel Sleep mode status has been updated to indicate a successful transition to Channel Sleep mode before the Channel Sleep Mode Request bit can be cleared again.

During Channel Sleep mode, do not write to channel related registers. Read operation is still possible.

7.9.3.3.2 CAN Channel Reset Mode

A CAN-FD CAN channel enters the CAN Channel Reset mode in the following ways:

- Channel Mode Control bits CFDCnCTR.CHMDC[1:0] in the Channel Control Registers is configured for Channel Reset mode while the related CAN channel is in Channel Halt mode or Channel Operation mode
- Channel Sleep Mode Request bit is cleared while the related CAN channel is in Channel Sleep mode
- Global Mode Control bits CFDGCTR.GMDC[1:0] is set to Global Reset mode and CAN channel is not in Channel Sleep mode or Channel Reset mode.

In Channel Reset mode, all CAN channel status and flag registers are initialized.

Additionally all channel-related transmission control bits are cleared and the channel-related TX Queue is disabled.

Configuration registers (except the Channel Test Mode registers) are not initialized in this mode and the CAN channel can be configured for communication.

See **7.9.3.4 Global Mode and Channel Mode Transition Interactions** for a detailed description of the behavior of all registers when transition to Channel Reset mode is performed.

After setting the Channel Mode Control bits CFDCnCTR.CHMDC[1:0] to Channel Reset mode, it is necessary to confirm that the Reset Mode Status bit CFDCnSTS.CRSTSTS in the related Channel Status Registers has been updated to indicate a successful transition to Channel Reset mode before the related CFDCnCTR.CHMDC[1:0] bits can be modified again.

See **Table 7.9-16** for the behavior of transitioning to Channel Reset mode while CAN communication is ongoing.

Table 7.9-16 Transition Behavior in CAN Reset Mode and Halt Mode

Mode	State		
	Receiver	Transmitter	Bus-Off
CAN Channel Reset mode (CFDCnCTR.CHMDC[1:0] = 01b)	The CAN channel transits to Channel Reset mode without waiting for the completion of the ongoing reception.*1	The CAN channel transits to Channel Reset mode without waiting for the completion of the ongoing transmission.*1	The CAN channel transits to Channel Reset mode without waiting for the completion of the bus-off recovery.
CAN Channel Halt Mode (CFDCnCTR.CHMDC[1:0] = 10b)	The CAN channel transits to Channel Halt mode at the end of the ongoing reception or error.*2	The CAN channel transits to Channel Halt mode after completion of the ongoing transmission.	When CFDCnCTR.BOM[1:0] is set to 00b, a Channel Halt mode request is accepted only after the completion of the full bus-off recovery sequence. When CFDCnCTR.BOM[1:0] is set to 10b, the CAN channel transits automatically to Channel Halt mode after waiting for the completion of the bus-off recovery. When CFDCnCTR.BOM[1:0] is set to 01b, the CAN channel transits automatically to Channel Halt mode without waiting for the completion of the bus-off recovery. When CFDCnCTR.BOM[1:0] is set to 11b, the CAN channel transits to Channel Halt mode as soon as Channel Halt mode is requested (without waiting for the completion of the bus-off recovery).

Note 1. If the entry to Channel Reset is required only at the end of an ongoing communication, then Channel Halt mode can be requested first to prevent interruption of CAN communication by direct transition to Channel Reset mode. After the CAN channel enters Channel Halt mode, the Channel Reset mode can be requested.

Note 2. If CAN communication is locked at a dominant level after an error flag, software can detect this situation by monitoring the channel related Bus Lock flag and resolve the lock condition by setting the CAN channel to Channel Reset mode.

7.9.3.3.3 CAN Channel Halt Mode

A CAN-FD CAN channel enters the CAN Channel Halt mode in the following ways:

- Channel Mode Control bits CFDCnCTR.CHMDC[1:0] in the Channel Control Registers are configured for Channel Halt mode while the related CAN channel is in Channel Reset mode or Channel Operation mode
- Global Mode Control bits CFDGCTR.GMDC[1:0] are set to Global Halt mode and CAN channel is in Channel Operation mode.

In Channel Halt mode, all channel CAN communication is suspended but all status and flag registers remain unchanged during Channel Halt mode entry (except for the bus-off case where REC and TEC values are cleared for the channel).

In addition, the Channel Test Mode Configuration and Control registers are not initialized in this mode.

The Channel Halt mode should be used to configure Channel Test modes.

See **7.9.3.4 Global Mode and Channel Mode Transition Interactions** for a detailed description of the behavior of all registers when transition to Channel Halt mode is performed.

After setting the Channel Mode Control bits CFDCnCTR.CHMDC[1:0] to Channel Halt mode, it is necessary to confirm that the Halt Mode Status bit CFDCnSTS.CHLTSTS in the related Channel Status Register is updated to indicate a successful transition to Channel Halt mode before the related CFDCnCTR.CHMDC[1:0] bits can be modified again.

See **Table 7.9-16** for the behavior of transitioning to Channel Halt mode while CAN communication is ongoing.

7.9.3.3.4 CAN Channel Operation Mode

The Channel Operation mode is activated by setting the CFDCnCTR.CHMDC[1:0] bits to 00b. If 11 consecutive recessive bits are detected after entering CAN Channel Operation mode, the CFDCnSTS.COMSTS bit is set and the CAN channel:

- Enables the functions of the channel communication by allowing the channel to become an active node on the CAN network
- Releases the internal fault confinement logic including receive and transmit error counters

At this point, the CAN channel can start transmission and reception of CAN messages.

Within the CAN Channel Operation mode, the channel may be in four different sub-modes, depending on which type of communication functions are performed (see **Figure 7.9-12**):

- Channel idle: The CAN channel is neither receiving nor transmitting
- Channel receives: The channel is receiving a CAN message sent by another CAN node
- Channel transmits: The channel is transmitting a CAN message

  Note: The channel may receive its own message simultaneously when Self Test mode is enabled.
- Channel is in bus-off state: The CAN channel is cut-off from CAN bus communication.

After setting the Channel Mode Control bits CFDCnCTR.CHMDC[1:0] to Channel Operation mode, it is necessary to confirm that the Channel Reset Mode Status bit CFDCnSTS.CRSTSTS and the Channel Halt Mode Status bit CFDCnSTS.CHLTSTS in the Channel Status Register have been updated to indicate a successful transition to Channel Operation mode before the related CFDCnCTR.CHMDC[1:0] bits can be changed again.

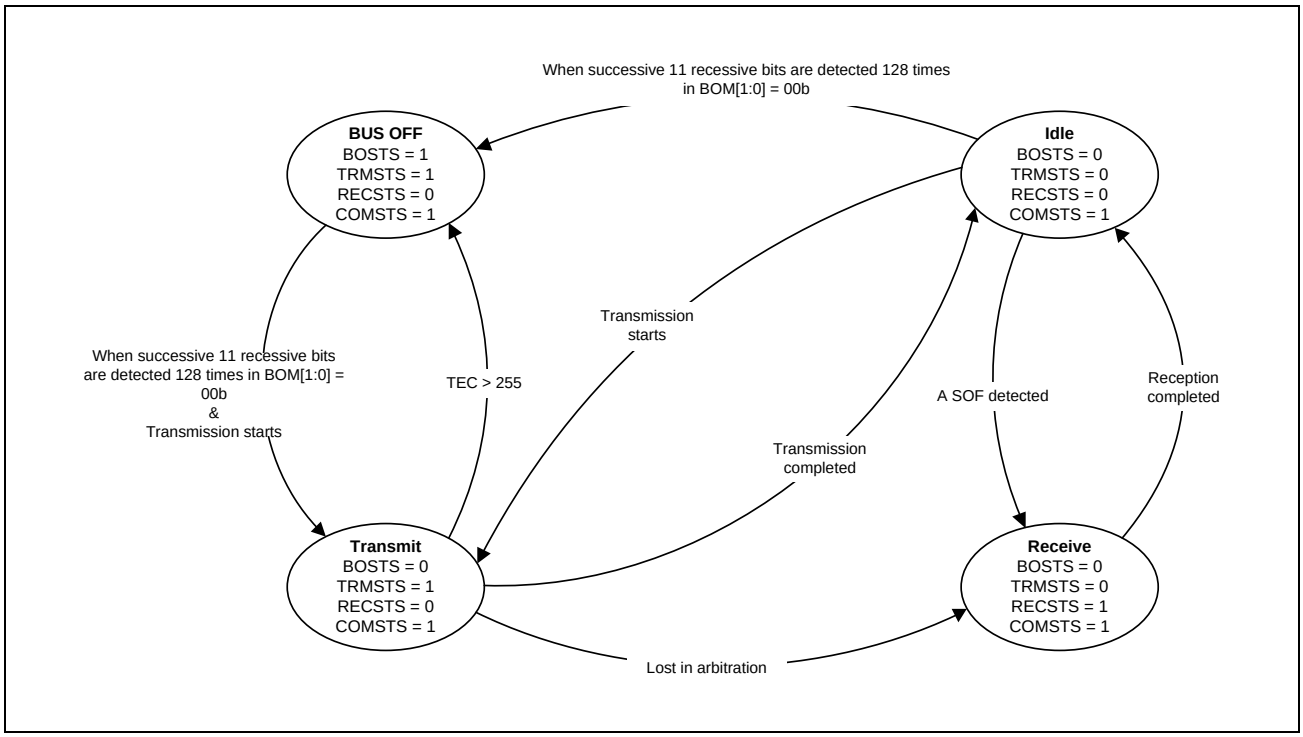


Figure 7.9-12 Sub-Modes of CAN Channel Operation Mode (Only when BOM[1:0] = 00b)

7.9.3.3.5 CAN Channel Bus-Off State

The CAN channel bus-off state is entered according to the fault confinement rules of the CAN specification. The following modes can be configured for returning to the CAN Channel Operation mode from the bus-off state:

- CFDCnCTR.BOM[1:0] = 00b:
  Bus-Off recovery is compliant to ISO 11898-1, namely the CAN channel re-enters CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. TEC and REC counters are initialized to 0b. The Bus- Off Recovery Flag CFDCnERFL.BORF is set in this case.
- CFDCnCTR.BOM[1:0] = 01b:
  The CAN channel changes the value of the CFDCnCTR.CHMDC[1:0] bits within the CAN Channel Control Register to 10b and switches immediately to Channel Halt mode automatically after entering bus-off state. TEC and REC counters are initialized to 0 and the Bus-Off Recovery Flag CFDCnERFL.BORF is not set in this case.
- CFDCnCTR.BOM[1:0] = 10b:
  The CAN channel changes the value of the CFDCnCTR.CHMDC[1:0] bits within the CAN Channel Control Register to 10b as soon as it reaches bus-off state and enters Channel Halt mode automatically after the CAN channel has completed the bus-off recovery sequence (after 11 consecutive recessive bits are detected 128 times). TEC and REC counters are initialized to 0b and the Bus-Off Recovery Flag CFDCnERFL.BORF is set in this case.
- CFDCnCTR.BOM[1:0] = 11b:
  Bus-off recovery is initiated but CAN channel can immediately enter Channel Halt mode when still in bus-off state if a request is made to enter Channel Halt mode.
  TEC and REC counters are initialized to 0b and the Bus-Off Recovery Flag CFDCnERFL.BORF is not set. Without setting CFDCnCTR.CHMDC[1:0] = 10b and when 11 recessive bits is detected 128 times continuously, transition conditions become the same as CFDCnCTR.BOM[1:0] = 00b.

Note: If the recovery from bus-off occurs normally in Channel Halt mode (after waiting for 128 sequences of 11 consecutive recessive bits), and no halt request has been generated during this period, then the Bus-Off Recovery flag CFDCnERFL.BORF is set.

When software writes to the CFDCnCTR.CHMDC[1:0] bit at the same time as the CAN channel enters Halt mode (at the start of bus-off when CFDCnCTR.BOM[1:0] = 01b, or at the end of bus-off when CFDCnCTR.BOM[1:0] = 10b), the software request has the highest priority.

Note: In the above case, the automatic setting of the CFDCnCTR.CHMDC[1:0] bits to Channel Halt mode request is performed when the CFDCnCTR.CHMDC[1:0] bits value is previously 00b (Channel Operation mode).

Additionally, it is possible to force the CAN channel to recover from the bus-off state by setting CFDCnCTR.RTBO to 1b. The error state changes from bus-off state to integrating state with a maximum delay of 1 CAN bit time, and the CAN communication becomes possible again after 11 consecutive recessive bits are detected. The Bus-Off Recovery Flag is not set in this case, and the TEC and REC counters are initialized to 0b.

Before setting CFDCnCTR.RTBO to 1b, all pending transmissions from the TX message buffers, TX Queues and/or Common FIFO in TX or GW mode should be disabled.

The disable of the pending transmission message buffer, TX Queue or FIFO must be confirmed by the corresponding acknowledge flags.

For the TX message buffer, the acknowledge flags are the Transmission Result Flags (CFDTMSTSn.TMTRF[1:0]). For the TX Queue, it is the TX Queue Empty flag (CFDTXQSTSn.TXQEMP). For the FIFO, it is the FIFO Empty flag (CFDCFSTSn.CFEMP).

The CFDCnCTR.RTBO bit should be used for bus-off recovery only when CFDCnCTR.BOM[1:0] is set to 00b.

Setting this bit in any state other than bus-off has no effect and the bit is cleared immediately.

Table 7.9-17 shows the settings for the Bus-Off Entry flag CFDCnERFL.BOEF and the Bus-Off Recovery flag CFDCnERFL.BORF for the different configurations of CFDCnCTR.BOM[1:0].

Table 7.9-17 Behavior of Bus-off Entry and Recovery Flags

BOM[1:0]	BOEF Bit Setting	BORF Bit Setting
00b	Always (on entry to bus-off)	Always (on exit from bus-off)
00b CFDCnCTR.RTBO set to 1b	Always (on entry to bus-off)	Only if normal bus-off recovery occurs before software sets CFDCnCTR.RTBO to 1b
01b	Always (on entry to bus-off)	Never
10b	Always (on entry to bus-off)	Always (on exit from bus-off)
11b	Always (on entry to bus-off)	Only if normal bus-off recovery occurs before software issues a Halt request

For an efficient software procedure, it is not necessary to wait for the bus-off recovery sequence to end.

It is possible to perform a transmission re-initialization during bus-off recovery. To do this, follow the recommended software flow in **Figure 7.9-13**.

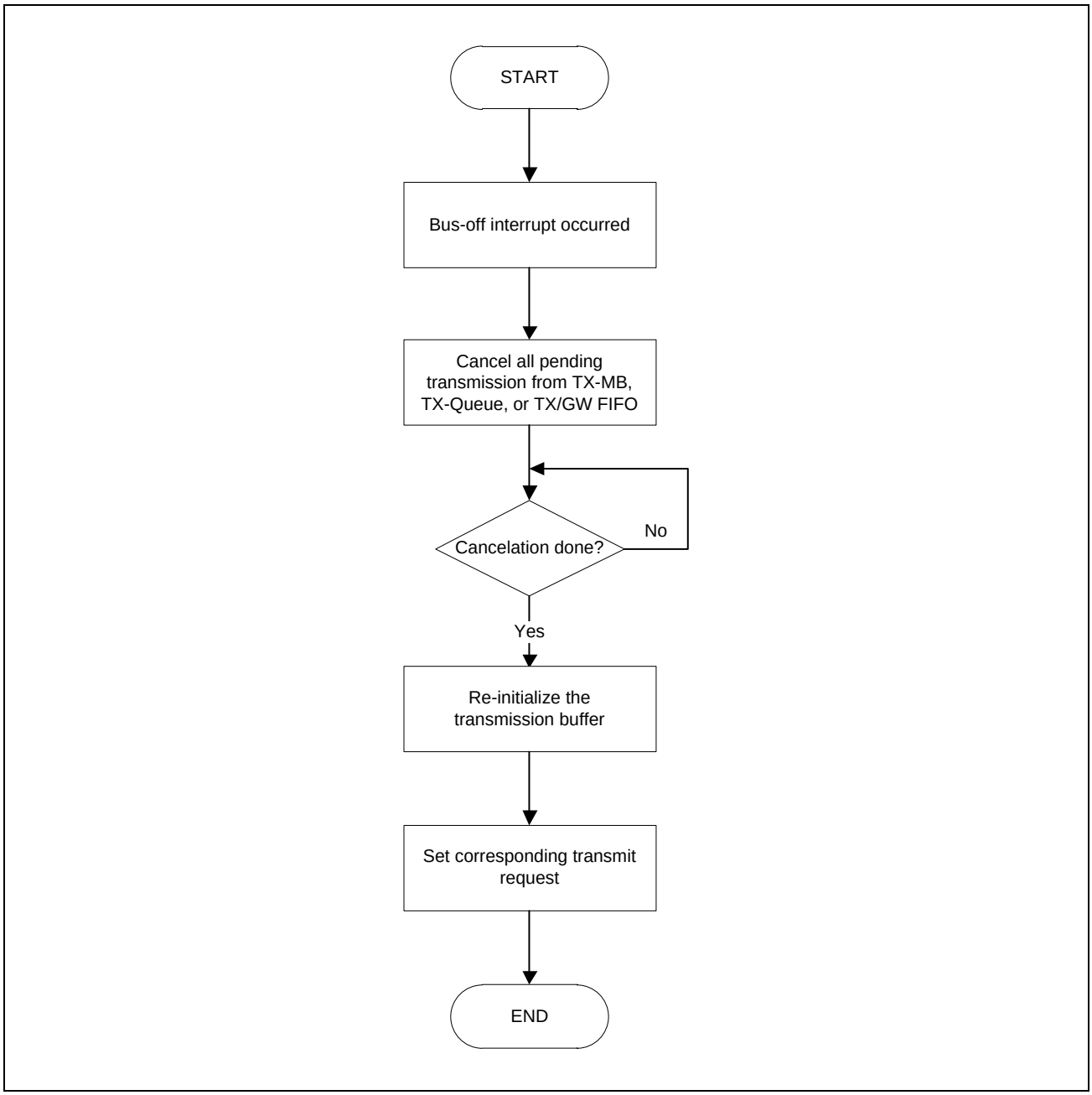


Figure 7.9-13 Transmission Re-Initialization during Bus-Off State

7.9.3.4 Global Mode and Channel Mode Transition Interactions

The interaction between Global mode setting and Channel mode setting is as follows:

- Changing the Channel Mode Control bits CFDCnCTR.CHMDC[1:0] in the Channel Control Registers does not affect the Global Mode Control bits CFDGCTR.GMDC[1:0].
- Changing the Global Mode Control bits CFDGCTR.GMDC[1:0] affects the channel mode control as described in **Table 7.9-18**.

Table 7.9-18 Instruction between Global and Channel Mode Transition

Global Mode Change	Channel Mode	Channel Mode Transition Action
Sleep → Reset	Sleep	Channel remains in Sleep mode
Sleep → Halt	— (Global mode change not possible)	
Sleep → Operation	— (Global mode change not possible)	
Reset → Sleep	Sleep	Channel remains in Sleep mode
	Reset	Channel Sleep request bit is set automatically, channel transits to Sleep mode
Reset → Halt	Sleep	Channel remains in Sleep mode
	Reset	Channel remains in Reset mode
Reset → Operation	Sleep	Channel remains in Sleep mode
	Reset	Channel remains in Reset mode
Halt → Sleep	— (Global mode change not possible)	
Halt → Reset	Sleep	Channel remains in Sleep mode
	Reset	Channel remains in Reset mode
	Halt	Channel mode control is set to Reset mode, channel transits to Reset mode
Halt → Operation	Sleep	Channel remains in Sleep mode
	Reset	Channel remains in Reset mode
	Halt	Channel remains in Halt mode
Operation → Sleep	— (Global mode change not possible)	
Operation → Reset	Sleep	Channel remains in Sleep mode
	Reset	Channel remains in Reset mode
	Halt	Channel mode control is set to Reset mode, channel transits to Reset mode
	Operation	Channel mode control is set to Reset mode, channel transits to Reset mode
Operation → Halt	Sleep	Channel remains in Sleep mode
	Reset	Channel remains in Reset mode
	Halt	Channel remains in Halt mode
	Operation	Channel mode control is set to Halt mode, channel transits to Halt mode after communication finished

7.9.3.4.1 Timing of Global Mode Change

The transition time for the Global mode changes are shown in the following table.

From	To	Maximum Transition Time
GL_SLEEP	GL_RESET	3 peripheral clock cycles*2
GL_RESET	GL_SLEEP	3 peripheral clock cycles
GL_RESET	GL_HALT	10 peripheral clock cycles
GL_RESET	GL_OPERATION	10 peripheral clock cycles
GL_HALT	GL_RESET	2 CAN bit times
GL_HALT	GL_OPERATION	3 peripheral clock cycles
GL_OPERATION	GL_RESET	2 CAN bit times
GL_OPERATION	GL_HALT	3 CAN frames*1,*3

Note 1. The given transition time is the time without any errors on the bus. In case of an error condition, the transition time can lengthen to an uncalculated result. The transition time can also come to a stuck condition for locked RX lines or continued error conditions.

Note 2. Exit GL_SLEEP mode only when CFDGSTS.GRAMINIT is cleared.

Note 3. TQ, CAN frame, and CAN bits are related to the individual channels. For maximum transition time, the channel with the lowest baud rate must be used.

7.9.3.4.2 Timing of Channel Mode Change

The transition time for the Channel mode changes are shown in the following table.

From	To	Maximum transition time
CH_SLEEP	CH_RESET	3 peripheral clock cycles
CH_RESET	CH_SLEEP	3 peripheral clock cycles
CH_RESET	CH_HALT	3 CAN bit times
CH_RESET	CH_OPERATION	4 CAN bit times
CH_HALT	CH_RESET	2 CAN bit times
CH_HALT	CH_OPERATION	4 CAN bit times*2
CH_OPERATION	CH_RESET	2 CAN bit times
CH_OPERATION	CH_HALT	2 CAN frames*1,*3

Note 1. The time specified for this transition does not include the case where channel enters bus-off state. For bus-off, the timing depends on the configuration of the CFDCnCTR.BOM[1:0] bits.

Note 2. The given transition time is the time without any errors on the bus. In case of an error condition, the transition time can lengthen to an uncalculated result. The transition time can also come to a stuck condition for locked RX lines or continued error conditions.

Note 3. In general, if the baud rate prescaler value CFDCnNCFG.NBRP is changed in CH_HALT mode, the transition time can deviate. The internal prescaler is a free running down counter that creates the TQ clock, and new BRP value is captured when the counter reaches the value 0.

7.9.4 Initialization

Before joining CAN communications, configure the following settings:

- Clock setting
- Bit timing setting (nominal and data rate)
- Baud rate setting (nominal and data rate)
- CAN-FD setting
- Acceptance filter setting (configuration of Global Acceptance Filter List)
- Reception, Transmission and GW-FIFO setting
- CAN Operation mode setting

7.9.4.1 Initialization of CAN Clock, Bit Timing and Baud Rate

7.9.4.1.1 Bit Timing Conditions

The following lines describe the composition of each segment and the limitations that apply to the segment setting.

1. Each segment setting SS = Fixed to 1 TQ
   TSEG1 = Refer to (CFDCnNCFG) and (CFDCnDCFG)
   TSEG2 = Refer to (CFDCnNCFG) and (CFDCnDCFG)
   SJW = Refer to (CFDCnNCFG) and (CFDCnDCFG)
   SS + TSEG1 + TSEG2 = 5 to 49 TQs for data bit rate and 8 to 385 for nominal bit rate
2. Limitations on TSEG1, TSEG2 and SJW
   TSEG1 (N) > TSEG2 (N) ≥ SJW (N)
   TSEG1 (D) ≥ TSEG2 (D) ≥ SJW (D)
   When only classical frames are used, configure the bit fields TSEG1 and TSEG2 of CFDCnDCFG to valid values.
   Table 7.9-19 shows an example of how to set the bit timing to achieve the required Sample Point settings.

Table 7.9-19 Transition Behavior in CAN Reset Mode and Halt Mode (1/2)

	Set Value (TQ)				
1 Bit	SS	TSEG1	TSEG2	SJW	Sample Point*1 (%)
5 TQ	1	2	2	1	60.00
8 TQ	1	4	3	1	62.50
	1	5	2	1	75.00
10 TQ	1	6	3	1	70.00
	1	7	2	1	80.00
12 TQ	1	8	3	1	75.00
	1	9	2	1	83.33
15 TQ	1	10	4	1	73.33
	1	11	3	1	80.00
16 TQ	1	10	5	1	68.75
	1	11	4	1	75.00
20 TQ	1	12	7	1	65.00
	1	13	6	1	70.00

Table 7.9-19 Transition Behavior in CAN Reset Mode and Halt Mode (2/2)

1 Bit	Set Value (TQ)				Sample Point*1 (%)
	SS	TSEG1	TSEG2	SJW	
24 TQ	1	15	8	1	66.66
	1	16	7	1	70.83
50 TQ	1	39	10	4	80.00

Note 1. Sample Point (in Case of 75%)

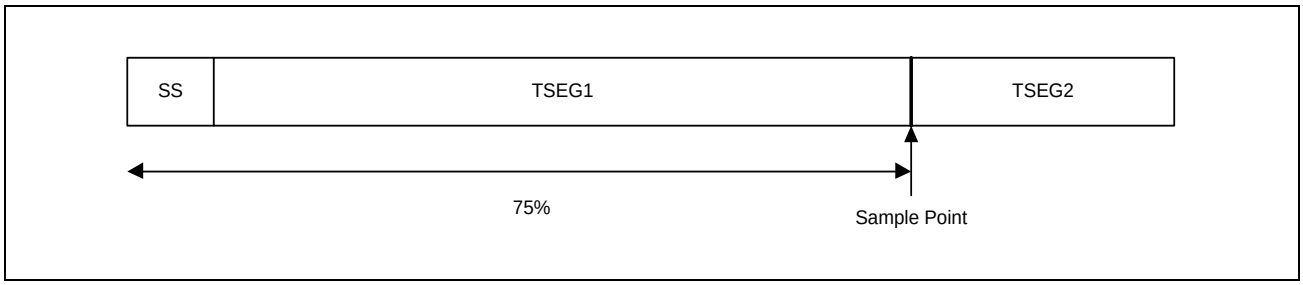


Figure 7.9-14 Sample Point (in Case of 75%)

7.9.4.1.2 CAN Bit Timing

In the CAN protocol, each bit in a communication frame is composed of three segments that can be configured individually for each channel using the related CFDCnNCFG and CFDCnDCFG registers.

Figure 7.9-15 shows the segment composition of a bit and the sample point in it.

Of these segments, the Time Segment 1 (TSEG1) and Time Segment 2 (TSEG2) are used to specify the position of the sample point, so that the timing at which each bit on the CAN bus is sampled can be altered by changing the values of these segments.

The minimum resolution for this timing is referred to as Time Quantum (TQ), which is determined by the clock frequency supplied to the CAN channel and the divide-by-N value of the baud rate prescaler (nominal and data rate).

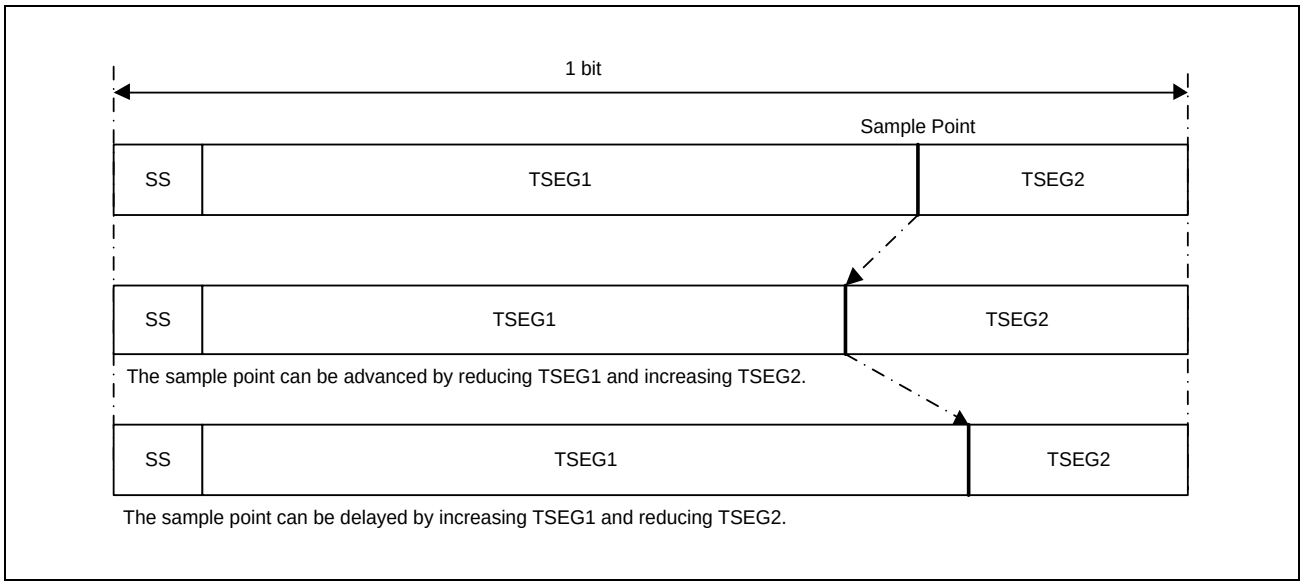


Figure 7.9-15 Segment Composition of a Bit and the Sample Point

1. SS: Synchronization Segment
   This segment is used to synchronize bits by monitoring a recessive-to-dominant edge during the interframe space. This comprises of intermission, suspend transmission, bus idle, during bus idle, and all nodes that can start transmission.
2. TSEG1: Time Segment 1
   This segment absorbs physical delays on the CAN network. A physical delay on the network is two times the total sum of a bus delay, input comparator delay, and output driver delay. It can be lengthened by SJW.
3. TSEG2: Time Segment 2
   This segment is used to correct a phase error by performing resynchronization. It can be shortened by SJW. While sending or receiving a message, communication frames between some nodes may get out of sync due to a drift in the oscillator frequency or a delay in the transmission path. This is referred to as a phase error.
4. SJW: Resynchronization Jump Width
   This is the maximum width by which bits that have become out of sync due to a phase error may be corrected.

Figure 7.9-15 shows only one symbolic sample point.

7.9.4.1.3 Baud Rate

Either CANFD_0_pclk or CANFD_0_clkc can be selected globally for all CAN channels as CAN communication clock.

The transfer speed is determined by the DLL clock, the divide-by-N value of the baud rate prescaler, and the number of TQs in one bit.

$$\text{baud rate} = \frac{\text{DLL_Clock}}{(\text{number_of_time_quanta_per_bit}) \times (\text{BRP} + 1)}$$

Figure 7.9-16 shows a block diagram of the circuit that generates the CAN channel system clock, **Table 7.9-20** and **Table 7.9-21** show baud rate examples.

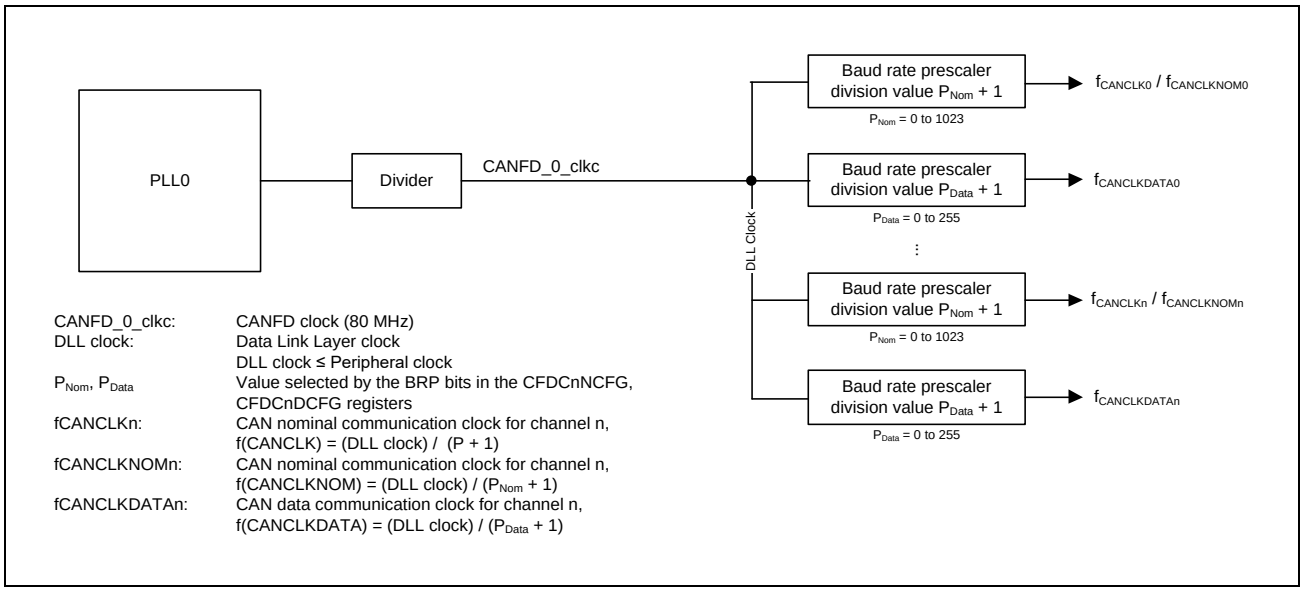


Figure 7.9-16 Block Diagram of the Circuit that Generates the CAN Channel Communication Clock

Table 7.9-20 Nominal Baud Rate Calculation Formula and Example CAN Communication Configurations

Baud Rate Calculation Formula	(DLL Clock) (Baud Rate Prescaler Divide-By-N Value*1) × (Number of TQs in One Bit)			
	80 MHz	40 MHz	100 MHz	75 MHz
1 Mbps	8 TQ (10) 20 TQ (4)	8 TQ (5) 20 TQ (2)	10 TQ (10) 20 TQ (5)	15 TQ (5)
500 Kbps	8 TQ (20) 20 TQ (8)	8 TQ (10) 20 TQ (4)	10 TQ (20) 20 TQ (10)	15 TQ (10)
250 Kbps	8 TQ (40) 20 TQ (16)	8 TQ (20) 20 TQ (8)	10 TQ (40) 20 TQ (20)	15 TQ (20)
125 Kbps	8 TQ (80) 20 TQ (32)	8 TQ (40) 20 TQ (16)	10 TQ (80) 20 TQ (40)	15 TQ (40)
83.3 Kbps	8 TQ (120) 12 TQ (80) 16 TQ (60) 24 TQ (40)	8 TQ (60) 12 TQ (40) 16 TQ (30) 24 TQ (20)	8 TQ (150) 10 TQ (120) 12 TQ (100) 15 TQ (80) 16 TQ (75) 20 TQ (60) 24 TQ (50)	10 TQ (90) 12 TQ (75) 15 TQ (60) 20 TQ (45)
33.3 Kbps	8 TQ (300) 12 TQ (200) 16 TQ (150) 20 TQ (120) 24 TQ (100)	8 TQ (150) 12 TQ (100) 16 TQ (75) 20 TQ (60) 24 TQ (50)	8 TQ (375) 10 TQ (300) 12 TQ (250) 15 TQ (200) 20 TQ (150) 24 TQ (125)	10 TQ (225) 15 TQ (150) 18 TQ (125)

Note: Shown in ( ) are the baud rate prescaler divided-by-N values.

Note 1. Baud rate prescaler divide-by-N value = P + 1 (P = 0 to 1023), P: value selected by the BRP bits in the Channel Configuration Registers.

Table 7.9-21 Baud Rate Calculation Example for Nominal and Data Bit Rate CAN Communication Configurations

Baud Rate Calculation Formula	(DLL Clock) (Baud Rate Prescaler Divide-By-N Value*1) × (Number of TQs in One Bit)			
	80 MHz	40 MHz	100 MHz	75 MHz
Nominal 1 Mbps Data 8 Mbps	80 TQ (1)	40 TQ (1)	100 TQ (1)	75 TQ (1)
	10 TQ (1)	5 TQ (1)	Not possible	Not possible
Nominal 1 Mbps Data 5 Mbps	80 TQ (1)	40 TQ (1)	100 TQ (1)	75 TQ (1)
	16 TQ (1)	8 TQ (1)	20 TQ (1)	15 TQ (1)
Nominal 500 Kbps Data 2 Mbps	160 TQ (1)	80 TQ (1)	200 TQ (1)	150 TQ (1)
	40 TQ (1)	20 TQ (1)	50 TQ (1)	Not possible

Note: Shown in ( ) are the baud rate prescaler divided-by-N values.

Note 1. Baud rate prescaler divide-by-N value = P + 1 (P = 0 to 1023), P: value selected by the BRP bits in the Channel Configuration Registers.

For optimum clock tolerance in networks using the FD frame format, the length of the time quantum should be the same in nominal bit time and in data bit time. This means CFDCnNCFG.NBRP = CFDCnDCFG.DBRP.

Additionally, if transceiver delay compensation is used, do not program the CFDCnDCFG.DBRP[7:0] bits to be greater than 1, as 1 means divide by 2.

7.9.4.1.4 Setting of CAN Clock, Bit Timing and Baud Rate

Figure 7.9-17 shows the procedure for setting the CAN clock and the baud rate for each channel.

These settings should be performed during Channel Reset mode (Configuration mode) for the CAN channels.

Before going to channel communication state, the baud rate must be configured, otherwise the mode does not switch correctly.

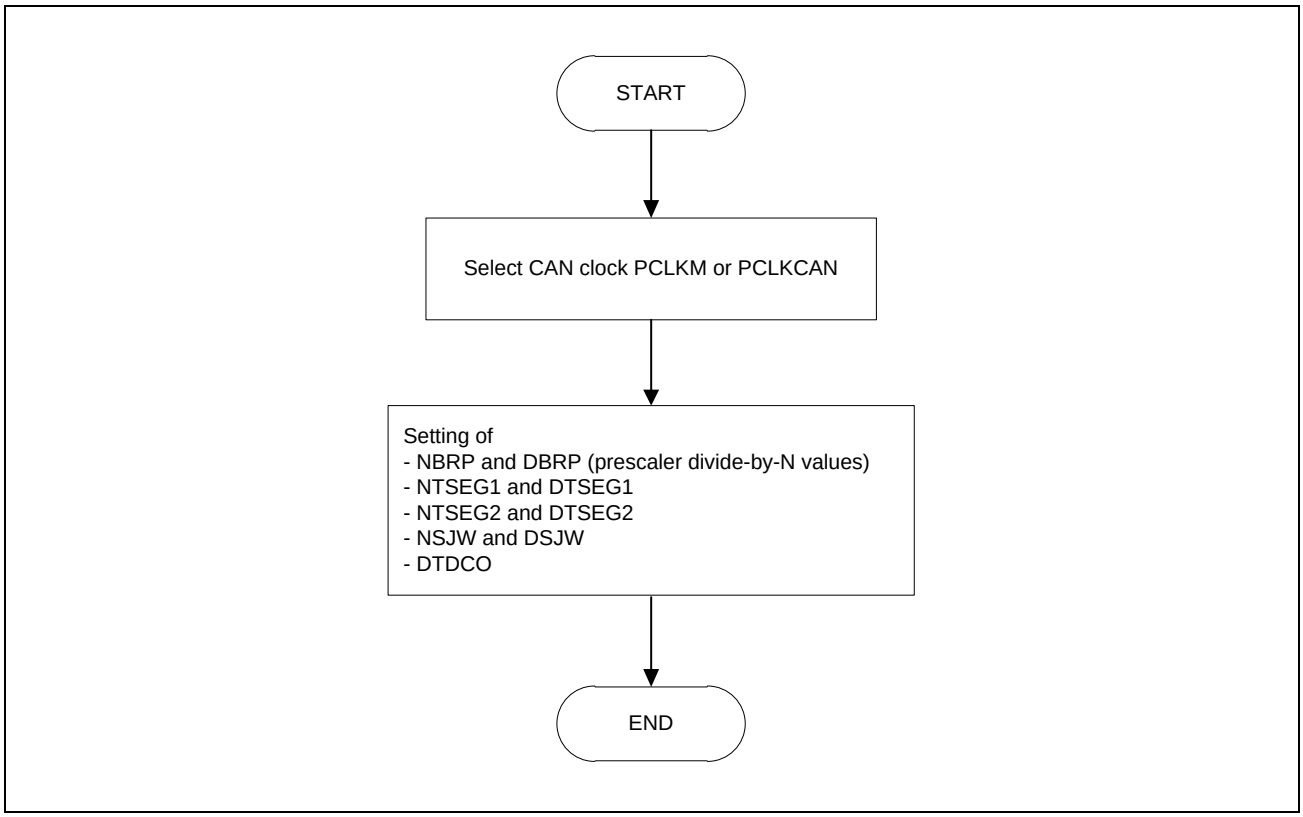


Figure 7.9-17 Procedure for Setting the CAN Bit Timing and Baud Rate

7.9.4.1.5 Transmitter Delay Compensation

When a high baud rate is used such as 5 to 8 Mbps for the data phase, the transmitter delay can become greater than TSEG1. In this case, the transmitter always detects a bit-error in the data phase of the CAN-FD frame. The TDC compensates for the inability of the transmitter to receive its own transmitted bit at the sample point of that bit.

There is another symbolic sample point known as the Secondary Sample Point (SSP) that is used only during the data phase of CAN-FD frames. This is derived from the Transceiver Delay Compensation Result bit (CFDCnFDSTS.TDCR) as shown in **Figure 7.9-18**.

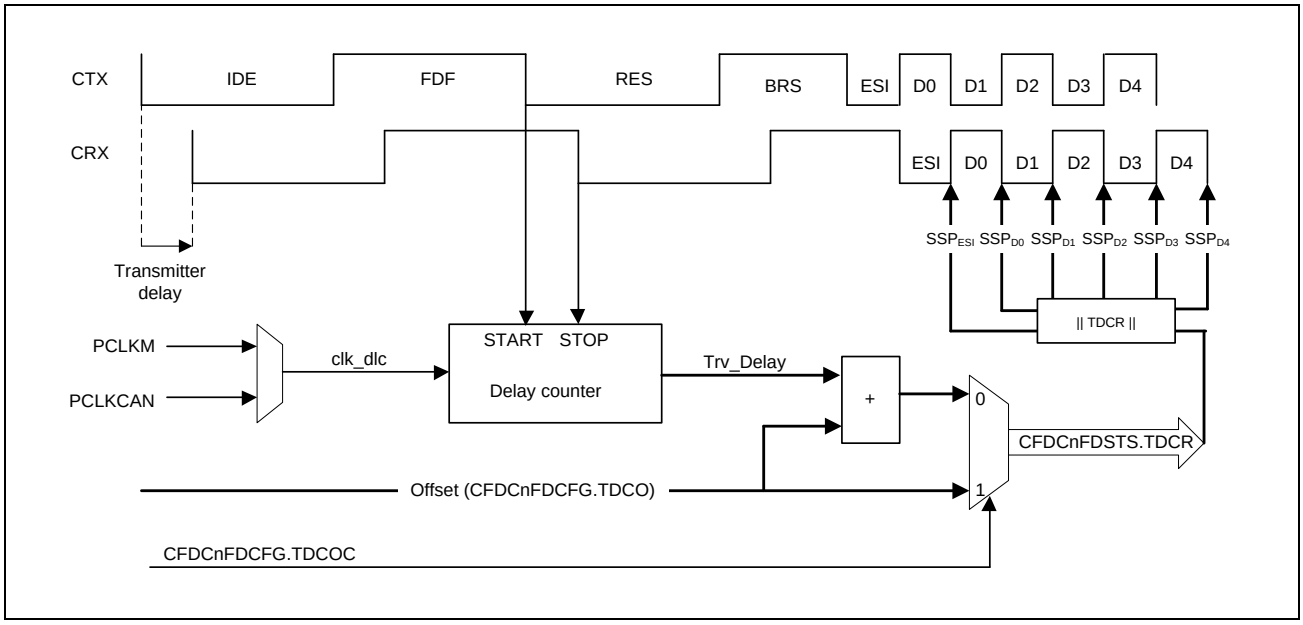


Figure 7.9-18 Transmitter Delay Compensation

The measured Trv_Delay is based on the number of clk_dlc clock cycles. The delay is counted up by 1 for each started clock until the dominant value is seen on CAN_RX. **Figure 7.9-19** shows the measured result. Trv_Delay counted to maximum 127 with a clk_dlc clock.

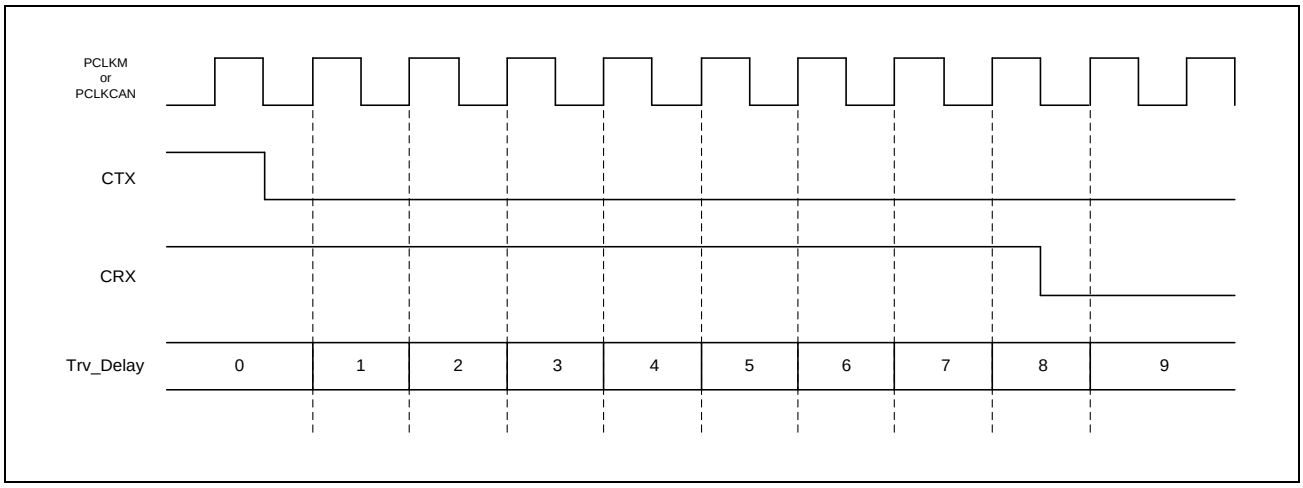


Figure 7.9-19 Trv_Delay Measurement Example

The SSP is calculated by taking the result from CFDCnFDSTS.TDCR and rounding the value down to the nearest integer number of data time quanta.

Figure 7.9-20 shows the positioning of the secondary sample point. When CFDCnFDCFG.TDCOC is equal to 0, the SSP is equal to the Trv_Delay (measured delay) + CFDCnFDCFG.TDCO, rounded down to the nearest integer number of time quanta. Usually, the TDCO value should have the size of (Sync Segment data + TSEG1 data) to position the SSP to a theoretical location of the sample point.

If the CFDCnFDCFG.TDCOC is equal to 1, the SSP is defined by CFDCnFDCFG.TDCO. If CFDCnDCFG.DBRP[7:0] is greater than 0, the value is also rounded down to the nearest integer number of time quanta.

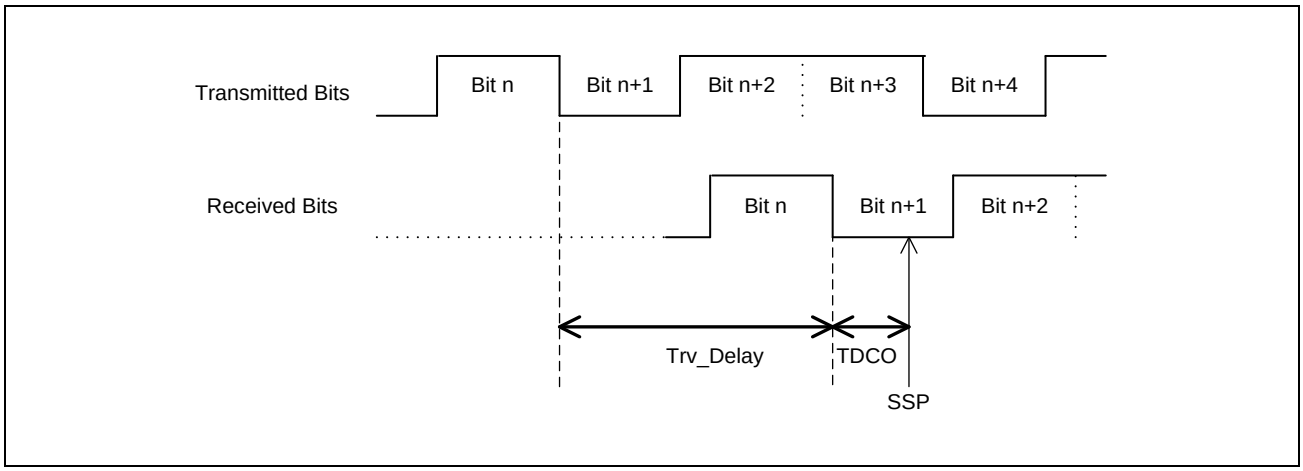


Figure 7.9-20 Position of the Secondary Sample Point

The maximum delay (Trv_Delay + TDCO) which can be compensated by the CAN-FD module is (6 data bits - 2clk_dlc).

The ISO 11898-1 allows you to set different values for BRP_data and BRP_nom.

If different values are used for CFDCnNCFG.NBRP and CFDCnDCFG.DBRP, then two CAN nodes may be out of synchronization at the point when the bit rate changes from nominal bit rate to data bit rate after sample point of the BRS bit. This condition is shown in **Figure 7.9-21**.

The length of the time quantum should be the same in the nominal bit time and in the data bit time. This means CFDCnNCFG.NBRP = CFDCnDCFG.DBRP.

Different bit rates can be achieved by selecting different configuration values for the Time Segments. The nominal bit rate can be configured from 8 to 385 TQs and the data bit rate from 5 to 49 TQs.

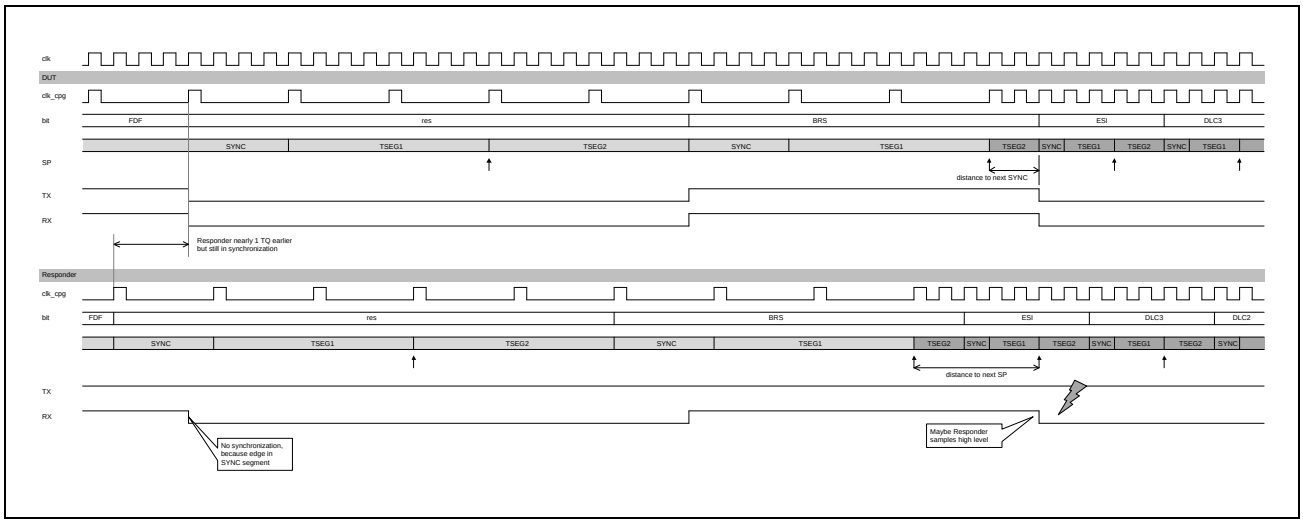


Figure 7.9-21 Loss of Synchronization between 2 CAN Nodes

The transmitter delay compensation measurement result is updated at the falling edge from FDF bit to RES bit when configured accordingly (CFDCnFDCFG.TDCE = 1b, CFDCnFDCFG.TDCOC = 0b).

Figure 7.9-22 shows the read flow to get the measured transmitter delay compensation result.

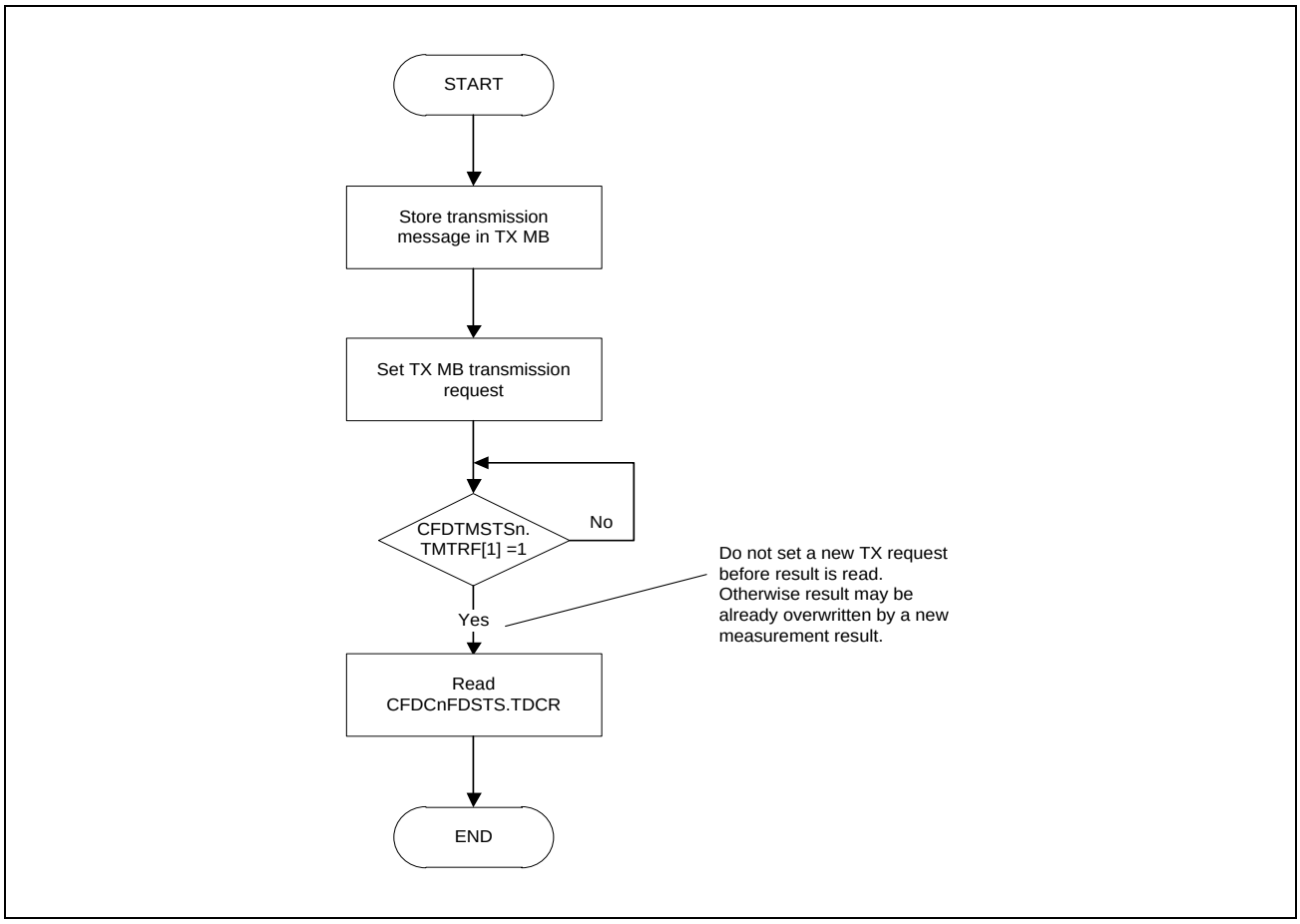


Figure 7.9-22 TDC Result Read Flow

7.9.4.1.6 CAN Module Configuration after Hardware Reset

After a hardware reset (power on reset) or after setting and clearing the CFDGRSTC.SRST bit, the CAN-FD module enters Global Sleep mode automatically.

To enable configuration of the CAN-FD module, you must exit Sleep mode by clearing the Global Sleep Request bit CFDGCTR.GSLPR to 0b.

After a hardware reset, the module starts RAM initialization, the CFDGSTS.GRAMINIT bit in the Global Status Register is set automatically to indicate that the CAN-FD logic is initializing the RAM.

After RAM initialization is complete, this bit is cleared automatically.

RAM initialization is necessary to avoid setting of false ECC error flag after hardware resets the random data presented in the RAM.

Do not access registers of the CAN-FD in either read or write until RAM initialization is complete and the CFDGSTS.GRAMINIT bit is cleared.

Before going to communication mode, the Global Acceptance Filter List and message FIFO buffers must be configured. In addition, each required CAN channel must be configured such as CAN bit timing. For this configuration, all required CAN channels must be released from Channel Sleep mode and must be configured for communication in Channel Reset mode (Configuration mode).

For this, all required CAN channels must be released from Channel Sleep mode and must be configured for communication in Channel Reset mode (Configuration mode).

Figure 7.9-23 shows the configuration procedure. For details about each step, see **7.9.5 Acceptance Filtering Function using Global Acceptance Filter List (AFL)**, **7.9.6 FIFO Buffers and Normal Message Buffer Configuration**, **7.9.7 Interrupt and DMA**, and **7.9.4.1.3 Baud Rate**.

The CAN-FD module does not perform RAM initialization sequence after executing a software reset by setting CFDGRSTC.SRST.

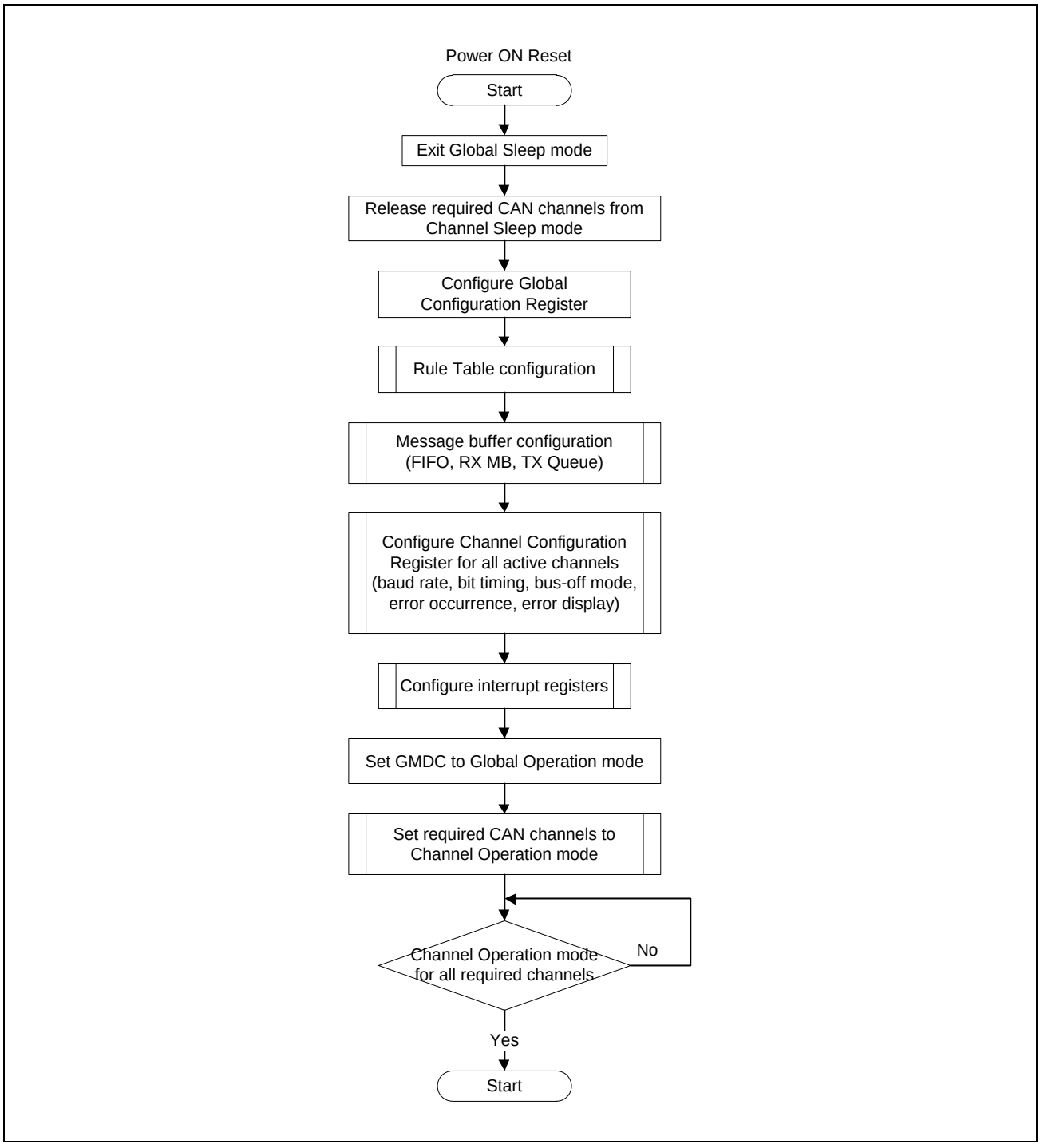


Figure 7.9-23 Configuration Procedure after a Hardware Reset

7.9.5 Acceptance Filtering Function using Global Acceptance Filter List (AFL)

7.9.5.1 Overview

The CAN-FD module can handle message acceptance filtering for all channels with a global Acceptance Filter List (AFL). Each element of the AFL defines a filter rule for messages received on a specific channel.

The following actions are performed based on the AFL entries:

- Acceptance filtering based on received CAN identifier and masking
- DLC filtering based on received DLC value
- Message data payload according to the CFDGCFG.CMPOC bit
- Storage of accepted messages in the message buffer objects defined in the related AFL entry
- Attaching a 16-bit pointer to the stored messages defined in the related AFL entry, for example to support AUTOSAR applications
- Attaching a 2-bit information label to the stored messages defined in the related AFL entry

The 6-channel CAN-FD module allows a maximum of 768 AFL entries across all channels with a maximum of 128 AFL entries per channel.

During the acceptance filtering process, each AFL entry in a channel is checked against the received message by the acceptance filter unit. The check starts from the lowest AFL entry number for this channel.

The AFL search stops when a match of the received identifier with a configured identifier/mask combination occurs or when the received identifier has been compared against all AFL entries defined for the related channel. If no match occurs, then the received message is rejected. No notification is given to the application in this case.

Additionally, an automatic DLC filtering is performed for each accepted message if DLC check is globally enabled. If the DLC value of the received message is equal to or higher than the configured DLC value in the matching AFL entry, the DLC check is passed.

If DLC replacement (CFDGCFG.DRE bit) is enabled, DLC value configured in the matching AFL entry is greater than 0h and DLC check passes, then the configured value of DLC in the matching AFL entry is stored in the destination RXMB or FIFO Buffer.

If the received value of DLC is greater than the configured DLC value in the matching AFL entry, then the additional data bytes received on the CAN bus are not stored in the destination RXMB or FIFO Buffer. These additional data bytes are stored as 00h in the destination RXMB or FIFO Buffer.

If DLC replacement is enabled and DLC value of matching AFL entry is 0h, then the received value of DLC is stored in the destination RX MB or FIFO Buffer.

If DLC replacement (the CFDGCFG.DRE bit) is disabled and DLC check passes, then the received value of DLC on the CAN bus is stored in the destination RXMB or FIFO buffer.

If the received value of DLC is greater than the configured DLC value in the matching AFL entry, then the additional data bytes received from the CAN bus are also stored in the destination RXMB or FIFO buffer.

If DLC value of the received message is less than the configured DLC value in the matching AFL entry, then DLC check fails. In this case, the received message is rejected and is not stored in any RXMB or FIFO buffer.

Additionally, DLC check failure is flagged by the DLC Error Flag in the Global Error Flag Register. If configured, an error interrupt is also generated. The DLC replacement configuration has no impact if the DLC check fails.

If a message has passed both acceptance filtering and DLC filtering, it is stored in a single reception message buffer and/or in FIFO buffers configured for reception or gateway function.

This message storage target information is also defined in the same AFL entry. Do not set a target at the AFL entry which is not configured.

Each accepted received message can be stored into a maximum of 8 different target destinations (single reception message buffer and/or FIFO buffers).

The programming of more than eight target destinations is not allowed. When more destinations are programmed then internal timing, a race condition can occur and received message may not be stored to the message RAM. Correct configuration of the numbers of target destination is the responsibility of the application.

Additional protection mechanism is made for the case when a received message contains more data payload bytes than possible to store in the target destination (CFDRMNB.RMPLS[2:0], CFDRFCCn.RFPLS[2:0] or CFDCFCCn.CFPLS[2:0]).

If CFDGCFG.CMPOC = 0b, the message is completely rejected and is stored in the target destination. When CFDGCFG.CMPOC = 0b and RX or Common FIFO full including the received message contains more data payload bytes than possible to store in the target destination (CFDRMNB.RMPLS[2:0], CFDRFCCn.RFPLS[2:0] or CFDCFCCn.CFPLS[2:0]), the corresponding CFDFMSTS.RFXMLT[7:0] or CFDFMSTS.CFXMLT[17:0] bits are not set to 1b, respectively.

When CFDGCFG.CMPOC = 1b, the received data bytes greater than CFDRMNB.RMPLS is rejected. When CFDGCFG.CMPOC = 1b and RX or Common FIFO full including the received message contains more data payload bytes than possible to store in the target destination (CFDRMNB.RMPLS[2:0], CFDRFCCn.RFPLS[2:0] or CFDCFCCn.CFPLS[2:0]), the corresponding CFDFMSTS.RFXMLT[7:0] or CFDFMSTS.CFXMLT[17:0] bits are set to 1b, respectively.

Depending on the CFDGCFG.DRE bit, the original received DLC or the DLC value configured at the AFL entry is stored.

Regardless of the CFDGCFG.CMPOC bit setting, CFDGERFL.CMPOF is set to 1b if a payload overflow condition is detected.

The DLC filtering is performed before the payload overflow function. So for one reception frame, only one flag can be set at the same time with CFDGERFL.DEF or CFDGERFL.CMPOF.

7.9.5.2 Allocation of AFL Entries to Each CAN Channel

The number of AFL entries per channel can be configured using the dedicated field in the related Global Acceptance Filter Configuration Registers (see **Figure 7.9-24**).

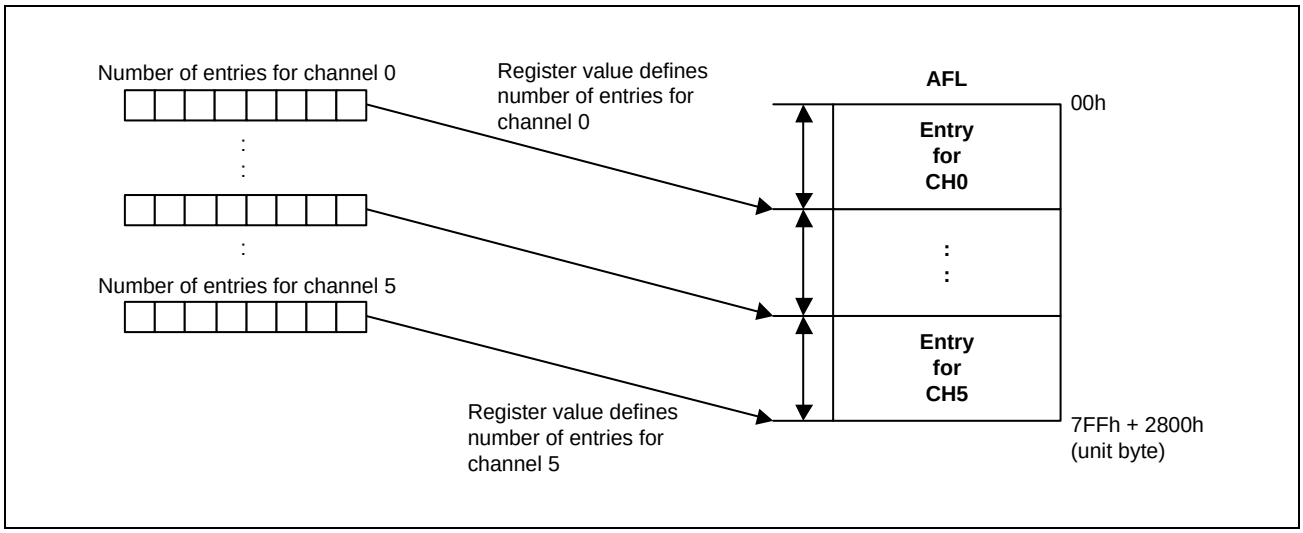


Figure 7.9-24 Configuration of AFL for Each Channel

The minimum number of entries for one channel is 0 (no entries defined for the channel) and the maximum number of entries for one channel is 128. The total number of entries for all channels should not exceed the maximum limit of (n + 1) × 128.

All entries are unique for a channel and overlapping or sharing of entries is not supported. Correct configuration of the AFL is the responsibility of the application.

The CAN-FD module does not flag errors related to the configuration of the AFL.

7.9.5.3 AFL Entry Description

Each AFL entry consists of 16 bytes. The fields in all entries are identical.

Each entry contains the following information for acceptance filtering and DLC filtering:

- Identifier (11 bits for Standard Frame format, 29 bits for Extended Frame format):
  Acceptance filter unit checks the identifier field of the received message against the identifier field of each AFL entry (full 29 bits masking of identifier bits is possible, see information that follows).
- IDE bit:
  Acceptance filter unit checks the IDE bit of the received message against this bit and selects the relevant part of the identifier field for acceptance filtering (masking of IDE bit is possible, see the information that follows).
- RTR bit:
  Acceptance filter unit only accepts data frames (RTR = 0b) or remote frames (RTR = 1b) according to the setting of this bit (masking of RTR bit is possible, see the information that follows).
- Loopback Configuration bit:
  This bit can enable or disable the AFL entry depending on the Loopback Configuration or Mirror mode condition.
- Mask for Identifier bits (29 bits):
  Each bit in the identifier mask can mask the corresponding identifier bit in the AFL entry during acceptance filtering, see **Figure 7.9-25**.
- Mask for IDE bit:
  If this Mask bit masks the IDE bit of the AFL entry in both Standard Identifier and Extended Identifier format, messages can be accepted by this AFL entry. The identifier of the received message is compared against the Standard Identifier part of the AFL entry for Standard Identifier format messages and against the Extended Identifier part of the AFL entry for Extended Identifier format messages.
- Mask for RTR bit:
  If this Mask bit masks the RTR bit of the AFL entry in both frame formats, data frame and remote frame formats are accepted by this AFL entry.
- Pointer information (16 bits):
  This 16-bit pointer is attached to a received message accepted by the related AFL entry. The pointer is added during message storage in the message buffer area and can be used by application as support function. The pointer information can be used for example to support PDU identifier allocation for the received message in AUTOSAR systems.
- Information label (2 bits):
  This 2-bit label is attached to a received message accepted by the related AFL entry. The label is added during message storage in the message buffer area and can be used by application as support function.
- DLC value for automatic DLC filtering:
  If the DLC value of the received message is equal or higher than the configured DLC value, the DLC check is passed. If the DLC value in this AFL entry is configured to 0b, DLC filtering is effectively disabled for this entry (all accepted messages pass DLC filtering).

Each AFL entry contains the following information for the handling of received messages:

- Message buffer number of one single reception message buffer as target for received message storage
- Single reception message buffer enable bit to configure the single reception message buffer number to be valid or invalid, as target for received message storage
- FIFO direction pointer - each bit of the FIFO direction pointer configures a dedicated FIFO as possible target for a received message

Note: A message received on channel A can be routed to Common FIFO buffer of another channel. If this Common FIFO buffer is configured in Gateway mode, then the message stored in this Common FIFO Buffer is transmitted on that channel because Common FIFO buffer is associated with channel.

There is no hardware protection against such storage of message. Therefore, the FIFO direction pointer must be configured carefully.

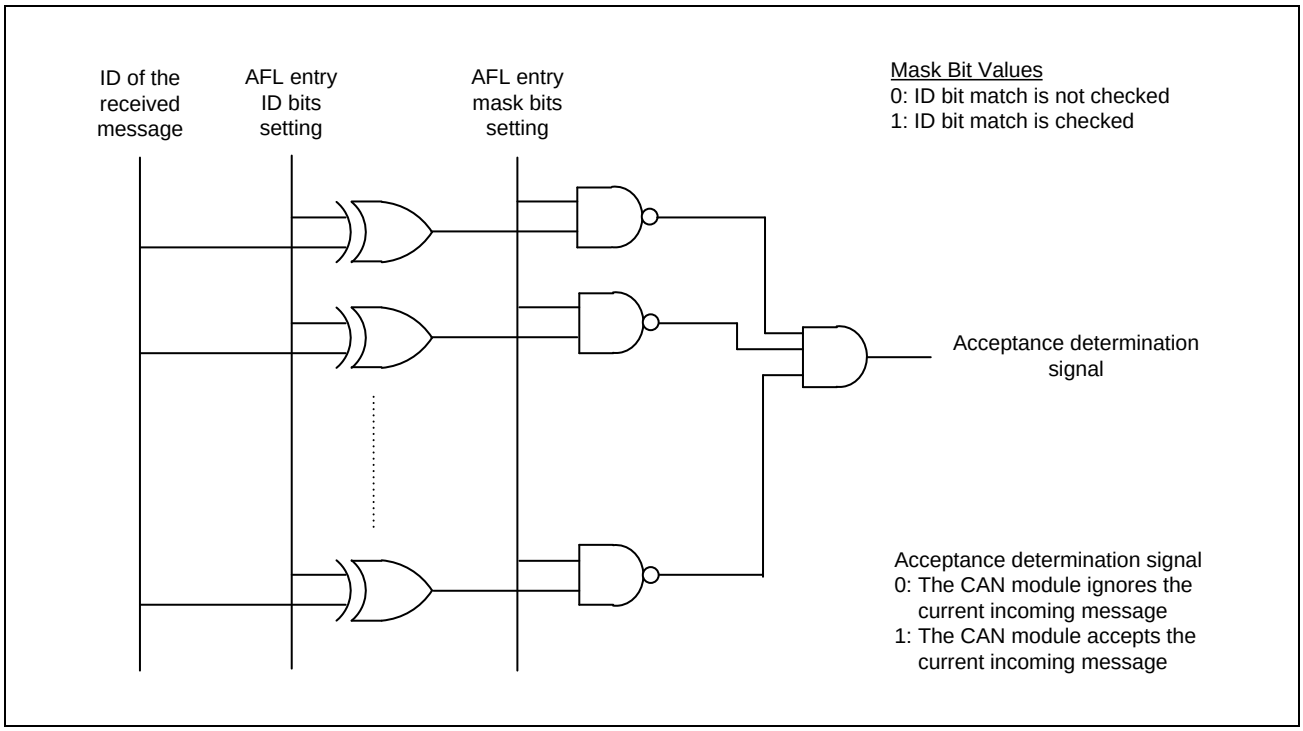


Figure 7.9-25 Acceptance Function

7.9.5.4 Entering Entries in the AFL

Application software can enter one full entry into the AFL using the following registers:

- Global AFL ID Entry Register: Part 1 of the AFL entry
- Global AFL Mask Entry Register: Part 2 of the AFL entry
- Global AFL Pointer 0 Entry Register: Part 3 of the AFL entry
- Global AFL Pointer 1 Entry Register: Part 4 of the AFL entry

16 sets of these registers form a group of AFL entries. Each group can be accessed through a page mechanism. For the CAN-FD module, 6-channel version, 48 of these pages exist to allow access to the whole AFL range. The AFL should only be configured in CH_RESET or CH_HALT mode. Pages are linked to the AFL entries in the following way:

Table 7.9-22 Mapping between Pages and AFL Entries

Page 0	Entry 0 - 15
Page 1	Entry 16 - 31
Page 2	Entry 32 - 47
Page 3	Entry 48 - 63
Page 4	Entry 64 - 79
Page 5	Entry 78 - 95
Page 6	Entry 96 - 111
Page 7	Entry 112 - 127
Page 8	Entry 128 - 143
Page 9	Entry 144 - 159
Page 10	Entry 160 - 175
Page 11	Entry 176 - 191
Page 12	Entry 192 - 207
Page 13	Entry 208 - 223
Page 14	Entry 224 - 239
Page 15	Entry 240 - 255
Page 16	Entry 256 - 271
Page 17	Entry 272 - 287
Page 18	Entry 288 - 303
Page 19	Entry 304 - 319
Page 20	Entry 320 - 335
Page 21	Entry 336 - 351
Page 22	Entry 352 - 367
Page 23	Entry 368 - 383
Page 24	Entry 384 - 399
Page 25	Entry 400 - 415
Page 26	Entry 416 - 431
Page 27	Entry 432 - 447
Page 28	Entry 448 - 463
Page 29	Entry 464 - 479
Page 30	Entry 480 - 495
Page 31	Entry 496 - 511
Page 32	Entry 512 - 527

Page 33	Entry 528 - 543
Page 34	Entry 544 - 559
Page 35	Entry 560 - 575
Page 36	Entry 576 - 591
Page 37	Entry 592 - 607
Page 38	Entry 608 - 623
Page 39	Entry 624 - 639
Page 40	Entry 640 - 655
Page 41	Entry 656 - 671
Page 42	Entry 672 - 687
Page 43	Entry 688 - 703
Page 44	Entry 704 - 719
Page 45	Entry 720 - 735
Page 46	Entry 736 - 751
Page 47	Entry 752 - 767

The selection of the AFL access page is done using the Global Acceptance Filter List Entry Control Register (CFDGAFLECTR) (see **Figure 7.9-26**). This register has the following fields:

- 4 bits to select the AFL page number
- 1 bit to enable or disable the AFL data access to prevent unwanted write access to the AFL

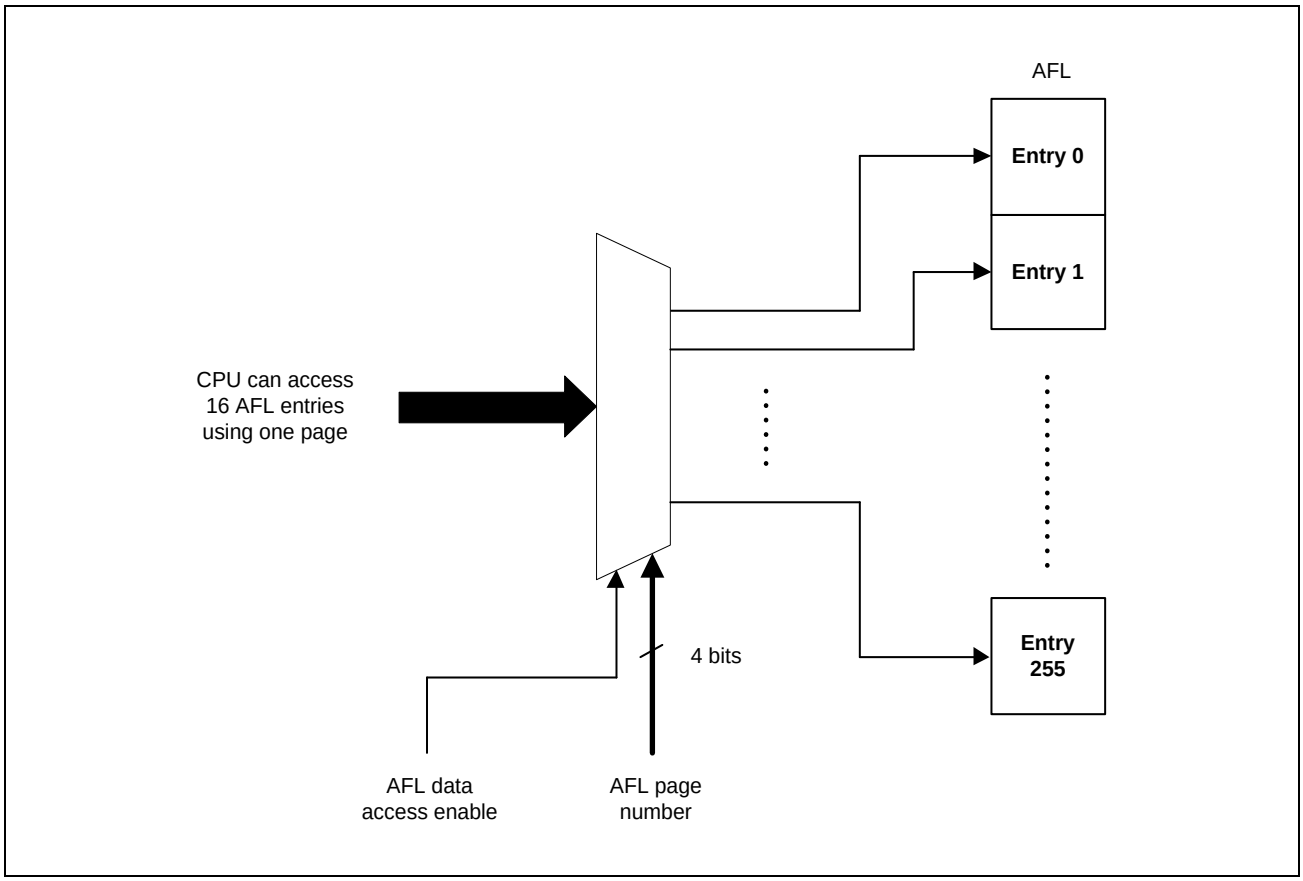


Figure 7.9-26 AFL Page Access

Application software should not write numbers higher than Fh for the AFL page number.

Follow the configuration shown in **Figure 7.9-27** to program the AFL.

After entering all entries in Configuration mode, locking of the AFL access should be performed to protect unwanted write access to the AFL.

Write protection is active during all Global modes (GL_RESET, GL_HALT, and GL_OPERATION) if the lock bit is set.

Read access to AFL is still possible during all Global modes even when AFL data access is disabled (consistency check of AFL contents is possible during run time).

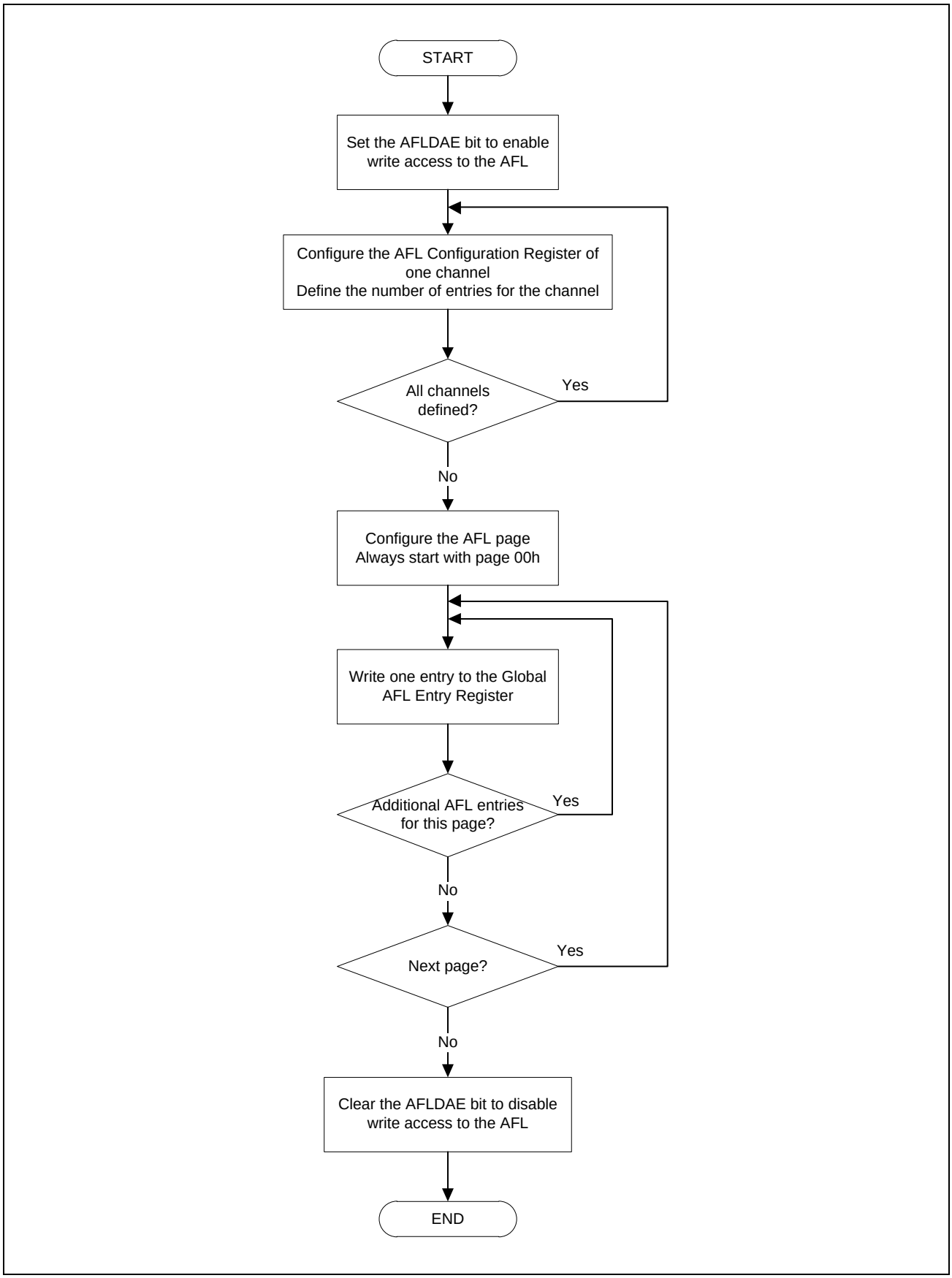


Figure 7.9-27 AFL Configuration Flow

7.9.5.5 Loopback Modes

If the Loopback Configuration bit is set, the AFL entry is only valid in loopback test mode (Self test mode 0 or Self test mode 1) or in mirror mode when receiving messages that were transmitted by the respective CAN channel itself.

The AFL entry is not valid for received messages in loopback mode transmitted by other CAN nodes on the bus. The expression valid or invalid for the related entry means that this AFL entry is or is not compared against the received message ID respectively.

If the Loopback Configuration bit is 0b, the AFL entry is only valid for:

- Received messages transmitted by other CAN nodes on the bus in normal (non-loopback mode) and mirror modes
- Received messages transmitted by other CAN nodes or the CAN channel itself in loopback test mode.

The mirror mode can be enabled with the CFDGCFG.MME bit in the Global Configuration Register. If CFDGCFG.MME bit is set, then a successfully transmitted message can be stored back in an RX message buffer or FIFO buffer if a matching entry is configured in the AFL for that channel.

The Loopback Configuration bit in the matching AFL entry must be set to store this frame.

If mirror mode and loopback test mode are configured at the same time, the loopback test mode behavior applies.

Table 7.9-23 shows the behavior of the acceptance filter unit depending on the setting of the related input signals.

Table 7.9-23 Behavior of Acceptance Filter Based on the Loopback Configuration Setting in AFL Entry

Mirror Mode Enable (MME Configuration Bit)	Loopback in Test Mode (Self test mode 0 or Self test mode 1)	Channel Mode	Loopback Configuration Bit in AFL Entry	AFL Entry
0	0	Receiver	0	Valid
			1	Invalid
		Transmitter	0	Invalid
			1	Invalid
	1	Receiver	0	Valid
			1	Invalid
		Transmitter	0	Valid
			1	Valid
1	0	Receiver	0	Valid
			1	Invalid
		Transmitter	0	Invalid
			1	Valid
	1	Receiver	0	Valid
			1	Invalid
		Transmitter	0	Valid
			1	Valid

Note: The expression valid or invalid for the related entry means that this AFL entry is or is not compared against the received message ID, respectively.

7.9.5.6 IDE Masking

When the GAFLIDEM bit is 0b in an AFL entry, the IDE bit configured in the AFL entry is not used for ID matching. In this case, the use of ID[10:0] or ID[28:0] matching is based on the received IDE bit.

Consider the following example:

- The ID and Mask fields of an AFL entry x is configured as follows:
  - CFDGAFLID [x] = C055_3A20h → IDE = 1b, RTR = 1b, LLB = 0b, ID[10:0] = 220h/ID[28:0] = 055_3A20h
  - CFDGAFLMn = 0000_FFFFh → IDEM = 0b, RTRM = 0b, IDM[10:0] = 7FFh/IDM[28:0] = 0000_FFFFh
- The comparison result for the four different received IDs with AFL entry x is described as follows:
  - If a frame with IDE = 0b and ID = 220h is received, this is considered as a match
  - If a frame with IDE = 0b and ID = 320h is received, this is not a match
  - If a frame with IDE = 1b and ID = 1FFF_3A20h is received, this is considered as a match
  - If a frame with IDE = 1b and ID = 0880_3220h is received, this is not a match.

7.9.6 FIFO Buffers and Normal Message Buffer Configuration

This section describes the process for configuring the number of RX message buffers, the FIFO buffers, and the flat TX message buffers in the CAN-FD module. The message buffers are mapped as shown in **Figure 7.9-28**.

The RX message buffers can be accessed with the RX Message Buffer Registers.

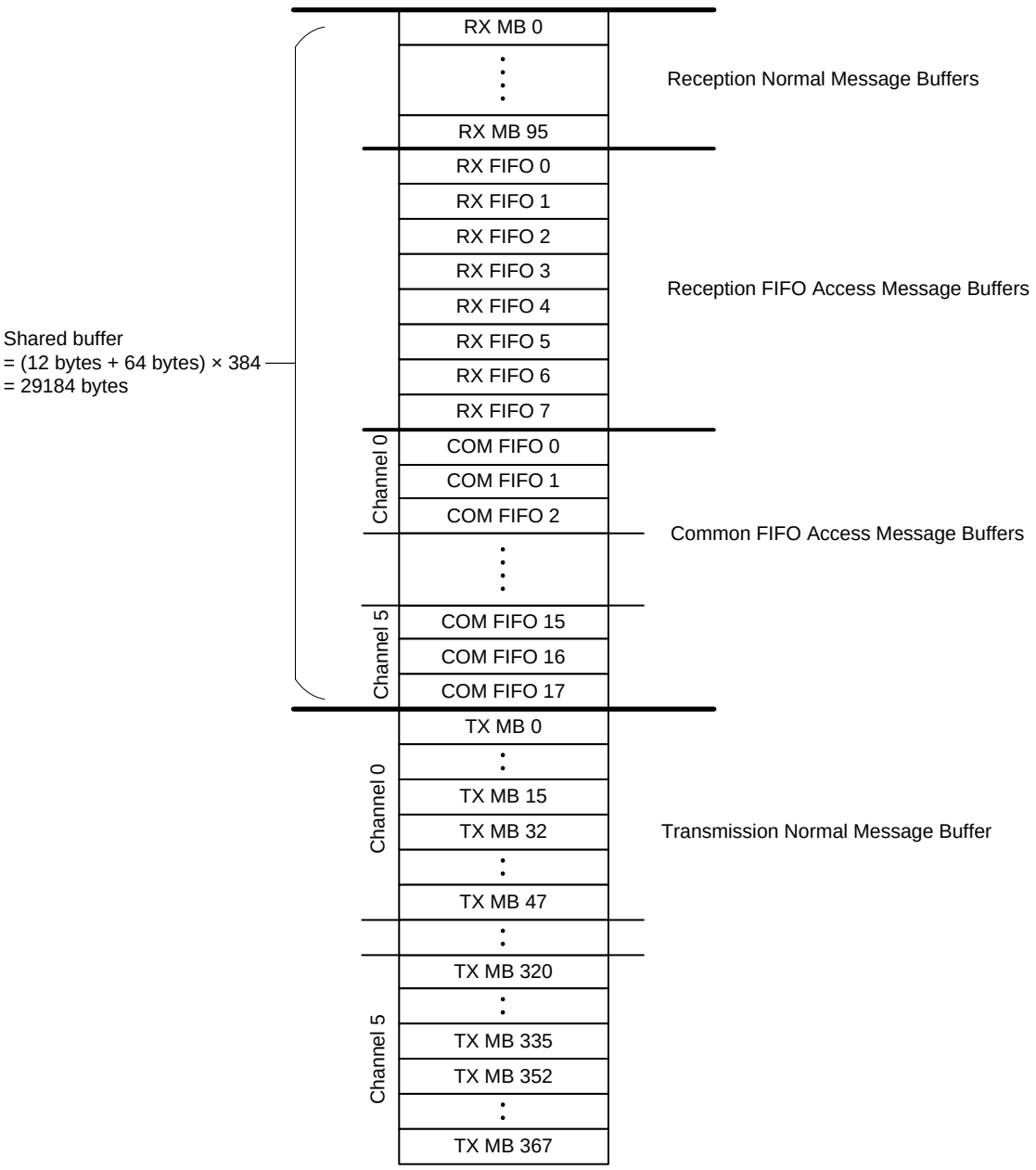
The RX FIFO buffers and the common FIFO buffers configured in RX mode, TX mode, or GW mode can only be accessed with the FIFO Access Registers.

If the common FIFO is configured in TX mode, you can only write data into the FIFO buffer using the FIFO Access registers.

If the common FIFO is configured in GW mode or RX mode, you can only read data from the FIFO Access Registers.

The TX message buffers can be accessed with the TX Message Buffer Registers.

If unused message buffer locations are read, the message buffer locations are read as unknown values.



Note: The total number of bytes in RXMB_cfg, RXFIFO_cfg, and CFIFO_cfg is less than or equal to 29184 bytes.
RXMB_cfg = (12bytes + CFDRMNB.RMPLS) × CFDRMNB.NRXMB
RXFIFO_cfg = SUM((12 bytes + CFDRFCCn.RFPLS) × CFDRFCCn.RFDC)
CFIFO_cfg = SUM((12 bytes + CFDCFCCn.CFPLS) × CFDCFCCn.CFDC)

Figure 7.9-28 Message Buffer Configuration

7.9.6.1 Normal RX Message Buffers

In CAN-FD module, the frames received by various channels can be stored in normal RX message buffers based on the configuration of the AFL entries.

Additionally, the number of normal RX message buffers required in the system can be chosen up to a fixed maximum limit.

7.9.6.1.1 Normal RX Message Buffer Configuration

In CAN-FD module, the number of normal RX message buffers can be configured by writing to the RX Message Buffer Number Register.

The limiting values for the configuration of number of message buffers are:

- Minimum value = 00h (no normal RX MB)
- Maximum value = (16 × number of CAN channels) = 16 × 6 = 96 = 60h (96 flat RX MBs for 6 channels)

Do not use values outside these limits.

The AFL entries for routing the received messages to normal RX message buffers must be configured to match the requirements of the system.

The AFL entries must also be configured properly, and an AFL entry for normal RX message buffers should not exceed the number of message buffers configured in the RX Message Buffer Number Register.

Note: There is no internal check procedure provided in CAN-FD module against incorrect configuration of the AFL.

The data field size of the RX message buffer can be configured with the CFDRMNB.RMPLS[2:0] bits. The default size is 8 bytes and the maximum data payload size is 64 bytes.

When the receiving frame exceeds the data field size, then the acceptance depends on the configuration of CFDGCFG.CMPOC (message rejecting or data payload cut).

7.9.6.2 FIFO Buffer

The CAN-FD module provides a fixed number of FIFO buffers to support storage of frames for reception, transmission and gateway functions for various CAN channels.

The number of reception-only FIFO buffers is fixed to 8. However, 3 common FIFO buffers per channel can be configured to store messages for transmission, reception, or gateway function.

These FIFO buffers can be enabled or disabled, and the following parameters can be configured to match the system requirements:

- Size
- Interrupt structure
- Message lost mechanism
- Message overwrite mechanism of the FIFO buffers
- Location of the TX FIFO or GW FIFO

When the receiving frame exceeds the data field size, the acceptance depends on the configuration of the CFDGCFG.CMPOC bit (message rejecting or data payload cut).

7.9.6.2.1 FIFO Buffers Configuration

In CAN-FD module, the FIFO buffers can be configured to match the system requirements.

The total number of FIFO buffers = 8 RX FIFO buffers + 18 common FIFO buffers = 26 FIFO buffers for 6 channels and message overwrite mechanism.

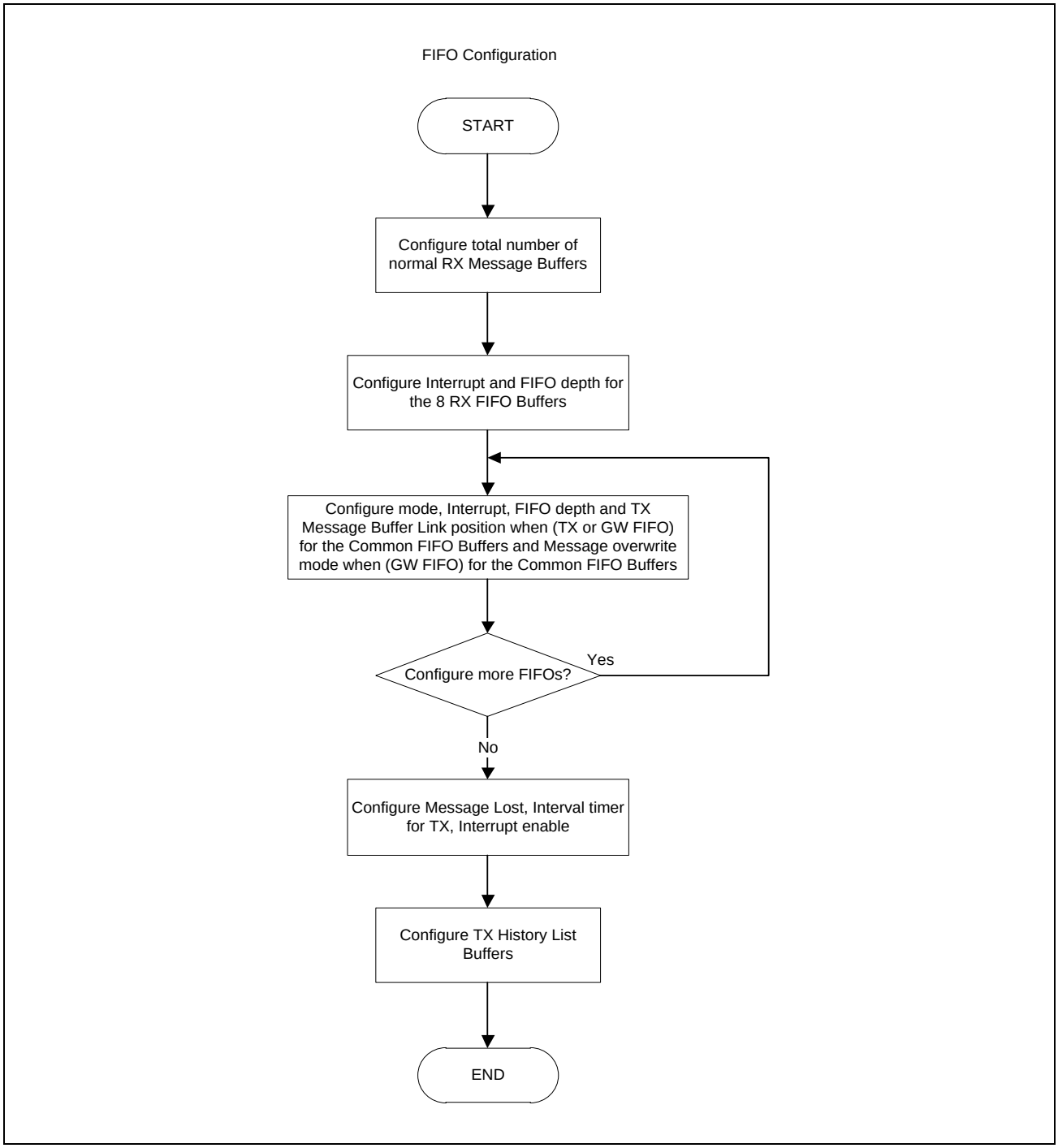


Figure 7.9-29 FIFO Buffer Configuration Flow in CAN-FD Module

As shown in **Figure 7.9-29**, the various FIFO buffers can be configured by writing to the RX FIFO Configuration/Control Registers and the Common FIFO Configuration/Control Registers.

For the 8 RX FIFO buffers, the following parameters can be configured:

- Interrupts
- FIFO depth
- FIFO payload data size

For the common FIFO buffers, the following parameters can be configured:

- Mode
- Interrupts FIFO depth
- FIFO payload data size
- FIFO TX link position

(1) FIFO Mode Configuration of Common FIFO Buffers

The mode of the common FIFO buffers can be configured by writing to the CFDCFCCn.CFM[1:0] bits in the Common FIFO Configuration/Control Registers. The possible modes of configuration for Common FIFO buffers are:

- 00b: RX mode (default mode after hardware reset)
- 01b: TX mode
- 10b: GW mode
- 11b: Reserved (do not write this value to the register bits)

Messages can only be read from the RX FIFO buffers and the Common FIFO buffers configured in RX mode. Messages are stored by the CAN module in these FIFO buffers based on the AFL entries.

Messages can be read and written into the Common FIFO buffers configured in TX mode. These messages are transmitted on the appropriate CAN channel.

Messages can only be read from the Common FIFO buffers configured in GW mode. However, the CPU read access has no impact on the read or write pointers. The pointers can only be incremented when a new message is stored in the FIFO buffer and decremented when a message is transmitted on the corresponding CAN channel by the CAN-FD module.

After a hardware reset, all the Common FIFO buffers are configured in RX mode by default. Only enable the FIFO buffers after configuring the Common FIFO buffers in the required modes.

In GW mode, when a transmit/receive FIFO buffer is trying to receive a new message while the transmit/receive FIFO buffer is already full of data, the oldest data of the buffer is overwritten with the message received or the message is discarded. The behavior is determined by setting the CFDCFCCEn.CFMOWM bit.

- When CFDCFCCEn.CFMOWM = 0b:
  When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the received message is discarded, and CFDCFSTSn.CFMLT bit is set to 1b.
- When CFDCFCCEn.CFMOWM = 1b:
  When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the oldest data in the buffer is overwritten with the received message. The read pointer of the transmit/receive FIFO buffer simultaneously moves to the next oldest message. The CFDCFSTSn.CFMOW bit is then set to 1b, which notifies that the oldest message has been overwritten with the received message.
  In addition, when a CAN bus error or arbitration-lost for the transmitting message occurs in the transmit/receive

FIFO buffer full, the transmitting message is lost and retransmission for the message is not performed. The read point is then moves to the next message automatically. Do not write to this bit when the CFDCFCCn.CFE bit is 1b.

(2) FIFO TX Message Buffer Link Configuration

When the common FIFO is configured as TX or GW FIFO, the FIFO buffer must be linked to a normal TX message buffer to participate in the transmission scan of a CAN channel.

The link to a normal TX message buffer must be unique, for example the same TX message buffer cannot be shared between 2 or more common FIFO buffers.

Do not write data into a TX message buffer that is linked to a Common FIFO buffer. Also, the TX message buffer linked to a Common FIFO buffer should not be a part of the TX Queue.

The TX message buffer link of each Common FIFO buffer can be configured by writing to the CFDCFCCn.CFTML[4:0] bits in the Common FIFO Configuration/Control Registers. Available options for TX message buffer link configuration are:

- 00h: TX Message Buffer 32
- 01h: TX Message Buffer 33
  ⋮
- 0Fh: TX Message Buffer 47

(3) FIFO Depth Configuration

The depth of each FIFO buffer can be configured by writing to the CFDRFCCn.RFDC[2:0] bits and CFDCFCCn.CFDC[2:0] bits in the RX FIFO Configuration/Control Registers and the Common FIFO Configuration/ Control Registers. The 8 available options for depth configuration are:

- 000b: 0 Messages (FIFO buffer cannot be enabled)
- 001b: 4 Messages
- 010b: 8 Messages
- 011b: 16 Messages
- 100b: 32 Messages
- 101b: 48 Messages
- 110b: 64 Messages
- 111b: 128 Messages

The RAM allocation for RX message buffers along with FIFO buffers is limited to (n + 1) × 256 messages. Configuration of the RX message buffers, along with FIFO buffers, which exceed this maximum limit should not be done.

CAN-FD module logic does not check the validity of the configuration.

Note: If the FIFO depth of a common FIFO is 4 messages or more (CFDCFCCn.CFDC[2:0] > 000b), then the Common FIFO TX message buffer link is valid when the FIFO is disabled or enabled.

Note: If FIFO depth is 0 messages, then the Common FIFO TX message buffer link is not valid when the FIFO is disabled or enabled.

(4) FIFO Payload Size Configuration

The data size of each FIFO buffer can be configured by writing to the CFDRFCCn.RFPLS[2:0] bits and CFDCFCCn.CFPLS[2:0] bits in the RX FIFO Configuration/Control Registers and the Common FIFO Configuration/ Control Registers. The eight available options for depth configuration are:

- 000b: 8 bytes
- 001b: 12 bytes
- 010b: 16 bytes
- 011b: 20 bytes
- 100b: 24 bytes
- 101b: 32 bytes
- 110b: 48 bytes
- 111b: 64 bytes

(5) FIFO Interrupt Configuration

The interrupt generation conditions for the FIFO buffers can be configured by writing to the CFDRFCCn.RFIM bit and the CFDCFCCn.CFIM bit in the RX FIFO Configuration/Control Registers and the Common FIFO Configuration/Control Registers. The two available options are:

- 0b:
  - RX FIFO mode: Interrupt generated when the Common FIFO counter reaches CFDRFCCn.RFIGCV/ CFDCFCCn.CFIGCV value
  - TX FIFO mode: Interrupt generated when the Common FIFO transmits the last message successfully
  - GW FIFO mode
    Frame RX: Interrupt generated when message counter increments and reaches the interrupt threshold value
    Frame TX: Interrupt generated when the last message is transmitted successfully from FIFO
- 1b:
  - RX FIFO mode: Interrupt generated at the end of storage of every received message
  - TX FIFO mode: Interrupt generated for every successfully transmitted message
  - GW FIFO mode
    Frame RX: Interrupt generated when message is stored in the FIFO
    Frame TX: Interrupt generated when message is successfully transmitted from the FIFO

If the Interrupt Mode bit is 0b for a RX FIFO, then interrupt is generated based on the configuration of the CFDRFCCn.RFIGCV[2:0] bits.

Similarly, if the Interrupt Mode bit is 0b for a Common FIFO configured in RX mode, then interrupt is generated based on the configuration of CFDCFCCn.CFIGCV[2:0] bits.

The eight available options for configuring the FIFO counter value for generation of an interrupt are:

- 000b: Interrupt generated when FIFO is 1/8th Full
- 001b: Interrupt generated when FIFO is 1/4th Full
- 010b: Interrupt generated when FIFO is 3/8th Full
- 011b: Interrupt generated when FIFO is 1/2 Full

- 100b: Interrupt generated when FIFO is 5/8th Full
- 101b: Interrupt generated when FIFO is 3/4th Full
- 110b: Interrupt generated when FIFO is 7/8th Full
- 111b: Interrupt generated when FIFO is Full

In this case, an interrupt is generated when the message count matches the configured value.

However, there are some limitations on the configuration of the CFDRFCCn.RFIGCV[2:0] and CFDCFCCn.CFIGCV[2:0] bits depending on the FDC[2:0] bits (FIFO Depth Configuration), see **Table 7.9-24**.

Table 7.9-24 FIFO Interrupt Generation Counter and FIFO Depth Configuration

RFDC[2:0] (FDC[2:0])	RFIGCV[2:0] (CFGCV[2:0])							
	111b	110b	101b	100b	011b	010b	001b	000b
000b	Don't care (FIFO cannot be enabled)							
001b	Allowed	Not allowed	Allowed	Not allowed	Allowed	Not allowed	Allowed	Not allowed
010b	Allowed							
011b	Allowed							
100b	Allowed							
101b	Allowed							
110b	Allowed							
111b	Allowed							

Common FIFO buffer can set an interrupt output at the completion time of transmitting one frame, or the completion of reception. In addition, Common FIFO and RX FIFO can set an interrupt output, when stored to the setup number (CFDC/ RFDC) of FIFO stages.

7.9.6.2.2 FIFO Buffers Control

The FIFO interrupt must be enabled by setting any one of the following bits in the RX FIFO Configuration/Control Registers:

- CFDRFCCn.RFIE
- CFDRFCCn.RFFIE

In addition, the FIFO interrupt must be enabled by setting any one of the following bits in the Common FIFO Configuration/Control Registers:

- CFDCFCCn.CFRXIE
- CFDCFCCn.CFTXIE
- CFDCFCCEn.CFFIE
- CFDCFCCEn.CFOFRXIE
- CFDCFCCEn.CFOFTXIE

After configuration is complete, each FIFO can be enabled by setting the CFDRFCCn.RFE and CFDCFCCn.CFE bits in the RX FIFO Configuration/Control Registers and the Common FIFO Configuration/Control Registers to allow transmission and reception of messages.

When CFDCFCCEn.CFBME = 1b, it becomes FIFO buffering mode, send data is stored in the Common FIFO, and transmission is stopped. Transmission starts when CFDCFCCEn.CFBME = 0b.

Do not write 1b from 0b for this bit when the CFDCFCCn.CFE bit is 1b.

7.9.7 Interrupt and DMA

7.9.7.1 Interrupts

The CAN-FD module generates several interrupts. The interrupt output, which is connected to the Interrupt Controller Unit (ICU), can be controlled by the corresponding interrupt enable bit.

The status flag is set independent from this enable bit.

The channel transmission interrupt has an additional Status Flag register. These status bits are only set when the corresponding interrupt enables are set.

The Status Flag register supports the identification of the interrupt source for the channel transmission, as this interrupt is driven by several trigger sources.

The interrupts in the CAN-FD module can be classified into 2 groups, global interrupts and channel interrupts:

- Global interrupts

The CAN-FD module can generate 2 global interrupts:

1. One Global interrupt for successful reception into the 8 RX FIFO buffers
2. One Global error interrupt

- Channel interrupts:

Each channel of the CAN-FD module can generate 3 channel interrupts:

1. Channel transmission
   Transmission completion from channel
   Transmission abort from channel
   Transmission from TX Queue for a channel
   Channel THL Interrupt
   Successful transmission from a Common FIFO in TX or GW mode for a channel
2. Channel error interrupt
3. Successful reception in a Common FIFO in RX or GW mode for a channel or successful routing in a TXQ

The interrupts are cleared when the corresponding flag bits are cleared or interrupt enable bits are cleared.

Table 7.9-25 provides an overview of interrupt.

Table 7.9-25 Overview of Interrupt Source

Parameter	Interrupt	Interrupt Source	Interrupt Clearing
Global Interrupts	Successful reception into at least one RX FIFO	Interrupt flag of corresponding RX FIFO for which interrupt is enabled	Clear the interrupt flag of corresponding RX FIFO buffer for which interrupt is enabled
	FIFO full into at least one RX FIFO	FIFO Full Interrupt flag of corresponding RX FIFO for which interrupt is enabled	Clear the FIFO Full Interrupt flags of corresponding RX FIFO buffer for which interrupt is enabled
	Global error	Any of the following: • DLC Error flag • Message Lost Status bit • Message Overwrite Status bit • TXQ Message Lost Status bit • TXQ Message Overwrite Status bit • TX History Entry Lost Status bit • CAN-FD Message Payload overflow flag	Clear all of: • DLC Error flag • Message Lost flags in all of the FIFO Status Registers • Message Overwrite flags in all of the Common FIFO Status Registers • Message Lost flags in all of the TXQ Status Registers • Message Overwrite Flags in all of the TXQ Status Registers • TX History List Entry Lost flag • CAN-FD Message Payload Overflow flag
Channel Transmission Interrupts	Channel n successful transmission	Any channel related TX MB Successful flag when interrupt is enabled Separate interrupts are provided for common FIFO buffers and TX Queue*1	Clear all channel related TX MB Result Status bits for which the interrupt is enabled
	Channel n abort	Any channel related TX MB Abort flag when interrupt is enabled Separate interrupts are provided for common FIFO buffers and TX Queue*1	Clear all channel related TX MB Result Status bits for which the interrupt is enabled globally
	Channel n transmission from TX Queue	Related channel TX Queue Interrupt flag	Clear related channel TX Queue Interrupt flag
	Channel n THL Interrupt	Channel n THL Interrupt Status flag	Clear the relevant THL Interrupt Status flag
	Channel n COM FIFO TX Interrupt	Interrupt flag for Common FIFOs in TX or GW mode belonging to the related channel	Clear the interrupt flags of Common FIFOs in TX or GW mode belonging to the related channel
	Channel n COM FIFO One Frame TX Interrupt	One Frame Transmission Interrupt flag for Common FIFOs belonging to the related channel	Clear the One Frame Transmission Interrupt flags of Common FIFOs belonging to the related channel
	Channel n TXQ One Frame TX Interrupt	One Frame Transmission Interrupt flag for TXQs belonging to the related channel	Clear the One Frame Transmission Interrupt flags of TXQs belonging to the related channel
Channel COM RX FIFO Interrupt	Channel n COM FIFO RX Interrupt	Interrupt flag for Common FIFOs in RX or GW mode belonging to the related channel	Clear the interrupt flags of Common FIFOs in RX or GW mode belonging to the related channel
	Channel n COM FIFO One Frame RX Interrupt	One Frame Reception Interrupt flag for Common FIFOs belonging to the related channel	Clear the One Frame Reception Interrupt flags of Common FIFOs belonging to the related channel
	Channel n COM FIFO Full Interrupt	FIFO Full Interrupt flag for Common FIFOs in RX or GW mode belonging to the related channel	Clear the FIFO Full Interrupt flags of Common FIFOs in RX or GW mode belonging to the related channel
	Channel n TXQ One Frame Routing Interrupt	One Frame Routing Interrupt flag for TXQs in GW mode belonging to the related channel	Clear the One Frame Routing Interrupt flags of TXQs in GW mode belonging to the related channel
	Channel n TXQ Full Interrupt	TXQ Full Interrupt flag for TXQs in GW mode belonging to the related channel	Clear the FIFO Full Interrupt flags of TXQs in GW mode belonging to the related channel
Channel Error Interrupt	Channel n Error	Any channel related error flag in the Channel Error Flag Register for which interrupt is enabled in the Channel Error Interrupt Enable Register	Clear all channel related error flags in the Channel Error Flag Register for which interrupt is enabled in the Channel Error Interrupt Enable Register

Note 1. These interrupts are only set for TX Message Buffers that do not belong to an enabled TX Queue and are not pointing to a common FIFO.

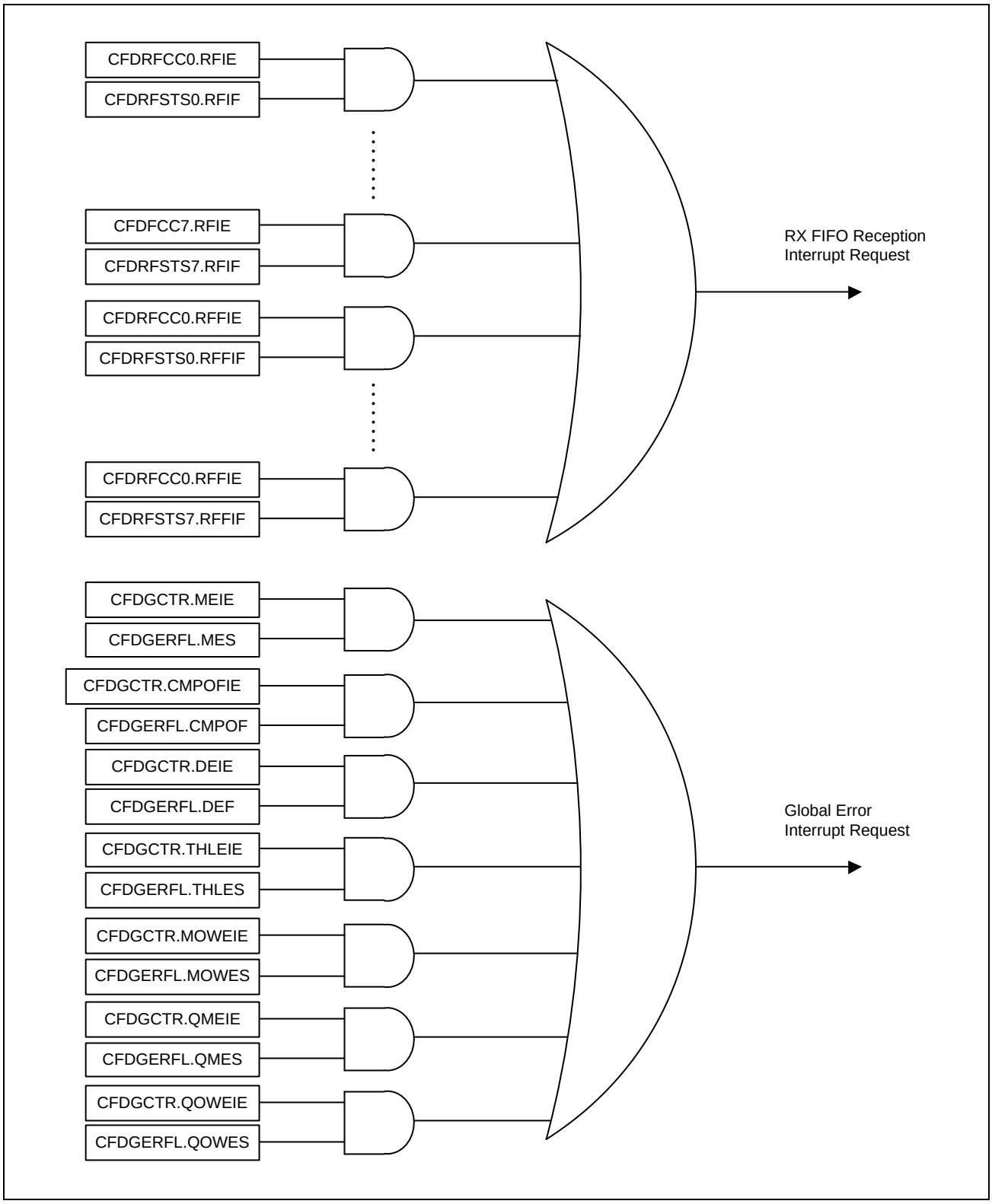


Figure 7.9-30 Block Diagram of Global Interrupt

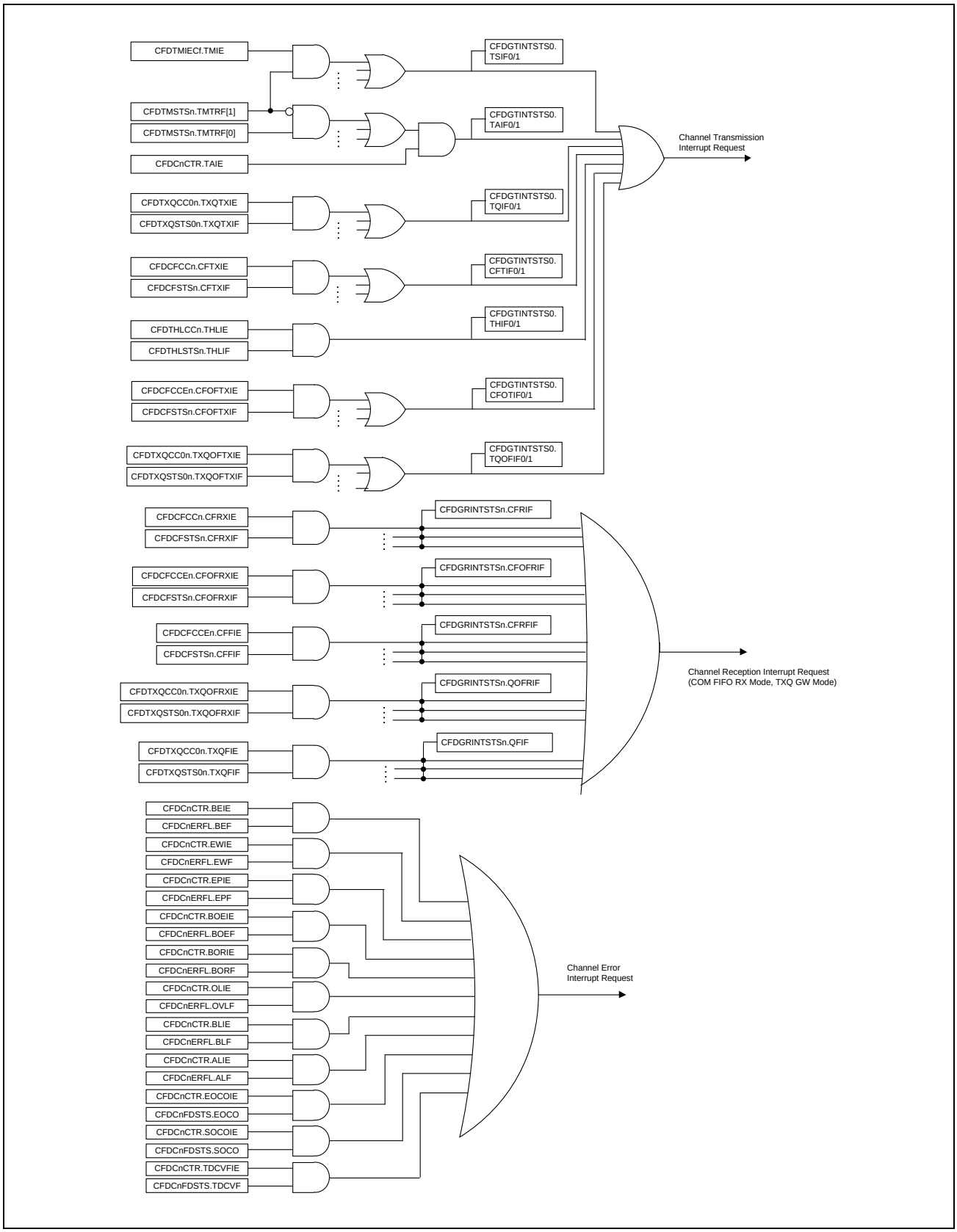


Figure 7.9-31 Block Diagram of Channel Interrupt

7.9.7.2 DMA Transfer

The CAN-FD module has some message buffer which can be associated with a DMA channel:

- Reception DMA
  - 8 RX FIFO Message Buffers
  - 6 Common FIFO Message Buffers
- Transmission DMA
  - 16 TXQ Message Buffers (TXQ0, TXQ3)
  - 6 Common FIFO Message Buffer

Figure 7.9-32 shows the potential DMA channels.

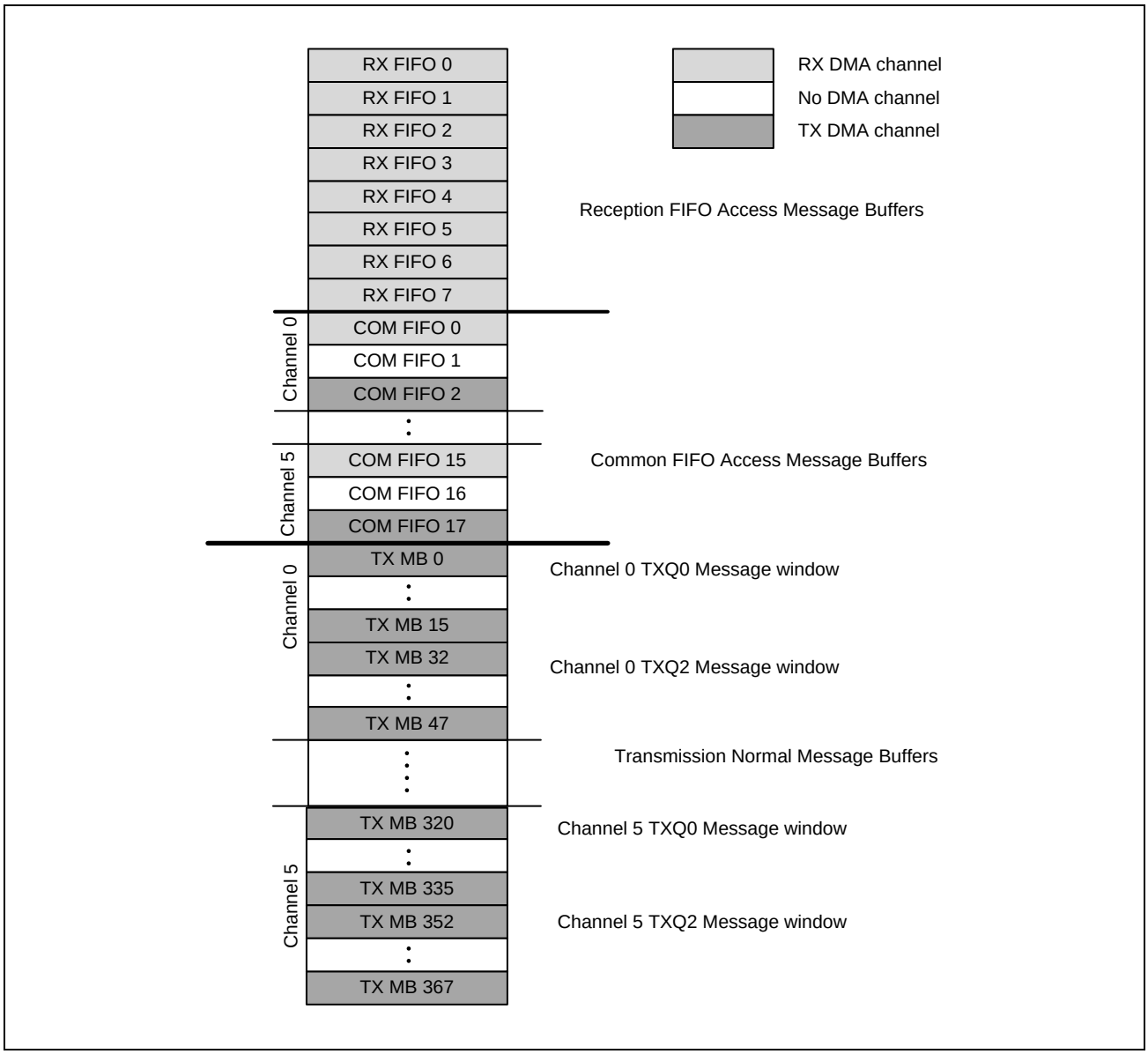


Figure 7.9-32 Message Buffer Connectable to DMA Channel

A DMA channel transfer request is generated for each FIFO entry to the DMAC when the related CFDCDTCT.RFDMAE or CFDCDTCT.CFDMAE is set to 1b and the belonging FIFO is not empty.

Reception FIFO interrupt should be disabled for this particular FIFO (CFDRFCCn.RFIE or CFDCFCCn.CFRXIE).

Use the regular start address for the DMA access window address and add 8000h to the regular start address for the debugger access window. See **Table 7.9-26**.

Table 7.9-26 DMA Channel Access Window Address

b = Message Buffer Component.Index	MBCP	Register	p	Start Address n = 0, 1
0 to number_of_RFMBCPs - 1	RFMBCPb	RFID	x	6000h + b × 0080h
		RFPTR	x	6004h + b × 0080h
		RFFDSTS	x	6008h + b × 0080h
		RFDFp	0 to 15	600Ch + p × 0004h + b × 0080h
0 to number_of_CFMBCPs_per_channel - 1	CFMBCPb	CFID	x	6400h + b × 0080h + n × 0180h
		CFPTR	x	6404h + b × 0080h + n × 0180h
		CFFDCSTS	x	6408h + b × 0080h + n × 0180h
		CFDFp	0 to 15	640Ch + p × 0004h + b × 0080h + n × 0180h

DMA FIFO pointer decrement is done automatically by reading the last configured data payload byte (CFDRFCCn.RFPLS or CFDCFCCn.CFPLS).

Note: The DMA must read the exact length of the configured data payload size (CFDRFCCn.RFPLS or CFDCFCCn.CFPLS).

The software debugger must access outside of the regular SFR address range from E000h to EFFFh.

Do not write to the FIFO and TXQ control registers when DMA is enabled.

The DMA enable of the particular DMA FIFO (CFDCDTCT.RFDMAE or CFDCDTCT.CFDMAE) can be set at any time, the procedure in this section is a configuration flow for an initial setup.

When CFDCDTTCT.TQ0DMAE or CFDCDTTCT.TQ3DMAE or CFDCDTTCT.CFDMAE is set, the messages of the corresponding TXQ or Common FIFO can be handled by DMA controller.

Use the following procedure when the TXQ or the Common FIFO can be handled by DMA controller.

1. CPU checks the TXQ or the Common FIFO is not full.
2. When transmit data can be used, CPU sets this data to Common FIFO or TXQ.
   When using Common FIFO, transmit data is write in CFDCFID, CFDCFPTR, CFDCFFDCSTS, and CFDTMBCPb[i] registers.
   When using TXQ, transmit data is write in CFDTMID, CFDTMPTR, CFDTMFDCTR, and CFDTMDFp registers.
3. For Common FIFO, the common FIFO pointer is incremented automatically when DMA controller writes the last data payload byte configured by CFDCFCCn.CFPLS.
   For TXQ, if the data of 64 data payload is written, a TXQ pointer increases automatically. When payload data is less than 64 bytes, dummy data must be written in and 64 data payload size must be done.

Note: Only 32-bit write-access can be possible on the DMA message handling.

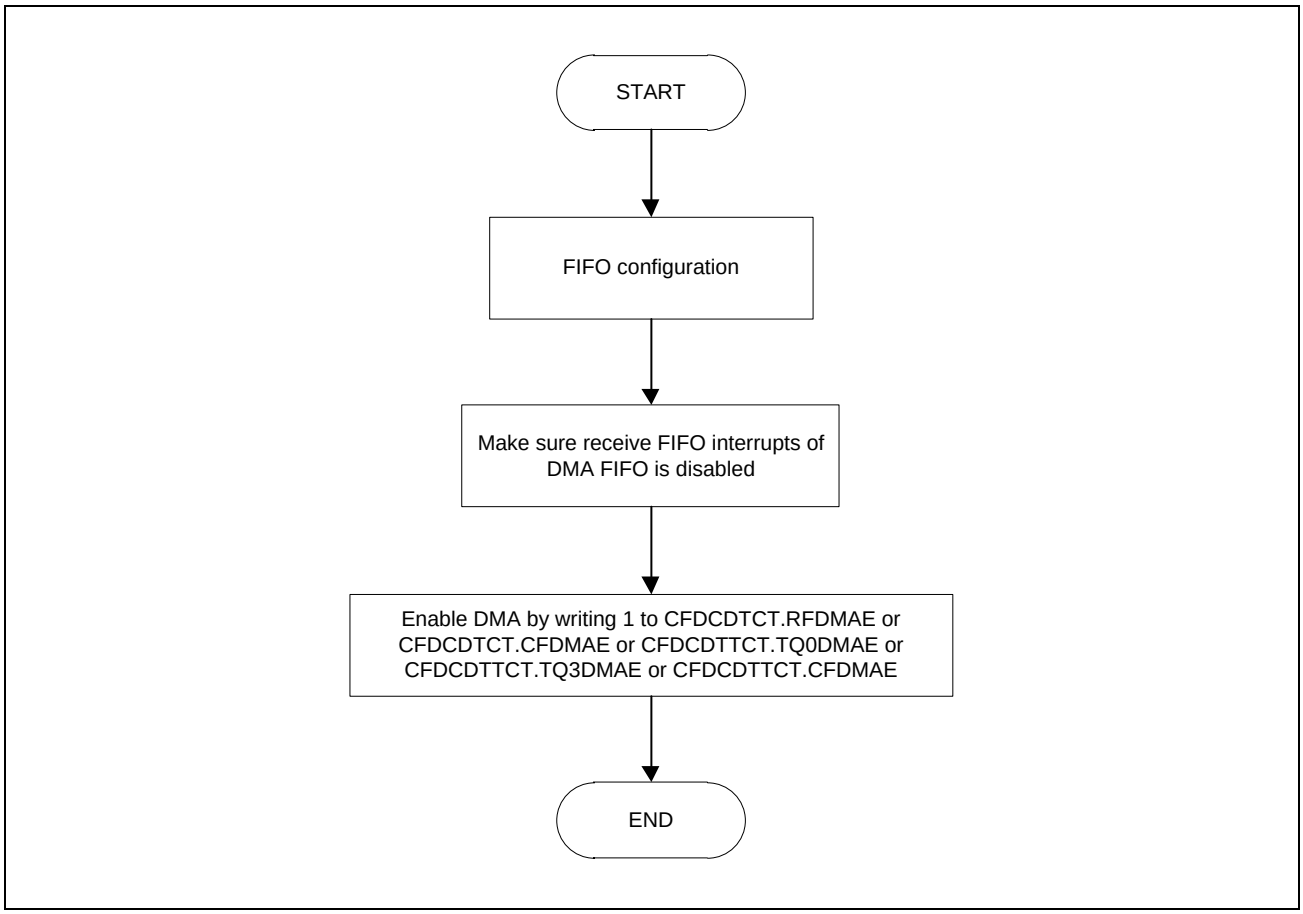


Figure 7.9-33 DMA Enable Flow

To disable a DMA transfer requested, disable the particular DMA enable bit (CFDCDTCT.RFDMAE or CFDCDTCT.CFDMAE). If the disable is made during an ongoing transfer, then this must be completed first before further action can be taken. The transfer status can be identified by the CFDCDTSTS.RFDMASTS or CFDCDTSTS.CFDMASTS. For reference, see the flow in **Figure 7.9-34**. When the DMA is disabled, then consider what to do with the remaining or new incoming messages to this particular FIFO reception.

When the FIFO is not disabled, reception to the FIFO continues.

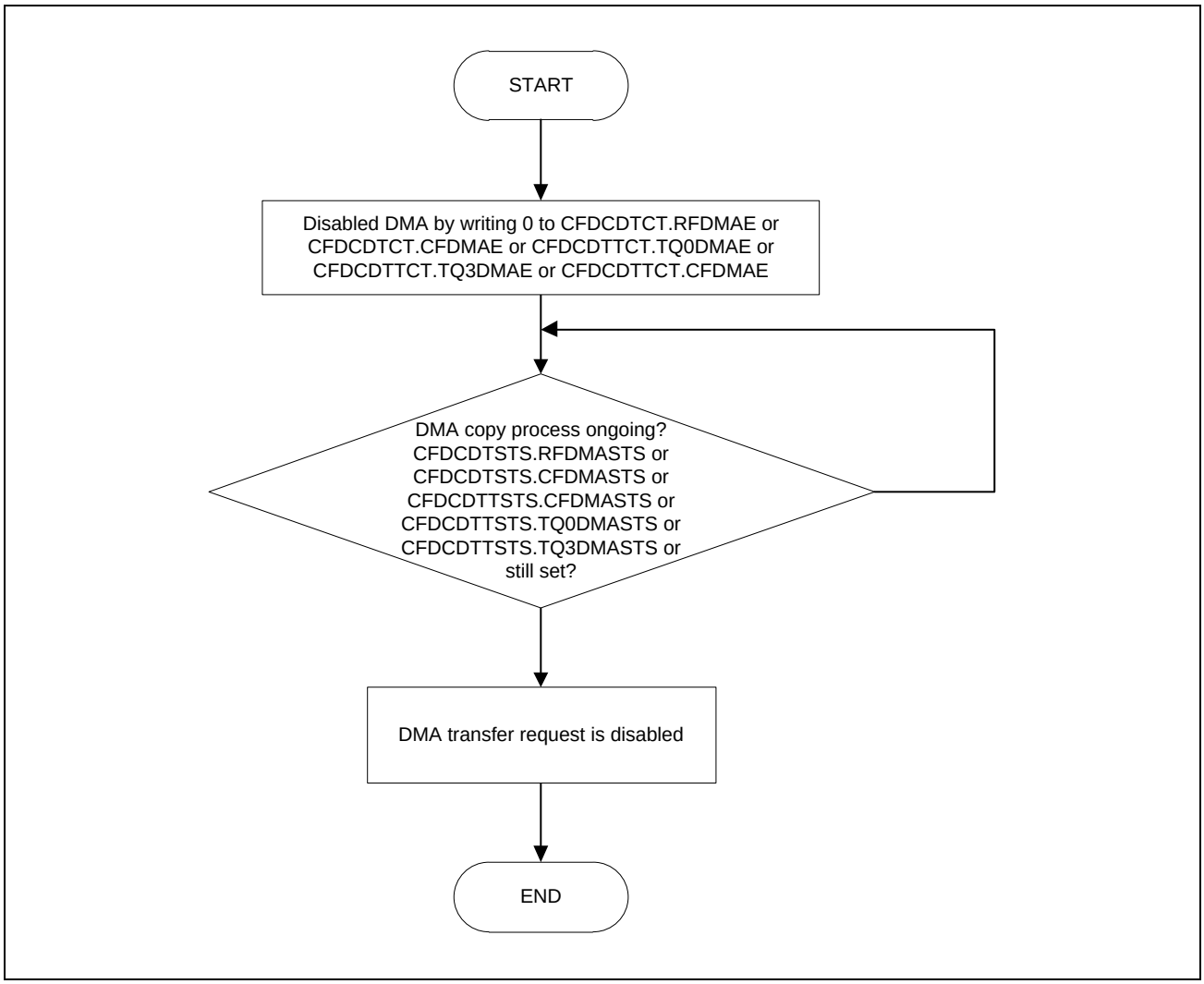


Figure 7.9-34 DMA Disable Flow

7.9.8 Reception and Transmission

7.9.8.1 Reception

FIFO buffers or Common FIFO buffers configured in RX mode or GW mode depending on the Acceptance Filter List entries are as follows:

- Up to 32 RX Message Buffers can be configured
- 8 RX FIFO buffers available
- Up to 6 Common FIFO buffers can be configured in RX mode or GW mode
- Up to 4 TX Queues can be configured in GW mode

7.9.8.1.1 Message Storage in RX Message Buffers

When a message is successfully received and stored in a RX Message Buffer, the corresponding New Data flag is set in the RX Message Buffer New Data Register.

The CAN message can be read from the corresponding RX Message Buffer.

If a new message is stored into a RX Message Buffer before the previous message in this message buffer can be read, then the original message is overwritten. There is no mechanism for preventing a new message from overwriting the current message in the RX Message Buffer. If such a loss of messages is not acceptable, then RX FIFO should be used to store related messages.

Note: Interrupts are not provided for the RX Message Buffers in the CAN-FD module and so the RX Message Buffer New Data Register should be accessed periodically to check if a new message has been stored in the RX Message Buffers.

Note: Unused data bytes are filled with 00h depending on the DLC value.

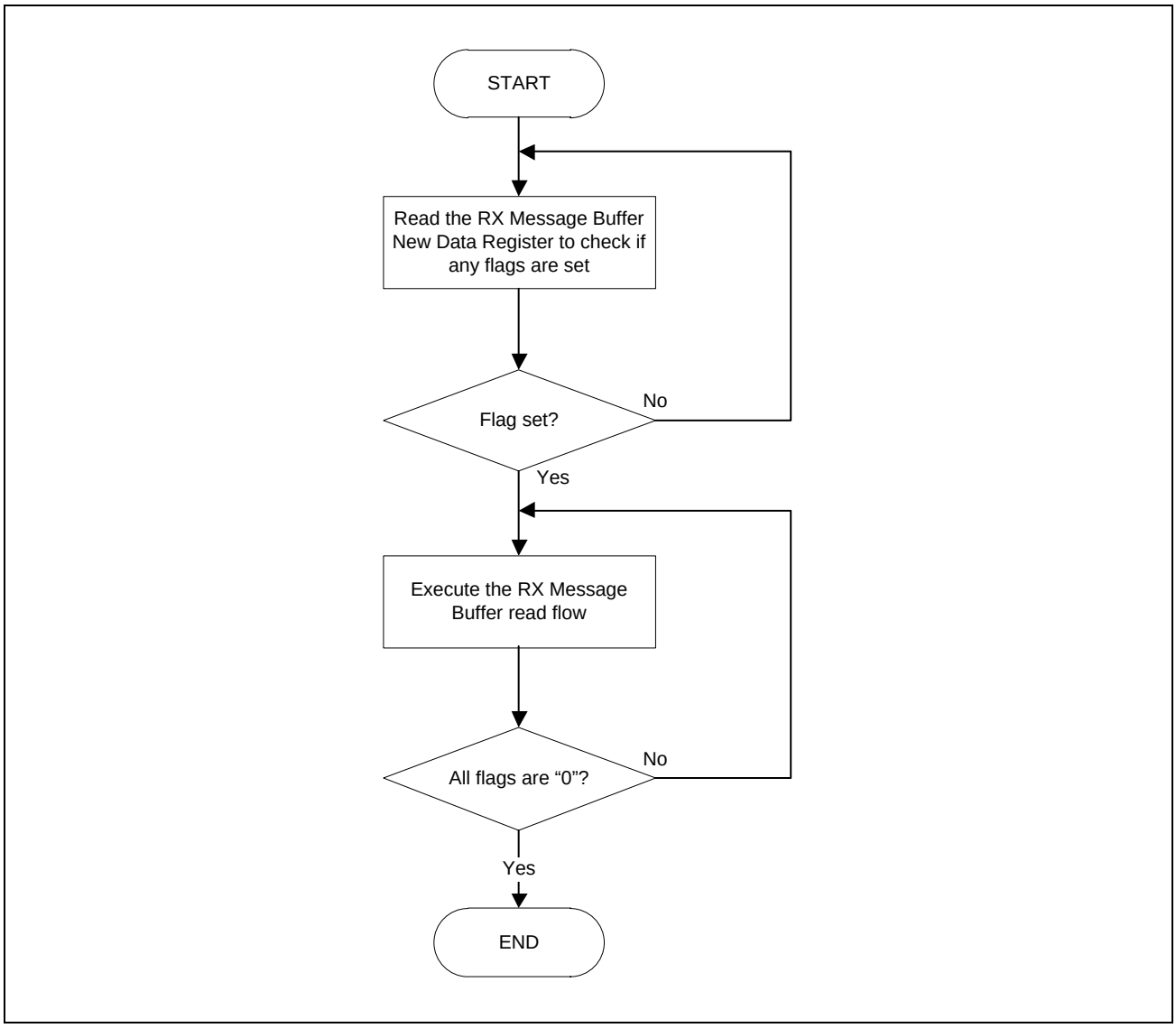


Figure 7.9-35 RX Message Buffer Message Access Flow

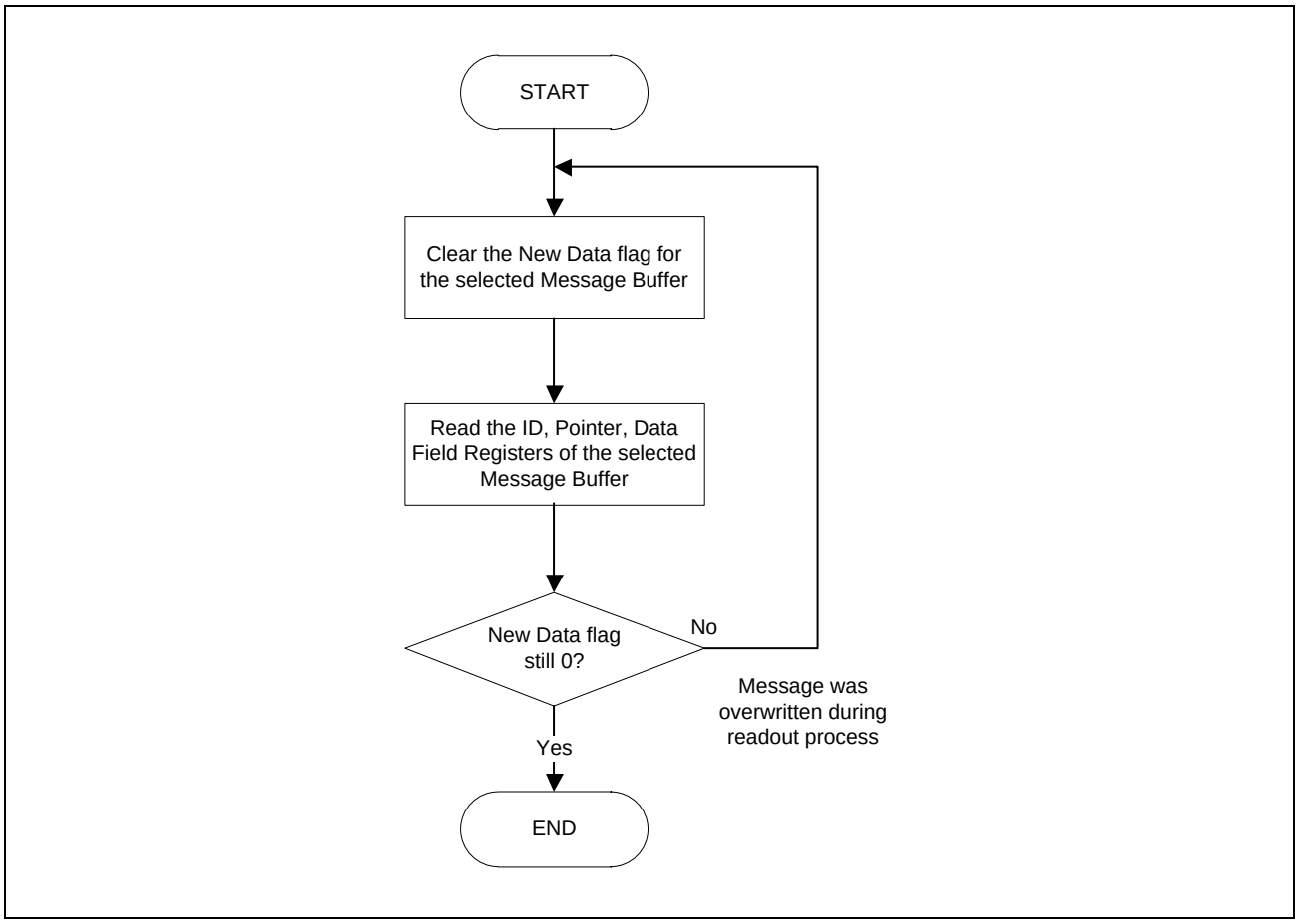


Figure 7.9-36 RX Message Buffer Read Flow

7.9.8.1.2 Message Storage in FIFO Buffers

The AFL entries for routing the received messages to RX FIFO buffers or Common FIFO buffers configured in RX or GW mode should be configured based on system requirements.

The CFDGAFLP1n.GAFLFDP[31:0] field in the matching AFL entry selects the FIFO buffers to which the related reception message is stored.

When the received message is stored in one or more RX FIFO buffers or Common FIFO buffers configured in RX mode or GW mode, the message counter value is incremented in the corresponding RX FIFO Status Registers or Common FIFO Status Registers.

Depending on the configuration of the FIFO buffers, an interrupt may also be generated.

The message can be read from the corresponding FIFO Access registers.

Note: Because many messages can be stored in the FIFO buffers, reading more than one message may be required to read the latest message stored in a FIFO buffer.

If the message count value matches the FIFO depth, the FIFO Full flag is set.

When the value FFh is written to the corresponding FIFO Pointer Control Register, the message count is decremented by 1.

Only write FFh to the FIFO Pointer Control register after reading the complete message from the FIFO Access registers of the corresponding FIFO.

When all the messages stored in the FIFO are read, the FIFO Empty flag is set.

If a new message is stored into the FIFO when the FIFO message count matches the FIFO depth (FIFO Full condition), the FIFO Message Lost flag is set and the new message is lost (no overwrite of already stored messages takes place).

An appropriate value can be configured as warning level to generate an interrupt before the FIFO full condition occurs to avoid loss of a message due to overrun condition.

In GW mode, when a transmit/receive FIFO buffer is trying to receive a new message while the transmit/receive FIFO buffer is already full of data, the oldest data of the buffer is overwritten with the message received or the message is discarded. The behavior is determined by setting CFDCFCCEn.CFMOWM bit.

- When CFDCFCCEn.CFMOWM = 0b:
  When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the received message is discarded and CFDCFSTSn.CFMLT bit is set to 1b.
- When CFDCFCCEn.CFMOWM = 1b:
  When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the oldest data in the buffer is overwritten with the received message.
  The read pointer of the transmit/receive FIFO buffer simultaneously moves to the next oldest message.
  The CFDCFSTSn.CFMOW bit is then set to 1b, which notifies that the oldest message has been overwritten with the received message.
  In addition, if a CAN bus error or arbitration-lost for the transmitting message occurs in the transmit/receive FIFO buffer full, the transmitting message is lost and retransmission of the message is not performed. The read point moves to the next message automatically.

Do not write change for this bit when the CFDCFCCn.CFE bit is 1b. Common FIFO can set interrupt when:

- CAN frame reception is completed
- FIFO is in full status in RX mode or GW mode

Note: The message lost can be set only in RX or GW mode by CAN, the flag is not set when the CPU is overloading the FIFO buffers.

Note: When CFDGAFLP0n.GAFLSRDi (i = 0 to 2) is set and CFDTXQCCin.TXQGWE (i = 0 to 2, n = 0, 1) is also set, a receiving frame is stored in the target TXQ as send data by routing.

The RX FIFO Buffers and the Common FIFO Buffers configured in RX or GW mode can be disabled at any time by clearing the CFDRFCCn.RFE or CFDCFCCn.CFE bit in the RX FIFO Configuration/Control Registers and the Common FIFO Configuration/Control Registers.

When the CFDRFCCn.RFE or CFDCFCCn.CFE bit is cleared, the message read and write pointers of the FIFO are cleared and are no longer active. Therefore, all messages in the FIFO Buffers are lost and no further messages can be stored into the FIFO.

When the RX FIFO Buffers or Common FIFO Buffers configured in RX mode are assigned as DMA channel, software should not access the FIFO Access Register of this FIFO buffer or write FFh to the FIFO Pointer Control Register (CFDCFPCTRn.CFPC or CFDRFPCTRn.RFPC), because this could lead to unintended FIFO message decrement. The DMA channel controls the FIFO decrement automatically.

Note: If the interrupt flag is set for a FIFO Buffer and then the FIFO is disabled, the interrupt flag is not automatically cleared. The interrupt flag should be cleared before disabling the FIFO.

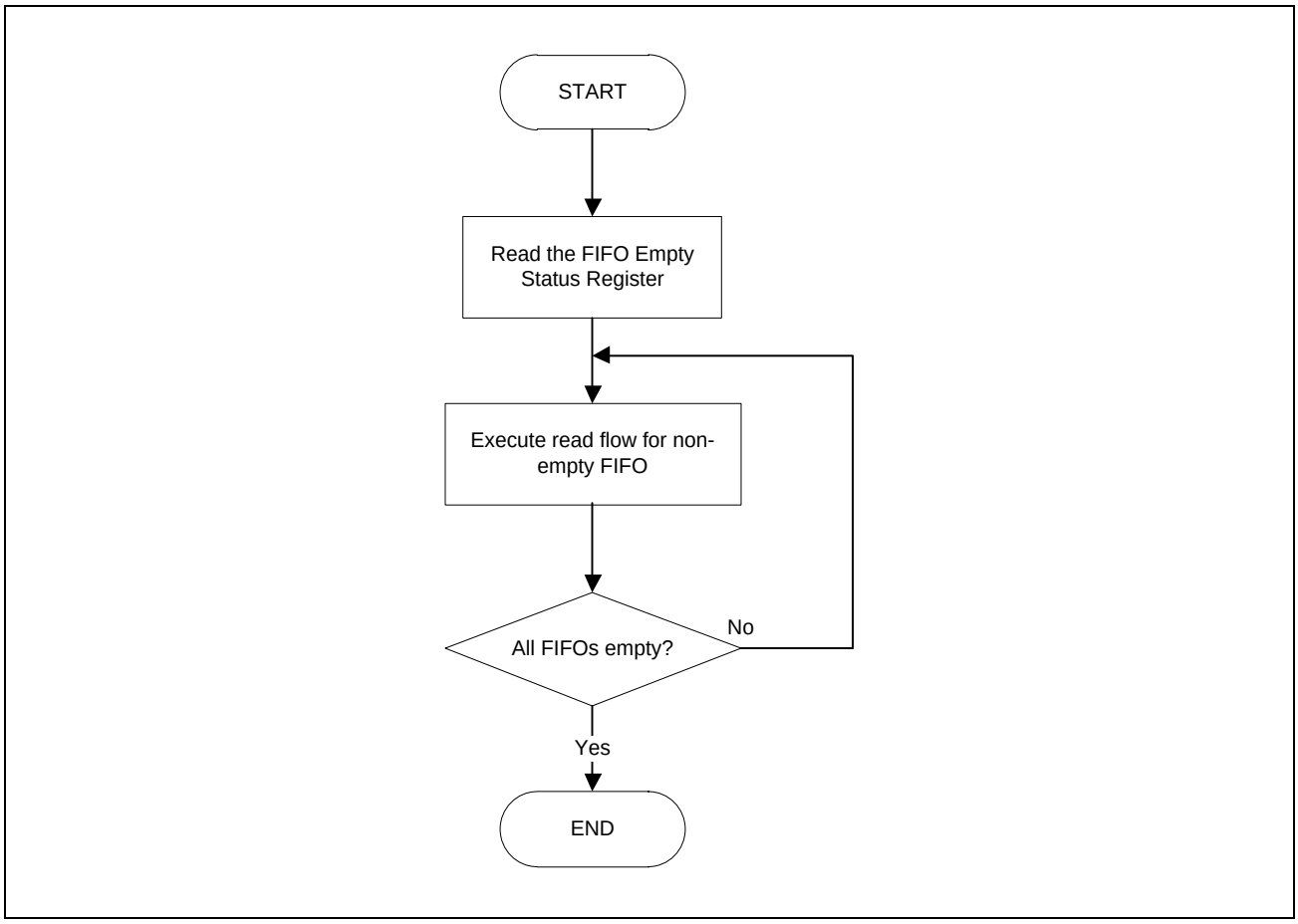


Figure 7.9-37 FIFO Buffer Message Access Flow (Example for Polling Case)

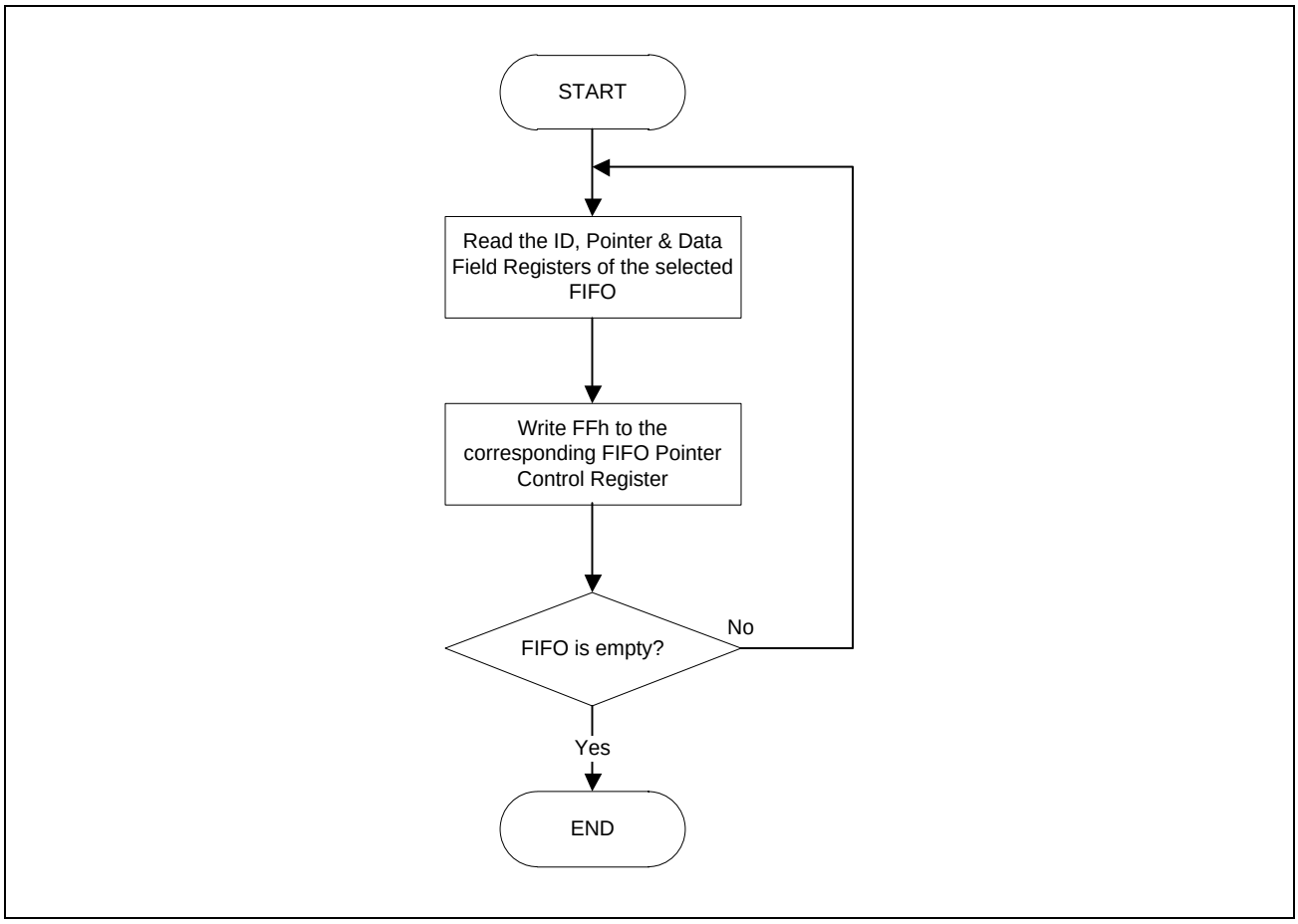


Figure 7.9-38 RX FIFO Buffer Read Flow (Example for Polling Case)

Note: When the next frame is received before clearing the completion interrupt flag of reception, the completion interrupt of reception is not set again.

Even if it clears an interruption flag after the completion processing of reception, the already received interrupt flag is not set.

It is necessary to perform the completion processing of reception even before the next completion of frame reception, and to clear an interruption flag.

When processing does not meet the deadline, after checking that receiving data is empty, interrupt flag is cleared and it checks that receiving data is empty again.

7.9.8.1.3 Timestamp

The Timestamp counter is a free-running counter that can be used to check reception time of an incoming message or transmission time of successful transmitted messages. The Timestamp counter value is captured based on the CFDGFDCFG.TSCCFG[1:0] configuration (at the sample point of start of frame, point in time when the frame is valid, or for CAN-FD frames also at the sample point of the RES bit). For reception, it is stored together with the message ID and Data into the target RX Message Buffer or RX/GW FIFO.

For transmit message, the Timestamp counter value is stored as part of the TX History List entry.

The counter can be clocked with the peripheral clock or with the CAN channel bit timing clock. The counter source clock can be configured with the CFDGCFG.TSSS bit of the Global Configuration Register. If it is 0b, the peripheral clock is used. If it is 1b, the selected CAN channel bit time clock is used.

The channel selection is done with the CFDGCFG.TSBTCS bit of the Global Configuration Register.

Care must be taken when using selected CAN channel bit time clock as the clock source. When entering Channel Halt mode or Channel Reset mode for this channel, the Timestamp counter is stopped. Therefore, for other CAN channels, the Timestamp counter value is also not updated.

If peripheral clock is selected as the Timestamp counter clock source, Channel modes do not influence the Timestamp counter function.

The source clock for the Timestamp counter can be divided by a factor defined by the CFDGCFG.TSP bits (Timestamp Prescaler) in the Global Configuration Register.

The Timestamp counter can be reset to 0000h with the CFDGCTR.TSRST bit (Timestamp Reset).

7.9.8.2 Transmission

There are several possible transmission configurations for each channel:

- Normal transmission
- FIFO transmission
- Gateway transmission
- TX Queue transmission
  A fixed number of transmission message buffers (32 TX message buffers) are dedicated for each channel. These message buffers are only used for transmission and cannot be configured for reception.
  Additionally transmission from TX Queue and/or Common FIFO in TX or GW mode can be configured in the following way (see **Figure 7.9-39**):
  - TX Queue: Up to 16 transmission message buffers for one channel can be grouped to form a TX Queue with a common access window.
    Upper transmission message buffers are used to form the TXQ1 or TXQ3. Lower transmission message buffers are used to form the TXQ0 or TXQ2.
    Transmission control and status registers of these transmission message buffers should not be used. One channel has four TX Queues. Each TXQ has an access window:
    - TXQ0 is transmission Message Buffer 0 of each channel
    - TXQ1 is transmission Message Buffer 15 of each channel
    - TXQ2 is transmission Message Buffer 32 of each channel
    - TXQ3 is transmission Message Buffer 47 of each channel

    When using TXQ1 and TXQ0 simultaneously, the sum of the depths of TXQ1 and TXQ0 should not exceed 16.
    When using TXQ3 and TXQ2 simultaneously, the sum of the depths of TXQ3 and TXQ2 should not exceed 16.
- Common FIFO (TX/GW mode): Each Common FIFO in TX or GW mode is linked to a dedicated channel.
  Each channel has a fixed number of three Common FIFOs assigned to it. Within the channel, a Common FIFO configured in TX or GW mode, can be freely linked (assigned) between 32 and 47 transmission message buffers (only one FIFO to one transmission message buffer).
  The Common FIFO buffer then replaces the transmission message buffer linked to it.
  Transmission control and status registers of these transmission message buffers should not be used.

See **Figure 7.9-28** for information on Common FIFO buffer assignment to related channels.

Note: Common FIFO buffers should not be linked to TX message buffers that are already part of a TX Queue.

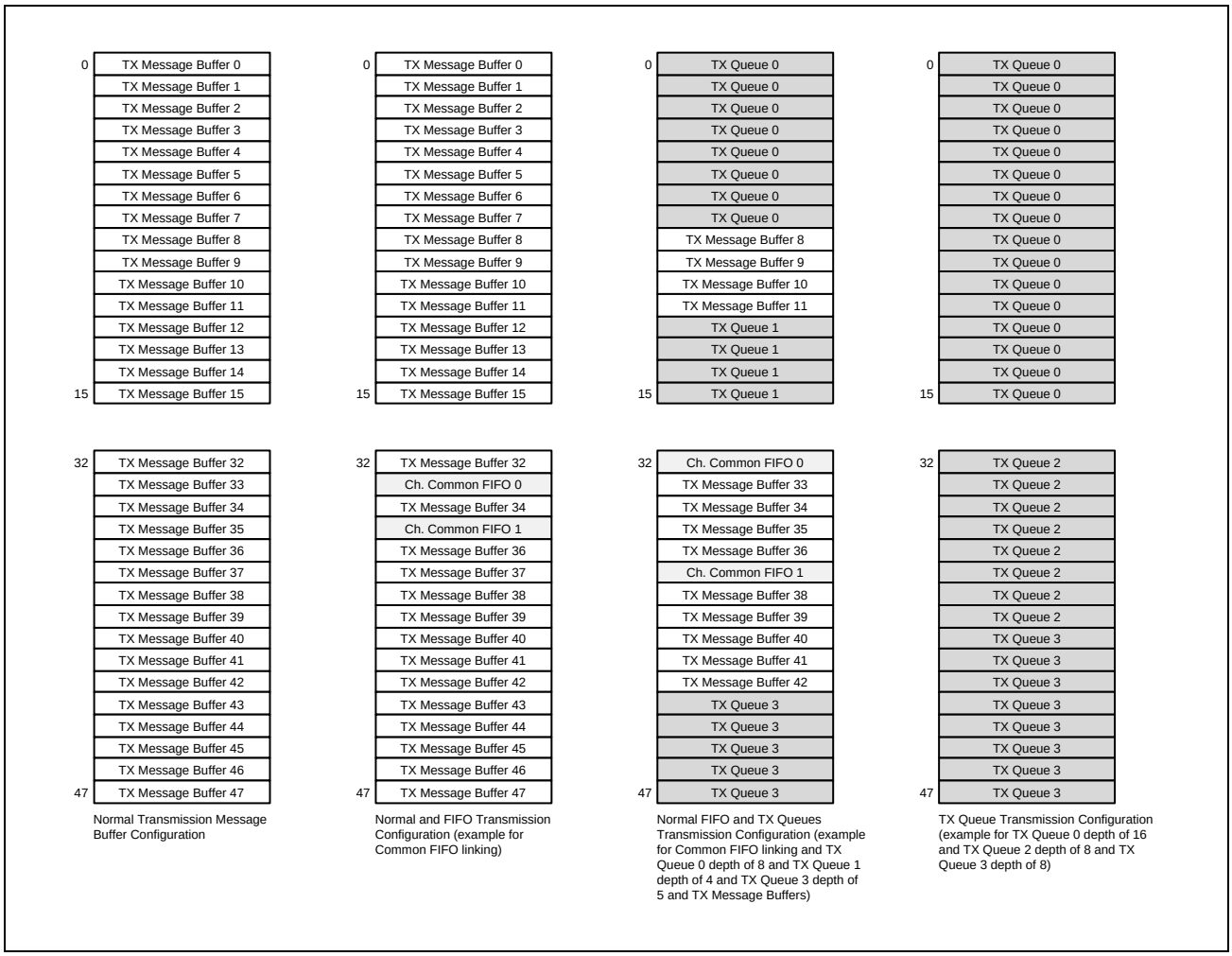


Figure 7.9-39 Channel Transmission Message Buffer Configuration

7.9.8.2.1 Transmission Priority

If two or more transmission message buffers of a channel are configured for transmission, the transmission priority in the CAN-FD module can be selected from the following two modes:

- CAN ID priority
- Message buffer number priority

The transmission priority mode is common for all message buffers and all CAN channels. It can be configured with the CFDGCFG.TPRI bit in the Global Configuration Register.

For message buffer number priority transmission, the smallest message buffer number with transmission request has the highest priority for transmission. This also includes the TX message buffers linked to the Common FIFO buffers configured in TX mode or GW mode.

However, message buffer number priority should not be used if TX Queue is enabled.

For CAN ID priority transmission, ID priority complies with the CAN bus arbitration rule (as specified in ISO 11898-1 specification). All TX message buffers can enter the ID priority comparison for message buffers configured for transmission. This also includes the TX message buffers linked to the Common FIFO buffers configured in TX mode or GW mode and includes the TX Queue message buffers.

If the ID of two or more message buffers is the same, the smaller message buffer number has higher priority for transmission.

Note: For Common FIFO buffers configured in TX mode or GW mode, only the message currently being pointed to by the FIFO read pointer can be included in the transmission arbitration.
If the message is being transmitted from the FIFO, the next pending message within the same FIFO will be considered in the transmission arbitration.
In contrast to this, all transmission message buffers of a TX Queue participate in internal transmission arbitration.

Figure 7.9-40 shows the transmission configuration flow.

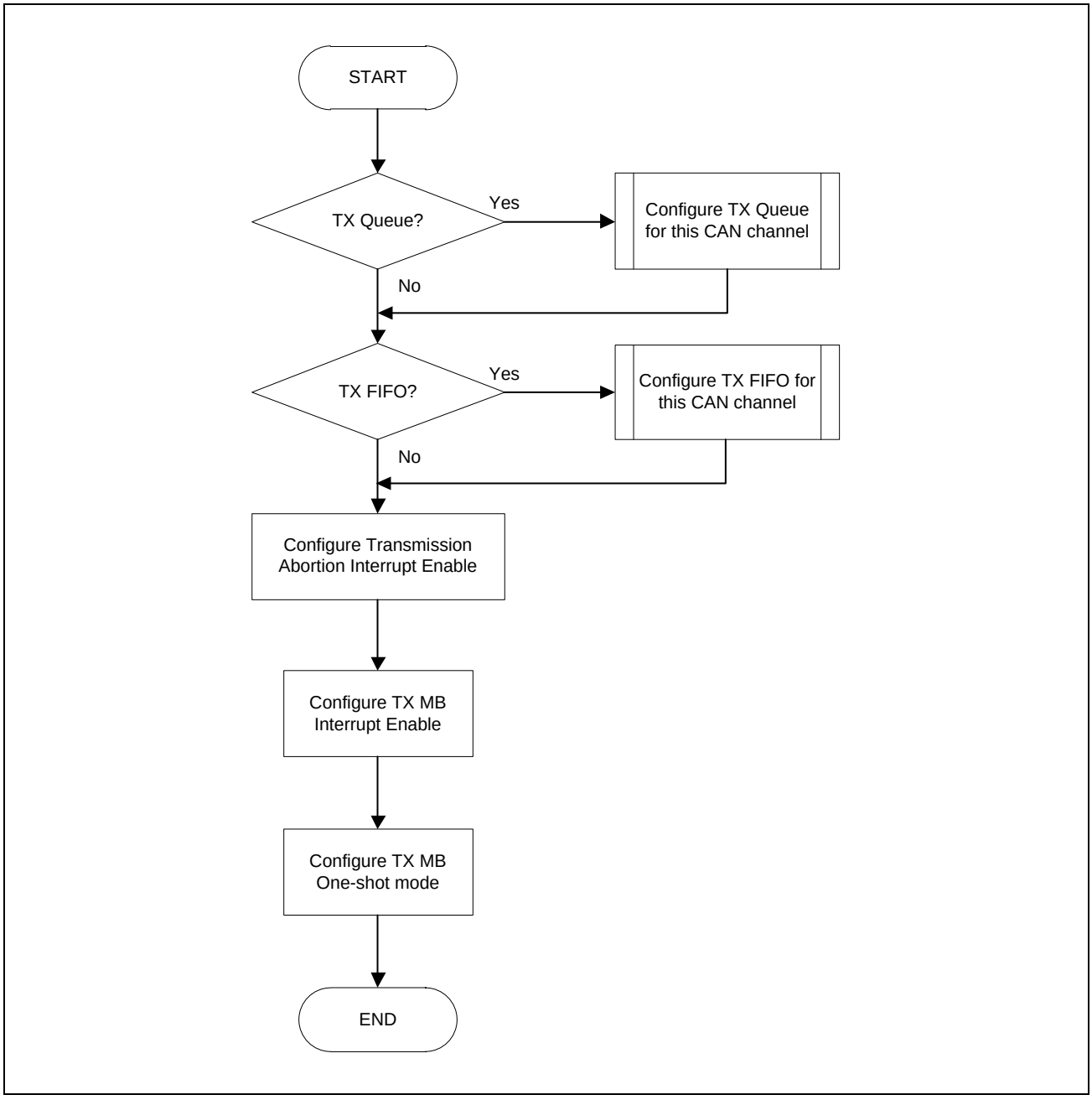


Figure 7.9-40 Transmission Configuration Flow

7.9.8.2.2 Normal Transmission

Each transmission message buffer has two modes of message transmission:

- Regular transmission mode
  If the message buffer is placed in regular transmission mode, the data frame or remote frame set in that message buffer can be transmitted.
  Completion of regular transmission can be checked through the related TX Message Buffer Transmission Result flag bits (CFDTMSTSn.TMTRF[1:0]) in the TX Message Buffer Status Registers. These bits are set to 10b or 11b when the regular transmission is successful.
  When arbitration is lost or an error occurs, message transmission will be attempted further if no transmission abort request is set for this transmission message buffer.
  New internal transmission arbitration for this channel is performed for all message buffers with transmission request.
- One-shot transmission mode
  When the CFDTMCn.TMOM bit of the TX Message Buffer Control Registers is set for a transmission message buffer, then the message buffer is placed in one-shot transmission mode and attempts to transmit a message only once.
  Completion of one-shot transmission can be checked through the related TX Message Buffer Transmission Result flag bits (CFDTMSTSn.TMTRF[1:0]) in the TX Message Buffer Status Registers. The CFDTMSTSn.TMTRF[1:0] bits are set to 10b or 11b when the one-shot transmission is successful.
  The CFDTMSTSn.TMTRF[1:0] bits are set to 01b when arbitration is lost or an error occurs during the transmission of the related message buffer.
  Additional message transmission will not be attempted in this case.

The regular transmission request procedure after a configuration is shown in **Figure 7.9-41**.

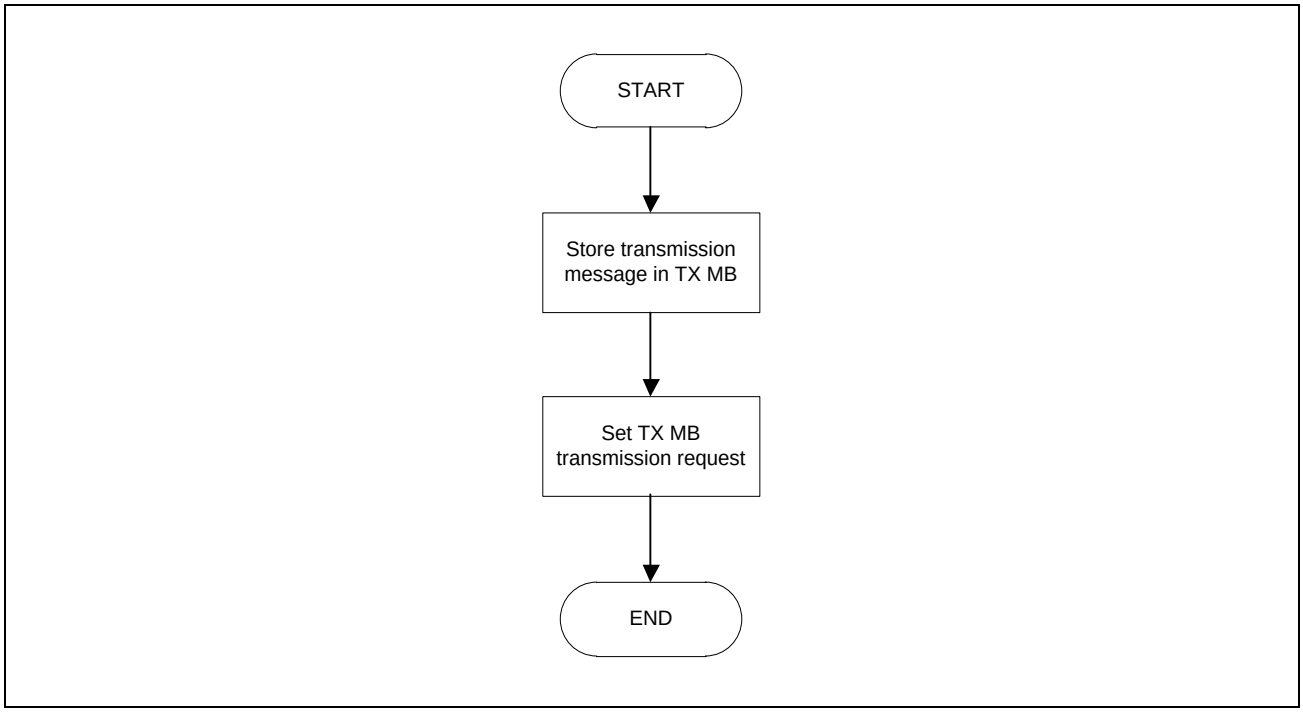


Figure 7.9-41 Transmission Request Procedure Using Normal TX Message Buffer Mode

(1) TX Message Buffer Control Register Setting

Table 7.9-27 shows configuration of the normal CAN transmission mode.

Table 7.9-27 Configuration of CAN Transmission Mode

Transmission Request CFDTMCn.TMTR	Transmission Abortion Request CFDTMCn.TMTAR	One Shot Enable CFDTMCn.TMOM	Communication Activity
0	0	0	Message buffer is disabled
0	0	1	Message buffer is disabled
1	0	0	Configured as a transmission message buffer for a data frame or a remote frame
1	0	1	Configured as a one shot transmission message buffer for a data frame or a remote frame
1	1	0	Transmission abortion is requested
1	1	1	One shot transmission abortion is requested

The configuration bits can be configured in the TX Message Buffer Control Registers.

Figure 7.9-42 shows timings for successful transmission for two message buffers of one channel.

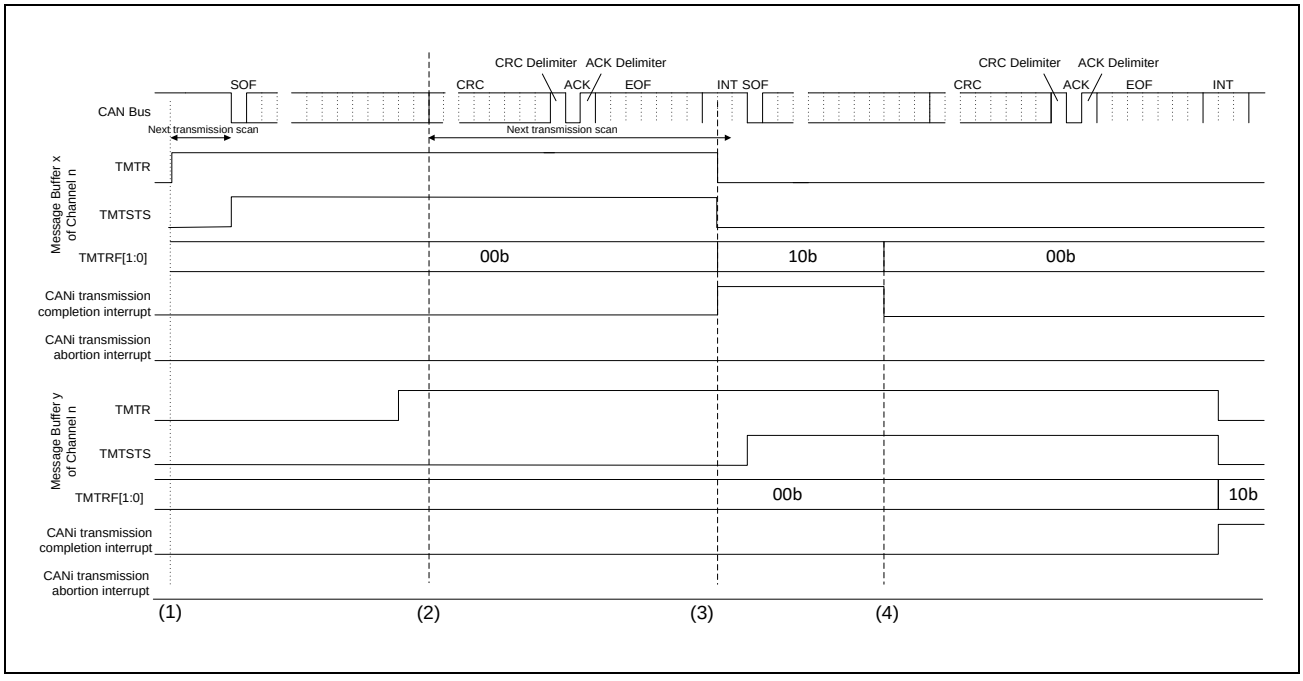


Figure 7.9-42 Timing of Request and Flag Bits for Successful Transmission

1. If the CFDTMCn.TMTR bit in the TX Message Buffer Control Registers is set in the bus idle state, message buffer scanning procedure starts to determine the highest priority message buffer for transmission.
   When the transmission message buffer is determined, the CFDTMSTSn.TMTSTS bit in the related TX Message Buffer Status Registers is set (Transmitting/Transmitter), and CAN channel starts the transmission*[1].
2. On the first bit of CRC, the transmission scanning procedure starts for the next possible transmission when pending transmission requests exist.
   The scan time can be delayed due to other transmission scan on other channels, but it will be finished before Intermission 3 to be able to continue transmission without any gaps.

3. If the message is successfully transmitted, the CFDTMSTSn.TMTRF[1:0] bits in the corresponding TX Message Buffer Status Registers are set to 10b and CFDTMSTSn.TMTSTS and the CFDTMCn.TMTR bits are cleared. When the TMIE bits in the TX Message Buffer Interrupt Enable Configuration Registers is set (interrupt enabled), the CAN successful transmission interrupt request is generated.
   To clear the related interrupt line the CFDTMSTSn.TMTRF flag bits must be cleared.
4. Before starting the next transmission, clear the CFDTMSTSn.TMTRF[1:0] bits. Load the next message in the transmission message buffer and set the CFDTMCn.TMTR bit again. CFDTMCn.TMTR bit cannot be set again before CFDTMSTSn.TMTRF[1:0] bits are cleared.

Note 1. If arbitration is lost after the CAN channel starts the transmission, the CFDTMSTSn.TMTSTS bit is cleared.

The transmission scanning procedure is performed again to search for the highest priority transmission message buffer from the beginning of the first CRC bit.

If an error occurs either during the transmission or following the loss of arbitration, then during Error Frame, the transmission scanning procedure is performed again to search for the highest priority transmission message buffer.

Note: The setting point of CFDTMSTSn.TMTSTS is not always fixed at the start of the SOF. It may be delayed up to the start of the standard ID due to the synchronization logic implemented for the PLL bypass.

Figure 7.9-43 shows timings for transmission abort for two message buffers of one channel.

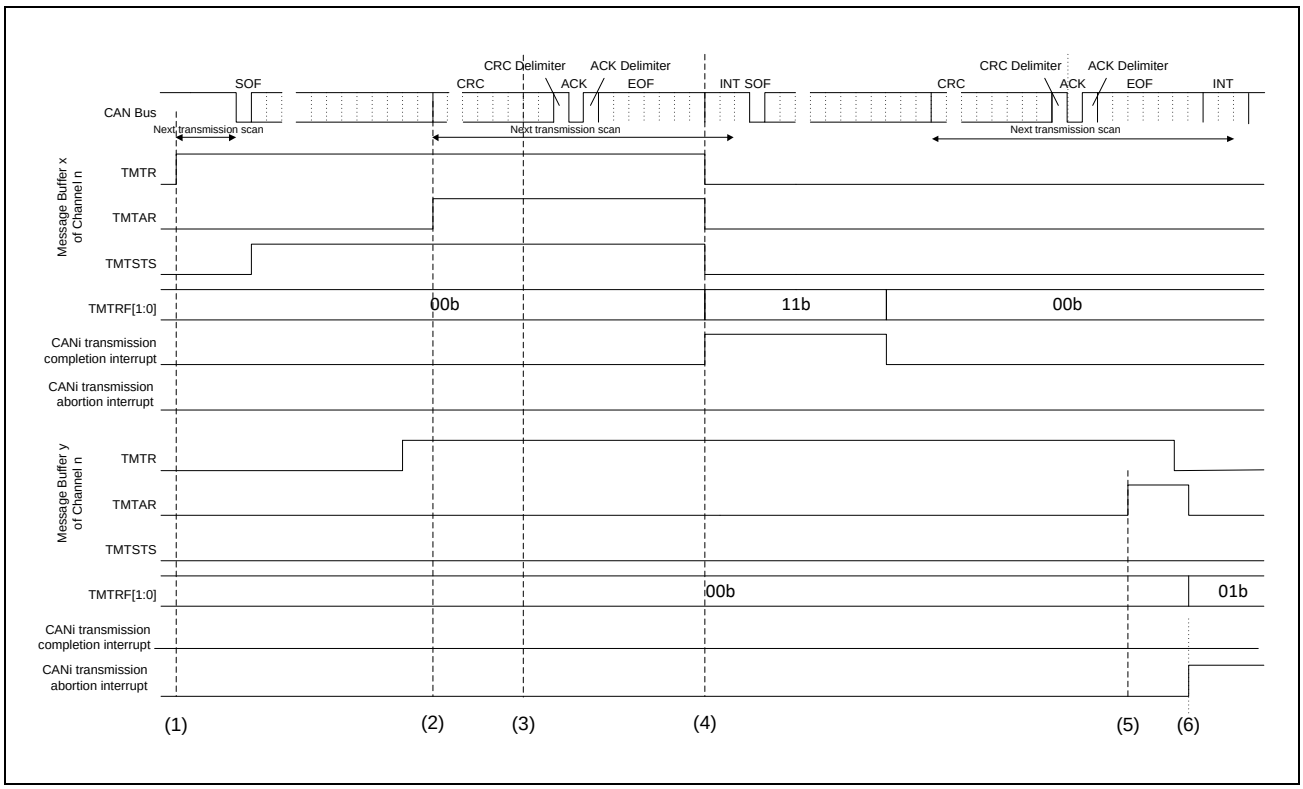


Figure 7.9-43 Timing of Request and Flag Bits for Transmission Abort

1. If the CFDTMCn.TMTR bit in the TX Message Buffer Control Registers is set in the bus idle state, message buffer scanning procedure starts to determine the highest priority message buffer for transmission.
   When the transmission message buffer is determined, the CFDTMSTSn.TMTSTS bit in the TX Message Buffer Status Registers is set (Transmitting/Transmitter), and CAN channel starts the transmission*[1].

2. If the CFDTMCn.TMTAR bit is set when the related message buffer is already selected for transmission or currently transmitting, the message is not aborted, if no error occurs or arbitration is lost.
3. On the first CRC bit, the transmission scanning procedure starts for the next transmission. In this example timing diagram, message buffer y is not selected as the next transmission message buffer.
   The scan time can be delayed due to other transmission scan on other channels, but it will be finished before Intermission 3 to be able to continue transmission without any gaps.
4. If the message is successfully transmitted, the CFDTMSTSn.TMTRF[1:0] bits in the corresponding TX Message Buffer Status Registers are set to 11b and CFDTMSTSn.TMTSTS and the CFDTMCn.TMTR bits are cleared. When the TMIE bits in the TX Message Buffer Interrupt Enable Configuration Registers is set (interrupt enabled), the CAN successful transmission interrupt request is generated.
   To clear the related interrupt line, clear the CFDTMSTSn.TMTRF[1:0] bits.
5. Another CAN node is transmitting on the CAN bus (CFDTMSTSn.TMTSTS not set). If the CFDTMCn.TMTAR bit is set when the related channel is under transmission scan, the transmission request cannot be cleared.
6. After internal processing time, the transmission is aborted and the CFDTMSTSn.TMTRF[1:0] bits are set to 01b. If the message buffer is not transmitting or selected as the next transmission message buffer or under transmit scan, then the abort is immediately accepted and the corresponding CFDTMSTSn.TMTRF[1:0] bits in the TX Message Buffer Status Registers are set to 01b.
   In addition, CFDTMCn.TMTR, and CFDTMCn.TMTAR bits are cleared automatically.
   When the transmission abort interrupt enable TAIE bit of the related Channel Control Register is set, an interrupt is generated for successful transmission abort.
   To clear the related interrupt line, clear the CFDTMSTSn.TMTRF[1:0] bits.

Note 1. If arbitration is lost after the CAN channel starts the transmission, the CFDTMSTSn.TMTSTS is cleared. The transmission scanning procedure is performed again to search for the highest priority transmission message buffer from the beginning of the first CRC bit. If an error occurs, either during the transmission, or following the loss of arbitration, then during Error Frame, the transmission scanning procedure is performed again to search for the highest priority transmission Message Buffer.

7.9.8.2.3 TX FIFO or GW FIFO Transmission

Three common FIFO buffers are assigned to each channel. The three FIFO buffers can be linked to any normal TX Message Buffer position for this channel by the CFDCFCCn.CFTML[4:0] bits in the Common FIFO Configuration/Control Register if configured in TX or GW mode.

When the transmission scan starts and the FIFO buffer corresponding to this TX Message Buffer is enabled, the relevant message in the FIFO buffer participates in the transmission scan.

Configuration of a TX Message Buffer linked to a FIFO buffer configured in TX or GW mode should not be done.

(1) TX FIFO Operation

CAN messages can be written into the TX FIFO by writing to the corresponding FIFO Access registers.

When the value FFh is written into the corresponding FIFO Pointer Control Register, the message count of the related FIFO is incremented by 1.

Only write to the FIFO Pointer Control register after writing the complete message to the corresponding FIFO Access registers.

If the message count matches the FIFO Depth, the FIFO Full flag is set.

The oldest message in the TX FIFO is included in the scan for transmission by the corresponding CAN-FD module channel logic.

When a message is successfully transmitted from the TX FIFO, the message count value is decremented by 1. When all the messages from the FIFO are transmitted, the FIFO Empty flag is set.

The interrupt generation conditions for the TX FIFO buffers can be configured by configuring the CFDCFCCn.CFIM bit in the corresponding Common FIFO Configuration/Control Registers.

If CFDCFCCn.CFIM bit is 0b, an interrupt is generated when the last message is successfully transmitted from the TX FIFO buffer.

If CFDCFCCn.CFIM bit is 1b, an interrupt is generated for every successfully transmitted message from the TX FIFO buffer.

Common FIFO can set interrupt, when CAN frame transmitted is completed.

The Common FIFO buffers configured in TX mode can be disabled by clearing the CFDCFCCn.CFE bit in the Common FIFO Configuration/Control Registers. If this bit is cleared to 0b, the FIFO Empty flag is set as follows:

- Immediately if the message from the TX FIFO is neither scheduled for the next transmission nor in transmission
- Following the transmission completion, the detection of an error on the CAN bus, loss of arbitration or transition to Channel or Global Halt mode if the transmission from the TX FIFO is already scheduled for transmission or already in transmission.

Note: The Common FIFO buffer is considered as disabled after clearing the CFDCFCCn.CFE bit only when the Empty flag is set for the corresponding Common FIFO Buffer.

Other possible messages pending from the TX FIFO are lost and their transmission must be requested again. Before CFDCFCCn.CFE is set again, ensure that CFDCFSTSn.CFEMP bit is set and that there is no pending abort from the TX FIFO.

When the CFDCFCCn.CFE bit is cleared, the message read and write pointers of the FIFO are cleared and are no longer active. Therefore, all messages in the FIFO buffers are lost and no further message can be stored into the FIFO.

The FIFO transmission request procedure after configuration is shown in **Figure 7.9-44**.

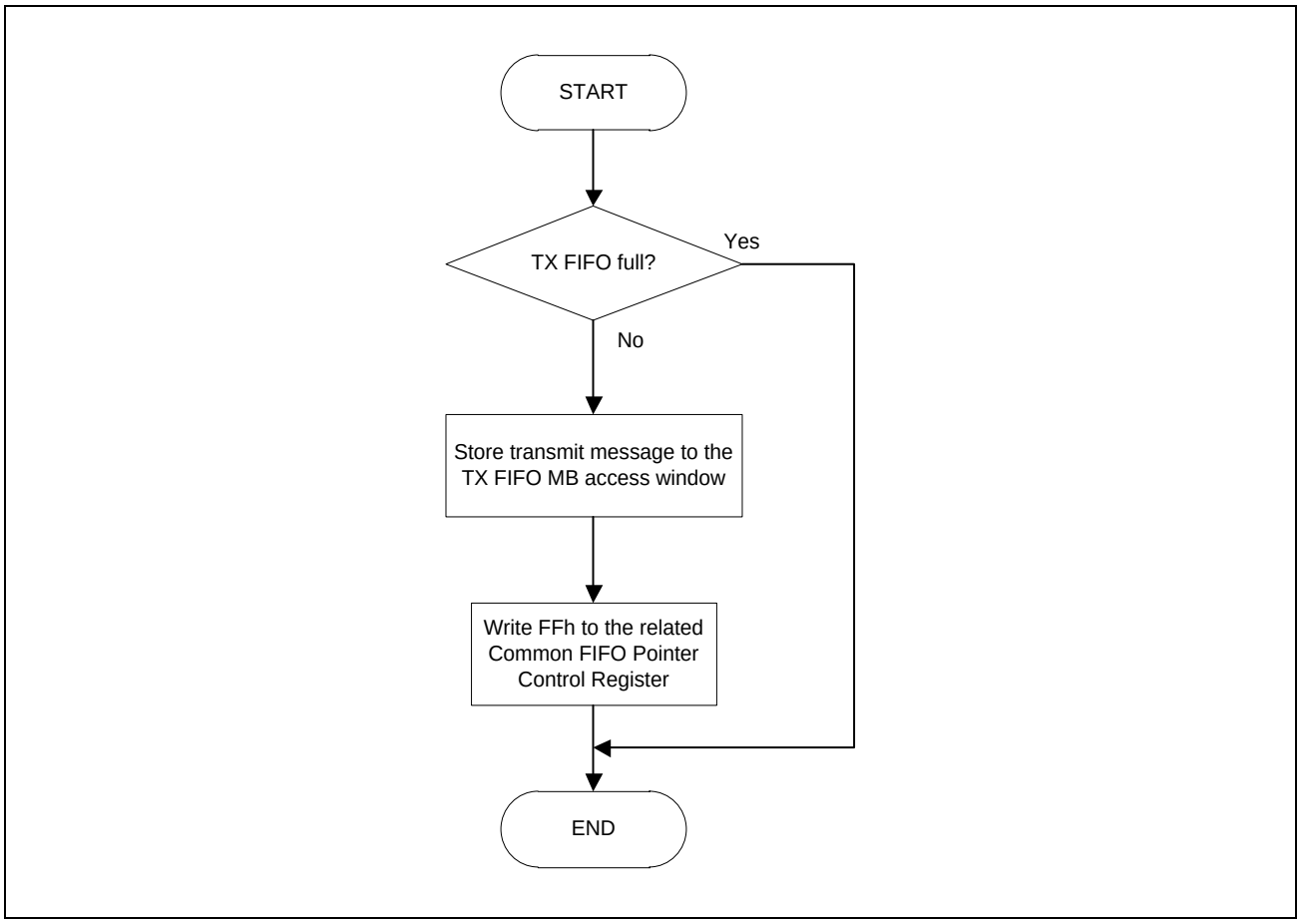


Figure 7.9-44 TX FIFO Transmission Request Procedure

(2) GW FIFO Operation

The AFL entries for routing the received messages to GW FIFO buffers should be configured based on the requirements of the system. The matching AFL entry selects the GW FIFO buffer for storage of a received message on any of the CAN channels.

When a message is successfully received and stored in a GW FIFO buffer, the FIFO message count in the corresponding FIFO Status Register is incremented by 1.

If the message count matches the FIFO depth, then the FIFO Full flag is set.

The oldest message in the GW FIFO is included in the scan for transmission by the corresponding CAN-FD module channel logic.

When a message is successfully transmitted from the GW FIFO, the message count value is decremented by 1. When all the messages from the GW FIFO are transmitted, the FIFO Empty flag is set.

If a new message is stored into the FIFO when the FIFO message count matches the FIFO depth (FIFO full condition), the FIFO Message Lost flag is set and the new message is lost (no overwrite of already stored messages takes place).

In GW mode, when a transmit/receive FIFO buffer is trying to receive a new message while the transmit/receive FIFO buffer is already full of data, the oldest data of the buffer is overwritten with the message received or the message is discarded. The behavior is determined by setting CFDCFCCEn.CFMOWM bit.

- When CFDCFCCEn.CFMOWM = 0b:
  When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the received message is discarded and CFDCFSTSn.CFMLT bit is set to 1b.
- When CFDCFCCEn.CFMOWM = 1b:
  When writing of data is required due to reception of a new message while a transmit/receive. FIFO buffer is full of data, the oldest data in the buffer is overwritten with the received message.
  The read pointer of the transmit/receive FIFO buffer simultaneously moves to the next oldest message.
  The CFDCFSTSn.CFMOW bit is then set to 1b, which notifies that the oldest message is overwritten with the received message.
  In addition, when a CAN bus error or arbitration-lost for the transmitting message occurs in transmit/receive FIFO buffer full, the transmitting message is lost and retransmission for the message is not performed. The read point moves to the next message automatically.

The interrupt generation conditions for the GW FIFO buffers can be configured by configuring the CFDCFCCn.CFIM bit in the corresponding Common FIFO Configuration/Control Registers.

If CFDCFCCn.CFIM bit is 0b, then RX interrupt flag is set when FIFO counter increments and reaches value configured by CFDCFCCn.CFIGCV and the TX interrupt flag is set when FIFO transmits the last message successfully.

If CFDCFCCn.CFIM bit is 1b, then RX interrupt flag is set at the end of storage of every received message and TX interrupt flag is set if a message is successfully transmitted from the FIFO.

Common FIFO can set interrupt when:

- CAN frame transmitted is completed
- CAN frame reception is completed
- FIFO is in full status in RX mode or GW mode

When CFDCFCCEn.CFBME = 1b, it becomes FIFO buffering mode, send data is stored in Common FIFO, and transmission is stopped. Transmission will be started if it is set as CFDCFCCEn.CFBME = 0b.

The Common FIFO buffers configured in GW mode can be disabled by clearing the CFDCFCCn.CFE bit in the Common FIFO Configuration/Control Register. If this bit is cleared, the GW FIFO becomes empty as follows:

- Immediately if the message from the GW FIFO is neither scheduled for the next transmission nor in transmission
- Following the transmission completion, the detection of an error on the CAN bus, loss of arbitration or transition to Channel or Global Halt mode if the transmission from the GW FIFO is already scheduled for transmission or already in transmission

Other possible messages pending from the GW FIFO are lost.

Before CFDCFCCn.CFE is set again, ensure that the CFDCFSTSn.CFEMP bit is set and that there is no pending abort from the GW FIFO.

When the CFDCFCCn.CFE bit is cleared and the CFDCFSTSn.CFEMP bit is set, the message read and write pointers of the GW FIFO are cleared and are no longer active. Therefore, all messages in the GW FIFO buffers are lost and no further message can be stored into the GW FIFO.

In applications intended to be used as CAN-to-CAN gateways, it is useful if the Error State Indication (ESI) information of the routing messages is not replaced by the sending node Error State Indication. For this, each channel has the control function register CFDCnFDCFG.ESIC to replace their own ESI information by the routing ESI information.

Note: If the sending node is error passive, the ESI bit is sent anyway as error passive (ESI = 1b).

(3) Interval Timer for FIFO Transmission

For each Common FIFO in TX or GW mode, it is possible to specify a delay between two consecutive messages that are configured for transmission from the same FIFO buffer. This delay is called interval time. This interval time starts after the first message has been successfully transmitted from the FIFO buffer after the CFDCFCCn.CFE bit is set.

When the Common FIFO in TX or GW mode is enabled, the first message is transmitted without considering this interval time.

The interval timer stops counting when:

- FIFO is disabled by clearing the CFDCFCCn.CFE bit
- CAN channel is in CH_RESET mode

The interval time is specified by the CFDCFCCn.CFITT value in the Common FIFO Configuration/Control Register and can be specified from 0 to 255 timer units.

The timer unit can be defined based on two different source clocks for the interval timer. To disable the interval timer for FIFO transmission, a value of 0 should be selected.

The timer source can be selected by the configuration bit CFITSS in the Common FIFO Configuration/Control Register. For the timer source the CAN bit timing clock of the FIFO related channel or a global reference clock can be selected.

If CAN channel bit time clock is configured as clock source and the CAN channel enters CH_HALT or CH_RESET or CH_SLEEP mode, the interval timer is stopped for that channel.

If peripheral clock is selected as interval timer clock source, then the interval timer is stopped only when the CAN channel is in CH_RESET or CH_SLEEP mode.

The reference clock can be used to configure the interval time in fixed time units. It is based on the peripheral clock. The reference clock prescaler value CFDGCFG.ITRCP in the Global Configuration Register defines the relation between the peripheral clock frequency/period and the reference clock period.

See **Table 7.9-28** for CFDGCFG.ITRCP configuration values to achieve different reference clock periods based on the peripheral clock frequency/period.

Table 7.9-28 Configuration Example for the FIFO Interval Timer Reference Clock

Peripheral Clock (CANFD_0_pclk)	Reference Clock		
	1 µs	100 µs	500 µs
75 MHz / 13.33 ns	75	7500	37500
100 MHz / 10 ns	100	10000	50000

Additionally, the reference clock resolution can be specified by the interval timer reference clock resolution value CFDCFCCn.CFITR in the Common FIFO Configuration/Control Register.

The interval time is based on the reference clock period multiplied by the configured value (×1 or ×10).

The reference clock based interval timer can be used to follow the requirements of the ISO 15765-2 Separation Time. The whole range for the separation time from 100 µs to 127 ms can be covered.

The specified interval time starts after successful transmission event (after EOF7 state of the CAN protocol).

When the interval time has elapsed, the next transmission request is raised by the related TX/GW FIFO. Therefore, the interval time defines the minimum time between two messages transmitted from one FIFO.

The next message is sent at earliest after this interval time.

Figure 7.9-45 shows an example timing of the internal processing.

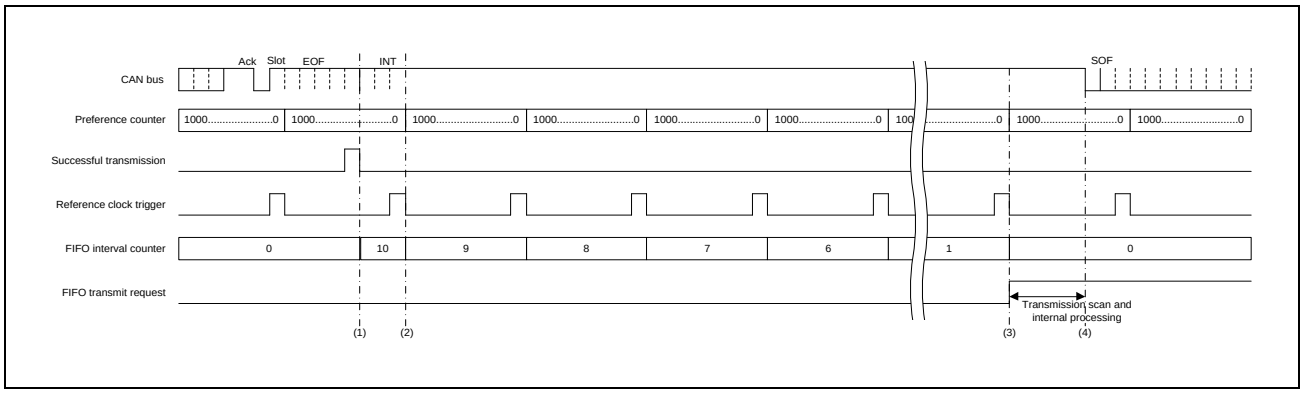


Figure 7.9-45 Example for Interval Processing Time

The configuration for the above timing is as follows:

- Peripheral clock frequency = 100 MHz
- Interval timer reference clock (CFDGCFG.ITRCP) = 1000 times
- Reference clock due to the above settings = 10 µs
- Common FIFO interval timer source selection (CFDCFCCn.CFITSS) = 0b
- Common FIFO interval timer resolution (CFDCFCCn.CFITR) = 0b
- Common FIFO interval transmission time (CFDCFCCn.CFITT) = 10 times
- Theoretical message separation interval = 100 µs

1. Internal FIFO interval timer is restarted with the occurrence of successful transmission result. This restart is not synchronized to the reference clock trigger. Therefore, the first interval is counting less or equal to one reference clock interval.
2. With the next reference clock trigger, the FIFO interval timer is decremented.
3. When the FIFO interval timer reached the value 0 the FIFO transmit request is set.
4. When the FIFO is selected for transmission, the transmission starts. Due to internal processing this usually takes less than 3 CAN bit time, between internal FIFO transmit request set 3 and actual transmission.

In the worst case when multi events such as reception scan, internal message routing, transmit scan on all channels occur, then it can take up to 432 peripheral clock cycles.

As shown in **Figure 7.9-45**, it is not guaranteed that the minimum interval is always equal to the configured value. If a minimum time must never be breached, configure CFDCFCCn.CFITT to the required minimum value + 1.

If additional TX message buffers or TX/GW FIFOs are configured for transmission for the same channel, the real delay between two messages transmitted from a TX FIFO can be much longer than specified by the interval time due to higher priority message transmission from these TX message buffers or TX/GW FIFOs.

Figure 7.9-46 shows a block diagram of the FIFO interval time generation circuit.

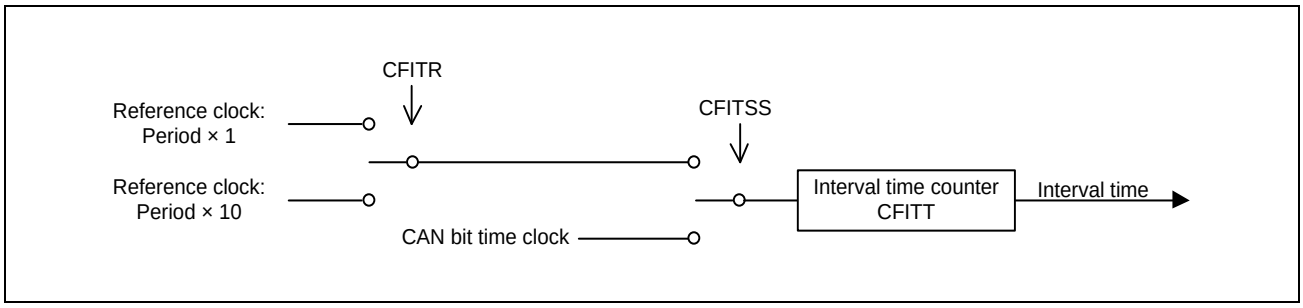


Figure 7.9-46 Block Diagram of FIFO Interval Timer

7.9.8.2.4 TX Queue

Each enabled TX Queue for a specific channel consists of 3 to 32 TX message buffers, which are accessed by one access window. One channel has 4 TX Queues. One TX Queue can be configured with a depth of 3 up to 32 buffers and it uses TX Message Buffer No. 0 as access window (referred to as TXQ0). The second TX Queue can be configured with a depth of 3 up to 32 buffers and it uses TX Message Buffer No. 31 as access window (referred to as TXQ1). The third TX Queue can be configured with a depth of 3 up to 32 buffers and it uses TX Message Buffer No. 32 as access window (referred to as TXQ2). The fourth TX Queue can be configured with a depth of 3 up to 32 buffers and it uses TX Message Buffer No. 63 as access window (referred to as TXQ3).

All TXQ0, TXQ1, TXQ2, and TXQ3 messages enter the priority comparison for the transmission, which should be only ID Priority (CFDGCFG.TPRI = 0b).

The registers for TXQ0 are CFDTXQCC0n, CFDTXQSTS0n, and CFDTXQPCTR0n.
The registers for TXQ1 are CFDTXQCC1n, CFDTXQSTS1n, and CFDTXQPCTR1n.
The registers for TXQ2 are CFDTXQCC2n, CFDTXQSTS2n, and CFDTXQPCTR2n.
The registers for TXQ3 are CFDTXQCC3n, CFDTXQSTS3n, and CFDTXQPCTR3n.

When access window TX Message Buffer No.63 (TXQ3) or TX Message Buffer No.32 (TXQ2) or TX Message Buffer No.31 (TXQ1) or TX Message Buffer No.0 (TXQ0) is used, refer to the related access registers TX Message Buffer ID Registers (CFDTMIDn), TX Message Buffer Pointer Registers (CFDTMPTRn), TX Message Buffer Data Field 0 Registers (CFDTMDF0_n), and TX Message Buffer Data Field 1 Registers (CFDTMDF1_n).

The depth of each TXQ0 buffer can be configured by writing to the CFDTXQCC0n.TXQDC[4:0] bits of the TX Queue Configuration/Control Register.

TXQ0 can set from TXMB0 to TXMB15 as a queue buffer at the maximum.

The 15 available options for depth configuration are:

- 00h: TX Queue disabled
- 01h: Reserved
- 02h: 3 messages
  ⋮
- 0Dh: 14 messages
- 0Eh: 15 messages
- 0Fh: 16 messages

The depth of each TXQ1 buffer can be configured by writing to the CFDTXQCC1n.TXQDC[4:0] bits of the TX Queue Configuration/Control Register.

TXQ1 can set from TXMB15 to TXMB0 as a queue buffer at the maximum.

The 15 available options for depth configuration are:

- 00h: TX Queue disabled
- 01h: Reserved
- 02h: 3 messages
  ⋮
- 0Dh: 14 messages
- 0Eh: 15 messages
- 0Fh: 16 messages

The depth of each TXQ2 buffer can be configured by writing to the CFDTXQCC2n.TXQDC[4:0] bits of the TX Queue Configuration/Control Register.

TXQ2 can set from TXMB32 to TXMB47 as a queue buffer at the maximum.

The 15 available options for depth configuration are:

- 00h: TX Queue disabled
- 01h: Reserved
- 02h: 3 messages
  ⋮
- 0Dh: 14 messages
- 0Eh: 15 messages
- 0Fh: 16 messages

The depth of each TXQ3 buffer can be configured by writing to the CFDTXQCC3n.TXQDC[4:0] bits of the TX Queue Configuration/Control Register.

TXQ3 can set from TXMB47 to TXMB32 as a queue buffer at the maximum.

The 15 available options for depth configuration are:

- 00h: TX Queue disabled
- 01h: Reserved
- 02h: 3 messages
  ⋮
- 0Dh: 14 messages
- 0Eh: 15 messages
- 0Fh: 16 messages

When using TXQ1 and TXQ0 simultaneously, the depth of TXQ is 16 or less in total.

When using TXQ3 and TXQ2 simultaneously, the depth of TXQ is 16 or less in total.

Do not access all the TX message buffers forming the TX Queue directly (except TX Message Buffer No. 63, TX Message Buffer No. 32, TX Message Buffer No. 31 and TX Message Buffer No. 0, which acts as TX Queue access window).

When CFDGAFLP0n.GAFLSRD i (i = 0 to 2) is set and the CFDTXQCCin.TXQGWE (i = 0 to 2, n = 0, 1) is also set, a receiving frame is stored in the target TXQ as send data by routing.

When CFDTXQCCn.TXQOWE bit is 1b, the TX Queue is in TX Queue overwrite mode. If the message of the same ID is stored in TX Queue when a frame is received and it is stored in TX Queue, an old message is overwritten by a new message. Therefore, an old message is not transmitted. When the old message of the same ID is transmitting and a CAN bus error and an arbitration-lost occur, the message of old ID is not resent.

When using the function in GW mode and TX Queue overwrite mode, the depth of TXQ (CFDTXQCC0n.TXQDC) should be configured to the value which is the various number of ID that is used in the TX Queue plus 3. If it accesses by routing in gateway mode when a TXQ buffer is full, CFDTXQSTS.TXQMLT is set and send data is thrown away.

The function is valid for the standard ID frame and is invalid for the extended ID frame.

Explanation of operation of the TX Queue with same ID over-writing function in GW mode is shown in **Figure 7.9-47**.

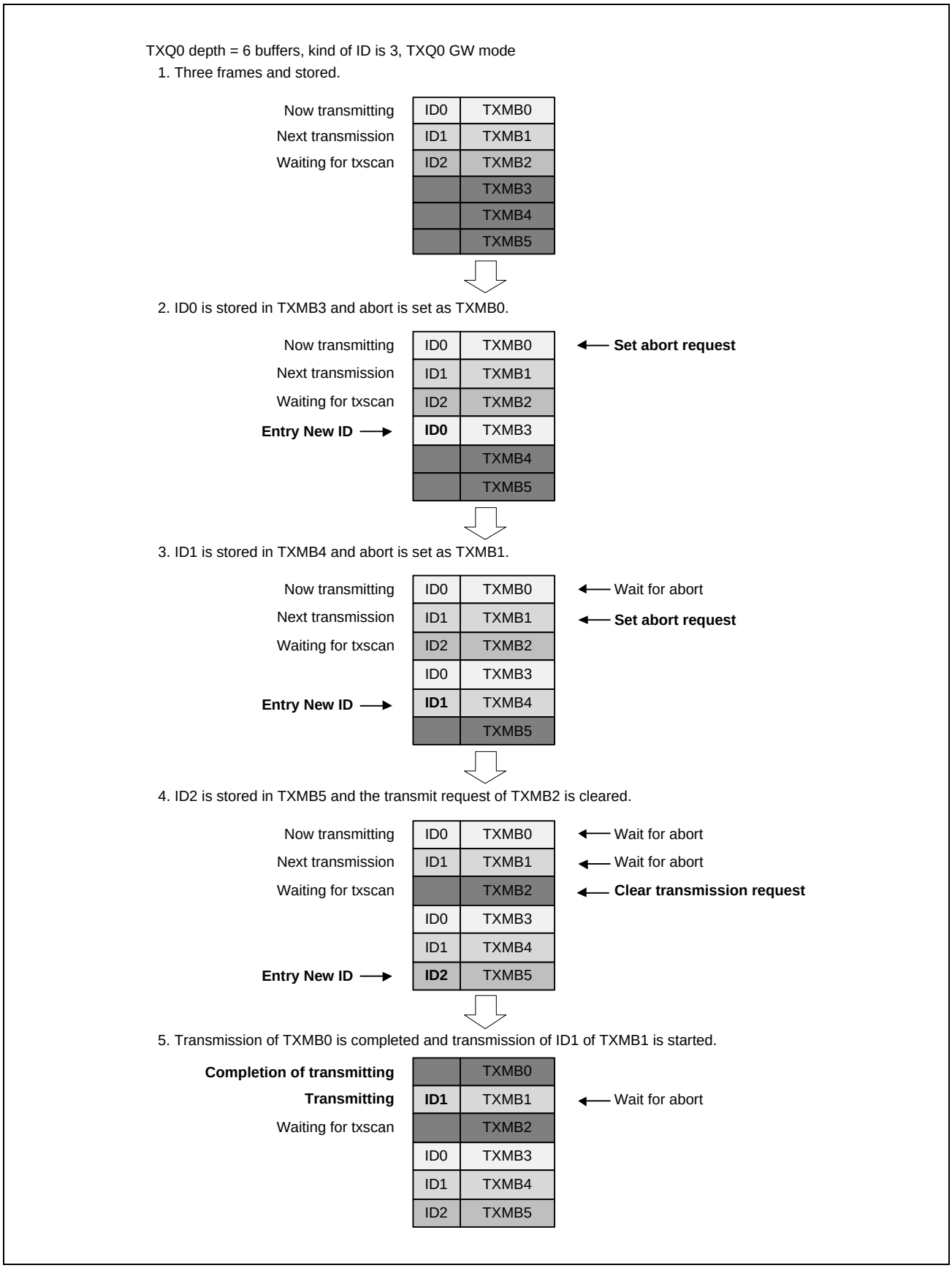


Figure 7.9-47 Operation of TXQ0 in Gateway Mode

When a system writes in TXQ, a system should write in send data, after checking the state of TXQ.

Do not access or configure the related TX Message Buffer Control Registers.

The messages stored to the TX Queue access window are internally stored to a free buffer of the TX Queue.

When the buffer is full then no further access should be done to the queue, until it is no longer full. If it accesses by software writing in when the buffer of TXQ is full, send data is overwritten.

The TX Queue can be disabled by clearing the TXQE bit in the TX Queue Configuration/Control Register. If this bit is cleared, the TX Queue Empty flag is set as follows:

- Immediately if the message from the TX Queue is neither scheduled for the next transmission nor in transmission
- Following the transmission completion, the detection of an error on the CAN bus, loss of arbitration or transition to Channel or Global Halt mode if the transmission from the TX Queue is already scheduled for transmission or already in transmission.

Note: The TX Queue is disabled only when the Empty flag is set after clearing the TXQE bit for the corresponding TX Queue.

Other possible messages pending from the TX Queue are lost and their transmission must be requested again.

Before TXQE is set again, ensure that the CFDTXQSTSn.TXQEMP bit is set and that there is no pending abort from the TX Queue.

When the TXQE bit is cleared, all messages in the TX Queue buffers are lost and no further message should be stored into the TX Queue.

When a message has been stored to the TX Queue, FFh must be written into the TX Queue Pointer Control Register. This sets the transmit request automatically and changes the internal message buffer pointer to the next free message buffer location of the TX Queue.

Note: If two messages with the same identifier are stored in the TX Queue, then the order of transmission of these messages can be different from the order in which they were stored in the TX Queue.

To avoid this condition, it is important to confirm that the previous message with the same ID was successfully transmitted before a new message with the same identifier is stored in the TX Queue. Or if TX queue overwrite mode is used, the frame of the same ID is rewritten on a new frame.

For the TX Queue a dedicated interrupt can be enabled by setting the TXQIE bit of the TX Queue Configuration/Control Register.

The interrupt mode can be configured with the CFDTXQCCn.TXQIM bit of the same register either to generate an interrupt for every transmitted message or for the last transmitted message.

The TX Queue transmission request procedure after configuration is shown in **Figure 7.9-48**.

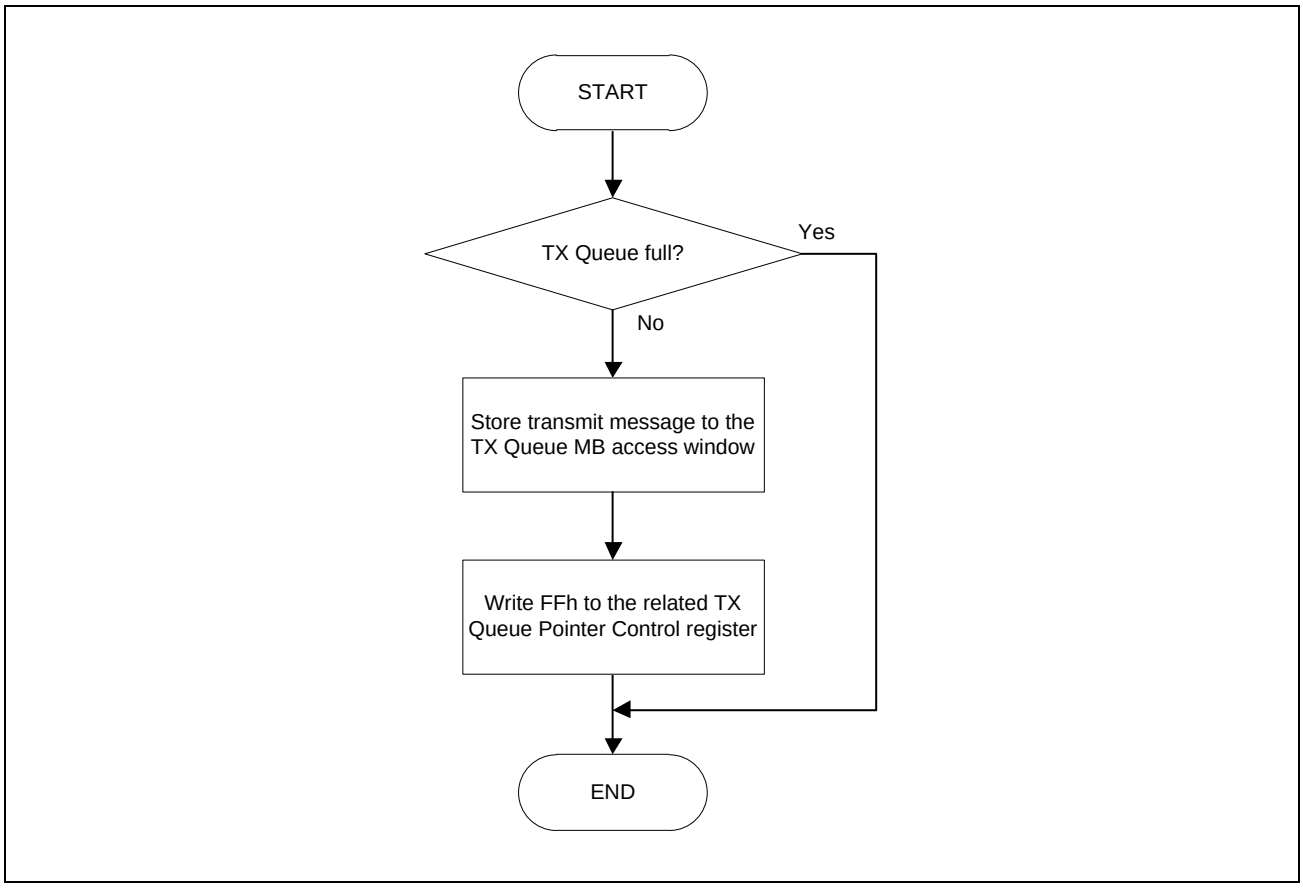


Figure 7.9-48 TX Queue Transmission Request

TXQ Name	Access Window	Range Width	Direction	Hardware Routing Access Point	CPU Access Point	DMA Access Point	Note
TXQ0	TXMB0	0, 3-16	TXMB0 → TXMB15	Yes	Yes	Yes	When using both TXQ0/1, the total number of stages is 16 or less
TXQ1	TXMB31	0, 3-16	TXMB15 → TXMB0	Yes	Yes	No	
TXQ2	TXMB32	0, 3-16	TXMB32 → TXMB47	Yes	Yes	No	When using both TXQ2/3, the total number of stages is 16 or less
TXQ3	TXMB63	0, 3-16	TXMB47 → TXMB32	No	Yes	Yes	

TXQ0 can use hardware routing, CPU access, and DMA access.

Hardware routing access, CPU access, and DMA access should not be used simultaneously. Choose one access method.

7.9.8.2.5 TX History List

The TX History List function records the information of the successfully transmitted message in the TX History List buffers for each CAN channel. Two TX History List buffers are provided for each CAN channel and each TX History List buffer can store up to 16 TX History List entries for a CAN channel.

The CFDTHLCCn.THLDTE bit of the TX History List Configuration/Control Register can be used to configure if only message information from TX FIFOs/TX Queue is stored or if all transmit message information from TX Queue, TX FIFO or normal TX message buffers should be stored in the TX History List for a CAN channel.

When a CFDTHLCCn.THLDGE bit is set up, the information on all the frames transmitted in GW mode is stored in TX History List.

Each transmit message can be individually configured for acceptance to the TX History List by the CFDCFID.THLEN bit in the Message Buffer Pointer Register.

The message information is stored to the TX History List Buffer of a CAN channel after the message is successfully transmitted on that CAN channel.

Storing to the List is not synchronized with the status of CFDTMSTSn.TMTRF[1:0] bits in the TX Message Buffer Status Register.

Due to internal processing, the storage to the List can happen with a delay after the successful transmission indication.

Storing the TX History List data can be recognized by the condition that the THLIF bit is set to 1b when the THLIE is configured to 1b or when the TX History List counter CFDTHLSTSn.THLMC[5:0] is increased.

The delay time is dependent on the number of channels due to internal processing.

- Maximum delay time from setting the CFDTMSTSn.TMTRF to storing the TX History List data is 224 peripheral bus clock cycles.

The History list records the following information of the transmitted message:

- Buffer type:
  - 001b: TX Message Buffer
  - 010b: TX FIFO
  - 100b: TX Queue
- Buffer number:
  TX Message Buffer, TX Queue Message Buffer or TX Message Buffer Link for the Common FIFO Buffer from which the transmission occurred. The number depends on the buffer type, see **Table 7.9-29**.
- Transmission ID:
  Transmission pointer stored in the transmission message
- Transmit timestamp:
  Message timestamp captured at the capture point as configured by CFDGFDCFG.TSCCFG.
- Transmission information label:
  Transmission information label stored in the transmission message.
- Transmit gateway buffer indication:
  For data transmitted from the gateway, CFDTHLACC0n.TGW bit is set to 1b.

Table 7.9-29 TX History List Buffer Number Entry

CFDTHLACC0n.BT[2:0] Buffer Type		
001b	101b	100b
TX Message Buffer	TX FIFO	TX Queue
TXMB0	—	Number shown corresponds to the message buffer belonging to the TX Queue for which the frame was transmitted.
TXMB1		
TXMB2		
⋮		
TXMB13		
TXMB14		
TXMB15		
TXMB32	Number shown corresponds to the common FIFO. TX Message Buffer Link CFTML of the related Common FIFO Configuration	
TXMB33		
TXMB34		
⋮		
TXMB45		
TXMB46		
TXMB47		

The Transmission ID entry is used to identify which message of a TX FIFO or TX Queue has been successfully transmitted because the TX FIFO or TX Queue number alone is not sufficient.

Therefore, a unique number can be attached to each transmission message stored in a TX FIFO or TX Queue. This unique identification number should be written to the CFDCFFDCSTSn.CFPTR[15:0] part of the Common FIFO Access Pointer Register for a TX FIFO or to the CFDTMFDCTRn.TMPTR[15:0] part of the TX Message Buffer Pointer Register of the TX Queue access window message buffer.

When the message is successfully transmitted, the identification number is then stored together with the other message related information to the TX History List and can be read with the Transmission ID (TID) of the TX History List Access Register.

Also for normal TX message buffers, the CFDTMFDCTRn.TMPTR[15:0] part of the TX Message Buffer Pointer Register is stored in the Transmission History List. Information label is the same.

Figure 7.9-49 shows a transmission preparation flow when TX History List is used.

Read access to the TX History List Access Register is done for every single entry.

After reading one entry, FFh must be written to the corresponding TX History List Pointer Control Register to be able to access the next entry until TX History List is empty.

Figure 7.9-50 shows an example flow for processing the TX History List information.

The TX History Lists have dedicated interrupts, which can be configured with the CFDTHLCCn.THLIM bit of the corresponding TX History List Configuration/Control Registers and enabled with the CFDTHLCCn.THLIE bit of the same registers, either to generate an interrupt when the history list reached a filling level of 75% or for every new TX History List entry.

An entry lost indication is flagged by the CFDTHLSTSn.THLELT bit in the TX History List Status Register.

Status of this bit is also shown by the THLES bit in the Global Error Flag Register.

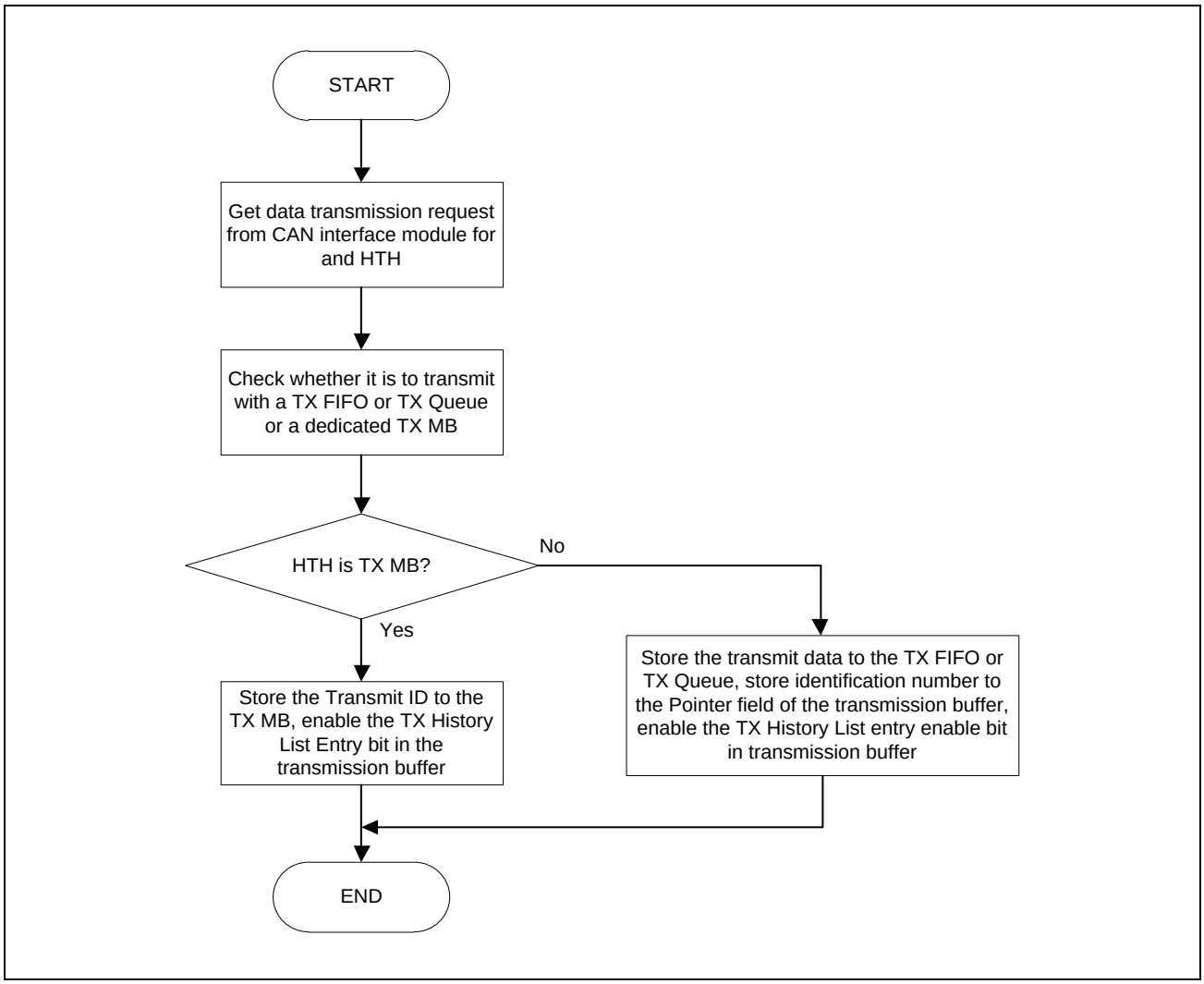


Figure 7.9-49 Transmission Preparation Flow when TX History List is Used

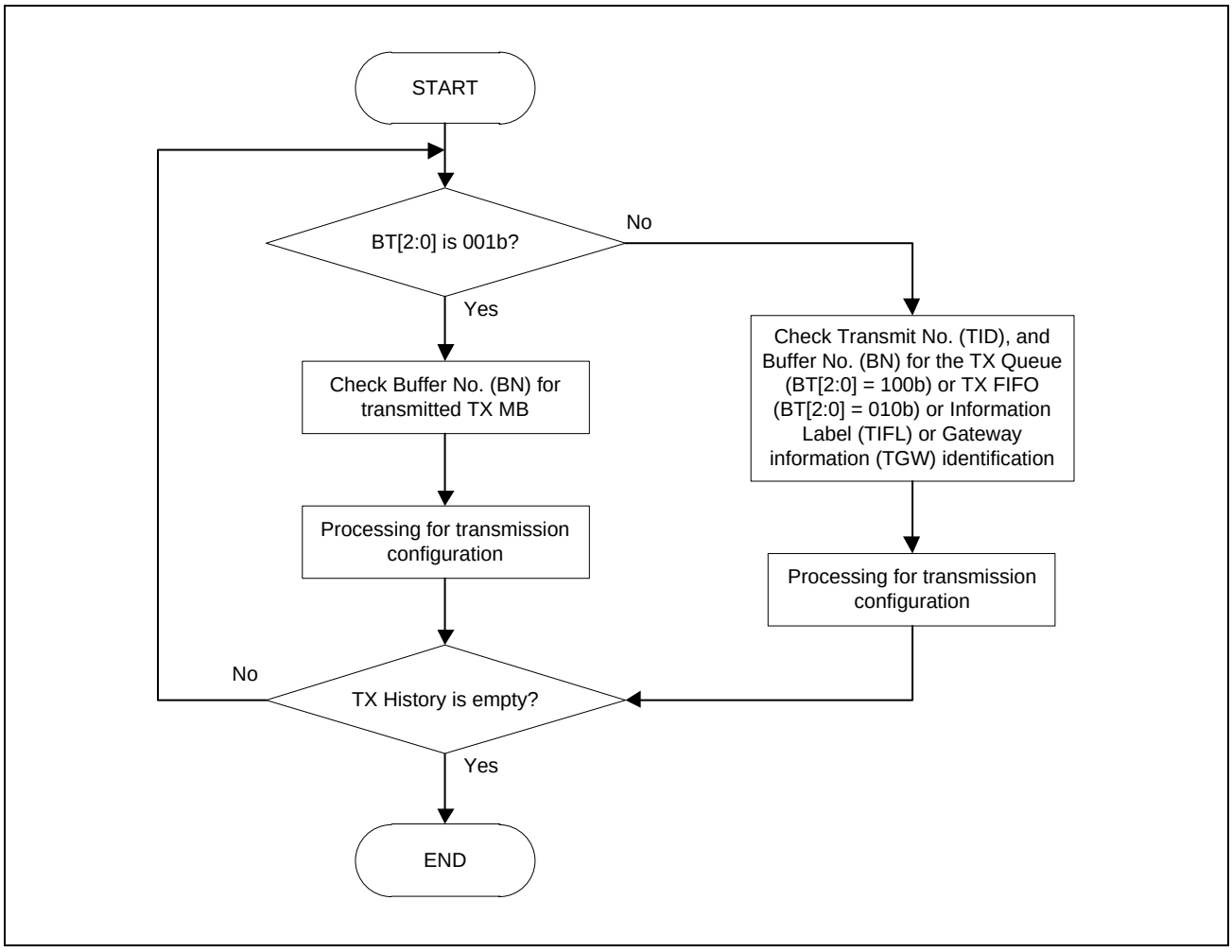


Figure 7.9-50 Example Flow for Processing TX History List Information

7.9.8.2.6 TX Data Padding

If the data length code (DLC) of the transmitting message has a higher number of data bytes than the buffer size, then the data bytes beyond the restricted range are replaced by bytes with the value of CC HEX.

This can happen for Common FIFOs configured as (TX or GW) when the transmit message DLC is higher than CFDCFCCn.CFPLS.

This can also happen in FD-only mode, if a Classical Frame is configured with a DLC larger than 8.

7.9.9 Test Mode

The CAN-FD module can be configured into test modes to allow testing of certain features. These features are provided only for special purposes and care must be taken when configuring the CAN-FD module in test modes.

All test modes are mutually exclusive unless it is explicitly stated that some functions can be enabled across other test modes.

Do not enable any combinations of the various test modes specified in this section.

The test modes can be broadly split into 2 groups:

- Channel specific test modes
- Global test modes

7.9.9.1 Channel Specific Test Modes

Each CAN channel can be configured into the following test modes:

- Basic test mode
- Listen-only mode
- Self-test mode 0 (External Loop Back mode)
- Self-test mode 1 (Internal Loop Back mode)
- Restricted operation mode

7.9.9.1.1 Basic Test Mode

The Basic test mode should be used when a particular test setting must be enabled other than when in Listen-Only and Self- test modes.

7.9.9.1.2 Listen-Only Mode

The ISO 11898-1 recommends an optional Bus-monitoring mode. In this mode, the CAN channel is able to receive valid data frames and valid remote frames. However, it sends only recessive bits on the CAN bus and is not allowed to transmit.

If the CAN engine is required to send a dominant bit (ACK bit, Overload flag, Active Error flag), the bit is routed internally so that the CAN engine monitors this as dominant. The external TX pin remains in recessive state.

This mode can be used for baud rate detection. In this mode, an error interrupt is generated if a bus error occurs and the interrupt is enabled.

In this mode, it is not permitted to request transmission from any normal TX Message Buffer or TX-/GW-FIFO of this channel.

Note: If a message is stored in GW FIFO or Routing TXQ, ensure that the transmitting channel is not in Listen-only mode so that transmission is not requested for this channel from the GW FIFO or Routing TXQ.

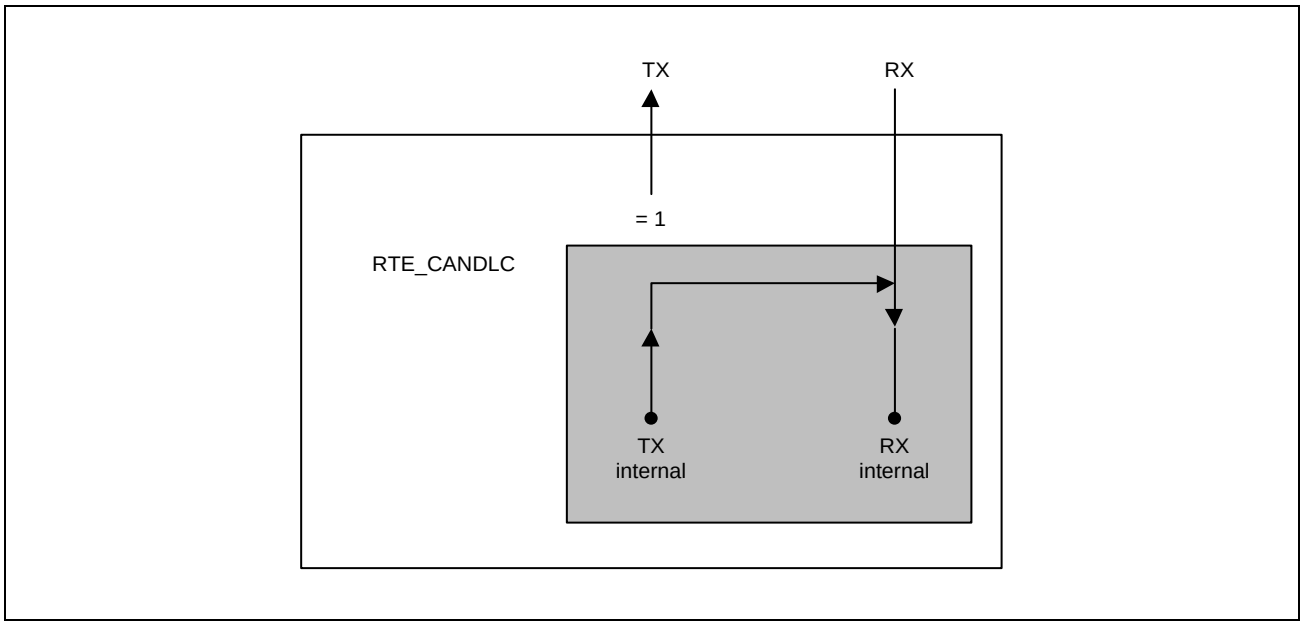


Figure 7.9-51 Listen-Only Mode

7.9.9.1.3 Self-Test Mode 0 (External Loop Back Mode)

In Self-test mode 0, the CAN engine treats its own transmitted messages as received messages through the CAN transceiver and can store them into its receive message buffers.

To be independent from external stimulation, the engine generates its own Acknowledge bit.

This test can be used for CAN transceiver tests.

The RX/TX pins should be connected to the transceiver.

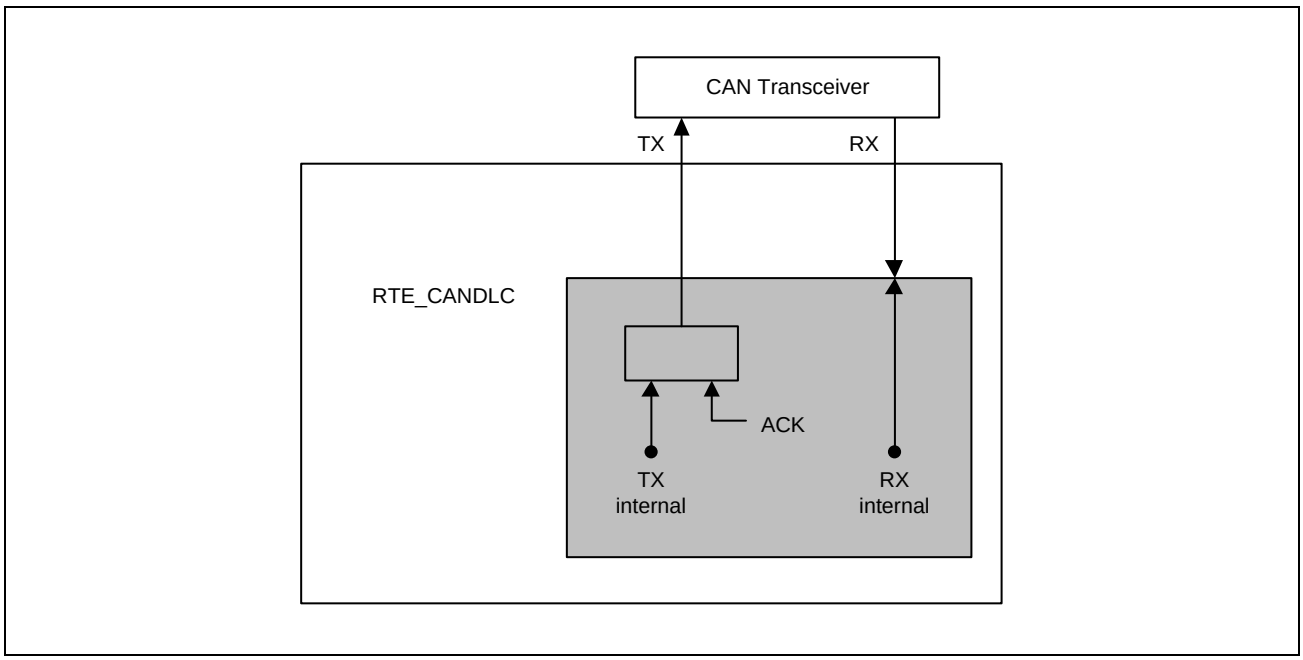


Figure 7.9-52 Self-Test Mode 0 (External Loop Back Mode)

7.9.9.1.4 Self-Test Mode 1 (Internal Loop Back Mode)

In Self-test mode 1, the CAN engine treats its own transmitted messages as received messages and stores them into the receive buffer. This mode is provided for self-test functions. To be independent from external stimulation, the CAN engine generates its own Acknowledge bit. In this mode the CAN engine performs an internal feedback from TX internal to RX internal. The actual value of the external RX input is disregarded by the CAN engine.

The external TX pin outputs only recessive bits.

The RX/TX pins do not need to be connected to the CAN bus or any external device.

Note: The channel pins are also disconnected from the internal CAN bus communication line.

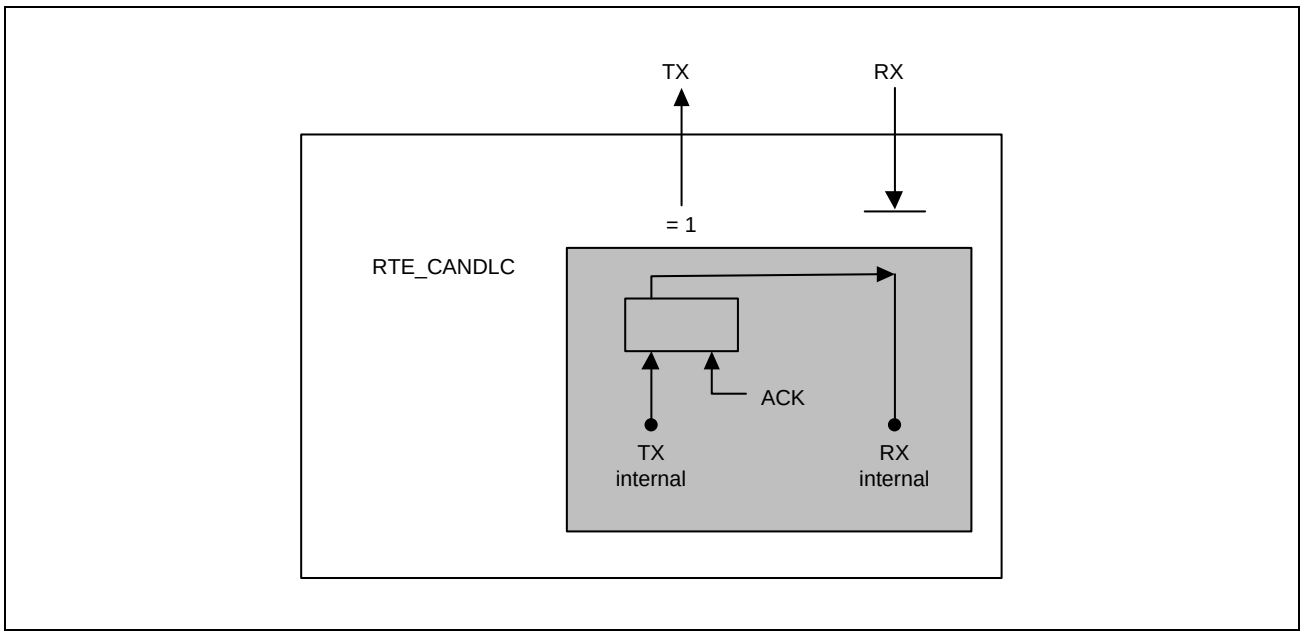


Figure 7.9-53 Self-Test Mode 1 (Internal Loop Back Mode)

7.9.9.1.5 Restricted Operation Mode

In Restricted operation mode, the CAN node is able to receive valid data and remote frames that generates the Acknowledge bit.

Active error and overload frames cannot be transmitted, instead it waits for the occurrence of bus idle condition to resynchronize itself to the CAN communication after an error or overload condition occurs.

Additionally, the Receive and Transmit Error Counter (REC and TEC) are frozen independently from the occurrence of errors.

The mode is specified as in ISO 11898-1. However, it is permitted to set any requested transmit.

7.9.9.2 Global Test Modes

The CAN-FD module can be configured into the following test modes:

- RAM test mode
- Internal CAN bus communication mode
- CRC error test

These test modes are protected by a special software procedure to enable the mode. This software procedure enables write access to the test mode by a specific unlock key, the related unlock key can be seen in the following table.

Test Mode	Unlock Key 1	Unlock Key 2
RAM test mode	7575h	8A8Ah

If the software sequence of the two consecutive unlock key write accesses (half-word or word accesses) is interrupted by any other write access to the SFR or if incorrect data is written to the Global Unlock Key Register then the corresponding test mode cannot be set and the sequence must be restarted.

After the two unlock key write accesses, the next write access should be to set the corresponding Test Mode Enable bit. If this is not followed, the unlock mechanism resets and the Test Mode Enable bit cannot be set. At this time, the unlock sequence must be restarted.

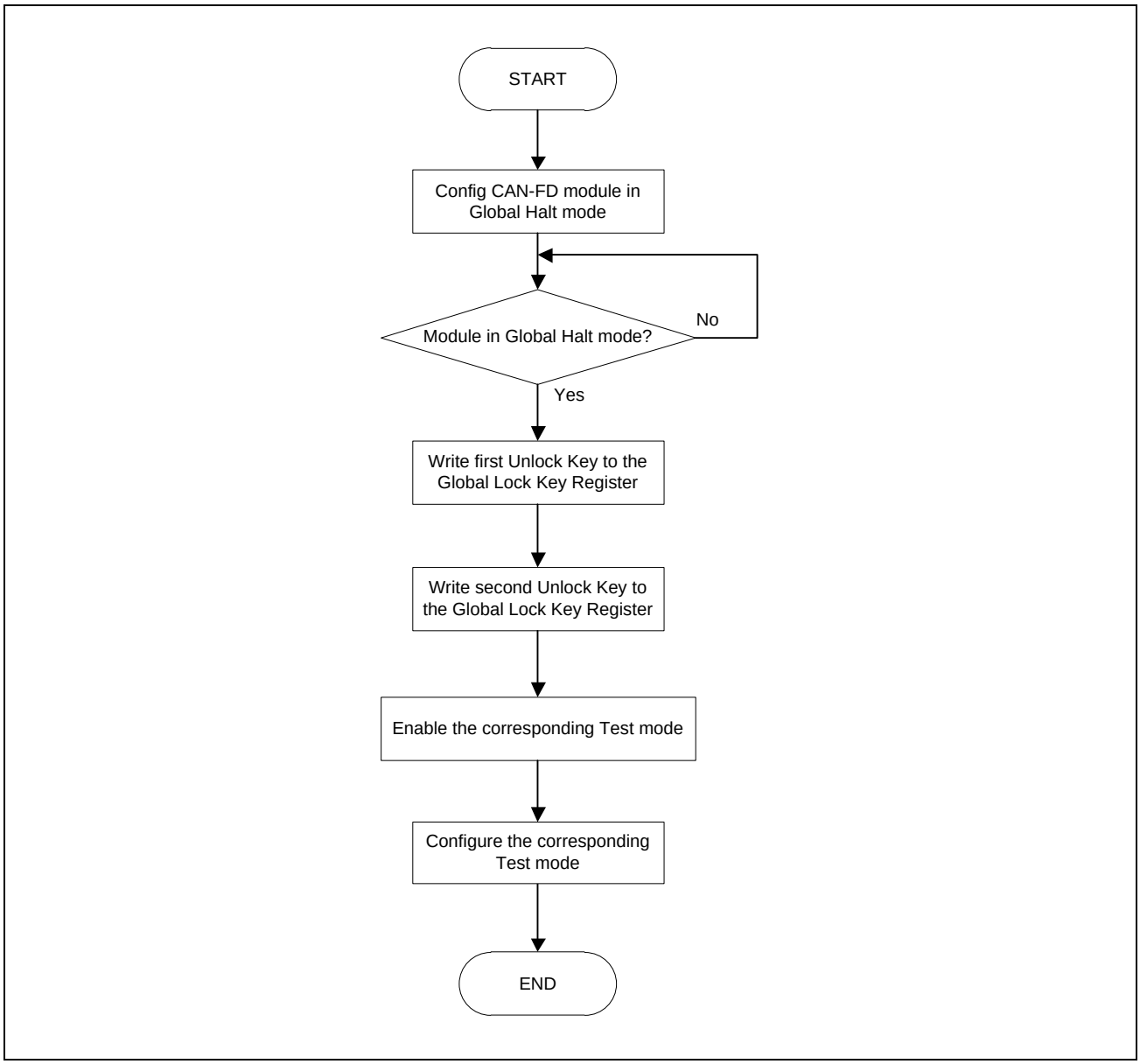


Figure 7.9-54 Unlock Software Protection Routine

7.9.9.2.1 RAM Test Mode

The CAN-FD module can be configured in RAM test mode by setting the CFDGTSTCTR.RTME bit in the Global Test Control Register when the corresponding lock key is previously written. This is a special test mode, in which, the complete RAM area can be accessed.

In this mode, the RAM area is split into number of pages (pn) of 256 bytes each. Which can be accessed with the CFDRPGACCn register.

The page should be selected for read/write access by writing to the CFDGTSTCFG.RTMPS[9:0] bits in the Global Test Control Register. Data can then be read from or written into the RAM Test Page Access Registers.

Figure 7.9-55 shows the structure of the pages in the RAM when performing a RAM test mode.

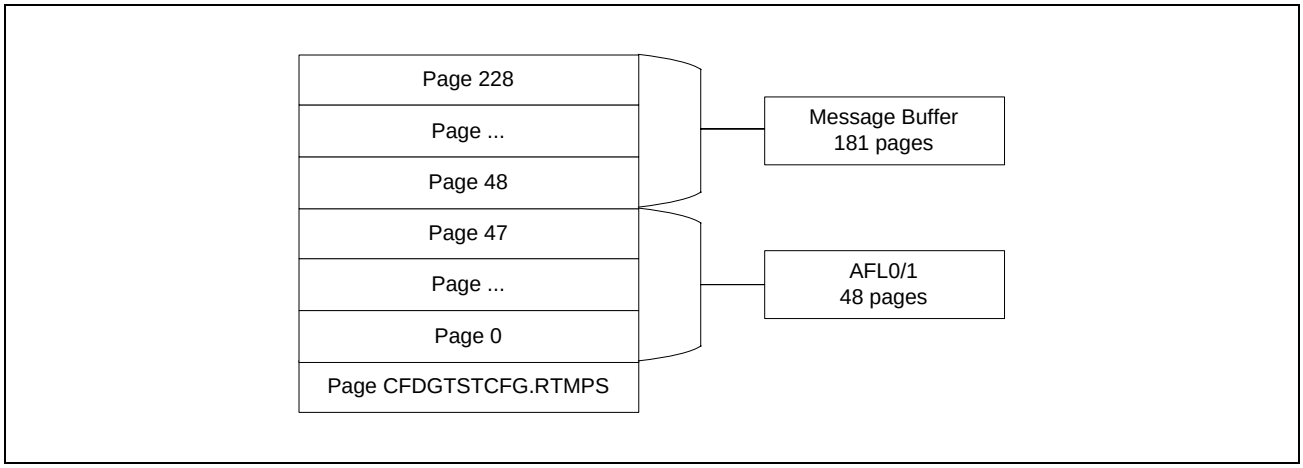


Figure 7.9-55 RAM Page Structure

The total available RAM size for a 6-CAN channel version is 12288 bytes for the AFL RAM and 46272 bytes for the Message Buffer RAM.

AFL RAM0/1 can treat RAM test mode as one RAM.

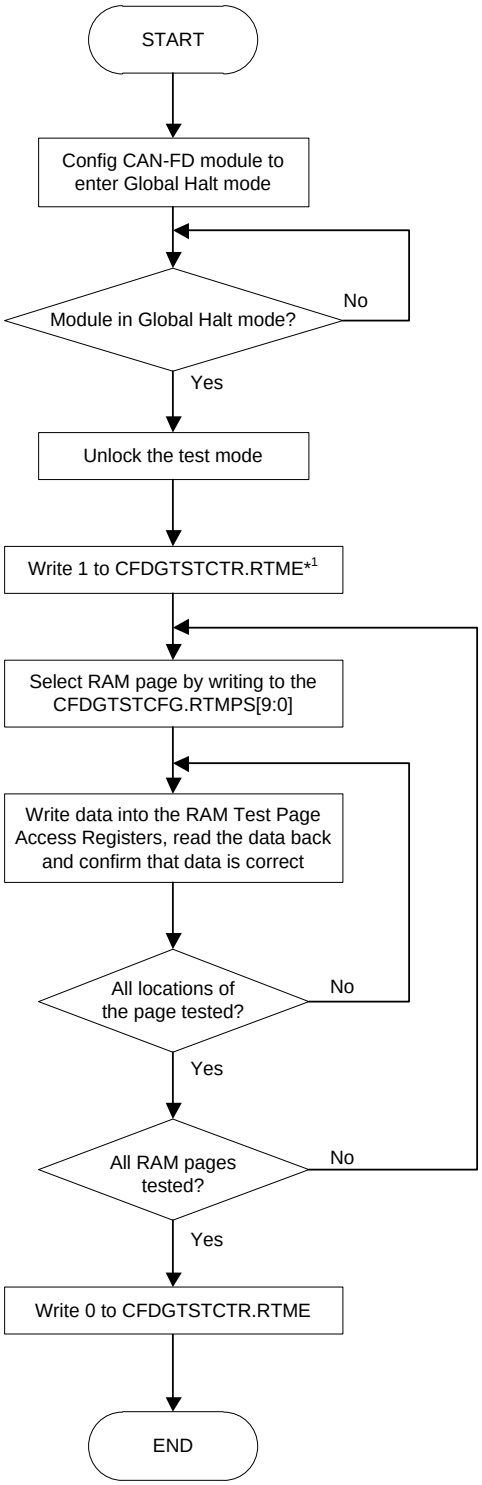
The number of pages (pn) and CFDGTSTCFG.RTMPS[9:0] values for the AFL and MB RAMs are calculated in the following way:

pn = ceil (total RAM size in bytes/number of bytes per page)

- AFL RAM:
  pn = ceil (12288/256) = 48 pages
  CFDGTSTCFG.RTMPS[9:0] = 0 to 47 (02Fh) inclusive
- MB RAM:
  pn = ceil (46272/256) = 181 pages
  CFDGTSTCFG.RTMPS[9:0] = 48 to 228 (0E4h) inclusive

Do not access more than 192 bytes of RAM on the last page (RTMPS[9:0] = 0E4h).

Figure 7.9-56 shows the software flow for RAM test mode.



Note 1. Change into the following status before changing to RAMTEST.
- Cancel of a request of transmission
- Disable of all the FIFO and TXQ
- Clear of the receiving flag of a receiving buffer

Figure 7.9-56 Software Flow of RAM Test Mode

To exit this test mode, clear the CFDGTSTCTR.RTME bit. The CFDGTSTCTR.RTME bit is cleared by writing 0b to it.

The CFDGTSTCTR.RTME bit is cleared automatically when the CAN-FD module enters Global Reset mode from the test mode.

7.9.9.2.2 Internal CAN Bus Communication Mode

The CAN-FD module can be configured in Internal CAN bus communication test mode by setting the CFDGTSTCTR.ICBCTME bit in the Global Test Control Register. This is a special test mode, in which the CAN channels can be connected together internally to generate a CAN cluster within the CAN-FD module.

Only use the following sequence to enter Internal CAN bus communication test mode:

1. Configure all channels in Halt mode and check that all channels have entered Global Halt mode.
2. Write data into the Global Test Configuration Register to select the channels participating in the Internal CAN bus communication test.
3. Set the CFDGTSTCTR.ICBCTME bit of the Global Test Control Register.
4. Check that CFDGTSTCTR.ICBCTME bit is set in the Global Test Control Register.

In this mode, the TXD outputs of the channels participating (configured) in Internal CAN bus communication mode are connected together using AND gate. The output of the AND gate is connected to the RXD inputs of all participating channels to create a CAN cluster within the CAN-FD module. The channels are isolated from the external CAN bus while the CAN-FD module is in this test mode.

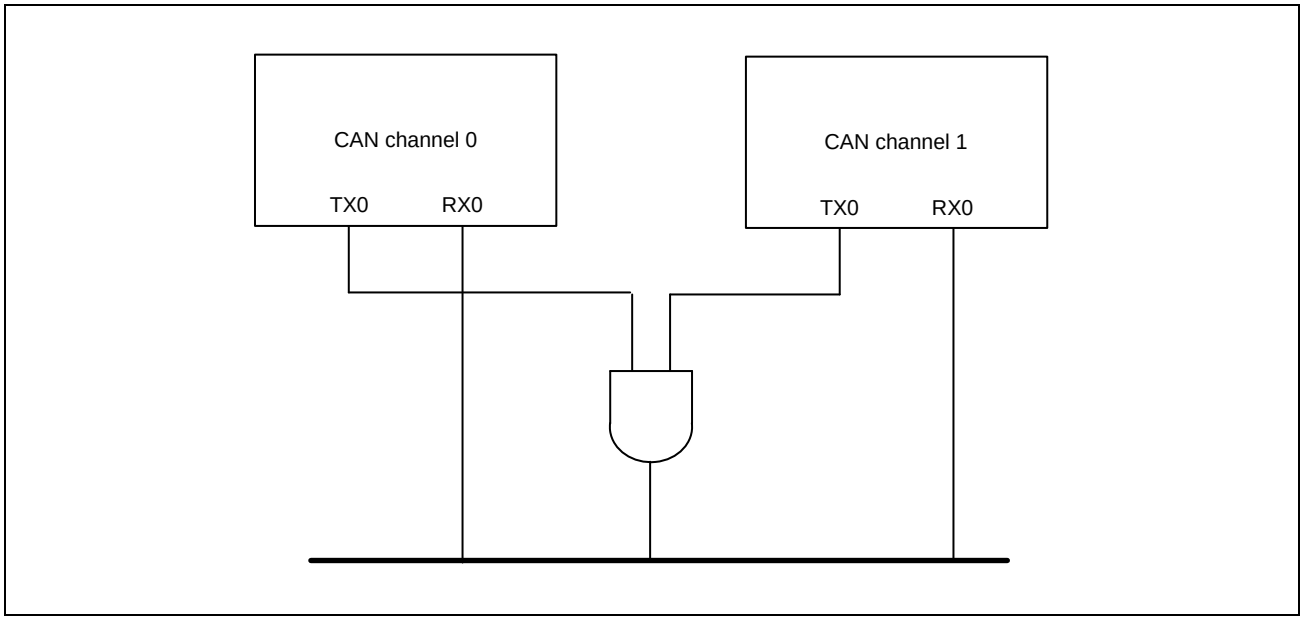


Figure 7.9-57 Internal CAN Bus Connections

The AFL, Flat RX message buffers, FIFO buffers, Flat TX message buffers, and various registers can now be configured as normal to start communication between channels.

The channels not participating in Internal CAN bus should only be configured in Halt mode.

7.9.9.2.3 CRC Error Test

After the CAN-FD module has been configured in Internal CAN bus communication test mode, use the following sequence to perform CRC error testing. In the following sequence, channel x is the reference transmitter CAN-FD module and channel y is the receiver CAN-FD module where (x, y = [0, …, n] and x ≠ y):

1. Configure channel x node to transmit one reference message.
2. Set the CFDCyCTR.CRCT bit to 1b, in order to invert the first bit of the incoming bit stream from channel x.
3. Set the CFDTMCx.TMTR.
4. Read either CFDCyERFL.CRCREG or CFDCyFDCRC.CRCREG (depending on the received frame type: Classical or FD). The value should be different from the received CRC value of the reference message from channel x.
5. Check that CFDCyERFL.CERR is 1b.

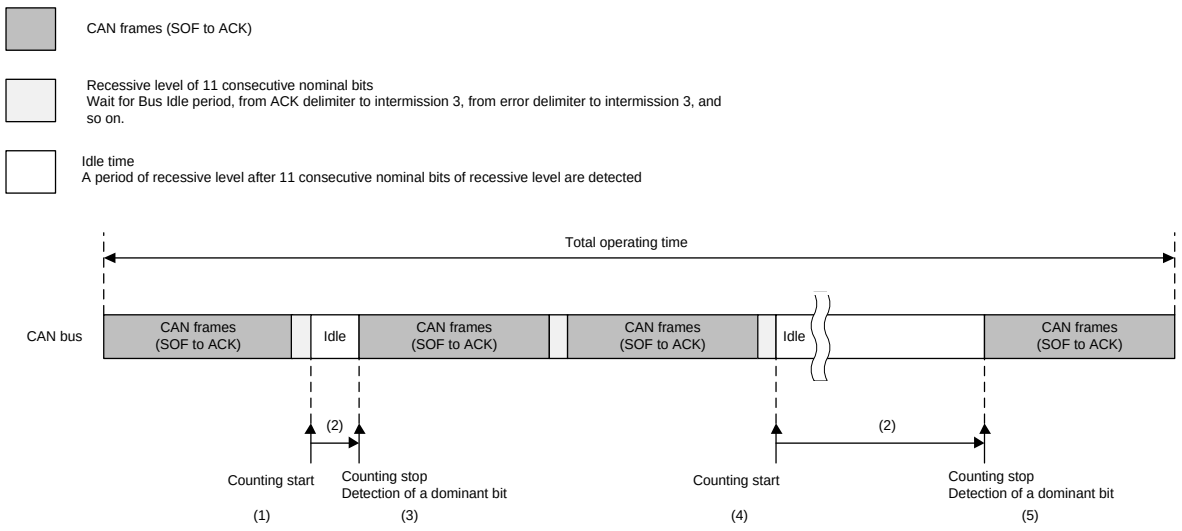
As the CRC generator logic is shared for RX and TX there is no need to create a separate TX CRC error test.

7.9.10 Bus Traffic Measurement

The idle time of the CAN bus can be measured using CANFD_0_pclk or CANFD_0_clkc. Bus traffic can be calculated based on the measurement results.

7.9.10.1 How to Count the CAN Bus Idle Time

Figure 7.9-58 shows the concept of measuring the idle time of the CAN bus.



(1) With the CANFD_0_pclk clock, counting of the idle time starts after a recessive level input of 11 consecutive nominal bits are detected.

(2) The counting continues until a dominant level is detected on the CAN bus.

(3) The counting stops after a dominant level is detected on the CAN bus.

(4) After that, the counting restarts when 11 consecutive nominal bits are detected.

Figure 7.9-58 How to Measure Idle Time of the CAN Bus

7.9.10.2 Operations and Measurement Procedure

The following procedure is for measuring the idle time of the CAN bus.

1. The channel to be measured transits to operation mode.
2. Write 1b to CFDCnBLCT.BLCE bit to set the measuring counter to operating mode.
3. Write 1b to CFDCnBLCT.BLCLD bit to clear the counter register.
4. Detect a recessive level input of 11 consecutive nominal bits.
5. Start counting the bus idle time.
6. Detect a dominant level.
7. The counting stops.
8. Detect a recessive level input of 11 consecutive nominal bits.
9. The counting starts.

10. Write 1b to CFDCnBLCT.BLCLD bit to clear the counter register and simultaneously load the counter value to CFDCnBLSTS.
11. Read the value of CFDCnBLSTS.

To stop the measurement counter, write 0b to CFDCnBLCT.BLCE bit.

To initialize the counter, write 1b to CFDCnBLCT.BLCLD bit.

This measurement is enabled when the channel to be measured is in Operation mode.

When the relevant channels are in reset mode, the counter does not operate.

Also, accurate measurements are not available in test mode.

Write 1b to CFDCnBLCT.BLCLD bit to clear the counter register and simultaneously load the value of the counter to CFDCnBLSTS.

The lower 3 bits of CFDCnBLSTS are fixed to 0b.

Based on the values of the counter, software can calculate the CAN bus traffic according to the following formulas:

$$\frac{(\text{Total operating time} - \text{Total idle time})}{\text{Total operating time}} = \frac{\text{Bus operating time}}{\text{Total operating time}} = \text{Bus usage ratio}$$

- Total idle time: a value read from CFDCnBLSTS × a clock cycle of CANFD_0_pclk
- Total operating time: a setting interval of CFDCnBLCT.BLCLD bit

Example: Below is a calculation example under the following conditions:

- Conditions: nominal bit rate = 1 Mbps
- CANFD_0_clkc = 80 MHz (= 12.5ns)
- A setting interval of CFDCnBLCT.BLCLD bit = cycle of 1 ms
- A read value of CFDCnBLSTS register = 4E20h

$$\frac{(\text{Total operating time} - \text{Total idle time})}{\text{Total operating time}} = \frac{(1000000\ \text{ns} - 20000 \times 12.5\ \text{ns})}{1000000\ \text{ns}} = 75\%$$

7.9.11 Flexible CAN mode

In this mode, it is possible to connect 2 channels of the CAN module to a single CAN driver.

The pairs of channels in Flexible CAN mode are as follows.

- When the CFDGFCMC.FLXC0 bit is set, channels 0 and 1 of the RS-CAN-FD module are in Flexible CAN mode.
- When the CFDGFCMC.FLXC1 bit is set, channels 2 and 3 of the RS-CAN-FD module are in Flexible CAN mode.
- When the CFDGFCMC.FLXC2 bit is set, channels 4 and 5 of the RS-CAN-FD module are in Flexible CAN mode.

Channel n+1 uses the TX/RX terminal of channel n.

The TX/RX terminal of channel n+1 cannot be used.

In Flexible CAN mode, each channel performs communication processing independently.

However, when one of the channels transmits, the other channel will not return an acknowledge bit.

Remark When operating in Flexible CAN mode, the error counters (TEC/REC) of the two CAN nodes are not synchronized with each other.

Flexible transmission buffer assignment configured in the CFDGFTBAC register and Flexible CAN mode configured in the CFDGFCMC register should not be used simultaneously.

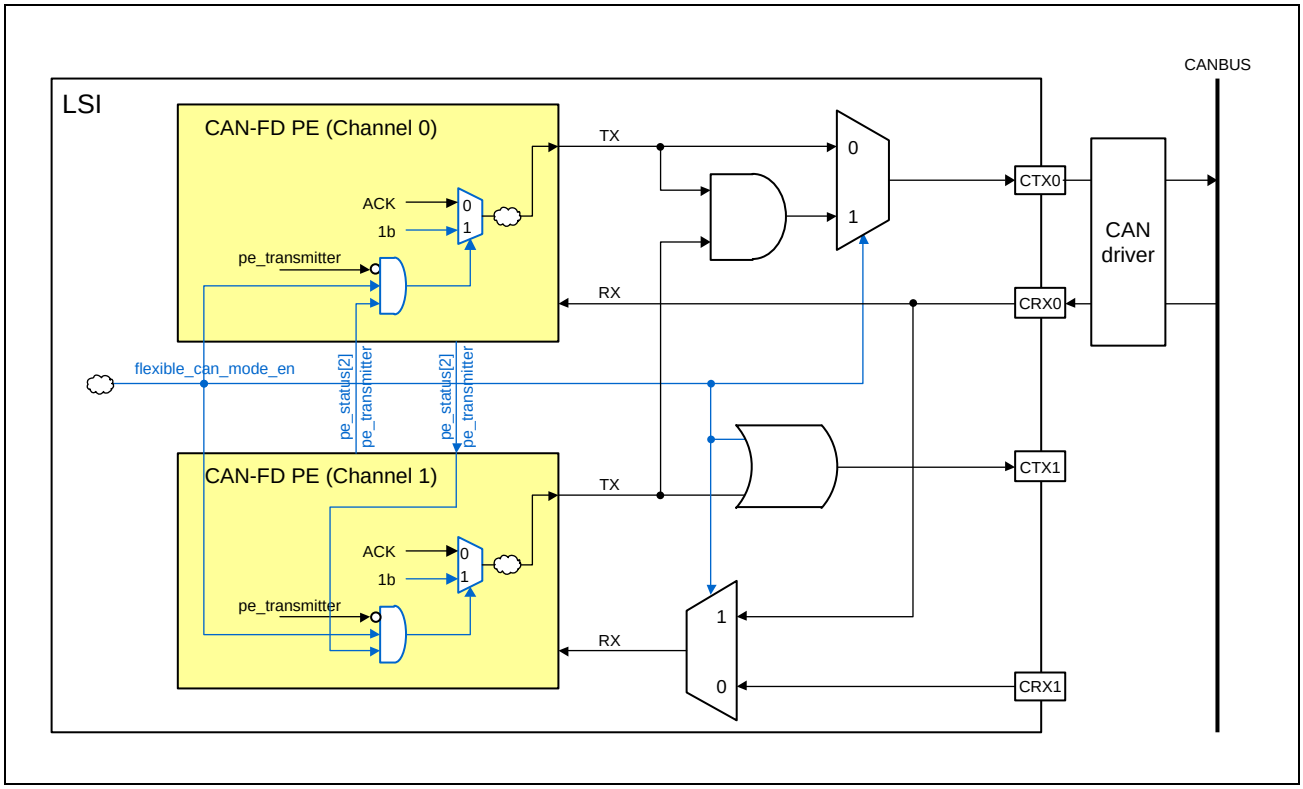


Figure 7.9-59 Diagram of the Flexible CAN for Channel 0 and Channel 1

7.9.12 Flexible Transmission Buffer Assignment

Each channel has 32 transmit buffers for exclusive use.

This channel can borrow up to 16 TXMBs from the next channel in case it runs out of TXMBs.

Buffers that can be rented range from 4 to 16 buffers in increments of 4 units.

The pairs of channels in Flexible CAN mode are as follows.

- When the CFDGFTBAC.FLXMB0 bit is set, flexible transmission buffer assignment between channel 0 and channel 1.
- When the CFDGFTBAC.FLXMB1 bit is set, flexible transmission buffer assignment between channel 2 and channel 3.
- When the CFDGFTBAC.FLXMB2 bit is set, flexible transmission buffer assignment between channel 4 and channel 5.

Flexible transmission buffer assignment configured in the CFDGFTBAC register and Flexible CAN mode configured in the CFDGFCMC register should not be used simultaneously.

The interrupt of the rented buffer is output to the interrupt of the rented channel.

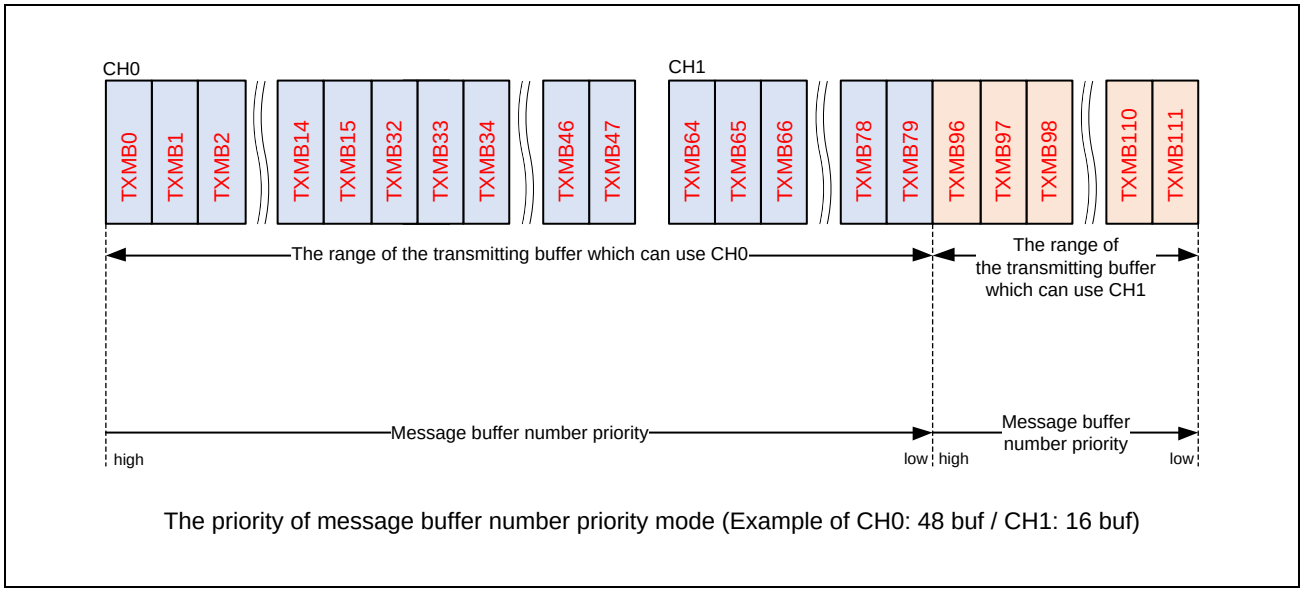
When using TXQ through the rented buffer, TXQ should only be set within the rented range.

The rented TXMB operates in the mode of the channel to be used.

For example, if channel 1 is in reset mode, channel 0 transmission is possible using a rented TXMB.

Also, the rented TXMB is affected by the transmission status of the rented channel, TX SCAN PROCESS, or aborted transmission.

In message buffer number priority mode, TXMB0 has a higher priority and TXMB79 (when 16 buffers are borrowed) has a lower priority. The priority of the rented TXMB is lowered.



The priority of message buffer number priority mode (Example of CH0: 48 buf / CH1: 16 buf)

Figure 7.9-60 Example of Message Buffer Number Priority Mode

The example of a rental channel 0 and channel 1 is shown below.

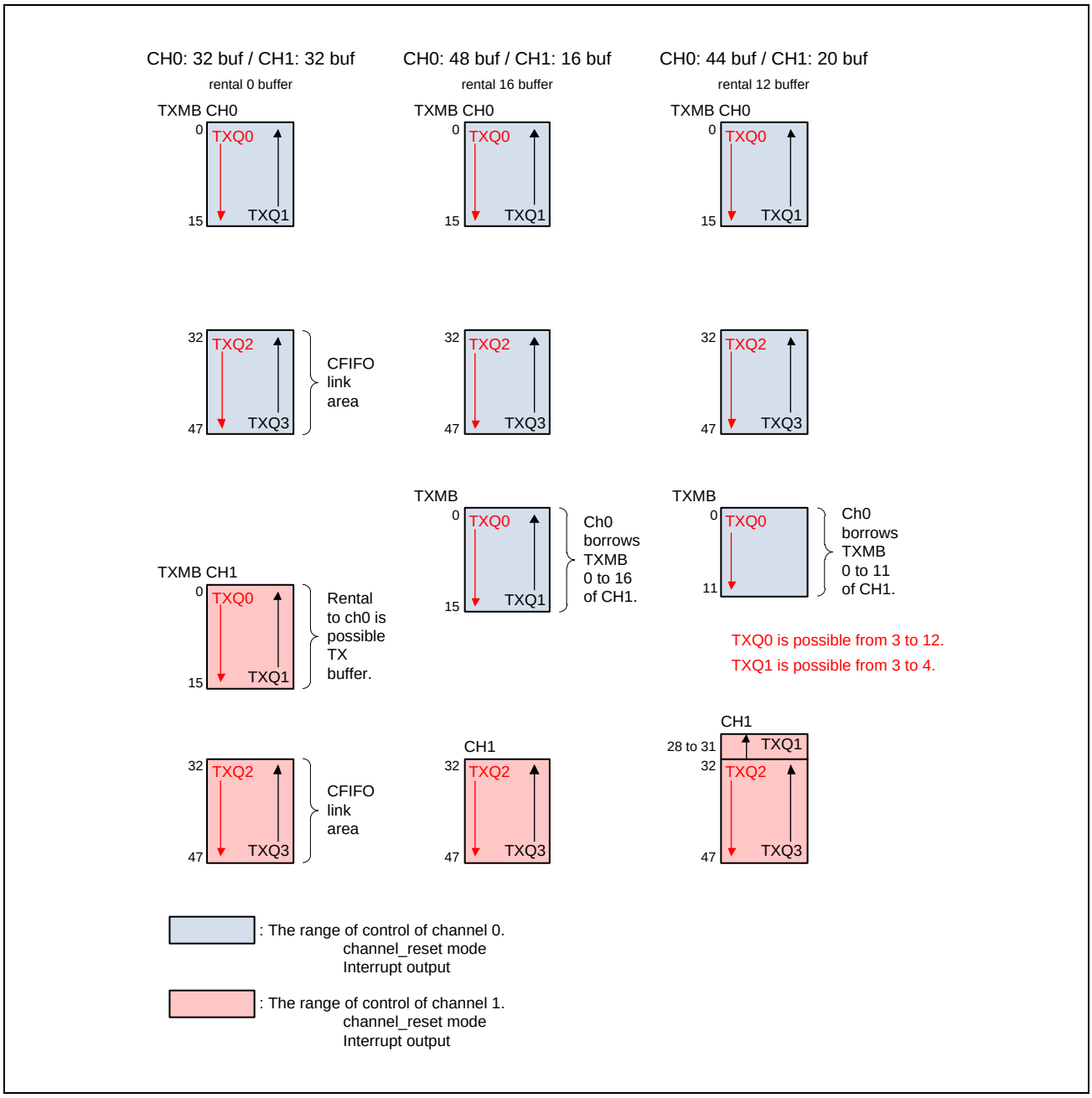


Figure 7.9-61 Example of Message Buffer Number Priority Mode (Channels 0 and 1)

SECTION 7 LOW-SPEED INTERFACE

7.10 12-Bit A/D Converter (ADC)

7.10.1 Overview

This LSI incorporates a 12-bit successive approximation A/D converter. Analog inputs of up to 8 channels are selectable.

The 12-bit A/D converter converts analog inputs of up to 8 selected channels to a 12-bit digital value through successive approximation.

The 12-bit A/D converter has three operating modes: single scan mode in which analog inputs selected arbitrarily are converted only once in ascending order of channel number; continuous scan mode in which analog inputs selected arbitrarily are continuously converted in ascending order of channel number; and group scan mode in which analog inputs are arbitrarily selected and divided into two groups (group A and group B) or three groups (group A, group B and group C), and then analog inputs of channels selected for each group are converted in ascending order of channel number.

In group scan mode, the conditions for scanning start of each groups (A, B or A, B, C) (internal trigger) can be independently selected, thus allowing A/D conversion of each groups to be started independently. In addition to the above actions, the group priority operation in group scan mode accepts the start of scanning the priority group during the scan of the low priority group, interrupts the scan of the low priority group, and starts the scan of the priority group. The priority of group priority operations is group A > group B > group C. In group priority operation, if the scan start of group B is accepted during the scan of group C, the scan of group C is interrupted and the scan of group B is started. If the scan start of group A is accepted during the scan of group C, the scan of group A is started. The scan of Group C is interrupted and the scan of Group A is started. Similarly, if the scan start of Group A is accepted during the scan of Group B, the scan of Group B is interrupted and the scan of Group A is started. The interrupted group scan is able to resume after the priority group scan is complete also.

In double trigger mode, one analog input channel arbitrarily selected is converted in single scan mode or group scan mode (group A), and the resulting data of A/D conversion started by the first and second internal triggers are stored into different registers (duplication of A/D conversion data).

Safety features such as data register overwrite check and automatic clearing of data registers are available.

The compare function is a function that outputs an interrupt (ada_compai_n / ada_compbi_n) when the A/D conversion value of the selected channel by specifying the upper and lower reference values for each window A/B matches the comparison conditions. The event (ada_elccondmtch / ada_elccondunmtch) is output according to the event conditions (A or B, A and B, A exor B). In addition, a comparator operation that compares the A/D conversion value with the lower reference value is also possible.

Table 7.10-1 lists the specifications of the 12-bit A/D converter and **Table 7.10-2** indicates the functions of the 12-bit A/D converter. **Figure 7.10-1** shows a block diagram of the 12-bit A/D converter.

Table 7.10-1 ADC Specifications (1/2)

Item	Description
Number of units	1 unit
Input channels	Up to 8 channels
A/D conversion method	Successive approximation method

Table 7.10-1 ADC Specifications (2/2)

Item	Description
Resolution	Selectable from 12 bits or 8 bits
Sampling rate	• 2.5 Msps (when A/D conversion clock ADC_0_ADCLK is 50 MHz) • 2.0 Msps (when A/D conversion clock ADC_0_ADCLK is 40 MHz) • 1.0 Msps (when A/D conversion clock ADC_0_ADCLK is 20 MHz) • 0.5 Msps (when A/D conversion clock ADC_0_ADCLK is 10 MHz) • 0.25 Msps (when A/D conversion clock ADC_0_ADCLK is 5 MHz)
A/D conversion clock	ADC_0_ADCLK (50 MHz / 40 MHz / 20 MHz / 10 MHz / 5 MHz)
Operating modes	• Single scan mode: A/D conversion is performed only once on the analog inputs of arbitrarily selected channels • Continuous scan mode: A/D conversion is performed repeatedly on the analog inputs of arbitrarily selected channels • Group scan mode: Two (groups A and B) or three (groups A, B, and C) can be selected as the number of the groups to be used. Only the combination of groups A and B can be selected when the number of the groups is two. Analog inputs that are arbitrarily selected are divided into two groups (group A and B) or three groups (group A, B, and C), and A/D conversion of the analog input selected on a group basis is performed only once. The conditions for scanning start of groups A, B, and C (internal trigger) can be independently selected, thus allowing A/D conversion of each group to be started independently. • Group scan mode (when Group A is given priority): If a priority-group trigger is input during scanning of the low-priority group, scan of the low-priority group is stopped and scan of the priority group is started. The priority order is group A (highest) > group B > group C (lowest). Whether or not to restart scanning of the low-priority group after processing for the high- priority group completes, is selectable. Rescan can also be set to start either from the beginning of the selected channel or the channel on which A/D conversion is not completed.
Conditions for A/D conversion start	• Software trigger • Internal trigger Trigger by the multi-function timer pulse unit (GPT) and event link controller (ELC)*1 • External trigger A/D conversion can be triggered by the external trigger ADTRG.
Functions	• Selectable A/D-converted value addition mode or average mode • Double-trigger mode (duplication of A/D conversion data) • Automatic clear function for A/D data registers • Digital comparison of values in the comparison and data registers, and between values in the data registers • Overwrite checking function of the A/D data register
Interrupt sources and ELC events	• ada_adireq_n: A/D scan end interrupt , A/D scan end interrupt for Group A) • ada_gbadireq_n: A/D scan end interrupt for Group B • ada_gcadireq_n: A/D scan end interrupt for Group C • ada_compai_n: Window A compare match • ada_compbi_n: Window B compare match • ada_adelcreq: A/D scan end interrupt for ELC • ada_elccondmtch: compare match • ada_elccondunmtch: compare unmatch • ada_adereq_n: A/D error interrupt request by overwrite check
ELC interface	• An ELC event is generated upon completion of all scans • Able to start scanning by a trigger from the ELC
Power-Down mode	The A/D converter can be set to power-down mode.

Note 1. The ELC is included in the Interrupt Control Unit (ICU).

Table 7.10-2 Function of ADC (1/2)

Parameter			Unit (ADC)
Analog input channel			ANI000 to ANI007
Conditions for A/D conversion start	Software	Software trigger	Enabled
	External trigger	Trigger input pin (Use only internal trigger as the scan start in group scan mode)	ADTRG
	Internal trigger	Compare match with GPT00.GTADTRA	GTADTRA0N
		Compare match with GPT00.GTADTRB	GTADTRB0N
		Compare match with GPT01.GTADTRA	GTADTRA1N
		Compare match with GPT01.GTADTRB	GTADTRB1N
		Compare match with GPT02.GTADTRA	GTADTRA2N
		Compare match with GPT02.GTADTRB	GTADTRB2N
		Compare match with GPT03.GTADTRA	GTADTRA3N
		Compare match with GPT03.GTADTRB	GTADTRB3N
		Compare match with GPT04.GTADTRA	GTADTRA4N
		Compare match with GPT04.GTADTRB	GTADTRB4N
		Compare match with GPT05.GTADTRA	GTADTRA5N
		Compare match with GPT05.GTADTRB	GTADTRB5N
		Compare match with GPT06.GTADTRA	GTADTRA6N
		Compare match with GPT06.GTADTRB	GTADTRB6N
		Compare match with GPT07.GTADTRA	GTADTRA7N
		Compare match with GPT07.GTADTRB	GTADTRB7N
		Compare match with GPT10.GTADTRA	GTADTRA8N
		Compare match with GPT10.GTADTRB	GTADTRB8N
		Compare match with GPT11.GTADTRA	GTADTRA9N
		Compare match with GPT11.GTADTRB	GTADTRB9N
		Compare match with GPT12.GTADTRA	GTADTRA10N
		Compare match with GPT12.GTADTRB	GTADTRB10N
		Compare match with GPT13.GTADTRA	GTADTRA11N
		Compare match with GPT13.GTADTRB	GTADTRB11N
		Compare match with GPT14.GTADTRA	GTADTRA12N
		Compare match with GPT14.GTADTRB	GTADTRB12N
		Compare match with GPT15.GTADTRA	GTADTRA13N
		Compare match with GPT15.GTADTRB	GTADTRB13N
		Compare match with GPT16.GTADTRA	GTADTRA14N
		Compare match with GPT16.GTADTRB	GTADTRB14N
		Compare match with GPT17.GTADTRA	GTADTRA15N
		Compare match with GPT17.GTADTRB	GTADTRB15N
		Compare match with GPT00.GTADTRA or Compare match with GPT00.GTADTRB	GTADTRA0N or GTADTRB0N
		Compare match with GPT01.GTADTRA or Compare match with GPT01.GTADTRB	GTADTRA1N or GTADTRB1N
		Compare match with GPT02.GTADTRA or Compare match with GPT02.GTADTRB	GTADTRA2N or GTADTRB2N
		Compare match with GPT03.GTADTRA or Compare match with GPT03.GTADTRB	GTADTRA3N or GTADTRB3N

Table 7.10-2 Function of ADC (2/2)

Parameter			Unit (ADC)
Conditions for A/D conversion start	Internal trigger	Compare match with GPT04.GTADTRA or Compare match with GPT04.GTADTRB	GTADTRA4N or GTADTRB4N
		Compare match with GPT05.GTADTRA or Compare match with GPT05.GTADTRB	GTADTRA5N or GTADTRB5N
		Compare match with GPT06.GTADTRA or Compare match with GPT06.GTADTRB	GTADTRA6N or GTADTRB6N
		Compare match with GPT07.GTADTRA or Compare match with GPT07.GTADTRB	GTADTRA7N or GTADTRB7N
		Compare match with GPT10.GTADTRA or Compare match with GPT10.GTADTRB	GTADTRA8N or GTADTRB8N
		Compare match with GPT11.GTADTRA or Compare match with GPT11.GTADTRB	GTADTRA9N or GTADTRB9N
		Compare match with GPT12.GTADTRA or Compare match with GPT12.GTADTRB	GTADTRA10N or GTADTRB10N
		Compare match with GPT13.GTADTRA or Compare match with GPT13.GTADTRB	GTADTRA11N or GTADTRB11N
		Compare match with GPT14.GTADTRA or Compare match with GPT14.GTADTRB	GTADTRA12N or GTADTRB12N
		Compare match with GPT15.GTADTRA or Compare match with GPT15.GTADTRB	GTADTRA13N or GTADTRB13N
		Compare match with GPT16.GTADTRA or Compare match with GPT16.GTADTRB	GTADTRA14N or GTADTRB14N
		Compare match with GPT17.GTADTRA or Compare match with GPT17.GTADTRB	GTADTRA15N or GTADTRB15N
		ELC trigger	ELCTRG0
Interrupt			ada_adireq_n ada_gbadireq_n ada_gcadireq_n ada_compai_n ada_compbi_n
Output to ELC			ada_adelcreq ada_elccondmtch ada_elccondunmtch
Power-Down mode settings			Available

Table 7.10-3 ADC Interrupt Sources

Name	Interrupt Sources	CPU request	DMAC Activation	ELC Activation	EMC request
ada_adireq_n	A/D scan end interrupt, A/D scan end interrupt for Group A	Possible	Possible	Possible	Not possible
ada_gbadireq_n	A/D scan end interrupt for Group B	Possible	Possible	Possible	Not possible
ada_gcadireq_n	A/D scan end interrupt for Group C	Possible	Possible	Possible	Not possible
ada_compai_n	Window A compare match	Possible	Not possible	Not possible	Not possible
ada_compbi_n	Window B compare match	Possible	Not possible	Not possible	Not possible
ada_adelcreq	A/D scan end interrupt for ELC	Not possible	Not possible	Possible	Not possible
ada_elccondmtch	Compare match	Not possible	Possible	Possible	Not possible
ada_elccondunmtch	Compare mismatch	Not possible	Possible	Possible	Not possible
ada_adereq_n	A/D error interrupt request by overwrite check	Not possible	Not possible	Not possible	Possible

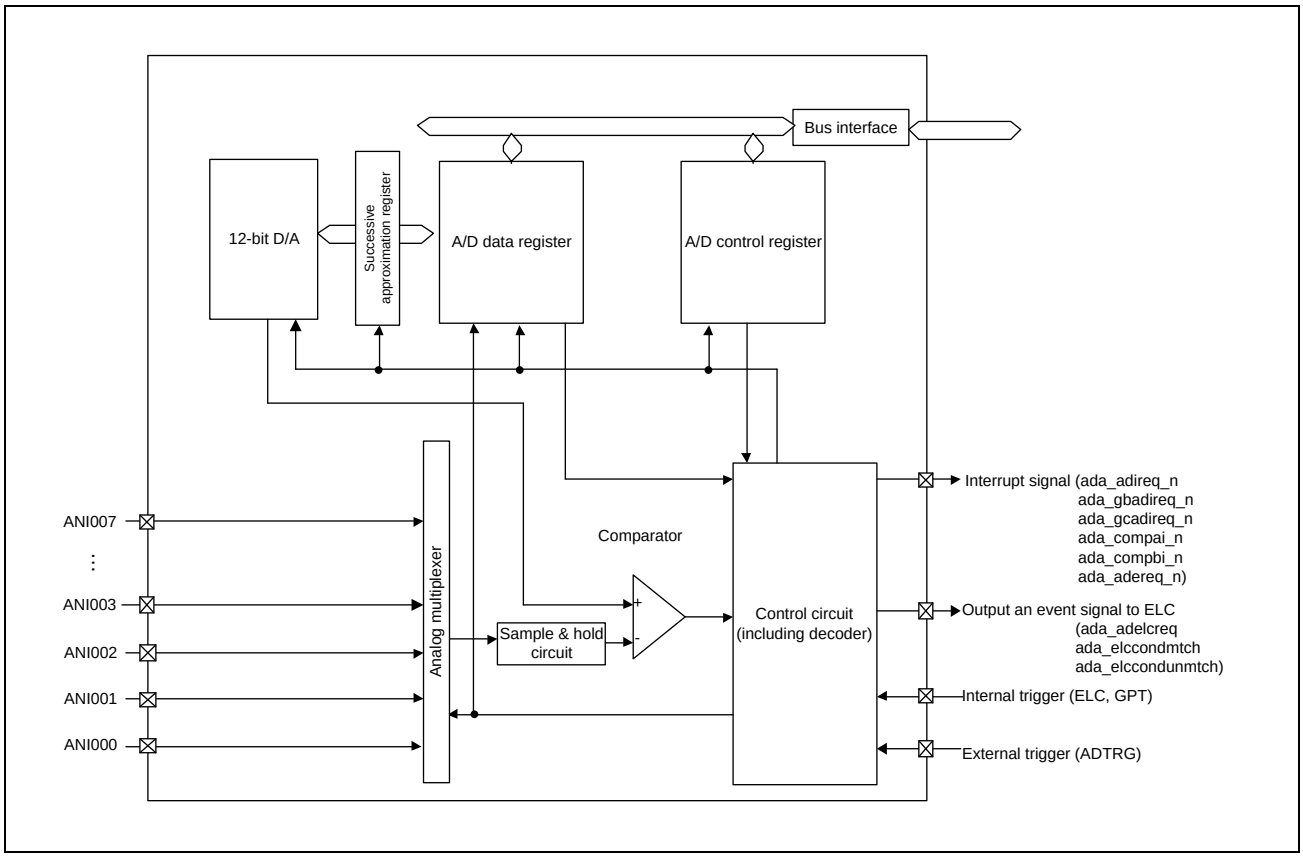


Figure 7.10-1 Block Diagram of 12-Bit A/D Converter (Unit0)

7.10.2 ADC Registers

The base address for the ADC is as follows.

Table 7.10-4 Register Base Address

Base Register Name	ADC ch	Base Address
<ADCm_base> (m = 0)	ADC0	0_11C0_0000h (51C0_0000h*1, 41C0_0000h*2)

Note 1. CM33 Address Space Non-Secure

Note 2. CM33 Address Space Secure

7.10.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
A/D Control Register	ADCm_ADCSR	0000h	0000h	16
A/D Conversion Status Register	ADCm_ADREF	00h	0002h	8
Reserve	-	-	0003h	
A/D Channel Select Register A0	ADCm_ADANSA0	0000h	0004h	16
Reserve	-	-	0006h to 0007h	-
A/D-Converted Value Addition/Average Function Channel Select Register 0	ADCm_ADADS0	0000h	0008h	16
Reserve	-	-	000Ah to 000Bh	-
A/D-Converted Value Addition/Average Count Select Register	ADCm_ADADC	00h	000Ch	8
Reserve	-	-	000Dh	-
A/D Control Extended Register	ADCm_ADCER	0000h	000Eh	16
A/D Conversion Start Trigger Select Register	ADCm_ADSTRGR	0000h	0010h	16
Reserve	-	-	0012h to 0013h	-
A/D Channel Select Register B0	ADCm_ADANSB0	0000h	0014h	16
Reserve	-	-	0016h to 0017h	-
A/D Data Duplication Register	ADCm_ADDBLDR	0000h	0018h	16
A/D Data Register 0	ADCm_ADDR0	0000h	0020h	16
A/D Data Register 1	ADCm_ADDR1	0000h	0022h	16
A/D Data Register 2	ADCm_ADDR2	0000h	0024h	16
A/D Data Register 3	ADCm_ADDR3	0000h	0026h	16
A/D Data Register 4	ADCm_ADDR4	0000h	0028h	16
A/D Data Register 5	ADCm_ADDR5	0000h	002Ah	16
A/D Data Register 6	ADCm_ADDR6	0000h	002Ch	16
A/D Data Register 7	ADCm_ADDR7	0000h	002Eh	16
Reserve	-	-	0030h to 007Ch	-
A/D Event Link Control Register	ADCm_ADELCCR	00h	007Dh	8
Reserve	-	-	007Eh to 007Fh	-
A/D Group Scan Priority Control Register	ADCm_ADGSPCR	0000h	0080h	16

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	0082h to 0083h	-
A/D Data Duplication Register A	ADCm_ADDBLDRA	0000h	0084h	16
A/D Data Duplication Register B	ADCm_ADDBLDRB	0000h	0086h	16
Reserve	-	-	0088h to 008Bh	-
A/D Compare Function Window A/B Status Monitoring Register	ADCm_ADWINMON	00h	008Ch	8
Reserve	-	-	008Dh to 008Fh	-
A/D Compare Function Control Register	ADCm_ADCMPCR	0000h	0090h	16
Reserve	-	-	0092h to 0093h	-
A/D Compare Function Window A Channel Select Register 0	ADCm_ADCMPANSR0	0000h	0094h	16
Reserve	-	-	0096h to 0097h	-
A/D Compare Function Window A Comparison Condition Setting Register 0	ADCm_ADCMPLR0	0000h	0098h	16
Reserve	-	-	009Ah to 009Bh	-
A/D Comparison Function Window A Lower Level Setting Register	ADCm_ADCMPDR0	0000h	009Ch	16
A/D Comparison Function Window A Upper Level Setting Register	ADCm_ADCMPDR1	0000h	009Eh	16
A/D Comparison Function Window A Channel Status Register 0	ADCm_ADCMPSR0	0000h	00A0h	16
Reserve	-	-	00A2h to 00A5h	-
A/D Compare Function Window B Channel Select Register	ADCm_ADCMPBNSR	00h	00A6h	8
Reserve	-	-	00A7h	-
A/D Compare Function Window B Lower-Side Level Setting Register	ADCm_ADWINLLB	0000h	00A8h	16
A/D Compare Function Window B Upper-Side Level Setting Register	ADCm_ADWINULB	0000h	00AAh	16
A/D Compare Function Window B Status Register	ADCm_ADCMPBSR	00h	00ACh	8
Reserve	-	-	00ADh to 00D3h	-
A/D Channel Select Register C0	ADCm_ADANSC0	0000h	00D4h	16
Reserve	-	-	00D6h to 00D8h	-
A/D Group C Trigger Select Register	ADCm_ADGCTRGR	00h	00D9h	8
Reserve	-	-	00DAh to 01E1h	-
A/D Error Control Register	ADCm_ADERCR	00h	1E2h	8
A/D Error Clear Register	ADCm_ADERCLR	00h	1E3h	8
Reserve	-	-	01E4h to 01E7h	-
A/D Overwrite Error Register	ADCm_ADOWER	0000h	1E8h	16
A/D Overwrite Error Extended Register	ADCm_ADOWEER	0000h	1EAh	16

7.10.2.2 ADC Register Descriptions

The prefix (ADCm_) of the register names is omitted in this and subsequent sections.

7.10.2.2.1 A/D Data Registers

The A/D-converted value has a 12-bit/8-bit accuracy. The formats for data in the ADDRy, ADDBLDR, ADDBLDRA, and ADDBLDRB registers vary according to the following conditions.

- The setting of the A/D data register format select bit (ADCER.ADRFMT) (determining whether the data are flush-left or flush-right in the registers)
- The setting of the A/D conversion accuracy specify bits (ADCER.ADPRC[1:0]) (12-bit or 8-bit accuracy is selectable.)
- The setting of the Addition count select bits (ADADC.ADC[2:0]) (once, twice, three times, four times, or sixteen times is selectable.)
- The setting of the average mode enable bit (ADADC.AVEE) (Addition or average is selectable.)

When the number of the conversion count is set from one to four in the addition mode, the A/D conversion addition result is expanded by 2 bits and stored in the A/D data register. When the number of the conversion count is sixteen in the addition mode, the A/D conversion addition result is expanded by 4 bits and stored in the A/D data register.

(1) A/D Data Register n (ADCm_ADDRn) (n = 0 to 7)

Access Size : 16 bits
Address : <ADCm_base> + 0020h + n x 0002h
Initial Value : 0000h

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	DR[15:0]	0h	R	The result of A/D conversion (n: Number of channel)

(2) A/D Data Duplication Register (ADCm_ADDBLDR)

Access Size : 16 bits
Address : <ADCm_base> + 0018h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DBLDR[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	DBLDR[15:0]	0h	R	The result of A/D conversion in response to the second trigger in double trigger mode.

(3) A/D Data Duplication Register A (ADCm_ADDBLDRA)

Access Size : 16 bits
Address : <ADCm_base> + 0084h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DBLDRA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	DBLDRA[15:0]	0h	R	The result of A/D conversion during extended operation in double trigger mode

(4) A/D Data Duplication Register B (ADCm_ADDBLDRB)

Access Size : 16 bits
Address : <ADCm_base> + 0086h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DBLDRB[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	DBLDRB[15:0]	0h	R	The result of A/D conversion during extended operation in double trigger mode

7.10.2.2.2 A/D Control Register (ADCm_ADCSR)

Access Size : 16 bits

Address : <ADCm_base> + 0000h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADST	ADCS[1:0]		ADIE	-	-	TRGE	EXTRG	DBLE	GBADIE	-	DBLANS[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	RW	RW	RW	RW	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15	ADST	0h	RW	A/D conversion Start 0b: Stops A/D conversion process 1b: Starts A/D conversion process
14, 13	ADCS[1:0]	0h	RW	Scan Mode Select 00b: Single scan mode 01b: Group scan mode 10b: Continuous scan mode 11b: Setting prohibited
12	ADIE	0h	RW	Scan End Interrupt Enable 0b: Disables ada_adireq_n interrupt generation upon scan completion 1b: Enables ada_adireq_n interrupt generation upon scan completion
11, 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	TRGE	0h	RW	Trigger Start Enable 0b: Disables A/D conversion to be started by the internal or external trigger 1b: Enables A/D conversion to be started by the internal or external trigger
8	EXTRG	0h	RW	Trigger Select 0b: A/D conversion is started by the internal trigger (GPT, ELC) 1b: A/D conversion is started by the external trigger (ADTRG)
7	DBLE	0h	RW	Double Trigger Mode Select 0b: Deselects double trigger mode 1b: Selects double trigger mode
6	GBADIE	0h	RW	Group B Scan End Interrupt Enable 0b: Disables ada_gbadireq_n interrupt generation upon group B scan completion 1b: Enables ada_gbadireq_n interrupt generation upon group B scan completion
5	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
4 to 0	DBLANS[4:0]	0h	RW	Double Trigger Channel Select These bits select one analog input channel for double triggered operation. The setting is only valid while double trigger mode is selected.

7.10.2.2.3 A/D Conversion Status Register (ADCm_ADREF)

The ADREF register indicates the status of the A/D conversion.

Access Size : 8 bits
Address : <ADCm_base> + 0002h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	ADSCACT	-	-	-	-	-	-	ADF
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	ADSCACT	0h	R	0b: A/D conversion is not in progress. 1b: A/D conversion is in progress.
6 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read.
0	ADF	0h	R	Reserved The read value is undefined.

7.10.2.2.4 A/D Channel Select Register (ADCm_ADANSA0)

Access Size : 16 bits
Address : <ADCm_base> + 0004h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ANSA0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 0	ANSA0[7:0]	0h	RW	A/D conversion Analog input Channel Select ANSA0[0] bit corresponds to ch0 and ANSA0[7] corresponds to ch7. 0b: ch0-ch7 are not subjected to conversion. 1b: ch0-ch7 are subjected to conversion.

7.10.2.2.5 A/D-Converted Value Addition/Average Function Channel Select Register 0 (ADCm_ADADS0)

Access Size : 16 bits

Address : <ADCm_base> + 0008h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ADS0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 0	ADS0[7:0]	0h	RW	A/D-Converted Value Addition/Average Channel Select ADS0[0] bit corresponds to ch0 and ADS0[7] corresponds to ch7. 0b: A/D-converted value addition/average mode for ch0 to ch7 is not selected 1b: A/D-converted value addition/average mode for ch0 to ch7 is selected

7.10.2.2.6 A/D-Converted Value Addition/Average Count Select Register (ADCm_ADADC)

Access Size : 8 bits

Address : <ADCm_base> + 000Ch

Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	AVEE	-	-	-	-	ADC[2:0]		
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	AVEE	0h	RW	Average Mode Enable 0b: Addition mode is selected 1b: Average mode is selected
6 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2 to 0	ADC[2:0]	0h	RW	Addition Count Select 000b: 1-time conversion (no addition; same as normal conversion) 001b: 2-time conversion (addition once) 010b: 3-time conversion (addition twice)*1 011b: 4-time conversion (addition three times) 101b: 16-time conversion (addition 15 times)*2 Others: Setting prohibited

Note 1. When average mode is selected by setting the ADADC.AVEE bit to 1, do not set the Addition count to 3 times (ADADC.ADC[2:0] = 010b).

Note 2. For the 12-bit conversion accuracy, the Addition count can only be set to select 16 times (ADADC. ADC[2:0] = 101b).

7.10.2.2.7 A/D Control Extended Register (ADCm_ADCER)

Access Size : 16 bits

Address : <ADCm_base> + 000Eh

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADRFMT	-	-	-	-	-	-	-	-	-	ACE	-	-	ADPRC[1:0]		-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	RW	R	R	RW	RW	R

Bit	Bit Name	Initial Value	R/W	Description
15	ADRFMT	0h	RW	A/D Data Register Format Select 0b: Right-alignment is selected for the A/D data register format 1b: Left-alignment is selected for the A/D data register format
14 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5	ACE	0h	RW	A/D Data Register Automatic Clearing Enable 0b: Disables automatic clearing 1b: Enables automatic clearing
4, 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2, 1	ADPRC[1:0]	0h	RW	A/D Conversion Accuracy Specify 00b: A/D conversion is performed with 12-bit accuracy 10b: A/D conversion is performed with 8-bit accuracy Others: Setting prohibited
0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

7.10.2.2.8 A/D Conversion Start Trigger Select Register (ADCm_ADSTRGR)

Access Size : 16 bits
Address : <ADCm_base> + 0010h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	TRSA[5:0]						-	-	TRSB[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
13 to 8	TRSA[5:0]	0h	RW	A/D Conversion Start Trigger Select Select the A/D conversion start trigger in single scan mode and continuous mode. In group scan mode, the A/D conversion start trigger for group A is selected. See **Table 7.10-6** for details.
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5 to 0	TRSB[5:0]	0h	RW	A/D Conversion Start Trigger Select for Group B Select the A/D conversion start trigger for group B in group scan mode. See **Table 7.10-5** for details.

Table 7.10-5 Selection of A/D Activation Sources by the TRSB[5:0] Bits (for Group B Only) (1/3)

Module	Source	Remarks	TRSB					
			[5]	[4]	[3]	[2]	[1]	[0]
Trigger source de-selection state			1	1	1	1	1	1
GPT	GTADTRA0N	Compare match with GPT00.GTADTRA	0	0	0	0	0	1
	GTADTRB0N	Compare match with GPT00.GTADTRB	0	0	0	0	1	0
	GTADTRA1N	Compare match with GPT01.GTADTRA	0	0	0	0	1	1
	GTADTRB1N	Compare match with GPT01.GTADTRB	0	0	0	1	0	0
	GTADTRA2N	Compare match with GPT02.GTADTRA	0	0	0	1	0	1
	GTADTRB2N	Compare match with GPT02.GTADTRB	0	0	0	1	1	0
	GTADTRA3N	Compare match with GPT03.GTADTRA	0	0	0	1	1	1
	GTADTRB3N	Compare match with GPT03.GTADTRB	0	0	1	0	0	0
	GTADTRA4N	Compare match with GPT04.GTADTRA	0	0	1	0	0	1
	GTADTRB4N	Compare match with GPT04.GTADTRB	0	0	1	0	1	0
	GTADTRA5N	Compare match with GPT05.GTADTRA	0	0	1	0	1	1
	GTADTRB5N	Compare match with GPT05.GTADTRB	0	0	1	1	0	0
	GTADTRA6N	Compare match with GPT06.GTADTRA	0	0	1	1	0	1
	GTADTRB6N	Compare match with GPT06.GTADTRB	0	0	1	1	1	0
	GTADTRA7N	Compare match with GPT07.GTADTRA	0	0	1	1	1	1
	GTADTRB7N	Compare match with GPT07.GTADTRB	0	1	0	0	0	0
	GTADTRA8N	Compare match with GPT10.GTADTRA	0	1	0	0	0	1
	GTADTRB8N	Compare match with GPT10.GTADTRB	0	1	0	0	1	0
	GTADTRA9N	Compare match with GPT11.GTADTRA	0	1	0	0	1	1
	GTADTRB9N	Compare match with GPT11.GTADTRB	0	1	0	1	0	0
	GTADTRA10N	Compare match with GPT12.GTADTRA	0	1	0	1	0	1
	GTADTRB10N	Compare match with GPT12.GTADTRB	0	1	0	1	1	0
	GTADTRA11N	Compare match with GPT13.GTADTRA	0	1	0	1	1	1

Table 7.10-5 Selection of A/D Activation Sources by the TRSB[5:0] Bits (for Group B Only) (2/3)

Module	Source	Remarks	TRSB					
			[5]	[4]	[3]	[2]	[1]	[0]
	GTADTRB11N	Compare match with GPT13.GTADTRB	0	1	1	0	0	0
	GTADTRA12N	Compare match with GPT14.GTADTRA	0	1	1	0	0	1
	GTADTRB12N	Compare match with GPT14.GTADTRB	0	1	1	0	1	0
	GTADTRA13N	Compare match with GPT15.GTADTRA	0	1	1	0	1	1
	GTADTRB13N	Compare match with GPT15.GTADTRB	0	1	1	1	0	0
	GTADTRA14N	Compare match with GPT16.GTADTRA	0	1	1	1	0	1
	GTADTRB14N	Compare match with GPT16.GTADTRB	0	1	1	1	1	0
	GTADTRA15N	Compare match with GPT17.GTADTRA	0	1	1	1	1	1
	GTADTRB15N	Compare match with GPT17.GTADTRB	1	0	0	0	0	0
	GTADTRA0N or GTADTRB0N	Compare match with GPT00.GTADTRA or Compare match with GPT00.GTADTRB	1	0	0	0	0	1
	GTADTRA1N or GTADTRB1N	Compare match with GPT01.GTADTRA or Compare match with GPT01.GTADTRB	1	0	0	0	1	0
	GTADTRA2N or GTADTRB2N	Compare match with GPT02.GTADTRA or Compare match with GPT02.GTADTRB	1	0	0	0	1	1
	GTADTRA3N or GTADTRB3N	Compare match with GPT03.GTADTRA or Compare match with GPT03.GTADTRB	1	0	0	1	0	0
	GTADTRA4N or GTADTRB4N	Compare match with GPT04.GTADTRA or Compare match with GPT04.GTADTRB	1	0	0	1	0	1
	GTADTRA5N or GTADTRB5N	Compare match with GPT05.GTADTRA or Compare match with GPT05.GTADTRB	1	0	0	1	1	0
	GTADTRA6N or GTADTRB6N	Compare match with GPT06.GTADTRA or Compare match with GPT06.GTADTRB	1	0	0	1	1	1
	GTADTRA7N or GTADTRB7N	Compare match with GPT07.GTADTRA or Compare match with GPT07.GTADTRB	1	0	1	0	0	0
	GTADTRA8N or GTADTRB8N	Compare match with GPT10.GTADTRA or Compare match with GPT10.GTADTRB	1	0	1	0	0	1
	GTADTRA9N or GTADTRB9N	Compare match with GPT11.GTADTRA or Compare match with GPT11.GTADTRB	1	0	1	0	1	0
	GTADTRA10N or GTADTRB10N	Compare match with GPT12.GTADTRA or Compare match with GPT12.GTADTRB	1	0	1	0	1	1
	GTADTRA11N or GTADTRB11N	Compare match with GPT13.GTADTRA or Compare match with GPT13.GTADTRB	1	0	1	1	0	0
	GTADTRA12N or GTADTRB12N	Compare match with GPT14.GTADTRA or Compare match with GPT14.GTADTRB	1	0	1	1	0	1
	GTADTRA13N or GTADTRB13N	Compare match with GPT15.GTADTRA or Compare match with GPT15.GTADTRB	1	0	1	1	1	0
	GTADTRA14N or GTADTRB14N	Compare match with GPT16.GTADTRA or Compare match with GPT16.GTADTRB	1	0	1	1	1	1

Table 7.10-5 Selection of A/D Activation Sources by the TRSB[5:0] Bits (for Group B Only) (3/3)

Module	Source	Remarks	TRSB					
			[5]	[4]	[3]	[2]	[1]	[0]
	GTADTRA15N or GTADTRB15N	Compare match with GPT17.GTADTRA or Compare match with GPT17.GTADTRB	1	1	0	0	0	0
ELC	ELCTRG0		1	1	0	0	0	1

Table 7.10-6 Selection of A/D Activation Sources by the TRSA[5:0] Bits (1/2)

Module	Source	Remarks	TRSA					
			[5]	[4]	[3]	[2]	[1]	[0]
Trigger source de-selection state			1	1	1	1	1	1
External	ADTRG	Input pin for the trigger	0	0	0	0	0	0
GPT	GTADTRA0N	Compare match with GPT00.GTADTRA	0	0	0	0	0	1
	GTADTRB0N	Compare match with GPT00.GTADTRB	0	0	0	0	1	0
	GTADTRA1N	Compare match with GPT01.GTADTRA	0	0	0	0	1	1
	GTADTRB1N	Compare match with GPT01.GTADTRB	0	0	0	1	0	0
	GTADTRA2N	Compare match with GPT02.GTADTRA	0	0	0	1	0	1
	GTADTRB2N	Compare match with GPT02.GTADTRB	0	0	0	1	1	0
	GTADTRA3N	Compare match with GPT03.GTADTRA	0	0	0	1	1	1
	GTADTRB3N	Compare match with GPT03.GTADTRB	0	0	1	0	0	0
	GTADTRA4N	Compare match with GPT04.GTADTRA	0	0	1	0	0	1
	GTADTRB4N	Compare match with GPT04.GTADTRB	0	0	1	0	1	0
	GTADTRA5N	Compare match with GPT05.GTADTRA	0	0	1	0	1	1
	GTADTRB5N	Compare match with GPT05.GTADTRB	0	0	1	1	0	0
	GTADTRA6N	Compare match with GPT06.GTADTRA	0	0	1	1	0	1
	GTADTRB6N	Compare match with GPT06.GTADTRB	0	0	1	1	1	0
	GTADTRA7N	Compare match with GPT07.GTADTRA	0	0	1	1	1	1
	GTADTRB7N	Compare match with GPT07.GTADTRB	0	1	0	0	0	0
	GTADTRA8N	Compare match with GPT10.GTADTRA	0	1	0	0	0	1
	GTADTRB8N	Compare match with GPT10.GTADTRB	0	1	0	0	1	0
	GTADTRA9N	Compare match with GPT11.GTADTRA	0	1	0	0	1	1
	GTADTRB9N	Compare match with GPT11.GTADTRB	0	1	0	1	0	0
	GTADTRA10N	Compare match with GPT12.GTADTRA	0	1	0	1	0	1
	GTADTRB10N	Compare match with GPT12.GTADTRB	0	1	0	1	1	0
	GTADTRA11N	Compare match with GPT13.GTADTRA	0	1	0	1	1	1
	GTADTRB11N	Compare match with GPT13.GTADTRB	0	1	1	0	0	0
	GTADTRA12N	Compare match with GPT14.GTADTRA	0	1	1	0	0	1
	GTADTRB12N	Compare match with GPT14.GTADTRB	0	1	1	0	1	0
	GTADTRA13N	Compare match with GPT15.GTADTRA	0	1	1	0	1	1
	GTADTRB13N	Compare match with GPT15.GTADTRB	0	1	1	1	0	0
	GTADTRA14N	Compare match with GPT16.GTADTRA	0	1	1	1	0	1
	GTADTRB14N	Compare match with GPT16.GTADTRB	0	1	1	1	1	0
	GTADTRA15N	Compare match with GPT17.GTADTRA	0	1	1	1	1	1
	GTADTRB15N	Compare match with GPT17.GTADTRB	1	0	0	0	0	0

Table 7.10-6 Selection of A/D Activation Sources by the TRSA[5:0] Bits (2/2)

Module	Source	Remarks	TRSA					
			[5]	[4]	[3]	[2]	[1]	[0]
	GTADTRA0N or GTADTRB0N	Compare match with GPT00.GTADTRA or Compare match with GPT00.GTADTRB	1	0	0	0	0	1
	GTADTRA1N or GTADTRB1N	Compare match with GPT01.GTADTRA or Compare match with GPT01.GTADTRB	1	0	0	0	1	0
	GTADTRA2N or GTADTRB2N	Compare match with GPT02.GTADTRA or Compare match with GPT02.GTADTRB	1	0	0	0	1	1
	GTADTRA3N or GTADTRB3N	Compare match with GPT03.GTADTRA or Compare match with GPT03.GTADTRB	1	0	0	1	0	0
	GTADTRA4N or GTADTRB4N	Compare match with GPT04.GTADTRA or Compare match with GPT04.GTADTRB	1	0	0	1	0	1
	GTADTRA5N or GTADTRB5N	Compare match with GPT05.GTADTRA or Compare match with GPT05.GTADTRB	1	0	0	1	1	0
	GTADTRA6N or GTADTRB6N	Compare match with GPT06.GTADTRA or Compare match with GPT06.GTADTRB	1	0	0	1	1	1
	GTADTRA7N or GTADTRB7N	Compare match with GPT07.GTADTRA or Compare match with GPT07.GTADTRB	1	0	1	0	0	0
	GTADTRA8N or GTADTRB8N	Compare match with GPT10.GTADTRA or Compare match with GPT10.GTADTRB	1	0	1	0	0	1
	GTADTRA9N or GTADTRB9N	Compare match with GPT11.GTADTRA or Compare match with GPT11.GTADTRB	1	0	1	0	1	0
	GTADTRA10N or GTADTRB10N	Compare match with GPT12.GTADTRA or Compare match with GPT12.GTADTRB	1	0	1	0	1	1
	GTADTRA11N or GTADTRB11N	Compare match with GPT13.GTADTRA or Compare match with GPT13.GTADTRB	1	0	1	1	0	0
	GTADTRA12N or GTADTRB12N	Compare match with GPT14.GTADTRA or Compare match with GPT14.GTADTRB	1	0	1	1	0	1
	GTADTRA13N or GTADTRB13N	Compare match with GPT15.GTADTRA or Compare match with GPT15.GTADTRB	1	0	1	1	1	0
	GTADTRA14N or GTADTRB14N	Compare match with GPT16.GTADTRA or Compare match with GPT16.GTADTRB	1	0	1	1	1	1
	GTADTRA15N or GTADTRB15N	Compare match with GPT17.GTADTRA or Compare match with GPT17.GTADTRB	1	1	0	0	0	0
ELC	ELCTRG0		1	1	0	0	0	1

7.10.2.2.9 A/D Channel Select Register B0 (ADCm_ADANSB0)

Access Size : 16 bits

Address : <ADCm_base> + 0014h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ANSB0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 0	ANSB0[7:0]	0h	RW	A/D Conversion Analog Input Channel Select ANSB0[0] bit corresponds to ch0 and ANSB0[7] corresponds to ch7. 0b: ch0-ch7 are not subjected to conversion 1b: ch0-ch7 are subjected to conversion

7.10.2.2.10 A/D Error Control Register (ADCm_ADERCR)

ADERCR controls the error detection function. For details about errors, see **7.10.3.10 Error Detection Function (Overwrite Check Function)**.

Access Size : 8 bits
Address : <ADCm_base> + 01E2h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	OWEIE	-	-
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	OWEIE	0h	RW	Overwrite Error Interrupt Enable 0b: Disables interrupt generation when an overwrite error is detected. 1b: Enables interrupt generation when an overwrite error is detected.
1, 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

7.10.2.2.11 A/D Error Clear Register (ADCm_ADERCLR)

The A/D error clear register is a write-only register for clearing errors.

Access Size : 8 bits
Address : <ADCm_base> + 01E3h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	OWEC	-	-
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	OWEC	0h	RW	Overwrite Error Clear Writing 0b: Disables clearing overwrite errors. Writing 1b: Enables clearing overwrite errors.
1, 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.

7.10.2.2.12 A/D Overwrite Error Register (ADCm_ADOWER)

The ADOWER register is a status register that indicates that the results of A/D conversion in the ADDRn register were not read and an overwrite occurred.

Access Size : 16 bits
Address : <ADCm_base> + 01E8h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	OWE[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read.
7 to 0	OWE[7:0]	0h	R	Overwrite Error OWE[0] bit corresponds to ch0 and OWE[7] corresponds to ch7. 0b: No overwrite error occurred in ADDR0 to ADDR7. 1b: An overwrite error occurred in ADDR0 to ADDR7.

7.10.2.2.13 A/D Overwrite Error Extended Register (ADCm_ADOWEER)

The ADOWEER register is a status register that indicates that the results of A/D conversion in the ADDBLDRB, ADDBLDRA, ADDBLDR registers were not read and an overwrite occurred.

Access Size : 16 bits
Address : <ADCm_base> + 01EAh
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	DOWE	DAOWE	DBOWE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read.
2	DOWE	0h	R	A/D Data Duplication Register overwrite Error 0b: No overwrite error occurred in ADDBLDR. 1b: An overwrite error occurred in ADDBLDR.
1	DAOWE	0h	R	A/D Data Duplication Register A overwrite Error 0b: No overwrite error occurred in ADDBLDRA. 1b: An overwrite error occurred in ADDBLDRA.
0	DBOWE	0h	R	A/D Data Duplication Register B overwrite Error 0b: No overwrite error occurred in ADDBLDRB. 1b: An overwrite error occurred in ADDBLDRB.

7.10.2.2.14 A/D Event Link Control Register (ADCm_ADELCCR)

Access Size : 8 bits

Address : <ADCm_base> + 007Dh

Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	GCELC	ELCC[1:0]	
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
2	GCELC	0h	RW	Event control bit for Group C 0b: Not output event signal at the end of group scan of group C. 1b: Output event signal at the end of group scan of group C.
1, 0	ELCC[1:0]	0h	RW	Event link control bits Control the end of scan event (ada_adelcreq). To enable the ELCC[1:0] setting, set GCELC to 0. 00b: Output event signal at the end of scan except the end of group scan of group B, C 01b: Output event signal at the end of group scan of group B 1xb: Output event signal at the end of scan

7.10.2.2.15 A/D Group Scan Priority Control Register (ADCm_ADGSPCR)

Access Size : 16 bits
Address : <ADCm_base> + 0080h
Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GBRP	LGRRS	-	-	-	-	-	-	-	-	-	-	-	-	GBRSCN	PGS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15	GBRP	0h	RW	Low priority Group Single Scan Continuous Start*2 (Enabled only when PGS = 1b. Set 0 when PGS = 0b.) 0b: Single scan for Low priority group is not continuously activated 1b: Single scan for Low priority group is continuously activated
14	LGRRS	0h	RW	Restart Channel Select (Enabled only when PGS = 1b and GBRSCN = 1b, Set 0b when PGS = 0b.) 0b: Restart scan with first channel 1b: Restart scan with suspended channel
13 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1	GBRSCN	0h	RW	Low priority Group Restart Setting (Enabled only when PGS = 1b. Set 0 when PGS = 0b.) 0b: Scanning for group is not restarted when priority control is performed 1b: Scanning for group is restarted when priority control is performed
0	PGS	0h	RW	Group Priority Control Setting*1 0b: Operation is without group priority control 1b: Operation is with group priority control

Note 1. When the PGS bit is to be set to 1, the ADCSR.ADCS[1:0] bits must be set to 01b (group scan mode). If the bits are set to any other values, proper operation cannot be guaranteed.

Note 2. When the GBRP bit has been set to 1, single scan is performed continuously for low priority group regardless of the setting of the GBRSCN bit.

7.10.2.2.16 A/D Compare Function Window A/B Status Monitoring Register (ADCm_ADWINMON)

Access Size : 8 bits

Offset Address : <ADCm_base> + 008Ch

Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	MONCMPB	MONCMPA	-	-	-	MONCOMB
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read.
5	MONCMPB	0h	R	Comparing result monitor for window B 0b: Window B comparing condition does not match. 1b: Window B comparing condition match.
4	MONCMPA	0h	R	Comparing result monitor for window A 0b: Window A comparing condition does not match. 1b: Window A comparing condition match.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read.
0	MONCOMB	0h	R	Combination result monitor 0b: Window A/B combination condition does not match. 1b: Window A/B combination condition match.

7.10.2.2.17 A/D Compare Function Control Register (ADCm_ADCMPCR)

Access Size : 16 bits

Offset Address : <ADCm_base> + 0090h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPAIE	WCMPE	CMPBIE	-	CMPAE	-	CMPBE	-	-	-	-	-	-	-	CMPAB[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	R	RW	R	RW	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15	CMPAIE	0h	RW	Compare window A Interrupt Enable 0b: Generation of an ada_compai_n interrupt in response to matches with a condition for comparison is disabled. 1b: Generation of an ada_compai_n interrupt in response to matches with a condition for comparison is enabled.
14	WCMPE	0h	RW	Window Function enable 0b: Window function disabled 1b: Window function enabled
13	CMPBIE	0h	RW	Compare window B Interrupt Enable 0b: Generation of an ada_compbi_n interrupt in response to matches with a condition for comparison is disabled. 1b: Generation of an ada_compbi_n interrupt in response to matches with a condition for comparison is enabled.
12	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
11	CMPAE	0h	RW	Window A operation permission 0b: Window A is not in operation. Output of ada_elccondmtch/ada_elccondummtch is not permitted. 1b: Window A is in operation.
10	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
9	CMPBE	0h	RW	Window B operation permission 0b: Window B is not in operation. Output of ada_elccondmtch/ada_elccondummtch is not permitted. 1b: Window B is in operation.
8 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
1, 0	CMPAB[1:0]	0h	RW	Window A/B combination condition setting This bit is valid when both windows A/B are valid (CMPAE = 1b and CMPBE = 1b). 00b: Output ada_elccondmtch when window A comparison condition match OR window B comparison condition match, otherwise output ada_elccondummtch 01b: Output ada_elccondmtch when window A comparison condition match EXOR window B comparison condition match, otherwise output ada_elccondummtch 10b: Output ada_elccondmtch when window A comparison condition match AND window B comparison condition match, otherwise output ada_elccondummtch 11b: Setting prohibited.

7.10.2.2.18 A/D Compare Function Window A Channel Select Register 0 (ADCm_ADCMPANSR0)

Access Size : 16 bits

Offset Address : <ADCm_base> + 0094h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CMPCHA0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 0	CMPCHA0[7:0]	0h	RW	Window A Channel Select Set the CMPCHA0[7:0] bits when ADCSR.ADST = 0b. CMPCHA0[0] bit corresponds to ch0 and CMPCHA0[7] corresponds to ch7. 0b: Channel is not target 1b: Channel is target

7.10.2.2.19 A/D Compare Function Window A Comparison Condition Setting Register 0 (ADCm_ADCMPLR0)

Access Size : 16 bits

Offset Address : <ADCm_base> + 0098h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CMPLCHA0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

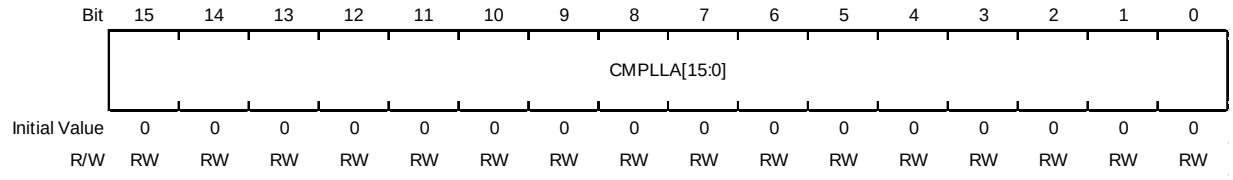
Bit	Bit Name	Initial Value	R/W	Description
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 0	CMPLCHA0 [7:0]	0h	RW	Window A comparison condition for target channel (ch0-7) setting CMPLCHA0[7] corresponds to channel 7 and CMPLCHA0[0] corresponds to channel 0. When the comparison result of each channel input matches the set condition, the ADCMPSR0.CMPSTCHA0n bit is set to 1b and a compare interrupt (ada_compai_n) is generated. - When window function is disabled (ADCMPCR.WCMPE = 0-: 0b: ADCMPDR0 register value > A/D-converted value 1b: ADCMPDR0 register value < A/D-converted value - When window function is enabled (ADCMPCR.WCMPE = 1)- 0b: A/D-converted value < ADCMPDR0 register value or ADCMPDR1 register value < A/D-converted value 1b: ADCMPDR0 register value < A/D-converted value < ADCMPDR1 register value

7.10.2.2.20 A/D Comparison Function Window A Lower Level Setting Register (ADCm_ADCMPDR0)

Access Size : 16 bits

Offset Address : <ADCm_base> + 009Ch

Initial Value : 0000h



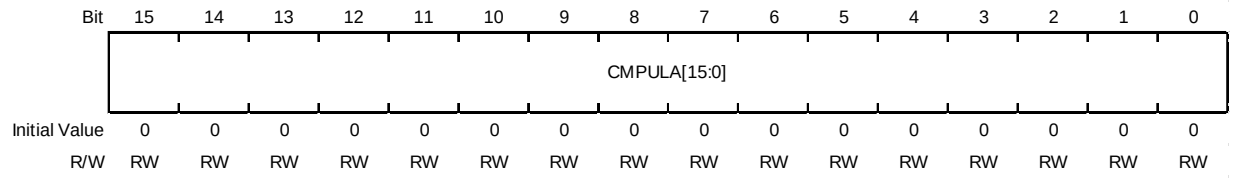
Bit	Bit Name	Initial Value	R/W	Description
15 to 0	CMPLLA[15:0]	0h	RW	Reference Lower data setting when using the compare function window A

7.10.2.2.21 A/D Comparison Function Window A Upper Level Setting Register (ADCm_ADCMPDR1)

Access Size : 16 bits

Offset Address : <ADCm_base> + 009Eh

Initial Value : 0000h



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	CMPULA[15:0]	0h	RW	Reference Upper data setting when using the compare function window A

7.10.2.2.22 A/D Comparison Function Window A Channel Status Register 0 (ADCm_ADCMPSR0)

Access Size : 16 bits

Address : <ADCm_base> + 00A0h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CMPSTCHA0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

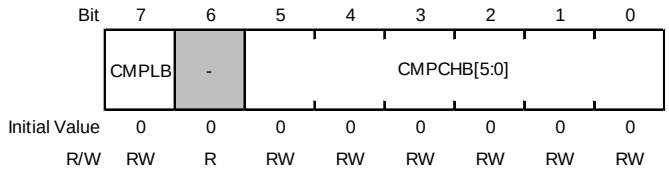
Bit	Bit Name	Initial Value	R/W	Description
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 0	CMPSTCHA0 [7:0]	0h	RW	Window A Status Flag CMPSTCHA0[0] bit corresponds to ch0 and CMPSTCHA0[7] corresponds to ch7. This flag shows the comparison result of each targeted channel, when window A is operating (ADCMPCR.CMPAE = 1b). If this flag is set while ADCMPCR.CMPAIE = 1b, a compare interrupt (ada_compai_n) request is generated. 0b: Condition does not match 1b: Condition matches

7.10.2.2.23 A/D Compare Function Window B Channel Select Register (ADCm_ADCMPBNSR)

Access Size : 8 bits

Address : <ADCm_base> + 00A6h

Initial Value : 00h



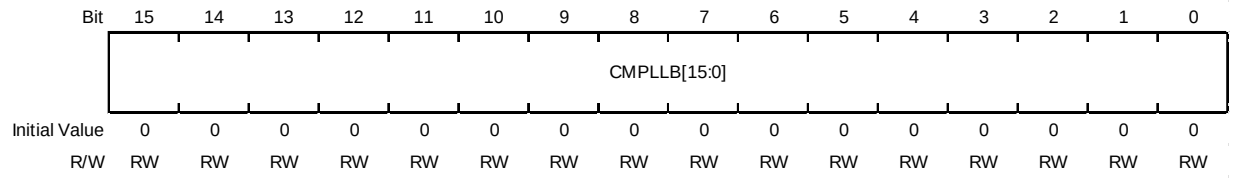
Bit	Bit Name	Initial Value	R/W	Description
7	CMPLB	0h	RW	Window B Comparison Condition Setting When the comparison result of target input matches the set condition, the ADCMPBSR0.CMPSTB bit is set to 1b and a compare interrupt (ada_compbi_n) is generated. - When window function is disabled (ADCMPCR.WCMPE = 0b)- 0b: ADWINLLB register value > A/D-converted value. 1b: ADWINLLB register value < A/D-converted value. - When window function is enabled (ADCMPCR.WCMPE = 1b)- 0b: A/D-converted value < ADWINLLB register value or ADWINULB register value < A/D-converted value. 1b: ADWINLLB register value < A/D-converted value < ADWINULB register value.
6	-	0h	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
5 to 0	CMPCHB[5:0]	0h	RW	Window B Channel Select Select channels for comparing with window B condition. Set the CMPCHB[5:0] bits when the ADCSR.ADST bit is 0b. 00h: ch0 01h: ch1 02h: ch2 : 07h: ch7 3Fh: No channel is selected Others: Setting prohibited

7.10.2.2.24 A/D Compare Function Window B Lower-Side Level Setting Register (ADCm_ADWINLLB)

Access Size : 16 bits

Address : <ADCm_base> + 00A8h

Initial Value : 0000h



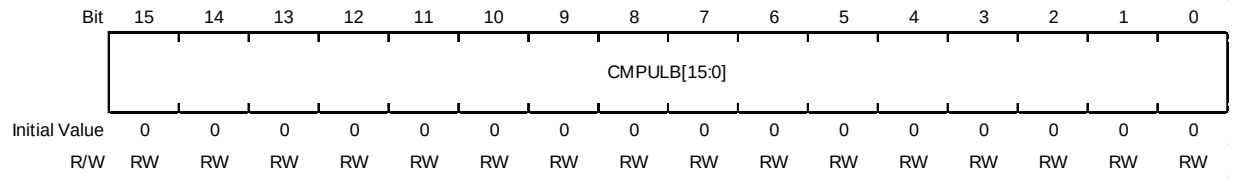
Bit	Bit Name	Initial Value	R/W	Description
15 to 0	CMPLLB[15:0]	0h	RW	Reference lower data setting when using the compare function window B

7.10.2.2.25 A/D Compare Function Window B Upper-Side Level Setting Register (ADCm_ADWINULB)

Access Size : 16 bits

Address : <ADCm_base> + 00AAh

Initial Value : 0000h



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	CMPULB[15:0]	0h	RW	Reference upper data setting when using the compare function window B

7.10.2.2.26 A/D Compare Function Window B Status Register (ADCm_ADCMPBSR)

Access Size : 8 bits

Address : <ADCm_base> + 00ACh

Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CMPSTB
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW0

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
0	CMPSTB	0h	RW0	Window B Flag When window B operation is enabled (ADCMPCR.CMPBE = 1b), this bit indicates the comparison result of channels to which window B comparison conditions are applied. 0b: Comparison conditions are not met 1b: Comparison conditions are met

7.10.2.2.27 A/D Channel Select Register C0 (ADCm_ADANSC0)

Access Size : 16 bits

Address : <ADCm_base> + 00D4h

Initial Value : 0000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ANSC0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 0	ANSC0[7:0]	0h	RW	A/D-Converted Channel Select for Group C in Group Scan Mode ANSC0[7] corresponds to ch7, and ANSC0[0] corresponds to ch0. 0b: Channel is not target 1b: Channel is target

7.10.2.2.28 A/D Group C Trigger Select Register (ADCm_ADGCTRGR)

Access Size : 8 bits
Offset Address : <ADCm_base> + 00D9h
Initial Value : 00h

Bit	7	6	5	4	3	2	1	0
	GRCE	GCADIE	TRSC[5:0]					
Initial Value	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
7	GRCE	0h	RW	Group C A/D Conversion Enable 0b: Group C A/D conversion is disabled 1b: Group C A/D conversion is enabled
6	GCADIE	0h	RW	Group C Scan Completion Interrupt Enable 0b: Group C scan completed interrupt (ada_gcadireq_n) is disabled 1b: Group C scan completed interrupt (ada_gcadireq_n) is enabled
5 to 0	TRSC[5:0]	0h	RW	Group C A/D Conversion Start Trigger Select Select starting trigger for A/D conversion of group C in group scan mode. See **Table 7.10-7** for more details.

Table 7.10-7 A/D Conversion Start Trigger for Group C in Group Scan Mode (1/3)

Module	Trigger	Description	TRSC					
			[5]	[4]	[3]	[2]	[1]	[0]
No trigger is selected			1	1	1	1	1	1
GPT	GTADTRA0N	Compare match with GPT00.GTADTRA	0	0	0	0	0	1
	GTADTRB0N	Compare match with GPT00.GTADTRB	0	0	0	0	1	0
	GTADTRA1N	Compare match with GPT01.GTADTRA	0	0	0	0	1	1
	GTADTRB1N	Compare match with GPT01.GTADTRB	0	0	0	1	0	0
	GTADTRA2N	Compare match with GPT02.GTADTRA	0	0	0	1	0	1
	GTADTRB2N	Compare match with GPT02.GTADTRB	0	0	0	1	1	0
	GTADTRA3N	Compare match with GPT03.GTADTRA	0	0	0	1	1	1
	GTADTRB3N	Compare match with GPT03.GTADTRB	0	0	1	0	0	0
	GTADTRA4N	Compare match with GPT04.GTADTRA	0	0	1	0	0	1
	GTADTRB4N	Compare match with GPT04.GTADTRB	0	0	1	0	1	0
	GTADTRA5N	Compare match with GPT05.GTADTRA	0	0	1	0	1	1
	GTADTRB5N	Compare match with GPT05.GTADTRB	0	0	1	1	0	0
	GTADTRA6N	Compare match with GPT06.GTADTRA	0	0	1	1	0	1
	GTADTRB6N	Compare match with GPT06.GTADTRB	0	0	1	1	1	0
	GTADTRA7N	Compare match with GPT07.GTADTRA	0	0	1	1	1	1
	GTADTRB7N	Compare match with GPT07.GTADTRB	0	1	0	0	0	0
	GTADTRA8N	Compare match with GPT10.GTADTRA	0	1	0	0	0	1
	GTADTRB8N	Compare match with GPT10.GTADTRB	0	1	0	0	1	0
	GTADTRA9N	Compare match with GPT11.GTADTRA	0	1	0	0	1	1
	GTADTRB9N	Compare match with GPT11.GTADTRB	0	1	0	1	0	0
	GTADTRA10N	Compare match with GPT12.GTADTRA	0	1	0	1	0	1
	GTADTRB10N	Compare match with GPT12.GTADTRB	0	1	0	1	1	0
	GTADTRA11N	Compare match with GPT13.GTADTRA	0	1	0	1	1	1
	GTADTRB11N	Compare match with GPT13.GTADTRB	0	1	1	0	0	0
	GTADTRA12N	Compare match with GPT14.GTADTRA	0	1	1	0	0	1

Table 7.10-7 A/D Conversion Start Trigger for Group C in Group Scan Mode (2/2)

Module	Trigger	Description	TRSC					
			[5]	[4]	[3]	[2]	[1]	[0]
	GTADTRB12N	Compare match with GPT14.GTADTRB	0	1	1	0	1	0
	GTADTRA13N	Compare match with GPT15.GTADTRA	0	1	1	0	1	1
	GTADTRB13N	Compare match with GPT15.GTADTRB	0	1	1	1	0	0
	GTADTRA14N	Compare match with GPT16.GTADTRA	0	1	1	1	0	1
	GTADTRB14N	Compare match with GPT16.GTADTRB	0	1	1	1	1	0
	GTADTRA15N	Compare match with GPT17.GTADTRA	0	1	1	1	1	1
	GTADTRB15N	Compare match with GPT17.GTADTRB	1	0	0	0	0	0
	GTADTRA0N or GTADTRB0N	Compare match with GPT00.GTADTRA or Compare match with GPT00.GTADTRB	1	0	0	0	0	1
	GTADTRA1N or GTADTRB1N	Compare match with GPT01.GTADTRA or Compare match with GPT01.GTADTRB	1	0	0	0	1	0
	GTADTRA2N or GTADTRB2N	Compare match with GPT02.GTADTRA or Compare match with GPT02.GTADTRB	1	0	0	0	1	1
	GTADTRA3N or GTADTRB3N	Compare match with GPT03.GTADTRA or Compare match with GPT03.GTADTRB	1	0	0	1	0	0
	GTADTRA4N or GTADTRB4N	Compare match with GPT04.GTADTRA or Compare match with GPT04.GTADTRB	1	0	0	1	0	1
	GTADTRA5N or GTADTRB5N	Compare match with GPT05.GTADTRA or Compare match with GPT05.GTADTRB	1	0	0	1	1	0
	GTADTRA6N or GTADTRB6N	Compare match with GPT06.GTADTRA or Compare match with GPT06.GTADTRB	1	0	0	1	1	1
	GTADTRA7N or GTADTRB7N	Compare match with GPT07.GTADTRA or Compare match with GPT07.GTADTRB	1	0	1	0	0	0
	GTADTRA8N or GTADTRB8N	Compare match with GPT10.GTADTRA or Compare match with GPT10.GTADTRB	1	0	1	0	0	1
	GTADTRA9N or GTADTRB9N	Compare match with GPT11.GTADTRA or Compare match with GPT11.GTADTRB	1	0	1	0	1	0
	GTADTRA10N or GTADTRB10N	Compare match with GPT12.GTADTRA or Compare match with GPT12.GTADTRB	1	0	1	0	1	1
	GTADTRA11N or GTADTRB11N	Compare match with GPT13.GTADTRA or Compare match with GPT13.GTADTRB	1	0	1	1	0	0
	GTADTRA12N or GTADTRB12N	Compare match with GPT14.GTADTRA or Compare match with GPT14.GTADTRB	1	0	1	1	0	1
	GTADTRA13N or GTADTRB13N	Compare match with GPT15.GTADTRA or Compare match with GPT15.GTADTRB	1	0	1	1	1	0
	GTADTRA14N or GTADTRB14N	Compare match with GPT16.GTADTRA or Compare match with GPT16.GTADTRB	1	0	1	1	1	1

Table 7.10-7 A/D Conversion Start Trigger for Group C in Group Scan Mode (3/3)

Module	Trigger	Description	TRSC					
			[5]	[4]	[3]	[2]	[1]	[0]
	GTADTRA15N or GTADTRB15N	Compare match with GPT17.GTADTRA or Compare match with GPT17.GTADTRB	1	1	0	0	0	0
ELC	ELCTRG0		1	1	0	0	0	1

7.10.3 Operation

7.10.3.1 Scanning Operation

In scanning, A/D conversion is performed sequentially on the analog inputs of the specified channels.

There are three operating modes: single scan mode, continuous scan mode, and group scan mode. In single scan mode, one or more specified channels are scanned once. In continuous scan mode, one or more specified channels are scanned repeatedly until the ADCSR.ADST bit is cleared to 0b from 1b. In group scan mode, the selected channels of group A and the selected channels of group B (and the selected channels of group C) are scanned once after starting to be scanned according to the respective internal triggers (GPT, ELC).

In single scan mode and continuous scan mode, A/D conversion is performed for ANn pins selected by the ADANSA0 register, starting from the pin with the smallest number n. In group scan mode, A/D conversion is performed for ANn pins selected by the ADANSA0 register for group A, and performed for ANn pins selected by the ADANSB0 register for group B and performed for ANn pins selected by the ADANSC0 register for group C, respectively, starting from the pin with the smallest number n.

Double trigger mode is to be used with single scan mode or group scan mode. With double trigger mode being enabled, A/D conversion data of a channel selected by the ADCSR.DBLANS[4:0] bits is duplicated only if the conversion is started by any of the internal triggers (GPT, ELC) selected by the ADSTRGR.TRSA[5:0] bits. In group scan mode, only group A can use the double trigger.

To enable double trigger extended mode, select double trigger mode, and then select the internal trigger by the ADSTRGR.TRSA[5:0].

- ADSTRGR.TRSA[5:0] = 100101b
  GTADTRA4N or GTAGTRB4N
- ADSTRGR.TRSA[5:0] = 101000b
  GTADTRA7N or GTADTRB7N

In the double trigger extended mode, in addition to the normal double trigger mode operation, the A/D conversion data activated by the odd number trigger (GTADTRA4N , GTADTRA7N) is stored in the A/D data duplex register A (ADDBLDRA), and the A/D conversion data activated by the even number trigger (GTADTRB4N , GTADTRB7N) is stored in the A/D Data Duplication register B (ADDBLDRB).

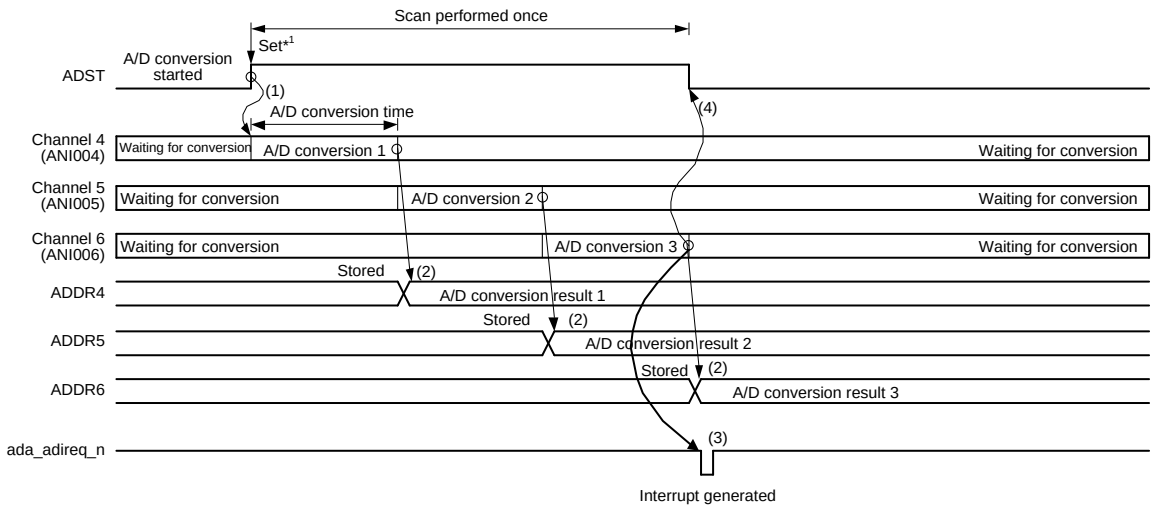
If two types of trigger source occur at the same time, the data is not sorted by the trigger factor and the A/D conversion data is stored in the Data Duplication register B (ADDBLDRB). If one trigger is performing A/D conversion and the other trigger is input, the other trigger will be ignored.

7.10.3.2 Single Scan Mode

7.10.3.2.1 Basic operation

In basic operation of single scan mode, A/D conversion is performed once on the analog input of the specified channels as below.

1. When the ADCSR.ADST bit becomes 1b (A/D conversion start) due to software or internal trigger (GPT, ELC) input, A/D conversion starts in ascending order of n of ANn selected in the ADANSA0 register.
2. When the A/D conversion of one channel is completed, the A/D conversion result is stored in the corresponding A/D data register (ADDRy).
3. If the ADCSR.ADIE bit is set to 1b (ada_adireq_n interrupt enabled at the end of scanning) after A/D conversion of all selected channels is completed, an ada_adireq_n interrupt request is generated.
4. ADCSR.ADST bit holds 1b (A/D conversion start) during A/D conversion and is automatically cleared when A/D conversion of all selected channels is completed, and the 12-bit A/D converter goes into a standby state.



Note 1. The arrows in this figure indicate instruction execution by software.

Figure 7.10-2 Example of Operation in Single Scan Mode (Basic Operation ANI004 to ANI006 Selected)

7.10.3.2.2 A/D conversion in double trigger mode

When double trigger mode is selected in single scan mode, A/D conversion is performed for two rounds of single scan operation started by a internal trigger (GPT, ELC).

A/D conversion Data Duplication is enabled by setting the channel number to be duplicated to the ADCSR.DBLANS[4:0] bits and setting ADCSR.DBLE to 1b. When ADCSR.DBLE is set to 1b, the channel selection of the ADANSA0 register is disabled.

To select the double trigger mode, use ADSTRGR.TRSA[5:0] to select the internal trigger (GPT, ELC), set the ADCSR.EXTRG bit to 0b, and set the ADCSR.TRGE bit to 1b. Also, do not use software trigger.

1. When the ADCSR.ADST bit becomes 1b (A/D conversion start) due to the internal trigger (GPT, ELC) input, the A/D conversion of the 1 channel selected by the ADCSR.DBLANS[4:0] bits starts.
2. When the A/D conversion is completed, the A/D conversion result is stored in the corresponding A/D data register (ADDRy).
3. ADCSR.ADST is automatically cleared and the 12-bit A/D converter is in the standby state. At this time, the ada_adireq_n interrupt does not occur regardless of the setting of the ADCSR.ADIE bit (ada_adireq_n interrupt enabled by the end of scanning).
4. When the ADCSR.ADST bit becomes 1b (A/D conversion start) by the second internal trigger input, A/D conversion of the 1 channel selected by the ADCSR.DBLANS[4:0] bits starts.
5. When the A/D conversion is completed, the A/D conversion result is stored in the A/D Data Duplication register (ADDBLDR) dedicated to the double trigger mode.
6. If the ADCSR.ADIE bit is set to 1 (ada_adireq_n interrupt enabled at the end of scanning), an ada_adireq_n interrupt request is generated.
7. The ADCSR.ADST bit holds 1 (A/D conversion start) during A/D conversion, is automatically cleared when A/D conversion is completed, and the 12-bit A/D converter is in the standby state.

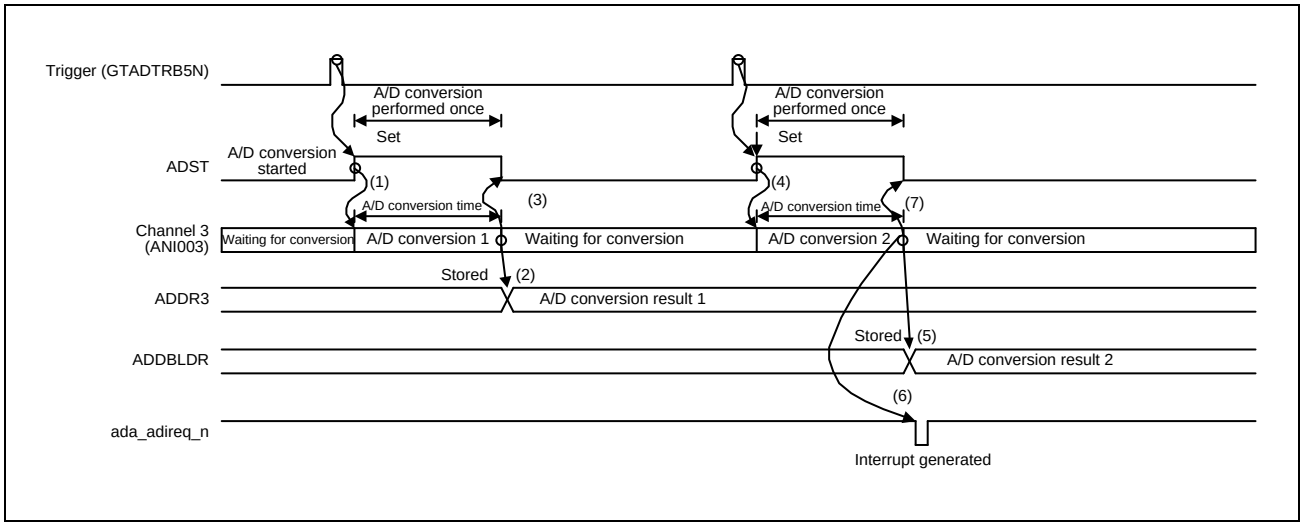


Figure 7.10-3 Example of Operation in Single Scan Mode (Double Trigger Mode, ANI003 Duplicated, GTADTRB5N is Selected as the Trigger)

7.10.3.2.3 Extended operations when double trigger mode is selected

When double trigger mode is selected in single scan mode, single scan is performed twice.

To enable double trigger extended mode, and then select the following internal trigger by the ADSTRGR.TRSA[5:0].

- ADSTRGR.TRSA[5:0] = 100001b
  GTADTRA0N or GTADTRB0N
- ADSTRGR.TRSA[5:0] = 100110b
  GTADTRA5N or GTADTRB5N

And set the ADCSR.EXTRG bit to 0b and the ADCSR.TRGE bit to 1b. Also, do not use software trigger.

A/D conversion Data Duplication is enabled by setting the channel number to be duplicated to the ADCSR.DBLANS[4:0] bits and setting ADCSR.DBLE to 1b. When ADCSR.DBLE is set to 1b, the channel selection of the ADANSA0 register is disabled.

1. When the ADCSR.ADST bit becomes 1b (A/D conversion start) by GTADTRA0N, A/D conversion of 1 channel selected by the ADCSR.DBLANS[4:0] bits starts.
2. When the A/D conversion is completed, the A/D conversion result is stored in the corresponding A/D data register (ADDRy) and A/D Data Duplication register A (ADDBLDRA).
3. ADCSR.ADST is automatically cleared and the 12-bit A/D converter is in the standby state. At this time, the ada_adireq_n interrupt does not occur regardless of the setting of the ADCSR.ADIE bit (ada_adireq_n interrupt enabled by the end of scanning).
4. When the ADCSR.ADST bit becomes 1b (A/D conversion start) by GTADTRB0N input, A/D conversion of 1 channel selected by the ADCSR.DBLANS[4:0] bits starts.
5. When the A/D conversion is completed, the A/D conversion result is stored in the A/D Data Duplication register (ADDBLDR) and the A/D Data Duplication register B (ADDBLDRB).
6. If the ADCSR.ADIE bit is set to 1b (ada_adireq_n interrupt enabled at the end of scanning), an ada_adireq_n interrupt request is generated.
7. The ADCSR.ADST bit holds 1b (A/D conversion start) during A/D conversion, is automatically cleared when A/D conversion is completed, and the 12-bit A/D converter is in the standby state.

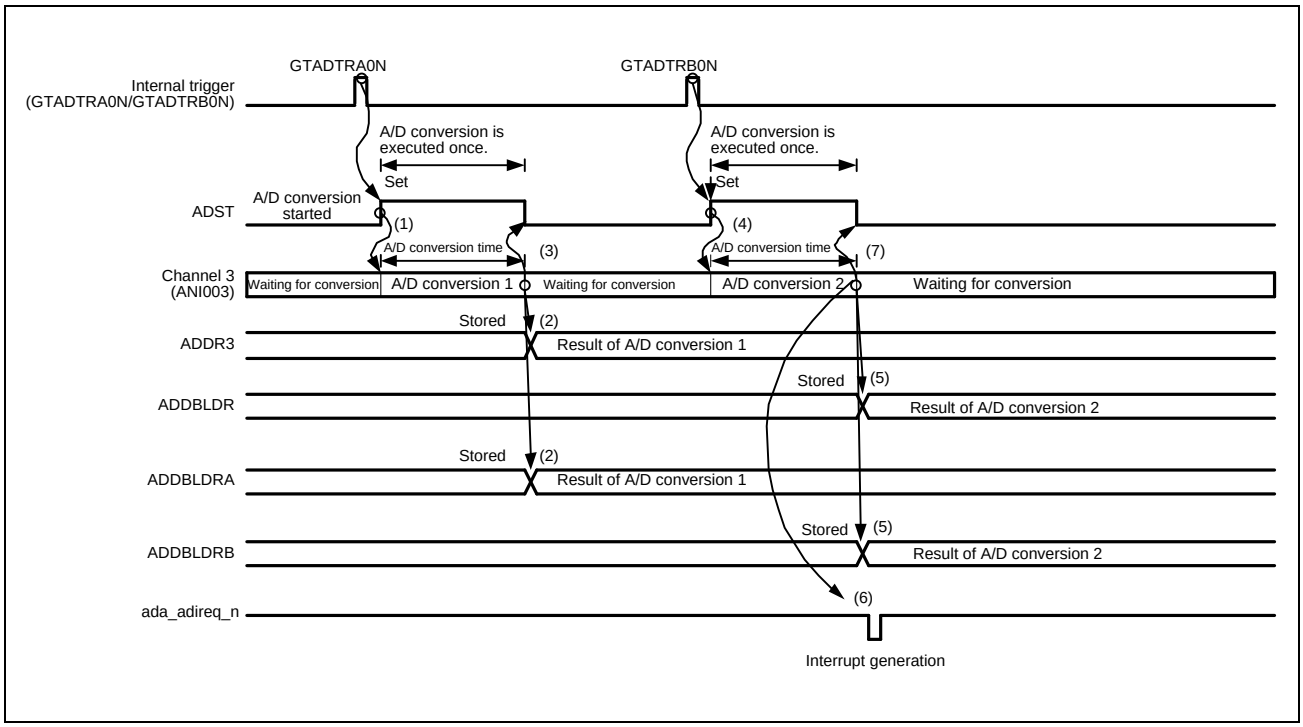


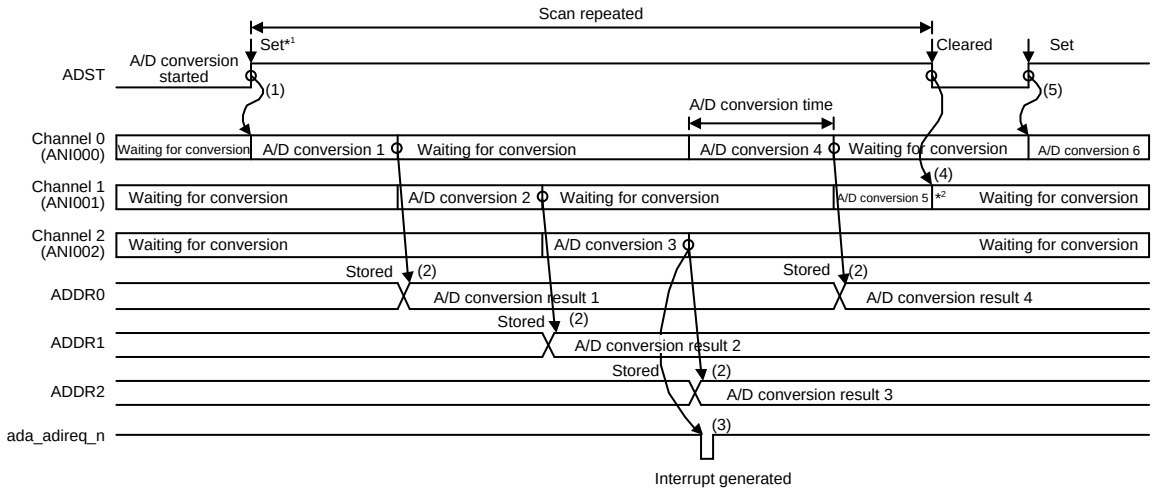
Figure 7.10-4 Example of Extended Operation in Double Trigger Mode (Duplication Selected for ANI003, GTADTRA0N , GTADTRB0N Selected)

7.10.3.3 Continuous Scan Mode

7.10.3.3.1 Basic operation (not use S/H circuits)

In basic operation of continuous scan mode, A/D conversion is performed repeatedly on the analog input of the channels selected by the ADANSA0 register.

1. When the ADCSR.ADST bit becomes 1 (A/D conversion start) by software, internal trigger (GPT, ELC), or external trigger, A/D conversion starts in ascending order of n of ANn selected in the ADANSA0 register.
2. When the A/D conversion of one channel is completed, the A/D conversion result is stored in the corresponding A/D data register (ADDRy).
3. If the ADCSR.ADIE bit is set to 1b (ada_adireq_n interrupt enabled at the end of scanning) after A/D conversion of all selected channels is completed, an ada_adireq_n interrupt request is generated. In addition, the 12-bit A/D converter continuously starts A/D conversion in ascending order of n of ANn selected in the ADANSA0 register.
4. The ADCSR.ADST bit is not cleared automatically, and (2) to (3) are repeated during 1b (A/D conversion start). When the ADCSR.ADST bit is set to 0b (A/D conversion stopped), A/D conversion is stopped and the 12-bit A/D converter is in the standby state.
5. After that, when the ADCSR.ADST bit becomes 1b (A/D conversion starts), A/D conversion starts again in ascending order of n of ANn selected in the ADANSA0 register.



Note 1. Ignored the result of A/D conversion 5.

Figure 7.10-5 Example of Operation in Continuous Scan Mode (Basic Operation ANI000 to ANI002 Selected)

7.10.3.4 Group Scan Mode

7.10.3.4.1 Basic operation

In basic operation of group scan mode, A/D conversion is performed once on the analog inputs of all the specified channels in each group (A and B or A, B and C) after scanning is started by an internal trigger (GPT, ELC). Scan operation of each group is similar to the scan operation in single scan mode.

To set the trigger for group scan mode, select the trigger for group A with the ADSTRGR.TRSA[5:0] bits, select the trigger for group B with the ADSTRGR.TRSB[5:0] bits, and select the trigger for group C with the ADGCTRGR.TRSC[5:0] bits. Triggers for groups A, B, and C should be separate triggers so that scans for groups A, B, and C do not occur at the same time. Also, do not use software triggers. For the channels to be scanned, select the group A channel in the ADANSA0 register, select the group B channel in the ADANSB0 register, and select the group C channel in the ADANSC0 register.

The following is an example of the operation of the group scan mode by the internal trigger from the GPT. Group A is set to start conversion with a GTADTRA0N trigger from GPT, Group B is set to start conversion with a GTADTRA3N trigger from GPT, and Group C is set to start conversion with a GTADTRA4N trigger from GPT.

1. Start scanning of group A with GTADTRA0N trigger from GPT.
2. If the ADCSR.ADIE bit is set to 1b (ada_adireq_n interrupt enabled) at the end of group A scanning, the ada_adireq_n interrupt is output.
3. Start scanning Group B with GTADTRA3N trigger from GPT.
4. If the ADCSR.GBADIE bit is set to 1b (ada_gbadireq_n interrupt enabled) at the end of Group B scanning, an ada_gbadireq_n interrupt is output.
5. Trigger GTADTRA4N from GPT to start scanning for Group C.
6. If the ADGCTRGR.GCADIE bit is set to 1b (ada_gcadireq_n interrupt enabled) at the end of Group C scanning, an ada_gcadireq_n interrupt is output.

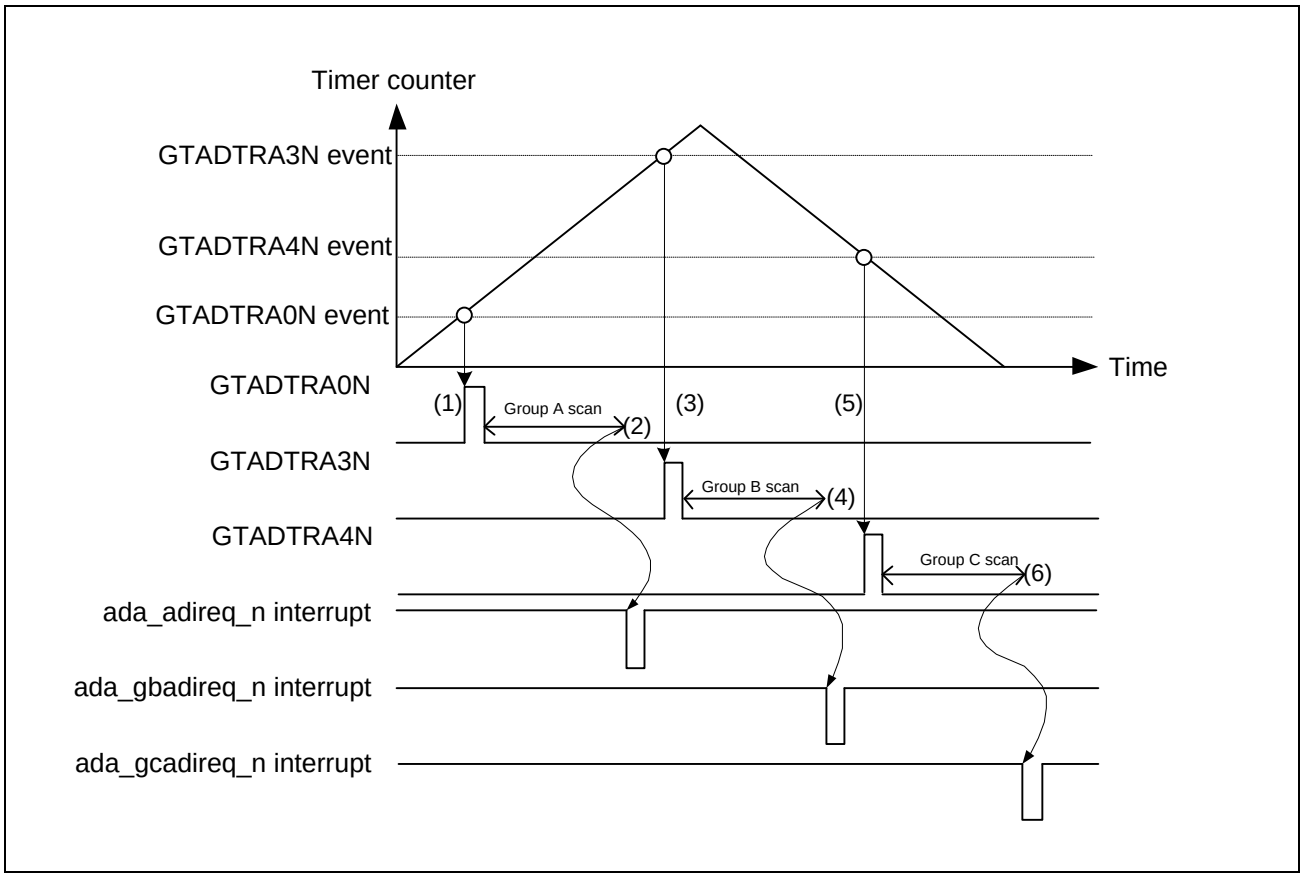


Figure 7.10-6 Example of Operation in Group Scan Mode (Basic Operation Trigger from GPT)

7.10.3.4.2 A/D conversion in double trigger mode

When double trigger mode is selected in group scan mode, group A controls two single scan mode executions starting with an internal trigger (GPT, ELC) as a series of operations. Group B and Group C behave in the same way as single scan mode started with an internal trigger (GPT, ELC).For group scan mode trigger settings, the ADSTRGR.TRSA[5:0] bits select the trigger for group A, the ADSTRGR.TRSB[5:0] bits select the trigger for group B, and ADGCTRGR.TRSC[5:0] bits selects the trigger for Group C. Triggers for groups A, B, and C should be separate triggers so that scans for groups A, B, and C do not occur at the same time. Also, do not use software triggers.

If you select “GTADTRA0N or GTADTRB0N” to “GTADTRA15N or GTADTRB15N” as the trigger for group A in the ADSTRGR.TRSA[5:0] bits, it operates in double trigger extended mode.

For the channels to be scanned, select the group A channel with the ADCSR.DBLANS[4:0] bits, select the group B channel with the ADANSB0 register, and select the group C channel with the ADANSC0 register.

A/D conversion Data Duplication is enabled by setting the channel number to be duplicated to the ADCSR.DBLANS[4:0] bits and setting the ADCSR.DBLE bit to 1.

The following shows an operation example when the group scan mode and double trigger mode are set by the internal trigger from the GPT. Group A is set to start conversion with a GTADTRA1N trigger from GPT, Group B is set to start conversion with a GTADTRB7N trigger from GPT, and Group C is set to start conversion with a GTADTRA5N trigger from GPT.

1. Start scanning Group C with the GTADTRA5N trigger from the GPT.
2. If the ADGCTRGR.GCADIE bit is set to 1 (ada_gcadireq_n interrupt enabled) at the end of Group C scanning, the ada_gcadireq_n interrupt is output.
3. The GTADTRB7N trigger from the GPT starts scanning for group B.
4. If the ADCSR.GBADIE bit is set to 1b (ada_gbadireq_n interrupt enabled) at the end of Group B scanning, an ada_gbadireq_n interrupt is output.
5. The first GTADTRA1N trigger from the GPT will start the first scan of Group A.
6. At the end of the first scan of Group A, the converted data is stored in ADDRy, and the ada_adireq_n interrupt request does not occur regardless of the setting of the ADCSR.ADIE bit.
7. The second GTADTRA1N trigger from the GPT will start the second scan of Group A.
8. At the end of the second scan of group A, the conversion data is stored in ADDBLDR, and if the ADCSR.ADIE bit is set to 1b (ada_adireq_n interrupt enabled), the ada_adireq_n interrupt is output.
9. The second GTADTRB7N trigger from the GPT starts the second scan of group B.
10. At the end of the second scan of Group B, if the ADCSR.GBADIE bit is set to 1b (ada_gcadireq_n interrupt enabled), an ada_gbadireq_n interrupt is output.
11. The second GTADTRA5N trigger from the GPT starts the second scan of Group C.
12. At the end of the second scan of Group C, if the ADGCTRGR.GCADIE bit is set to 1b (ada_gcadireq_n interrupt enabled), an ada_gcadireq_n interrupt is output.

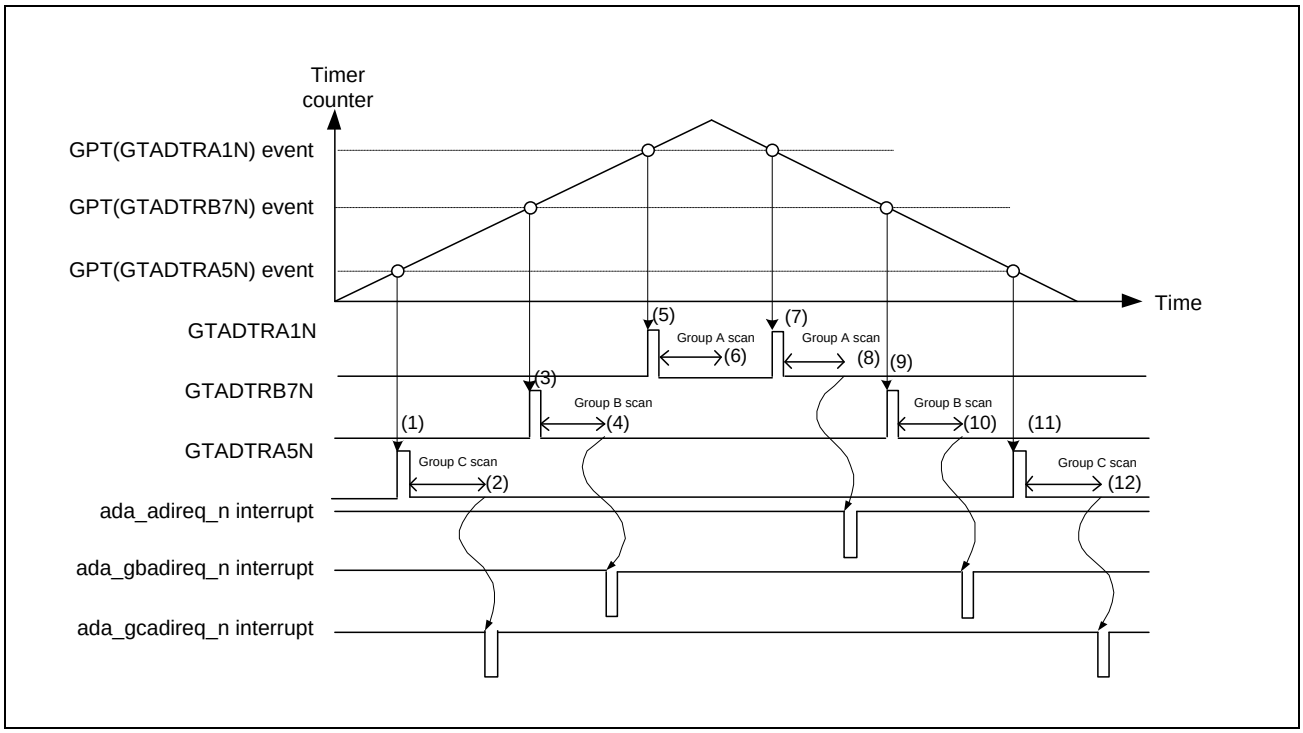


Figure 7.10-7 Example of Operation in Group Scan Mode with Double Trigger (Trigger from GPT)

7.10.3.4.3 Operation under Group Priority Control

When the ADGSPCR.PGS bit is set to 1b in group scan mode, group priority operation is performed. Group priority is in the order of group A > group B > group C. The number of groups used in group scan mode can be selected from either 2 (Groups A, B) or 3 (Groups A, B, C) depending on the setting of ADGCTRGR.GRCE. When setting the PGS bit in the ADPGSCR register to 1, follow the procedure in **Figure 7.10-8** to set the relevant registers. If the procedure is not followed, A/D conversion operation and stored data are not guaranteed.

In the basic operation of group scan mode, the trigger input generated during scanning of groups A, B, and C is ignored, and the scanning operation of groups A, B, and C is the same as that of single scan mode.

In the group priority operation, if there is a trigger input of the priority group during the scan of the low priority group, the scan of the low priority group is interrupted and the priority group is scanned.

When the ADGSPCR.GBRSCN bit is 0b, the low priority group is in the standby state after the priority group scan is completed. Also, the low priority group trigger input that occurs during the scan is ignored.

When the ADGSPCR.GBRSCN bit is 1b, the scan of the low priority group is automatically re-executed after the scan of the priority group is completed. In addition, the trigger input of the low priority group that occurred during the scan of the priority group is valid, and the scan of the low priority group is automatically executed after the scan of the priority group is completed.

When the ADGSPCR.GBRSCN bit is 1b and the ADGSPCR.LGRRS is 0b, the scan of the low priority group is restarted from the beginning. Also, when ADGSPCR.LGRRS is 1b, the scan of the low priority group is re-executed from the interrupted channel.

Table 7.10-8 shows the setting of the ADGSPCR.GBRSCN bit and the operation when a trigger is input during scanning.

When the ADGSPCR.GBRP bit is set to 1, the scan operation of the lowest priority group is the operation of executing a single scan continuously.

For the trigger setting in group scan mode, select the group A internal trigger with the ADSTRGR.TRSA[5:0] bits, and set the group B internal trigger different from the group A trigger with the ADSTRGR.TRSB[5:0] bits. Select a Group C internal trigger that is different from the Group A and B triggers on the ADGCTRGR.TRSC[5:0] bits.

When the group scan mode is set to 2 groups (ADGCTRGR.GRCE bit is set to 0b) and the ADGSPCR.GBRP bit is set to 1, the ADSTRGR.TRSB[5:0] bits is set to 3Fh.

If the group scan mode is set to 3 groups (ADGCTRGR.GRCE bit is set to 1) and the ADGSPCR.GBRP bit is set to 1, set the ADGCTRGR.TRSC[5:0] bits to 3Fh.

For the channels to be scanned, select the group A channel in the ADANSA0 register, select the group B channel in the ADANSB0 register, and select the group C channel in the ADANSC0 register.

Table 7.10-9 shows Group priority operation setting and operation mode of 2 groups, and **Table 7.10-10** shows Group priority operation setting and operation mode of 3 groups.

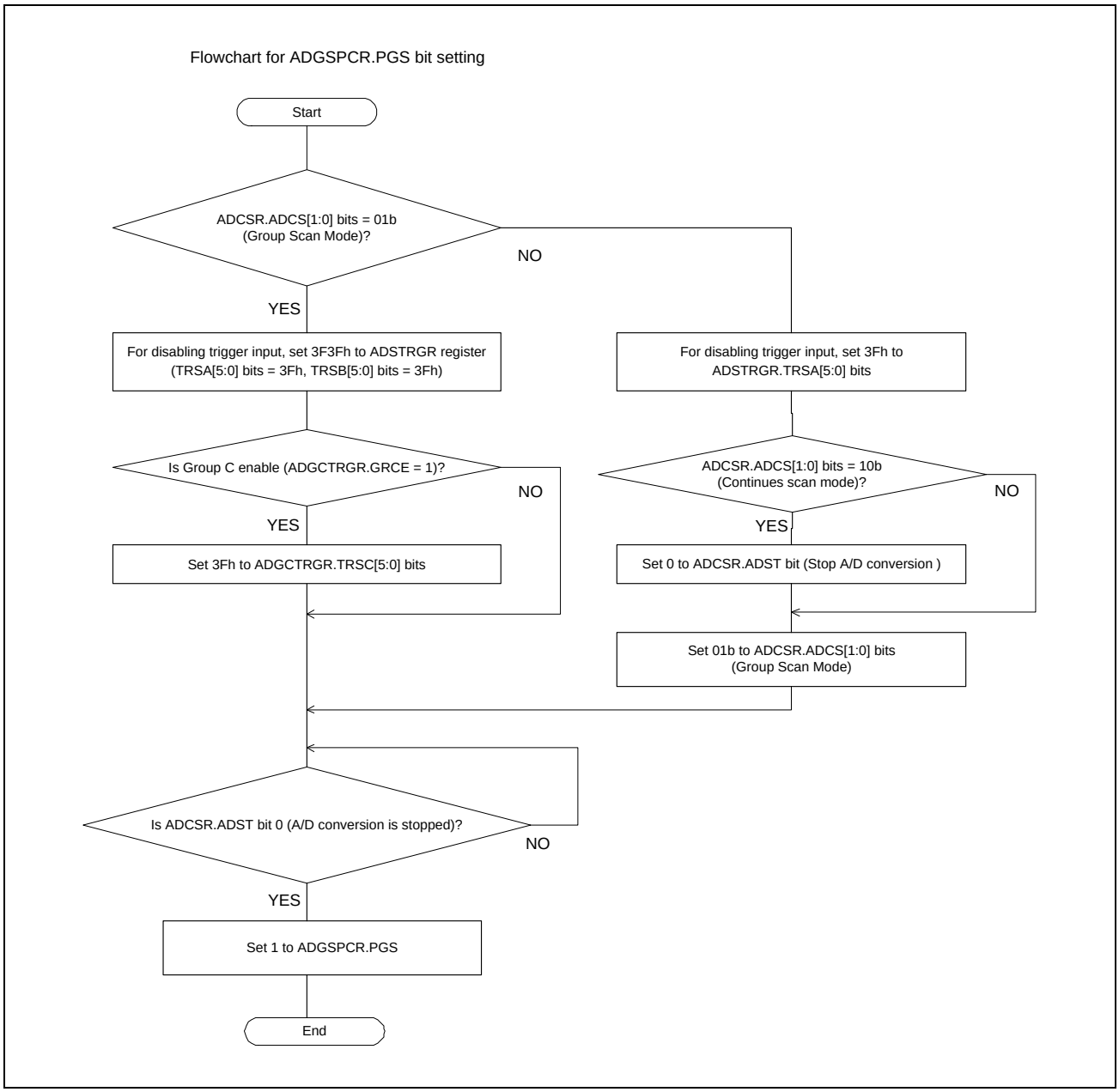


Figure 7.10-8 Flowchart for ADGSPCR.PGS Bit Setting

Table 7.10-8 Control of A/D Conversion Operations according to the Settings of the ADGSPCR.GBRSCN Bit

A/D Conversion Operation	Trigger Input	ADGSPCR.GBRSCN = 0b	ADGSPCR.GBRSCN = 1b
When A/D conversion for group A is in progress	Input of trigger for group A	Trigger input is ineffective.	Trigger input is ineffective.
	Input of trigger for group B	Trigger input is ineffective.	A/D conversion is performed on group B after A/D conversion on group A is completed.
	Input of trigger for group C	Trigger input is ineffective.	A/D conversion is performed on group C after A/D conversion on group A is completed.
When A/D conversion for group B is in progress	Input of trigger for group A	Conversion for group B that is in progress is discontinued and conversion for group A starts.	• Conversion in progress for group B is discontinued and conversion for group A starts. • Conversion for group B starts after conversion for group A is completed.
	Input of trigger for group B	Trigger input is ineffective.	Trigger input is ineffective.
	Input of trigger for group C	Trigger input is ineffective.	A/D conversion is performed on group C after A/D conversion on group B is completed.
When A/D conversion for group C is in progress	Input of trigger for group A	Conversion for group C that is in progress is discontinued and conversion for group A starts.	• Conversion in progress for group C is discontinued and conversion for group A starts. • Conversion for group C starts after conversion for group A is completed.
	Input of trigger for group B	Conversion for group C that is in progress is discontinued and conversion for group B starts.	• Conversion in progress for group C is discontinued and conversion for group B starts. • Conversion for group C starts after conversion for group B is completed.
	Input of trigger for group C	Trigger input is ineffective.	Trigger input is ineffective.

Table 7.10-9 Group Priority Operation Setting and Operation Mode for 2 Groups (ADGSPCR.PGS = 1b, ADGCTRGR.GRCE = 0b)

ADGSPCR			
GBRSCN	LGRRS	GBRP	Operation Mode
0	—	0	Group priority operation of 2 groups (groups A and B). When the trigger of group A is input, the scan of group B ends (does not re-execute).
1	0	0	Group priority operation of 2 groups (groups A and B). After the scan of group B is interrupted, group B restarts the scan from the beginning of the channel specified by ADANSB0 after the scan of group A is completed.
1	1	0	Group priority operation of 2 groups (groups A and B). After the scan of group B is interrupted, group B resumes the scan from the interrupted channel among the channels specified by ADANSB0 after the scan of group A is completed.
—	0	1	Group priority operation of 2 groups (groups A and B). Group B continuously performs a single scan without inputting a start trigger. After the group B scan is interrupted, the single scan is restarted from the beginning of the channel specified by ADANSB0 after the group A scan is completed.
1	1	1	Group priority operation of 2 groups (groups A and B). Group B continuously performs a single scan without inputting a start trigger. After the scan of group B is interrupted, the single scan is restarted from the interrupted channel among the channels specified by ADANSB0 after the scan of group A is completed.

Table 7.10-10 Group Priority Operation Setting and Operation Mode for 3 Groups (ADGSPCR.PGS = 1b, ADGCTRGR.GRCE = 1b)

ADGSPCR			
GBRSCN	LGRRS	GBRP	Operation Mode
0	—	0	Group priority operation of 3 groups (groups A, B, C). When the trigger of group A is input, the scan of group B ends (does not re-execute). When the trigger of group A or group B is input, the scan of group C ends (does not re- execute).
0	—	1	Group priority operation of 3 groups (groups A, B, C). When the trigger of group A is input, the scan of group B ends (does not re-execute). Group C continuously performs a single scan without inputting a start trigger. After the group C scan is interrupted, the scan is restarted from the beginning of the channel specified by ADANSC0 after the group A/B scan is completed.
1	0	0	Group priority operation of 3 groups (groups A, B, C). After the scan of group B is suspended, the scan restarts from the top channel of the channels specified by ADANSB0 after the group A scan is completed. After the scan of group C is suspended, the scan restarts from the top channel of the channels specified by ADANSC0 after the group A/B scan is completed.
1	1	0	Group priority operation of 3 groups (groups A, B, C). After the scan of group B is suspended, the scan restarts from the interrupted channel of the channels specified by ADANSB0 after the group A scan is completed. After the scan of group C is suspended, the scan restarts from the interrupted channel of the channels specified by ADANSC0 after the group A/B scan is completed.
1	0	1	Group priority operation of 3 groups (groups A, B, C). After the scan of group B is interrupted, the scan is restarted from the beginning of the channel specified by ADANSB0 after the scan of group A is completed. Group C continuously performs a single scan without inputting a start trigger. After the group C scan is interrupted, the single scan is restarted from the beginning of the channel specified by ADANSC0 after the group A/B scan is completed.
1	1	1	Group priority operation of 3 groups (groups A, B, C). After the A/D conversion process of group B is interrupted, the interrupted channel is restarted from the channels specified by ADANSB0 after the scan of group A is completed. Group C continuously performs a single scan without inputting a start trigger. After the group C scan is interrupted, the single scan is restarted from the interrupted channel among the channels specified by ADANSC0 after the group A/B scan is completed.

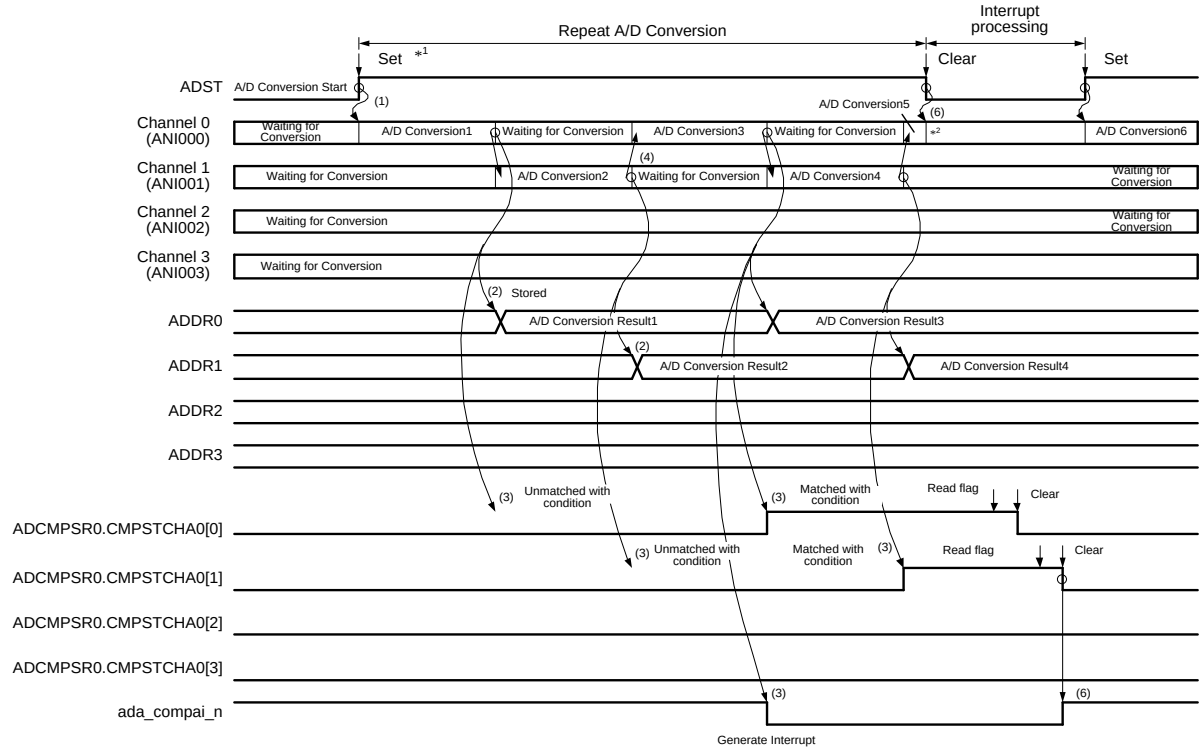
7.10.3.5 Compare Function (Window A, Window B)

7.10.3.5.1 Compare function window A/B

The compare function is a function that compares the reference value set in the register with the A/D conversion result, and the reference value can be set for each window (A/B). Double trigger mode cannot be used when using the compare function. The major difference between window A and window B is that window B has one selectable channel and different interrupt output signals.

Operation using compare function in combination with continuous scan mode is described below.

1. When the ADCSR.ADST bit becomes 1b (A/D conversion start) by software, internal trigger (GPT, ELC), or external trigger, A/D conversion of the selected channel is started.
2. When the A/D conversion is completed, the A/D conversion result is stored in the corresponding A/D data register (ADDRy). When ADCMPCR.CMPAE = 1b, if it is set for window A in the ADCMPANSR0 register, it is compared with the ADCMPDR0 register setting value. When ADCMPCR.CMPBE = 1b, if the ADCMPBNSR register is set for window B, it will be compared with the ADWINULB/ADWINLLB register setting value.
3. As a result of comparison, when the condition set in ADCMPLR0 is matched in window A, the flag (ADCMPSR0. CMPSTCHA0n) bit of the compare function window A is set to 1b. At this time, if the ADCMPCR.CMPAIE bit is set to 1b, an ada_compai_n interrupt request (level) is generated. Similarly, when window B matches the conditions set in ADCMPBNSR.CMPLB, the compare function window B flag (ADCMPBSR.CMPSTB) bit is set to 1. At this time, if the ADCMPCR.CMPBIE bit is set to 1b, an ada_compbi_n interrupt request (level) is generated.
4. When all the selected A/D conversions and comparisons are completed, scanning is performed again.
5. After accepting the ada_compai_n/ ada_compbi_n interrupt, set the ADCSR.ADST bit to 0b (A/D conversion stopped) and execute processing for the channel for which the compare flag is set.
6. When all the compare flags in window A are cleared, the ada_compai_n interrupt request is canceled. Similarly, clearing the compare window B flag cancels the ada_compbi_n interrupt request. If you want to perform the compare again, start the A/D conversion again.



Note 1. The data under conversion is ignored.

Figure 7.10-9 Example of Operation of Compare Function (Compare Target: ANI000-ANI003)

7.10.3.5.2 Event output in compare function

For the event output of the compare function, specify the upper and lower reference values for each of windows A/B, compare the A/D conversion value of the selected channel with the upper/lower reference values, and then the event (ada_elccondmtch / ada_elccondunmtch) is output according to the event condition (A or B, A and B, A exor B) from the result of satisfying/failing the comparison condition of windows A and B. When using this function, perform A/D conversion in single scan mode.

In window A, any channel of ANn00-ANn07 can be selected. When multiple channels are selected in window A and the comparison condition is satisfied in one of the channels, the comparison condition is satisfied in window A. In window B, one channel of ANn00-ANn15 can be selected.

The setting procedure and setting example when using the event output of the compare function are shown below.

1. ADCSR.ADCS bit must be 00b (single scan mode).
2. Select the channel to be used for window A in the ADCMPANSR0 register, set the comparison condition of the compare function window in the ADCMPLR0 register, and set the upper/lower reference value in the ADCMPDR0/1 register.
3. Select the channel to be used for window B and the comparison condition in the ADCMPBNSR register, and set the upper/lower reference values in the ADWINULB/ADWINLLB registers.
4. In the ADCMPCR register, set the window A/B compound condition, window A/B operation permission, and interrupt output permission.

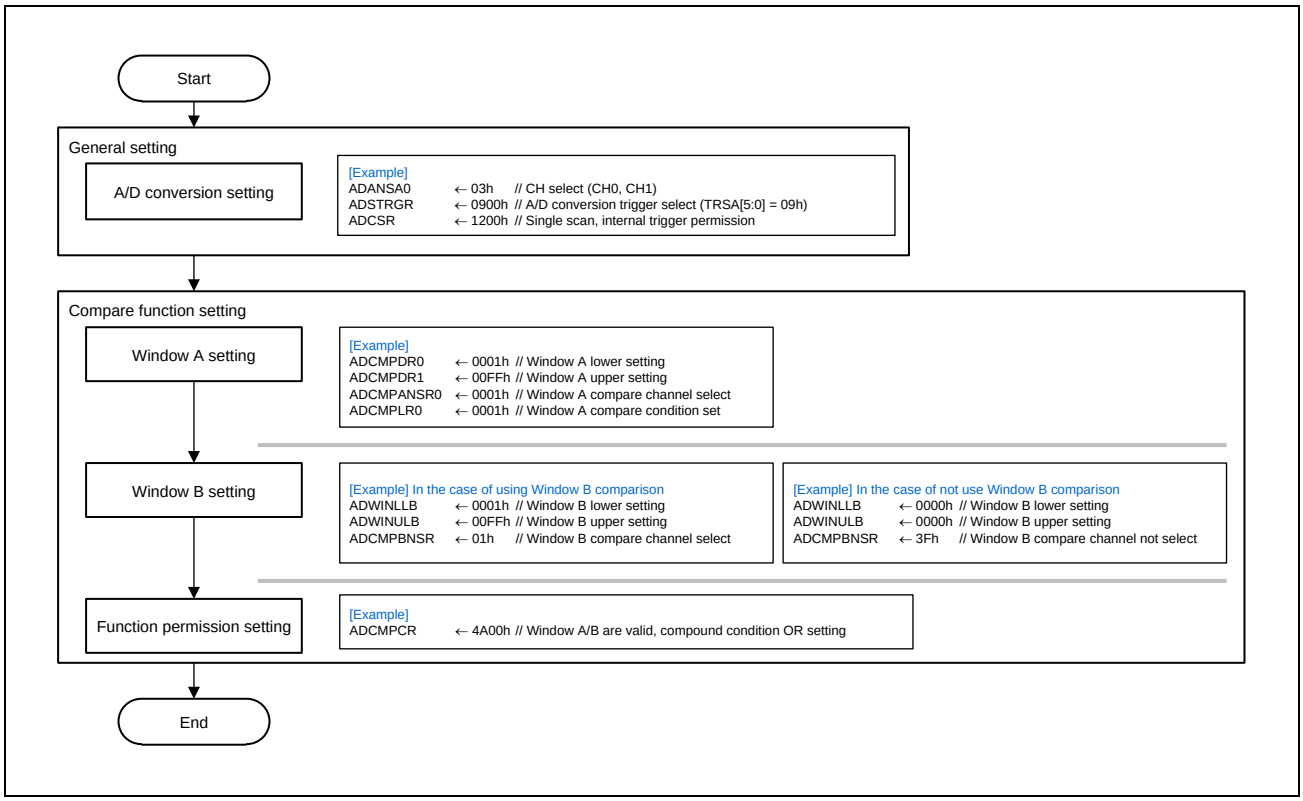


Figure 7.10-10 Example of Setting for Using Event Output of Compare Function

Precautions when using event output using only window A of the compare function

- Enable both windows A and B. (ADCMPCR.CMPAE = 1b, ADCMPCR.CMPBE = 1b)
- Set the compound condition setting for windows A and B to “OR condition”. (ADCMPCR.CMPAB[1:0] = 00b)

- Set the comparison target channel in window B to “Unselected”. (ADCMPBNSR.CMPCHB[5:0] = 111111b).
- Set the comparison condition of window B to 0 < conversion result < 0, which is “always inconsistent”. (ADCMPCR.WCMPE = 1b, ADWINLLB.CMPLLB[15:0] = ADWINULB.CMPULB[15:0] = 0000h, ADCMPBNSR.CMPLB = 1b)

Figure 7.10-11 shows example of event output operation of compare function.

The scan end event is output when one single scan ends. Also, depending on the setting of ADCMPCR.CMPAB[1:0], the match/unmatch event (ada_elccondmtch / ada_elccondunmtch) is output with a delay of 1.

NOTE

Match/unmatch events are exclusive outputs, and both events are not output at the same time.

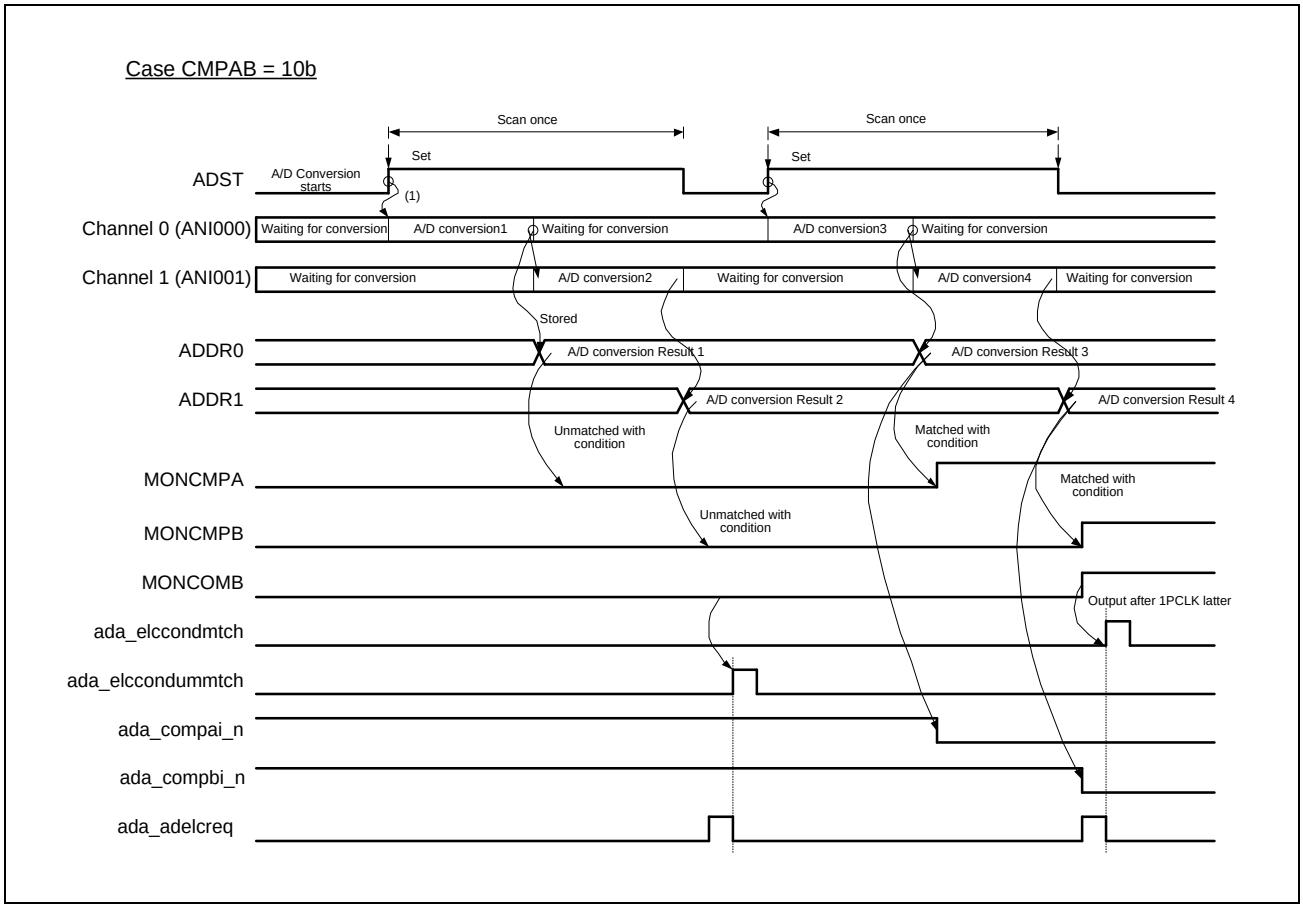


Figure 7.10-11 Example of Operation for Event Output of Compare Function

Precautions when using event output of the compare function

The event output of the compare function outputs match/unmatch according to the setting of ADCMPCR.CMPAB[1:0] from the comparison result of window A and the comparison result of window B. The comparison result of window A is the logical sum of the comparison result of the channel to be compared in window A.

The comparison result of window A and the comparison result of window B are updated every A/D conversion of the channel to be compared, and the comparison result is retained even after the single scan is completed. To clear the comparison result (0), set ADCMPCR.CMPAE and ADCMPCR.CMPBE to 0b.

7.10.3.5.3 Compare function constraint

The compare function has the following restrictions.

- Double trigger mode cannot be used together.
  (ADDBLDR, ADDBLDRA, ADDBLDRB are not subject to the compare function.)
- If you want to use match/unmatch event output, set the single scan mode.
- The same CH cannot be set in window A and window B.
- Set so that the upper reference value ≥ the lower reference value.

7.10.3.6 A/D Data Register Automatic Clearing Function

By setting the ADCER.ACE bit to 1b, these are used when the CPU and DMAC read the A/D data registers (ADDRy, ADDBLDR, ADDBLDRA, ADDBLDRB). The register can be cleared automatically to 0000h.

By using this function, it is possible to detect an un-updated failure of the ADDRy, ADDBLDR, ADDBLDRA, and ADDBLDRB registers. The following is an example when the automatic clear function of the ADDRy register is disabled/ enabled.

When the ADCER.ACE bit is 0b (automatic clearing prohibited), the old data (0111h) becomes the value of the ADDRy register when the A/D conversion result (0222h) is not written to the ADDRy register for some reason. Furthermore, when the value of this ADDRy register is read to the general-purpose register by using the A/D conversion end interrupt, the old data (0111h) can be saved in the general-purpose register. However, when checking for un-updated data, it is necessary to check while holding old data in RAM and general-purpose registers one by one.

When the ADCER.ACE bit is 1 (automatic clearing enabled), the ADDRy register is automatically cleared to 0000h when ADDRy = 0111h is read by the CPU and DMAC. After that, when 0222h of the A/D conversion result cannot be transferred to the ADDRy register for some reason, the cleared data (0000h) remains as the ADDRy register value. If the value of this ADDRy register is read to a general-purpose register by using the A/D conversion end interrupt, 0000h is held in the general-purpose register. By simply checking that the read data value is 0000h, it can be determined that there was an un-updated failure of the ADDRy register.

7.10.3.7 A/D-Converted Value Addition/Average Function

In A/D-converted value addition function, the same channel is A/D-converted two, three, four or sixteen consecutive times and the sum of the converted values is stored in the data register. In A/D-converted value average function, the same channel is A/D-converted two or four consecutive times and the mean of the converted values is stored in the data register. The use of the average of these results can improve the accuracy of A/D conversion, depending on the types of noise components that affect A/D conversion. This function, however, cannot always guarantee an improvement in A/D conversion accuracy. The A/D-converted value addition/average mode can be specified when A/D conversion of the analog input of the selected channels is selected.

7.10.3.8 Starting A/D Conversion with External Trigger

The A/D conversion can be started by the input of an external trigger. To start up the A/D converter by an external trigger, the A/D conversion start trigger select bits (ADSTRGR.TRSA[5:0]) should be set to 00_0000b and a low-level signal should be input to the external trigger (ADTRG). Both the ADCSR.TRGE and ADCSR.EXTRG bits then should be set to 1b.

Internal triggers cannot be selected for groups B and C used in group scan mode.

7.10.3.9 Starting A/D Conversion with Internal Trigger from Peripheral Module

A/D conversion can be started by the internal trigger from GPT and ELC. When starting A/D conversion with an internal trigger, the ADCSR.TRGE bit is set to 1b, the ADCSR.EXTRG bit is set to 0b, the ADSTRGR.TRSA[5:0], ADSTRGR.TRSB[5:0], and ADGCTRGR.TRSC[5:0] bits are set to the appropriate trigger factor.

7.10.3.10 Error Detection Function (Overwrite Check Function)

The overwrite error detection function detects update of the A/D conversion results held in various A/D data registers, without a need of reading those results. The 12-bit A/D converter can generate an ada_adereq_n interrupt, which is an overwrite error interrupt request to the error control module (ECM).

Setting the ADERCR.OWEIE to 1b enables generation of an ada_adereq_n interrupt and clearing the bit to 0b disables generation of an ada_adereq_n interrupt. Reading the ADOWER and ADOWEER registers can determine the A/D data register in which the overwrite error occurred. Writing 1b to the ADERCLR.OWEC clears all the overwrite error flags held in the ADOWER and ADOWEER registers.

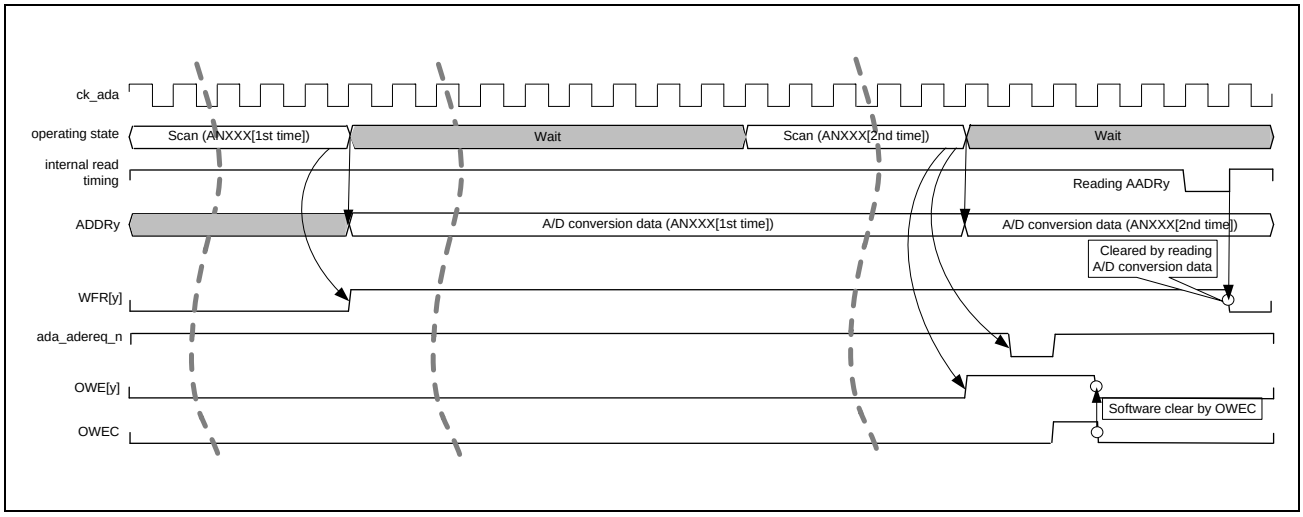


Figure 7.10-12 Example of Generation of AD Overwrite Error Interrupt

7.10.3.11 Event Link Function

7.10.3.11.1 Event output to ELC

ELC can use the ada_adireq_n interrupt request signal as an event signal (ada_adelcreq) to link to a preset module (ada_gbadireq_n / ada_gcadireq_n interrupt, ada_compai_n/ ada_compbi_n interrupt cannot be used as an event signal). The event signal can be output regardless of the setting of the corresponding interrupt request permission bit.

The 12-bit A/D converter outputs the scan end event (ada_adelcreq), compare function match event (ada_elccondmtch), and unmatch event (ada_elccondunmtch). The scan end event (ada_adelcreq) outputs a “High pulse” for 1 cycle of ADC_0_PCLK at the same output timing as the interrupt output (ada_adireq_n) regardless of the setting of ADCSR.ADIE.

The compare match/unmatch event (ada_elccondmtch / ada_elccondunmtch) to ELC outputs a “High pulse” for 1 cycle of ADC_0_PCLK at a timing delayed by 1 cycle of ADC_0_PCLK from the interrupt output (ada_adireq_n) regardless of the setting of ADCSR.ADIE. When using the compare match/unmatch event to ELC (ada_elccondmtch / ada_elccondunmtch), set it to single scan mode.

7.10.3.11.2 Starting A/D conversion with ELC event

The 12-bit A/D converter can start A/D conversion by a preset event by setting EVTSELx in ICU.

7.10.3.11.3 Notes on ELC trigger

The trigger from the ELC (ELCTRG0) is invalid during A/D conversion.

7.10.4 Usage Notes

7.10.4.1 Notes on Reading Data Registers

The A/D data registers, A/D data duplication registers, A/D data duplication register A, and A/D data duplication register B must be read in 16-bit word units, to avoid unmatch with the upper value and lower value by reading in byte units separately.

7.10.4.2 Notes on Stopping A/D Conversion

7.10.4.2.1 A/D conversion stop procedure

When using an external or internal trigger as the A/D conversion start condition, follow the procedure in the flowchart in **Figure 7.10-13** to stop the A/D conversion.

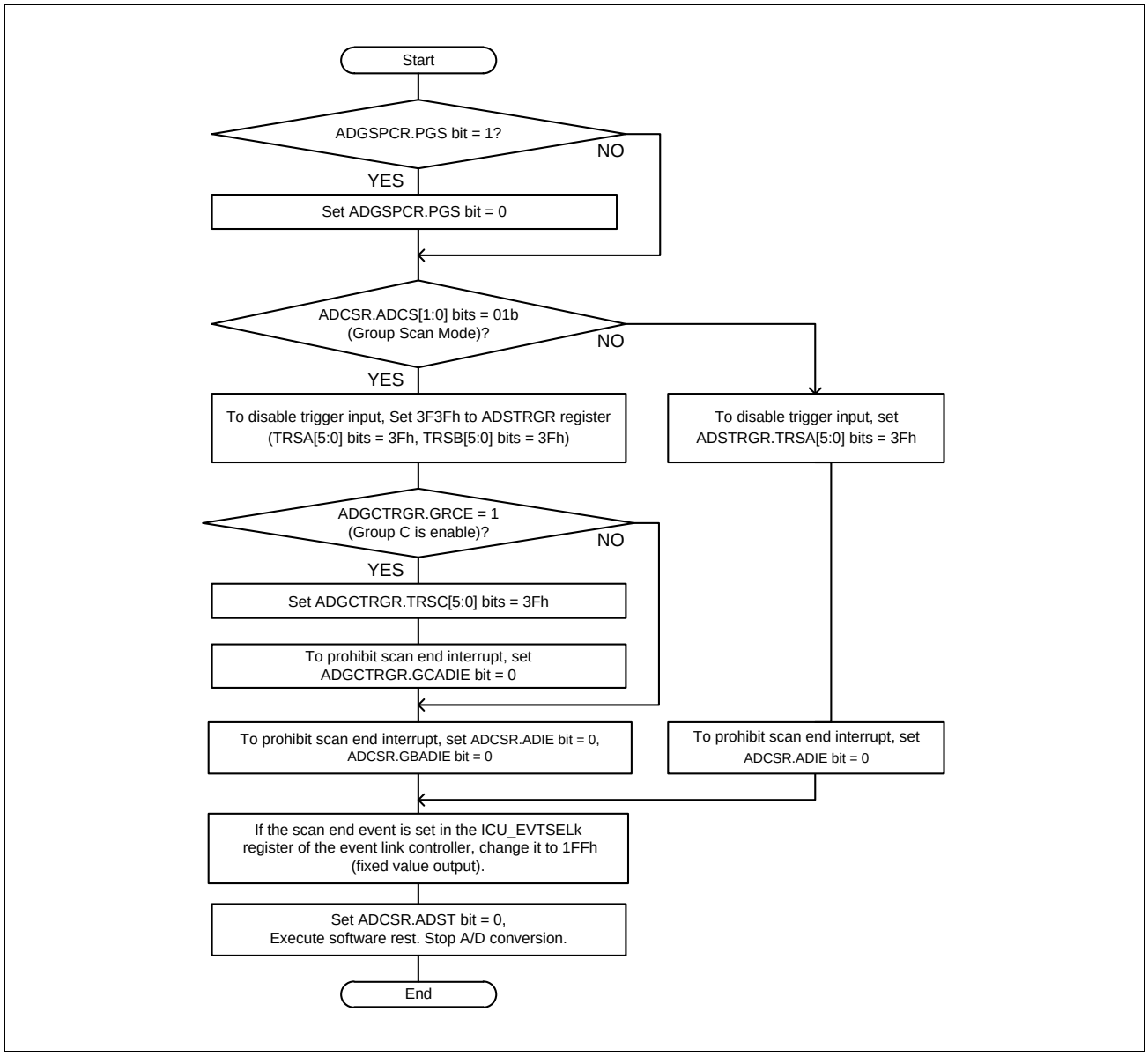


Figure 7.10-13 A/D Conversion Stop Procedure

It takes 2 cycles of ADC_0_ADCLK from a software clear until the soft macro stops scanning. If you want to set the following after executing a software clear, enter a wait of 2 cycles of ADC_0_ADCLK or more.

- Scan end interrupt enabled
- Event link controller scan end event valid setting
- Start of A/D conversion by software
- Trigger input valid setting

7.10.4.2.2 Notes on mode/status bits

Initialize or reset the odd/even judgment and monitor bits of the compare function in the double trigger mode as necessary.

- Double trigger mode starts from the first scan operation when ADCSR.DBLE is set from 0b to 1b.
- To initialize the monitor bits (MONCMPA, MONCMPB) of the compare function, set ADCMPCR.CMPAE and ADCMPCR.CMPBE to 0b.

7.10.4.3 A/D conversion Forced Stop and Operation Timing at Starts

Up to 6 cycles of the ADC_0_ADCLK are required to start the A/D converter by setting the ADCSR.ADST bit = 1b from the stopped status.

When the ADCSR.ADST bit is set to 0b and the A/D conversion is forcibly stopped, the analog part of the 12-bit A/D converter requires a maximum of 2 ADC_0_ADCLK cycles to stop operating.

7.10.4.4 Notes on Scan End Interrupt Processing

When scanning the same analog input twice using any trigger, if the CPU does not complete reading out the A/D-converted data by the time the A/D conversion of the first analog input for the second scan ends after the first scan end interrupt is generated, the first A/D-converted data is overwritten with the second A/D-converted data and an overwrite error occurs. For details on how to detect overwrite errors in A/D conversion results, please see **7.10.3.10 Error Detection Function (Overwrite Check Function)**.

7.10.4.5 Notes on Using Hardware and Software Triggers Simultaneously.

If the next A/D conversion start trigger is input during A/D conversion, the trigger will be ignored. Therefore, when using hardware trigger (i.e., internal or external trigger) and software trigger at the same time, ensure that the hardware trigger is not input during the execution of A/D conversion started by the software trigger. Also, ensure that the software trigger is not input during A/D conversion started by the hardware trigger. Software can check whether A/D conversion is in progress by ADREF. ADSCACT bit.

7.10.4.6 Power-down Mode Settings

The 12-bit A/D converter can be set to power-down mode by SYS_ADC_CFG.SY_MSTP_ADA bit. By default, the A/D converter is in power-down mode (SYS_ADC_CFG.SY_MSTP_ADA = 1b). After leaving power-down mode by setting the SYS_ADC_CFG.SY_MSTP_ADA bit to 0b, it is necessary to wait at least 20 µs with ADC_0_ADCLK running before starting A/D conversion. When entering power-down mode, the SYS_ADC_CFG.SY_MSTP_ADA bit must be set to 1b before stopping ADC_0_ADCLK.

a) Procedure for leaving power-down mode

1. Set the SYS_ADC_CFG.SY_MSTP_ADA bit to 0b.
2. Wait at least 20 µs with ADC_0_ADCLK running (CPG_CLKON_16.CLK8_ON = 1b) before starting A/D conversion.

b) Procedure for entering power-down mode

1. If A/D conversion is in operation, stop A/D conversion according to **7.10.4.2.1 A/D conversion stop procedure**.
2. Set the SYS_ADC_CFG.SY_MSTP_ADA bit to 1b.
3. Stop ADC_0_ADCLK.

7.10.4.7 Notes on Entering Low Power Consumption States

When transitioning to module standby mode or power-down mode, stop A/D conversion. When stopping the A/D conversion, it is necessary to secure the time until the analog part of the 12-bit A/D converter stops after setting the ADCSR.ADST bit to 0b. For details, see **7.10.4.2.1 A/D conversion stop procedure**.

SECTION 7 LOW-SPEED INTERFACE

7.11 Temperature Sensor Unit (TSU)

This section describes the functions of the temperature sensor unit (TSU).

7.11.1 Functional Overview

The temperature sensor unit (TSU) includes a temperature sensor IP to acquire a code corresponding to the temperature. The temperature sensor IP is composed of a main sensor and an ADC that converts analog outputs of the sensor to digital code.

The acquired temperature code is readable via the APB. Additionally, an interrupt can be output in response to the result of comparison with set upper and lower limits of the temperature code.

There are two ways to start conversion: by software through a register setting and by an ELC trigger. The conversion mode is single scan mode in which conversion is only performed once with averaging of the set number of samples.

Table 7.11-1 TSU Functions in Outline

Classification	Function	Description
Temperature sensor unit	Temperature code read	• Temperature sensor IP control function • Temperature code averaging function • Comparison with the upper and lower limit settings • Interrupt function • Conversion start trigger (by software and by an ELC trigger) • Single scan
AMBA interface	APB slave interface	• Register access

7.11.2 Connection Configuration

Figure 7.11-1 shows the connection configuration of the TSU.

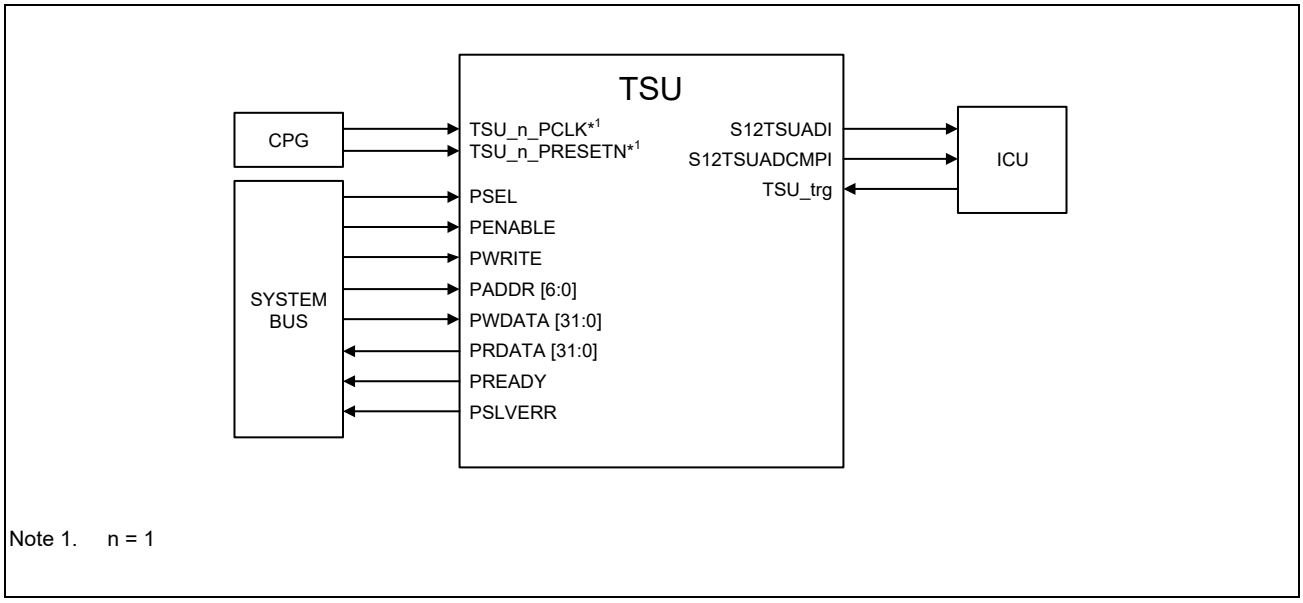


Figure 7.11-1 TSU Connection Configuration Diagram

Table 7.11-2 lists the connected units.

Table 7.11-2 List of Connected Units

Unit Name	Connected Unit Name	Function of Connected Unit
TSU	CPG	Clock pulse generator
	SYSTEM BUS	System bus
	ICU	Interrupt control unit

7.11.3 Pin Description

7.11.3.1 List of Internal Pins

Table 7.11-3 lists the internal pins of the TSU.

Table 7.11-3 List of Internal I/O Pins

Pin Name	Input/ Output	Description				
Clock & Reset						
TSU_1_PCLK	Input	Ch1 APB port clock				
TSU_1_PRESETN	Input	Ch1 APB port reset				
IP Control Interface						
TSU_trg	Input	Conversion start trigger by the ELC (via the ICU) (pulse)				
S12TSUADI	Output	Conversion complete interrupt signal (pulse)				
S12TSUADCMPI	Output	Comparison result interrupt signal (level)				

7.11.3.2 Internal Pin Functions

7.11.3.2.1 Clock and Reset

Clock and reset inputs of the TSU:

■ **TSU_n_PCLK (n = 1)**

A clock input of the TSU as a whole. The frequency is fixed to 24 MHz.

■ **TSU_n_PRESETN (n = 1)**

A reset input of the TSU as a whole. This signal resets the TSU when low and releases the reset when high.

7.11.3.2.2 Control Interface

The signals to control the TSU:

■ **TSU_trg**

This conversion start trigger signal is input to the TSU from the ELC unit via the ICU. This signal is active low and asynchronous relative to TSU_n_PCLK (n = 1), and is effective when a low pulse with a length of eight cycles at 100 MHz is input. After synchronization within the TSU, conversion starts in response to the detection of a falling edge.

■ **S12TSUADI**

This active-low interrupt output (pulse) signal is generated after the completion of conversion.
It can be masked by a register setting.

■ **S12TSUADCMPI**

This active-low interrupt output (level) signal is generated when the conditions are met in temperature code comparison.
It can be masked by a register setting.

7.11.4 Clock

7.11.4.1 Clock Signals

Table 7.11-4 Clock Pins

Pin Name	Input/ Output	Frequency (MHz)	Duty Cycle (%)	Function	Value after Reset	Handling of Pins when Not in Use
TSU_1_PCLK	Input	24	—	APB port clock for ch1	—	Low

CAUTION

Fix the TSU_n_PCLK (n = 1) frequency of this unit to 24 MHz. When it is lower than 24 MHz, the temperature sensor IP cannot operate.

7.11.5 Interrupts

The TSU has two interrupt signals, a conversion complete interrupt (S12TSUADI) and a comparison result interrupt (S12TSUADCMPI).

An interrupt enable/disable switching bit, status confirmation bit, and status clearing bit are in place for each of these signals.

7.11.5.1 Conversion Complete Interrupt

A conversion complete interrupt is generated on completion of conversion.

- When the interrupt is enabled (the ADIE bit of the sensor interrupt enable register = 1)
  - The S12TSUADI signal is low during the period of one TSU_n_PCLK (n = 1) clock cycle and the conversion complete interrupt is generated on completion of conversion by the TSU.
  - The ADF bit of the sensor interrupt status register is set to 1 and the conversion complete status flag is set on completion of conversion by the TSU.
  - Writing 1 to the ADCLR bit of the sensor interrupt clear register sets the ADF bit of the sensor interrupt status register to 0 to clear the status flag (the ADF bit retains the value 1 unless 1 is written to the ADCLR bit).
- When the interrupt is disabled (the ADIE bit of the sensor interrupt enable register = 0)
  - The S12TSUADI signal is not driven to the low level even when conversion by the TSU is completed, and the conversion complete interrupt is not generated.
  - The ADF bit of the sensor interrupt status register is set to 1 on completion of conversion by the TSU and the conversion complete status flag is set, but the interrupt is masked.
  - Writing 1 to the ADCLR bit of the sensor interrupt clear register sets the ADF bit of the sensor interrupt status register to 0 to clear the status flag (the ADF bit retains the value 1 unless 1 is written to the ADCLR bit).

Remark The conversion complete status flag (the ADF bit) is always set to 1 on completion of conversion regardless of the value of the interrupt enable bit ADIE. After the completion of conversion, write 1 to the ADCLR bit to clear the status flag.

7.11.5.2 Comparison Result Interrupt

A comparison result interrupt is generated according to the selected condition for comparison when the result of comparison meets the condition.

- When the interrupt is enabled (the CMPIE bit of the sensor interrupt enable register = 1)
  - The S12TSUADCMPI signal is driven to the low level when the result of comparison meets the condition (**Figure 7.11-2**), and the comparison result interrupt is generated.
  - At that time, the CMPF bit of the sensor interrupt status register is set to 1 and the comparison result interrupt status flag is set.
  - Writing 1 to the CMPCLR bit of the sensor interrupt clear register sets the CMPF bit of the sensor interrupt status register to 0 to clear the status flag (the CMPF bit retains the value 1 unless 1 is written to the CMPCLR bit).

- When the interrupt is disabled (when the CMPIE bit of the sensor interrupt enable register = 0)
  - The S12TSUADCMPI signal is not driven to the low level and the comparison result interrupt is not generated.
  - The CMPF bit of the sensor interrupt status register is set to 1 when the result of comparison meets the condition (**Figure 7.11-2**) and the comparison result interrupt status flag is set, but the interrupt is masked.
  - Writing 1 to the CMPCLR bit of the sensor interrupt clear register sets the CMPF bit of the sensor interrupt status register to 0 to clear the status flag (the CMPF bit retains the value 1 unless 1 is written to the CMPCLR bit).

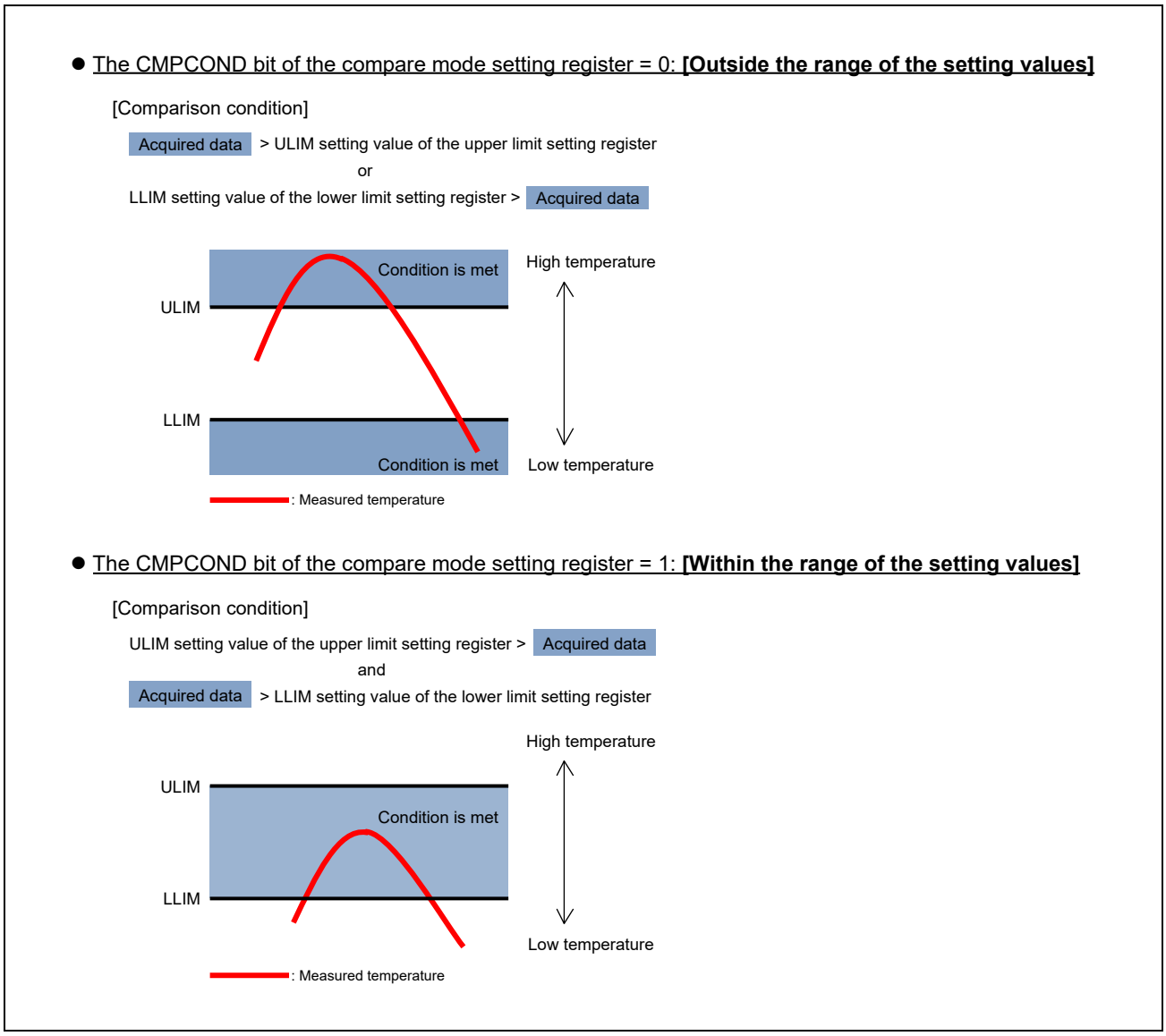


Figure 7.11-2 Comparison Conditions

7.11.6 Registers

The register addresses of TSU are given as offsets from the individual base addresses <TSUm_base>. The register base addresses of each TSU are listed in the following table.

Table 7.11-5 Register Base Addresses

Base Address Name	Unit Name	Base Address
<TSU1_base>	TSU1	0_1400_2000h (5400_2000h*1, 4400_2000h*2)

Note 1. CM33 Address Space Non-Secure

Note 1. CM33 Address Space Secure

7.11.6.1 List of Registers

The registers are listed below. [m = 1]

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Sensor Suspend Register	TSUm_SSUSR	0000_0006h	0000h	32
Sensor Trigger Register	TSUm_STRGR	0000_0000h	0004h	32
Sensor Operation Setting 1 Register	TSUm_SOSR1	0000_000Bh	0008h	32
Reserve	-	-	000Ch to 000Fh	-
Sensor Code Read Register	TSUm_SCRR	0000_0000h	0010h	32
Sensor Status Register	TSUm_SSR	0000_0000h	0014h	32
Compare Mode Setting Register	TSUm_CMSR	0000_0003h	0018h	32
Lower Limit Setting Register	TSUm_LLSR	0000_0000h	001Ch	32
Upper Limit Setting Register	TSUm_ULSR	0000_0000h	0020h	32
Reserve	-	-	0024h to 002Fh	-
Sensor Interrupt Status Register	TSUm_SISR	0000_0000h	0030h	32
Sensor Interrupt Enable Register	TSUm_SIER	0000_0000h	0034h	32
Sensor Interrupt Clear Register	TSUm_SICR	0000_0000h	0038h	32

7.11.6.2 Register Descriptions

The function description of each register is described below.

7.11.6.2.1 Sensor Suspend Register (TSUm_SSUSR)

This register controls settings related to suspension of the temperature sensor IP.

Access Size : 32 bits
Address : <TSUm_base> + 0000h
Initial Value : 0000_0006h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SOC_TS_EN	ADC_PD_TS	EN_TS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	SOC_TS_EN	1h	RW	SOC_TS Enable 0b: SOC_TS disabled (SOC_TS fixed low) 1b: SOC_TS enabled
1	ADC_PD_TS	1h	RW	ADC Power Down 0b: Normal ADC operation 1b: ADC power down
0	EN_TS	0h	RW	Temperature Sensor Enable 0b: Temperature sensor power down 1b: Normal temperature sensor operation

7.11.6.2.2 Sensor Trigger Register (TSUm_STRGR)

This register controls the start trigger for the ADC in the temperature sensor IP.

Access Size : 32 bits
Address : <TSUm_base> + 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ADST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W1	W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read.
1	-	0h	W1	Reserved This bit is read as 0b. The write value should always be 0b.
0	ADST	0h	W1	TSU ADC Start Trigger 0b: Invalid 1b: Conversion starts

7.11.6.2.3 Sensor Operation Setting 1 Register (TSUm_SOSR1)

This register controls settings related to operation of the ADC.

Access Size : 32 bits
Address : <TSUm_base> + 0008h
Initial Value : 0000_000Bh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	OUTSEL	-	-	-	-	ADCS	TRGE	-	ADCT[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
R/W	R	R	R	R	R	R	RW	R	R	R	R	RW	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	OUTSEL	0h	RW	Select data read from Sensor Code Read register 0b: Latest read data 1b: Average data
8 to 5	-	All 0	R	Reserved Whenever it is read, an undefined value is read. The written value will be ignored.
4	ADCS	0h	RW	ADC Scan Mode Select 0b: Single scan 1b: Setting prohibited
3	TRGE	1h	RW	TSU_trg Enable 0b: TSU_trg pin disabled 1b: TSU_trg pin enabled
2	-	0h	R	Reserved Whenever it is read, an undefined value is read. The written value will be ignored.
1, 0	ADCT[1:0]	3h	RW	AD Count Setting 00b: 1 01b: 2 10b: 4 11b: 8

7.11.6.2.4 Sensor Code Read Register (TSUm_SCRR)

This register is a read-only register to read acquired code data. Any written values are ignored.

Access Size : 32 bits
Address : <TSUm_base> + 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	OUT12BIT_TS[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	OUT12BIT_TS [11:0]	0h	R	Temperature sensor code data

7.11.6.2.5 Sensor Status Register (TSUm_SSR)

This register is a read-only register to read the TSU operating status. Any written values are ignored.

Access Size : 32 bits
Address : <TSUm_base> + 0014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CONV
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	CONV	0h	R	TSU operating status 0b: Conversion is finished. 1b: Conversion is in progress. Set: Setting of the ADST bit of the Sensor Trigger Register or assertion of the signal on the TSU_trg pin. Reset: Completion of AD conversion the number of times required for averaging.

7.11.6.2.6 Compare Mode Setting Register (TSUm_CMSR)

This register controls settings related to compare operation.

Access Size : 32 bits
Address : <TSUm_base> + 0018h
Initial Value : 0000_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CMPCOND	CMPEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	CMPCOND	1h	RW	Compare Mode Setting 0b: CMP judges that lower limit (LLILM) > code data or code data > upper limit (ULIM). 1b: CMP judges that lower limit (LLILM) < code data < upper limit (ULIM).
0	CMPEN	1h	RW	Compare Enable 0b: Comparison disabled 1b: Comparison enabled

7.11.6.2.7 Lower Limit Setting Register (TSUm_LLSR)

This register controls the lower limit value to be compared with the acquired code data.

Access Size : 32 bits
Address : <TSUm_base> + 001Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	LLIM[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	LLIM[11:0]	0h	RW	Lower limit setting value

7.11.6.2.8 Upper Limit Setting Register (TSUm_ULSR)

This register controls the upper limit value to be compared with the acquired code data.

Access Size : 32 bits
Address : <TSUm_base> + 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	ULIM[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	ULIM[11:0]	0h	RW	Upper limit setting value

7.11.6.2.9 Sensor Interrupt Status Register (TSUm_SISR)

This register a read-only register which monitors the interrupt status.

Access Size : 32 bits
Address : <TSUm_base> + 0030h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CMPF	ADF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	CMPF	0h	R	Compare result detect flag 0b: Undetected 1b: Detected Set condition: The comparison conditions are met. Reset condition: The CMPCLR bit of the sensor interrupt clear register is set to 1b.
0	ADF	0h	R	AD conversion complete flag 0b: Not completed 1b: Completed Set condition: Completion of AD conversion the number of times required for averaging. Reset condition: The ADCLR bit of the sensor interrupt clear register is set to 1b.

7.11.6.2.10 Sensor Interrupt Enable Register (TSUm_SIER)

This register enables or disables output of the interrupt request output signals (S12TSUADI and S12TSUADCMPI).

Access Size : 32 bits
Address : <TSUm_base> + 0034h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CMPIE	ADIE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
1	CMPIE	0h	RW	Compare Result Detect Interrupt Enable 0b: Interrupt disabled 1b: Interrupt enabled
0	ADIE	0h	RW	AD Conversion Complete Interrupt Enable 0b: Interrupt disabled 1b: Interrupt enabled

7.11.6.2.11 Sensor Interrupt Clear Register (TSUm_SICR)

This register clears the interrupt status.

Access Size : 32 bits
Address : <TSUm_base> + 0038h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CMPCLR	ADCLR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W1	W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, an undefined value is read. The written value will be ignored.
1	CMPCLR	0h	W1	The read value is undefined Compare Result Detect Interrupt Clear Write only. This bit is read as 0b. 0b: No operation 1b: Interrupt cleared
0	ADCLR	0h	W1	The read value is undefined AD Conversion Complete Interrupt Clear Write only. This bit is read as 0b. 0b: No operation 1b: Interrupt cleared

7.11.7 Functional Description

This section describes the functions of the TSU.

The prefix (TSUm_) (m = 1) of the register names is omitted in this and subsequent sections.

7.11.7.1 Acquiring Temperature Code

Conversion starts on reception of the conversion start trigger (when the ADST bit of the sensor trigger register is set to 1b or the TSU_trg signal is 0). When the OUT12BIT_TS bit of the sensor code read register is read after the completion of conversion, a 12-bit temperature code can be acquired.

If the conversion start trigger comes within 120T (CLK_TS period) after release of the temperature sensor and ADC from the power-down state (setting the sensor suspend register so that the ADC_PD_TS bit is 0b and the EN_TS bit is 1b), conversion only starts after 120T has elapsed following release.

7.11.7.2 Temperature Code Averaging Function

The temperature codes acquired within the TSU are averaged the number of times set by the ADCT[1:0] bits of the sensor operation setting 1 register.

7.11.7.3 Temperature Code Reading Function

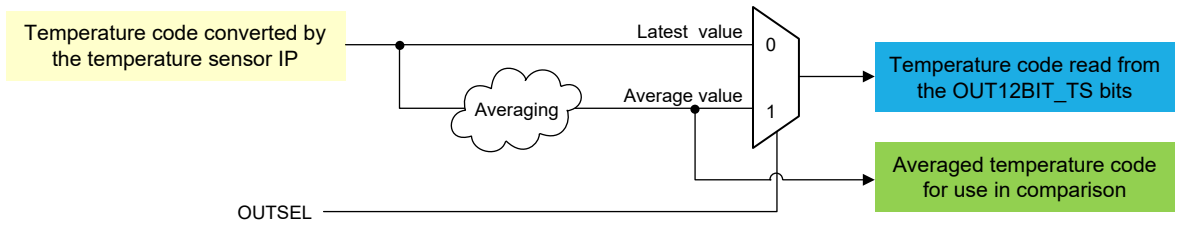
The temperature code can be read from the sensor code read register via the APB.

When the OUTSEL bit of the sensor operation setting 1 register is 0b, the latest converted value is read. When it is 1b, the average value is read.

7.11.7.4 Temperature Code Comparison Functions

This function compares the averaged temperature code and the values set by the LLIM[11:0] bits of the lower limit setting register and the ULIM[11:0] bits of the upper limit setting register when the CMPEN bit of the compare mode setting register is 1b.

The temperature code for use in the comparison function is the averaged temperature code, so the OUTSEL bit of the sensor operation setting 1 register must be set to 1b (average data output) when reading the temperature code for use in comparison. When the averaging count setting is 1, however, the latest value is the same as the average value, so the temperature code for use in comparison will be read regardless of the setting of the OUTSEL bit (see **Figure 7.11-3**).



- When the OUTSEL bit = 0b

  After conversion has proceeded the set number of times for averaging, the latest value is read from the OUT12BIT_TS bits.

  [Example] When ADCT[1:0] =11b (the count setting is 8), the latest value is the 8th converted temperature code.

- When the OUTSEL bit = 1b

  After conversion has proceeded the set number of times for averaging, the average value is read from the OUT12BIT_TS bits.
  The average value is also used for the comparison function.

  [Example] When ADCT[1:0] = 11b (the count setting is 8), the average value is the average of 8 sampled temperature codes.

Averaging: Acquiring temperature codes the number of times set by the ADCT[1:0] bits of the sensor operation setting 1 register and averaging the codes.

[Example] When ADCT[1:0] = 11b (the count setting is 8), 8 temperature codes are acquired and averaged.

Note: When the averaging count setting is 1, only one temperature code is acquired, so the latest value is the same as the average value.

Figure 7.11-3 OUTSEL Bit Setting when Temperature Code Comparison is Used

The CMPF bit of the sensor interrupt status register is set to 1b when the value meets the selected condition for comparison as shown in **Figure 7.11-2**.

The values set in the LLIM[11:0] and ULIM[11:0] bits can be calculated from the formula shown in **7.11.7.9 Comparison Setting Value Calculation**.

7.11.7.5 Interrupt Output Function

An interrupt is output when conversion is completed and the result of temperature code comparison meets the condition.

For details of the interrupt output function, see **7.11.5 Interrupts**.

7.11.7.6 Conversion Start Trigger

There are two ways to start conversion by the TSU: by software through a register setting and by an ELC trigger (TSU_trg).

The TRGE bit of the sensor operation setting 1 register is used to set the ELC trigger.

(TRGE bit = 0b: TSU_trg disabled; TRGE bit = 1b: TSU_trg enabled)

7.11.7.7 Single Scan

In single scan mode, conversion is only performed once with averaging of the number of samples set by the ADCT[1:0] bits of the sensor operation setting 1 register.

Single scan mode is set by default in the ADCS bit of the sensor operation setting 1 register.

(ADCS bit = 0b: Single scan mode; ADCS bit = 1b: Setting prohibited)

7.11.7.8 Temperature Compensation Calculation

After acquiring the temperature, the compensated temperature T (°C) must be calculated from the value (a) of the OUT12BIT_TS bits of the sensor code read register and the values (b) to (c) read from the SYS register.

Follow the procedure below to calculate the compensated temperature. (Maximum error is ±5°C)

$$\mathrm{T(°C)} = \left(\frac{e-d}{c-b}\right) \times (a-b) + d$$

Remarks: *b* = SYS: OTPTSUnTRMVAL0[11:0]
c = SYS: OTPTSUnTRMVAL1[11:0]
d = −41
e = 126

7.11.7.9 Comparison Setting Value Calculation

The comparison setting values (a) can be calculated from each temperature T (°C) you want to set and the values from (b) to (c) which are read from the SYS register.

$$\mathrm{a} = \frac{(T-d) \times (c-b)}{(e-d)} + b$$

Remarks: *b* = SYS: OTPTSUnTRMVAL0[11:0]
c = SYS: OTPTSUnTRMVAL1[11:0]
d = −41
e = 126

Set the calculated values (a) in the LLIM[11:0] bits of the low limit setting register and the ULIM[11:0] bits of the upper limit setting register.

Calculation of the comparison setting values is a reverse calculation of the (**7.11.7.8 Temperature Compensation Calculation**).

- Temperature compensation calculation: Calculate the temperature T (°C) from the acquired temperature code (a).
- Comparison setting value calculation: Calculate the temperature code (a) from the temperature T (°C) you want to set.

7.11.8 Operation Procedure

The TSU is mainly used in the following two ways. This section describes the procedures of operations for usage in each of these ways.

- Using an interrupt for each completion of conversion (**7.11.8.1 Using Interrupt for Each Completion of Conversion**)

  Conversion is started by software or an ELC trigger and the interrupt is generated on completion of conversion.

  After generation of the interrupt, the temperature code is read.

  The comparison function is enabled by default. The upper and lower limit settings for use in comparison are both all 0s by default and the conditions for comparison are not met even while the comparison function is enabled, but the conditions may inadvertently be met if operation proceeds after the upper and lower limits have been set.

  When the comparison function is not to be used, set the CMPEN bit of the compare mode setting register to 0b to disable the comparison function.

- Using the comparison result interrupt (**7.11.8.2 Using Comparison Result Interrupt**)

  Conversion is started by software or an ELC trigger after the conditions for comparison (within or outside the range, and the upper and lower limits) have been set and the interrupt is generated when the result of comparison with the settings meets the conditions.

  After generation of the interrupt, the temperature code is read.

7.11.8.1 Using Interrupt for Each Completion of Conversion

7.11.8.1.1 Single Scan, Start by Software

Table 7.11-6 shows an example of the procedure for operations when an interrupt is to be used to indicate each completion of conversion (single scan, start by software). This is an example of reading the average value when the number of samples for averaging is 8.

Table 7.11-6 Operation Procedure Example when Interrupt is to be Used for Each Completion of Conversion (Single Scan, Start by Software) (1/2)

Step		Register Name	Bit Name	Processing Details	Outline
(1)	Supply TSU_n_PCLK (n = 1).	CPG:CPG_CLKON_16	CLK10_ONWEN, CLK10_ON(TSU1)	Write “1b” (_ONWEN is write enable)	Supply TSU_1_PCLK to TSU1.
(2)	Check TSU_n_PCLK (n = 1) supply.	CPG:CPG_CLKMON_8	CLK10_MON(TSU1)	Read “1b”	Check that TSU_1_PCLK was supplied to TSU1.
(3)	De-assert TSU_n_PRESETN (n = 1).	CPG:CPG_RST_15	RSTB8_WEN, RSTB8(TSU1)	Write “1b” (_WEN is write enable)	De-assert TSU_1_PRESETN of TSU1.
(4)	Check TSU_n_PRESETN (n = 1) de-assertion.	CPG:CPG_RSTMON_7	RST9_MON(TSU1)	Read “0b”	Check that TSU_1_PRESETN of TSU1 was de-asserted.
(5)	Release the temperature sensor IP from the power-down state.	Sensor Suspend Register	SOC_TS_EN	Write “1b”	Enable the SOC_TS signal.
			ADC_PD_TS	Write “0b”	Normal ADC operation
			EN_TS	Write “1b”	Normal temperature sensor operation

Table 7.11-6 Operation Procedure Example when Interrupt is to be Used for Each Completion of Conversion (Single Scan, Start by Software) (2/2)

Step		Register Name	Bit Name	Processing Details	Outline
(6)	Set the operating mode.	Sensor Operation Setting 1 Register	OUTSEL	Write “1b”	Average data output
			ADCS	Write “0b”	Single scan mode
			TRGE	Write “0b”	Disable the ELC trigger.
			ADCT[1:0]	Write “11b”	Set the number of samples for averaging to 8.
		Compare Mode Setting Register	CMPCOND	Write “0b”	The setting remains default.
			CMPEN	Write “0b”	Disable the comparison function.
		Sensor Interrupt Enable Register	CMPIE	Write “0b”	Mask the comparison result interrupt.
			ADIE	Write “1b”	Enable the conversion complete interrupt.
(7)	Check the conversion status.	Sensor Status Register	CONV	Read “0b”	Read the CONV bit as 0b (conversion is not in progress).
(8)	Start conversion.	Sensor Trigger Register	ADST	Write “1b”	Start conversion by software.
(9)	The conversion complete interrupt is generated. → Check the interrupt status.	Sensor Interrupt Status Register	ADF	Read “1b”	Read the conversion complete status flag.
(10)	Check the conversion status.	Sensor Status Register	CONV	Read “0b”	Read the CONV bit as 0b (conversion is not in progress).
(11)	Read the converted code.	Sensor Code Read Register	OUT12BIT_TS[11:0]	12-bit read	Read the 12-bit converted temperature code.
(12)	Calculate the temperature compensation.	See **7.11.7.8 Temperature Compensation Calculation**.			
(13)	Clear the interrupt status.	Sensor Interrupt Clear Register	ADCLR	Write “1b”	Clear the conversion complete status flag.
(14)	Check the interrupt status.	Sensor Interrupt Status Register	ADF	Read “0b”	Check that the conversion complete status flag was cleared.
(15)	Power down the temperature sensor IP.	Sensor Suspend Register	SOC_TS_EN	Write “0b”	Disable the SOC_TS signal.
			ADC_PD_TS	Write “1b”	Power down the ADC.
			EN_TS	Write “0b”	Power down the temperature sensor.
(16)	Assert TSU_n_PRESETN (n = 1).	CPG:CPG_RST_15	RSTB8_WEN, RSTB8(TSU1)	Write “0b” (_WEN is write enable)	Assert TSU_1_PRESETN of TSU1.
(17)	Check TSU_n_PRESETN (n = 1) assertion.	CPG:CPG_RSTMON_7	RST9_MON(TSU1)	Read “1b”	Check that TSU_1_PRESETN of TSU1 was asserted.
(18)	Stop TSU_n_PCLK (n = 1).	CPG:CPG_CLKON_16	CLK10_ONWEN, CLK10_ON(TSU1)	Write “0b” (_ONWEN is write enable)	Stop TSU_1_PCLK to TSU1.
(19)	Check stopping of TSU_n_PCLK (n = 1).	CPG:CPG_CLKMON_8	CLK10_MON(TSU1)	Read “0b”	Check that TSU_1_PCLK supply to TSU1 was stopped.

7.11.8.1.2 Single Scan, Start by an ELC Trigger

Table 7.11-7 shows an example of the procedure for operations when an interrupt is to be used to indicate each completion of conversion (single scan, start by an ELC trigger). This is an example of reading the average value when the number of samples for averaging is 8.

Table 7.11-7 Operation Procedure Example when Interrupt is to be Used for Each Completion of Conversion (Single Scan, Start by ELC Trigger) (1/2)

Step		Register Name	Bit Name	Processing Details	Outline
(1)	Supply TSU_n_PCLK (n = 1).	CPG:CPG_CLKON_16	CLK10_ONWEN, CLK10_ON(TSU1)	Write “1b” (_ONWEN is write enable)	Supply TSU_1_PCLK to TSU1.
(2)	Check TSU_n_PCLK (n = 1) supply.	CPG:CPG_CLKMON_8	CLK10_MON(TSU1)	Read “1b”	Check that TSU_1_PCLK was supplied to TSU1.
(3)	De-assert TSU_n_PRESETN (n = 1).	CPG:CPG_RST_15	RSTB8_WEN, RSTB8(TSU1)	Write “1b” (_WEN is write enable)	De-assert TSU_1_PRESETN of TSU1.
(4)	Check TSU_n_PRESETN (n = 1) de-assertion.	CPG:CPG_RSTMON_7	RST9_MON(TSU1)	Read “0b”	Check that TSU_1_PRESETN of TSU1 was de-asserted.
(5)	Release the temperature sensor IP from power down.	Sensor Suspend Register	SOC_TS_EN	Write “1b”	Enable the SOC_TS signal.
			ADC_PD_TS	Write “0b”	Normal ADC operation
			EN_TS	Write “1b”	Normal temperature sensor operation
(6)	Set the operating mode.	Sensor Operation Setting 1 Register	OUTSEL	Write “1b”	Average data output
			ADCS	Write “0b”	Single scan mode
			TRGE	Write “1b”	Enable the ELC trigger.
			ADCT[1:0]	Write “11b”	Set the number of samples for averaging to 8.
		Compare Mode Setting Register	CMPCOND	Write “0b”	The setting remains default.
			CMPEN	Write “0b”	Disable the comparison function.
		Sensor Interrupt Enable Register	CMPIE	Write “0b”	Mask the comparison result interrupt.
			ADIE	Write “1b”	Enable the conversion complete interrupt.
(7)	Check the conversion status.	Sensor Status Register	CONV	Read “0b”	Read the CONV bit as 0 (conversion is not in progress).
(8)	Set the ELC trigger.	ICU: EVTSEL6(TSU1)	TSU1: Bit position [9:0]	Write the ELC no.*1 of the event to be used.	Select the ELC trigger to be used from the ICU register.
(9)	Start conversion.	—	—	—	Receive the ELC trigger from the ICU.
(10)	The conversion complete interrupt is generated. → Check the interrupt status.	Sensor Interrupt Status Register	ADF	Read “1b”	Read the conversion complete status flag.
(11)	Check the conversion status.	Sensor Status Register	CONV	Read “0b”	Read the CONV bit as 0 (conversion is not in progress).
(12)	Read the converted code.	Sensor Code Read Register	OUT12BIT_TS[11:0]	12-bit read	Read the 12-bit converted temperature code.-
(13)	Calculate the temperature compensation.	See **7.11.7.8 Temperature Compensation Calculation**.			
(14)	Clear the interrupt status.	Sensor Interrupt Clear Register	ADCLR	Write “1b”	Clear the conversion complete status flag.

Table 7.11-7 Operation Procedure Example when Interrupt is to be Used for Each Completion of Conversion (Single Scan, Start by ELC Trigger) (2/2)

Step		Register Name	Bit Name	Processing Details	Outline
(15)	Check the interrupt status.	Sensor Interrupt Status Register	ADF	Read “0b”	Check that the conversion complete status flag was cleared.
(16)	Power down the temperature sensor IP.	Sensor Suspend Register	SOC_TS_EN	Write “0b”	Disable the SOC_TS signal.
			ADC_PD_TS	Write “1b”	Power down the ADC.
			EN_TS	Write “0b”	Power down the temperature sensor.
(17)	Assert TSU_n_PRESETN (n = 1).	CPG:CPG_RST_15	RSTB8_WEN, RSTB8(TSU1)	Write “0b” (_WEN is write enable)	Assert TSU_1_PRESETN of TSU1.
(18)	Check TSU_n_PRESETN (n = 1) assertion.	CPG:CPG_RSTMON_7	RST9_MON(TSU1)	Read “1b”	Check that TSU_1_PRESETN of TSU1 was asserted.
(19)	Stop TSU_n_PCLK (n = 1).	CPG:CPG_CLKON_16	CLK10_ONWEN, CLK10_ON(TSU1)	Write “0b” (_ONWEN is write enable)	Stop TSU_1_PCLK to TSU1.
(20)	Check stopping of TSU_n_PCLK (n = 1).	CPG:CPG_CLKMON_8	CLK10_MON(TSU1)	Read “0b”	Check that TSU_1_PCLK supply to TSU1 was stopped.

Note 1. For the ELC no., refer to the ICU document “Event_list file”.

7.11.8.2 Using Comparison Result Interrupt

7.11.8.2.1 Single Scan, Start by Software

Table 7.11-8 shows an example of the procedure for operations when the comparison result interrupt is to be used (single scan, start by software). This is an example of reading the average value when the number of samples for averaging is 8.

Table 7.11-8 Operation Procedure Example when Comparison Result Interrupt is to be Used (Single Scan, Start by Software) (1/2)

Step		Register Name	Bit Name	Processing Details	Outline
(1)	Supply TSU_n_PCLK (n = 1).	CPG:CPG_CLKON_16	CLK10_ONWEN, CLK10_ON(TSU1)	Write “1b” (_ONWEN is write enable)	Supply TSU_1_PCLK to TSU1.
(2)	Check TSU_n_PCLK (n = 1) supply.	CPG:CPG_CLKMON_8	CLK10_MON(TSU1)	Read “1b”	Check that TSU_1_PCLK was supplied to TSU1.
(3)	De-assert TSU_n_PRESETN (n = 1).	CPG:CPG_RST_15	RSTB8_WEN, RSTB8(TSU1)	Write “1b” (_WEN is write enable)	De-assert TSU_1_PRESETN of TSU1.
(4)	Check TSU_n_PRESETN (n = 1) de-assertion.	CPG:CPG_RSTMON_7	RST9_MON(TSU1)	Read “0b”	Check that TSU_1_PRESETN of TSU1 was de-asserted.
(5)	Release the temperature sensor IP from the power-down state.	Sensor Suspend Register	SOC_TS_EN	Write “1b”	Enable the SOC_TS signal.
			ADC_PD_TS	Write “0b”	Normal ADC operation
			EN_TS	Write “1b”	Normal temperature sensor operation
(6)	Set the operating mode.	Sensor Operation Setting 1 Register	OUTSEL	Write “1b”	Average data output
			ADCS	Write “0b”	Single scan mode
			TRGE	Write “0b”	Disable the ELC trigger.
			ADCT[1:0]	Write “11b”	Set the number of samples for averaging to 8.
		Compare Mode Setting Register	CMPCOND	Write “1b”	Compare mode (within the range setting)
			CMPEN	Write “1b”	Enable the comparison function.
		Sensor Interrupt Enable Register	CMPIE	Write “1b”	Enable the comparison result interrupt.
			ADIE	Write “0b”	Enable the conversion complete interrupt.
(7)	Set the comparison setting values.	Lower Limit Setting Register	LLIM[11:0]	12-bit write	For how to calculate the comparison setting values, see **7.11.7.9 Comparison Setting Value Calculation**.
		Upper Limit Setting Register	ULIM[11:0]	12-bit write	For how to calculate the comparison setting values, see **7.11.7.9 Comparison Setting Value Calculation**.
(8)	Check the conversion status.	Sensor Status Register	CONV	Read “0b”	Read the CONV bit as 0b (conversion is not in progress).
(9)	Start conversion.	Sensor Trigger Register	ADST	Write “1b”	Start conversion by software.
(10)	The comparison result interrupt is generated. → Check the interrupt status.	Sensor Interrupt Status Register	CMPF, ADF	Read “11b”	Read the comparison result status and conversion complete status flags.

Table 7.11-8 Operation Procedure Example when Comparison Result Interrupt is to be Used (Single Scan, Start by Software) (2/2)

Step		Register Name	Bit Name	Processing Details	Outline
(11)	Check the conversion status.	Sensor Status Register	CONV	Read “0b”	Read the CONV bit as 0b (conversion is not in progress).
(12)	Read the converted code.	Sensor Code Read Register	OUT12BIT_TS[11:0]	12-bit read	Read the 12-bit converted temperature code.
(13)	Calculate the temperature compensation.	See **7.11.7.8 Temperature Compensation Calculation**.			
(14)	Clear the interrupt status.	Sensor Interrupt Clear Register	CMPCLR, ADCLR	Write “1b”	Clear the comparison result status and conversion complete status flags.
(15)	Check the interrupt status.	Sensor Interrupt Status Register	CMPF,ADF	Read “00b”	Check that the comparison result status and conversion complete status flags were cleared.
(16)	Power down the temperature sensor IP.	Sensor Suspend Register	SOC_TS_EN	Write “0b”	Disable the SOC_TS signal.
			ADC_PD_TS	Write “1b”	Power down the ADC.
			EN_TS	Write “0b”	Power down the temperature sensor.
(17)	Assert TSU_n_PRESETN (n = 1).	CPG:CPG_RST_15	RSTB8_WEN, RSTB8(TSU1)	Write “0b” (_WEN is write enable)	Assert TSU_1_PRESETN of TSU1.
(18)	Check TSU_n_PRESETN (n = 1) assertion.	CPG:CPG_RSTMON_7	RST9_MON(TSU1)	Read “1b”	Check that TSU_1_PRESETN of TSU1 was asserted.
(19)	Stop TSU_n_PCLK (n = 1).	CPG:CPG_CLKON_16	CLK10_ONWEN, CLK10_ON(TSU1)	Write “0b” (_ONWEN is write enable)	Stop TSU_1_PCLK to TSU1.
(20)	Check stopping of TSU_n_PCLK (n = 1).	CPG:CPG_CLKMON_8	CLK10_MON(TSU1)	Read “0b”	Check that TSU_1_PCLK supply to TSU1 was stopped.

7.11.8.2.2 Single Scan, Start by an ELC Trigger

Table 7.11-9 shows an example of the procedure for operations when the comparison result interrupt is to be used (single scan, start by an ELC trigger). This is an example of reading the average value when the number of samples for averaging is 8.

Table 7.11-9 Operation Procedure Example when Comparison Result Interrupt is to be Used (Single Scan, Start by ELC Trigger) (1/2)

Step		Register Name	Bit Name	Processing Details	Outline
(1)	Supply TSU_n_PCLK (n = 1).	CPG:CPG_CLKON_16	CLK10_ONWEN, CLK10_ON(TSU1)	Write “1b” (_ONWEN is write enable)	Supply TSU_1_PCLK to TSU1.
(2)	Check TSU_n_PCLK (n = 1) supply.	CPG:CPG_CLKMON_8	CLK10_MON(TSU1)	Read “1b”	Check that TSU_1_PCLK was supplied to SU1.
(3)	De-assert TSU_n_PRESETN (n = 1).	CPG:CPG_RST_15	RSTB8_WEN, RSTB8(TSU1)	Write “1b” (_WEN is write enable)	De-assert TSU_1_PRESETN of TSU1.
(4)	Check TSU_n_PRESETN (n = 1) de-assertion.	CPG:CPG_RSTMON_7	RST9_MON(TSU1)	Read “0b”	Check that TSU_1_PRESETN of TSU1 was de-asserted.
(5)	Release the temperature sensor IP from the power-down state.	Sensor Suspend Register	SOC_TS_EN	Write “1b”	Enable the SOC_TS signal.
			ADC_PD_TS	Write “0b”	Normal ADC operation
			EN_TS	Write “1b”	Normal temperature sensor operation
(6)	Set the operating mode.	Sensor Operation Setting 1 Register	OUTSEL	Write “1b”	Average data output
			ADCS	Write “0b”	Single scan mode
			TRGE	Write “1b”	Enable the ELC trigger.
			ADCT[1:0]	Write “11b”	Set the number of samples for averaging to 8.
		Compare Mode Setting Register	CMPCOND	Write “1b”	Compare mode (within the range setting)
			CMPEN	Write “1b”	Enable the comparison function.
		Sensor Interrupt Enable Register	CMPIE	Write “1b”	Enable the comparison result interrupt.
			ADIE	Write “0b”	Enable the conversion complete interrupt.
(7)	Set the comparison setting values.	Lower Limit Setting Register	LLIM[11:0]	12-bit write	For how to calculate the comparison setting values, see **7.11.7.9 Comparison Setting Value Calculation**.
		Upper Limit Setting Register	ULIM[11:0]	12-bit write	For how to calculate the comparison setting values, see **7.11.7.9 Comparison Setting Value Calculation**.
(8)	Check the conversion status.	Sensor Status Register	CONV	Read “0b”	Read the CONV bit as 0 (conversion is not in progress).
(9)	Set the ELC trigger.	ICU: EVTSEL6(TSU1)	TSU1: Bit position [9:0]	Write the ELC no.*1 of the event to be used.	Select the ELC trigger to be used from the ICU register.
(10)	Start conversion.	—	—	—	Receive the ELC trigger from the ICU.
(11)	The comparison result interrupt is generated. → Check the interrupt status.	Sensor Interrupt Status Register	CMPF, ADF	Read “11b”	Read the comparison result status and conversion complete status flags.

Table 7.11-9 Operation Procedure Example when Comparison Result Interrupt is to be Used (Single Scan, Start by ELC Trigger) (2/2)

Step		Register Name	Bit Name	Processing Details	Outline
(12)	Check the conversion status.	Sensor Status Register	CONV	Read “0b”	Read the CONV bit as 0b (conversion is not in progress).
(13)	Read the converted code.	Sensor Code Read Register	OUT12BIT_TS[11:0]	12-bit read	Read the 12-bit converted temperature code.
(14)	Calculate the temperature compensation.	See **7.11.7.8 Temperature Compensation Calculation**.			
(15)	Clear the interrupt status.	Sensor Interrupt Clear Register	CMPCLR, ADCLR	Write “1b”	Clear the comparison result status and conversion complete status flags.
(16)	Check the interrupt status.	Sensor Interrupt Status Register	CMPF, ADF	Read “00b”	Check the comparison result status and conversion complete status flags were cleared.
(17)	Power down the temperature sensor IP.	Sensor Suspend Register	SOC_TS_EN	Write “0b”	Disable the SOC_TS signal.
			ADC_PD_TS	Write “1b”	Power down the ADC.
			EN_TS	Write “0b”	Power down the temperature sensor.
(18)	Assert TSU_n_PRESETN (n = 1).	CPG:CPG_RST_15	RSTB8_WEN, RSTB8(TSU1)	Write “0b” (_WEN is write enable)	Assert TSU_1_PRESETN of TSU1.
(19)	Check TSU_n_PRESETN (n = 1) assertion.	CPG:CPG_RSTMON_7	RST9_MON(TSU1)	Read “1b”	Check that TSU_1_PRESETN of TSU1 was asserted.
(20)	Stop TSU_n_PCLK (n = 1).	CPG:CPG_CLKON_16	CLK10_ONWEN, CLK10_ON(TSU1)	Write “0b” (_ONWEN is write enable)	Stop TSU_1_PCLK to TSU1.
(21)	Check stopping of TSU_n_PCLK (n = 1).	CPG:CPG_CLKMON_8	CLK10_MON(TSU1)	Read “0b”	Check that TSU_1_PCLK supply to TSU1 was stopped.

Note 1. For the ELC no., refer to the ICU document “Event_list file”.

7.11.8.3 Initialization Procedure

Returning the TSU to its initial state requires asserting the reset signal from the CPG register and then de-asserting the signal.

Follow the procedure in **Table 7.11-10** to perform initialization.

Table 7.11-10 TSU Initialization Procedure

Step		Register Name	Bit Name	Processing Details	Outline
(1)	Assert TSU_n_PRESETN (n = 1).	CPG:CPG_RST_15	RSTB8_WEN, RSTB8(TSU1)	Write “0b” (_WEN is write enable)	Assert TSU_1_PRESETN of TSU1.
(2)	Check TSU_n_PRESETN (n = 1) assertion.	CPG:CPG_RSTMON_7	RST9_MON(TSU1)	Read “1b”	Check that TSU_1_PRESETN of TSU1 was asserted.
(3)	De-assert TSU_n_PRESETN (n = 1).	CPG:CPG_RST_15	RSTB8_WEN, RSTB8(TSU1)	Write “1b” (_WEN is write enable)	De-assert TSU_1_PRESETN of TSU1.
(4)	Check TSU_n_PRESETN (n = 1) de-assertion.	CPG:CPG_RSTMON_7	RST9_MON(TSU1)	Read “0b”	Check that TSU_1_PRESETN of TSU1 was de-asserted.

7.11.8.4 Procedure for Changing Operating Mode

The operating mode can be changed while conversion is stopped. **Table 7.11-11** shows the procedure for changing the operating mode after checking the CONV bit of the sensor status register = 0.

Table 7.11-11 Procedure for Changing Operating Mode

Step		Register Name	Bit Name	Processing Details	Outline
(1)	Check the conversion status.	Sensor Status Register	CONV	Read “0b”	Read the CONV bit as 0b (conversion is not in progress).
(2)	Set the operating mode.	Set the operating mode to change.			

7.11.9 Usage Notes

7.11.9.1 Release from Power-down

Set the SOC_TS_EN bit of the Sensor Suspend Register to 1b at the same time or before setting the EN_TS bit of the same register to 1b and the ADC_PD_TS bit to 0b to release the temperature sensor IP from the power-down state.

7.11.9.2 Conversion Start Trigger

Before making the setting to start conversion by software or an ELC trigger, check that the CONV bit of the sensor status register is 0b.

7.11.9.3 Restarting the Stopped Clock

Restarting the clock after having stopped it requires placing the module in the power-down state once and then releasing it from the power-down state.

- Power down: Sensor Suspend Register

  SOC_TS_EN bit = 0b
  ADC_PD_TS bit = 1b
  EN_TS bit = 0b

- Release from power-down: Sensor Suspend Register

  SOC_TS_EN bit = 1b
  ADC_PD_TS bit = 0b
  EN_TS bit = 1b

SECTION 8 AUDIO

8.1 Audio Overview

8.1.1 Overview

This section describes the data path between audio modules, list of routings, and transfer flow.

8.1.1.1 Features

The audio module is a module connecting ADMAC, SCU, SSIU, SPDIF, PDM, and ADG.

Refer to the section of each module for the detailed function.

8.1.1.2 Block Diagram

Figure 8.1-1 is a block diagram of the audio module. **Figure 8.1-2** shows the data paths between SSIU and SCU and ADMAC.

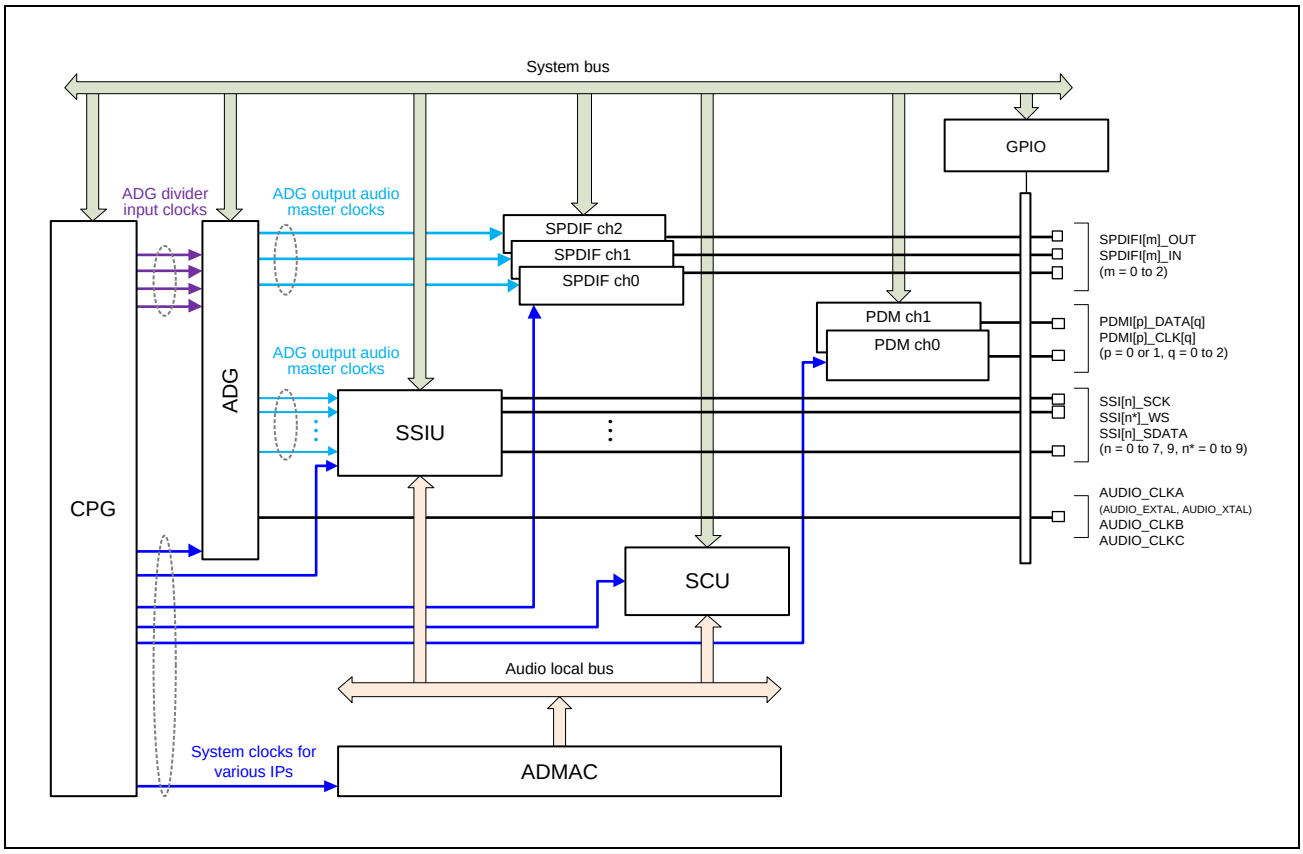


Figure 8.1-1 Audio Module Block Diagram

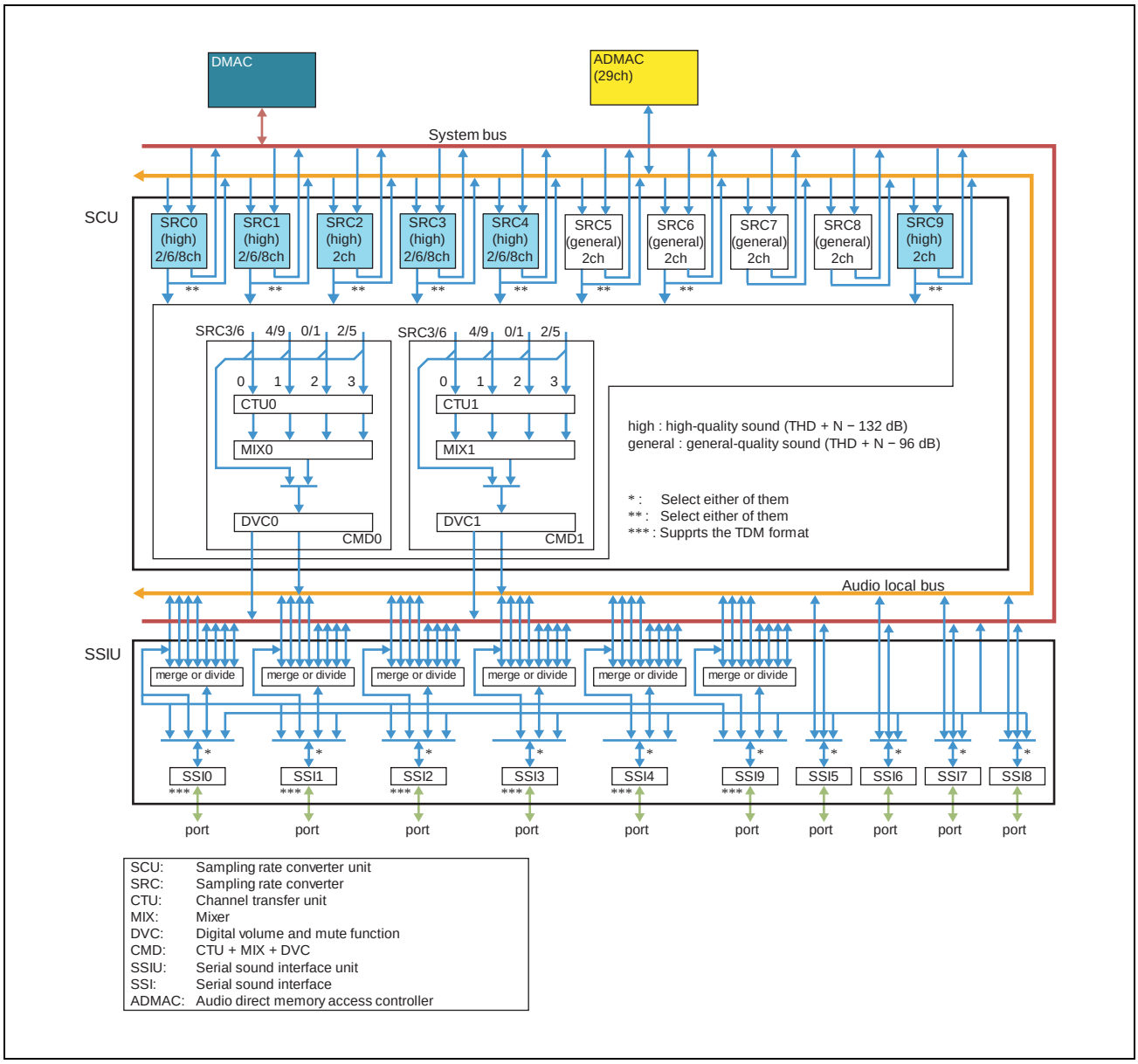


Figure 8.1-2 Data Paths between Audio Modules

8.1.1.3 External Pins

Refer to the following sections for more information about terminal information.

- Section ADG
- Section SSIU
- Section SPDIF
- Section PDM

8.1.1.4 Register Configuration

The audio module does not have a register.

8.1.1.5 Connected Module

Table 8.1-1 Connected Modules

Module Name		Connected Module Name	Function of Connected Module
Audio	Related	SYSTEMBUS	System bus (access the register)
		GPIO	General-purpose I/O, pin function controller
		CPG	Clock pulse generator (Clock ON/OFF, software reset ON/OFF)
		ICU	Interrupt control unit
	Including	SSIU	Serial sound interface unit
		SSI	Serial sound interface
		ADG	Audio clock generator (Output clocks for audio module)
		SCU	Sampling rate converter unit
		SPDIF	SPDIF interface
		PDM	Pulse density modulation
		DMAC	Direct memory access controller
		ADMAC	Audio-DMAC-Peripheral-Peripheral (for audio modules connected to the audio local bus)

8.1.2 Register Description

The audio module does not have a register.

8.1.3 Operation

8.1.3.1 Audio Module Description

1. Audio-DMAC-Peripheral-Peripheral (ADMAC)

The ADMAC controls data transfer between the audio modules (SSIU, SCU) connected to the audio local bus.

- Number of channels: 29 channels
- Data transfer size: Longword (4 bytes)
- Addressing mode: Dual addressing; fixed access size
- Transfer count: Not programmable
- Interrupt processing: None

2. Sampling Rate Converter Unit (SCU)

The SCU has ten SRC modules (six for high-sound-quality type; four for general-sound-quality type) that are useful for synchronization of asynchronous data, which is necessary for data transfer with external memory or external devices. It also provides the functions to change the number of channels, perform mixing, and control the volume.

2-1) Sampling Rate Converter (SRC)

- Asynchronous sampling rate conversion is available
- Supports resolutions up to 24 bits
- High-sound-quality type (THD + N*[1] is − 132 dB) and general-sound-quality type (THD + N*[1] is -96 dB)
- Automatically generates antialiasing filter coefficients
- Four modules support one, two, four, six, or eight channels, and six modules support one or two channels.

Note 1. Total harmonic distortion plus noise

2-2) Channel Transfer Unit (CTU)

- Downmixing and splitter functions
  - Conversion of eight input channels into two output channels
  - Conversion of six input channels into two output channels
  - Conversion of two input channels into four sets of two output channels
  - Conversion of one input channel into eight sets of one output channel
  - No conversion

2-3) Mixer (MIX)

- Mixing (adds) two to four sources into one
- Ratio for adding sources is selectable
- Ratio is dynamically changeable
- Mixing with volume ramp is available (ramp period is selectable)

2-4) Digital Volume and Mute Function (DVC)

- Volume control function including digital volume, volume ramp, and zero-crossing mute
- The digital volume function is specified by a 24-bit fixed-point value within the range from 0 to 8 times (mute or -120 to 18 dB)
- The volume ramp function can be used for soft mute, fade-in, fade-out, or desired volume adjustment
- The volume ramp period can be changed within the sampling range from the 0th to 23rd power of 2
- The zero-crossing mute function silences the sound at the zero-crossing point of the audio data

Remark The CTU, MIX, and DVC functional blocks are collectively called “CMD”.

3. Serial Sound Interface Unit (SSIU)

The SSIU, incorporating ten SSI modules, allows independent operation of SSI modules, operation of multiple SSI modules sharing the same serial clock, and connection or split of multichannel data.

- Incorporates ten SSI modules.
- Supports 6 channels/1 sound source with three SSI modules (SSI0, SSI1, and SSI2).
- Supports 8 channels/1 sound source with four SSI modules (SSI0, SSI1, SSI2, and SSI9).
- Operation of multiple modules on the same serial clock (SSI0/SSI1/SSI2/SSI3/SSI9, SSI3/SSI4/SSI9, SSI5/SSI6, or SSI7/SSI8).
- TDM format (Basic Configuration) corresponds to 4, 6, or 8-channel data.
- Handles 8-channel data on the serial bus and 6-channel data in the SoC (TDM extend mode).
- Handles 6-channel data on the serial bus and 8-channel data in the SoC (TDM extend mode).
- Connects monaural or stereo data to output TDM format data (TDM split mode).
- Splits TDM format input data into monaural or stereo data (TDM split mode).
- Connects stereo or multi data to output TDM format data (TDM ex-split mode).
- Splits TDM format input data into stereo or multi data (TDM ex-split mode).
- The frequency range of SCK signal is from 297.3 kHz to 12.5 MHz in master mode, and from 297.3 kHz to 12.5 MHz in slave mode.

4. SPDIF Interface (SPDIF)

The SPDIF is an interface that complies with the IEC60958 standard. Three SPDIF IPs are incorporated in the chip.

- Compliant with the IEC60958 standard (stereo and consumer use modes only)
- Sampling frequency: 32 kHz, 44.1 kHz, 48 kHz
- Audio word size: 16 to 24 bits/sample
- Biphase mark encoding
- Double buffer for data
- Serial data with parity
- Capable of simultaneous transmission and reception

- The receiver autodetects the IEC61937 compressed mode data.

5. Pulse Density Modulation (PDM)

The PDM has 3 PDM function channels and allows independent operation of each channel. 2 PDM IPs are incorporated in the chip.

- Capable of filtering 1-bit digital input data PDMDATnm (n = 0, 1; m = 0 to 2) and converting them into 20-bit or 16-bit digital data. The bit order is little-endian.
- IP supports stereo microphone (L/R sampling by rising/falling clock edge).
- IP supports sound activity detector.
- Each channel of IP includes programmable filters: 4th order sinc filter, high-pass filter (for suppression of DC bias), correction filter (for sinc passband distortion), half-band decimation filter (for aliasing distortion).
- IP supports programmable and flexible decimation ratios.
- The sinc filter is selectable as first-, second-, third-, fourth-order.
- IP supports DMA operation through APB.
- Each channel of IP has an internal buffer.
- 64 stages
- Capable of storing voice data during low power mode
- Error detection functions can be used for debugging.

6. Audio Clock Generator (ADG)

The ADG selects and supplies the necessary clock for the SSIU, SCU, or SPDIF module. It also divides the frequency of the selected clock and sends it outside the chip.

- Selects the clock signal from the AUDIO_CLKA, AUDIO_CLKB, or AUDIO_CLKC pin or the internal clock and supplies it to the SSIU, SCU, or SPDIF module.
- The frequency of the clock signals from the AUDIO_CLKA, AUDIO_CLKB, and AUDIO_CLKC pins and the internal clock can be divided before use.
- The divided clock can be output through the AUDIO_CLKOUT pin.

8.1.3.2 List of Routings

Data transmission paths in the audio module are shown in **Figure 8.1-3**.

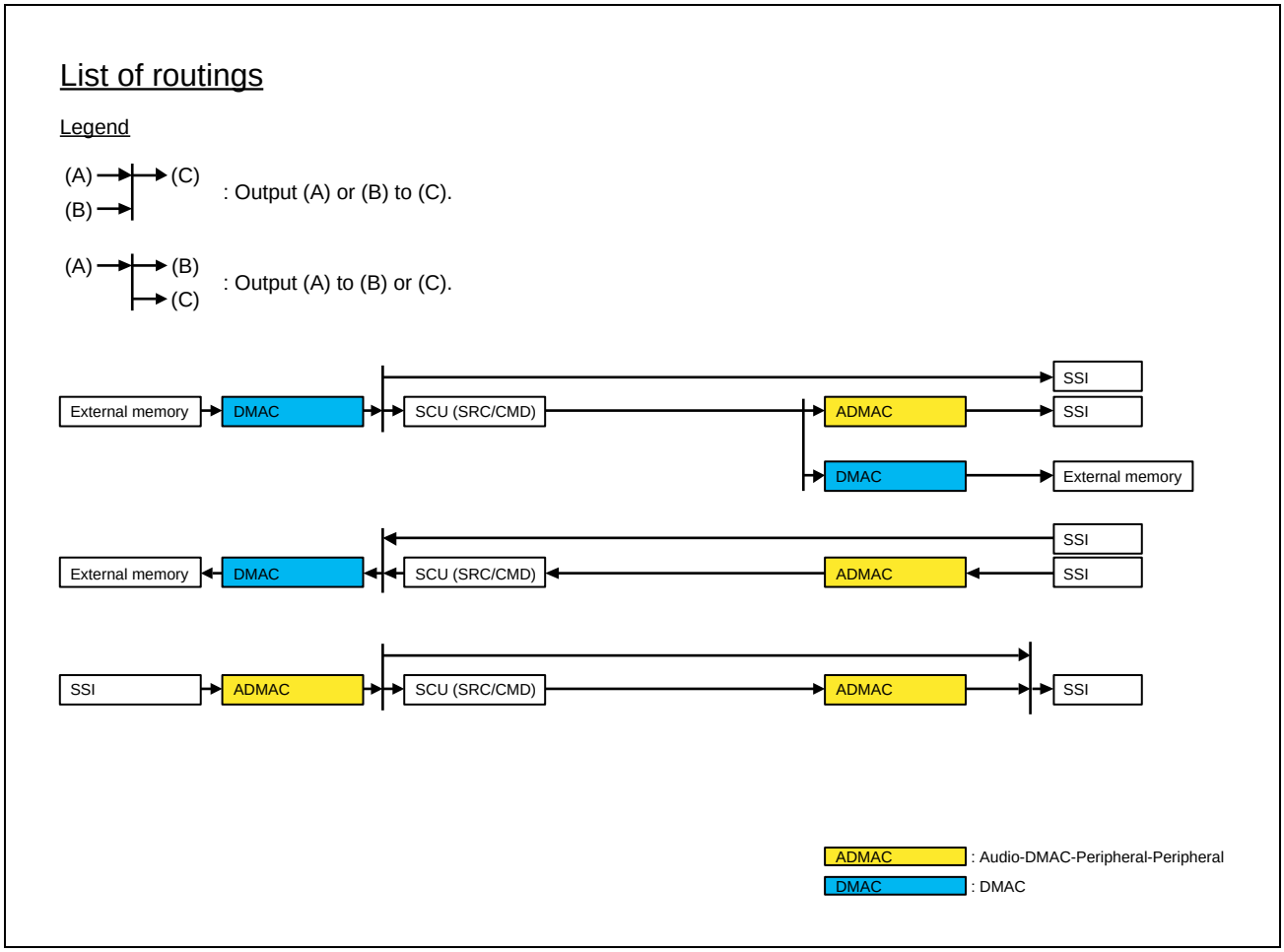


Figure 8.1-3 List of Routings

8.1.3.3 DMAC

The DMAC is used to transfer data between the internal/external memory of the chip and the audio module. **Table 8.1-2** lists the settings of the DMAC source/destination addresses.

SSI_BASE and SCU_BASE represent the base addresses of the SSI and SCU, respectively. For details, refer to section Address Map.

Table 8.1-2 DMAC Source/Destination Address Settings (1/2)

Name of Register	DMASAR/DMADAR	
SSI0-0_BUSIF	SSI_BASE	0000h
SSI0-1_BUSIF		0400h
SSI0-2_BUSIF		0800h
SSI0-3_BUSIF		0C00h
SSI1-0_BUSIF		1000h
SSI1-1_BUSIF		1400h
SSI1-2_BUSIF		1800h
SSI1-3_BUSIF		1C00h
SSI2-0_BUSIF		2000h
SSI2-1_BUSIF		2400h
SSI2-2_BUSIF		2800h
SSI2-3_BUSIF		2C00h
SSI3-0_BUSIF		3000h
SSI3-1_BUSIF		3400h
SSI3-2_BUSIF		3800h
SSI3-3_BUSIF		3C00h
SSI4-0_BUSIF		4000h
SSI4-1_BUSIF		4400h
SSI4-2_BUSIF		4800h
SSI4-3_BUSIF		4C00h
SSI5_BUSIF		5000h
SSI6_BUSIF		6000h
SSI7_BUSIF		7000h
SSI8_BUSIF		8000h
SSI9-0_BUSIF		9000h
SSI9-1_BUSIF		9400h
SSI9-2_BUSIF		9800h
SSI9-3_BUSIF		9C00h

Table 8.1-2 DMAC Source/Destination Address Settings (2/2)

Name of Register	DMASAR/DMADAR	
SRC0in_BUSIF*1	SCU_BASE	0000h
SRC1in_BUSIF*1		0400h
SRC2in_BUSIF*1		0800h
SRC3in_BUSIF*1		0C00h
SRC4in_BUSIF*1		1000h
SRC5in_BUSIF*1		1400h
SRC6in_BUSIF*1		1800h
SRC7in_BUSIF*1		1C00h
SRC8in_BUSIF*1		2000h
SRC9in_BUSIF*1		2400h
SRC0out_BUSIF*2		4000h
SRC1out_BUSIF*2		4400h
SRC2out_BUSIF*2		4800h
SRC3out_BUSIF*2		4C00h
SRC4out_BUSIF*2		5000h
SRC5out_BUSIF*2		5400h
SRC6out_BUSIF*2		5800h
SRC7out_BUSIF*2		5C00h
SRC8out_BUSIF*2		6000h
SRC9out_BUSIF*2		6400h
CMD0out_BUSIF*2		8000h
CMD1out_BUSIF*2		8400h

Note 1. Used only by DMASAR.

Note 2. Used only by DMADAR.

8.1.3.4 Data Format on Audio Local Bus

Figure 8.1-4 shows data format processing on the audio local bus. When write/read data passes through the DMAC, the order of data is rearranged to the appropriate data format in **Figure 8.1-4**.

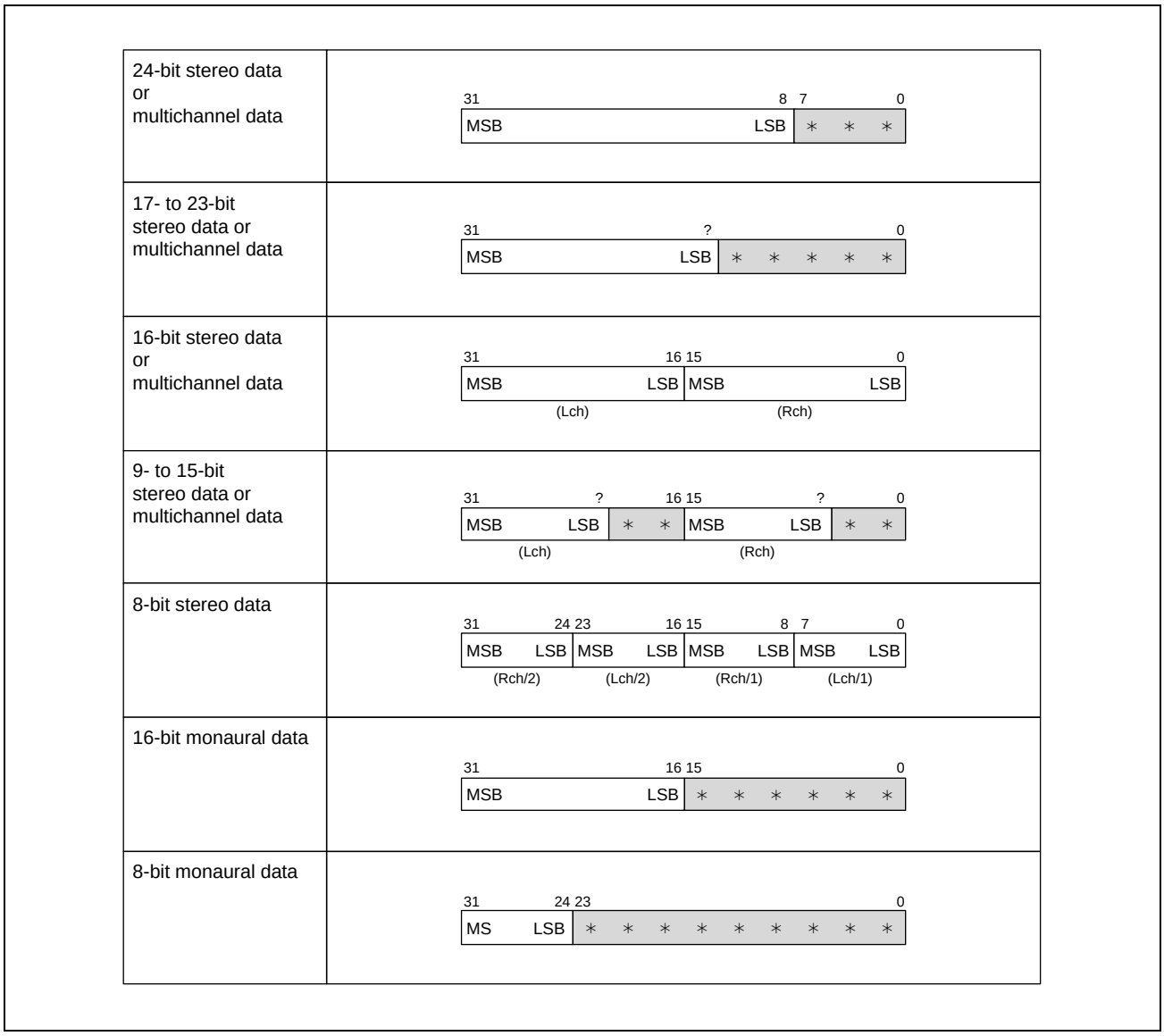


Figure 8.1-4 Data Format

Remark 1. Write 0 to the * bits in the figure when writing data. When reading data, ignore the values read from these bits.

Remark 2. In the 8-bit stereo data format, (Lch/1) and (Rch/1) indicate the data pair to be processed first and (Lch/2) and (Rch/2) indicate the next data pair to be processed.

Remark 3. Only the MSB-first data formats can be used.

8.1.3.5 Rearranging the Order of Data

For the audio modules (SSI, SCU), rearranging the order of data is possible in the channel unit. Places of data in each data format are defined as in **Table 8.1-3**.

Table 8.1-3 Definition of Data Places in Each Data Format (1)

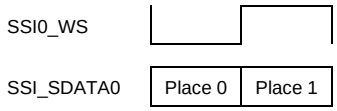
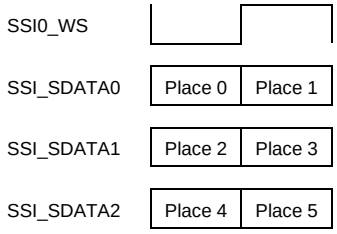
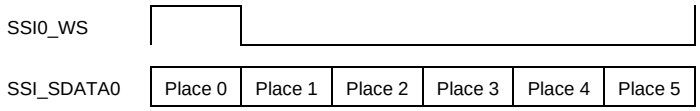
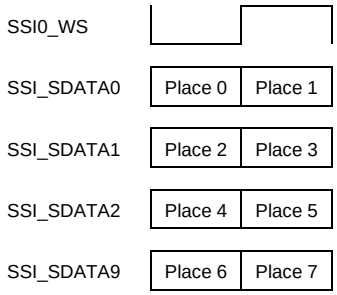
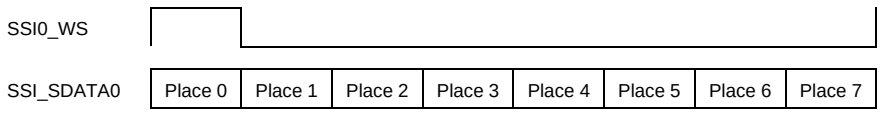
SSI • Stereo (2 channels)	
SSI • Stereo × 3 (6 channels)	
SSI • TDM (6 channels)	
SSI • Stereo × 4 (8 channels)	
SSI • TDM (8 channels)	

Table 8.1-3 Definition of Data Places in Each Data Format (2)

BUSIF • Stereo (2 channels) • 24 bits	External memory image 31 0 00h Place 0 ✳ 04h Place 1 ✳ 08h Place 0 ✳ 0Ch Place 1 ✳ … … ✳
BUSIF • Stereo (2 channels) • 16 bits	External memory image 31 0 00h Place 0 Place 1 04h Place 0 Place 1 08h Place 0 Place 1 0Ch Place 0 Place 1 … … …
BUSIF • Stereo (2 channels) • 8 bits	External memory image 31 0 00h Place 1 Place 0 Place 1 Place 0 04h Place 1 Place 0 Place 1 Place 0 … … … … …
BUSIF • Monaural (1 channel) • 8/16 bits	External memory image 31 0 00h Place 0 ✳ 04h Place 0 ✳ 08h Place 0 ✳ 0Ch Place 0 ✳ … … ✳
BUSIF • Multichannel (4 channels) • 24 bits	External memory image 31 0 00h Place 0 ✳ 04h Place 1 ✳ 08h Place 2 ✳ 0Ch Place 3 ✳ 10h Place 0 ✳ 14h Place 1 ✳ 18h Place 2 ✳ 1Ch Place 3 ✳ … … ✳
BUSIF • Multichannel (4 channels) • 16 bits	External memory image 31 0 00h Place 0 Place 1 04h Place 2 Place 3 08h Place 0 Place 1 0Ch Place 2 Place 3 … … …

Table 8.1-3 Definition of Data Places in Each Data Format (3)

<table>
<tr>
<td>BUSIF
• Multichannel (6 channels)
• 24 bits</td>
<td>External memory image
<table>
<tr><td></td><td>31</td><td>0</td></tr>
<tr><td>00h</td><td>Place 0</td><td>*</td></tr>
<tr><td>04h</td><td>Place 1</td><td>*</td></tr>
<tr><td>08h</td><td>Place 2</td><td>*</td></tr>
<tr><td>0Ch</td><td>Place 3</td><td>*</td></tr>
<tr><td>10h</td><td>Place 4</td><td>*</td></tr>
<tr><td>14h</td><td>Place 5</td><td>*</td></tr>
<tr><td>18h</td><td>Place 0</td><td>*</td></tr>
<tr><td>1Ch</td><td>Place 1</td><td>*</td></tr>
<tr><td>…</td><td>…</td><td>*</td></tr>
</table>
</td>
</tr>
<tr>
<td>BUSIF
• Multichannel (6 channels)
• 16 bits</td>
<td>External memory image
<table>
<tr><td></td><td>31</td><td>0</td></tr>
<tr><td>00h</td><td>Place 0</td><td>Place 1</td></tr>
<tr><td>04h</td><td>Place 2</td><td>Place 3</td></tr>
<tr><td>08h</td><td>Place 4</td><td>Place 5</td></tr>
<tr><td>0Ch</td><td>Place 0</td><td>Place 1</td></tr>
<tr><td>…</td><td>…</td><td>…</td></tr>
</table>
</td>
</tr>
<tr>
<td>BUSIF
• Multichannel (8 channels)
• 24 bits</td>
<td>External memory image
<table>
<tr><td></td><td>31</td><td>0</td></tr>
<tr><td>00h</td><td>Place 0</td><td>*</td></tr>
<tr><td>04h</td><td>Place 1</td><td>*</td></tr>
<tr><td>08h</td><td>Place 2</td><td>*</td></tr>
<tr><td>0Ch</td><td>Place 3</td><td>*</td></tr>
<tr><td>10h</td><td>Place 4</td><td>*</td></tr>
<tr><td>14h</td><td>Place 5</td><td>*</td></tr>
<tr><td>18h</td><td>Place 6</td><td>*</td></tr>
<tr><td>1Ch</td><td>Place 7</td><td>*</td></tr>
<tr><td>20h</td><td>Place 0</td><td>*</td></tr>
<tr><td>…</td><td>…</td><td>*</td></tr>
</table>
</td>
</tr>
<tr>
<td>BUSIF
• Multichannel (8 channels)
• 16 bits</td>
<td>External memory image
<table>
<tr><td></td><td>31</td><td>0</td></tr>
<tr><td>00h</td><td>Place 0</td><td>Place 1</td></tr>
<tr><td>04h</td><td>Place 2</td><td>Place 3</td></tr>
<tr><td>08h</td><td>Place 4</td><td>Place 5</td></tr>
<tr><td>0Ch</td><td>Place 6</td><td>Place 7</td></tr>
<tr><td>10h</td><td>Place 0</td><td>Place 1</td></tr>
<tr><td>…</td><td>…</td><td>…</td></tr>
</table>
</td>
</tr>
</table>

8.1.3.6 Clock and Reset

The clock and reset signals for the audio modules (ADG, ADMAC, SCU, SSIU, SPDIF, and PDM) can be switched on or off by setting the registers of the CPG.

8.1.3.6.1 Clock and Reset Configuration

Figure 8.1-5 shows the clock and reset configuration of the audio modules (ADG, ADMAC, SCU, SSIU, SPDIF, and PDM).

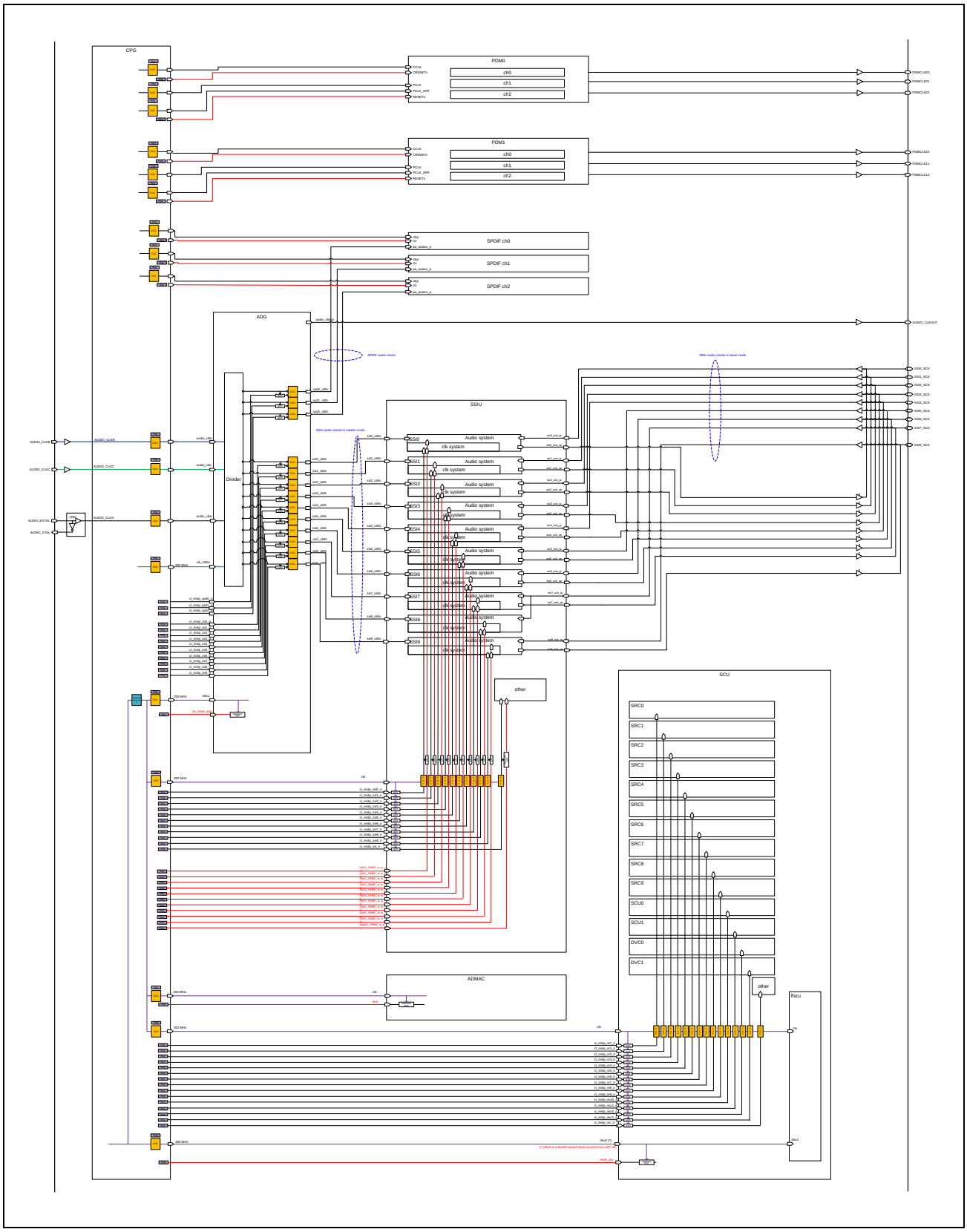


Figure 8.1-5 Clock and Reset Configuration of Audio Module

8.1.3.6.2 System Clock for IP

The system clock for each IP can be supplied or stopped by using the corresponding register of the CPG. The system clock for each IP is also used as a bus clock. Supply the system clock to the target IP before access to the register.

(1) System Clock for ADG

The system clock clks1 input to the ADG can be supplied or stopped by using the CLK9_ON bit of the CPG_CLKON_15 register. The clock is stopped at its initial state after power is on.

Table 8.1-4 ADG System Clock Control

Register Name	Bit	Function
CPG_CLKON_15	CLK9_ON	Supplying or stopping the system clock clks1 input to ADG (0: Stopped; 1: Supplied)

(2) System Clock for ADMAC

The system clock clk input to the ADMAC can be supplied or stopped by using the CLK8_ON bit of the CPG_CLKON_15 register. The clock is stopped at its initial state after power is on.

Table 8.1-5 ADMAC System Clock Control

Register Name	Bit	Function
CPG_CLKON_15	CLK8_ON	Supplying or stopping the system clock clk input to ADMAC (0: Stopped; 1: Supplied)

(3) System Clock for SCU

The system clocks clk and clkx2 input to the SCU can be supplied or stopped by using the CLK6_ON and CLK7_ON bits of the CPG_CLKON_15 register. The clocks are stopped at their initial states after power is on.

Table 8.1-6 SCU System Clock Control

Register Name	Bit	Function
CPG_CLKON_15	CLK6_ON	Supplying or stopping the system clock clk input to SCU (0: Stopped; 1: Supplied)
	CLK7_ON	Supplying or stopping the system clock clkx2 input to SCU (0: Stopped; 1: Supplied)

The system clocks clk and clkx2 input to the SCU are distributed to the given function modules within the SCU via the CGC cells. The system clock for the function modules can be supplied or stopped individually by using the CLK0_ON to CLK14_ON bits of the CPG_CLKON_23 register.

Table 8.1-7 System Clock Control for Internal Modules of SCU

Register Name	Bit	Function
CPG_CLKON_23	CLK0_ON	Supplying or stopping the system clock supplied to the DVC ch0 module of SCU (0: Stopped; 1: Supplied)
	CLK1_ON	Supplying or stopping the system clock supplied to the DVC ch1 module of SCU (0: Stopped; 1: Supplied)
	CLK2_ON	Supplying or stopping the system clock supplied to the CTU/MIX ch0 module of SCU (0: Stopped; 1: Supplied)
	CLK3_ON	Supplying or stopping the system clock supplied to the CTU/MIX ch1 module of SCU (0: Stopped; 1: Supplied)
	CLK4_ON	Supplying or stopping the system clock supplied to the SRC ch0 module of SCU (0: Stopped; 1: Supplied)
	CLK5_ON	Supplying or stopping the system clock supplied to the SRC ch1 module of SCU (0: Stopped; 1: Supplied)
	CLK6_ON	Supplying or stopping the system clock supplied to the SRC ch2 module of SCU (0: Stopped; 1: Supplied)
	CLK7_ON	Supplying or stopping the system clock supplied to the SRC ch3 module of SCU (0: Stopped; 1: Supplied)
	CLK8_ON	Supplying or stopping the system clock supplied to the SRC ch4 module of SCU (0: Stopped; 1: Supplied)
	CLK9_ON	Supplying or stopping the system clock supplied to the SRC ch5 module of SCU (0: Stopped; 1: Supplied)
	CLK10_ON	Supplying or stopping the system clock supplied to the SRC ch6 module of SCU (0: Stopped; 1: Supplied)
	CLK11_ON	Supplying or stopping the system clock supplied to the SRC ch7 module of SCU (0: Stopped; 1: Supplied)
	CLK12_ON	Supplying or stopping the system clock supplied to the SRC ch8 module of SCU (0: Stopped; 1: Supplied)
	CLK13_ON	Supplying or stopping the system clock supplied to the SRC ch9 module of SCU (0: Stopped; 1: Supplied)
	CLK14_ON	Supplying or stopping the system clock supplied to the modules other than SCU (0: Stopped; 1: Supplied)

Table 8.1-8 shows the relationship between the CPG registers (CPG_CLKON_15 and CPG_CLKON_23 registers) and the target function modules for clock control.

Table 8.1-8 SCU System Clock Control and Target Modules

CPG_CLKON_15 register		CPG_CLKON_23 register															
$\overline{\text{CLK6_ON}}$ bit	$\overline{\text{CLK7_ON}}$ bit	$\overline{\text{CLK0_ON}}$ bit	$\overline{\text{CLK1_ON}}$ bit	$\overline{\text{CLK2_ON}}$ bit	$\overline{\text{CLK3_ON}}$ bit	$\overline{\text{CLK4_ON}}$ bit	$\overline{\text{CLK5_ON}}$ bit	$\overline{\text{CLK6_ON}}$ bit	$\overline{\text{CLK7_ON}}$ bit	$\overline{\text{CLK8_ON}}$ bit	$\overline{\text{CLK9_ON}}$ bit	$\overline{\text{CLK10_ON}}$ bit	$\overline{\text{CLK11_ON}}$ bit	$\overline{\text{CLK12_ON}}$ bit	$\overline{\text{CLK13_ON}}$ bit	$\overline{\text{CLK14_ON}}$ bit	Supply or stop clock for module
1	1	1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Supply the clock for DVC0
1	1	—	1	—	—	—	—	—	—	—	—	—	—	—	—	—	Supply the clock for DVC1
1	1	—	—	1	—	—	—	—	—	—	—	—	—	—	—	—	Supply the clock for CTU0/MIX0
1	1	—	—	—	1	—	—	—	—	—	—	—	—	—	—	—	Supply the clock for CTU1/MIX1
1	1	—	—	—	—	1	—	—	—	—	—	—	—	—	—	—	Supply the clock for SRC0
1	1	—	—	—	—	—	1	—	—	—	—	—	—	—	—	—	Supply the clock for SRC1
1	1	—	—	—	—	—	—	1	—	—	—	—	—	—	—	—	Supply the clock for SRC2
1	1	—	—	—	—	—	—	—	1	—	—	—	—	—	—	—	Supply the clock for SRC3
1	1	—	—	—	—	—	—	—	—	1	—	—	—	—	—	—	Supply the clock for SRC4
1	1	—	—	—	—	—	—	—	—	—	1	—	—	—	—	—	Supply the clock for SRC5
1	1	—	—	—	—	—	—	—	—	—	—	1	—	—	—	—	Supply the clock for SRC6
1	1	—	—	—	—	—	—	—	—	—	—	—	1	—	—	—	Supply the clock for SRC7
1	1	—	—	—	—	—	—	—	—	—	—	—	—	1	—	—	Supply the clock for SRC8
1	1	—	—	—	—	—	—	—	—	—	—	—	—	—	1	—	Supply the clock for SRC9
1	1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	1	Supply the clock for other modules*1
1	1	0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Stop the clock for DVC0
1	1	—	0	—	—	—	—	—	—	—	—	—	—	—	—	—	Stop the clock for DVC1
1	1	—	—	0	—	—	—	—	—	—	—	—	—	—	—	—	Stop the clock for CTU0/MIX0
1	1	—	—	—	0	—	—	—	—	—	—	—	—	—	—	—	Stop the clock for CTU1/MIX1
1	1	—	—	—	—	0	—	—	—	—	—	—	—	—	—	—	Stop the clock for SRC0
1	1	—	—	—	—	—	0	—	—	—	—	—	—	—	—	—	Stop the clock for SRC1
1	1	—	—	—	—	—	—	0	—	—	—	—	—	—	—	—	Stop the clock for SRC2
1	1	—	—	—	—	—	—	—	0	—	—	—	—	—	—	—	Stop the clock for SRC3
1	1	—	—	—	—	—	—	—	—	0	—	—	—	—	—	—	Stop the clock for SRC4
1	1	—	—	—	—	—	—	—	—	—	0	—	—	—	—	—	Stop the clock for SRC5
1	1	—	—	—	—	—	—	—	—	—	—	0	—	—	—	—	Stop the clock for SRC6
1	1	—	—	—	—	—	—	—	—	—	—	—	0	—	—	—	Stop the clock for SRC7
1	1	—	—	—	—	—	—	—	—	—	—	—	—	0	—	—	Stop the clock for SRC8
1	1	—	—	—	—	—	—	—	—	—	—	—	—	—	0	—	Stop the clock for SRC9
1	1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0	Stop the clock for other modules*1
0	0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Stop all clocks

—: Don't care

Note 1. Function modules excluding SRC0 to SRC9, DVC0 and DVC1, CTU0 and CTU1, and MIX0 and MIX1 (including the setting registers, etc.)

(4) System clock for SSIU

The system clock clk input to the SSIU can be supplied or stopped by using the CLK5_ON bit of the CPG_CLKON_15 register. The clock is stopped at its initial state after power is on.

Table 8.1-9 SSIU System Clock Control

Register Name	Bit	Function
CPG_CLKON_15	CLK5_ON	Supplying or stopping the system clock clk input to SSIU (0: Stopped; 1: Supplied)

The system clock clk input to the SSIU is distributed to the given function modules within the SSIU via the CGC cells. The system clock for the function modules can be supplied or stopped individually by using the CLK0_ON to CLK10_ON bits of the CPG_CLKON_24 register.

Table 8.1-10 System Clock Control for Internal Modules of SSIU

Register Name	Bit	Function
CPG_CLKON_24	CLK0_ON	Supplying or stopping the system clock supplied to other modules (0: Stopped; 1: Supplied)
	CLK1_ON	Supplying or stopping the system clock supplied to the SSI ch0 module of SSIU (0: Stopped; 1: Supplied)
	CLK2_ON	Supplying or stopping the system clock supplied to the SSI ch1 module of SSIU (0: Stopped; 1: Supplied)
	CLK3_ON	Supplying or stopping the system clock supplied to the SSI ch2 module of SSIU (0: Stopped; 1: Supplied)
	CLK4_ON	Supplying or stopping the system clock supplied to the SSI ch3 module of SSIU (0: Stopped; 1: Supplied)
	CLK5_ON	Supplying or stopping the system clock supplied to the SSI ch4 module of SSIU (0: Stopped; 1: Supplied)
	CLK6_ON	Supplying or stopping the system clock supplied to the SSI ch5 module of SSIU (0: Stopped; 1: Supplied)
	CLK7_ON	Supplying or stopping the system clock supplied to the SSI ch6 module of SSIU (0: Stopped; 1: Supplied)
	CLK8_ON	Supplying or stopping the system clock supplied to the SSI ch7 module of SSIU (0: Stopped; 1: Supplied)
	CLK9_ON	Supplying or stopping the system clock supplied to the SSI ch8 module of SSIU (0: Stopped; 1: Supplied)
	CLK10_ON	Supplying or stopping the system clock supplied to the SSI ch9 module of SSIU (0: Stopped; 1: Supplied)

Table 8.1-11 shows the relationship between the CPG registers (CPG_CLKON_15 and CPG_CLKON_24 registers) and the target function modules for clock control.

Table 8.1-11 SSIU System Clock Control and Target Modules

CPG_CLKON_15 register	CPG_CLKON_24 register											
CLK5_ON bit	CLK0_ON bit	CLK1_ON bit	CLK2_ON bit	CLK3_ON bit	CLK4_ON bit	CLK5_ON bit	CLK6_ON bit	CLK7_ON bit	CLK8_ON bit	CLK9_ON bit	CLK10_ON bit	Supply or stop clock for module
1	1	—	—	—	—	—	—	—	—	—	—	Supply the system clock for other modules*1
1	—	1	—	—	—	—	—	—	—	—	—	Supply the system clock for SSI ch0 of SSIU
1	—	—	1	—	—	—	—	—	—	—	—	Supply the system clock for SSI ch1 of SSIU
1	—	—	—	1	—	—	—	—	—	—	—	Supply the system clock for SSI ch2 of SSIU
1	—	—	—	—	1	—	—	—	—	—	—	Supply the system clock for SSI ch3 of SSIU
1	—	—	—	—	—	1	—	—	—	—	—	Supply the system clock for SSI ch4 of SSIU
1	—	—	—	—	—	—	1	—	—	—	—	Supply the system clock for SSI ch5 of SSIU
1	—	—	—	—	—	—	—	1	—	—	—	Supply the system clock for SSI ch6 of SSIU
1	—	—	—	—	—	—	—	—	1	—	—	Supply the system clock for SSI ch7 of SSIU
1	—	—	—	—	—	—	—	—	—	1	—	Supply the system clock for SSI ch8 of SSIU
1	—	—	—	—	—	—	—	—	—	—	1	Supply the system clock for SSI ch9 of SSIU
1	0	—	—	—	—	—	—	—	—	—	—	Stop the system clock for other modules*1
1	—	0	—	—	—	—	—	—	—	—	—	Stop the system clock for SSI ch0 of SSIU
1	—	—	0	—	—	—	—	—	—	—	—	Stop the system clock for SSI ch1 of SSIU
1	—	—	—	0	—	—	—	—	—	—	—	Stop the system clock for SSI ch2 of SSIU
1	—	—	—	—	0	—	—	—	—	—	—	Stop the system clock for SSI ch3 of SSIU
1	—	—	—	—	—	0	—	—	—	—	—	Stop the system clock for SSI ch4 of SSIU
1	—	—	—	—	—	—	0	—	—	—	—	Stop the system clock for SSI ch5 of SSIU
1	—	—	—	—	—	—	—	0	—	—	—	Stop the system clock for SSI ch6 of SSIU
1	—	—	—	—	—	—	—	—	0	—	—	Stop the system clock for SSI ch7 of SSIU
1	—	—	—	—	—	—	—	—	—	0	—	Stop the system clock for SSI ch8 of SSIU
1	—	—	—	—	—	—	—	—	—	—	0	Stop the system clock for SSI ch9 of SSIU
0	—	—	—	—	—	—	—	—	—	—	—	Stop all system clocks

—: Don't care

Note 1. Function modules excluding SSI ch0 to SSI ch9 (including the setting registers, etc.)

(5) System Clock for SPDIF

The system clock input to ch0 and ch1 of the SPDIF and the system clock input to ch2 of the SPDIF can be supplied or stopped by using the CLK14_ON and CLK15_ON bits of the CPG_CLKON_15 register and the CLK0_ON bit of the CPG_CLKON_16 register, respectively. The clocks are stopped at their initial states after power is on.

Table 8.1-12 System Clock Control for SPDIF Ch0 and Ch1

Register Name	Bit	Function
CPG_CLKON_15	CLK14_ON	Supplying or stopping the system clock input to SPDIF ch0 (0: Stopped; 1: Supplied)
CPG_CLKON_15	CLK15_ON	Supplying or stopping the system clock input to SPDIF ch1 (0: Stopped; 1: Supplied)

Table 8.1-13 System Clock Control for SPDIF Ch2

Register Name	Bit	Function
CPG_CLKON_16	CLK0_ON	Supplying or stopping the system clock input to SPDIF ch2 (0: Stopped; 1: Supplied)

(6) System Clock for PDM

The system clock input to ch0 of the PDM and the system clock input to ch1 of the PDM can be supplied or stopped by using the CLK1_ON to CLK3_ON bits of the CPG_CLKON_16 register and the CLK4_ON to CLK6_ON bits of the CPG_CLKON_16 register, respectively. The clocks are stopped at their initial states after power is on.

Table 8.1-14 System Clock Control for PDM Ch0

Register Name	Bit	Function
CPG_CLKON_16	CLK1_ON	Supplying or stopping the system clock PCLK input to PDM ch0 (0: Stopped; 1: Supplied)
	CLK2_ON	Supplying or stopping the system clock PCLK_SFR input to PDM ch0 (0: Stopped; 1: Supplied)

Table 8.1-15 External I/F Reference Clock Control for PDM Ch0

Register Name	Bit	Function
CPG_CLKON_16	CLK3_ON	Supplying or stopping the external I/F reference clock CCLK for PDM ch0 (0: Stopped; 1: Supplied)

Table 8.1-16 System Clock Control for PDM Ch1

Register Name	Bit	Function
CPG_CLKON_16	CLK4_ON	Supplying or stopping the system clock PCLK input to PDM ch1 (0: Stopped; 1: Supplied)
	CLK5_ON	Supplying or stopping the system clock PCLK_SFR input to PDM ch1 (0: Stopped; 1: Supplied)

Table 8.1-17 External I/F Reference Clock Control for PDM Ch1

Register Name	Bit	Function
CPG_CLKON_16	CLK6_ON	Supplying or stopping the external I/F reference clock CCLK for PDM ch1 (0: Stopped; 1: Supplied)

8.1.3.6.3 ADG Divider Input Clocks

The clocks input to the divider within the ADG can be supplied or stopped by using the CLK10_ON to CLK13_ON bits of the CPG_CLKON_15 register. The clocks are stopped at their initial states after power is on.

After releasing the ADG from the reset state, set the operating mode by the corresponding register of the ADG before supplying the target clock input to the divider.

Table 8.1-18 ADG Internal Divider Input Clock Control

Register Name	Bit	Function
CPG_CLKON_15	CLK10_ON	Supplying or stopping the internal clock (200 MHz) input to ADG (0: Stopped; 1: Supplied)
	CLK11_ON	Supplying or stopping the audio_clka clock input to ADG (0: Stopped; 1: Supplied)
	CLK12_ON	Supplying or stopping the audio_clkb clock input to ADG (0: Stopped; 1: Supplied)
	CLK13_ON	Supplying or stopping the audio_clkc clock input to ADG (0: Stopped; 1: Supplied)

8.1.3.6.4 ADG Output Audio Master Clocks

(1) ADG Output Audio Master Clocks (for SSIU)

Audio master clocks input to the SSIU (ssi[0-9]_clkfs) are generated via the divider within the ADG. The audio master clocks generated by the ADG can be supplied or stopped by using the CLK0_ON to CLK9 _ON bits of the CPG_CLKON_22 register. The clocks are stopped at their initial states after power is on.

Table 8.1-19 ADG Output Audio Master Clock Control (for SSIU)

Register Name	Bit	Function
CPG_CLKON_22	CLK0_ON	Supplying or stopping the audio master clock ssi0_clkfs output by ADG for SSI ch0 (0: Stopped; 1: Supplied)
	CLK1_ON	Supplying or stopping the audio master clock ssi1_clkfs output by ADG for SSI ch1 (0: Stopped; 1: Supplied)
	CLK2_ON	Supplying or stopping the audio master clock ssi2_clkfs output by ADG for SSI ch2 (0: Stopped; 1: Supplied)
	CLK3_ON	Supplying or stopping the audio master clock ssi3_clkfs output by ADG for SSI ch3 (0: Stopped; 1: Supplied)
	CLK4_ON	Supplying or stopping the audio master clock ssi4_clkfs output by ADG for SSI ch4 (0: Stopped; 1: Supplied)
	CLK5_ON	Supplying or stopping the audio master clock ssi5_clkfs output by ADG for SSI ch5 (0: Stopped; 1: Supplied)
	CLK6_ON	Supplying or stopping the audio master clock ssi6_clkfs output by ADG for SSI ch6 (0: Stopped; 1: Supplied)
	CLK7_ON	Supplying or stopping the audio master clock ssi7_clkfs output by ADG for SSI ch7 (0: Stopped; 1: Supplied)
	CLK8_ON	Supplying or stopping the audio master clock ssi8_clkfs output by ADG for SSI ch8 (0: Stopped; 1: Supplied)
	CLK9_ON	Supplying or stopping the audio master clock ssi9_clkfs output by ADG for SSI ch9 (0: Stopped; 1: Supplied)

When stopping the audio master clocks, only do so after stopping operation of the target SSIU channels.

The audio master clocks should be supplied after setting the ADG registers related to the selection of the frequency and clock source of the audio master clock and setting the registers related to the operating mode of the target SSIU channels.

(2) ADG Output Audio Master Clocks (for SPDIF)

Audio master clocks input to the SPDIF (spd[0-2]_clkfs) are generated via the divider within the ADG. The audio master clocks generated by the ADG can be supplied or stopped by using the CLK10_ON to CLK12 _ON bits of the CPG_CLKON_22 register. The clocks are stopped at their initial states after power is on.

Table 8.1-20 ADG Output Audio Master Clock Control (for SPDIF)

Register Name	Bit	Function
CPG_CLKON_22	CLK10_ON	Supplying or stopping the audio master clock spd0_clkfs output by ADG for SPDIF ch0 (0: Stopped; 1: Supplied)
	CLK11_ON	Supplying or stopping the audio master clock spd1_clkfs output by ADG for SPDIF ch1 (0: Stopped; 1: Supplied)
	CLK12_ON	Supplying or stopping the audio master clock spd2_clkfs output by ADG for SPDIF ch2 (0: Stopped; 1: Supplied)

When stopping the audio master clocks, only do so after stopping operation of the target SPDIF channels.

The audio master clocks should be supplied after setting the ADG registers related to the selection of the frequency and clock source of the audio master clock and setting the registers related to the operating mode of the target SPDIF channels.

8.1.3.6.5 Reset for IP

Initialization of each IP can be controlled by using the corresponding register of the CPG. Be sure to release the target IP from the reset state before access to the register for the IP.

(1) Reset for ADG

For the ADG, initialization of the whole IP can be controlled by using the RSTB14 bit of the CPG_RST_14 register.

Note that the ADG should be reset after stopping operation of the SSIU, SCU, and SPDIF. Otherwise, correct operation is not guaranteed. When releasing the ADG from the reset state, only do so after supplying the system clock for the ADG.

Table 8.1-21 ADG Reset Control

Register Name	Bit	Function
CPG_RST_14	RSTB14	Resetting the whole ADG or releasing the reset (0: Reset; 1: Release from reset)

(2) Reset for ADMAC

For the ADMAC, initialization of the whole IP can be controlled by using the RSTB13 bit of the CPG_RST_14 register.

Note that the ADMAC should be reset after stopping operation of the target channels of the IP involved in the data transfer path. Otherwise, correct operation is not guaranteed. When releasing the ADMAC from the reset state, only do so after supplying the system clock for the ADMAC.

Table 8.1-22 ADMAC Reset Control

Register Name	Bit	Function
CPG_RST_14	RSTB13	Resetting the whole ADMAC or releasing the reset (0: Reset; 1: Release from reset)

(3) Reset for SCU

For the SCU, initialization of the whole IP can be controlled by using the RSTB12 bit of the CPG_RST_14 register.

Note that the SCU should be reset after stopping operation of the target channels of the IP involved in the data transfer path. Otherwise, correct operation is not guaranteed. When releasing the SCU from the reset state, only do so after supplying the system clock for the SCU.

Table 8.1-23 SCU Reset Control

Register Name	Bit	Function
CPG_RST_14	RSTB12	Resetting the whole SCU or releasing the reset (0: Reset; 1: Release from reset)

(4) Reset for SSIU

For the SSIU, initialization of the whole SSIU or initialization of the individual modules within the SSIU can be controlled by using the RSTB1 to RSTB11 bits of the CPG_RST_14 register.

Note that the SSIU should be reset after stopping operation of the target channels of the IP involved in the data transfer path. Otherwise, correct operation is not guaranteed. When releasing the SSIU from the reset state, only do so after supplying the system clock for the SSIU.

Table 8.1-24 SSIU Reset Control

Register Name	Bit	Function
CPG_RST_14	RSTB1	Resetting the modules other than those below (the setting register modules, etc.) or releasing the reset (0: Reset; 1: Release from reset)
	RSTB2	Resetting the SSI ch0 module or releasing the reset (0: Reset; 1: Release from reset)
	RSTB3	Resetting the SSI ch1 module or releasing the reset (0: Reset; 1: Release from reset)
	RSTB4	Resetting the SSI ch2 module or releasing the reset (0: Reset; 1: Release from reset)
	RSTB5	Resetting the SSI ch3 module or releasing the reset (0: Reset; 1: Release from reset)
	RSTB6	Resetting the SSI ch4 module or releasing the reset (0: Reset; 1: Release from reset)
	RSTB7	Resetting the SSI ch5 module or releasing the reset (0: Reset; 1: Release from reset)
	RSTB8	Resetting the SSI ch6 module or releasing the reset (0: Reset; 1: Release from reset)
	RSTB9	Resetting the SSI ch7 module or releasing the reset (0: Reset; 1: Release from reset)
	RSTB10	Resetting the SSI ch8 module or releasing the reset (0: Reset; 1: Release from reset)
	RSTB11	Resetting the SSI ch9 module or releasing the reset (0: Reset; 1: Release from reset)

Table 8.1-25 lists the RSTB1 to RSTB11 bits of the CPG_RST_14 register and the system clock settings required for resetting the target channels or releasing them from the reset state.

Table 8.1-25 SSIU Reset Control Registers and Required System Clock Settings

Clock setting required for reset or reset release operation by RSTB1 to RSTB11 of CPG_RST_14 register	CPG_CLKON_15 register	CPG_CLKON_24 register										
	CLK5_ON bit	CLK0_ON bit	CLK1_ON bit	CLK2_ON bit	CLK3_ON bit	CLK4_ON bit	CLK5_ON bit	CLK6_ON bit	CLK7_ON bit	CLK8_ON bit	CLK9_ON bit	CLK10_ON bit
RSTB1	1	1	—	—	—	—	—	—	—	—	—	—
RSTB2	1	—	1	—	—	—	—	—	—	—	—	—
RSTB3	1	—	—	1	—	—	—	—	—	—	—	—
RSTB4	1	—	—	—	1	—	—	—	—	—	—	—
RSTB5	1	—	—	—	—	1	—	—	—	—	—	—
RSTB6	1	—	—	—	—	—	1	—	—	—	—	—
RSTB7	1	—	—	—	—	—	—	1	—	—	—	—
RSTB8	1	—	—	—	—	—	—	—	1	—	—	—
RSTB9	1	—	—	—	—	—	—	—	—	1	—	—
RSTB10	1	—	—	—	—	—	—	—	—	—	1	—
RSTB11	1	—	—	—	—	—	—	—	—	—	—	1

(5) Reset for SPDIF

For ch0 of the SPDIF, initialization of the whole IP can be controlled by using the RSTB15 bit of the CPG_RST_14 register. For ch1 and ch2 of the SPDIF, initialization of the whole IP can be controlled by using the RSTB0 and RSTB1 bits of the CPG_RST_15 register.

Note that the SPDIF should be reset after stopping operation of the target channels of the IP involved in the data transfer path. Otherwise, correct operation is not guaranteed. When releasing the SPDIF from the reset state, only do so after supplying the system clock for the SPDIF.

Table 8.1-26 SPDIF Reset Control (1)

Register Name	Bit	Function
CPG_RST_14	RSTB15	Resetting SPDIF ch0 or releasing the reset (0: Reset; 1: Release from reset)

Table 8.1-26 SPDIF Reset Control (2)

Register Name	Bit	Function
CPG_CLKON_15	RSTB0	Resetting SPDIF ch1 or releasing the reset (0: Reset; 1: Release from reset)
	RSTB1	Resetting SPDIF ch2 or releasing the reset (0: Reset; 1: Release from reset)

(6) Reset for PDM

For ch0 of the PDM, initialization of the whole IP can be controlled by using the RSTB2 and RSTB3 bits of the CPG_RST_15 register. For ch1 of the PDM, initialization of the whole IP can be controlled by using the RSTB4 and RSTB5 bits of the CPG_RST_15 register.

Note that the PDM should be reset after stopping operation of the target channels of the IP involved in the data transfer path. Otherwise, correct operation is not guaranteed. When releasing the PDM from the reset state, only do so after supplying the system clock for the PDM.

Table 8.1-27 PDM Reset Control (1)

Register Name	Bit	Function
CPG_RST_15	RSTB2	Resetting the PCK/PCLK_SFR clock circuit for PDM ch0 or releasing the reset (0: Reset; 1: Release from reset)
	RSTB3	Resetting the CCLK clock circuit for PDM ch0 or releasing the reset (0: Reset; 1: Release from reset)

Table 8.1-27 PDM Reset Control (2)

Register Name	Bit	Function
CPG_CLKON_15	RSTB4	Resetting the PCK/PCLK_SFR clock circuit for PDM ch1 or releasing the reset (0: Reset; 1: Release from reset)
	RSTB5	Resetting the CCLK clock circuit for PDM ch1 or releasing the reset (0: Reset; 1: Release from reset)

8.1.3.6.6 Module Reset for SCU

Table 8.1-28 lists the software reset registers of the SCU. By using these registers, the SCU can control initialization of the target function modules and the associated registers (excluding the software reset registers).

Table 8.1-28 SCU Internal Software Reset Control

Register Name	Bit	Function
SRCm_SWRSR (m = 0 to 9)	SWRST	Resetting the circuit of the corresponding SRC channel within SCU and the associated register (excluding SRC[m]_SWRSR) or releasing the reset (0: Reset; 1: Release from reset)
CTUn_SWRSR (n = 00, 01, 02, 03, 10, 11, 12, or 13)	SWRST	Resetting the circuit of the corresponding CTU channel within SCU and the associated register (excluding CTU[n]_SWRSR) or releasing the reset (0: Reset; 1: Release from reset)
MIXp_SWRSR (p = 0 or 1)	SWRST	Resetting the circuit of the corresponding MIX channel within SCU and the associated register (excluding MIX[p]_SWRSR) or releasing the reset (0: Reset; 1: Release from reset)
DVCp_SWRSR (p = 0 or 1)	SWRST	Resetting the circuit of the corresponding DVC channel within SCU and the associated register (excluding DVC[p]_SWRSR) or releasing the reset (0: Reset; 1: Release from reset)

For reset or reset release operations using the software reset registers, the system clock must be supplied to the target function modules beforehand. **Table 8.1-29** lists the relationship between the software reset registers of the SCU and the clock settings required for initialization control.

Table 8.1-29 SCU Internal Software Reset Control Registers and Required Clock Settings

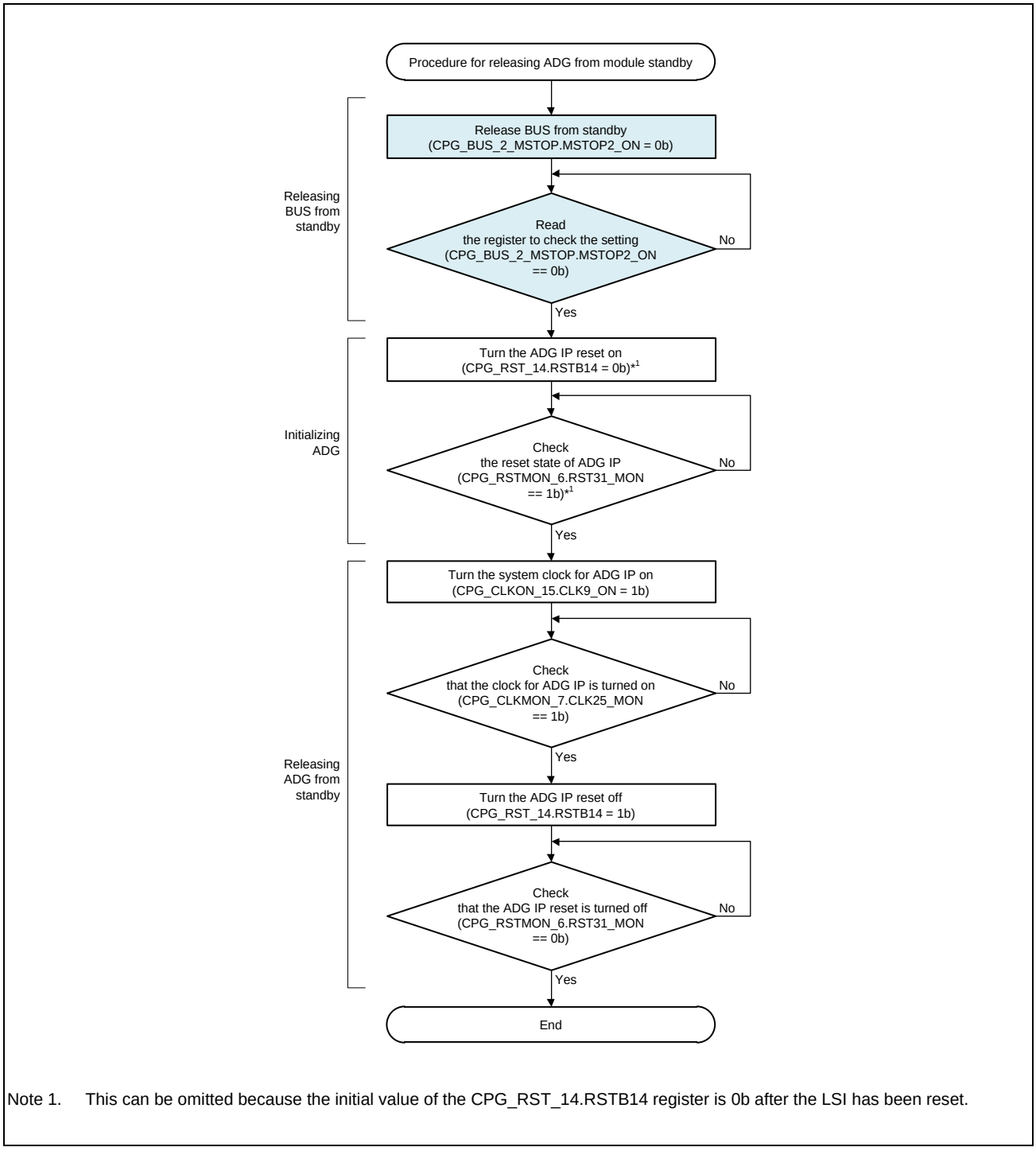
Software reset register within SCU	CPG_CLKON_15 register		CPG_CLKON_23 register													
	CLK6_ON bit	CLK7_ON bit	CLK0_ON bit	CLK1_ON bit	CLK2_ON bit	CLK3_ON bit	CLK4_ON bit	CLK5_ON bit	CLK6_ON bit	CLK7_ON bit	CLK8_ON bit	CLK9_ON bit	CLK10_ON bit	CLK11_ON bit	CLK12_ON bit	CLK13_ON bit	CLK14_ON bit
DVC0_SWRSR	1	1	1	—	—	—	—	—	—	—	—	—	—	—	—	—	1
DVC1_SWRSR	1	1	—	1	—	—	—	—	—	—	—	—	—	—	—	—	1
CTU00_SWRSR	1	1	—	—	1	—	—	—	—	—	—	—	—	—	—	—	1
CTU01_SWRSR																	
CTU02_SWRSR																	
CTU03_SWRSR																	
MIX0_SWRSR																	
CTU10_SWRSR	1	1	—	—	—	1	—	—	—	—	—	—	—	—	—	—	1
CTU11_SWRSR																	
CTU12_SWRSR																	
CTU13_SWRSR																	
MIX1_SWRSR																	
SRC0_SWRSR	1	1	—	—	—	—	1	—	—	—	—	—	—	—	—	—	1
SRC1_SWRSR	1	1	—	—	—	—	—	1	—	—	—	—	—	—	—	—	1
SRC2_SWRSR	1	1	—	—	—	—	—	—	1	—	—	—	—	—	—	—	1
SRC3_SWRSR	1	1	—	—	—	—	—	—	—	1	—	—	—	—	—	—	1
SRC4_SWRSR	1	1	—	—	—	—	—	—	—	—	1	—	—	—	—	—	1
SRC5_SWRSR	1	1	—	—	—	—	—	—	—	—	—	1	—	—	—	—	1
SRC6_SWRSR	1	1	—	—	—	—	—	—	—	—	—	—	1	—	—	—	1
SRC7_SWRSR	1	1	—	—	—	—	—	—	—	—	—	—	—	1	—	—	1
SRC8_SWRSR	1	1	—	—	—	—	—	—	—	—	—	—	—	—	1	—	1
SRC9_SWRSR	1	1	—	—	—	—	—	—	—	—	—	—	—	—	—	1	1

8.1.3.7 Procedure for Releasing IP from Module Standby

The following describes the procedure for releasing each IP from module standby.

8.1.3.7.1 Procedure for Releasing ADG from Module Standby

Figure 8.1-6 shows the procedure for releasing the ADG from module standby.

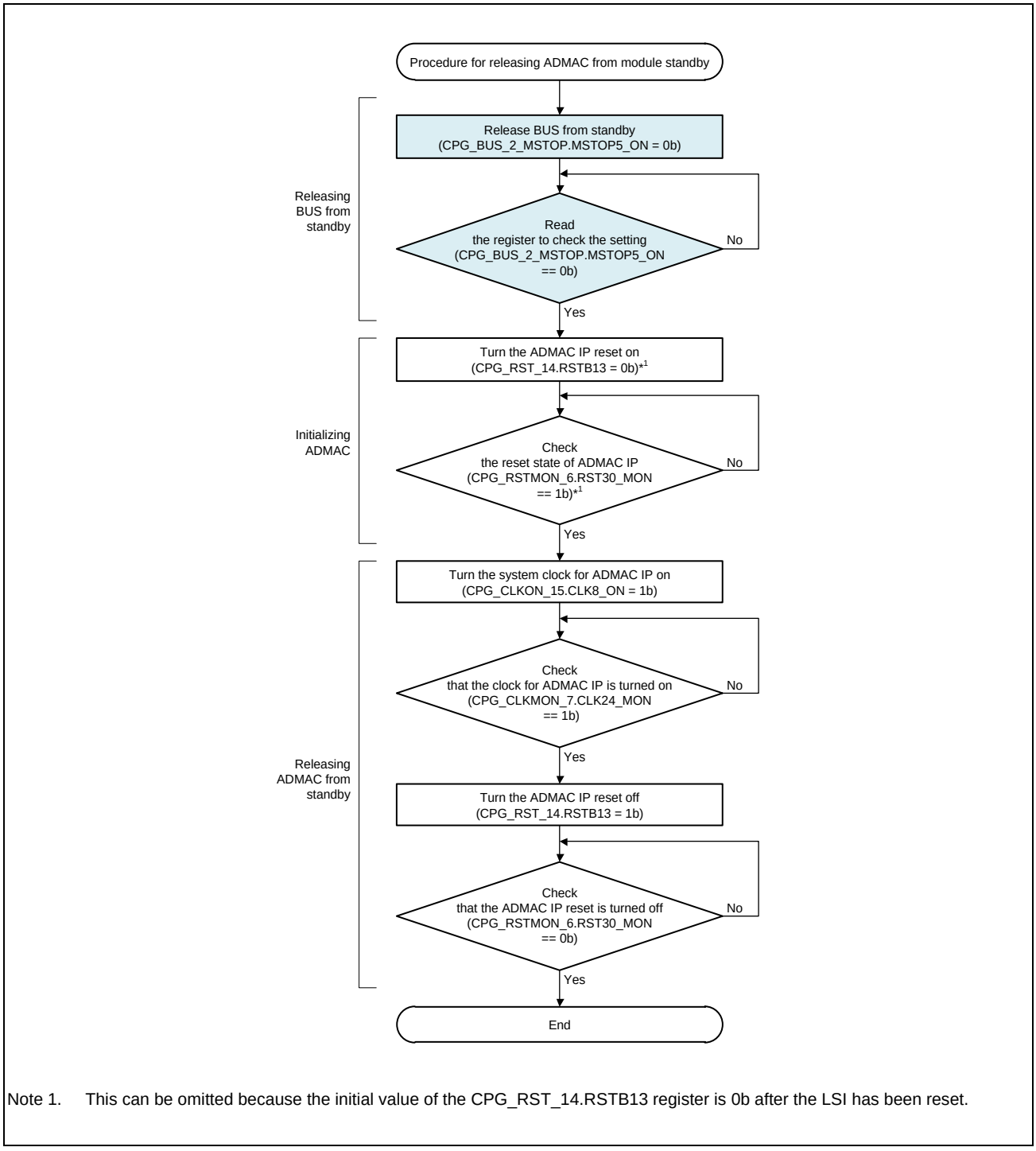


Note 1. This can be omitted because the initial value of the CPG_RST_14.RSTB14 register is 0b after the LSI has been reset.

Figure 8.1-6 Procedure for Releasing ADG from Module Standby

8.1.3.7.2 Procedure for Releasing ADMAC from Module Standby

Figure 8.1-7 shows the procedure for releasing the ADMAC from module standby.

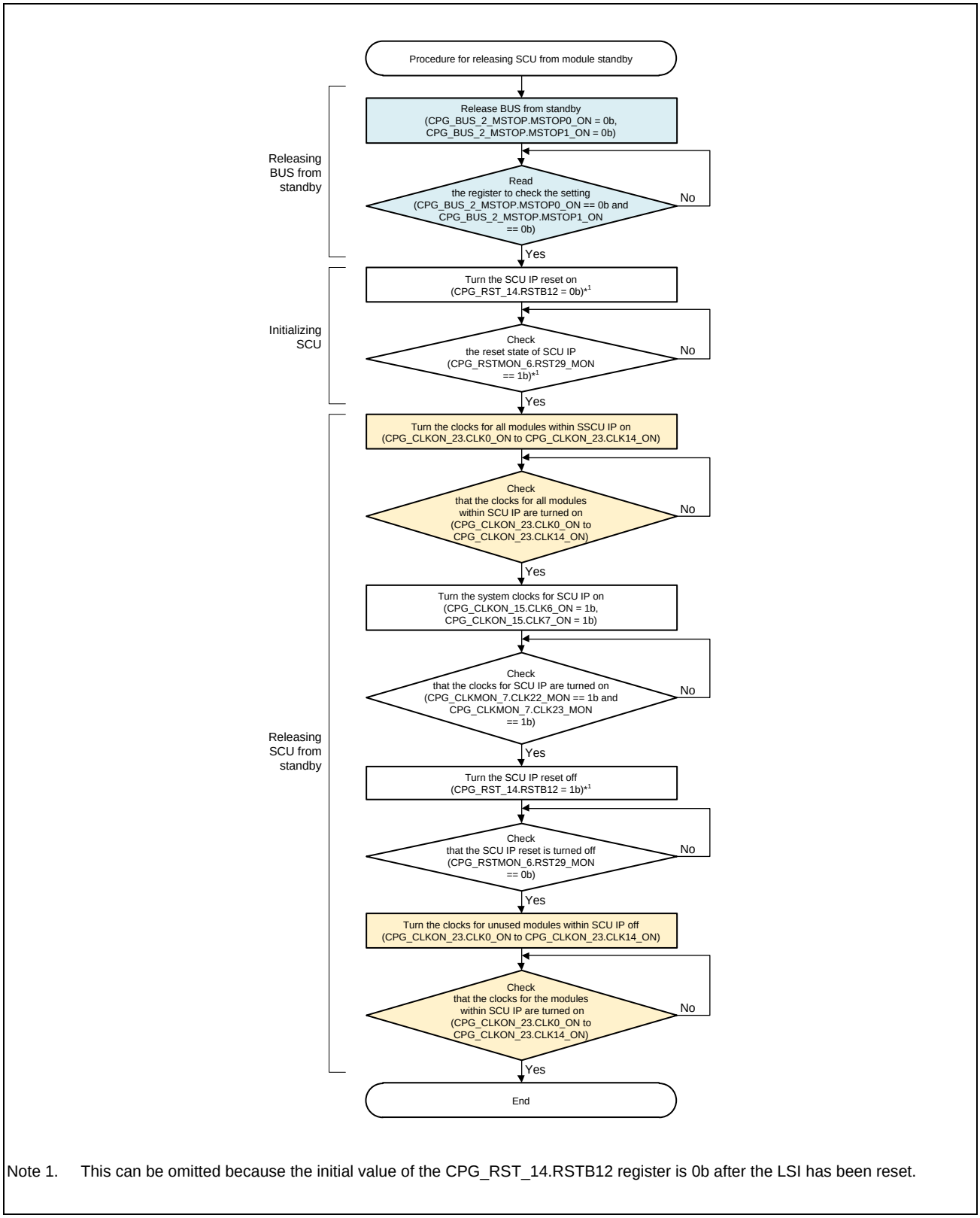


Note 1. This can be omitted because the initial value of the CPG_RST_14.RSTB13 register is 0b after the LSI has been reset.

Figure 8.1-7 Procedure for Releasing ADMAC from Module Standby

8.1.3.7.3 Procedure for Releasing SCU from Module Standby

Figure 8.1-8 shows the procedure for releasing the SCU from module standby.

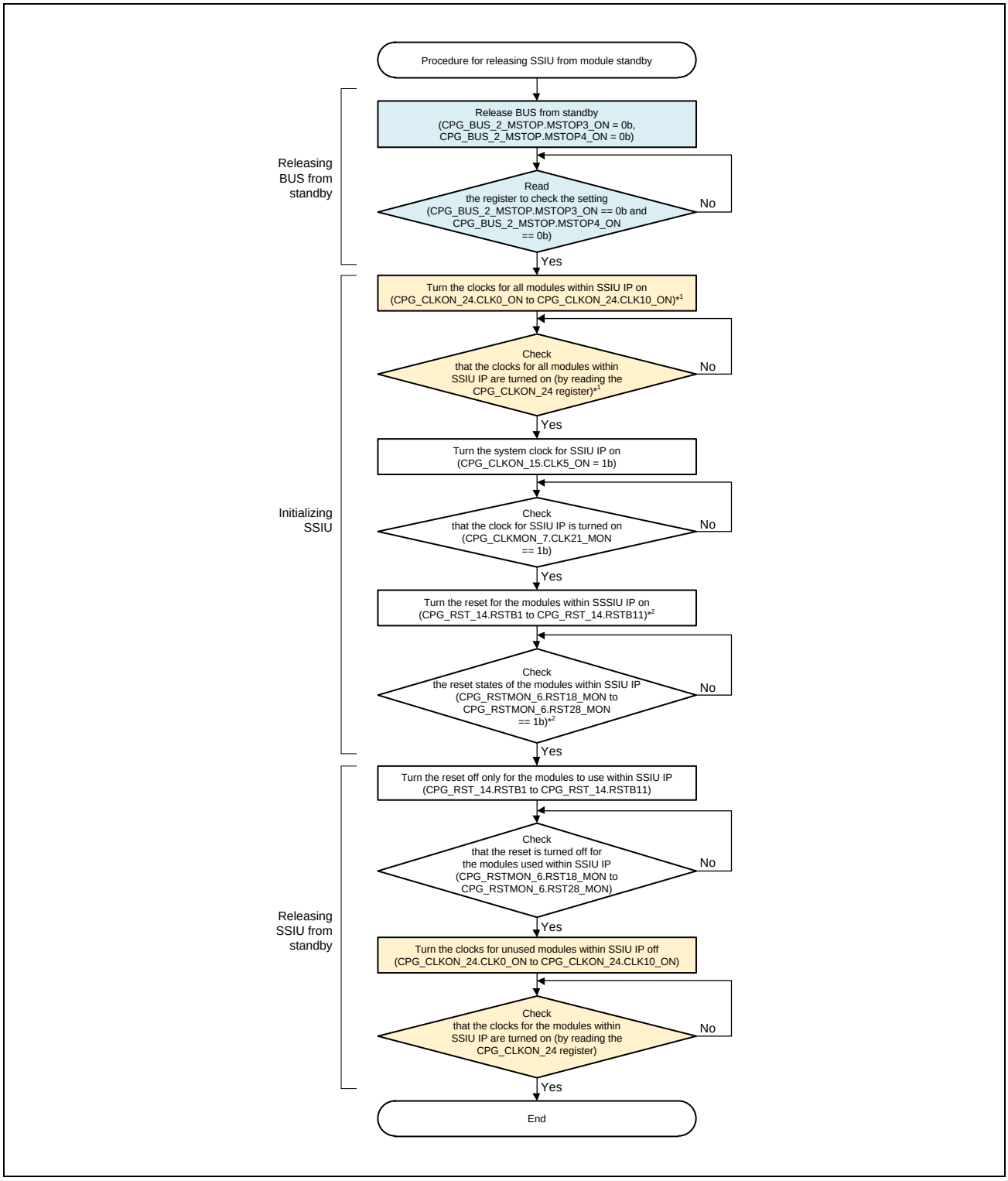


Note 1. This can be omitted because the initial value of the CPG_RST_14.RSTB12 register is 0b after the LSI has been reset.

Figure 8.1-8 Procedure for Releasing SCU from Module Standby

8.1.3.7.4 Procedure for Releasing SSIU from Module Standby

Figure 8.1-9 shows the procedure for releasing the SSIU from module standby.



(Continued on next page)

(Continued from previous page)

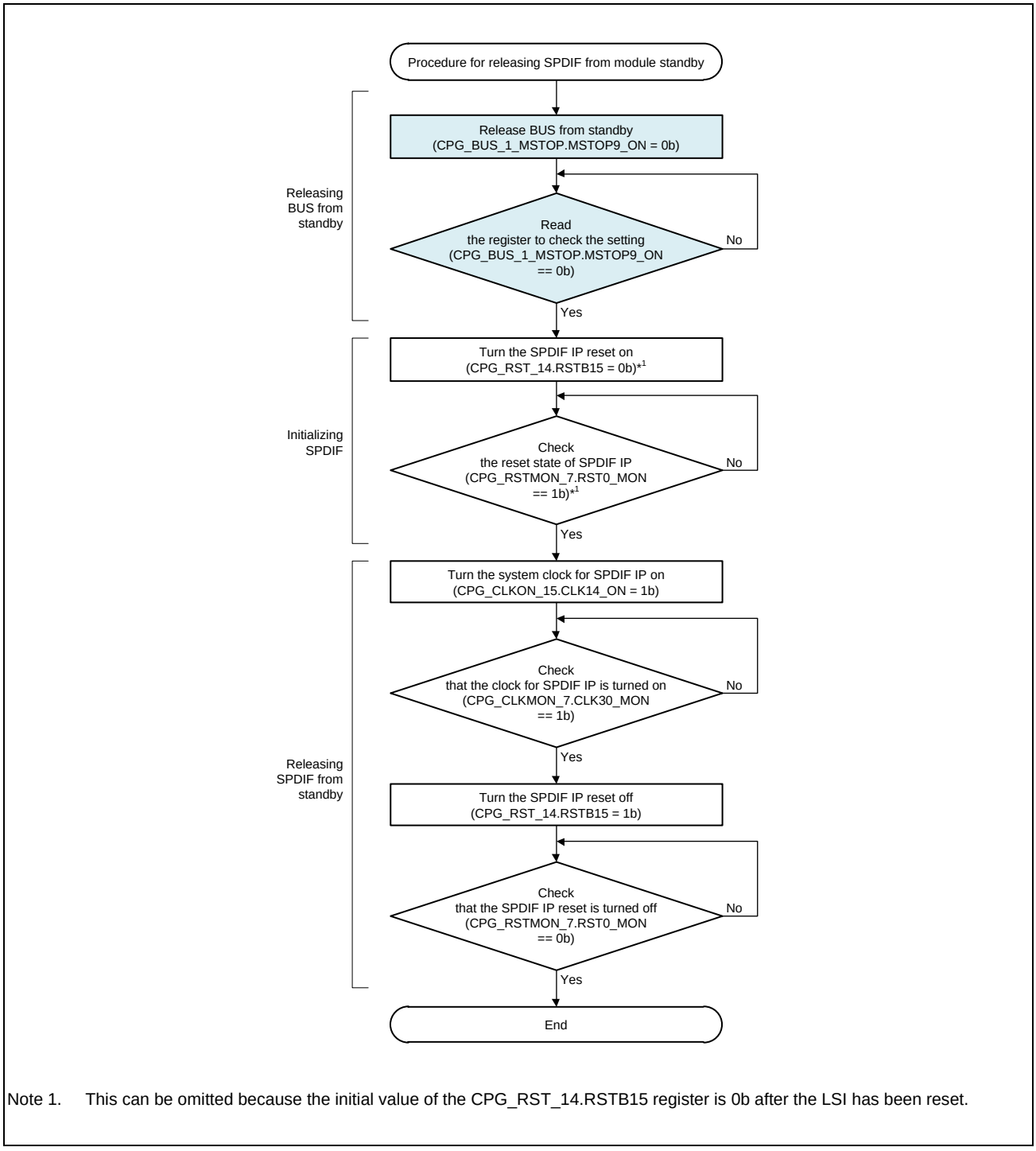
Note 1. This can be omitted because the initial value of the CPG_CLKON_24.CLK0_ON to CPG_CLKON_24.CLK10_ON registers is 1b after the LSI has been reset.

Note 2. This can be omitted because the initial value of the CPG_RST_14.RSTB1 to CPG_RST_14.RSTB11 registers is 0b after the LSI has been reset.

Figure 8.1-9 Procedure for Releasing SSIU from Module Standby

8.1.3.7.5 Procedure for Releasing SPDIF from Module Standby

Figure 8.1-10 shows the procedure for releasing the SPDIF from module standby, taking SPDIF IP ch0 as an example.

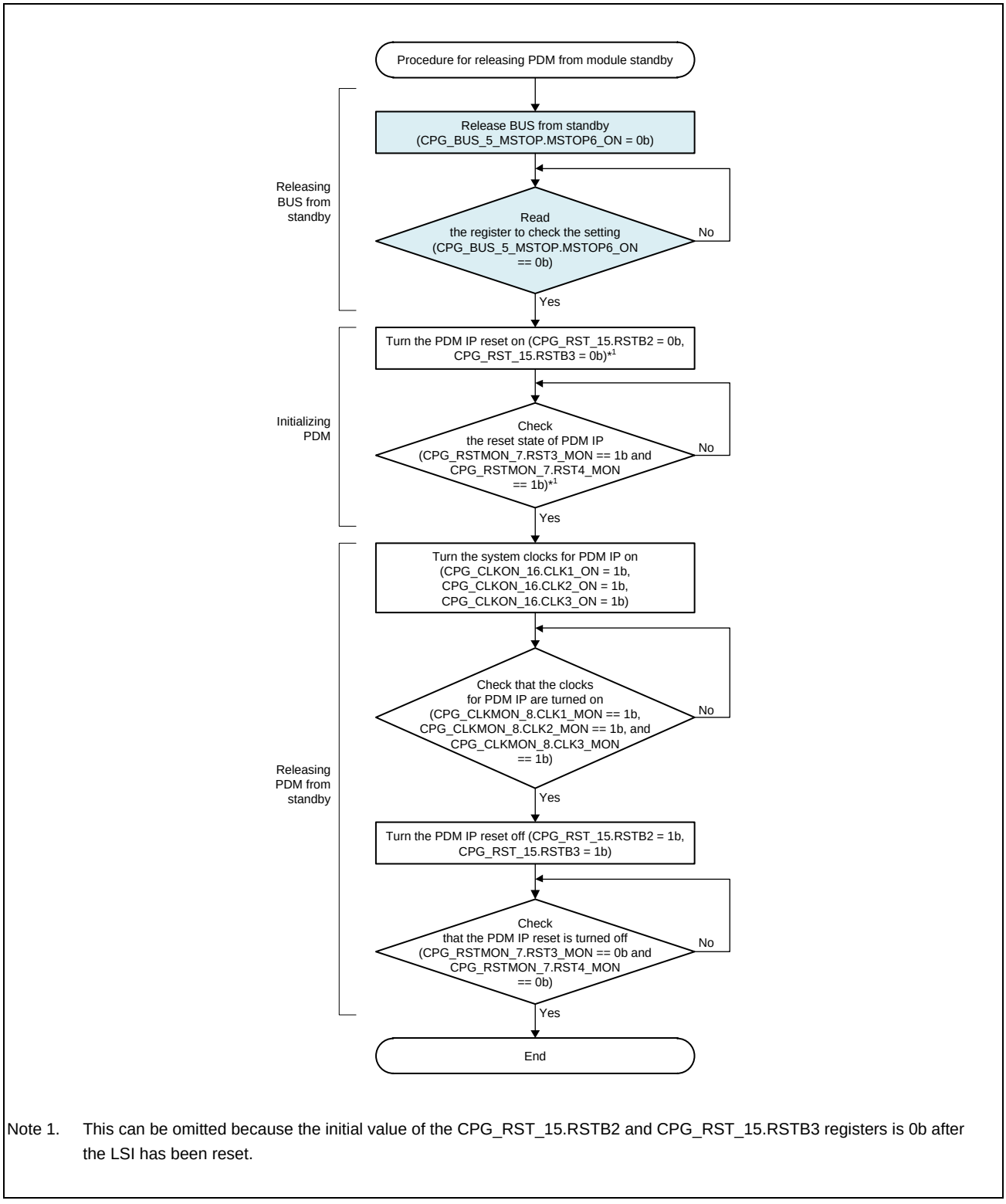


Note 1. This can be omitted because the initial value of the CPG_RST_14.RSTB15 register is 0b after the LSI has been reset.

Figure 8.1-10 Procedure for Releasing SPDIF from Module Standby

8.1.3.7.6 Procedure for Releasing PDM from Module Standby

Figure 8.1-11 shows the procedure for releasing the PDM from module standby, taking PDM IP ch0 as an example.



Note 1. This can be omitted because the initial value of the CPG_RST_15.RSTB2 and CPG_RST_15.RSTB3 registers is 0b after the LSI has been reset.

Figure 8.1-11 Procedure for Releasing PDM from Module Standby

8.1.3.8 SSIU/SCU Setup Procedure Overview

Before setting up the audio related modules, you must set the pin functions and release the IPs to use from module standby.

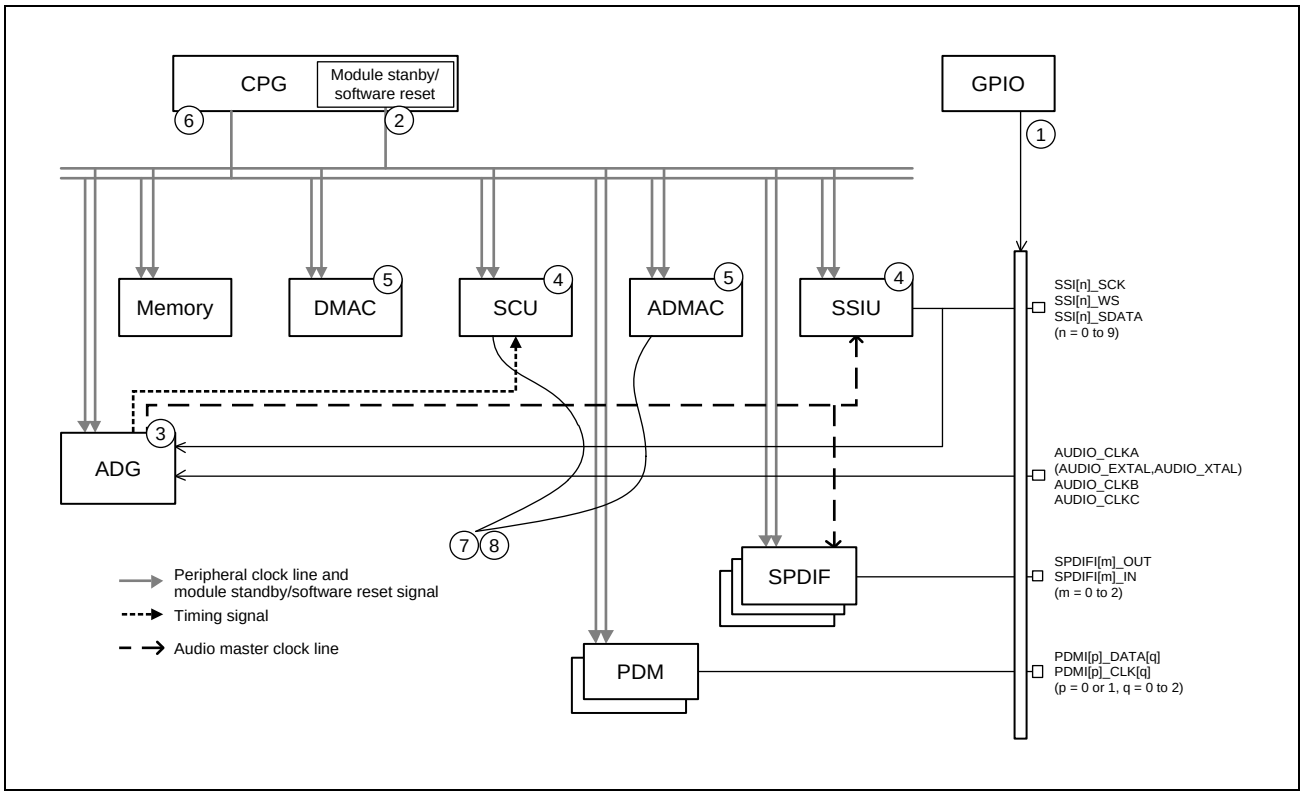


Figure 8.1-12 SSIU/SCU Setup Procedure Diagram

The following is the procedure in outline.

1. Set the pin functions (GPIO).
2. Release module standby (software reset).
3. Set ADG mode.
4. Set SCU/SSIU mode.
5. Set DMAC/ADMAC mode.
6. Turn the audio clocks on (CPG).
   Supply the ADG frequency-divided input audio clocks (on*[1]) and supply the ADG output audio master clocks for the SSIU (on*[2]).
7. Initiate transfer between the DMAC/ADMAC and SCU and SSIU.
8. Stop transfer between the DMAC/ADMAC and SCU and SSIU.
9. Restart transfer between the DMAC/ADMAC and SCU and SSIU.

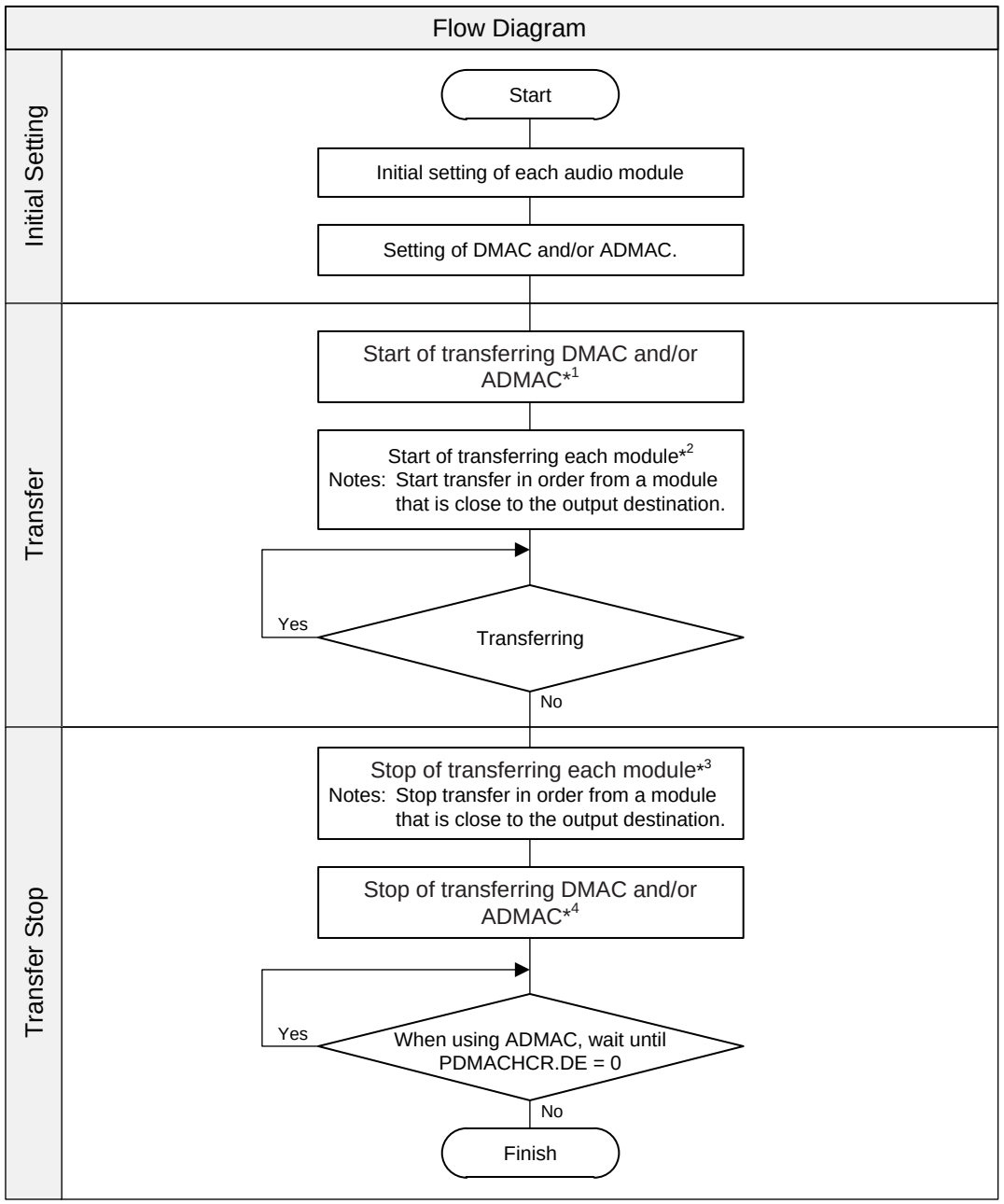
When restarting transfer at step 9, follow the procedure from step 7. You do not need to follow the order of steps 3 to 5 to make initial settings for the audio related modules. Regarding the mode setting registers of the individual IPs, refer to the functional specifications of these IPs.

Note 1. CLK9_ON to CLK13_ON of the CPG_CLKON_15 register. Only the clocks required for operation are turned on.

Note 2. CLK0_ON to CLK9_ON of the CPG_CLKON_22 register. Only the clocks required for operation are turned on.

8.1.3.8.1 Transfer Flow

When data transfer is performed between the audio modules, use the DMAC and ADMAC to perform settings in accord with the transfer flow shown in **Figure 8.1-13**.



Note 1. For DMAC, set CHCTRL_n.SETEN.
For ADMAC, set PDMACHCR.DE.
Set up registers except the above in “setting of DMAC and/or ADMAC”.

(*Continued on next page*)

(*Continued from previous page*)

Note 2. For SSI (except for the case when TDM split mode, SSI0, SSI1, SSI2, or SSI9 are used at the same time), when transmitting (from SSI to the external device), set SSIq/r_CONTROL.start after setting SSICR.EN/DMEN). When receiving (from the external device to SSI), set SSICR.EN/DMEN after setting SSIq/r_CONTROL.start.
For SSI (in the case of TDM split mode) regardless of transmission and reception, set SSIq/r_CONTROL.start after setting SSICR.EN/DMEN.

For SSI (when using SSI0, SSI1, SSI2, or SSI9 at the same time), when transmitting (from SSI to the external device), set SSIq/r_CONTROL.start after setting SSI_CONTROL. When receiving (from the external device to SSI), set SSI_CONTROL after setting SSIq/r_CONTROL.start. (Set SSICR.DMEN at the time of the initial setting.)
For SCU, Set SRCm_CONTROL and CMDn_CONTROL.

Set up registers except the above in “initial setting of each audio module”.

Note 3. For SSI (except for the case when TDM split mode, SSI0, SSI1, SSI2, or SSI9 are used at the same time), when transmitting (from SSI to the external device), clear SSIq/r_CONTROL.start after clearing SSICR.EN/DMEN. When receiving (from the external device to SSI), clear SSICR.EN/DMEN after clearing SSIq/r_CONTROL.start.

For SSI (in the case of TDM split mode), regardless of transmission and reception, clear all SSIq/r_CONTROL.start of the relevant channels to clear SSICR.EN/DMEN.

For SSI (when using SSI0, SSI1, SSI2, or SSI9 at the same time), when transmitting (from SSI to the external device), clear SSIq/r_CONTROL.start after clearing SSICR.DMEN of each module and SSI_CONTROL. When receiving (from the external device to SSI), clear SSICR.DMEN of each module and SSI_CONTROL after clearing SSIq/r_CONTROL.start.

For SCU, clear SRCm_CONTROL and CMDn_CONTROL.

Because transfer is stopped in order from a module that is close to the output destination, overflow may occur in a preceding module. Be careful.

Note 4. For ADMAC, clear PDMACHCR.DE. At this time, other bits must not be changed.
For DMAC, clear CHCTRL_n.SETEN. Clear processing of DMAC can be cleared anywhere in a TransferStop sequence.

Figure 8.1-13 Transfer Flow

8.1.3.9 SPDIF Setup Procedure Overview

Before setting up the audio related modules, you must set the pin functions and release the IPs to use from module standby.

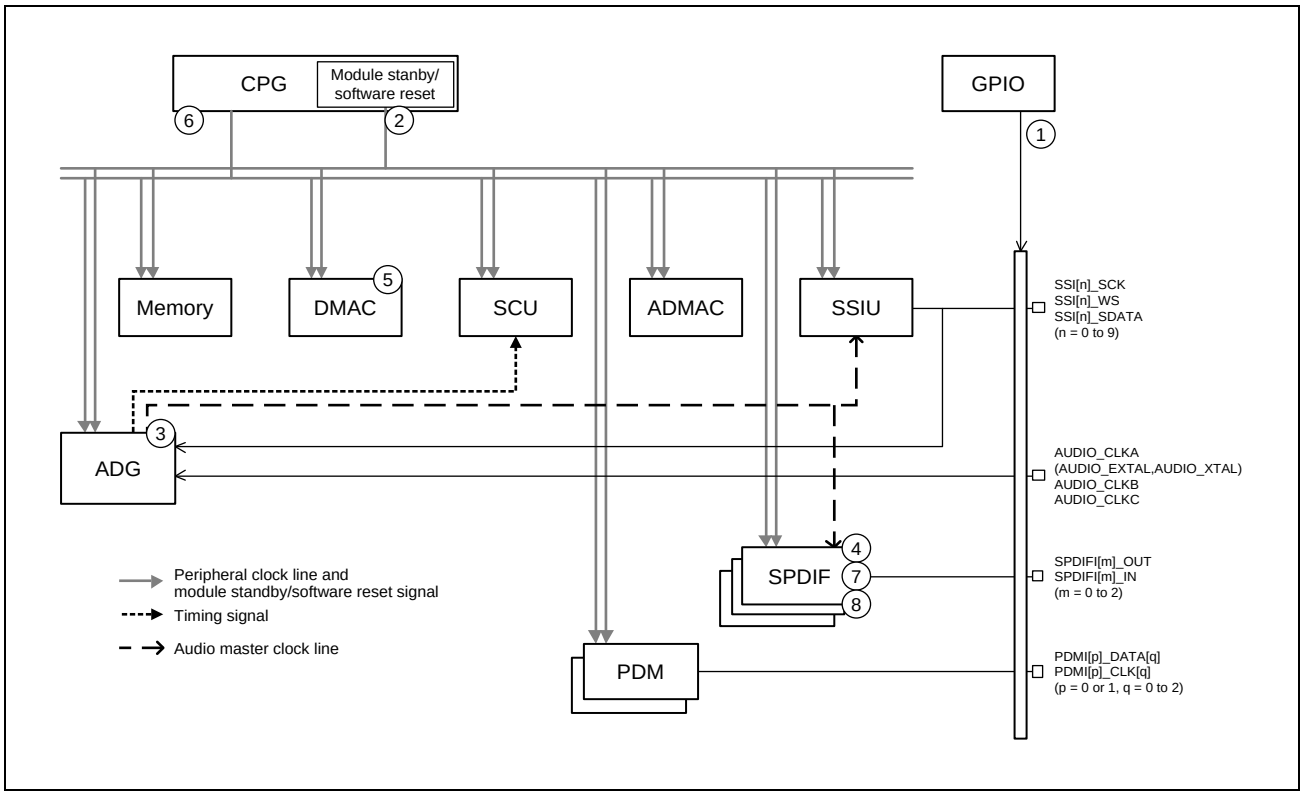


Figure 8.1-14 SPDIF Setup Procedure Diagram

The following is the procedure in outline.

1. Set the pin functions (GPIO).
2. Release module standby (software reset).
3. Set the audio clocks (ADG).
4. Set SPDIF mode.
5. Set DMAC mode.
6. Turn the audio clocks on (CPG).
   Supply the ADG frequency-divided input audio clocks (on*[1]) and supply the ADG output audio master clocks for the SPDIF (on*[2]).
7. Initiate transfer between the DMAC and SPDIF.
8. Stop transfer between the DMAC and SPDIF.
9. Restart transfer between the DMAC and SPDIF.

When restarting transfer at step 9, follow the procedure from step 7. You do not need to follow the order of steps 3 to 5 to make initial settings for the audio related modules. Regarding the mode setting registers of the individual IPs, refer to the functional specifications of these IPs.

Note 1. CLK9_ON to CLK13_ON of the CPG_CLKON_15 register. Only the clocks required for operation are turned on.

Note 2. CLK10_ON to CLK12_ON of the CPG_CLKON_22 register. Only the clocks required for operation are turned on.

8.1.3.10 PDM Setup Procedure Overview

Before setting up the audio related modules, you must set the pin functions and release the IPs to use from module standby.

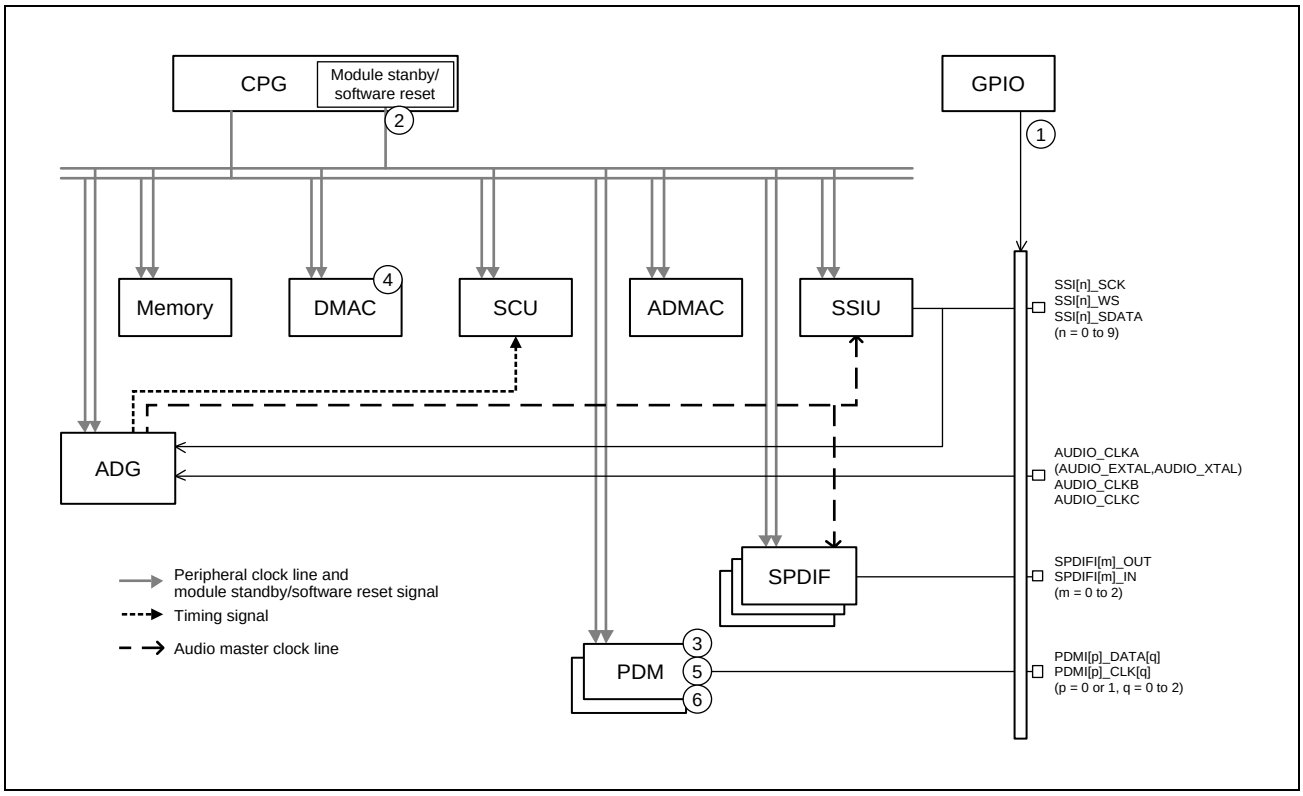


Figure 8.1-15 PDM Setup Procedure Diagram

The following is the procedure in outline.

1. Set the pin functions (GPIO).
2. Release module standby (software reset).
3. Set PDM mode.
4. Set DMAC mode.
5. Initiate transfer between the DMAC and PDM.
6. Stop transfer between the DMAC and PDM.
7. Restart transfer between the DMAC and PDM.

When restarting transfer at step 7, follow the procedure from step 5.

You do not need to follow the order of steps 3 and 4 to make initial settings for the PDM and DMAC

Regarding the mode setting registers of the individual IPs, refer to the functional specifications of these IPs.

8.1.3.11 Module Standby Function

For the register settings to make a transition to or from module standby mode, refer to section CPG.

8.1.3.11.1 Transition to Module Standby Mode

To make a transition to module standby mode by the module standby function, refer to the following transition procedure for each module.

(1) ADG

1. Set the CLK0_ON to CLK9_ON bits in CPG_CLKON_22 register to 0
2. Set the CLK9_ON bit in CPG_CLKON_15 register to 0

Remark The above bits should be set while the module operation has been completed and is placed in the idle state in which the module cannot be activated by external pins or other modules.

(2) ADMAC

1. Set the CLK8_ON bit in CPG_CLKON_15 register to 0

Remark The above bits should be set while the module operation has been completed and is placed in the idle state in which the module cannot be activated by external pins or other modules.

(3) SCU

1. Check the following registers' settings.
   - All bits in the SRCm control register (SRCm_CONTROL) (m = 0 to 9) are cleared to 0 (data transfer stopped).
   - All bits in the CMDn control register (CMDn_CONTROL) (n = 0, 1) are cleared to 0 (data transfer stopped).
   - All bits in the SRCm interrupt enable register 0 (SRCm_INT_ENABLE0) (m = 0 to 6, 9) are cleared to 0 (interrupt disabled).
   - All bits in the SRCn interrupt enable register 0 (SRCn_INT_ENABLE0) (n = 7, 8) are cleared to 0 (interrupt disabled).
2. Set the CLK0_ON to CLK14_ON bits in CPG_CLKON_23 register to 0.

Remark The above bits should be set while the module operation has been completed and is placed in the idle state in which the module cannot be activated by external pins or other modules.

(4) SSIU

1. Check the following registers' settings.

[SSIU]

- All bits in the SSIq control register (SSIq_CONTROL) (q = 0-0, 1-0, 2-0, 3-0, 4-0, or 9-0) are cleared to 0 (data transfer stopped).
- All bits in the SSIr control register (SSIr_CONTROL) (r = 5 to 8) are cleared to 0 (data transfer stopped).
- All bits in the SSIq interrupt enable register (SSIq_INT_ENABLE_MAIN) (q = 0-0, 1-0, 2-0, 3-0, 4-0, or 9-0) are cleared to 0 (interrupt disabled).
- All bits in the SSIr interrupt enable register (SSIr_INT_ENABLE_MAIN) (r = 5 to 8) are cleared to 0 (interrupt disabled).
- All bits in the SSI control register (SSI_CONTROL) are cleared to 0 (data transfer stopped)

[SSI]

- The DMEN and EN bits in the control register (SSICRn) (n = 0 to 9) are cleared to 0 (disabled).
- The IDST bit in the status register (SSISRn) (n = 0 to 9) is set to 1.
- The CONT bit in the WS mode register (SSIWSRn) (n = 0 to 9) is cleared to 0 (disabled).

2. Set the CLK0_ON to CLK10_ON bits in CPG_CLKON_24 register to 0

Remark The above bits should be set while the module operation has been completed and is placed in the idle state in which the module cannot be activated by external pins or other modules.

(5) SPDIF

1. If the module is SPDIF ch0, set the CLK14_ON bit in CPG_CLKON_15 register to 0.
   If the module is SPDIF ch1, set the CLK15_ON bit in CPG_CLKON_15 register to 0.
   If the module is SPDIF ch2, set the CLK0_ON bit in CPG_CLKON_16 register to 0.

Remark The above bits should be set while the module operation has been completed and is placed in the idle state in which the module cannot be activated by external pins or other modules.

(6) PDM

1. If the module is PDM ch0, set the CLK1_ON to CLK3_ON bits in CPG_CLKON_16 register to 0.
   If the module is PDM ch1, set the CLK4_ON to CLK6_ON bits in CPG_CLKON_16 register to 0.

Remark The above bits should be set while the module operation has been completed and is placed in the idle state in which the module cannot be activated by external pins or other modules.

8.1.3.11.2 Releasing and Restarting from Module Standby Mode

After the transition to module standby mode, modules can be released from module standby mode by a power-on reset or the appropriate procedure from the list below.

(1) ADG

1. Supply the clock signal to ADG modules.
2. Set the CLK9_ON bit in CPG_CLKON_15 register to 1.

(2) ADMAC

1. Set the CLK8_ON bit in CPG_CLKON_15 register to 1.

(3) SCU

1. Set the CLK0_ON to CLK14_ON bits in CPG_CLKON_23 register to 1.

(4) SSIU

1. Supply the clock signal to SSIU modules.
2. Set the CLK0_ON to CLK10_ON bits in CPG_CLKON_24 register to 1

(5) SPDIF

1. If the module is SPDIF ch0, set the CLK14_ON bit in CPG_CLKON_15 register to 1.
   If the module is SPDIF ch1, set the CLK15_ON bit in CPG_CLKON_15 register to 1.
   If the module is SPDIF ch2, set the CLK0_ON bit in CPG_CLKON_16 register to 1.

(6) PDM

1. If the module is PDM ch0, set the CLK1_ON to CLK3_ON bits in CPG_CLKON_16 register to 1.
   If the module is PDM ch1, set the CLK4_ON to CLK6_ON bits in CPG_CLKON_16 register to 1.

8.1.4 Usage Notes

8.1.4.1 Notes on Route Switching and Setting Changes

When switching the transfer route, stop the current transfer, specify the new transfer route, and then start transfer. Also, when changing the quantization bit count or the formats, once stop transfer, and then start transfer again after resetting. If this procedure is not followed, correct operation is not guaranteed.

8.1.4.2 Note when SSI0, SSI1, and SSI2, or SSI0, SSI1, SSI2, and SSI9 are Used for Single-Source Audio Data

When SSI0, SSI1, and SSI2, or SSI0, SSI1, SSI2, and SSI9 are used for single-source audio data, specifying eight-bit quantization and six- or eight-channel operation at the same time is not possible.

8.1.4.3 Note on Transfer

When an underflow or an overflow occurs in any audio module, stop the transfer.

8.1.4.4 Note on Usage of the SCU Module

Before starting data transfer, the SCU module needs certain “input data timing” and “output data timing”. When “input data timing” and “output data timing” are not inputted before the transmission starts, it does not operate correctly.

In addition, if the SSI module operates in master mode, and if the WS signal is used as “input/output timing signal for the SRC”, set SSIWSRn.CONT to 1.
(The WS signal inputted into the ADG module is the same signal as the SSI_WS terminal.)

8.1.4.5 Note on Software Reset During Data Transfer

If software reset*[1] is performed during data transfer, abnormal operation may occur. At this time, normal values cannot be guaranteed for the transferred data.

Note 1. RSTB1 to RSTB15 bits of CPG_RST_14 register in the CPG module
RSTB0 to RSTB5 bits of CPG_RST_15 register in the CPG module
SWRST bit of SRCm_SWRSR register (m = 0 to 9) in the SCU module
SWRST bit of CTUn_SWRSR register (n = 00, 01, 02, 03, 10, 11, 12, or 13) in the SCU module
SWRST bit of MIXp_SWRSR register (p = 0 or 1) in the SCU module
SWRST bit of DVCp_SWRSR register (p = 0 or 1) in the SCU module

8.1.4.6 Notes when Issuing SCU/SSIU Transfer Instructions

Follow the order below to issue an instruction to start or stop transfer by SCU and SSIU.

<To start transfer>

Issue the instruction to start DMAC/ADMAC transfer, then issue the transfer start instruction in order from the module on the output side of the path.

<To stop transfer>

Issue the transfer stop instruction in order from the module on the output side of the path, then issue the instruction to stop DMAC/ADMAC transfer.

The following describes two use cases as examples.

(1) In transmission (Memory → DMAC → SCU → ADMAC → SSIU)

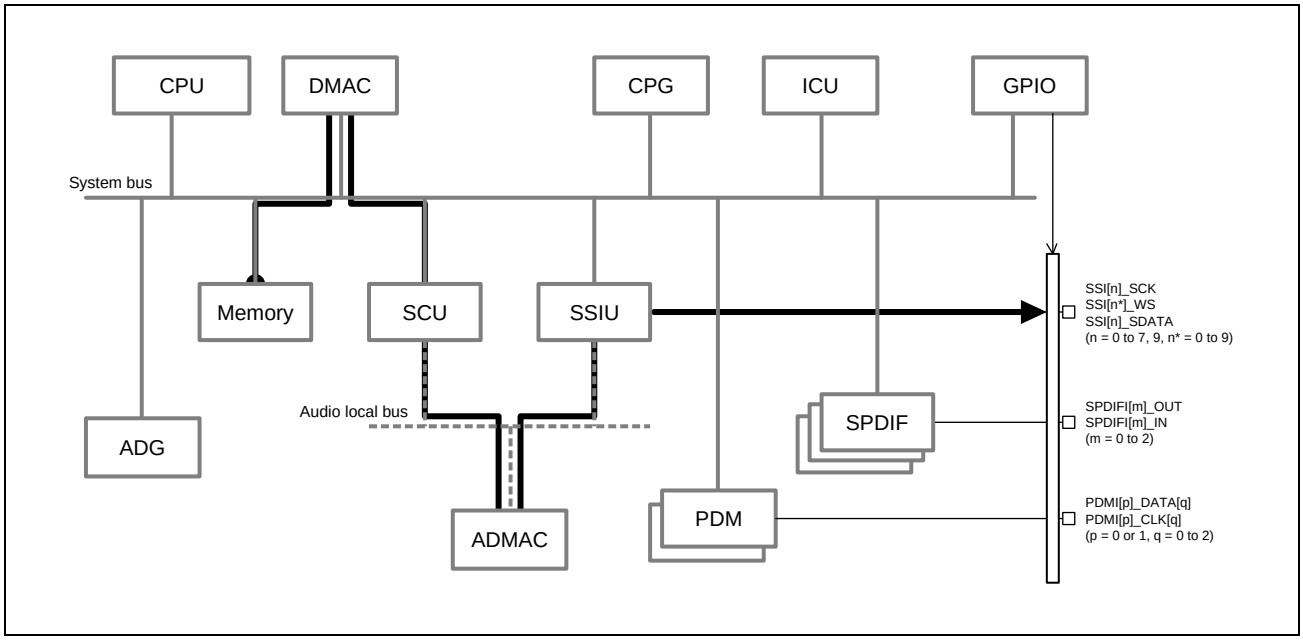


Figure 8.1-16 When Transmission is in the Order “Memory → DMAC → SCU → ADMAC → SSIU”

The order of issuing the transfer start instructions:

1. DMAC/ADMAC transfer start instruction
2. SSIU transfer start instruction
3. SCU transfer start instruction

The order of issuing the transfer stop instructions:

1. SSIU transfer stop instruction
2. SCU transfer stop instruction
3. DMAC/ADMAC transfer stop instruction

(2) In reception (SSIU → ADMAC → SCU → DMAC → Memory)

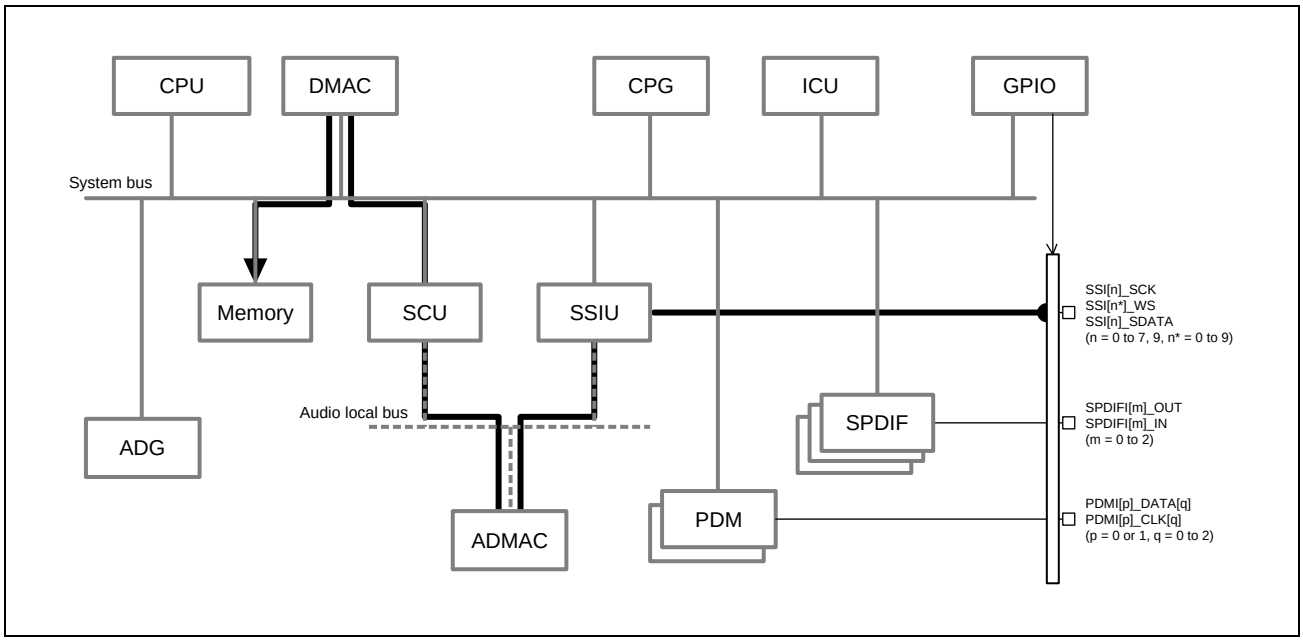


Figure 8.1-17 When Reception is in the Order “SSIU → ADMAC → SCU → DMAC → Memory”

Preparation to initiate transfer:

1. DMAC/ADMAC transfer start instruction
2. SCU transfer start instruction
3. SSIU transfer start instruction

Preparation to stop transfer:

1. SCU transfer stop instruction
2. SSIU transfer stop instruction
3. DMAC/ADMAC transfer stop instruction

SECTION 8 AUDIO

8.2 Sampling Rate Converter Unit (SCU)

This manual is a simplified version. For more information, refer to the User's Manual Additional Document.

8.2.1 Overview

The SCU has ten SRC modules (six for high-sound-quality type; four for general-sound-quality type) that are useful for synchronization of asynchronous data, which is necessary for data transfer with external memory or external devices. It also provides the functions to change the number of channels, perform mixing, and control the volume.

Remark When the SRCs are not in use, configure the system with the same clock sources for AUDIO CLOCK that is input to ADG or devices for I2S that are connected to external modules, and all thus operate at the same sampling frequency.

8.2.1.1 Features

Sampling Rate Converter (SRC)

- Asynchronous sampling rate conversion is available
- Supports resolutions up to 24 bits
- High-sound-quality type (THD + N*[1] is −132 dB) and general-sound-quality type (THD + N*[1] is −96 dB)
- Automatically generates antialiasing filter coefficients
- Four modules support one, two, four, six, or eight channels, and six modules support one or two channels.

Note 1. Total harmonic distortion plus noise

Channel Transfer Unit (CTU)

- Downmixing and splitter functions
  - Conversion of eight input channels into two output channels
  - Conversion of six input channels into two output channels
  - Conversion of two input channels into four sets of two output channels
  - Conversion of one input channel into eight sets of one output channel
  - No conversion

Mixer (MIX)

- Mixing (adds) two to four sources into one
- Ratio for adding sources is selectable
- Ratio is dynamically changeable
- Mixing with volume ramp is available (ramp period is selectable)

Digital Volume and Mute Function (DVC)

- Volume control function including digital volume, volume ramp, and zero-crossing mute
- The digital volume function is specified by a 24-bit fixed-point value within the range from 0 to 8 times (mute or -120 to 18 dB)
- The volume ramp function can be used for soft mute, fade-in, fade-out, or desired volume adjustment
- The volume ramp period can be changed within the sampling range from the 0th to 23rd power of 2
- The zero-crossing mute function silences the sound at the zero-crossing point of the audio data

The CTU, MIX, and DVC functional blocks are collectively called “CMD”.

SECTION 8 AUDIO

8.3 Audio Clock Generator (ADG)

This manual is a simplified version. For more information, refer to the User's Manual Additional Document.

8.3.1 Overview

The audio clock generator (ADG) selects and supplies the necessary clock for the SSIU, SCU, and SPDIF modules. It also divides the frequency of the selected clock and sends it outside the chip.

8.3.1.1 Features

- Selects the clock signal from the AUDIO_CLKA, AUDIO_CLKB, or AUDIO_CLKC pin or the internal clock and supplies it to the SSIU, SCU, and SPDIF modules.
- The frequency of the clock signals from the AUDIO_CLKA, AUDIO_CLKB, and AUDIO_CLKC pins and the internal clock can be divided before use.
- The divided clock can be output through the AUDIO_CLKOUT pin.

SECTION 8 AUDIO

8.4 Audio-DMAC-Peripheral-Peripheral (ADMAC)

This manual is a simplified version. For more information, refer to the User's Manual Additional Document.

8.4.1 Overview

The Audio-DMAC-Peripheral-Peripheral module controls data transfer between the audio modules (SSIU, SCU) connected to the audio local bus.

8.4.1.1 Features

- Number of channels: 29 channels
- Data transfer size: Longword (4 bytes)
- Addressing mode: Dual addressing; fixed access size
- Transfer count: Not programmable
- Interrupt processing: None

SECTION 8 AUDIO

8.5 Serial Sound Interface Unit (SSIU)

This manual is a simplified version. For more information, refer to the User's Manual Additional Document.

8.5.1 Overview

The SSIU, incorporating ten SSI modules, allows independent operation of SSI modules, operation of multiple SSI modules sharing the same serial clock, and connection or split of multi-channel data.

The serial sound interface (hereinafter referred to as the "SSI") is a transceiver module designed to send or receive audio data interfacing with a variety of devices offering I2S format. It also supports multi-channel mode in addition to other common formats.

Remark Configure the system so that connected external devices can operate on the same clock source.

8.5.1.1 Features

- Incorporates ten SSI modules.
- Supports 6 channels/1 sound source with three SSI modules (SSI0, SSI1, and SSI2).
- Supports 8 channels/1 sound source with four SSI modules (SSI0, SSI1, SSI2, and SSI9).
- Operation of multiple modules on the same serial clock (SSI0/SSI1/SSI2/SSI3/SSI9, SSI3/SSI4/SSI9, SSI5/SSI6, or SSI7/SSI8).
- TDM format (Basic Configuration) corresponds to 4, 6, or 8-channel data.
- Handles 8-channel data on the serial bus and 6-channel data in the SoC (TDM extend mode).
- Handles 6-channel data on the serial bus and 8-channel data in the SoC (TDM extend mode).
- Connects monaural or stereo data to output TDM format data (TDM split mode).
- Splits TDM format input data into monaural or stereo data (TDM split mode).
- Connects stereo or multi data to output TDM format data (TDM ex-split mode).
- Splits TDM format input data into stereo or multi data (TDM ex-split mode).
- The frequency range of SCK signal is from 297.3 kHz to 12.5 MHz at master mode, and from 297.3 kHz to 12.5 MHz at slave mode.

The SSI has the following features:

- Number of channels: Maximum of four (when a multi-channel format is specified).
- The SSI module can serve as both a transmitter and a receiver.
- Asynchronous transfer takes place between the data buffer and the shift register.
- Only the MSB first data alignment is supported.
- A value as the dividing ratio for the clock used by the serial bus interface is selectable.

- Controlling of data transmission or reception with DMAC or interrupt requests is possible.
- TDM format is supported.
- The frequency range of SCK signal is from 297.3 kHz to 12.5 MHz at master mode, and from 297.3 kHz to 12.5 MHz at slave mode.
- The WS continue function by which operation can be performed without stopping the WS signal is supported.
- Monaural mode (8 bits or 16 bits) is supported.
- In monaural mode, the pulse width of the WS signal can be changed (short or long frame).
- If the sampling clock frequency is switched during transfer, the CPU is notified of an interrupt (a function to detect the frequency switching).
- Operating mode: Non-compressed mode (not support for compressed mode).
- Supports versatile serial audio formats (I2S/left justified/right justified).
- Supports master/slave functions.
- Programmable word clock, bit clock generation functions.

SECTION 8 AUDIO

8.6 SPDIF Interface (SPDIF)

This section describes the SPDIF interface (SPDIF).

8.6.1 Overview

The SPDIF complies with the IEC 60958 standard. For the features and configuration of the SPDIF, see the following subsections.

8.6.1.1 Features

The features of the SPDIF are described below.

- Compliant with the IEC 60958 standard (stereo and consumer use modes only)
- Sampling frequency: 32 kHz, 44.1 kHz, 48 kHz
- Audio word size: 16 to 24 bits/sample
- Biphase mark encoding
- Double buffer for data
- Serial data with parity
- Capable of simultaneous transmission and reception
- The receiver autodetects the IEC 61937 compressed mode data.
- DMAC support

The following restrictions apply to the SPDIF macro.

— Only an external-facing SPDIF device that can support 512 fs can be connected.

— Set the externally input oversampling clock (pa_audiox_a) frequency to 512 fs.

8.6.1.2 SPDIF Frame Format

A frame for the SPDIF consists of two sub-frames (channels 1 and 2), each containing a 4-bit preamble, up to 24 bits of audio data, V flag, user bit, channel status bit, and even parity bit. **Figure 8.6-1** shows the format of a sub-frame. The SPDIF performs modulation (channel coding) based on the biphase mark method, which minimizes the DC component on the transmission line for this format.

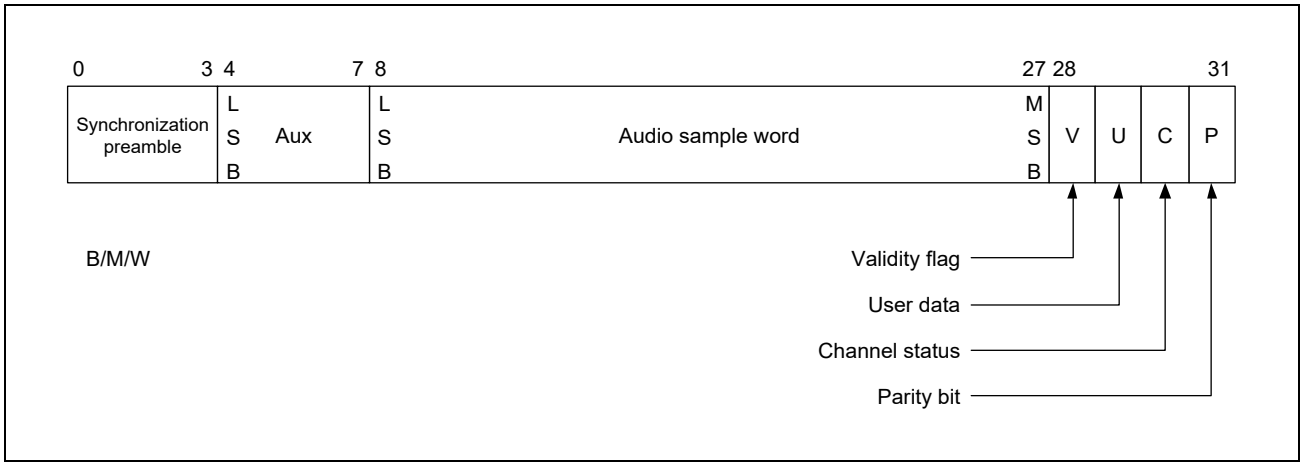


Figure 8.6-1 Sub-Frame Format

Figure 8.6-2 shows the format of a block, which consists of consecutive 192 frames. One block extends from the start frame (preamble B) to frame 191, which is the 192nd frame, and each sub-frame is identified by a preamble. 384 sub-frames are in one block and the frames consist of three types of sub-frames: sub-frame 0 indicating the start of a block, sub-frame 1 (generally, left channel), and sub-frame 2 (generally, right channel). Generally, the music data that are transmitted and received by the SPDIF are consecutive, so the data are transferred in consecutive blocks.

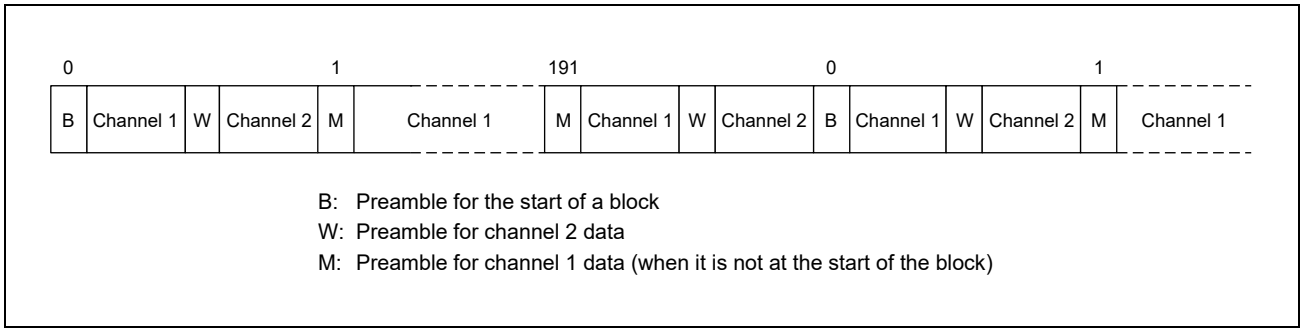


Figure 8.6-2 Block Format

Table 8.6-1 shows the values of the SPDIF preambles (in binary). The state of the preceding symbol (parity bit) determines the logical sense of each preamble.

Table 8.6-1 Preamble Values

Preamble	State of Preceding Symbol = 0	State of Preceding Symbol = 1
B	11101000	00010111
M	11100010	00011101
W	11100100	00011011

***Remark*:** As shown in **Figure 8.6-1**, the even parity bit in time slot 31 of the sub-frame determines the preamble type in one transmission. Therefore, one of the set states to be normally sent via the SPDIF is selected. However, in IEC 60958 and so on, decoding of both types is defined in consideration of connection of the preamble with its polarity reversed. The SPDIF decodes preambles as listed in **Table 8.6-1**.

The channel status information is encoded in one bit of each sub-frame. Therefore, the amount of channel status information for one block is 192 bits for each of sub-frames 1 and 2. For the format of the channel status, refer to the IEC 60958 standard.

8.6.1.3 About Sampling Frequency Selection Method

The SPDIF macro supports three sampling frequencies: 1) 32 kHz, 2) 44.1 kHz, and 3) 48 kHz. To support each sampling frequency, the oversampling clock at a frequency of 512 fs must be entered through the input terminal for the oversampling clock (pa_audiox_a) of the SPDIF macro. The pa_audio_a terminal input clock frequencies for each of the sampling frequencies are listed below.

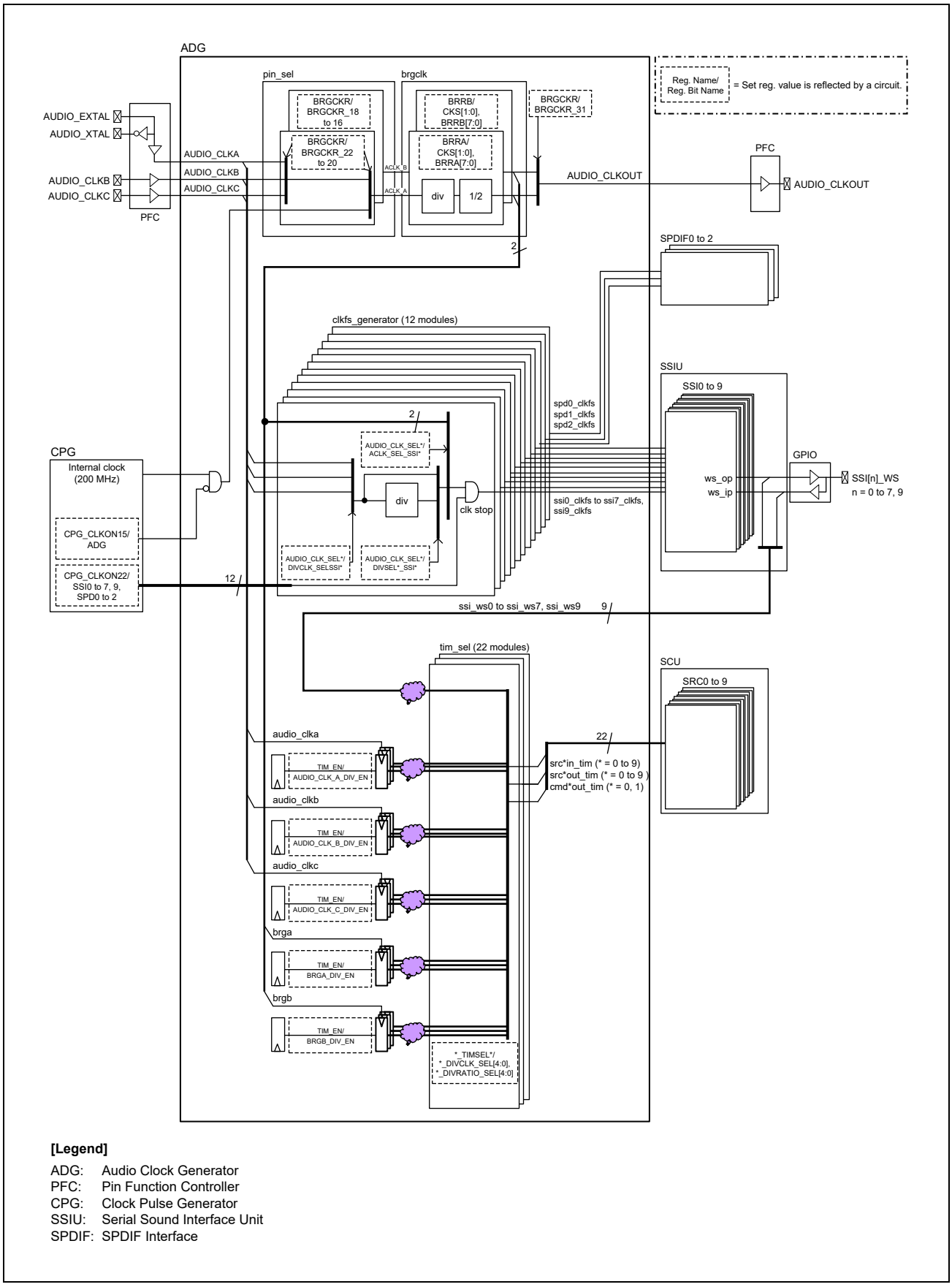
Table 8.6-2 Sampling Frequency versus pa_audio_a Terminal Input Clock Frequency

Sampling Frequency	pa_audio_a Terminal Input Clock Frequency (512 fs)
32 kHz	16.384 MHz
44.1 kHz	22.5792 MHz
48 kHz	24.576 MHz

Clock input to the SPDIF pa_audiox_a terminal is possible from any of the following three externally input pins.

1) AUDIO_EXTAL/AUDIO_XTAL
2) AUDIO_CLKB
3) AUDIO_CLKC

For selection of the external input, see the sections PFC and ADG.



[Legend]

ADG: Audio Clock Generator
PFC: Pin Function Controller
CPG: Clock Pulse Generator
SSIU: Serial Sound Interface Unit
SPDIF: SPDIF Interface

Figure 8.6-3 SPDIF External Clock Input Configuration

8.6.1.4 Block Diagram

The figure below is a block diagram of the SPDIF.

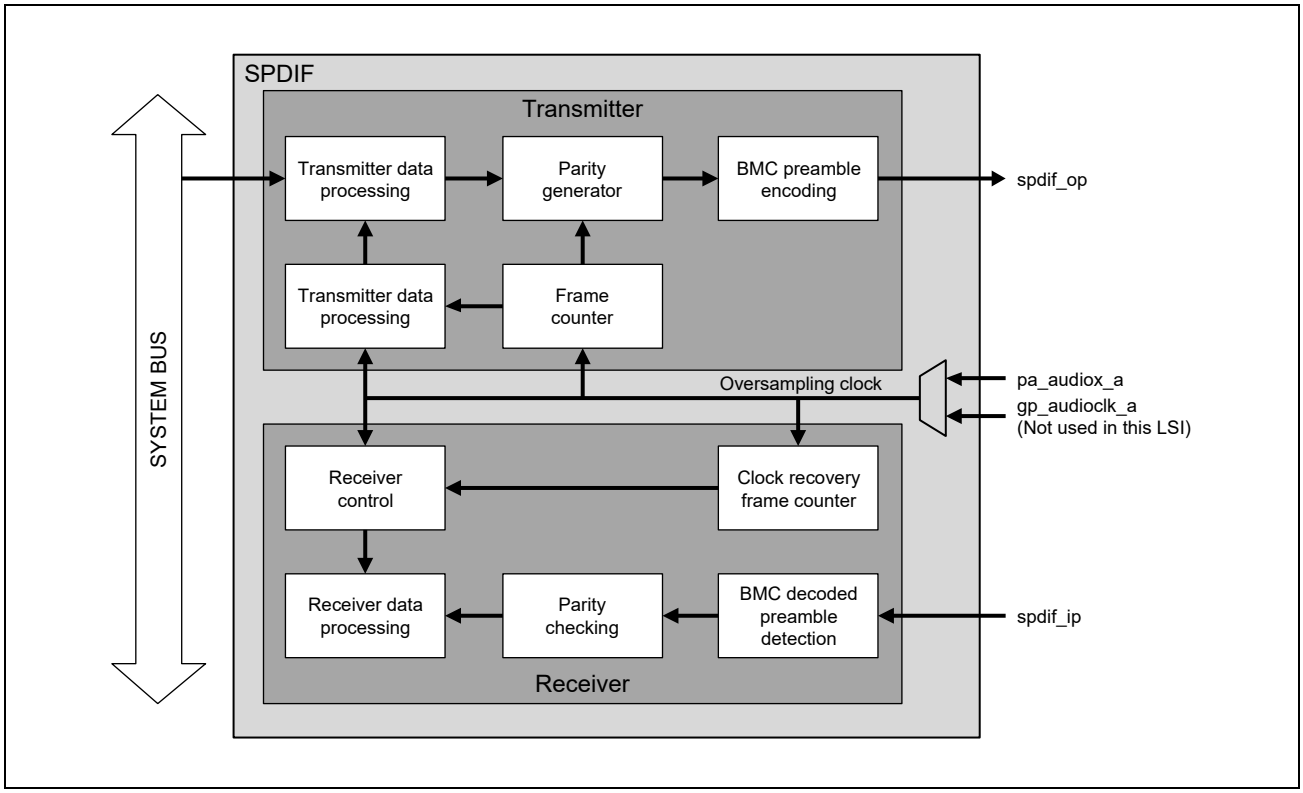


Figure 8.6-4 Block Diagram of SPDIF

8.6.1.5 External Pins

Table 8.6-3 lists the specifications of the SPDIF external pins.

Table 8.6-3 Pin Configuration

Pin Name	Input/ Output	Function
SPDIFn_OUT	Output	SPDIF signal transmission pin
SPDIFn_IN	Input	SPDIF signal reception pin

Note: n = 0 to 2

8.6.1.6 Connected Units

Table 8.6-4 lists the units that are connected to the SPDIF.

Table 8.6-4 Connected Units

Unit Name	Connected Unit Name	Function of Connected Unit
SPDIF	CPG	Clock pulse generator (CPG)
	SYSTEM BUS	Internal bus
	ICU	Interrupt control unit (ICU_TOP)
	PFC	Pin Function Controller (PFC)
	ADG	Audio clock generator (ADG)

8.6.2 Registers

The base addresses for the respective channels are as follows.

Table 8.6-5 Register Base Addresses

Base Register Name	Unit Name	Base Address
<SPDIF0_base>	SPDIF0	0_1440_2400h (5440_2400h*1, 4440_2400h*2)
<SPDIF1_base>	SPDIF1	0_1440_2800h (5440_2800h*1, 4440_2800h*2)
<SPDIF2_base>	SPDIF2	0_1440_2C00h (5440_2C00h*1, 4440_2C00h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

8.6.2.1 List of Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Transmitter Channel 1 Audio Register	SPDIFm_TLCA	0000_0000h	0000h	32
Transmitter Channel 2 Audio Register	SPDIFm_TRCA	0000_0000h	0004h	32
Transmitter Channel 1 Status Register	SPDIFm_TLCS	0000_0000h	0008h	32
Transmitter Channel 2 Status Register	SPDIFm_TRCS	0000_0000h	000Ch	32
Transmitter User Data Register	SPDIFm_TUI	0000_0000h	0010h	32
Receiver Channel 1 Audio Register	SPDIFm_RLCA	0000_0000h	0014h	32
Receiver Channel 2 Audio Register	SPDIFm_RRCA	0000_0000h	0018h	32
Receiver Channel 1 Status Register	SPDIFm_RLCS	0000_0000h	001Ch	32
Receiver Channel 2 Status Register	SPDIFm_RRCS	0000_0000h	0020h	32
Receiver User Data Register	SPDIFm_RUI	0000_0000h	0024h	32
Control Register	SPDIFm_CTRL	0000_0000h	0028h	32
Status Register	SPDIFm_STAT	0000_C000h	002Ch	32
Transmitter DMA Audio Data Register	SPDIFm_TDAD	0000_0000h	0030h	32
Receiver DMA Audio Data Register	SPDIFm_RDAD	0000_0000h	0034h	32

Always access the above registers as 32-bit units because they are all longword registers.

Always write 0 to a bit for which the writing of 0 is stated in the register descriptions on the following pages (when the bit is writable). 0 is always read from the bit when such a bit is read (when the bit is readable).

8.6.3 Register Description

The prefix (SPDIFm_) of the register names is omitted in this and subsequent sections.

8.6.3.1 Transmitter Channel 1 Audio Register (SPDIFm_TLCA)

This register is for writing the channel 1 audio PCM data for the transmitter.

Access Size : 32 bits
Address : <SPDIFm_base> + 0000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	PCM_DATA[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCM_DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	W	Reserved A register that can be written. Reading is invalid. The write value should always be 0b.
23 to 0	PCM_DATA [23:0]	0h	W	The read value is undefined Audio PCM data PCM-encoded audio data (stored flush against the LSB side)

8.6.3.2 Transmitter Channel 2 Audio Register (SPDIFm_TRCA)

This register is for writing the channel 2 audio PCM data for the transmitter.

Access Size : 32 bits
Address : <SPDIFm_base> + 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	PCM_DATA[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCM_DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	W	Reserved A register that can be written. Reading is invalid. The write value should always be 0b.
23 to 0	PCM_DATA [23:0]	0h	W	The read value is undefined Audio PCM data PCM-encoded audio data (stored flush against the LSB side)

8.6.3.3 Transmitter Channel 1 Status Register (SPDIFm_TLCS)

The channel status information to be transmitted is stored in a 30-bit register. The channel status information for each channel consists of 192 bits per one frame. However, the essential data are only the 30 bits to be set in the register below, so 0 is transmitted after transmission of the first 30 bits.

Refer to IEC 60958 regarding the values to be set in the fields.

Access Size : 32 bits
Address : <SPDIFm_base> + 0008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	CLAC[1:0]		FS[3:0]				CHNO[3:0]				SRCNO[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATCD[7:0]								-	-	CTL[4:0]					-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	W	Reserved A register that can be written. Reading is invalid. The write value should always be 0b.
29, 28	CLAC[1:0]	0h	W	The read value is undefined Clock accuracy 00b: Level 2 01b: Level 1 10b: Level 3 11b: Reserved
27 to 24	FS[3:0]	0h	W	The read value is undefined Sampling frequency (fs) 0000b: 44.1 kHz 0100b: 48 kHz 1100b: 32 kHz
23 to 20	CHNO[3:0]	0h	W	The read value is undefined Channel number 0000b: Unspecified 0001b: A (left channel) 0010b: B (right channel) 0011b: C
19 to 16	SRCNO[3:0]	0h	W	The read value is undefined Transmission source number 0000b: Unspecified 0001b: 1 0010b: 2 0011b: 3
15 to 8	CATCD[7:0]	0h	W	The read value is undefined Category code (example) 0000_0000b: 2-channel general-purpose format 0000_0001b: 2-channel compact disk (IEC 908) 0000_0010b: 2-channel PCM encoder/decoder 0000_0011b: 2-channel digital audio tape recorder
7, 6	-	All 0	W	Reserved A register that can be written. Reading is invalid. The write value should always be 0b.
5 to 1	CTL[4:0]	0h	W	The read value is undefined Control The control bits are copied from the transmission source (refer to the IEC 60958 standard).
0	-	0h	W	Reserved A register that can be written. Reading is invalid. The write value should always be 0b.

8.6.3.4 Transmitter Channel 2 Status Register (SPDIFm_TRCS)

The channel status information to be transmitted is stored in a 30-bit register. The channel status information for each channel consists of 192 bits per one frame. However, the essential data are only the 30 bits to be set in the register below, so 0 is transmitted after transmission of the first 30 bits.

Refer to IEC 60958 regarding the values to be set in the fields.

Access Size : 32 bits
Address : <SPDIFm_base> + 000Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	CLAC[1:0]		FS[3:0]				CHNO[3:0]				SRCNO[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATCD[7:0]								-	-	CTL[4:0]					-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	W	Reserved A register that can be written. Reading is invalid. The write value should always be 0b.
29, 28	CLAC[1:0]	0h	W	The read value is undefined Clock accuracy 00b: Level 2 01b: Level 1 10b: Level 3 11b: Reserved
27 to 24	FS[3:0]	0h	W	The read value is undefined Sampling frequency (fs) 0000b: 44.1 kHz 0100b: 48 kHz 1100b: 32 kHz
23 to 20	CHNO[3:0]	0h	W	The read value is undefined Channel number 0000b: Unspecified 0001b: A (left channel) 0010b: B (right channel) 0011b: C
19 to 16	SRCNO[3:0]	0h	W	The read value is undefined Transmission source number 0000b: Unspecified 0001b: 1 0010b: 2 0011b: 3
15 to 8	CATCD[7:0]	0h	W	The read value is undefined Category code (example) 0000_0000b: 2-channel general-purpose format 0000_0001b: 2-channel compact disk (IEC 908) 0000_0010b: 2-channel PCM encoder/decoder 0000_0011b: 2-channel digital audio tape recorder
7, 6	-	All 0	W	Reserved A register that can be written. Reading is invalid. The write value should always be 0b.
5 to 1	CTL[4:0]	0h	W	The read value is undefined Control The control bits are copied from the transmission source (refer to the IEC 60958 standard).
0	-	0h	W	Reserved A register that can be written. Reading is invalid. The write value should always be 0b.

8.6.3.5 Transmitter User Data Register (SPDIFm_TUI)

This register is for writing the U-bit data of the sub-frame. Update the U-bit data every 16 frames because the data are transmitted in order of sub-frame 1 and then sub-frame 2. For the contents of the user byte, refer to the standard for the user information that corresponds to the SPDIF device to be used. The user bits to be transmitted are set in order from the LSB side.

Access Size : 32 bits
Address : <SPDIFm_base> + 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BYTE4[7:0]								BYTE3[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BYTE2[7:0]								BYTE1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	BYTE4[7:0]	0h	W	The read value is undefined The U-bit information is stored.
23 to 16	BYTE3[7:0]	0h	W	The read value is undefined The U-bit information is stored.
15 to 8	BYTE2[7:0]	0h	W	The read value is undefined The U-bit information is stored.
7 to 0	BYTE1[7:0]	0h	W	The read value is undefined The U-bit information is stored.

8.6.3.6 Receiver Channel 1 Audio Register (SPDIFm_RLCA)

This register is for reading the channel 1 audio PCM data for the receiver.

Access Size : 32 bits
Address : <SPDIFm_base> + 0014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	PCM_DATA[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCM_DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 0	PCM_DATA [23:0]	0h	R	Audio PCM data PCM-encoded audio data (stored flush against the LSB side)

8.6.3.7 Receiver Channel 2 Audio Register (SPDIFm_RRCA)

This register is for reading the channel 2 audio PCM data for the receiver.

Access Size : 32 bits
Address : <SPDIFm_base> + 0018h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	PCM_DATA[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCM_DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 0	PCM_DATA [23:0]	0h	R	Audio PCM data PCM-encoded audio data (stored flush against the LSB side)

8.6.3.8 Receiver Channel 1 Status Register (SPDIFm_RLCS)

The channel status is stored from the start of the block to received sub-frame 1 in order from the LSB bit of this register. For the contents of the channel status, refer to IEC 60958.

Access Size : 32 bits
Address : <SPDIFm_base> + 001Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	CLAC[1:0]		FS[3:0]				CHNO[3:0]				SRCNO[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATCD[7:0]								-	-	CTL[4:0]					-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29, 28	CLAC[1:0]	0h	R	Clock accuracy 00b: Level 2 01b: Level 1 10b: Level 3 11b: Reserved
27 to 24	FS[3:0]	0h	R	Sampling frequency (fs) 0000b: 44.1 kHz 0100b: 48 kHz 1100b: 32 kHz
23 to 20	CHNO[3:0]	0h	R	Channel number 0000b: Unspecified 0001b: A (left channel) 0010b: B (right channel) 0011b: C
19 to 16	SRCNO[3:0]	0h	R	Transmission source number 0000b: Unspecified 0001b: 1 0010b: 2 0011b: 3
15 to 8	CATCD[7:0]	0h	R	Category code (example) 0000_0000b: 2-channel general-purpose format 0000_0001b: 2-channel compact disk (IEC 908) 0000_0010b: 2-channel PCM encoder/decoder 0000_0011b: 2-channel digital audio tape recorder
7, 6	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 1	CTL[4:0]	0h	R	Control The control bits are copied from the transmission source (refer to the IEC 60958 standard).
0	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

8.6.3.9 Receiver Channel 2 Status Register (SPDIFm_RRCS)

The channel status is stored from the start of the block to received sub-frame 2 in order from the LSB bit of the register. For the contents of the channel status, refer to IEC 60958.

Access Size : 32 bits
Address : <SPDIFm_base> + 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	CLAC[1:0]		FS[3:0]				CHNO[3:0]				SRCNO[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATCD[7:0]								-	-	CTL[4:0]					-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29, 28	CLAC[1:0]	0h	R	Clock accuracy 00b: Level 2 01b: Level 1 10b: Level 3 11b: Reserved
27 to 24	FS[3:0]	0h	R	Sampling frequency (fs) 0000b: 44.1 kHz 0100b: 48 kHz 1100b: 32 kHz
23 to 20	CHNO[3:0]	0h	R	Channel number 0000b: Unspecified 0001b: A (left channel) 0010b: B (right channel) 0011b: C
19 to 16	SRCNO[3:0]	0h	R	Transmission source number 0000b: Unspecified 0001b: 1 0010b: 2 0011b: 3
15 to 8	CATCD[7:0]	0h	R	Category code (example) 0000_0000b: 2-channel general-purpose format 0000_0001b: 2-channel compact disk (IEC 908) 0000_0010b: 2-channel PCM encoder/decoder 0000_0011b: 2-channel digital audio tape recorder
7, 6	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 1	CTL[4:0]	0h	R	Control The control bits are copied from the transmission source (refer to the IEC 60958 standard).
0	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

8.6.3.10 Receiver User Data Register (SPDIFm_RUI)

This register stores the U-bit data of the sub-frame. Read the U-bit data every 16 frames because the data are stored in order of sub-frame 1 and then sub-frame 2 from the LSB side. For the contents of the user byte, refer to the standard that corresponds to the device to be used.

Access Size : 32 bits
Address : <SPDIFm_base> + 0024h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BYTE4[7:0]								BYTE3[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BYTE2[7:0]								BYTE1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	BYTE4[7:0]	0h	R	The U-bit information is stored.
23 to 16	BYTE3[7:0]	0h	R	The U-bit information is stored.
15 to 8	BYTE2[7:0]	0h	R	The U-bit information is stored.
7 to 0	BYTE1[7:0]	0h	R	The U-bit information is stored.

8.6.3.11 Control Register (SPDIFm_CTRL)

This register controls the SPDIF unit. All control is applied in accord with the settings of this register.

Access Size : 32 bits
Address : <SPDIFm_base> + 0028h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	RASS[1:0]		TASS[1:0]		RDE	TDE	NCSI	AOS	RME	TME
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	REIE	TEIE	UBOI	UBUI	CREI	PAEI	PREI	CSEI	ABOI	ABUI	RUII	TUII	RCSI	RCBI	TCSI	TCBI
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
28	-	0h	RW	Reserved. The write value should always be 0.
27	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
26	-	0h	RW	Reserved. The write value should always be 0.
25, 24	RASS[1:0]	0h	RW	Receiver audio sample bit size (audio word size) Specify the number of the receiver audio sample bits (16, 20, or 24 bits) for data alignment. 00b: 16-bit sample 01b: 20-bit sample 10b: 24-bit sample 11b: Reserved
23, 22	TASS[1:0]	0h	RW	Transmitter audio sample bit size (audio word size) Specify the number of the transmitter audio sample bits (16, 20, or 24 bits) for data alignment. 00b: 16-bit sample 01b: 20-bit sample 10b: 24-bit sample 11b: Reserved
21	RDE	0h	RW	Receiver DMA enable Enables receiver DMA requests. 0b: Disables receiver DMA. 1b: Enables receiver DMA.
20	TDE	0h	RW	Transmitter DMA enable Enables transmitter DMA requests. 0b: Disables transmitter DMA. 1b: Enables transmitter DMA.
19	NCSI	0h	RW	New channel status information Set this bit to 1 when new channel status information that requires correction is present in the transmitter. 0b: No new channel status information is present. 1b: Channel status information is present.
18	AOS	0h	RW	Audio only sample Set this bit to 0 when user information is present in the user data register. When this bit is set to 1, the user bits are set to all 0s. 0b: User information is present. 1b: No user information is present.
17	RME	0h	RW	Receiver enable Enables the receiver. 0b: The receiver is disabled. 1b: The receiver is enabled.

Bit	Bit Name	Initial Value	R/W	Description
16	TME	0h	RW	Transmitter enable Enables the transmitter. 0b: The transmitter is disabled. 1b: The transmitter is enabled.
15	REIE	0h	RW	Receiver error interrupt enable Clearing this bit masks all error interrupts from the receiver. Setting this bit to 1 enables all receiver error interrupts. 0b: Disables receiver error interrupts. 1b: Enables receiver error interrupts.
14	TEIE	0h	RW	Transmitter error interrupt enable Clearing this bit masks all error interrupts from the transmitter. Setting this bit to 1 enables all transmitter error interrupts. 0b: Disables transmitter error interrupts. 1b: Enables transmitter error interrupts.
13	UBOI	0h	RW	User buffer overrun interrupt enable Enables the user buffer overrun interrupt. 0b: Disables the user buffer overrun interrupt. 1b: Enables the user buffer overrun interrupt.
12	UBUI	0h	RW	User buffer underrun interrupt enable Enables the user buffer underrun interrupt. 0b: Disables the user buffer underrun interrupt. 1b: Enables the user buffer underrun interrupt.
11	CREI	0h	RW	Clock recovery error interrupt enable Enables the clock recovery error interrupt. 0b: Disables the clock recovery error interrupt. 1b: Enables the clock recovery error interrupt.
10	PAEI	0h	RW	Parity error interrupt enable Enables the parity checking error interrupt. 0b: Disables the parity checking error interrupt. 1b: Enables the parity checking error interrupt.
9	PREI	0h	RW	Preamble error interrupt enable Enables the preamble checking error interrupt. 0b: Disables the preamble error interrupt. 1b: Enables the preamble error interrupt.
8	CSEI	0h	RW	Channel status error interrupt enable Enables the channel status error interrupt. 0b: Disables the channel status error interrupt. 1b: Enables the channel status error interrupt.
7	ABOI	0h	RW	Audio buffer overrun interrupt enable Enables the receiver audio buffer overrun interrupt. 0b: Disables the audio buffer overrun interrupt. 1b: Enables the audio buffer overrun interrupt.
6	ABUI	0h	RW	Audio buffer underrun interrupt enable Enables the transmitter audio buffer underrun interrupt. 0b: Disables the audio buffer underrun interrupt. 1b: Enables the audio buffer underrun interrupt.
5	RUII	0h	RW	Receiver user information interrupt enable Enables the receiver user information register full interrupt. 0b: Disables the receiver user information interrupt. 1b: Enables the receiver user information interrupt.
4	TUII	0h	RW	Transmitter user information interrupt enable Enables the transmitter user information register empty interrupt. 0b: Disables the transmitter user information interrupt. 1b: Enables the transmitter user information interrupt.
3	RCSI	0h	RW	Receiver channel status interrupt enable Enables the receiver channel status register full interrupt. 0b: Disables the receiver channel status interrupt. 1b: Enables the receiver channel status interrupt.
2	RCBI	0h	RW	Receiver channel buffer interrupt enable Enables the receiver audio channel buffer full interrupt. 0b: Disables the receiver audio channel interrupt. 1b: Enables the receiver audio channel interrupt.

Bit	Bit Name	Initial Value	R/W	Description
1	TCSI	0h	RW	Transmitter channel status interrupt enable Enables the transmitter channel status register empty interrupt. 0b: Disables the transmitter channel status interrupt. 1b: Enables the transmitter channel status interrupt.
0	TCBI	0h	RW	Transmitter channel buffer interrupt enable Enables the transmitter audio channel buffer empty interrupt. 0b: Disables the transmitter audio channel interrupt. 1b: Enables the transmitter audio channel interrupt.

8.6.3.12 Status Register (SPDIFm_STAT)

This register indicates the status information of the SPDIF unit.

Access Size : 32 bits
Address : <SPDIFm_base> + 002Ch
Initial Value : 0000_C000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CMD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RIS	TIS	UBO	UBU	CE	PARE	PREE	CSE	ABO	ABU	RUIR	TUIR	CSRX	CBRX	CSTX	CBTX
Initial Value	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW0	RW0	RW0	RW0	RW0	RW0	RW0	RW0	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CMD	0h	R	Compressed mode data This bit is set when the received data are compressed mode data (the V flag and bit 1 of channel status = 1). 0b: The data are not compressed mode data. 1b: The data are compressed mode data.
15	RIS	1h	R	Receiver idle state This bit is set when the receiver is in the idle state. 0b: The receiver is not in the idle state. 1b: The receiver is in the idle state.
14	TIS	1h	R	Transmitter idle state This bit is set when the transmitter is in the idle state. 0b: The transmitter is not in the idle state. 1b: The transmitter is in the idle state.
13	UBO	0h	RW0	User buffer overrun*1 This bit is set in the case of an overrun of the receiver user buffer. This bit is cleared by writing 0 to it. When the REIE and UBOI bits of the control register are set, an interrupt occurs. 0b: The user buffer has not overrun. 1b: The user buffer has overrun.
12	UBU	0h	RW0	User buffer underrun*1 This bit is set in the case of an underrun of the transmitter user buffer. This bit is cleared by writing 0 to it. When the TEIE and UBUI bits of the control register are set, an interrupt occurs. 0b: The user buffer has not underrun. 1b: The user buffer has underrun.
11	CE	0h	RW0	Clock error*1 This bit is set when synchronization cannot be obtained in clock recovery. This bit is cleared by writing 0 to it. When the REIE and CREI bits of the control register are set, an interrupt occurs. 0b: Normal clock recovery 1b: Clock recovery error
10	PARE	0h	RW0	Parity error*1 This bit is set when the result of parity checking is the detection of an error. This bit is cleared by writing 0 to it. When the REIE and PAEI bits of the control register are set, an interrupt occurs. 0b: The result of parity checking is normal. 1b: Parity error

Bit	Bit Name	Initial Value	R/W	Description
9	PREE	0h	RW0	Preamble error*1 This bit is set when a preamble is undefined or a preamble does not appear in the right position. This bit is cleared by writing 0 to it. When the REIE and PREI bits of the control register are set, an interrupt occurs. Note: The bit is only set after a start of block preamble. 0b: A preamble is in the right position. 1b: Preamble error
8	CSE	0h	RW0	Channel status error*1 This bit is set when the channel status information is written before the 32nd frame that is in the transfer state. This bit is cleared by writing 0 to it. When the TEIE and CSEI bits of the control register are set, an interrupt occurs. 0b: Normal channel status 1b: Channel status error
7	ABO	0h	RW0	Audio buffer overrun*1 This bit indicates that both the first and second stages of the receiver audio buffer are full so that further received data will be written over data that have already been received. This bit is cleared by writing 0 to it. When the REIE and ABOI bits of the control register are set, an interrupt occurs. 0b: The receiver audio buffer has not overrun. 1b: The receiver audio buffer has overrun.
6	ABU	0h	RW0	Audio buffer underrun*1 This bit indicates that both the first and second stages of the transmitter audio buffer are empty so that further transmission will be repetition of the last data to have been transmitted. This bit is cleared by writing 0 to it. When the TEIE and ABUI bits of the control register are set, an interrupt occurs. 0b: The transmitter audio buffer has not underrun. 1b: The transmitter audio buffer has underrun.
5	RUIR	0h	R	Receiver user information register This bit indicates the state of the receiver user information register. Reading the receiver user register clears this bit. When the RUII bit of the control register is set, an interrupt occurs. 0b: The receiver user information register is empty. 1b: The receiver user information register is full.
4	TUIR	0h	R	Transmitter user information register This bit indicates the state of the transmitter user information register. Writing to the transmitter user register clears this bit. When the TUII bit of the control register is set, an interrupt occurs. 0b: The transmitter user information register is full. 1b: The transmitter user information register is empty.
3	CSRX	0h	R	Channel 1 and 2 status (for receiver) This bit indicates the state of the receiver channel status register. Reading the receiver channel status register clears this bit. When the RCSI bit of the control register is set, an interrupt occurs. 0b: The receiver channel status register is empty. 1b: The receiver channel status register is full.
2	CBRX	0h	R	Channel 1 and 2 buffers (for receiver) This bit indicates the state of the receiver audio channel register. Reading the receiver audio channel register clears this bit. When the RCBI bit of the control register is set, an interrupt occurs. 0b: The receiver audio channel register is empty. 1b: The receiver audio channel register is full.
1	CSTX	0h	R	Channel 1 and 2 status (for transmitter) This bit indicates the state of the transmitter channel status register. Writing the transmitter channel status register clears this bit. When the TCSI bit of the control register is set, an interrupt occurs. 0b: The transmitter channel status register is full. 1b: The transmitter channel status register is empty.
0	CBTX	0h	R	Channel 1 and 2 buffers (for transmitter) This bit indicates the state of the transmitter audio channel register. Writing to the transmitter audio channel register clears this bit. When the TCBI bit of the control register is set, an interrupt occurs. 0b: The transmitter audio channel register is full. 1b: The transmitter audio channel register is empty.

Note 1. When an error bit is detected during DMA transfer, re-making the DMA transfer settings is required. In this case, set the enable bit (the RME or TME bit) and the DMA enable bit (the RDE or TDE bit) to "disabled" to release the error state, and then re-set the direct memory access controller. After that, set the module enable bit to "enabled", and the DMA transfer can be resumed.

8.6.3.13 Transmitter DMA Audio Data Register (SPDIFm_TDAD)

This register is for DMA writing of the channel 1/2 audio PCM data for the transmitter.

Access Size : 32 bits
Address : <SPDIFm_base> + 0030h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	PCM_DATA[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCM_DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	W	Reserved A register that can be written. Reading is invalid. The write value should always be 0b.
23 to 0	PCM_DATA [23:0]	0h	W	The read value is undefined Audio PCM data PCM-encoded audio data (stored flush against the LSB side)

8.6.3.14 Receiver DMA Audio Data Register (SPDIFm_RDAD)

This register is for DMA reading of the channel 1/2 audio PCM data for the receiver.

Access Size : 32 bits
Address : <SPDIFm_base> + 0034h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	PCM_DATA[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCM_DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 0	PCM_DATA [23:0]	0h	R	Audio PCM data PCM-encoded audio data (stored flush against the LSB side)

8.6.4 Operation

8.6.4.1 Transmitter Control

8.6.4.1.1 Overview of Transmitter Control

The transmitter encodes the PCM data and auxiliary information to be sent with the biphase mark modulation in conformance with the IEC 60958 standard (SPDIF) and sends them.

■ **Clock**

The clock for the transmitter is the externally supplied oversampling clock. An oversampling value that is 8 times the clock frequency required for the biphase mark encoding is usually selected for this clock. In this case, the clock frequency that is required for transmitting the 32 time slots in a sub-frame will be 512 times the sampling frequency of the audio data.

■ **Audio Data and Channel Status Information**

The audio data and channel status information for channel 1 and then channel 2 of the module are written.

Writing of the channel status is only required when the information is changed. Writing of the channel status information will be required by the SPDIF module after 30 frames (when all transmission of the current channel status data has been completed), so write it from the 31st frame and before the next block (= 192 frames) starts.

Audio data are stored in a double-buffer configuration. There are two ways to check that data are not present in the first stage of the double buffer.

- Sending an interrupt request
- Reading the status register

■ **DMA Transfer**

Channel 1 audio data are transmitted with the use of DMA transfer in response to the first request and channel 2 audio data are transmitted in response to the second request.

■ **Channel Status**

The channel status information to be transmitted is set in the 30-bit registers for channels 1 and 2. Each block has 192 bits per channel of channel status information. However, only 30 of the 192 bits are essential data, so 0 is transmitted after the first 30 bits have been transmitted until completion of the block.

■ **User Data**

User data are in a 32-bit double-buffer arrangement. Issue an interrupt request or read the status register to check that the first stage of the double buffer is empty. Generally, the data length between blocks is insufficient for the user data information, and the user data information is transmitted in order of channel 1 and channel 2. After the 384 bits of user data in the interval of one block have been transmitted, the next block is transmitted in succession.

■ **Audio Data**

The format of audio data handled by the SPDIF is linear PCM, and the maximum settable size is 24 bits. For this reason, the V flag that indicates the audio data are linear PCM is always 0. No register setting is required in relation to the V flag. An even parity bit is generated for each 32 bits of the serial output data (that is, without including the preamble).

8.6.4.1.2 Transmitter Initialization

The transmitter is placed in the idle state by default after a reset. It can also be placed in the idle state by writing 0 to the TME bit of the CTRL register. In the idle state, the settings of the transmitter will be as follows.

- The idle status bit (TIS) of the transmitter is set to 1, and the other status bits are all cleared to 0.
- Preamble generation is invalid.
- Synchronization of channels 1 and 2 is set to 0 (channel 1: 0, channel 2: 1).
- The word count and frame count will become 0.
- Output from the encoder based on the biphase mark will be 0.

The channel status register, user data register, and audio data register retain their values before the state transition to the idle state. Write 1 to the TME bit of the CTRL register to cause return from the idle state.

8.6.4.1.3 Transmitter Initialization Settings

Setting the TME bit to 1 causes the TUIR and CSTX bits to be set to 1. At this time, writing data in the order 1) TUI, then 2) TLCS, TRCS causes a channel status error. To suppress the occurrence of an error, write data in the order 1) TLCS, TRCS, then 2) TUI.

In addition, after the TME bit is set to 1, writing of the first audio data (writing to TLCA and TRCA by CPU access or writing to TDAD by DMA transfer) should proceed after clearing of CSTX and TUIR has been checked following writing to TLCS, TRCS, and TUI.

8.6.4.1.4 Data Transfer to the Transmitter

When the transmitter has returned from the idle state, it is ready for data transfer. Data transfer can be started in any of three ways:

- Using interrupts
- Using DMA requests
- Reading the status register

Interrupt signals that are common to transmission and reception and individual DMA request signals for the transmitter are prepared.

For details of the transfer of data in response to a DMA request, see **8.6.4.1.5 DMA Data Transfer to the Transmitter**.

Figure 8.6-5 shows the flow of data transfer to the transmitter by using interrupts.

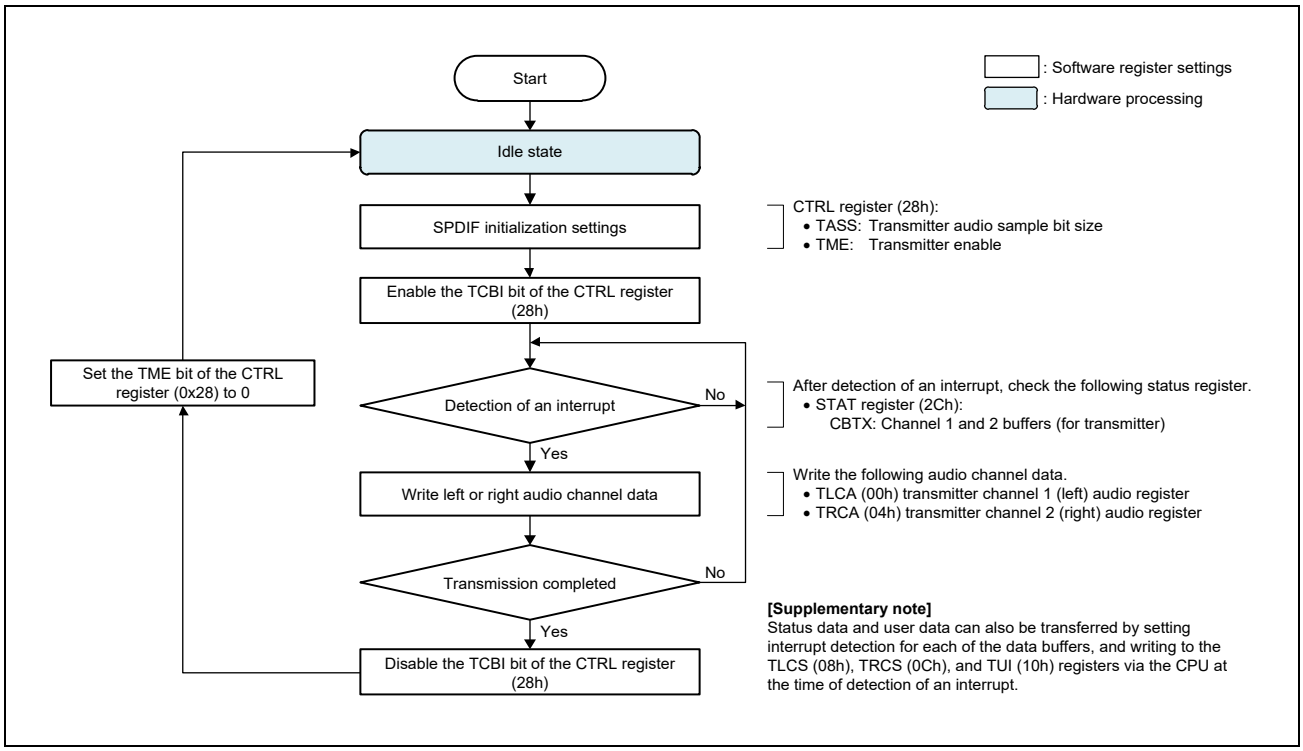


Figure 8.6-5 Flow of Data Transfer to the Transmitter (Using Interrupts)

The channel status information is updated in response to a change in the information.

Because the updating should proceed before transmission of the next block, write the channel status information for updating after the transmission of 30 frames has been completed. Completion of the transmission of 30 frames can be indicated by an interrupt or detected by reading the status bit. Writing the channel status information before the transmission of 30 frames has been completed (during transmission of the current information) leads to setting of the channel status error bit (CSE) of the status register and generates an interrupt.

Remark All valid information of one channel status block can be transmitted with 30 frames.

8.6.4.1.5 DMA Data Transfer to the Transmitter

This section describes the procedure for transmitting the SPDIF data with the use of the DMAC for DMA transfer.

Figure 8.6-6 shows the flow of data transfer to the transmitter by using DMA requests.

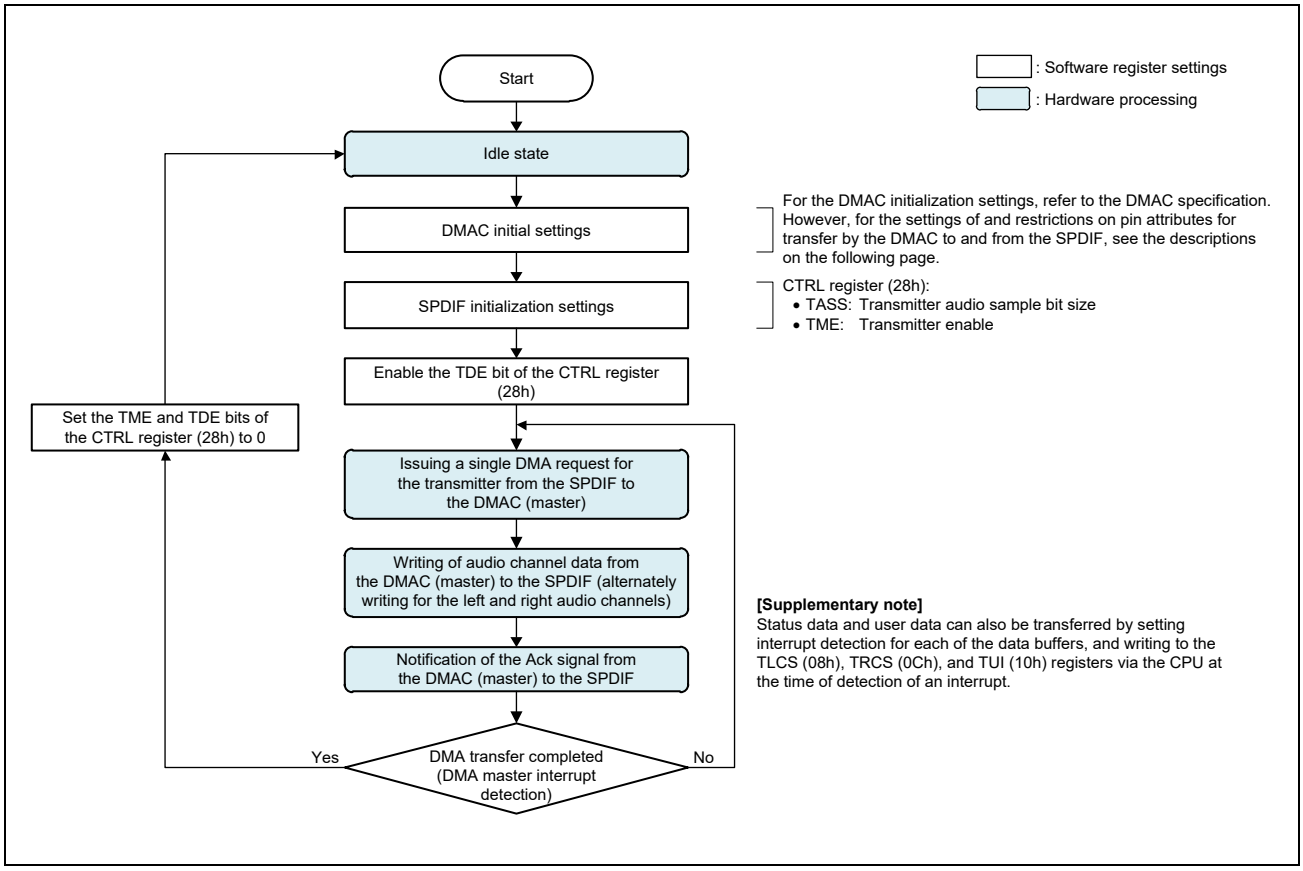


Figure 8.6-6 Flow of Data Transfer to the Transmitter (Using DMA Requests)

Set the DMAC master (in the CHCFG register) at the time of data transfer by DMA for the transmitter as follows.

1. Set the single transfer mode (one round of DMA transfer processing proceeds in response to one DMA request).
   Register setting on the DMA master side: the TM bit: 0b

2. DMA transfer is in 32-bit units.
   Register setting on the DMA master side: the DDS bit: 32-bit setting

3. Set DMAC REQ and DMAC ACK as level signals (DMA ACK is asserted until DMA REQ is deasserted). The active sense of the REQ signal is set to the high level.
   Register settings on the DMA master side: AM[2:0] = 001b, LVL = 1b, HIEN = 1b, LOEN = 0b

   Note: The active sense of the ACK signal is fixed to the high level.

4. Set the DMAC ACK signal to be activated at the time of write access to the SPDIF.
   Register setting on the DMA master side: REQD = 1

5. Data for DMA transfer are written to the TDAD register (transmission) of the SPDIF unit.

6. When writing to the TDAD register at the time of DMA transfer, audio data are alternately written to the left and right channels.

For the SPDIF initialization at the time of DMA transfer, initialize the SPDIF unit according to **8.6.4.1.2 Transmitter Initialization**. At that time, enabling DMA transfer by setting the DMA enable bit (TDE = 1b) of the SPDIF unit is also required.

8.6.4.2 Receiver Control

8.6.4.2.1 Overview of Receiver Control

The receiver demodulates the data and clock signals from the input encoded in accord with the IEC 60958 standard.

The demodulated data are stored in the audio data register in a linear PCM format, and the channel status and user information that are received at the same time as auxiliary information are stored in the appropriate registers.

■ **Clock**

The main clock for the receiver is the externally supplied oversampling clock.
The receiver operates at 4 times the frequency of the oversampling clock.

Remark The oversampling clock is the same for the transmitter and receiver.

■ **Clock Recovery**

Clock recovery proceeds with the use of the pulse-width counter and averaging filter, and the sampling pulse is generated at the mid-point of each bit of the input data. The clock error status bit indicates that clock synchronization has failed. Synchronization starts when a first preamble appears in the input data. As long as changes in the jitter and clock frequency are in the range of clock recovery specification, effects of these changes on clock recovery can be constrained by continual adjustment.

When the clock recovery has been successful, the decoder starts preamble detection with the use of the biphase mark method. The decoder searches for a preamble to indicate the start of a block (see **Table 8.6-1**). The preamble error status bit indicates that the next preamble did not appear with the right timing. This can be considered as being due to an omission or the failure of transfer.

■ **Parity Checking**

An even-parity checking is run on the decoded data. If a mismatch is encountered, the parity error status bit is set.

■ **Audio Data**

Audio data are stored in a double-buffer configuration. Data can be detected to be readable by transmitting an interrupt request due to the buffer-full state or reading the corresponding status bit. Channel 1 audio data are transmitted with the use of DMA transfer in response to the first request and channel 2 audio data are transmitted in response to the second request.

■ **User Data and Channel Status Information**

The SPDIF module acquires the user data and channel status information as well as the audio data.

The channel status information is stored in a 30-bit register. Storage in the register is not completed until a total of 30 frames for each channel have been received because one bit is received per sub-frame. The new channel status information is compared with the current value, and the CPU only reads the new information if a change has occurred. The user data are also received at the same time. However, the reception is completed per 16 frames because the user data are stored in the register per sub-frame.

Remark 1. Requests for channel status information cannot be transferred by DMA transfer.

Remark 2. When an overrun of the receiver user buffer occurs, the current data in the SPDIF buffer are overwritten by the next data input from the SPDIF.

8.6.4.2.2 Receiver Initialization

The receiver is placed in the idle state by default after a reset. It can also be placed in the idle state by writing 0 to the RME bit of the CTRL register. In the idle state, the settings of the receiver will be as follows.

- The idle status bit of the receiver is set to 1, and the other status bits are all cleared to 0.
- Synchronization of channels 1 and 2 is set to 0 (channel 1: 0, channel 2: 1).
- The word count and frame count will become 0.

The channel status register, user data register, and audio data register retain their values before the state transition to the idle state. Write 1 to the RME bit of the CTRL register to cause return from the idle state.

8.6.4.2.3 Data Transfer from the Receiver

When the receiver has returned from the idle state, it is ready for data transfer. Data transfer can be started in any of three ways:

- Using interrupts
- Reading the status register
- Using DMA requests

For details of the transfer of data in response to a DMA request, see **8.6.4.2.4 DMA Data Transfer from the Receiver**.

Interrupt signals that are common to transmission and reception and individual DMA request signals for the receiver are prepared. An interrupt may be generated by an error signal due to any of the following factors during data transfer from the receiver.

1. Clock recovery error
2. Failure of or a fault in transfer is indicated by a preamble error
3. Parity checking error

A failure of or fault in transfer may lead to the preamble for starting a sub-frame or block having a shifted position or not appearing.

A parity checking error occurs when the parity bit is wrong. This error may occur due to any of the factors above.

■ **Tolerance of Clock Recovery**

The margin for clock recovery in reception is expressed by the following formula.

$$M = \left| \left(0.5 - \frac{1}{2N}\right) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\%$$

M: Reception margin
N: Oversampling rate
L: Frame length = 33
D: Duty cycle = 0.6
F: Oversampling clock deviation = Level II accuracy = $1000 \times 10e^{-6}$

Figure 8.6-7 shows the reception margin M.

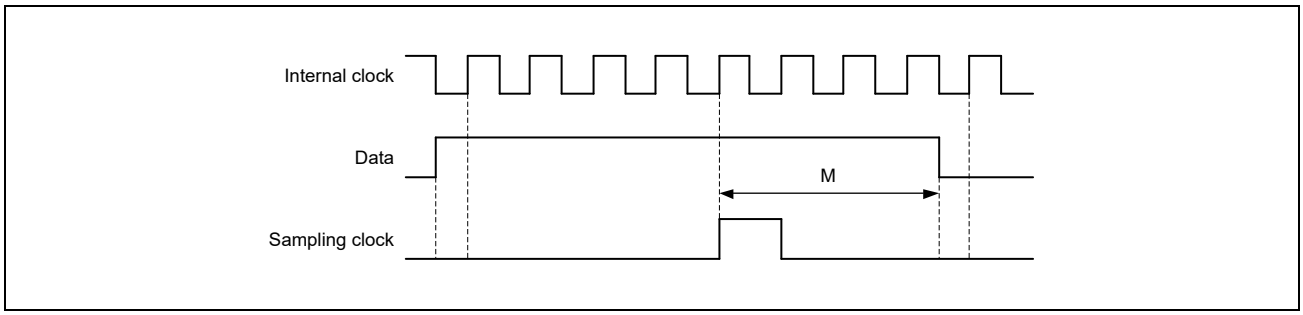


Figure 8.6-7 Reception Margin

Introducing jitter causes the previous formula to become the following inequality.

$$J \leq \left| \left(0.5 - \frac{1}{2N}\right) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\%$$

J: Clock jitter
8 times the oversampling rate: reception margin = 39.25%
4 times the oversampling rate: reception margin = 31.75%
Twice the oversampling rate: reception margin = 16.75%

The fastest sampling frequency is 48 kHz. In this case, the required clock frequency will be 128 × 48 kHz = 6.144 MHz. In this case, the worst-case jitter in one cycle is defined as 40 ns = 24.5% of a cycle. Therefore, the above inequality will be satisfied when the oversampling rate is greater than or equal to 4.

Figure 8.6-8 shows the flow of data transfer from the receiver by using interrupts.

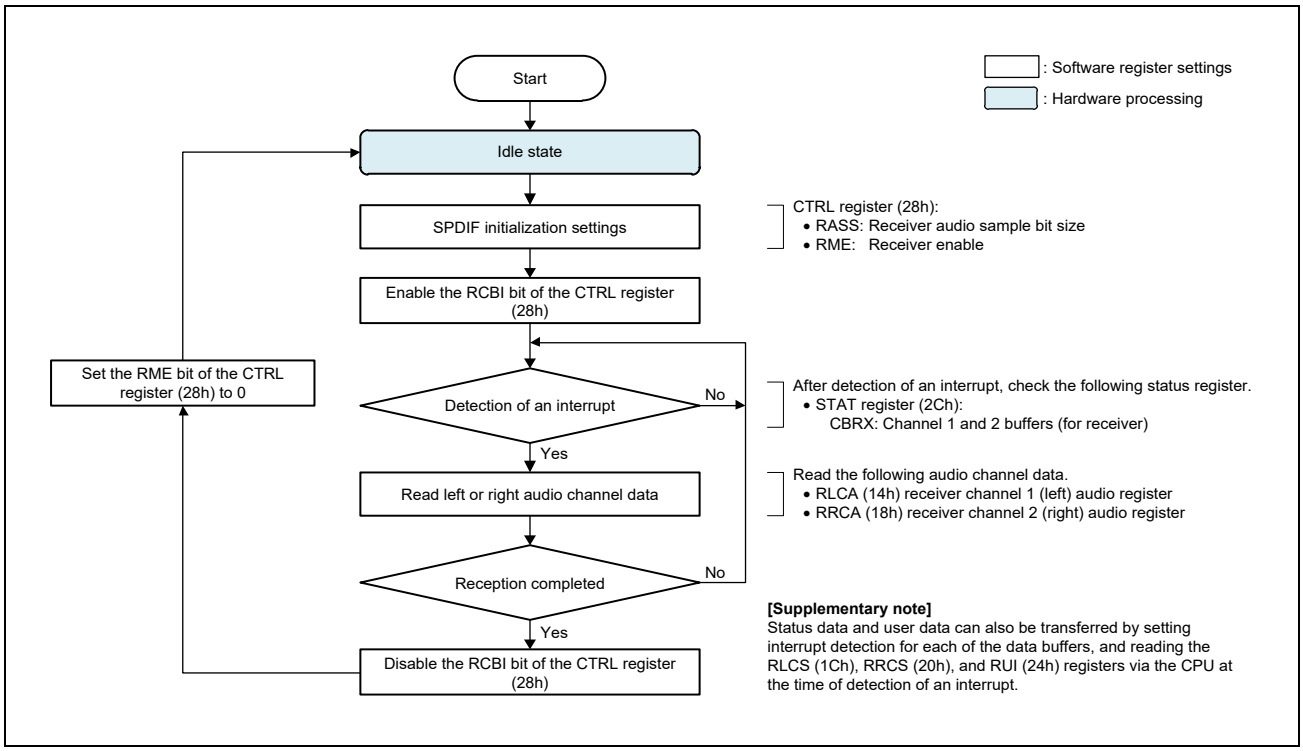


Figure 8.6-8 Flow of Data Transfer from the Receiver (Using Interrupts)

The interrupt that provides notification of information having been stored in the channel status information register is only generated when the information is found to have changed after reception of the 30th frame has been completed.

8.6.4.2.4 DMA Data Transfer from the Receiver

This section describes the procedure for receiving the SPDIF data with the use of the DMAC for DMA transfer.

Figure 8.6-9 shows the flow of data transfer from the receiver by using DMA requests.

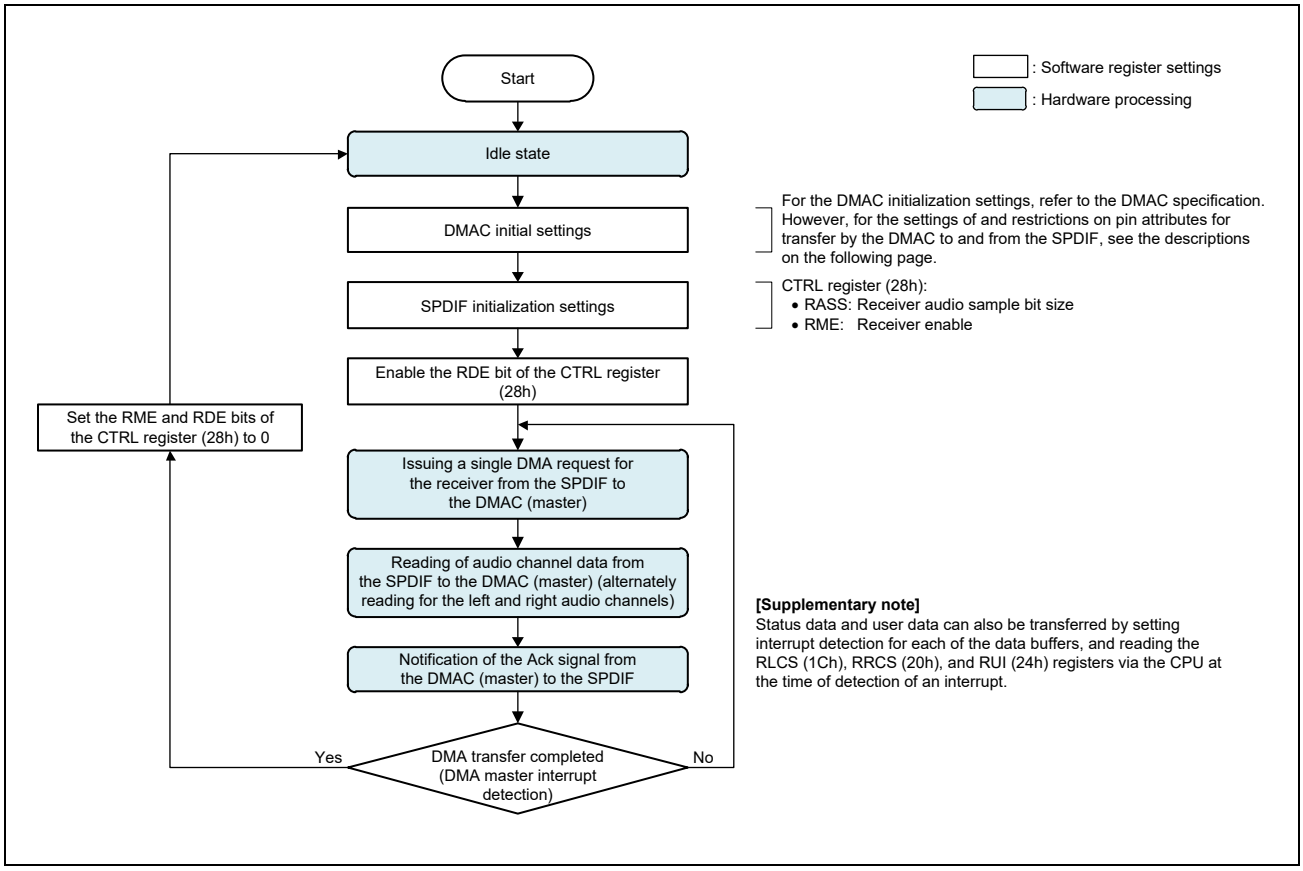


Figure 8.6-9 Flow of Data Transfer from the Receiver (Using DMA Requests)

Set the DMAC master (in the CHCFG register) at the time of data transfer by DMA from the receiver as follows.

1. Set the single transfer mode (one round of DMA transfer processing proceeds in response to one DMA request).
   Register setting on the DMA master side: the TM bit: 0b

2. One DMA transfer is in 32-bit units.
   Register setting on the DMA master side: the SDS bit: 32-bit setting

3. Set DMAC REQ and DMAC ACK as level signals (DMA ACK is asserted until DMA REQ is deasserted). The active sense of the REQ signal is set to the high level.
   Register settings on the DMA master side: AM[2:0] = 001b, LVL = 1b, HIEN = 1b, LOEN = 0b

   Note: The active sense of the ACK signal is fixed to the high level.

4. Set the DMAC ACK signal to be activated at the time of read access to the SPDIF.
   Register setting on the DMA master side: REQD = 0b

5. Data for DMA transfer are read from the RDAD register (reception) of the SPDIF unit.

6. Reading of the RDAD register at the time of DMA transfer is alternately of data for the left and right channels.

For the SPDIF initialization at the time of DMA transfer, initialize the SPDIF unit according to **8.6.4.2.2 Receiver Initialization**. At that time, enabling DMA transfer by setting the DMA enable bit (RDE = 1b) of the SPDIF unit is also required.

8.6.4.3 Stopping Transmitter and Receiver

The operation of the transmitter and receiver can be invalidated by writing 0b to the enable bits in the control register (the TME bit for the transmitter, and the RME bit for the receiver). This idle state can be detected by reading the idle bits (the TIS bit for the transmitter, and the RIS bit for the receiver).

8.6.4.4 Compressed Mode Data

Compressed mode data is defined in the IEC 61937 specification. The SPDIF module only detects the compressed mode data. Check the validity flag (V flag) and bit 1 of the channel status data, and when both are 1, set the CMD bit of the status register to 1 to select compressed mode data.

Remark Only the receiver detects compressed mode data. This information has no effect on the transmitter.

8.6.4.5 Reset and Clock Control

In terms of controlling a reset and clock, the following reset and clock pins require control.

Table 8.6-6 Relationship between Reset and Clock

	Reset Signal	Clock Signal	Description
System 1	rst	clkp	For internal bus control circuit
System 2	rst*1	pa_audio_a	For external interface control circuit

Note 1. No measures for non-synchronization with the circuit.

The above reset and clock pins are controlled by the CPG unit, and each reset and clock signal can be switched off and on by the CPG register. The mechanism for stopping the clkp clock before and after release from the reset when the rst reset is released is implemented on the CPG side. However, a mechanism for stopping before and after release from the rst reset is not implemented for the pa_audio_a clock, so handling of reset and clock control in the way described below is required as a measure for non-synchronization. Perform the reset and clock control while the operation of the SPDIF unit is stopped, and while access by related units is also stopped.

1. Release the SPDIF from the reset state and supply the clock:
   Step1: Start supply of the clkp clock through the register setting of the CPG unit.
   Step 2: Release (deassert) the rst reset signal through the register setting of the CPG unit.
   Step 3: Start supply of the pa_audiox_a clock through the register setting of the CPG unit.

   Note: The state immediately after starting the device is as stated below.
   clkp clock: Supply stopped state, pa_audiox_a clock: Supply stopped state, rst reset: Reset state

2. Enable the SPDIF reset and stop the clock:
   Step 1: Stop supply of the pa_audiox_a clock through the register setting of the CPG unit.
   Step 2: Enable (assert) the rst reset signal through the register setting of the CPG unit.
   Step 3: Stop supply of the clkp clock through the register setting of the CPG unit.

3. Apply a brief hardware reset of the whole SPDIF macro.
   Step 1: Activate (assert) the rst reset signal through the register setting of the CPG unit.
   Step 2: Stop supply of the pa_audiox_a clock through the register setting of the CPG unit.
   Step 3: Release (deassert) the rst reset signal through the register setting of the CPG unit.
   Step 4: Start supply of the pa_audiox_a clock through the register setting of the CPG unit.

   Note: The above steps are performed while the clkp clock is being supplied.

8.6.5 Interrupt

8.6.5.1 Interrupt List

Table 8.6-7 lists the interrupt signal of this unit.

Table 8.6-7 Interrupt List

Interrupt signal	Function	Pulse/Level	Min. Pulse	Active Level	Clock
intreq_spdif_n	SPDIF interrupt signal	Level	—	Low	clkp

The above interrupt signal operates as the logical OR of multiple interrupt sources.

The specifiable interrupt sources are as follows.

- Receiver error interrupt
- Transmitter error interrupt
- User buffer overrun interrupt
- User buffer underrun interrupt
- Clock recovery error interrupt
- Parity error interrupt
- Preamble error interrupt
- Channel status error interrupt
- Audio buffer overrun interrupt
- Audio buffer underrun interrupt
- Receiver user information interrupt
- Transmitter user information interrupt
- Receiver channel status interrupt
- Receiver channel buffer interrupt
- Transmitter channel status interrupt
- Transmitter channel buffer interrupt

Switching the interrupt sources listed above on and off can be specified in the control register (CRTL).

Checking the interrupt source when an interrupt occurs and clearing the source can be done through the status register (STAT).

8.6.6 Usage Notes

8.6.6.1 Clearing TUIR

After writing to TUI and until TUIR is cleared, a wait for the completion of the transmission of up to 1 frame is required. When writing data to TUI in response to a transmitter user information interrupt, check that TUIR has been cleared, and then stop the interrupt processing routine to prevent the interrupt being accepted again.

8.6.6.2 Frequency of Audio Input Clock

The frequency of the externally input oversampling clock (pa_audiox_a) requires setting to 512 fs.

In addition, the frequency of the externally input oversampling clock (pa_audiox_a) must be kept lower than the SPDIF internal HPB bus clock (clkp) frequency.

SECTION 8 AUDIO

8.7 PDM Interface (PDM)

This section describes the PDM interface.

8.7.1 Overview

This PDM interface is a unit for receiving voice data from an external PDM (pulse density modulation) device. For features and configuration, see the following sections.

8.7.1.1 Feature

Table 8.7-1 lists the features of this PDM interface.

Table 8.7-1 PDM-IF IP Specifications

Item	Description
Number of channels	Maximum 3 channels per 1 IP (× 2 units)
Functions	• Capable of filtering 1-bit digital input data PDM_DATn (n = 0, 1, 2) and converting them into 20-bit or 16-bit digital data. The bit order is little-endian. • The IP supports stereo microphone (L/R sampling by rising/falling clock edge). • The IP supports sound activity detector. • Each channel of IP includes programmable filters: 4th order sinc filter, high-pass filter (for suppression of DC bias), correction filter (for sinc passband distortion), half-band decimation filter (for aliasing distortion). – IP supports programmable and flexible decimation ratios. – The sinc filter is selectable as first-, second-, third-, and fourth-order. • The IP supports DMA operation through APB. • Each channel of IP has an internal buffer. – 64 stages when designed – Capable of storing voice data during low power mode • Error detection functions can be used for debugging.
Interrupt sources	Maximum 7 sources in 1 IP 1. Data reception interrupt per channel (Maximum 3 interrupts) 2. Sound detection interrupt (shared between channels) 3. Error detection interrupt per channel (Maximum 3 interrupts)
Low power consumption function	The IP supports microphone low power configuration (slower clock). The IP can switch to the higher clock speed after sound detection recognized.
Bus interface	The IP has an APB interface which is confirmed to APB4.

Since this LSI supports 24 MHz/5 = 4.8 MHz CLK (CCLK), the PDM_CLK frequency can be discretely selected in the range of 2.4 MH to 0.15 MHz for the following target sampling frequencies. For details, see **8.7.3.1 Clock Specifications**.

Target sampling frequencies: 48, 40, 30, 25, 24, 20, 16, 15, 12, 10, 8 kHz

8.7.1.2 Block Diagram

The following is a block diagram of the PDM interface.

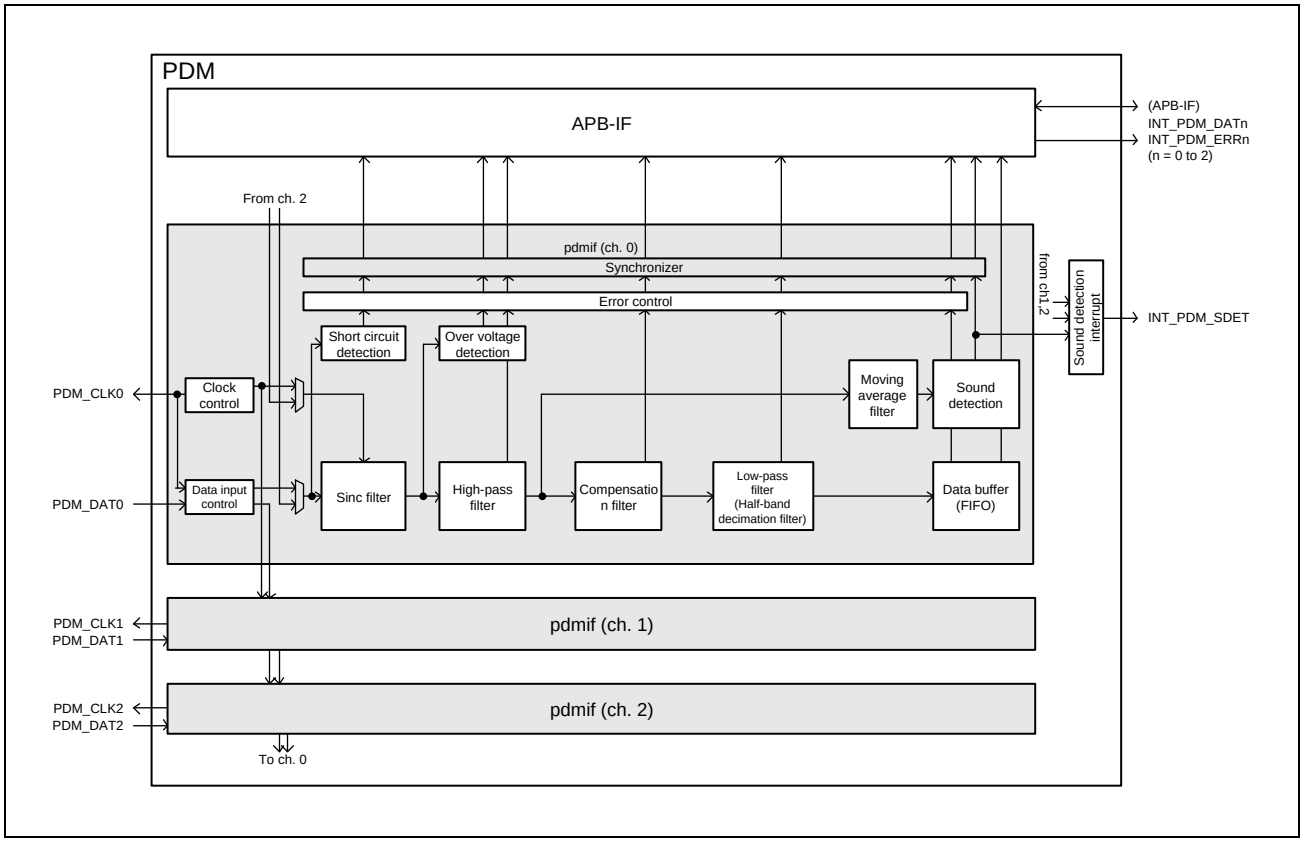


Figure 8.7-1 Three-Channel PDM-IF Block Diagram

Each channel of PDM-IF consists of following blocks:

- Clock control: Generates PDM_CLKn (n = 0, 1, 2) to the microphone, and internal timing signals to the sinc filter.
- Data input control: Receives PDM_DATn (n = 0, 1, 2) from the microphone.
- Sinc filter: Converts the 1-bit digital data stream input from PDM_DATn (n = 0, 1, 2) to 34-bit signed data. Then, the 34-bit signed data is clipped to 20-bit signed data.
- High-pass filter: A filter for suppression of DC bias
- Compensation filter: A filter for sinc passband distortion
- Low-pass (half-band decimation) filter: A filter for aliasing distortion
- Moving average filter: A simple filter for noise reduction before sound detection
- Sound detection: Generates the wakeup interrupt to recover from low power mode.
- Data buffer: Stores 20-bit or 16-bit data filtered by some filters.
- Short circuit detection: Detects the short circuit error which indicates consecutive 0’s or 1’s input from PDM_DATn (n = 0, 1, 2).
- Overvoltage detection: Detects the overvoltage error.
- Error control: Generates error flags from other blocks.

The 1-bit digital data stream input from PDM_DATn pin (n = 0, 1, 2) is converted to 34-bit signed data by the sinc filter and then the data is clipped to 20-bit signed data. The 20-bit singed data is used for sound detection and data storing in buffer.

The sound detection block receives 20-bit signed data through high-pass filter block and moving average filter block. The block compares received data to the programmed thresholds and generates a wakeup interrupt in case that the data is above the upper threshold or below the lower threshold.

The data buffer block receives 20-bit signed data through high-pass filter, compensation filter, and low-pass filter. The block stores the 20-bit signed data or the clipped 16-bit signed data. And then the block sends the 20-bit signed data or 16-bit signed data to APB-IF once APB read access occurs.

8.7.1.3 External Pins

Table 8.7-2 lists the PDM external pin specifications.

Table 8.7-2 Pin Configuration

External Pin Name	Input/Output	Function	Pin Name in This Section
PDMDATn0	Input	External PDM CH0 data reception pin	PDM_DAT0
PDMDATn1	Input	External PDM CH1 data reception pin	PDM_DAT1
PDMDATn2	Input	External PDM CH2 data reception pin	PDM_DAT2
PDMCLKn0	Output	External PDM CH0 clock output pin	PDM_CLK0
PDMCLKn1	Output	External PDM CH1 clock output pin	PDM_CLK1
PDMCLKn2	Output	External PDM CH2 clock output pin	PDM_CLK2

Note: n = 0, 1

8.7.1.4 Connected Unit

Table 8.7-3 lists the connected units.

Table 8.7-3 Connected Unit

Unit Name	Connected Unit Name	Function of Connected Unit
PDMm	CPG	Clock pulse generator (CPG)
	SYSTEMBUS	System bus
	ICU	Interrupt control unit (ICU)
	PFC	Pin function controller (PFC)

8.7.2 Registers

The register addresses of PDM are given as offsets from the individual base addresses <PDMm_base>. The register base addresses of each PDM are listed in the following table.

Table 8.7-4 Register Base Addresses

Base Address Name	Unit	Base Address
<PDM0_base>	PDM0	0_1104_0000h (5104_0000h*1, 4104_0000h*2)
<PDM1_base>	PDM1	0_1105_0000h (5105_0000h*1, 4105_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

8.7.2.1 List of Registers

The list of PDM registers and the memory addresses are as follows.

For the actual addresses, the offset values indicated in the following table are added to the base addresses.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
Channel Software Start Trigger Register	PDMm_PDCSTRTR	0000_0000h	0000h	8, 16, 32
Channel Software Stop Trigger Register	PDMm_PDCSTPTR	0000_0000h	0004h	8, 16, 32
Channel Software Change Trigger Register	PDMm_PDCCHGTR	0000_0000h	0008h	8, 16, 32
Channel Interrupt Control Register	PDMm_PDCICR	0000_0000h	000Ch	8, 16, 32
Channel Status Register	PDMm_PDCSR	0000_0000h	0010h	8, 16, 32
Channel Status Clear Register	PDMm_PDCSCR	0000_0000h	0014h	8, 16, 32
Reserve	-	-	0018h to 001Fh	-
Channel Sound Detection Control Register	PDMm_PDCSDCR	0000_0000h	0020h	8, 16, 32
Channel Data Read Control Register	PDMm_PDCDRCR	0000_0000h	0024h	8, 16, 32
Channel Data Clear Register	PDMm_PDCDCR	0000_0000h	0028h	8, 16, 32
Reserve	-	-	002Ch to 007Fh	-
Version Register	PDMm_PDVR	0000_0010h	0080h	8, 16, 32
Reserve	-	-	0084h to 00FFh	-
Software Start Trigger Register Channel 0	PDMm_PDSTRTRCH0	0000_0000h	0100h	8, 16, 32
Software Stop Trigger Register Channel 0	PDMm_PDSTPTRCH0	0000_0000h	0104h	8, 16, 32
Software Change Trigger Register Channel 0	PDMm_PDCHGTRCH0	0000_0000h	0108h	8, 16, 32
Interrupt Control Register Channel 0	PDMm_PDICRCH0	0000_0000h	010Ch	8, 16, 32
Status Detection Control Register Channel 0	PDMm_PDSDCRCH0	0000_0000h	0110h	8, 16, 32
Status Register Channel 0	PDMm_PDSRCH0	0000_0000h	0114h	8, 16, 32
Status Clear Register Channel 0	PDMm_PDSCRCH0	0000_0000h	0118h	8, 16, 32
Reserve	-	-	011Ch to 011Fh	-
Mode Setting Register Channel 0	PDMm_PDMDSRCH0	0000_0000h	0120h	8, 16, 32
Sinc filter Control Register Channel 0	PDMm_PDSFCRCH0	057C_0000h	0124h	8, 16, 32
High-pass filter Coefficient s(0) Register Channel 0	PDMm_PDHFCS0RCH0	0000_3F61h	0128h	8, 16, 32
High-pass filter Coefficient k(1) Register Channel 0	PDMm_PDHFCK1RCH0	0000_3EC1h	012Ch	8, 16, 32
High-pass filter Coefficient h(0) Register Channel 0	PDMm_PDHFCH0RCH0	0000_4000h	0130h	8, 16, 32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
High-pass filter Coefficient h(1) Register Channel 0	PDMm_PDHFCH1RCH0	0000_C000h	0134h	8, 16, 32
Compensation filter Coefficient h(0) Register Channel 0	PDMm_PDCFCH00RCH0	0000_1FE8h	0138h	8, 16, 32
Compensation filter Coefficient h(1) Register Channel 0	PDMm_PDCFCH01RCH0	0000_0039h	013Ch	8, 16, 32
Compensation filter Coefficient h(2) Register Channel 0	PDMm_PDCFCH02RCH0	0000_003Ch	0140h	8, 16, 32
Compensation filter Coefficient h(3) Register Channel 0	PDMm_PDCFCH03RCH0	0000_1E56h	0144h	8, 16, 32
Compensation filter Coefficient h(4) Register Channel 0	PDMm_PDCFCH04RCH0	0000_01DCh	0148h	8, 16, 32
Compensation filter Coefficient h(5) Register Channel 0	PDMm_PDCFCH05RCH0	0000_06E1h	014Ch	8, 16, 32
Compensation filter Coefficient h(6) Register Channel 0	PDMm_PDCFCH06RCH0	0000_01DCh	0150h	8, 16, 32
Compensation filter Coefficient h(7) Register Channel 0	PDMm_PDCFCH07RCH0	0000_1E56h	0154h	8, 16, 32
Compensation filter Coefficient h(8) Register Channel 0	PDMm_PDCFCH08RCH0	0000_003Ch	0158h	8, 16, 32
Compensation filter Coefficient h(9) Register Channel 0	PDMm_PDCFCH09RCH0	0000_0039h	015Ch	8, 16, 32
Compensation filter Coefficient h(10) Register Channel 0	PDMm_PDCFCH10RCH0	0000_1FE8h	0160h	8, 16, 32
Low-pass filter Coefficient h0(10) Register Channel 0	PDMm_PDLFCH010RCH0	0000_0400h	0164h	8, 16, 32
Low-pass filter Coefficient h1(0) Register Channel 0	PDMm_PDLFCH10RCH0	0000_1FF8h	0168h	8, 16, 32
Low-pass filter Coefficient h1(1) Register Channel 0	PDMm_PDLFCH101RCH0	0000_000Ah	016Ch	8, 16, 32
Low-pass filter Coefficient h1(2) Register Channel 0	PDMm_PDLFCH102RCH0	0000_1FF0h	0170h	8, 16, 32
Low-pass filter Coefficient h1(3) Register Channel 0	PDMm_PDLFCH103RCH0	0000_0018h	0174h	8, 16, 32
Low-pass filter Coefficient h1(4) Register Channel 0	PDMm_PDLFCH104RCH0	0000_1FDCh	0178h	8, 16, 32
Low-pass filter Coefficient h1(5) Register Channel 0	PDMm_PDLFCH105RCH0	0000_0034h	017Ch	8, 16, 32
Low-pass filter Coefficient h1(6) Register Channel 0	PDMm_PDLFCH106RCH0	0000_1FB3h	0180h	8, 16, 32
Low-pass filter Coefficient h1(7) Register Channel 0	PDMm_PDLFCH107RCH0	0000_0076h	0184h	8, 16, 32
Low-pass filter Coefficient h1(8) Register Channel 0	PDMm_PDLFCH108RCH0	0000_1F2Eh	0188h	8, 16, 32
Low-pass filter Coefficient h1(9) Register Channel 0	PDMm_PDLFCH109RCH0	0000_0289h	018Ch	8, 16, 32
Low-pass filter Coefficient h1(10) Register Channel 0	PDMm_PDLFCH110RCH0	0000_0289h	0190h	8, 16, 32
Low-pass filter Coefficient h1(11) Register Channel 0	PDMm_PDLFCH111RCH0	0000_1F2Eh	0194h	8, 16, 32
Low-pass filter Coefficient h1(12) Register Channel 0	PDMm_PDLFCH112RCH0	0000_0076h	0198h	8, 16, 32
Low-pass filter Coefficient h1(13) Register Channel 0	PDMm_PDLFCH113RCH0	0000_1FB3h	019Ch	8, 16, 32
Low-pass filter Coefficient h1(14) Register Channel 0	PDMm_PDLFCH114RCH0	0000_0034h	01A0h	8, 16, 32
Low-pass filter Coefficient h1(15) Register Channel 0	PDMm_PDLFCH115RCH0	0000_1FDCh	01A4h	8, 16, 32
Low-pass filter Coefficient h1(16) Register Channel 0	PDMm_PDLFCH116RCH0	0000_0018h	01A8h	8, 16, 32
Low-pass filter Coefficient h1(17) Register Channel 0	PDMm_PDLFCH117RCH0	0000_1FF0h	01ACh	8, 16, 32
Low-pass filter Coefficient h1(18) Register Channel 0	PDMm_PDLFCH118RCH0	0000_000Ah	01B0h	8, 16, 32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
Low-pass filter Coefficient h1(19) Register Channel 0	PDMm_PDLFCH119RCH0	0000_1FF8h	01B4h	8, 16, 32
Sound Detection Lower Threshold Register Channel 0	PDMm_PDSDLTRCH0	0000_0000h	01B8h	8, 16, 32
Sound Detection Upper Threshold Register Channel 0	PDMm_PDSDUTRCH0	0000_0000h	01BCh	8, 16, 32
Data Buffer Control Register Channel 0	PDMm_PDDBCRCH0	0000_0000h	01C0h	8, 16, 32
Short Circuit Threshold Setting Register Channel 0	PDMm_PDSCTSRCH0	0000_0000h	01C4h	8, 16, 32
Overvoltage Lower Threshold Register Channel 0	PDMm_PDOVLTRCH0	0000_0000h	01C8h	8, 16, 32
Overvoltage Upper Threshold Register Channel 0	PDMm_PDOVUTRCH0	0000_0000h	01CCh	8, 16, 32
Reserve	-	-	01D0h to 01DFh	-
Data Read Control Register Channel 0	PDMm_PDDRCRCH0	0000_0000h	01E0h	8, 16, 32
Data Clear Register Channel 0	PDMm_PDDCRCH0	0000_0000h	01E4h	8, 16, 32
Data Read Register Channel 0	PDMm_PDDRRCH0	0000_0000h	01E8h	8, 16, 32
Data Status Register Channel 0	PDMm_PDDSRCH0	0000_0000h	01ECh	8, 16, 32
Reserve	-	-	01F0h to 01FFh	-
Software Start Trigger Register Channel 1	PDMm_PDSTRTRCH1	0000_0000h	0200h	8, 16, 32
Software Stop Trigger Register Channel 1	PDMm_PDSTPTRCH1	0000_0000h	0204h	8, 16, 32
Software Change Trigger Register Channel 1	PDMm_PDCHGTRCH1	0000_0000h	0208h	8, 16, 32
Interrupt Control Register Channel 1	PDMm_PDICRCH1	0000_0000h	020Ch	8, 16, 32
Status Detection Control Register Channel 1	PDMm_PDSDCRCH1	0000_0000h	0210h	8, 16, 32
Status Register Channel 1	PDMm_PDSRCH1	0000_0000h	0214h	8, 16, 32
Status Clear Register Channel 1	PDMm_PDSCRCH1	0000_0000h	0218h	8, 16, 32
Reserve	-	-	021Ch to 021Fh	-
Mode Setting Register Channel 1	PDMm_PDMDSRCH1	0000_0000h	0220h	8, 16, 32
Sinc filter Control Register Channel 1	PDMm_PDSFCRCH1	057C_0000h	0224h	8, 16, 32
High-pass filter Coefficient s(0) Register Channel 1	PDMm_PDHFCS0RCH1	0000_3F61h	0228h	8, 16, 32
High-pass filter Coefficient k(1) Register Channel 1	PDMm_PDHFCK1RCH1	0000_3EC1h	022Ch	8, 16, 32
High-pass filter Coefficient h(0) Register Channel 1	PDMm_PDHFCH0RCH1	0000_4000h	0230h	8, 16, 32
High-pass filter Coefficient h(1) Register Channel 1	PDMm_PDHFCH1RCH1	0000_C000h	0234h	8, 16, 32
Compensation filter Coefficient h(0) Register Channel 1	PDMm_PDCFCH00RCH1	0000_1FE8h	0238h	8, 16, 32
Compensation filter Coefficient h(1) Register Channel 1	PDMm_PDCFCH01RCH1	0000_0039h	023Ch	8, 16, 32
Compensation filter Coefficient h(2) Register Channel 1	PDMm_PDCFCH02RCH1	0000_003Ch	0240h	8, 16, 32
Compensation filter Coefficient h(3) Register Channel 1	PDMm_PDCFCH03RCH1	0000_1E56h	0244h	8, 16, 32
Compensation filter Coefficient h(4) Register Channel 1	PDMm_PDCFCH04RCH1	0000_01DCh	0248h	8, 16, 32
Compensation filter Coefficient h(5) Register Channel 1	PDMm_PDCFCH05RCH1	0000_06E1h	024Ch	8, 16, 32
Compensation filter Coefficient h(6) Register Channel 1	PDMm_PDCFCH06RCH1	0000_01DCh	0250h	8, 16, 32
Compensation filter Coefficient h(7) Register Channel 1	PDMm_PDCFCH07RCH1	0000_1E56h	0254h	8, 16, 32
Compensation filter Coefficient h(8) Register Channel 1	PDMm_PDCFCH08RCH1	0000_003Ch	0258h	8, 16, 32
Compensation filter Coefficient h(9) Register Channel 1	PDMm_PDCFCH09RCH1	0000_0039h	025Ch	8, 16, 32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
Compensation filter Coefficient h(10) Register Channel 1	PDMm_PDCFCH10RCH1	0000_1FE8h	0260h	8, 16, 32
Low-pass filter Coefficient h0(10) Register Channel 1	PDMm_PDLFCH010RCH1	0000_0400h	0264h	8, 16, 32
Low-pass filter Coefficient h1(0) Register Channel 1	PDMm_PDLFCH10RCH1	0000_1FF8h	0268h	8, 16, 32
Low-pass filter Coefficient h1(1) Register Channel 1	PDMm_PDLFCH101RCH1	0000_000Ah	026Ch	8, 16, 32
Low-pass filter Coefficient h1(2) Register Channel 1	PDMm_PDLFCH102RCH1	0000_1FF0h	0270h	8, 16, 32
Low-pass filter Coefficient h1(3) Register Channel 1	PDMm_PDLFCH103RCH1	0000_0018h	0274h	8, 16, 32
Low-pass filter Coefficient h1(4) Register Channel 1	PDMm_PDLFCH104RCH1	0000_1FDCh	0278h	8, 16, 32
Low-pass filter Coefficient h1(5) Register Channel 1	PDMm_PDLFCH105RCH1	0000_0034h	027Ch	8, 16, 32
Low-pass filter Coefficient h1(6) Register Channel 1	PDMm_PDLFCH106RCH1	0000_1FB3h	0280h	8, 16, 32
Low-pass filter Coefficient h1(7) Register Channel 1	PDMm_PDLFCH107RCH1	0000_0076h	0284h	8, 16, 32
Low-pass filter Coefficient h1(8) Register Channel 1	PDMm_PDLFCH108RCH1	0000_1F2Eh	0288h	8, 16, 32
Low-pass filter Coefficient h1(9) Register Channel 1	PDMm_PDLFCH109RCH1	0000_0289h	028Ch	8, 16, 32
Low-pass filter Coefficient h1(10) Register Channel 1	PDMm_PDLFCH110RCH1	0000_0289h	0290h	8, 16, 32
Low-pass filter Coefficient h1(11) Register Channel 1	PDMm_PDLFCH111RCH1	0000_1F2Eh	0294h	8, 16, 32
Low-pass filter Coefficient h1(12) Register Channel 1	PDMm_PDLFCH112RCH1	0000_0076h	0298h	8, 16, 32
Low-pass filter Coefficient h1(13) Register Channel 1	PDMm_PDLFCH113RCH1	0000_1FB3h	029Ch	8, 16, 32
Low-pass filter Coefficient h1(14) Register Channel 1	PDMm_PDLFCH114RCH1	0000_0034h	02A0h	8, 16, 32
Low-pass filter Coefficient h1(15) Register Channel 1	PDMm_PDLFCH115RCH1	0000_1FDCh	02A4h	8, 16, 32
Low-pass filter Coefficient h1(16) Register Channel 1	PDMm_PDLFCH116RCH1	0000_0018h	02A8h	8, 16, 32
Low-pass filter Coefficient h1(17) Register Channel 1	PDMm_PDLFCH117RCH1	0000_1FF0h	02ACh	8, 16, 32
Low-pass filter Coefficient h1(18) Register Channel 1	PDMm_PDLFCH118RCH1	0000_000Ah	02B0h	8, 16, 32
Low-pass filter Coefficient h1(19) Register Channel 1	PDMm_PDLFCH119RCH1	0000_1FF8h	02B4h	8, 16, 32
Sound Detection Lower Threshold Register Channel 1	PDMm_PDSDLTRCH1	0000_0000h	02B8h	8, 16, 32
Sound Detection Upper Threshold Register Channel 1	PDMm_PDSDUTRCH1	0000_0000h	02BCh	8, 16, 32
Data Buffer Control Register Channel 1	PDMm_PDDBCRCH1	0000_0000h	02C0h	8, 16, 32
Short Circuit Threshold Setting Register Channel 1	PDMm_PDSCTSRCH1	0000_0000h	02C4h	8, 16, 32
Overvoltage Lower Threshold Register Channel 1	PDMm_PDOVLTRCH1	0000_0000h	02C8h	8, 16, 32
Overvoltage Upper Threshold Register Channel 1	PDMm_PDOVUTRCH1	0000_0000h	02CCh	8, 16, 32
Reserve	-	-	02D0h to 02DFh	-
Data Read Control Register Channel 1	PDMm_PDDRCRCH1	0000_0000h	02E0h	8, 16, 32
Data Clear Register Channel 1	PDMm_PDDCRCH1	0000_0000h	02E4h	8, 16, 32
Data Read Register Channel 1	PDMm_PDDRRCH1	0000_0000h	02E8h	8, 16, 32
Data Status Register Channel 1	PDMm_PDDSRCH1	0000_0000h	02ECh	8, 16, 32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
Reserve	-	-	02F0h to 02FFh	-
Software Start Trigger Register Channel 2	PDMm_PDSTRTRCH2	0000_0000h	0300h	8, 16, 32
Software Stop Trigger Register Channel 2	PDMm_PDSTPTRCH2	0000_0000h	0304h	8, 16, 32
Software Change Trigger Register Channel 2	PDMm_PDCHGTRCH2	0000_0000h	0308h	8, 16, 32
Interrupt Control Register Channel 2	PDMm_PDICRCH2	0000_0000h	030Ch	8, 16, 32
Status Detection Control Register Channel 2	PDMm_PDSDCRCH2	0000_0000h	0310h	8, 16, 32
Status Register Channel 2	PDMm_PDSRCH2	0000_0000h	0314h	8, 16, 32
Status Clear Register Channel 2	PDMm_PDSCRCH2	0000_0000h	0318h	8, 16, 32
Reserve	-	-	031Ch to 031Fh	-
Mode Setting Register Channel 2	PDMm_PDMDSRCH2	0000_0000h	0320h	8, 16, 32
Sinc filter Control Register Channel 2	PDMm_PDSFCRCH2	057C_0000h	0324h	8, 16, 32
High-pass filter Coefficient s(0) Register Channel 2	PDMm_PDHFCS0RCH2	0000_3F61h	0328h	8, 16, 32
High-pass filter Coefficient k(1) Register Channel 2	PDMm_PDHFCK1RCH2	0000_3EC1h	032Ch	8, 16, 32
High-pass filter Coefficient h(0) Register Channel 2	PDMm_PDHFCH0RCH2	0000_4000h	0330h	8, 16, 32
High-pass filter Coefficient h(1) Register Channel 2	PDMm_PDHFCH1RCH2	0000_C000h	0334h	8, 16, 32
Compensation filter Coefficient h(0) Register Channel 2	PDMm_PDCFCH00RCH2	0000_1FE8h	0338h	8, 16, 32
Compensation filter Coefficient h(1) Register Channel 2	PDMm_PDCFCH01RCH2	0000_0039h	033Ch	8, 16, 32
Compensation filter Coefficient h(2) Register Channel 2	PDMm_PDCFCH02RCH2	0000_003Ch	0340h	8, 16, 32
Compensation filter Coefficient h(3) Register Channel 2	PDMm_PDCFCH03RCH2	0000_1E56h	0344h	8, 16, 32
Compensation filter Coefficient h(4) Register Channel 2	PDMm_PDCFCH04RCH2	0000_01DCh	0348h	8, 16, 32
Compensation filter Coefficient h(5) Register Channel 2	PDMm_PDCFCH05RCH2	0000_06E1h	034Ch	8, 16, 32
Compensation filter Coefficient h(6) Register Channel 2	PDMm_PDCFCH06RCH2	0000_01DCh	0350h	8, 16, 32
Compensation filter Coefficient h(7) Register Channel 2	PDMm_PDCFCH07RCH2	0000_1E56h	0354h	8, 16, 32
Compensation filter Coefficient h(8) Register Channel 2	PDMm_PDCFCH08RCH2	0000_003Ch	0358h	8, 16, 32
Compensation filter Coefficient h(9) Register Channel 2	PDMm_PDCFCH09RCH2	0000_0039h	035Ch	8, 16, 32
Compensation filter Coefficient h(10) Register Channel 2	PDMm_PDCFCH10RCH2	0000_1FE8h	0360h	8, 16, 32
Low-pass filter Coefficient h0(10) Register Channel 2	PDMm_PDLFCH010RCH2	0000_0400h	0364h	8, 16, 32
Low-pass filter Coefficient h1(0) Register Channel 2	PDMm_PDLFCH10RCH2	0000_1FF8h	0368h	8, 16, 32
Low-pass filter Coefficient h1(1) Register Channel 2	PDMm_PDLFCH101RCH2	0000_000Ah	036Ch	8, 16, 32
Low-pass filter Coefficient h1(2) Register Channel 2	PDMm_PDLFCH102RCH2	0000_1FF0h	0370h	8, 16, 32
Low-pass filter Coefficient h1(3) Register Channel 2	PDMm_PDLFCH103RCH2	0000_0018h	0374h	8, 16, 32
Low-pass filter Coefficient h1(4) Register Channel 2	PDMm_PDLFCH104RCH2	0000_1FDCh	0378h	8, 16, 32
Low-pass filter Coefficient h1(5) Register Channel 2	PDMm_PDLFCH105RCH2	0000_0034h	037Ch	8, 16, 32
Low-pass filter Coefficient h1(6) Register Channel 2	PDMm_PDLFCH106RCH2	0000_1FB3h	0380h	8, 16, 32
Low-pass filter Coefficient h1(7) Register Channel 2	PDMm_PDLFCH107RCH2	0000_0076h	0384h	8, 16, 32
Low-pass filter Coefficient h1(8) Register Channel 2	PDMm_PDLFCH108RCH2	0000_1F2Eh	0388h	8, 16, 32
Low-pass filter Coefficient h1(9) Register Channel 2	PDMm_PDLFCH109RCH2	0000_0289h	038Ch	8, 16, 32
Low-pass filter Coefficient h1(10) Register Channel 2	PDMm_PDLFCH110RCH2	0000_0289h	0390h	8, 16, 32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]*1
Low-pass filter Coefficient h1(11) Register Channel 2	PDMm_PDLFCH111RCH2	0000_1F2Eh	0394h	8, 16, 32
Low-pass filter Coefficient h1(12) Register Channel 2	PDMm_PDLFCH112RCH2	0000_0076h	0398h	8, 16, 32
Low-pass filter Coefficient h1(13) Register Channel 2	PDMm_PDLFCH113RCH2	0000_1FB3h	039Ch	8, 16, 32
Low-pass filter Coefficient h1(14) Register Channel 2	PDMm_PDLFCH114RCH2	0000_0034h	03A0h	8, 16, 32
Low-pass filter Coefficient h1(15) Register Channel 2	PDMm_PDLFCH115RCH2	0000_1FDCh	03A4h	8, 16, 32
Low-pass filter Coefficient h1(16) Register Channel 2	PDMm_PDLFCH116RCH2	0000_0018h	03A8h	8, 16, 32
Low-pass filter Coefficient h1(17) Register Channel 2	PDMm_PDLFCH117RCH2	0000_1FF0h	03ACh	8, 16, 32
Low-pass filter Coefficient h1(18) Register Channel 2	PDMm_PDLFCH118RCH2	0000_000Ah	03B0h	8, 16, 32
Low-pass filter Coefficient h1(19) Register Channel 2	PDMm_PDLFCH119RCH2	0000_1FF8h	03B4h	8, 16, 32
Sound Detection Lower Threshold Register Channel 2	PDMm_PDSDLTRCH2	0000_0000h	03B8h	8, 16, 32
Sound Detection Upper Threshold Register Channel 2	PDMm_PDSDUTRCH2	0000_0000h	03BCh	8, 16, 32
Data Buffer Control Register Channel 2	PDMm_PDDBCRCH2	0000_0000h	03C0h	8, 16, 32
Short Circuit Threshold Setting Register Channel 2	PDMm_PDSCTSRCH2	0000_0000h	03C4h	8, 16, 32
Overvoltage Lower Threshold Register Channel 2	PDMm_PDOVLTRCH2	0000_0000h	03C8h	8, 16, 32
Overvoltage Upper Threshold Register Channel 2	PDMm_PDOVUTRCH2	0000_0000h	03CCh	8, 16, 32
Reserve	-	-	03D0h to 03DFh	-
Data Read Control Register Channel 2	PDMm_PDDRCRCH2	0000_0000h	03E0h	8, 16, 32
Data Clear Register Channel 2	PDMm_PDDCRCH2	0000_0000h	03E4h	8, 16, 32
Data Read Register Channel 2	PDMm_PDDRRCH2	0000_0000h	03E8h	8, 16, 32
Data Status Register Channel 2	PDMm_PDDSRCH2	0000_0000h	03ECh	8, 16, 32

Note 1. The read access size is fixed to 32 bits.

8.7.2.2 Register Description

The prefix (PDMm_) of the register names is omitted in this and subsequent sections.

8.7.2.2.1 Channel Software Start Trigger Register (PDMm_PDCSTRTR) (m = 0, 1)

Access Size : 8, 16, 32 bits
Address : <PDMm_base> + 0000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	STRTRG[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	STRTRG[2:0]	0h	R0W	The read value is always 0b. Channel n start trigger 0b: Do nothing 1b: Start channel n

Note: The read access size is fixed to 32 bits.

8.7.2.2.2 Channel Software Stop Trigger Register (PDMm_PDCSTPTR) (m = 0, 1)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	STPTRG[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	STPTRG[2:0]	0h	R0W	The read value is always 0b. Channel n stop trigger 0b: Do nothing 1b: Stop channel n

Note: The read access size is fixed to 32 bits.

8.7.2.2.3 Channel Software Change Trigger Register (PDMm_PDCCHGTR) (m = 0, 1)

Access Size : 8, 16, 32 bits
Address : <PDMm_base> + 0008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	CHGTRG[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0W	R0W	R0W	R0	R0	R0	R0	R0	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10 to 8	-	All0	R0W	Reserved. This register is readable and the initial value can be written to it.
7 to 3	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	CHGTRG[2:0]	0h	R0W	The read value is always 0b. Channel n change trigger 0b: Do nothing 1b: Change clock (PDM_CLKn) setting

Note: The read access size is fixed to 32 bits.

8.7.2.2.4 Channel Interrupt Control Register (PDMm_PDCICR) (m = 0, 1)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 000Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	IEDE[2:0]			-	-	-	-	-	IDRE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	RW	RW	RW	R0	R0	R0	R0	R0	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	ISDE[2:0]			-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	RW	RW	RW	R0	R0	R0	R0	R0	R0	R0	R0

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
26 to 24	IEDE[2:0]	0h	RW	Channel n error detection interrupt enable bit 0b: Don't allow to issue INT_PDM_ERRn interrupt 1b: Allow to issue INT_PDM_ERRn interrupt
23 to 19	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	IDRE[2:0]	0h	RW	Channel n data reception interrupt enable bit 0b: Don't allow to issue INT_PDM_DATn interrupt 1b: Allow to issue INT_PDM_DATn interrupt
15 to 11	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10 to 8	ISDE[2:0]	0h	RW	Channel n sound detection interrupt enable bit 0b: Don not allow to issue INT_PDM_SDET interrupt when the sound for channel n is detected. 1b: Allow to issue INT_PDM_SDET interrupt when the sound for channel n is detected.
7 to 0	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Note: The read access size is fixed to 32 bits.

8.7.2.2.5 Channel Status Register (PDMm_PDCSR) (m = 0, 1)

Access Size : 8, 16, 32 bits
Address : <PDMm_base> + 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	EDF[2:0]			-	-	-	-	-	DRF[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R	R	R	R0	R0	R0	R0	R0	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	SDF[2:0]			-	-	-	-	-	STATE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R	R	R	R0	R0	R0	R0	R0	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
26 to 24	EDF[2:0]	0h	R	Channel n error detection flag 0b: Indicates that no data is detected. 1b: Indicates that an error is detected. Refer to the PDSRCHn register for details of errors.
23 to 19	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	DRF[2:0]	0h	R	Channel n data reception flag 0b: Indicates that the number of data stored in buffer does not exceed the threshold. 1b: Indicates that the number of data stored in buffer exceeded the threshold.
15 to 11	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10 to 8	SDF[2:0]	0h	R	Channel n sound detection flag 0b: Indicates that the sound which exceeded the threshold is not detected. 1b: Indicates that the sound which exceeded the threshold is detected.
7 to 3	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	STATE[2:0]	0h	R	Channel n state 0b: Channel n stop 1b: Channel n in operation

Note: The read access size is fixed to 32 bits.

EDFn

[Condition to become “1b”]

- When any error occurs. Refer to the PDSRCHn register for the kind of error.

[Condition to become “0b”]

- When all error flags are cleared by writing the PDSCRCHn, this flag is also cleared.

DRFn

Refer to **8.7.3.7 Data Buffer** for details.

[Condition to become “1b”]

- When PDCDRCR.DATREn = 1b and PDDSRCHn.DATNUM[7:0] is greater than or equal to a threshold which is configured by PDDBCRCHn.DATRITHR[2:0].

[Condition to become “0b”]

- When PDCDRCR.DATREn = 0b or PDDSRCHn.DATNUM[7:0] is less than a threshold.

SDFn

Refer to **8.7.3.6 Sound Detection** for details.

[Conditions to become “1b”]

- When PDCSDCR.SDEn = 1b and a moving average filter result is greater than or equal to PDSDUTRCHn.SDETU[19:0].
- When PDCSDCR.SDEn = 1b and a moving average filter result is less than or equal to PDSDLTRCHn.SDETL[19:0].

[Condition to become “0b”]

- When writing 1b to a flag clear register bit (PDCSCR.SDFCn).

STATEn

Refer to **8.7.3.2 Channel Start and Channel Stop** for details.

[Condition to become “1b”]

- When “1b” is written to PDCSTRTR.STRTRGn.

[Condition to become “0b”]

- When “1b” is written to PDCSTPTR.STPTRGn.

8.7.2.2.6 Channel Status Clear Register (PDMm_PDCSCR) (m = 0, 1)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0014h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	SDFC[2:0]			-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0W	R0W	R0W	R0	R0	R0	R0	R0	R0	R0	R0

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10 to 8	SDFC[2:0]	0h	R0W	The read value is always 0b. Channel n sound detection flag clear 0b: Do nothing 1b: Clear PDCSR.SDFn
7 to 0	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Note: The read access size is fixed to 32 bits.

8.7.2.2.7 Channel Sound Detection Control Register (PDMm_PDCSDCR) (m = 0, 1)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SDE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDE[2:0]	0h	RW	Channel n sound detection enable bit 0b: Do not allow to detect sound 1b: Allow to detect sound

Note: The read access size is fixed to 32 bits.

8.7.2.2.8 Channel Data Read Control Register (PDMm_PDCDRCR) (m = 0, 1)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0024h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	DATRE[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	DATRE[2:0]	0h	RW	Channel n data read enable bit 0b: Do not allow to read data from data buffer 1b: Allow to read data from data buffer

Note: The read access size is fixed to 32 bits.

8.7.2.2.9 Channel Data Clear Register (PDMm_PDCDCR) (m = 0, 1)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0028h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	DATC[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	DATC[2:0]	0h	R0W	The read value is always 0b. Channel n data clear 0b: Do nothing 1b: Clear data

Note: The read access size is fixed to 32 bits.

8.7.2.2.10 Version Register (PDMm_PDVR) (m = 0, 1)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0080h

Initial Value : 0000_0010h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	VER[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R0	R0	R0	R0	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	VER[11:0]	10h	R	IP version is shown.

Note: The read access size is fixed to 32 bits.

8.7.2.2.11 Software Start Trigger Register Channel n (PDMm_PDSTRTRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0100h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STRTRG
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	STRTRG	0h	R0W	The read value is always 0b. Start trigger 0b: Do nothing 1b: Start the channel

Note: The read access size is fixed to 32 bits.

8.7.2.2.12 Software Stop Trigger Register Channel n (PDMm_PDSTPTRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0104h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STPTRG
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	STPTRG	0h	R0W	The read value is always 0b. Stop trigger 0b: Do nothing 1b: Stop the channel

Note: The read access size is fixed to 32 bits.

8.7.2.2.13 Software Change Trigger Register Channel n (PDMm_PDCHGTRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0108h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CHGTRG
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0W	R0	R0	R0	R0	R0	R0	R0	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	-	0h	R0W	Reserved This register is readable and the initial value can be written to it.
7 to 1	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	CHGTRG	0h	R0W	The read value is always 0b. Change trigger 0b: Do nothing 1b: Change settings of PDM_CLKn and sinc filter

Note: The read access size is fixed to 32 bits.

8.7.2.2.14 Interrupt Control Register Channel n (PDMm_PDICRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 010Ch + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	IEDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	IDRE	ISDE	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	R0

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	IEDE	0h	RW	Error detection interrupt enable bit 0b: Do not allow to issue INT_PDM_ERRn interrupt 1b: Allow to issue INT_PDM_ERRn interrupt
15 to 3	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	IDRE	0h	RW	Data reception interrupt enable bit 0b: Do not allow to issue INT_PDM_DATn interrupt 1b: Allow to issue INT_PDM_DATn interrupt
1	ISDE	0h	RW	Sound detection interrupt enable bit 0b: Do not allow to issue INT_PDM_SDET interrupt 1b: Allow to issue INT_PDM_SDET interrupt
0	-	0h	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Note: The read access size is fixed to 32 bits.

8.7.2.2.15 Status Detection Control Register Channel n (PDMm_PDSDCRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0110h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	BFOWDE	-	-	-	-	-	-	-	-	OVUDE	OVLDE	SCDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SDE	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	R0

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27	BFOWDE	0h	RW	Buffer overwriting detection enable bit 0b: Do not allow to detect buffer overwriting. 1b: Allow to detect buffer overwriting.
26 to 19	-	All0	RW	Reserved This register is readable and the initial value can be written to it.
18	OVUDE	0h	RW	Overvoltage upper limit exceeded detection enable bit 0b: Do not allow to detect data exceeding the upper limit. 1b: Allow to detect data exceeding the upper limit
17	OVLDE	0h	RW	Overvoltage lower limit exceeded detection enable bit 0b: Do not allow to detect data falling below the lower limit. 1b: Allow to detect data falling below the lower limit
16	SCDE	0h	RW	Short circuit detection enable bit 0b: Do not allow to detect the PDM_DATn pin short circuit 1b: Allow to detect the PDM_DATn pin short circuit
15 to 2	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SDE	0h	RW	Sound detection enable bit 0b: Do not allow to detect sound 1b: Allow to detect sound
0	-	0h	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Note: The read access size is fixed to 32 bits.

8.7.2.2.16 Status Register Channel n (PDMm_PDSRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0114h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	BFOWDF	-	-	-	-	-	-	-	-	OVUDF	OVLDF	SCDF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	DRF	SDF	STATE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27	BFOWDF	0h	R	Buffer overwriting detection flag 0b: Indicates that buffer overwriting does not occur. 1b: Indicates that buffer overwriting occurred.
26 to 19	-	All0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18	OVUDF	0h	R	Overvoltage upper limit exceeded detection flag 0b: Indicates that the data is not above the upper limit 1b: Indicates that the data has exceeded the upper limit
17	OVLDF	0h	R	Overvoltage lower limit exceeded detection flag 0b: Indicates that the data is not below the lower limit 1b: Indicates that the data has falling below the lower limit
16	SCDF	0h	R	Short circuit detection flag 0b: Indicates that the PDM_DATn pin short circuit is not detected 1b: Indicates that a short circuit on the PDM_DATn pin has been detected
15 to 3	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	DRF	0h	R	Data reception flag 0b: Indicates that the number of data stored in buffer is less than a threshold 1b: Indicates that the number of data stored in buffer is greater than or equal to a threshold
1	SDF	0h	R	Sound detection flag 0b: Indicates that the sound is not detected 1b: Indicates that the sound is detected
0	STATE	0h	R	State 0b: Channel stop 1b: Channel in operation

Note: The read access size is fixed to 32 bits.

BFOWDF

Refer to **8.7.3.7 Data Buffer** for details.

[Condition to become “1b”]

- When PDSDCRCHn.BFOWDE = 1b and data buffer is full and data is written to the buffer without reading.

[Condition to become “0b”]

- When writing 1b to a flag clear register bit (PDSCRCHn.BFOWDFC).

OVUDF

Refer to **8.7.3.9 Overvoltage Detection** for details.

[Condition to become “1b”]

- When PDSDCRCHn.OVUDE = 1b and a clipped sinc filter result is larger than an upper limit (PDOVUTRCHn.OVDU[19:0]).

[Condition to become “0b”]

- When writing 1b to a flag clear register bit (PDSCRCHn.OVUDFC).

OVLDF

Refer to **8.7.3.9 Overvoltage Detection** for details.

[Condition to become “1b”]

- When PDSDCRCHn.OVLDE = 1b and a clipped sinc filter result is smaller than a lower limit (PDOVLTRCHn.OVDL[19:0]).

[Condition to become “0b”]

- When writing 1b to a flag clear register bit (PDSCRCHn.OVLDFC).

SCDF

Refer to **8.7.3.8 Short-Circuit Detection** for details.

[Conditions to become “1b”]

- When PDSDCRCHn.SCDE = 1b and 1b is continuously input to the PDM_DATn pin, exceeding the number of times set in PDSCTSRCHn.SCDH[12:0].
- When PDSDCRCHn.SCDE = 1b and 0b is continuously input to the PDM_DATn pin, exceeding the number of times set in PDSCTSRCHn.SCDL[12:0].

[Condition to become “0b”]

- When writing 1b to a flag clear register bit (PDSCRCHn.SCDFC).

DRF

Refer to **8.7.3.7 Data Buffer** for details.

[Condition to become “1b”]

- When PDDRCRCHn.DATRE = 1b and PDDSRCHn.DATNUM[7:0] is greater than or equal to a threshold which is configured by PDDBCRCHn.DATRITHR[2:0].

[Condition to become “0b”]

- When PDDRCRCHn.DATRE = 0b or PDDSRCHn.DATNUM[7:0] is less than a threshold.

SDF

Refer to **8.7.3.6 Sound Detection** for details.

[Conditions to become “1b”]

- When PDSDCRCHn.SDE = 1b and a moving average filter result is PDSDUTRCHn.SDETU[19:0] or greater.
- When PDSDCRCHn.SDE = 1b and a moving average filter result is PDSDLTRCHn.SDETL[19:0] or less.

[Condition to become “0b”]

- When writing 1b to a flag clear register bit (PDSCRCHn.SDFC).

STATE

Refer to **8.7.3.2 Channel Start and Channel Stop** for details.

[Condition to become “1b”]

- When “1b” is written to PDSTRTRCHn.STRTRG.

[Condition to become “0b”]

- When “1b” is written to PDSTPTRCHn.STPTRG.

8.7.2.2.17 Status Clear Register Channel n (PDMm_PDSCRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0118h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	BFOWDFC	-	-	-	-	-	-	-	-	OVUDFC	OVLDFC	SCDFC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SDFC	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W	R0

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27	BFOWDFC	0h	R0W	The read value is always 0b. Buffer overwriting detection flag clear 0b: Do nothing 1b: Clear PDSRCHn.BFOWDF
26 to 19	-	All0	R0W	Reserved This register is readable and the initial value can be written to it.
18	OVUDFC	0h	R0W	The read value is always 0b. Overvoltage upper limit exceeded detection flag clear 0b: Do nothing 1b: Clear PDSRCHn.OVUDF
17	OVLDFC	0h	R0W	The read value is always 0b. Overvoltage lower limit exceeded detection flag clear 0b: Do nothing 1b: Clear PDSRCHn.OVLDF
16	SCDFC	0h	R0W	The read value is always 0b. Short circuit detection flag clear 0b: Do nothing 1b: Clear PDSRCHn.SCDF
15 to 2	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	SDFC	0h	R0W	The read value is always 0b. Sound detection flag clear 0b: Do nothing 1b: Clear PDSRCHn.SDF
0	-	0h	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Note: The read access size is fixed to 32 bits.

8.7.2.2.18 Mode Setting Register Channel n (PDMm_PDMDSRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0120h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DBIS[3:0]				-	-	SDMAMD[1:0]		-	-	-	-	-	-	LFIS[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R0	R0	RW	RW	R0	R0	R0	RW	R0	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	CFIS[1:0]		-	-	HFIS[1:0]		-	SFMD[2:0]			-	-	-	INPSEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	RW	RW	RW	R0	RW	RW	RW	R0	RW	RW	RW	R0	R0	R0	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	DBIS[3:0]	0h	RW	Data buffer input shift setting 0000b: 20-bit mode, {1{S},[18:0]} 0001b: 20-bit mode, {2{S},[18:1]} 0010b: 20-bit mode, {3{S},[18:2]} 0011b: 20-bit mode, {4{S},[18:3]} (0100b-0111b: Setting prohibited) 1000b: 16-bit mode, {S,D[18:4]} 1001b: 16-bit mode, {S,D[17:3]} 1010b: 16-bit mode, {S,D[16:2]} 1011b: 16-bit mode, {S,D[15:1]} 1100b: 16-bit mode, {S,D[14:0]} (1101b-1111b: Setting prohibited) *S: Sign bit
27, 26	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25, 24	SDMAMD[1:0]	0h	RW	Moving average mode for sound detection data 00b: 1-order (filter is skipped) (default) 01b: 2-order 10b: 4-order 11b: Setting prohibited
23 to 21	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	-	0h	RW	Reserved This register is readable and the initial value can be written to it.
19	-	0h	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18	-	0h	RW	Reserved This register is readable and the initial value can be written to it.
17, 16	LFIS[1:0]	0h	RW	Low-pass (half-band decimation) filter input shift setting 00b: No shift 01b: 1-bit right shift 10b: 2-bit right shift 11b: 3-bit right shift
15	-	0h	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14	-	0h	RW	Reserved This register is readable and the initial value can be written to it.
13, 12	CFIS[1:0]	0h	RW	Compensation filter input shift setting 00b: No shift 01b: 1-bit right shift 10b: 2-bit right shift 11b: 3-bit right shift

Bit	Bit Name	Initial Value	R/W	Description
11	-	0h	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	-	0h	RW	Reserved This register is readable and the initial value can be written to it.
9, 8	HFIS[1:0]	0h	RW	High-pass filter input shift setting 00b: No shift 01b: 1-bit right shift 10b: 2-bit right shift 11b: 3-bit right shift
7	-	0h	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 4	SFMD[2:0]	0h	RW	Sinc filter mode setting 001b: 1-order 010b: 2-order 011b: 3-order Other: 4-order (default)
3 to 1	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	INPSEL	0h	RW	Input data select 0b: Rise-edge data of channel n 1b: Fall-edge data of channel n-1 (In case of n = 0, fall-edge data of channel 2 is selected.)

Note: The read access size is fixed to 32 bits.

8.7.2.2.19 Sinc Filter Control Register Channel n (PDMm_PDSFCRCHn) (m = 0 to 1, n = 0 to 2)

See **Table 8.7-8** for the setting values of SINCDEC[7:0] and SINCRNG[4:0].

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0124h + n × 0100h

Initial Value : 057C_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	SINCRNG[4:0]					SINCDEC[7:0]							
Initial Value	0	0	0	0	0	1	0	1	0	1	1	1	1	1	0	0
R/W	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	CKDIV[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 24	SINCRNG[4:0]	5h	RW	Sinc filter output valid range 00000b: {S, [32:14]} 10000b: {S, [16:0],00b} 00001b: {S, [31:13]} 10001b: {S, [15:0],000b} 00010b: {S, [30:12]} 10010b: {S, [14:0],0000b} 00011b: {S, [29:11]} 10011b: {S, [13:0],00000b} 00100b: {S, [28:10]} 10100b: {S, [12:0],000000b} 00101b: {S, [27:9]} (default) 10101b: {S, [11:0],0000000b} 00110b: {S, [26:8]} 10110b: {S, [10:0],00000000b} 00111b: {S, [25:7]} 10111b: {S, [9:0],000000000b} 01000b: {S, [24:6]} 11000b: {S, [8:0],0000000000b} 01001b: {S, [23:5]} 11001b: {S, [7:0],00000000000b} 01010b: {S, [22:4]} 11010b: {S, [6:0],000000000000b} 01011b: {S, [21:3]} 11011b: {S, [5:0],0000000000000b} 01100b: {S, [20:2]} 11100b: {S, [4:0],00000000000000b} 01101b: {S, [19:1]} 11101b: {S, [3:0],000000000000000b} 01110b: {S, [18:0]} 11110b: {S, [2:0],0000000000000000b} 01111b: {S, [17:0], 0b} 11111b: {S, [1:0],00000000000000000b} (prohibited) S: Signed bit. When the overflow occurs, 7FFFFh is output. When the underflow occurs, 80000h is output.
23 to 16	SINCDEC[7:0]	7Ch	RW	Sinc filter decimation ratio M indicates the decimation ratio. Decimation ratio M is set as follows. M = SINCDEC + 1 00h: M = 1 (prohibited) 01h: M = 2 (prohibited) 02h: M = 3 (prohibited) 03h: M = 4 … 7Ch: M = 125 (default) ... FFh: M = 256 Setting a value less than M = 4 is prohibited. If such a value is set, operation of the sinc filter is not guaranteed. The setting should satisfy D*M > 12. Otherwise, operation of other filters is not guaranteed. (D: PDM_CLKn's division ratio)
15 to 4	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	CKDIV[3:0]	0h	RW	PDM_CLKn's division ratio to core clock 0000b: 1/2 (default) 0001b: 1/4 0010b: 1/6 ... 1101b: 1/28 1110b: 1/30 1111b: 1/32

Note: The read access size is fixed to 32 bits.

8.7.2.2.20 High-Pass Filter Coefficient s(0) Register Channel n (PDMm_PDHFCS0RCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0128h + n × 0100h

Initial Value : 0000_3F61h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HFSm[15:0]															
Initial Value	0	0	1	1	1	1	1	1	0	1	1	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	HFSm[15:0]	3F61h	RW	High-pass filter coefficient s0 16-bit signed data (Fraction bit [13:0]) For the initial value, see Value after Reset of Each Coefficient

Note: The read access size is fixed to 32 bits.

● Value after Reset of Each Coefficient

Symbol Name	Coefficient Name	Value after Reset		Comment
		Hex	Decimal	
HFS0	s(0)	3F61h	0.99029541015625	Cut-off 100 Hz

8.7.2.2.21 High-Pass Filter Coefficient k(1) Register Channel n (PDMm_PDHFCK1RCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 012Ch + n × 0100h

Initial Value : 0000_3EC1h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HFKm[15:0]															
Initial Value	0	0	1	1	1	1	1	0	1	1	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	HFKm[15:0]	3EC1h	RW	High-pass filter coefficient k1 16-bit signed data (Fraction bit [13:0]) For the initial value, see Value after Reset of Each Coefficient

Note: The read access size is fixed to 32 bits.

● Value after Reset of Each Coefficient

Symbol Name	Coefficient Name	Value after Reset		Comment
		Hex	Decimal	
HFK1	k(1)	3EC1h	0.98052978515625	Cut-off 100 Hz

8.7.2.2.22 High-Pass Filter Coefficient h(p) Register Channel n (PDMm_PDHFCHpRCHn) (m = 0 to 1, n = 0 to 2, p = 0 to 1)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0130h + n × 0100h + p × 0004h

Initial Value : 0000_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HFH0[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	HFH0[15:0]	All x	RW	High-pass filter coefficient h(p) 16-bit signed data (Fraction bit [13:0]) For the initial value, see Value after Reset of Each Coefficient

Note: The read access size is fixed to 32 bits.

● Value after Reset of Each Coefficient

Symbol Name	Coefficient Name	Value after Reset		Comment
		Hex	Decimal	
HFH0	h(0)	4000h	1	
HFH1	h(1)	C000h	−1	

8.7.2.2.23 Compensation Filter Coefficient h(p) Register Channel n (PDMm_PDCFCHpRCHn) (m = 0 to 1, n = 0 to 2, p = 00 to 10)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0138h + n × 0100h + p × 0004h

Initial Value : 0000_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	CFH00[12:0]												
Initial Value	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	CFH00[12:0]	All x	RW	Compensation filter coefficients h(p) 13-bit signed data (Fraction bit [10:0]) For the initial value, see Value after Reset of Each Coefficient

Note: The read access size is fixed to 32 bits.

● Value after Reset of Each Coefficient

Symbol Name	Coefficient Name	Value after Reset		Comment
		Hex	Decimal	
CFH00	h(0)	1FE8h	−0.01171875	
CFH01	h(1)	0039h	0.02783203125	
CFH02	h(2)	003Ch	0.029296875	
CFH03	h(3)	1E56h	−0.2080078125	
CFH04	h(4)	01DCh	0.232421875	
CFH05	h(5)	06E1h	0.85986328125	
CFH06	h(6)	01DCh	0.232421875	
CFH07	h(7)	1E56h	−0.2080078125	
CFH08	h(8)	003Ch	0.029296875	
CFH09	h(9)	0039h	0.02783203125	
CFH10	h(10)	1FE8h	−0.01171875	

8.7.2.2.24 Low-Pass Filter Coefficient h0(10) Register Channel n (PDMm_PDLFCH010RCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0164h + n × 0100h

Initial Value : 0000_0400h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	LFH010[12:0]												
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	LFH010[12:0]	0400h	RW	Low-pass (half-band decimation) filter coefficient h0(m) 13-bit signed data (Fraction bit [10:0]) For the initial value, see Value after Reset of Each Coefficient

Note: The read access size is fixed to 32 bits.

● Value after Reset of Each Coefficient

Symbol Name	Coefficient Name	Value after Reset		Comment
		Hex	Decimal	
LFH010	h0(10)	0400h	0.5	Cut-off 7 kHz

8.7.2.2.25 Low-Pass Filter Coefficient h1(p) Register Channel n (PDMm_PDLFCH1pRCHn) (m = 0 to 1, n = 0 to 2, p = 0 to 19)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 0168h + n × 0100h + p × 0004h

Initial Value : 0000_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	LFH100[12:0]												
Initial Value	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	LFH100[12:0]	All x	RW	Low-pass (half-band decimation) filter coefficient h1(p) 13-bit signed data (Fraction bit [10:0]) For the initial value, see Value after Reset of Each Coefficient

Note: The read access size is fixed to 32 bits.

● Value after Reset of Each Coefficient

Symbol Name	Coefficient Name	Value after Reset		Comment
		Hex	Decimal	
LFH100	h1(0)	1FF8h	−0.00390625	Cut-off 7 kHz
LFH101	h1(1)	000Ah	0.0048828125	
LFH102	h1(2)	1FF0h	−0.0078125	
LFH103	h1(3)	0018h	0.01171875	
LFH104	h1(4)	1FDCh	−0.017578125	
LFH105	h1(5)	0034h	0.025390625	
LFH106	h1(6)	1FB3h	−0.03759765625	
LFH107	h1(7)	0076h	0.0576171875	
LFH108	h1(8)	1F2Eh	−0.1025390625	
LFH109	h1(9)	0289h	0.31689453125	
LFH110	h1(10)	0289h	0.31689453125	
LFH111	h1(11)	1F2Eh	−0.1025390625	
LFH112	h1(12)	0076h	0.0576171875	
LFH113	h1(13)	1FB3h	−0.03759765625	
LFH114	h1(14)	0034h	0.025390625	
LFH115	h1(15)	1FDCh	−0.017578125	
LFH116	h1(16)	0018h	0.01171875	
LFH117	h1(17)	1FF0h	−0.0078125	
LFH118	h1(18)	000Ah	0.0048828125	
LFH119	h1(19)	1FF8h	−0.00390625	

8.7.2.2.26 Sound Detection Lower Threshold Register Channel n (PDMm_PDSDLTRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 01B8h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	SDETL[19:16]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDETL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 0	SDETL[19:0]	0h	RW	Sound detection lower limit This bit sets the lower limit value (20-bit signed data) for sound detection.

Note: The read access size is fixed to 32 bits.

8.7.2.2.27 Sound Detection Upper Threshold Register Channel n (PDMm_PDSDUTRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 01BCh + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	SDETU[19:16]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDETU[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 0	SDETU[19:0]	0h	RW	Sound detection upper limit This bit sets the upper limit value (20-bit signed data) for sound detection.

Note: The read access size is fixed to 32 bits.

8.7.2.2.28 Data Buffer Control Register Channel n (PDMm_PDDBCRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 01C0h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	DATRITHR[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	R0	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	-	0h	RW	Reserved This register is readable and the initial value can be written to it.
3	-	0h	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	DATRITHR [2:0]	0h	RW	Data reception interrupt threshold 000b: Output interrupt when receiving 1 or more data 001b: Output interrupt when receiving 2 or more data 010b: Output interrupt when receiving 4 or more data 011b: Output interrupt when receiving 8 or more data others: Output interrupt when receiving 16 or more data

Note: The read access size is fixed to 32 bits.

8.7.2.2.29 Short Circuit Threshold Setting Register Channel n (PDMm_PDSCTSRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 01C4h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	SCDH[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SCDL[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	SCDH[12:0]	0h	RW	Short circuit detection High Continuous detection count Set the upper limit for the number of times "1" are input continuously to the PDM_DATn pin.
15 to 13	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	SCDL[12:0]	0h	RW	Continuous detection count Set the upper limit for the number of times "0" are input continuously to the PDM_DATn pin.

Note: The read access size is fixed to 32 bits.

8.7.2.2.30 Overvoltage Lower Threshold Register Channel n (PDMm_PDOVLTRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 01C8h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	OVDL[19:16]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVDL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 0	OVDL[19:0]	0h	RW	Overvoltage detection lower limit These bits set the lower limit of the active voltage value (20-bit signed data) for overvoltage detection.

Note: The read access size is fixed to 32 bits.

8.7.2.2.31 Overvoltage Upper Threshold Register Channel n (PDMm_PDOVUTRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 01CCh + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	OVDU[19:16]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVDU[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 0	OVDU[19:0]	0h	RW	Overvoltage detection upper limit These bits set the upper limit of the active voltage value (20-bit signed data) for overvoltage detection.

Note: The read access size is fixed to 32 bits.

8.7.2.2.32 Data Read Control Register Channel n (PDMm_PDDRCRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 01E0h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DATRE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	DATRE	0h	RW	Data read enable bit 0b: Do not allow to read data from buffer 1b: Allow to read data from buffer

Note: The read access size is fixed to 32 bits.

8.7.2.2.33 Data Clear Register Channel n (PDMm_PDDCRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 01E4h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DATC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	DATC	0h	R0W	The read value is always 0b. Data clear 0b: Do nothing 1b: Clear data

Note: The read access size is fixed to 32 bits.

8.7.2.2.34 Data Read Register Channel n (PDMm_PDDRRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 01E8h + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	DAT[19:16]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R0	R0	R0	R0	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DAT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 20	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 0	DAT[19:0]	0h	R	Data 20-bit signed sound data or 16-bit signed sound data is read out. In case of 20-bit mode (PDMDSRCHn.DBIS[3:0] = 0xxxb), the signed bit is [19]. In case of 16-bit mode (PDMDSRCHn.DBIS[3:0] = 1xxxb), the signed bits are [19:15]. Writing is invalid.

Note: The read access size is fixed to 32 bits.

8.7.2.2.35 Data Status Register Channel n (PDMm_PDDSRCHn) (m = 0 to 1, n = 0 to 2)

Access Size : 8, 16, 32 bits

Address : <PDMm_base> + 01ECh + n × 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	DATNUM[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0	R0	R0	R0	R0	R0	R0	R0	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All0	R0	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	DATNUM[7:0]	0h	R	Number of data stored in buffer When PDDRCRCHn.DATRE = 0b, 0b is read out. When PDDRCRCHn.DATRE = 1b, the number of data stored in buffer is read out. In case that buffer size is 64 stages, the maximum value is 63.

Note: The read access size is fixed to 32 bits.

8.7.2.3 Mirroring Register

The common registers except for the version register have mirroring registers in each channel. PDCSR.EDFn is a register which ORs all the error flags of the mirroring registers in each channel.

Use the common registers for synchronous control between channels.

Table 8.7-5 Mirroring Register

Common Register		Mirroring Register in Each Channel	
Register Name	Bit Name	Register Name	Bit Name
PDCSTRTR	STRTRGn	PDSTRTRCHn	STRTRG
PDCSTPTR	STPTRGn	PDSTPTRCHn	STPTRG
PDCCHGTR	CHGTRGn	PDCHGTRCHn	CHGTRG
PDCICR	ISDEn	PDICRCHn	ISDE
	IDREn		IDRE
	IEDEn		IEDE
PDCSR	STATEn	PDSRCHn	STATE
	SDFn		SDF
	DRFn		DRF
	EDFn		SCDF or OVLDF or OVUDF or BFOWDF
PDCSCR	SDFCn	PDSCRCHn	SDFC
PDCSDCR	SDEn	PDSDCRCHn	SDE
PDCDRCR	DATREn	PDDRCRCHn	DATRE
PDCDCR	DATCn	PDDCRCHn	DATC

8.7.3 Operation

This section describes the PDM basic control method. See **8.7.4.6 Setting PDM_CLKn (n = 0, 1, 2) Frequency and Sampling Frequency Generated by the Unit** for information on setting the PDM_CLKn (n = 0, 1, 2) frequency and the sampling frequency generated by the PDM unit, and see **8.7.4.7 DMA Transfer Method** for information on transferring data with the DMA master.

8.7.3.1 Clock Specifications

The frequency-divided core clock is generated and output to PDM_CLKn.

Table 8.7-6 lists the guaranteed clock frequency and clock mode in PDM-IF. Core clock (CCLK) and PCLK are asynchronous.

Table 8.7-6 Guaranteed Clock Frequency and Clock Mode

CCLK	PCLK PCLK_SFR	PDM_CLKn		
Frequency (MHz)	Frequency (MHz)	Frequency (MHz)	Supported Core Clock Division Ratio	Setting Value (PDSFCRCHn.CKDIV[3:0])
4.8	1 to 100	2.4	2	0000b
		1.2	4	0001b
		0.8	6	0010b
		0.6	8	0011b
		0.48	10	0100b
		0.4	12	0101b
		0.34	14	0110b
		0.3	16	0111b
		0.27	18	1000b
		0.24	20	1001b
		0.22	22	1010b
		0.2	24	1011b
		0.18	26	1100b
		0.17	28	1101b
		0.16	30	1110b
		0.15	32	1111b

8.7.3.2 Channel Start and Channel Stop

The channel start and the channel stop is controlled by the registers listed in **Table 8.7-7**.

Set the channels to be synchronized to 1 at the same time by using the common registers.

Set the channel start according to **8.7.4.1 Start Flow**.

Set the channel stop according to **8.7.4.2 Stop Flow**.

Table 8.7-7 Channel Start Trigger and Channel Stop Trigger

	Register Bit Name	
Category	Common Register	Channel Register
Channel start	PDCSTRTR.STRTRG0	PDSTRTRCH0.STRTRG
	PDCSTRTR.STRTRG1	PDSTRTRCH1.STRTRG
	PDCSTRTR.STRTRG2	PDSTRTRCH2.STRTRG
Channel stop	PDCSTPTR.STPTRG0	PDSTPTRCH0.STPTRG
	PDCSTPTR.STPTRG1	PDSTPTRCH1.STPTRG
	PDCSTPTR.STPTRG2	PDSTPTRCH2.STPTRG

8.7.3.3 PDM_CLKn's Frequency Change

The external microphone transitions the power mode according to the input clock frequency. PDM-IF enables the transition of microphone's power mode by changing PDM_CLKn's operating frequency which is output to the microphone when operating. The PDM_CLKn's operating frequency can be changed by using the PDSFCRCHn.CKDIV[3:0] bits. When the frequency is changed, the decimation ratio of the sinc filter also needs to be changed. Set PDSFCRCHn.SINCDEC[7:0] in accordance with the sampling frequency after the decimation. Furthermore, when the decimation ratio of the sinc filter is changed, its output range is changed. Set PDSFCRCHn.SINCRNG[4:0] together in accordance with the output range. Refer to **Table 8.7-8**, which lists example settings when the sampling frequency after the decimation is 30 kHz*[1].

For the method of calculating the set values of PDSFCRCHn.CKDIV[3:0], PDSFCRCHn.SINCDEC[7:0], and PDSFCRCHn.SINCRNG[4:0] for the expected sampling frequency of PDM_CLKn, refer to **8.7.4.6 Setting PDM_CLKn (n = 0, 1, 2) Frequency and Sampling Frequency Generated by the Unit**.

After changing the configuration of these registers, software needs to wait for the settling time of microphone and PDM-IF channel. Refer to **8.7.4.4 Low Power Mode Transition Flow** and **8.7.4.5 Normal Mode Transition Flow** for details.

Note 1. If the same filter coefficient has been set in filtering after the sinc filter, the characteristics of the filter changes with the change of the sampling frequency.

8.7.3.4 Filter

8.7.3.4.1 Data Clipping and Shifting between Filters

Voice sample data is clipped or shifted between filters. **Figure 8.7-2** shows this operation.

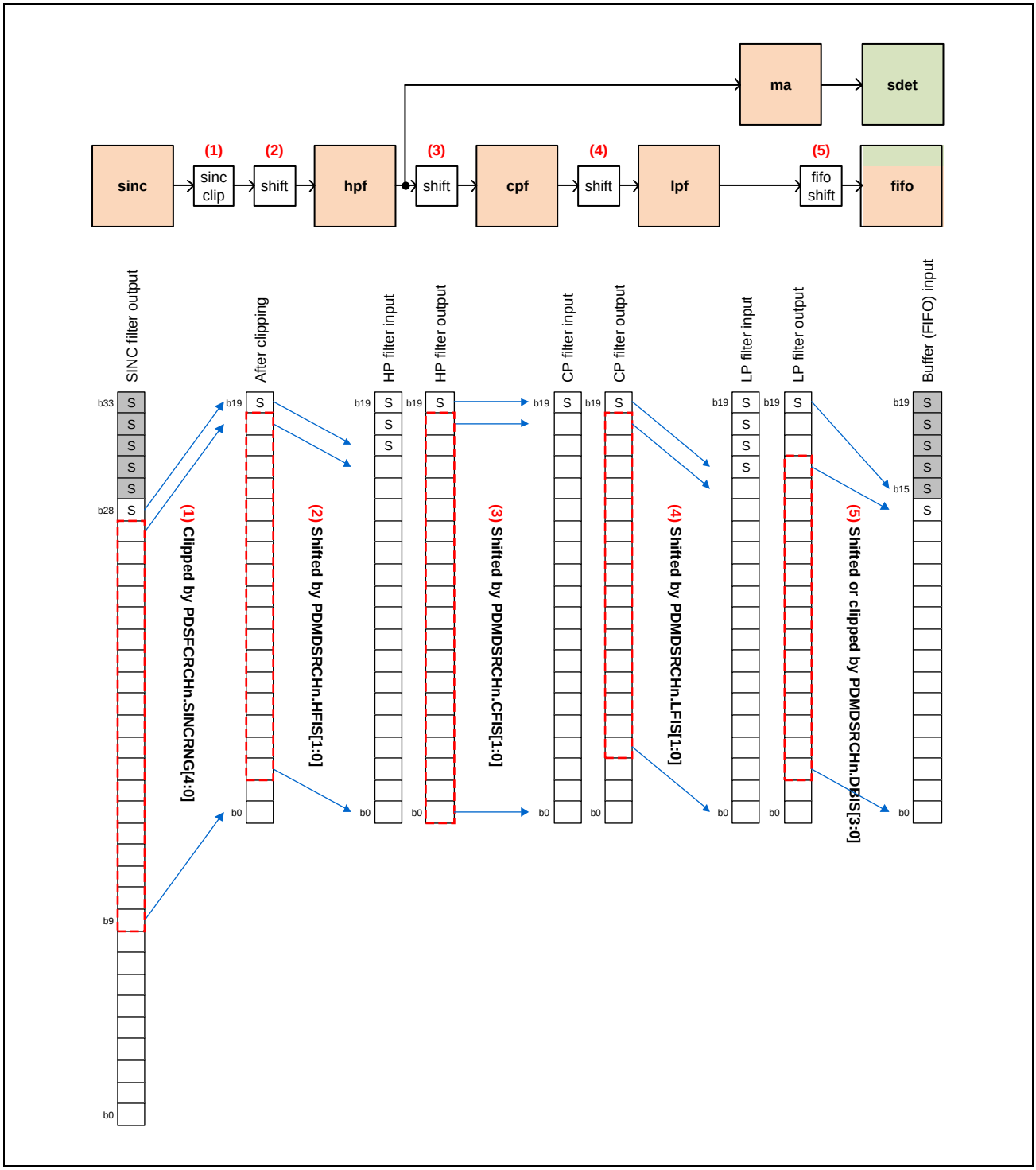


Figure 8.7-2 Data Clipping and Shifting between Filters

a) Sinc filter output clip

It is controlled by PDSFCRCHn.SINCRNG[4:0].

Table 8.7-8 lists the possible combinations of settings when the sampling frequency is 30 kHz.

Table 8.7-8 Sinc Filter Output Clip (1/2)

CCLK (MHz)	Sinc Filter Order	PDMDSRCHn.SFMD[2:0]	PDM_CLK's Division Ratio	PDSFCRCHn.CKDIV[3:0]	Sinc filter Decimation Ratio	PDSFCRCHn.SINCDEC[7:0]	Sinc Filter Output Effective Bits	Sinc Filter Output Clipping (20 bits)	PDSFCRCHn.SINCRNG[4:0]	Resolution	PDM_CLKn Frequency (MHz)	PDMIF Output Sampling Frequency (kHz)
4.8	4	0h	2	0h	40	27h	b22 to b0	b22 to b3	01011b	20	2.4	30
			4	1h	20	13h	b18 to b0	b18 to b0, 0b	01111b	19	1.2	30
			6	2h	13	Ch	b15 to b0	b15 to b0, 0000b	10010b	16	0.8	30.77
			8	3h	10	9h	b14 to b0	b14 to b0, 00000b	10011b	15	0.6	30
			10	4h	8	7h	b13 to b0	b13 to b0, 000000b	10010b	14	0.48	30
			12	5h	6	5h	b11 to b0	b11 to b0, 00000000b	10110b	12	0.4	33.33
			14	6h	6	5h	b11 to b0	b11 to b0, 00000000b	10110b	12	0.34	28.57
			16	7h	5	4h	b10 to b0	b10 to b0, 000000000b	10111b	11	0.3	30
			18	8h	4	3h	b9 to b0	b9 to b0, 0000000000b	11000b	10	0.27	33.33
			20	9h	4	3h	b9 to b0	b9 to b0, 0000000000b	11000b	10	0.24	30
			22	Ah	4	3h	b9 to b0	b9 to b0, 0000000000b	11000b	10	0.22	27.27
4.8	3	3h	2	0h	40	27h	b16 to b0	b16 to b0, 000b	10001b	20	2.4	30
			4	1h	20	13h	b13 to b0	b13 to b0, 000000b	10100b	19	1.2	30
			6	2h	13	Ch	b12 to b0	b12 to b0, 0000000b	10101b	16	0.8	30.77
			8	3h	10	9h	b10 to b0	b10 to b0, 000000000b	10111b	15	0.6	30
			10	4h	8	7h	b10 to b0	b10 to b0, 000000000b	10111b	14	0.48	30
			12	5h	6	5h	b8 to b0	b8 to b0, 0000000000b	11001b	12	0.4	33.33
			14	6h	6	5h	b8 to b0	b8 to b0, 0000000000b	11001b	12	0.34	28.57
			16	7h	5	4h	b7 to b0	b7 to b0, 00000000000b	11010b	11	0.3	30
			18	8h	4	3h	b7 to b0	b7 to b0, 00000000000b	11010b	10	0.27	33.33
			20	9h	4	3h	b7 to b0	b7 to b0, 00000000000b	11010b	10	0.24	30
			22	Ah	4	3h	b7 to b0	b7 to b0, 00000000000b	11010b	10	0.22	27.27
4.8	2	2h	2	0h	40	27h	b11 to b0	b11 to b0, 00000000b	10110b	20	2.4	30
			4	1h	20	13h	b9 to b0	b9 to b0, 0000000000b	11000b	19	1.2	30
			6	2h	13	Ch	b8 to b0	b8 to b0, 00000000000b	11001b	16	0.8	30.77
			8	3h	10	9h	b7 to b0	b7 to b0, 00000000000b	11010b	15	0.6	30
			10	4h	8	7h	b7 to b0	b7 to b0, 00000000000b	11010b	14	0.48	30
			12	5h	6	5h	b6 to b0	b6 to b0, 0000000000000b	11011b	12	0.4	33.33
			14	6h	6	5h	b6 to b0	b6 to b0, 0000000000000b	11011b	12	0.34	28.57
			16	7h	5	4h	b5 to b0	b5 to b0, 00000000000000b	11100b	11	0.3	30
			18	8h	4	3h	b5 to b0	b5 to b0, 00000000000000b	11100b	10	0.27	33.33
			20	9h	4	3h	b5 to b0	b5 to b0, 00000000000000b	11100b	10	0.24	30
			22	Ah	4	3h	b5 to b0	b5 to b0, 00000000000000b	11100b	10	0.22	27.27

Table 8.7-8 Sinc Filter Output Clip (2/2)

CCLK (MHz)	Sinc Filter Order	PDMDSRCHn. SFMD[2:0]	PDM_CLK's Division Ratio	PDSFCRCHn. CKDIV[3:0]	Sinc filter Decimation Ratio	PDSFCRCHn. SINCDEC[7:0]	Sinc Filter Output Effective Bits	Sinc Filter Output Clipping (20 bits)	PDSFCRCHn. SINCRNG[4:0]	Resolution	PDM_CLKn Frequency (MHz)	SINC Filter Output Sampling Frequency (kHz)
4.8	1	1h	2	0h	40	27h	b6 to b0	b6 to b0, 0000000000000b	11011b	20	2.4	30
			4	1h	20	13h	b5 to b0	b5 to b0, 00000000000000b	11100b	19	1.2	30
			6	2h	13	Ch	b4 to b0	b4 to b0, 000000000000000b	11101b	16	0.8	30.77
			8	3h	10	9h	b4 to b0	b4 to b0, 000000000000000b	11101b	15	0.6	30
			10	4h	8	7h	b4 to b0	b4 to b0, 000000000000000b	11101b	14	0.48	30
			12	5h	6	5h	b3 to b0	b3 to b0, 00000000000000000b	11110b	12	0.4	33.33
			14	6h	6	5h	b3 to b0	b3 to b0, 00000000000000000b	11110b	12	0.34	28.57
			16	7h	5	4h	b3 to b0	b3 to b0, 00000000000000000b	11110b	11	0.3	30
			18	8h	4	3h	b3 to b0	b3 to b0, 00000000000000000b	11110b	10	0.27	33.33
			20	9h	4	3h	b3 to b0	b3 to b0, 00000000000000000b	11110b	10	0.24	30
			22	Ah	4	3h	b3 to b0	b3 to b0, 00000000000000000b	11110b	10	0.22	27.27

An overflow or an underflow never occurs as long as the combinations which are shown in **Table 8.7-8** are set to PDMDSRCHn.SFMD[2:0], PDSFCRCHn.CKDIV[3:0], PDSFCRCHn.SINCDEC[7:0] and PDSFCRCHn.SINCRNG[4:0]. However, when other combinations are set to them, the overflow or the underflow might occur. When the overflow occurs, 7FFFFh is output. When the underflow occurs, 80000h is output. Use the overvoltage detection function if you want to know if the overflow or the underflow occurs.

b) High-pass filter input shift

The clipped sinc filter output is right-shifted and input to the high-pass filter.

It is controlled by PDMDSRCHn.HFIS[1:0].

Table 8.7-9 High-Pass Filter Input Shift

PDMDSRCHn.HFIS[1:0]	Function	Shifted Data*1
0h	No shift	S, b18-b0
1h	1-bit right-shift	S, S, b18-b1
2h	2-bit right-shift	S, S, S, b18-b2
3h	3-bit right-shift	S, S, S, S, b18-b3

Note 1. S: Signed bit (= bit 19)

c) Compensation filter input shift

The high-pass filter output is right-shifted and input to the compensation filter.

It is controlled by PDMDSRCHn.CFIS[1:0].

Table 8.7-10 Compensation Filter Input Shift

PDMDSRCHn.CFIS[1:0]	Function	Shifted Data*1
0h	No shift	S, b18-b0
1h	1-bit right-shift	S, S, b18-b1
2h	2-bit right-shift	S, S, S, b18-b2
3h	3-bit right-shift	S, S, S, S, b18-b3

Note 1. S: Signed bit (= bit 19)

d) Low-pass filter input shift

The compensation filter output is right-shifted and input to the low-pass (half-band decimation) filter.

It is controlled by PDMDSRCHn.LFIS[1:0].

Table 8.7-11 Low-Pass Filter Input Shift

PDMDSRCHn.LFIS[1:0]	Function	Shifted Data*1
0h	No shift	S, b18-b0
1h	1-bit right-shift	S, S, b18-b1
2h	2-bit right-shift	S, S, S, b18-b2
3h	3-bit right-shift	S, S, S, S, b18-b3

Note 1. S: Signed bit (= bit 19)

e) Buffer input shift or clip

The compensation filter output is right-shifted, or 16-bit data is clipped from the compensation filter output. Then, the shifted or clipped data is input to buffer.

It is controlled by PDMDSRCHn.DBIS[3:0].

Table 8.7-12 Buffer Input Shift or Clip

PDMDSRCHn.DBIS[3:0]	Function	Valid Data Width	Shifted or Clipped Data*1
0000b	No shift	20-bit	S, b18-b0
0001b	1-bit right-shift	20-bit	S, S, b18-b1
0010b	2-bit right-shift	20-bit	S, S, S, b18-b2
0011b	3-bit right-shift	20-bit	S, S, S, S, b18-b3
1000b	Signed bit and b18-b4 is clipped	16-bit	S, b18-b4
1001b	Signed bit and b17-b3 is clipped*2	16-bit	S, b17-b3
1010b	Signed bit and b16-b2 is clipped*2	16-bit	S, b16-b2
1011b	Signed bit and b15-b1 is clipped*2	16-bit	S, b15-b1
1100b	Signed bit and b14-b0 is clipped*2	16-bit	S, b14-b0

Note 1. S: Signed bit (= bit 19)

Note 2. When an overflow occurs, 7FFFh is input to the buffer.
When an underflow occurs, 8000h is input to the buffer.

8.7.3.4.2 Sinc Filter

The operating clock of the sinc filter is CCLK. Sinc filter is operated by using the 1-bit stream data latched with PDM_CLKn's rise-edge or fall-edge. Refer to **8.7.3.5 Data Input Control for Stereo Sound** for details.

Figure 8.7-3 shows a sinc filter block diagram.

The differentiation stage is operated by decimation clock (the frequency is 1/M of PDM_CLKn). M is the decimation ratio and is set in the PDSFCRCHn.SINCDEC[7:0] bits. The filter result is output in register every decimation clock.

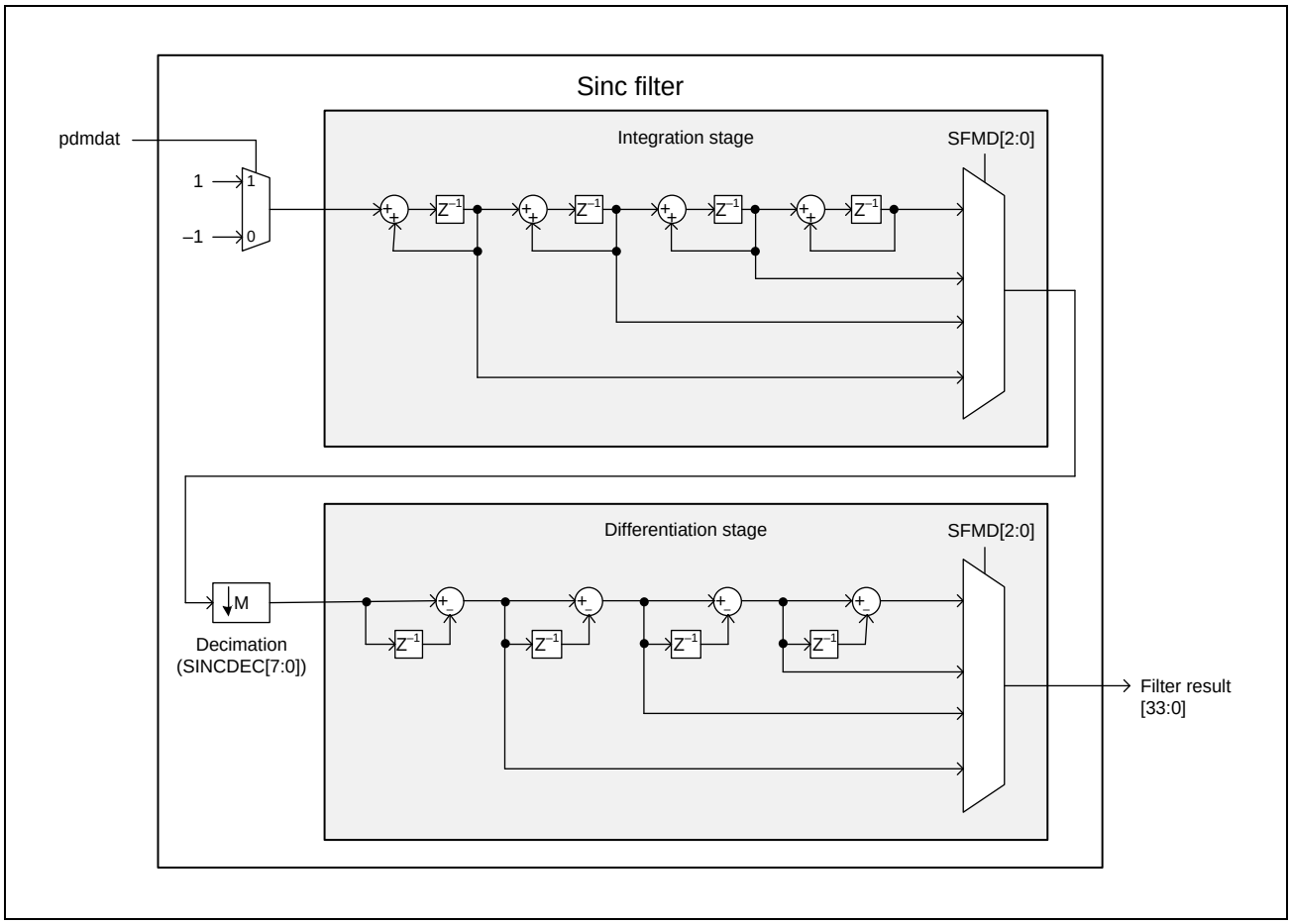


Figure 8.7-3 Sinc Filter Block Diagram

8.7.3.4.3 High-Pass Filter

The operating clock of the high-pass filter is CCLK. The high-pass filter is operated by using the result of the sinc filter.

Figure 8.7-4 shows a high-pass filter block diagram.

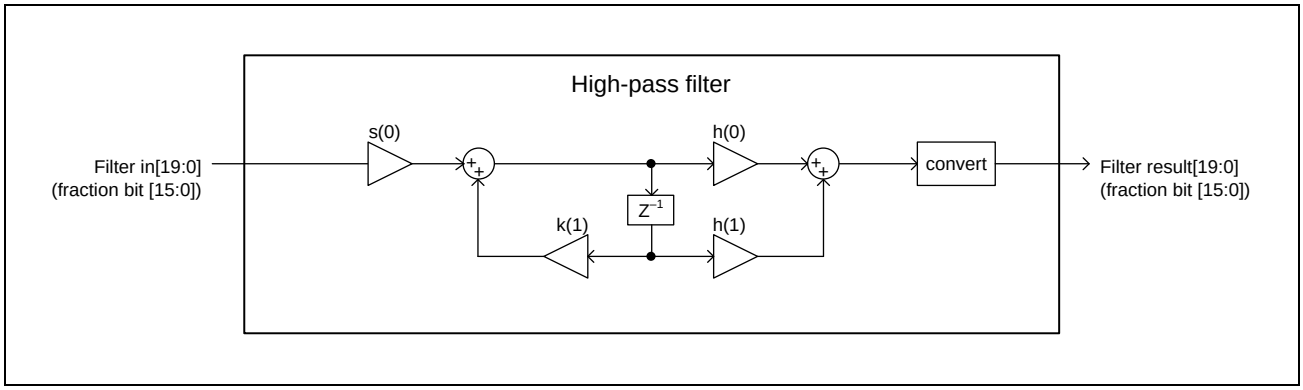


Figure 8.7-4 High-Pass Filter Block Diagram

Table 8.7-13 lists the high-pass filter specification.

Table 8.7-13 High-Pass Filter Specification

Function		Specification
Input	Bit width	Signed 20-bit (fraction 16-bit)
Output	Bit width	Signed 20-bit (fraction 16-bit)
Coefficient	Bit width	Signed 16-bit (fraction 14-bit) Refer to **8.7.2.2.20 High-Pass Filter Coefficient s(0) Register Channel n (PDMm_PDHFCS0RCHn) (m = 0 to 1, n = 0 to 2)**, **8.7.2.2.21 High-Pass Filter Coefficient k(1) Register Channel n (PDMm_PDHFCK1RCHn) (m = 0 to 1, n = 0 to 2)**, and **8.7.2.2.22 High-Pass Filter Coefficient h(p) Register Channel n (PDMm_PDHFCHpRCHn) (m = 0 to 1, n = 0 to 2, p = 0 to 1)** for details.
Delay Z	Bit width	Signed 32-bit (fraction 18-bit)
Multiplier	Input A bit width	Signed 32-bit (fraction 18-bit)
	Input B bit width	Signed 16-bit (fraction 14-bit)
	Output bit width	Signed 47-bit (fraction 32-bit)
Adder	Input A bit width	Signed 32-bit (fraction 18-bit)
	Input B bit width	Signed 32-bit (fraction 18-bit)
	Output bit width	Signed 33-bit (fraction 18-bit)
Output conversion	Input bit width	Signed 32-bit (fraction 18-bit)
	Output bit width	Signed 20-bit (fraction 16-bit)
Excessive fraction bits processing		Excessive fraction bits are rounded down at functional unit input or final output.

8.7.3.4.4 Compensation Filter

The operating clock of the compensation filter is CCLK. The compensation filter is operated by using the result of the high-pass filter.

Figure 8.7-5 shows a compensation filter block diagram.

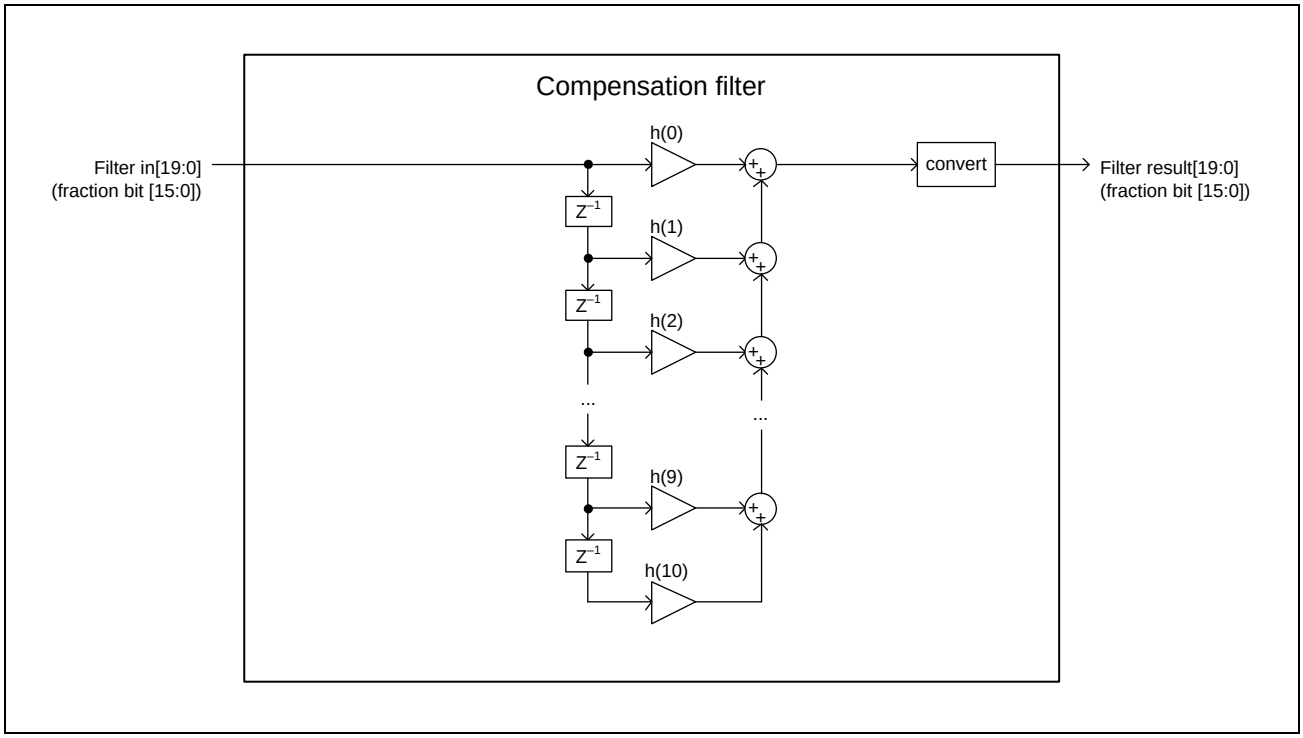


Figure 8.7-5 Compensation Filter Block Diagram

Table 8.7-14 shows a compensation filter specification.

Table 8.7-14 Compensation Filter Specification

Function		Specification
Input	Bit width	Signed 20-bit (fraction 16-bit)
Output	Bit width	Signed 20-bit (fraction 16-bit)
Coefficient	Bit width	Signed 13-bit (fraction 11-bit) Refer to **8.7.2.2.23 Compensation Filter Coefficient h(p) Register Channel n (PDMm_PDCFCHpRCHn) (m = 0 to 1, n = 0 to 2, p = 00 to 10)** for details.
Delay Z	Bit width	Signed 20-bit (fraction 16-bit)
Multiplier	Input A bit width	Signed 20-bit (fraction 16-bit)
	Input B bit width	Signed 13-bit (fraction 11-bit)
	Output bit width	Signed 32-bit (fraction 27-bit)
Adder	Input A bit width	Signed 22-bit (fraction 18-bit)
	Input B bit width	Signed 22-bit (fraction 18-bit)
	Output bit width	Signed 23-bit (fraction 18-bit)
Output conversion	Input bit width	Signed 22-bit (fraction 18-bit)
	Output bit width	Signed 20-bit (fraction 16-bit)
Excessive fraction bits processing		Excessive fraction bits are rounded down at functional unit input or final output.

8.7.3.4.5 Low-Pass (Half-Band Decimation) Filter

The operating clock of the low-pass filter is CCLK. The low-pass filter is operated by using the result of the compensation filter. Output data is decimated to a half of input data.

Figure 8.7-6 shows a low-pass filter block diagram.

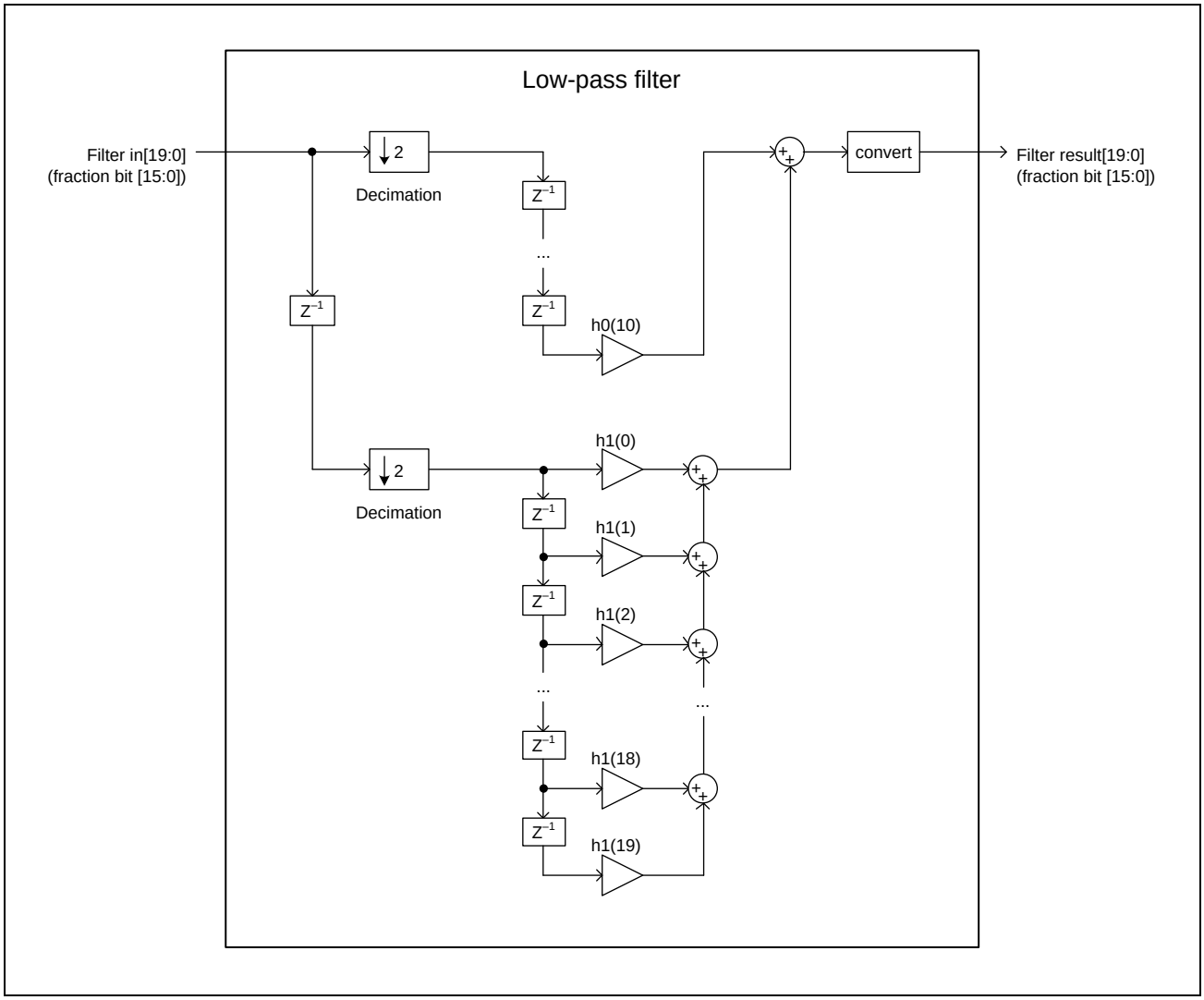


Figure 8.7-6 Low-Pass Filter Block Diagram

Table 8.7-15 lists a low-pass filter specification.

Table 8.7-15 Low-Pass Filter Specification

Function		Specification
Input	Bit width	Signed 20-bit (fraction 16-bit)
Output	Bit width	Signed 20-bit (fraction 16-bit)
Coefficient	Bit width	Signed 13-bit (fraction 11-bit) Refer to **8.7.2.2.24 Low-Pass Filter Coefficient h0(10) Register Channel n (PDMm_PDLFCH010RCHn) (m = 0 to 1, n = 0 to 2)** and **8.7.2.2.25 Low-Pass Filter Coefficient h1(p) Register Channel n (PDMm_PDLFCH1pRCHn) (m = 0 to 1, n = 0 to 2, p = 0 to 19)** for details.
Delay Z	Bit width	Signed 20-bit (fraction 16-bit)
Multiplier	Input A bit width	Signed 20-bit (fraction 16-bit)
	Input B bit width	Signed 13-bit (fraction 11-bit)
	Output bit width	Signed 32-bit (fraction 27-bit)
Adder	Input A bit width	Signed 22-bit (fraction 18-bit)
	Input B bit width	Signed 22-bit (fraction 18-bit)
	Output bit width	Signed 23-bit (fraction 18-bit)
Output conversion	Input bit width	Signed 22-bit (fraction 18-bit)
	Output bit width	Signed 20-bit (fraction 16-bit)
Excessive fraction bits processing		Excessive fraction bits are rounded down at functional unit input or final output.

8.7.3.4.6 Moving Average Filter

The operating clock of the moving average filter is CCLK. The moving average filter is operated by using the result of the high-pass filter. The order of the moving average filter can be changed by the setting of the PDMDSRCHn.SDMAMD[1:0] bits.

Figure 8.7-7 shows a moving average filter block diagram.

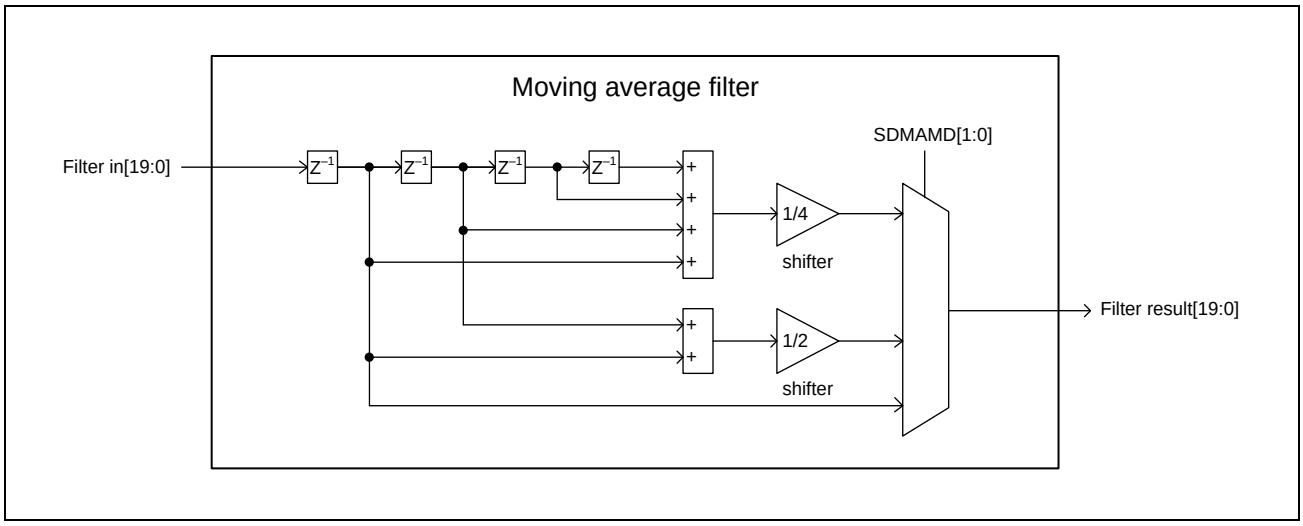


Figure 8.7-7 Moving Average Filter Block Diagram

Table 8.7-16 lists a moving average filter specification.

Table 8.7-16 Moving Average Filter Specification

Function		Specification
Input	Bit width	Signed 20-bit (fraction 16-bit)
Output	Bit width	Signed 20-bit (fraction 16-bit)
Delay Z	Bit width	Signed 20-bit (fraction 16-bit)
Excessive fraction bits processing		Excessive fraction bits are rounded down at shifter.

8.7.3.4.7 Settling Time of Channel Activation / Setting Change

The settling time of channel activation is the time after a start trigger PDCSTRTR.STRTRGn (or PDSTRTRCHn.STRTRG) is set before the filter result is stable and output. The settling time of channel setting change is the time after a change trigger PDCCHGTR.CHGTRGn (or PDCHGTRCHn.CHGTRG) is set before the filter result is stable and output. **Figure 8.7-8** shows the settling time measurement point.

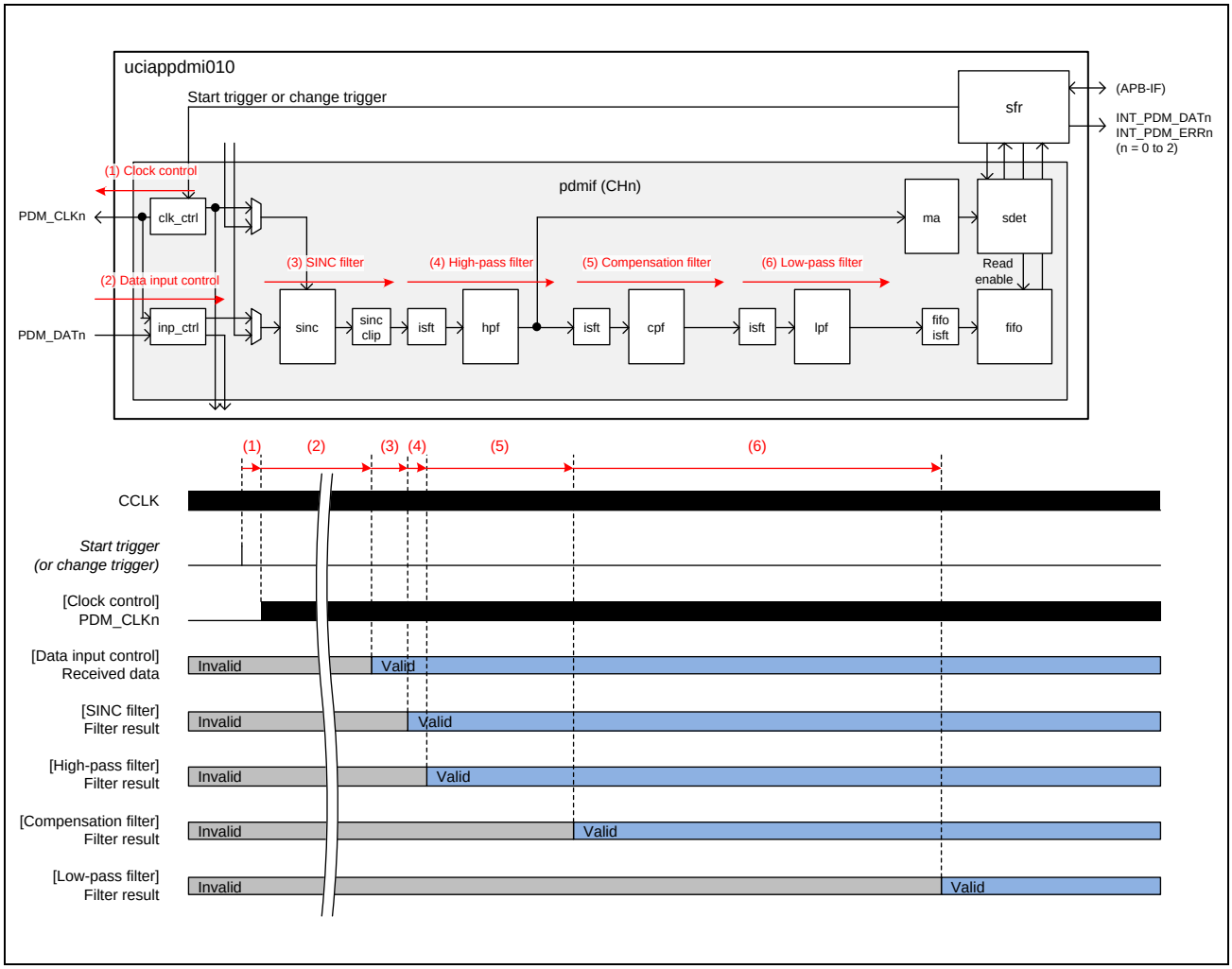


Figure 8.7-8 Settling Time of Channel Activation

The settling time is calculated as shown in **Table 8.7-17**.

Table 8.7-17 Settling Time of Channel Activation / Setting Change

No.	Function	Settling Time	
		Channel Activation	Channel Setting Change
(1)	Clock control	(4 + (D/2)) × Cp	(3 + Do + (D/2)) × Cp
(2)	Data input control	(D/2) × Cp + (Stable time for microphone*1)	
(3)	Sinc filter	(K × D × M) × Cp	
(4)	High-pass filter	(D × M + 5) × Cp	
(5)	Compensation filter	(10 × D × M + 12) × Cp	
(6)	Low-pass filter	(38 × D × M + 22) × Cp	

Note: Cp: CCLK period
D: PDM_CLKn's division ratio
Do: PDM_CLKn's division ratio (old setting)
K: Sinc filter order
M: Sinc filter decimation ratio

Note 1. Wakeup time from low power mode, Switching time, etc. Refer to the datasheet of microphone for details.

8.7.3.5 Data Input Control for Stereo Sound

The microphone which supports the stereo sound outputs the right sound synchronized with PDM_CLKn's rise-edge and the left sound synchronized with PDM_CLKn's fall-edge.

PDM-IF can process the right sound and the left sound using different channels.

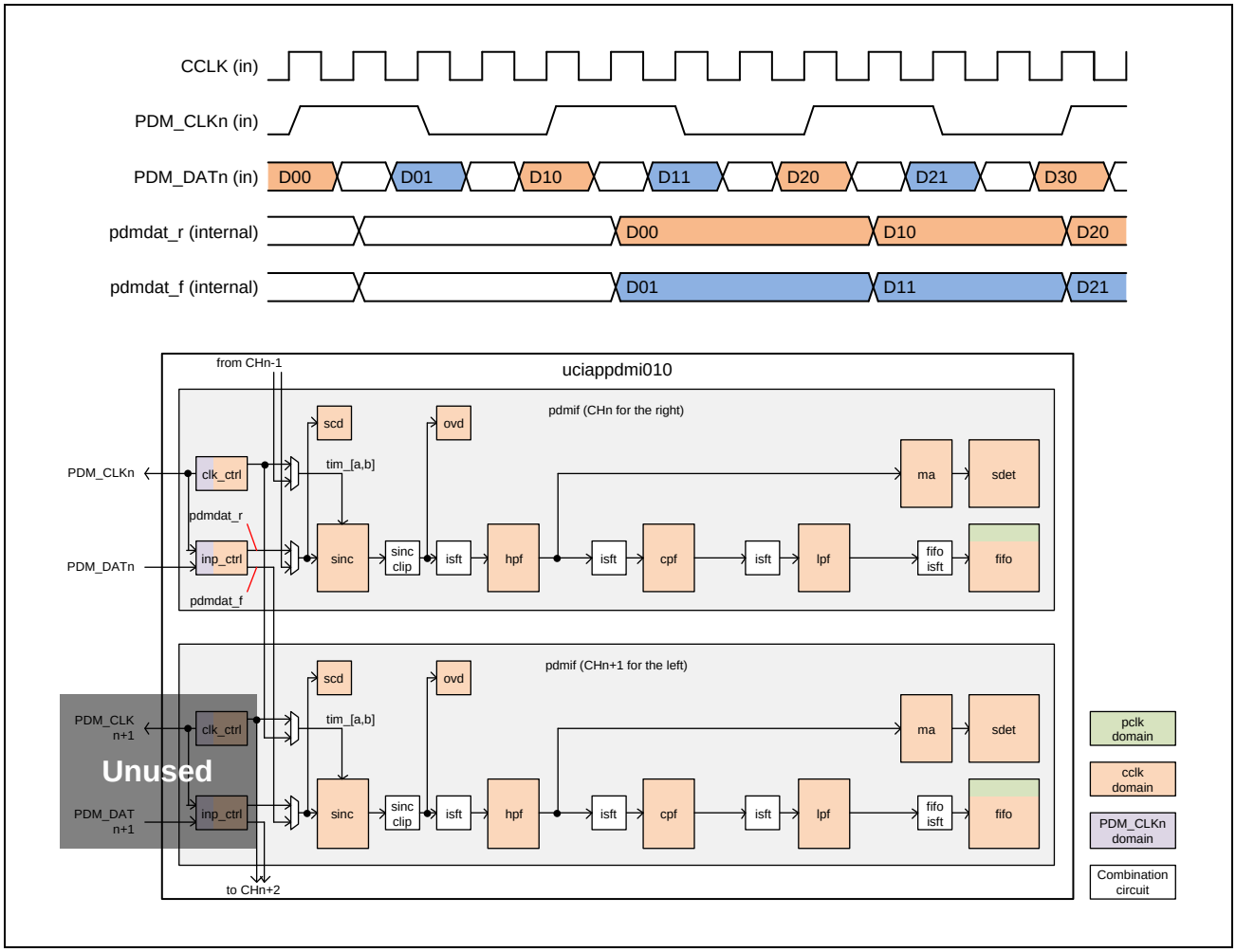


Figure 8.7-9 Data Input Control for Stereo Sound

Table 8.7-18 lists the channel combinations of stereo sound processing.

Table 8.7-18 Channel Combinations of Stereo Sound Processing

Input	Three Channels PDM-IF		Two Channels PDM-IF	
	Data Processing Latched at Rise-Edge (Right Sound)	Data Processing Latched at Fall-Edge (Left Sound)	Data Processing Latched at Rise-Edge (Right Sound)	Data Processing Latched at Fall-Edge (Left Sound)
PDM_CLK0 PDM_DAT0	Channel 0	Channel 1	Channel 0	Channel 1
PDM_CLK1 PDM_DAT1	Channel 1	Channel 2	Channel 1	Channel 0
PDM_CLK2 PDM_DAT2	Channel 2	Channel 0	Not implemented	

Table 8.7-19 lists an example of register settings for stereo sound processing using channel 0 and channel 1.

Table 8.7-19 Register Settings for Stereo Sound Processing

Register	Bit	Channel 0 Setting	Channel 1 Setting
PDICRCHn	All bits	Channel unique settings	Channel unique settings
PDSDCRCHn	All bits	Channel unique settings	Channel unique settings
PDMDSRCHn	INPSEL	0 (Use channel 0 data synchronized with rise-edge)	1 (Use channel 0 data synchronized with fall-edge)
	SFMD[2:0]	Same settings for channel 0 and channel 1 (Mandatory)	
	HFIS[1:0]	Channel unique settings	Channel unique settings
	HFMD	Same settings for channel 0 and channel 1 (Mandatory)	
	CFIS[1:0]	Channel unique settings	Channel unique settings
	CFMD	Same settings for channel 0 and channel 1 (Mandatory)	
	LFIS[1:0]	Channel unique settings	Channel unique settings
	LFMD	Same settings for channel 0 and channel 1 (Mandatory)	
	SDHFMD	Same settings for channel 0 and channel 1 (Mandatory)	
	SDMAMD[1:0]	Same settings for channel 0 and channel 1 (Mandatory)	
	DBIS[3:0]	Channel unique settings	Channel unique settings
PDSFCRCHn	CKDIV[3:0]	Setting of the clock which is supplied to the microphone connected to channel 0	Unuse
	SINCDEC[7:0]	Same settings for channel 0 and channel 1 (Mandatory)	
	SINCRNG[4:0]	Same settings for channel 0 and channel 1 (Mandatory)	
PDHFCS0RCHn	All bits	Channel unique settings	Channel unique settings
PDHFCK1RCHn	All bits	Channel unique settings	Channel unique settings
PDHFCH0RCHn	All bits	Channel unique settings	Channel unique settings
PDHFCH1RCHn	All bits	Channel unique settings	Channel unique settings
PDCFCHmRCHn	All bits	Channel unique settings	Channel unique settings
PDLFCH0mRCHn	All bits	Channel unique settings	Channel unique settings
PDLFCH1mRCHn	All bits	Channel unique settings	Channel unique settings
PDSDLTRCHn	All bits	Channel unique settings	Channel unique settings
PDSDUTRCHn	All bits	Channel unique settings	Channel unique settings
PDDBCRCHn	All bits	Same settings for channel 0 and channel 1 (Mandatory)	
PDSCTSRCHn	All bits	Channel unique settings	Channel unique settings
PDOVLTRCHn	All bits	Channel unique settings	Channel unique settings
PDOVUTRCHn	All bits	Channel unique settings	Channel unique settings

8.7.3.6 Sound Detection

The operating clock of the sound detection function is CCLK. When PDSDCRCHn.SDE = 1b, the function judges the sound detection using the moving average filter results.

A sound detection flag (PDSRCHn.SDF) is set to 1b when the result is above the upper limit PDSDUTRCHn.SDETU[19:0]. The sound detection flag is also set to 1b when the result is below the lower limit PDSDLTRCHn.SDETL[19:0]. The sound detection flag is cleared to 0b when 1b is written to PDSCRCHn.SDFC.

Figure 8.7-10 shows the sound detection function.

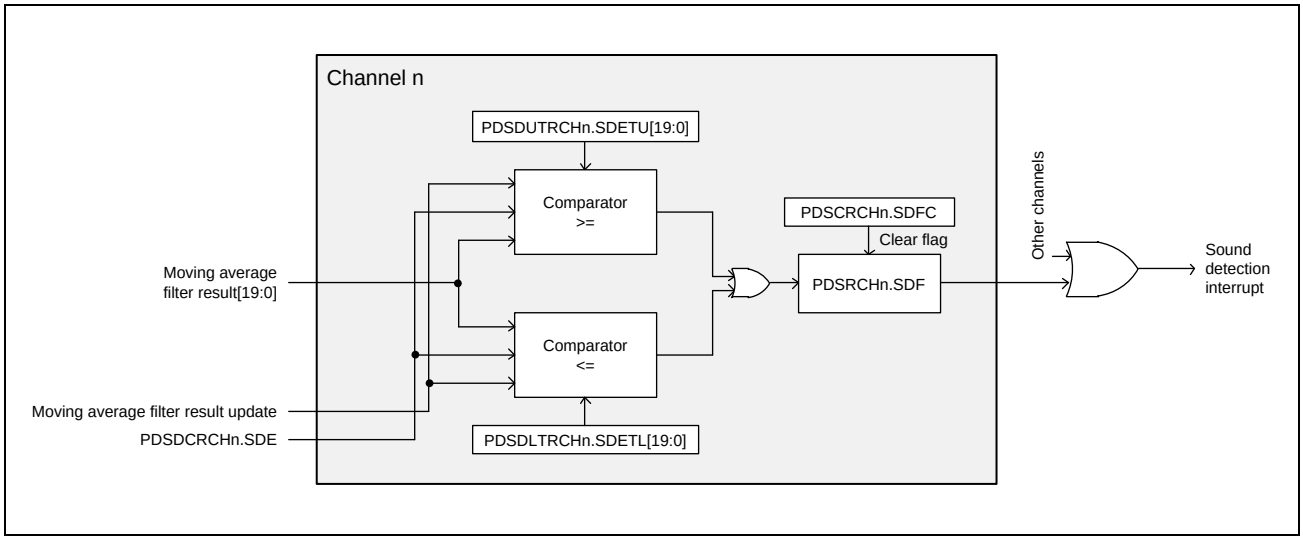


Figure 8.7-10 Sound Detection Function

8.7.3.7 Data Buffer

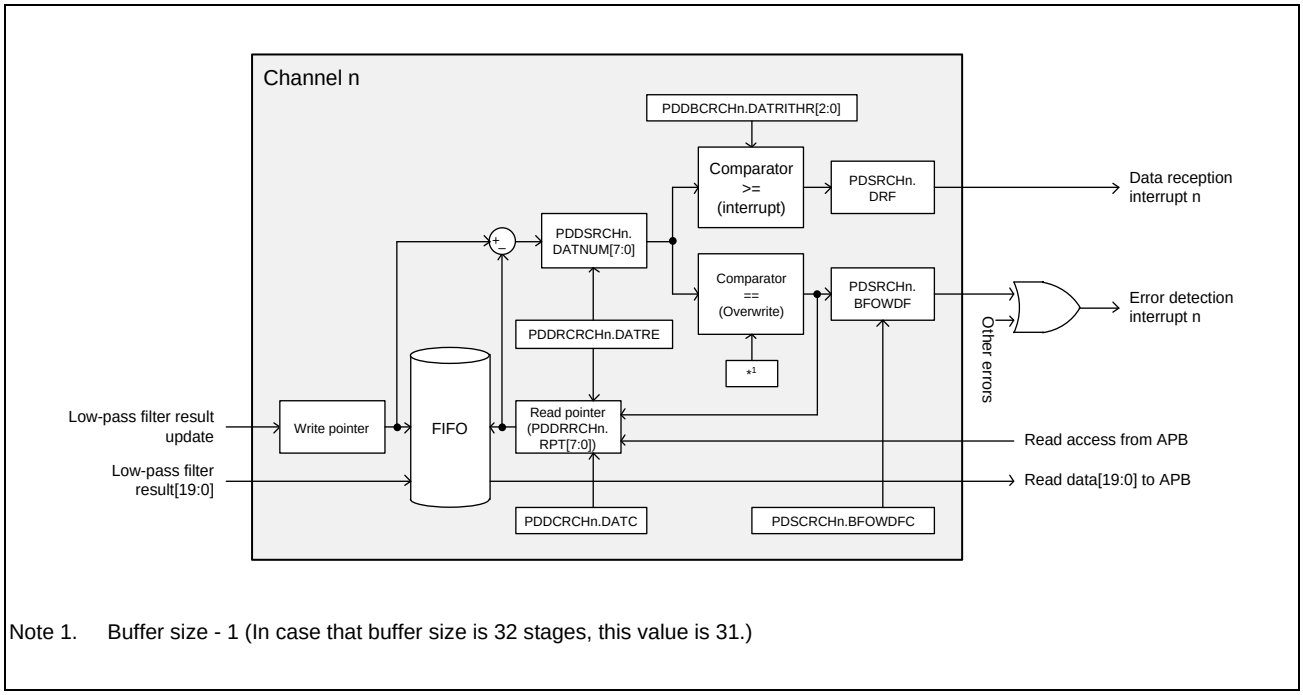
Data buffer writes sound data to FIFO synchronizing with CCLK and reads sound data from FIFO synchronizing with PCLK. Data buffer stores sound data in FIFO until it is read out.

While PDDRCRCHn.DATRE = 0b (hereinafter, DATRE), PDDSRCHnDATNUM[7:0] (hereinafter, DATNUM[7:0]) shows 0 and PDM-IF does not allow to read sound data. After DATRE is changed from 0b to 1b, DATNUM[7:0] shows the buffer size minus 2 and PDM-IF allows to read sound data. While DATRE = 1b, data buffer increments DATNUM[7:0] when data is written and decrements DATNUM[7:0] when data is read.

When DATNUM[7:0] is greater than or equal to a threshold which is configured by PDDBCRCHn.DATRITHR[2:0], a data reception flag PDSRCHn.DRF (hereinafter, DRF) is set to 1b. When DATNUM[7:0] is less than the threshold by reading data or etc., a data reception flag DRF is cleared to 0b.

When DATNUM[7:0] is the buffer size minus 2 and data is written without reading, data buffer overwrites the oldest data with a new one and a buffer overwriting detection flag PDSRCHn.BFOWDF (hereinafter, BFOWDF) is set to 1b. And then DATNUM[7:0] is set to the buffer size minus 2 again after overwriting. However, data buffer overwrites but BFOWDF is not set after DATRE changes from 0b to 1b to before the first data is read.

Figure 8.7-11 shows the data buffer function.



Note 1. Buffer size - 1 (In case that buffer size is 32 stages, this value is 31.)

Figure 8.7-11 Data Buffer Function

8.7.3.8 Short-Circuit Detection

When Short-circuit detection enable (PDSDCRCHn.SCDE) is valid, the short-circuit detection can operate.

A dedicated 13-bit counter counts consecutive 0’s or 1’s input from pdm_dat which is outputs by data input control block of channel n or channel n+1.

When the number of consecutive 0’s exceeds the value set in the PDSCTSRCHn.SCDL[12:0] register or the number of consecutive 1’s exceeds the value set in the PDSCTSRCHn.SCDH[12:0] register, a short-circuit detection interrupt request is generated.

Figure 8.7-12 shows the configuration for the short-circuit detection function.

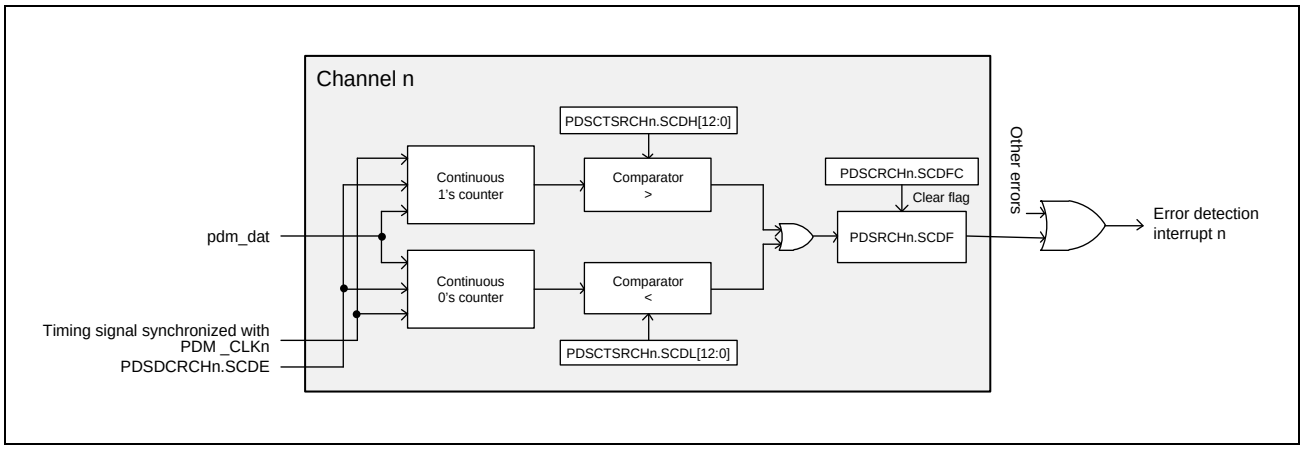


Figure 8.7-12 Short-Circuit Detection Function

8.7.3.9 Overvoltage Detection

When the overvoltage upper limit detection enable (PDSDCRCHn.OVUDE) is valid, the overvoltage upper limit detection can operate. When the overvoltage lower limit detection enable (PDSDCRCHn.OVLDE) is valid, the overvoltage lower limit detection can operate.

The sound data is clipped as a 20-bit data from 34-bit data output from sinc filter by the setting of the PDSFCRCHn.SINCRNG[4:0] bits. Refer to **Table 8.7-8** for details.

An overvoltage upper limit detection flag is set to 1b when the clipped sound data is greater than PDOVUTRCHn.OVDU[19:0].

An overvoltage lower limit detection flag is set to 1b when the clipped sound data is less than PDOVLTRCHn.OVDL[19:0].

Figure 8.7-13 shows the overvoltage detection function.

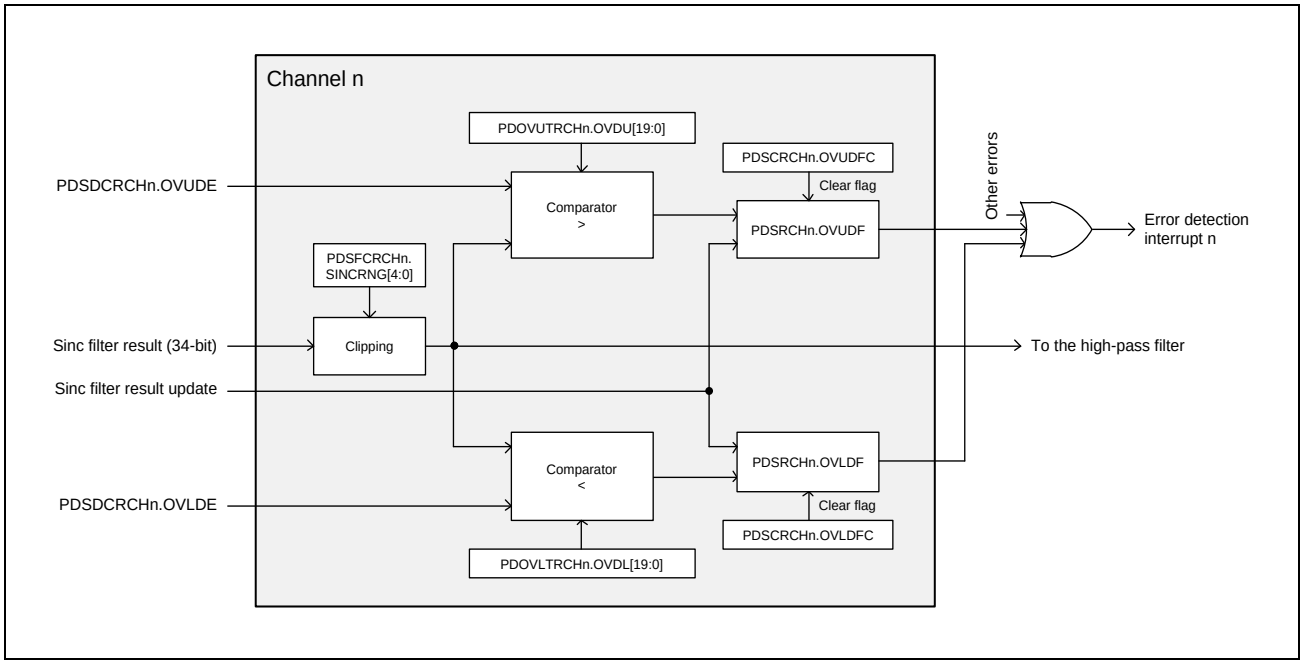


Figure 8.7-13 Overvoltage Detection Function

8.7.3.10 Error

Table 8.7-20 shows a PDM-IF error list.

Table 8.7-20 PDM-IF Error List

No.	Register Name	Description	Comment	Function (Related Section)
1	PDSRCHn.SCDF	Short circuit detection flag		**8.7.3.8 Short-Circuit Detection**
2	PDSRCHn.OVLDF	Overvoltage lower limit detection flag		**8.7.3.9 Overvoltage Detection**
3	PDSRCHn.OVUDF	Overvoltage upper limit detection flag		**8.7.3.9 Overvoltage Detection**
4	PDSRCHn.BFOWDF	Buffer overwriting detection flag	For debugging	**8.7.3.7 Data Buffer**

The errors in No.4 are for debugging. Software needs to set registers and operate PDM-IF so that these errors do not occur. If any of these error occur, stop PDM-IF following **8.7.4.2 Stop Flow** and change the settings.

8.7.3.11 Interrupt Function

PDM-IF has the following interrupt function.

Table 8.7-21 PDM-IF Interrupt List

No.	Signal Name	Interrupt Name	Number of Signals	Pulse or Level	Related Register Bit Name		Function (Related Section)
					Common Register	Channel Register	
1	INT_PDM_DATn (n = 0, 1, 2)	Data reception interrupt CHn	3	Level	PDCSR.DRFn	PDSRCHn.DRF	**8.7.3.7 Data Buffer**
2	INT_PDM_SDET	Sound detection interrupt (3-channel status flags are ORed.)	1	Level	PDCSR.SDFn	PDSRCHn.SDF	**8.7.3.6 Sound Detection**
3	INT_PDM_ERRn (n = 0, 1, 2)	Error detection interrupt CHn	3	Level	PDCSR.EDFn	PDSRCHn.SCDF	**8.7.3.8 Short-Circuit Detection**
						PDSRCHn.OVLDF	**8.7.3.9 Overvoltage Detection**
						PDSRCHn.OVUDF	**8.7.3.9 Overvoltage Detection**
						PDSRCHn.BFOWDF	**8.7.3.7 Data Buffer**

8.7.4 Setting Flow

Figure 8.7-14 and **Figure 8.7-15** show the example of the main procedure of PDM-IF.

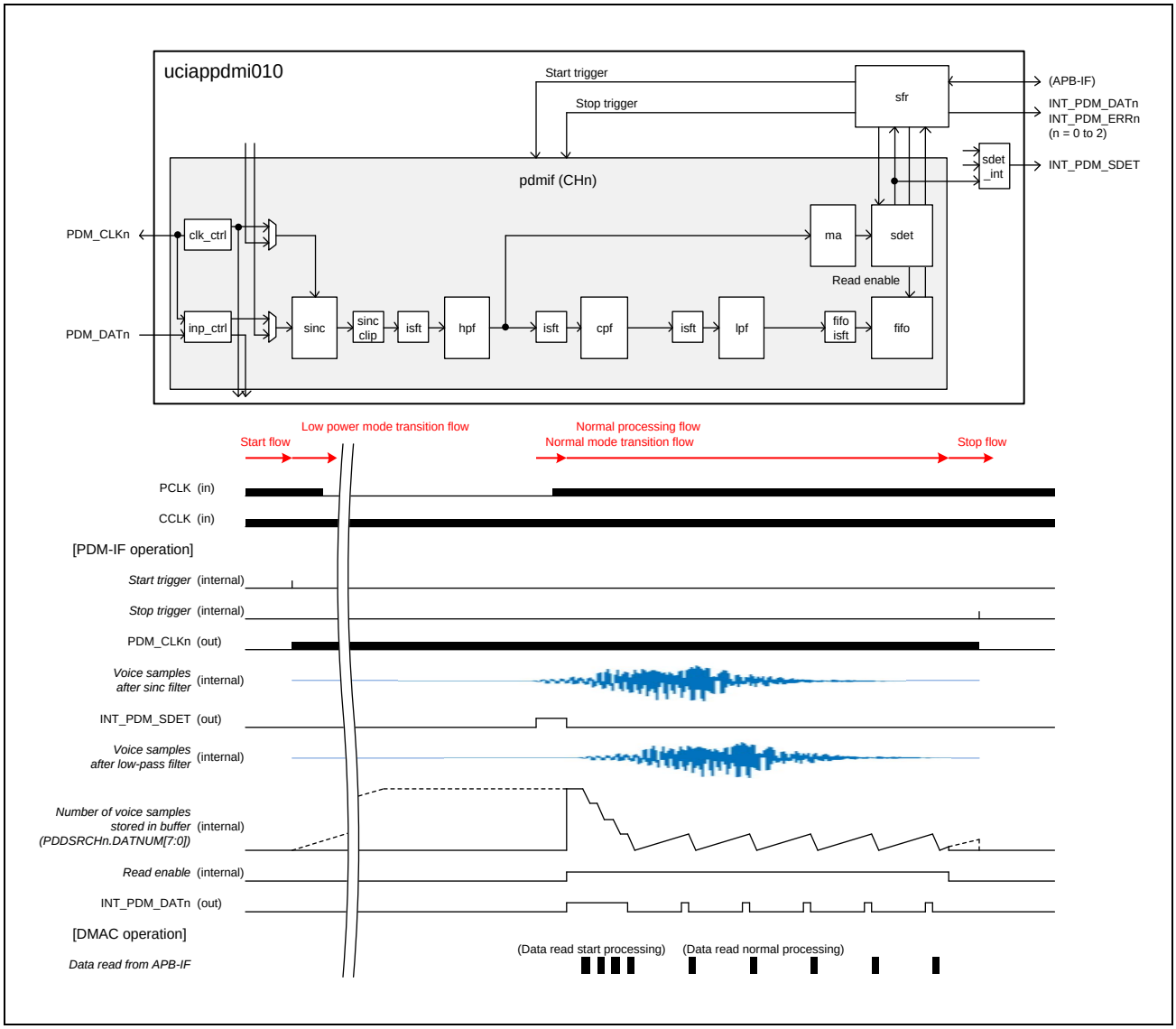


Figure 8.7-14 PDM-IF Operation Example

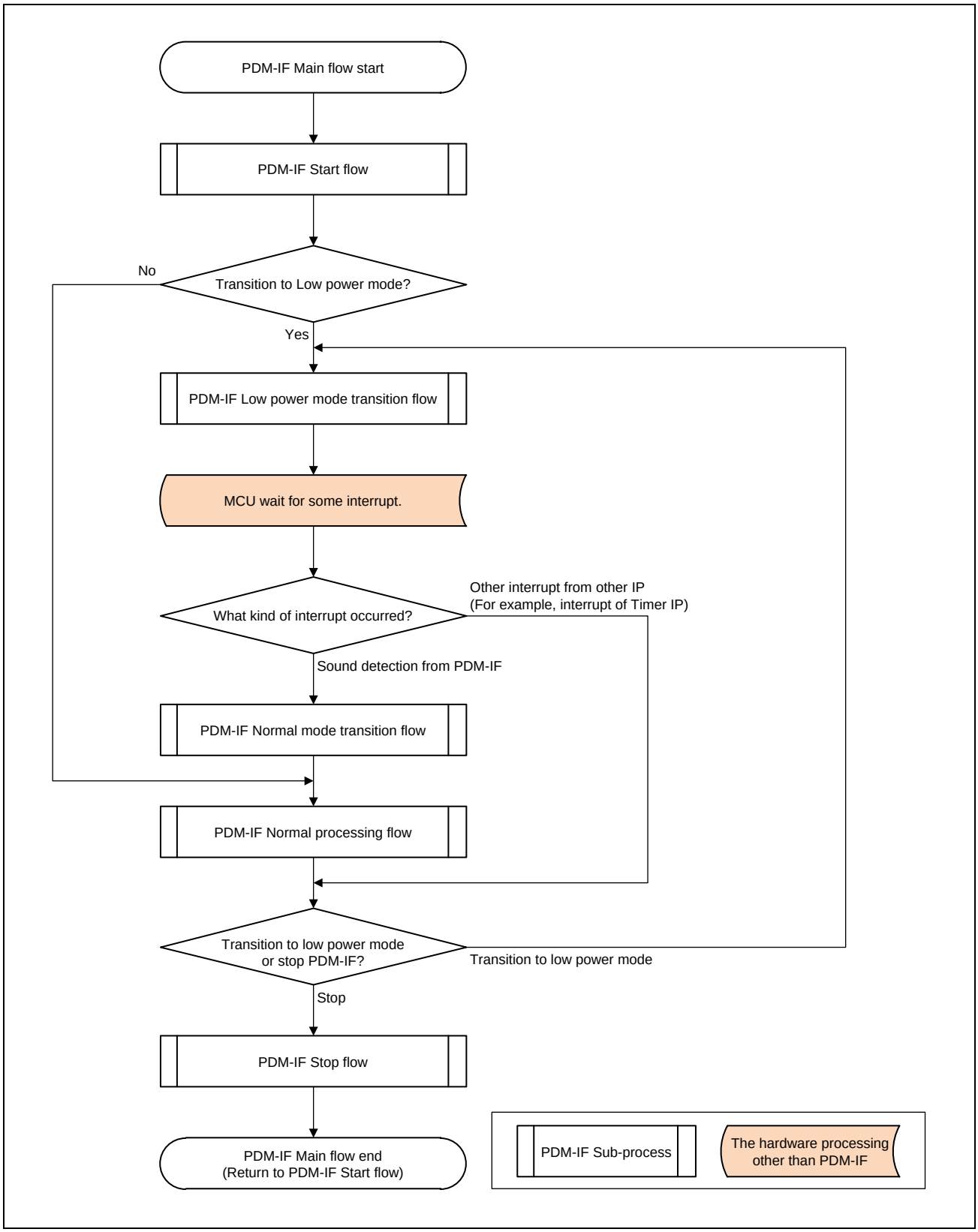


Figure 8.7-15 PDM-IF Setting Main Flow with Low Power Mode Transition

Figure 8.7-16 shows the example of the main procedure without low-power mode transition.

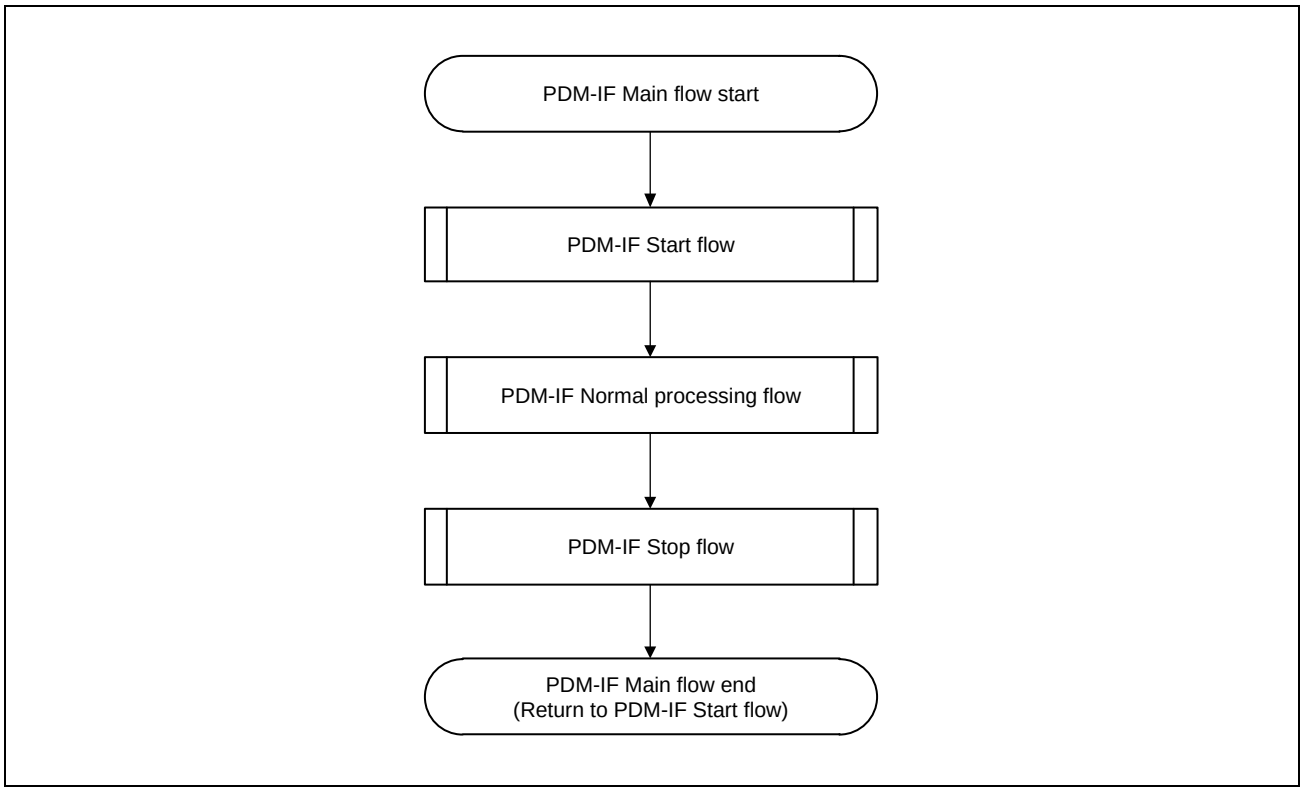
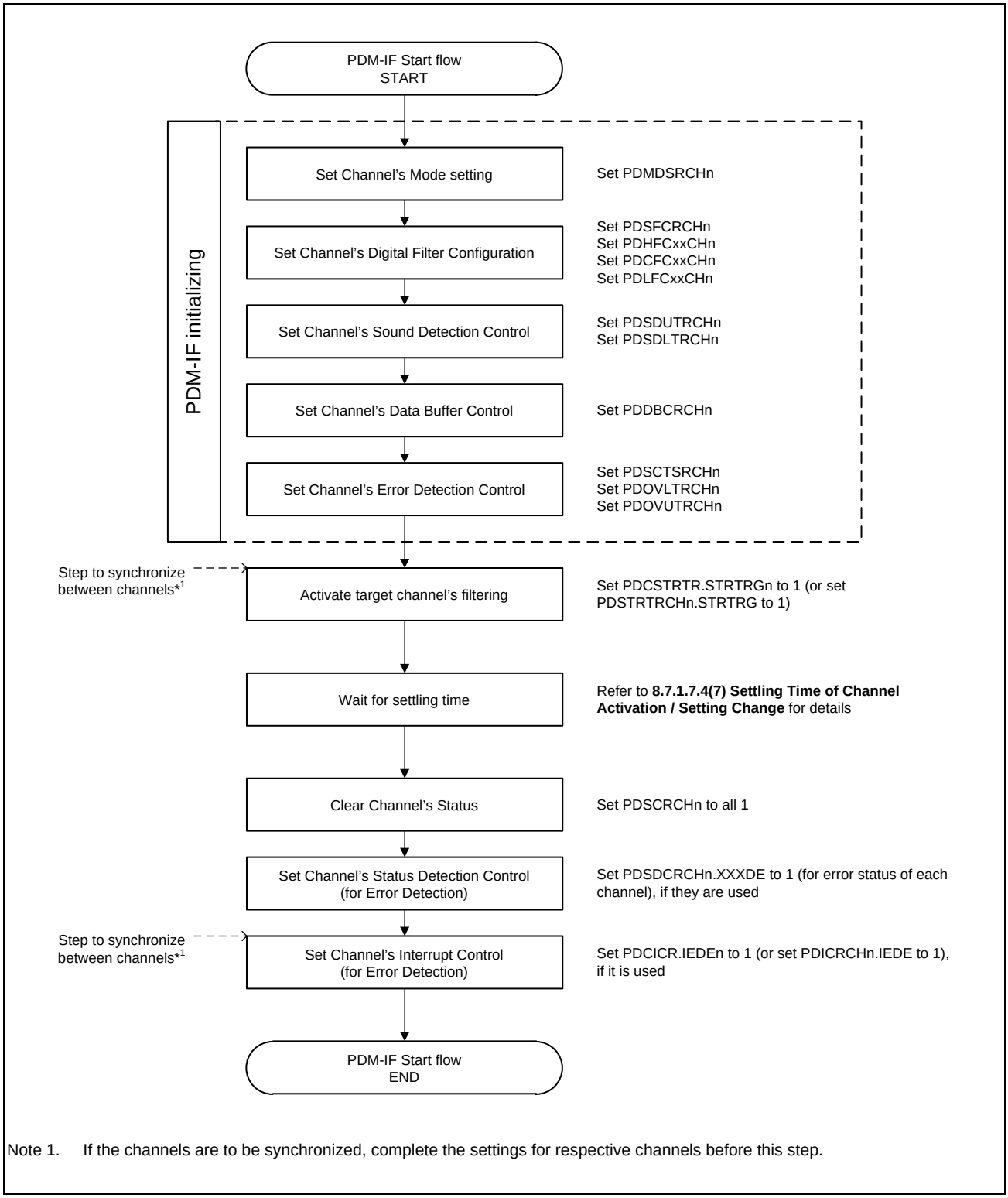


Figure 8.7-16 PDM-IF Setting Main Flow without Low Power Mode Transition

8.7.4.1 Start Flow

Figure 8.7-17 shows the example of the start procedure of PDM-IF.



Note 1. If the channels are to be synchronized, complete the settings for respective channels before this step.

Figure 8.7-17 PDM-IF Start Flow

8.7.4.2 Stop Flow

Figure 8.7-18 shows the example of the stop procedure of PDM-IF.

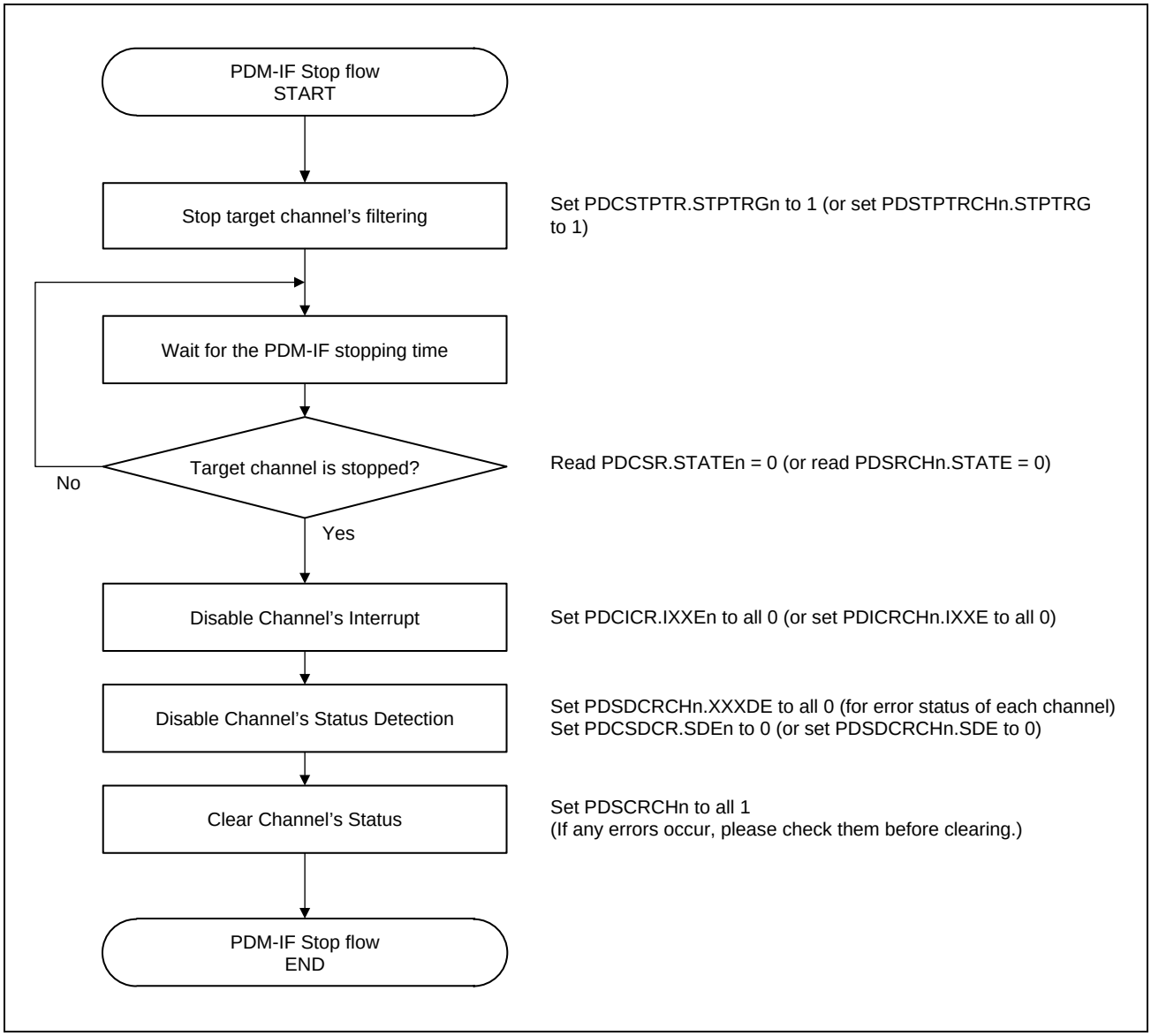


Figure 8.7-18 PDM-IF Stop Flow

8.7.4.3 Normal Processing Flow

Figure 8.7-19 shows the example of the normal processing procedure of PDM-IF.

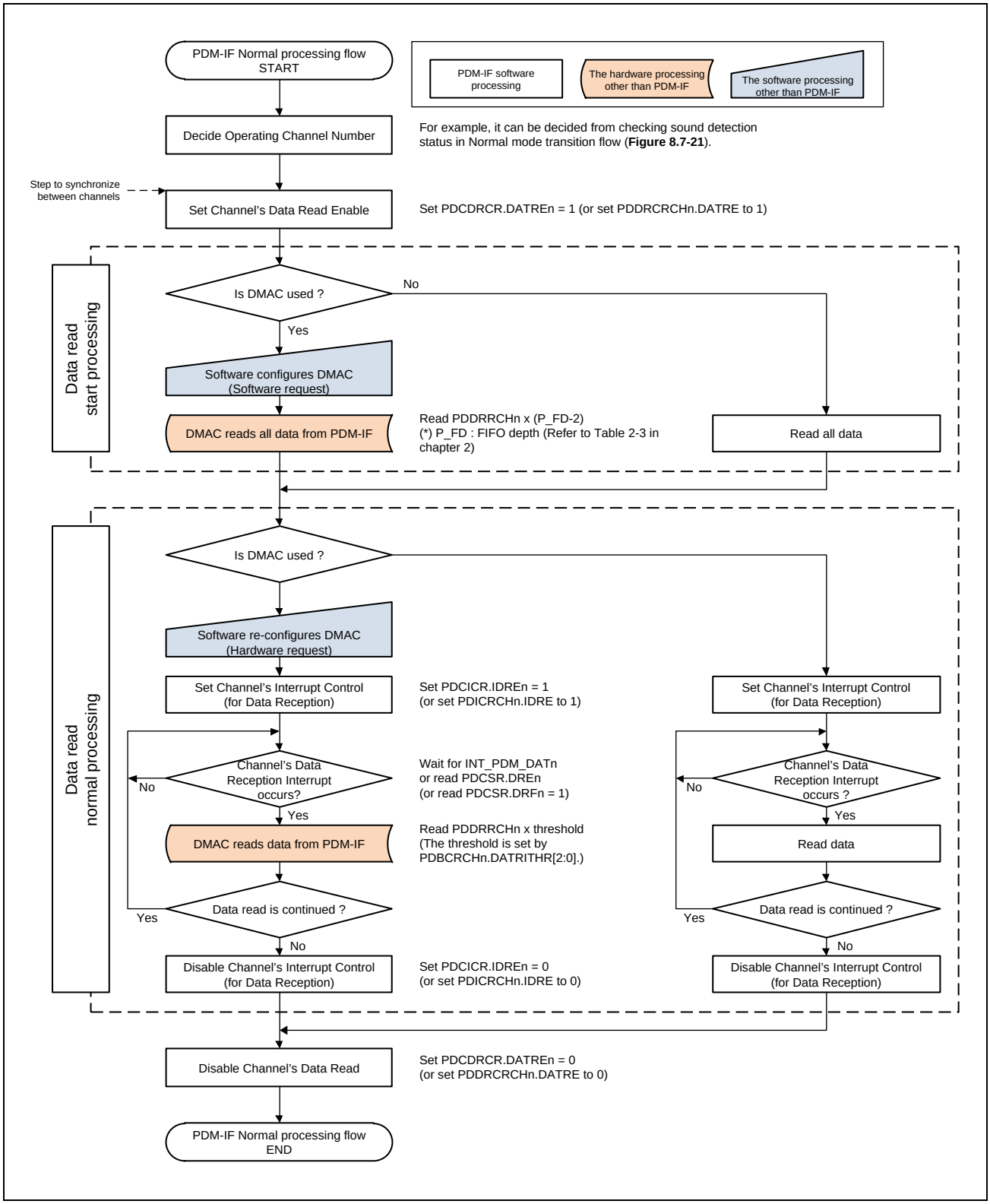
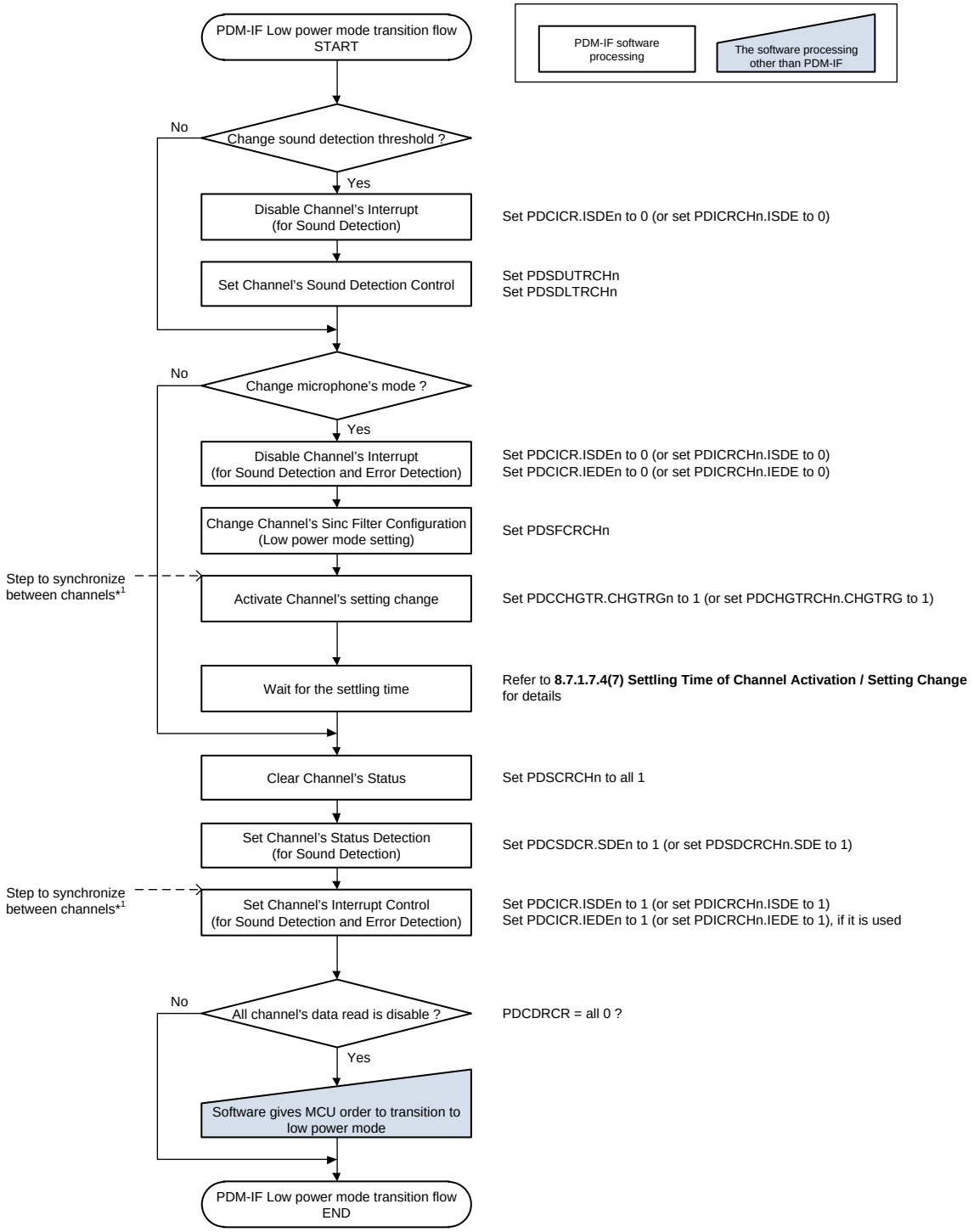


Figure 8.7-19 PDM-IF Normal Processing Flow

8.7.4.4 Low Power Mode Transition Flow

Figure 8.7-20 shows the example of the low power mode transition procedure of PDM-IF.

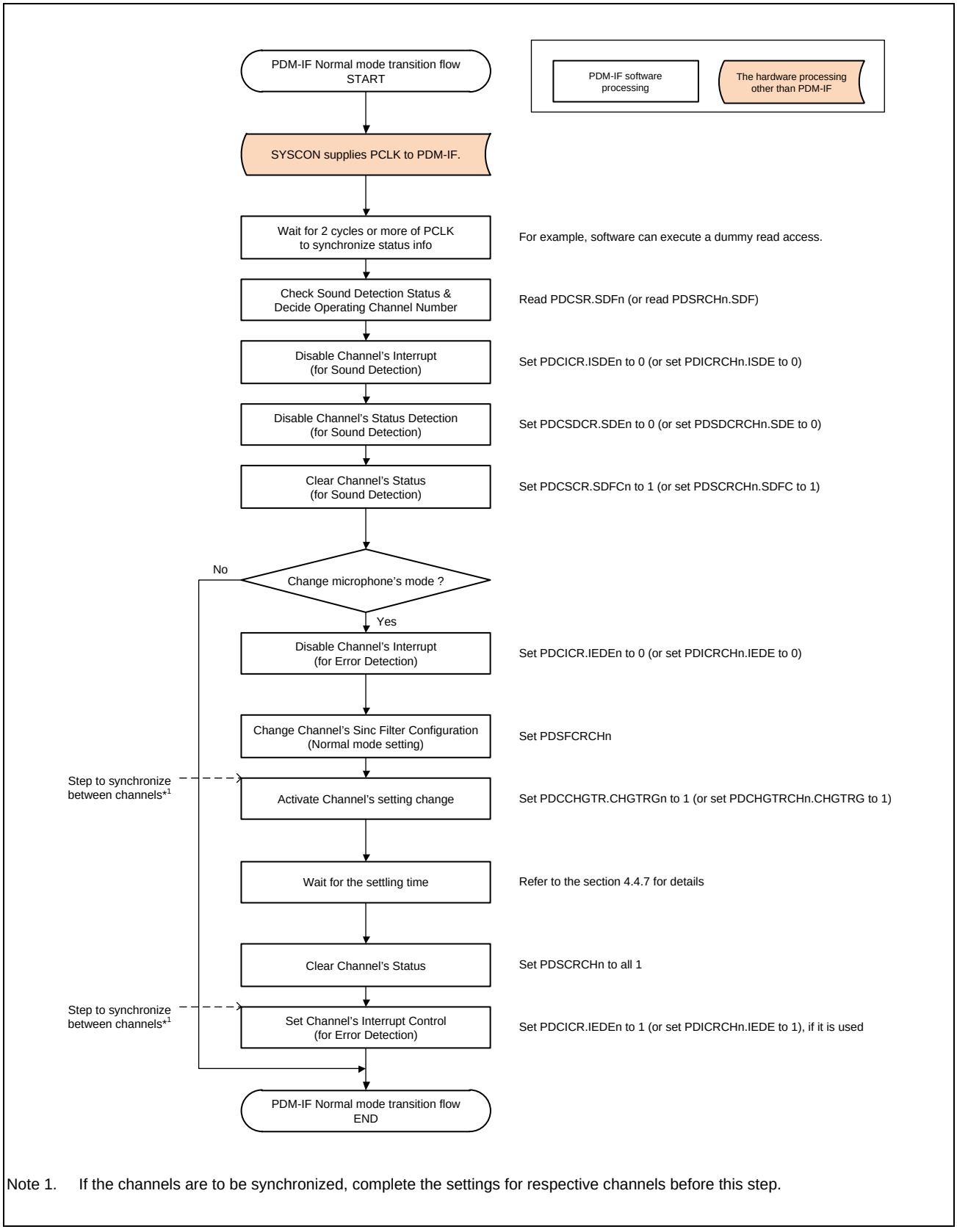


Note 1. If the channels are to be synchronized, complete the settings for respective channels before this step.

Figure 8.7-20 PDM-IF Low Power Mode Transition Flow

8.7.4.5 Normal Mode Transition Flow

Figure 8.7-21 shows the example of normal mode transition procedure of PDM-IF.



Note 1. If the channels are to be synchronized, complete the settings for respective channels before this step.

Figure 8.7-21 PDM-IF Normal Mode Transition Flow

8.7.4.6 Setting PDM_CLKn (n = 0, 1, 2) Frequency and Sampling Frequency Generated by the Unit

The frequency of PDM_CLKn (n = 0, 1, 2), which is the reference for receiving data from the external device, and the sampling frequency generated by the PDM unit for external input PDM data can be changed by setting the register. The PDM_CLKn (n = 0, 1, 2) frequency must match the clock input specifications of the external device. The sampling frequency generated by the unit should be set to the expected value.

In this LSI, the PDM reference clock CCLK is fixed at 4.8 MHz, so the PDM_CLK output frequency and the unit-generation sampling frequency follow the equation below when implementing the circuit.

- PDM_CLKn (n = 0, 1, 2) frequency

$$= \frac{\text{CCLK frequency}}{(\text{CKDIV} + 1) \times 2};$$

- Unit-generation sampling frequency

$$= \frac{\text{PDM_CLK frequency}}{(\text{SINCDEC} + 1) \times 2^{*1}} = \frac{\text{CCLK frequency}}{(\text{CKDIV} + 1) \times 2} \div (\text{SINCDEC} + 1) \times 2^{*1};$$

Remark: CKDIV: PDSFCRCHn.CKDIV[3:0] register set value (Set the PDM_CLKn division ratio to CCLK)
SINCDEC: PDSFCRCHn.SINCDEC[7:0] register set value (Set the decimation ratio of the Sinc filter)

Note 1. Reason for ×2: Low-pass filter decimation ratio

- PDMIF sampling rate:

$$\frac{\text{CCLK frequency}}{\big((\text{CKDIV} + 1) \times 2\big) \times (\text{SINCDEC} + 1) \times 2}$$

The output range of the sinc filter is changed by changing the decimation ratio of the sinc filter (SINCDEC) with the change of the sampling frequency above. Since the maximum number of effective bits for processing of audio data within the PDM following sinc filtering is 20 bits, PDSFCRCHn.SINCRNG[4:0] must also be changed according to the output range of the sinc filter. This allows avoiding the occurrence of overflow and underflow and maximizing the effective audio amplitude.

Recommended setting: Set PDSFCRCHn.SINCRNG[4:0] such that the higher-order 20 bits of valid data of the sinc filter output are effective in calculation processing following sinc filtering.

For details on clipping or sifting of the sinc filter output, see **8.7.3.4.1 Data Clipping and Shifting between Filters**.

- Output range of the sinc filter
  = ±decimation ratio (PDSFCRCHn.SINCDEC[7:0]) ^ order (PDMDSRCHn.SFMD[2:0]);

[Example]

When SINCDEC[7:0] = 27h (decimation ratio: 40) and SFMD[2:0] = 0 (4-order), the output range of the sinc filter is ±2560000 (23 bits consisting of 22 bits of significant digits and a sign bit). The recommended setting of PDSFCRCHn.SINCRNG[4:0] in this case is 0_1011b (b22-b3).

The following describes the impact of the circuit structure on the frequency calculations.

The table below shows the relationship between the selectable N/M values (register settings) with respect to the PDM_CLKn (n = 0, 1, 2) frequency and the unit-generation sampling frequency, according to the above calculation formula, and the target-generated sampling frequency.

In the table below, “Not selectable” indicates that the N value or M value cannot be selected for the target-generated sampling frequency. Since the N value can be selected from 2 to 32 in multiples of 2 and the M value can only be selected as an integer of 1, 2, 3, ..., 256, the corresponding target-generated sampling frequency may not be generated. However, in some cases, a frequency near the target-generation sampling frequency may be selectable. If that frequency is acceptable, refer to the PDM_CLKn (n = 0, 1, 2) frequency and unit-generation sampling frequency calculation formula above to derive the desired N value/M value and the corresponding register setting value.

Table 8.7-22 Selectable PDM_CLKn (n = 0, 1, 2) Frequency and Unit-Generation Sampling Frequency

	Sampling frequency (kHZ)	48	40	30	25	24	20	16	15	12	10	8
N value	PDM_CLK frequency (MHz)	M value	M value	M value	M value	M value	M value	M value	M value	M value	M value	M value
2	2.4	25	30	40	48	50	60	75	80	100	120	150
4	1.2	Not selectable	15	20	24	25	30	Not selectable	40	50	60	75
6	0.8	Not selectable	10	Not selectable	16	Not selectable	20	25	Not selectable	Not selectable	40	50
8	0.6	Not selectable	Not selectable	10	12	Not selectable	15	Not selectable	20	25	30	Not selectable
10	0.48	5	6	8	Not selectable	10	12	15	16	20	24	30
12	0.4	Not selectable	5	Not selectable	8	Not selectable	10	Not selectable	Not selectable	Not selectable	20	25
14	0.342857143	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable
16	0.3	Not selectable	Not selectable	5	6	Not selectable	Not selectable	Not selectable	10	Not selectable	15	Not selectable
18	0.266666667	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable
20	0.24	Not selectable	3	4	Not selectable	5	6	Not selectable	8	10	12	15
22	0.218181818	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable
24	0.2	Not selectable	Not selectable	Not selectable	4	Not selectable	5	Not selectable	Not selectable	Not selectable	10	Not selectable
26	0.184615385	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable
28	0.171428571	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable	Not selectable
30	0.16	Not selectable	2	Not selectable	Not selectable	Not selectable	4	5	Not selectable	Not selectable	8	10
32	0.15	Not selectable	Not selectable	Not selectable	3	Not selectable	Not selectable	Not selectable	5	Not selectable	Not selectable	Not selectable

The N value/M value in the above table is based on the following formula.

Definition of N value: (CKDIV + 1) × 2
Definition of M value: (SINCDEC + 1)

Remark: CKDIV: PDSFCRCHn.CKDIV[3:0] register set value
SINCDEC: PDSFCRCHn.SINCDEC[7:0] register set value

8.7.4.7 DMA Transfer Method

This PDM unit can only receive voice data from external sources, and does not have a transmit function. The PDM unit does not have a dedicated REQ/ACK signal to the DMAC master, but only reads data from the PDM buffer read register via the APB I/F. Reading data via the CPU and reading data via the DMAC master are not different in terms of PDM functionality.

Figure 8.7-22 shows an overview of the PDM data read process.

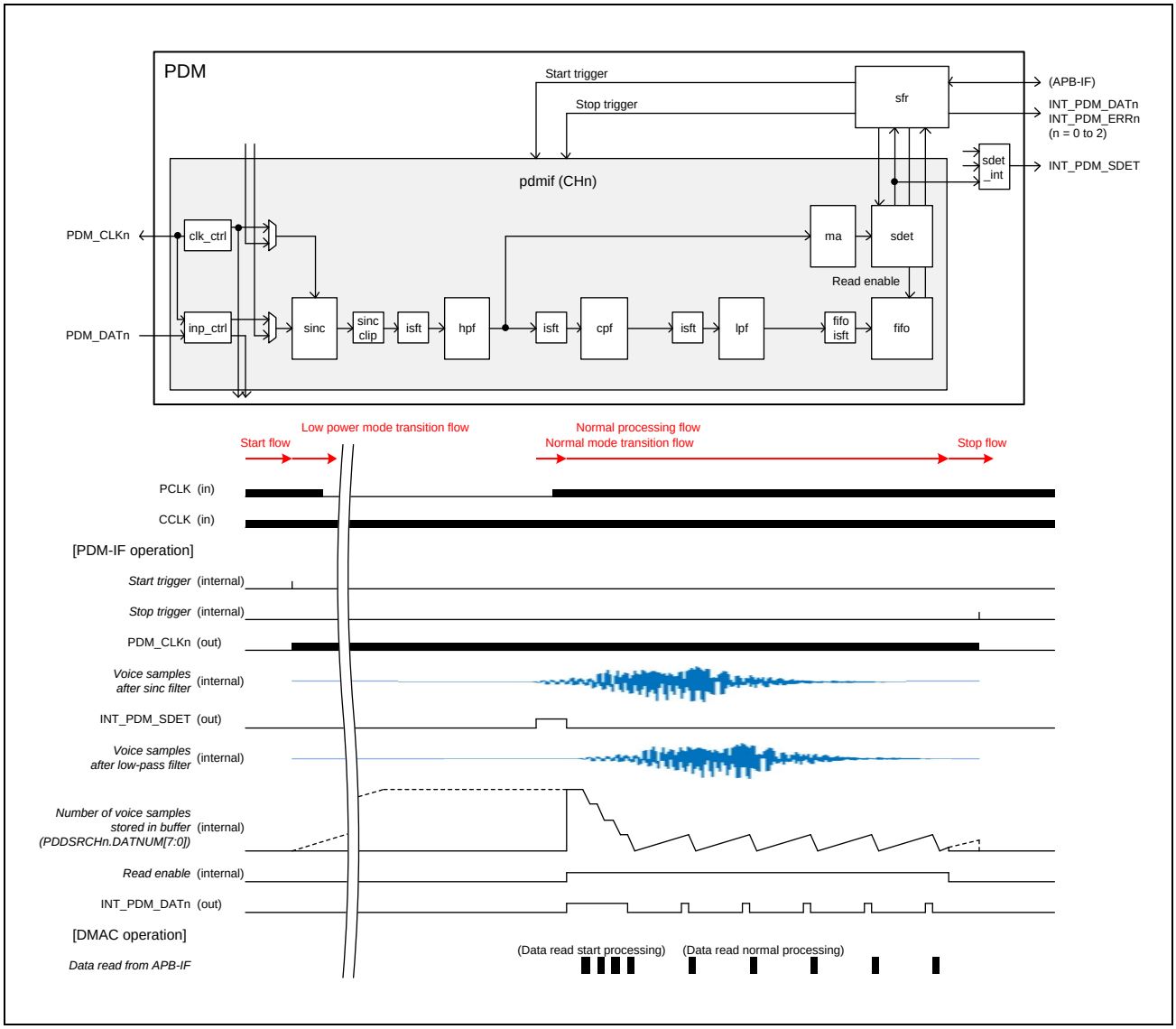


Figure 8.7-22 PDM-IF Operation Example

The DMA data transfer using the DMA master must follow the description in **8.7.4.3 Normal Processing Flow**. The following explains the connection configuration with the DMA master, initialization settings on the DMA master, and recommended settings for the INT_PDM_DAT0/1/2 signal at PDM initialization.

The connection configuration with the DMA master in this LSI is as follows, issuing a DMA request to the DMA master based on the INT_PDM_DAT0/1/2 signal of the PDM. The INT_PDM_DAT0/1/2 interrupt signal is active (high) depending on the amount of data stored in the receive buffer in the PDM (the amount of interrupt generation buffer can be changed by setting the register / PDDBCRCHn (n = 0, 1, 2).DATRITHR[2:0]).

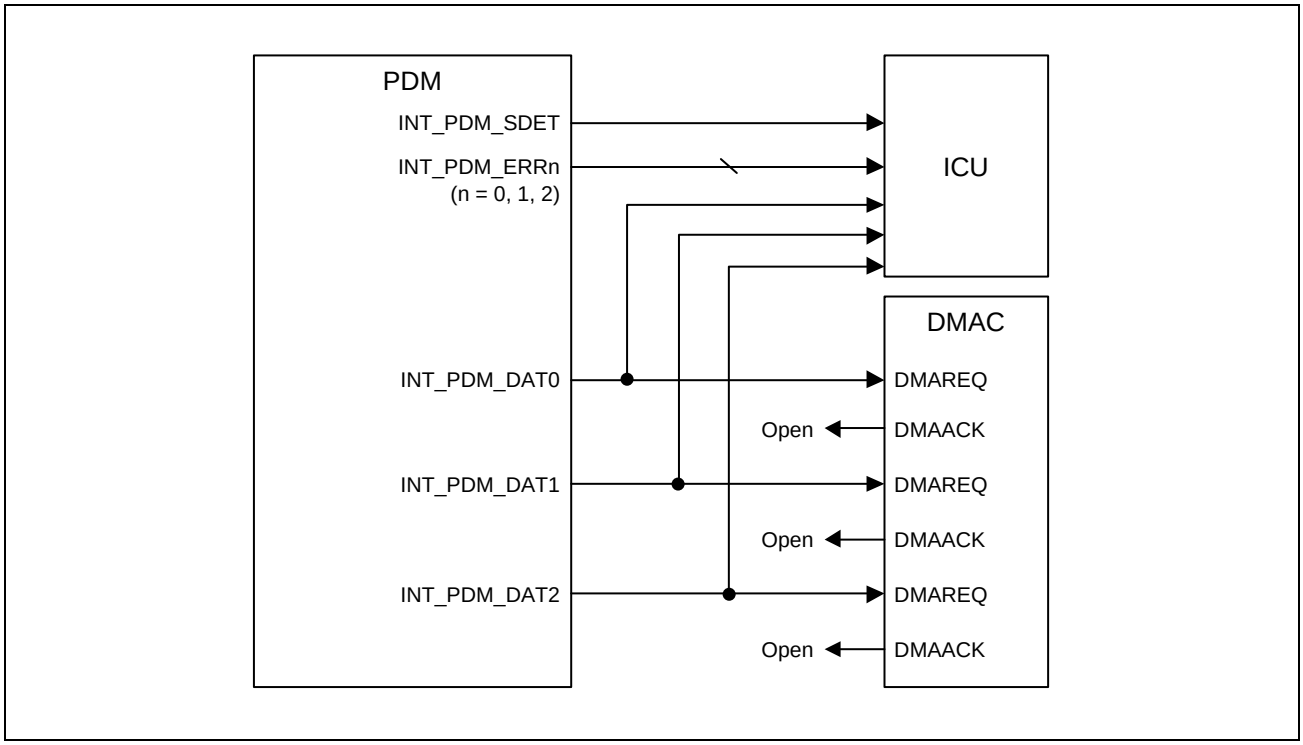


Figure 8.7-23 PDM Connection Configuration

This section describes the restrictions at DMA transfer and the active condition of the INT_PDM_DAT0/1/2 signal at PDM initialization.

1) Restrictions at DMA transfer:

The initialization setting on the DMA master is required in “Software configures DMAC/Software re-configures DMAC” in **Figure 8.7-19** of **8.7.4.3 Normal Processing Flow**. The restrictions on PDM DMA transfers are described below.

- DMA transfer mode (1): Single

  (One DMA transfer process is performed for one DMA request.)
  Reference value for the DMA master setting: TM register = 0b

- DMA transfer mode (2): Bus cycle mode

  (DMAREQ is masked in the bus cycle period (while DMAACK is active.))
  Reference value for the DMA master setting: AM[2:0] = 010b

- One DMA transfer: 32 bits

  Reference value for the DMA master: SDS register = 32 bits

- DMA REQ signal attribute: Level/High

  (The active polarity of the INT_PDM_DAT0/1/2 signal is high, so the DMA request signal must also be high.)
  Reference value for the DMA master: LVL = 1b, HIEN = 1b, LOEN = 0b

- DMA REQ signal active release timing: After the completion of data read from the PDM unit

  (DMA REQ is masked while DMA ACK is enabled.)
  Reference value for the DMA master: REQD = 0b

- DMA transfer data read register: PDDRRCHn (n = 0, 1, 2).DAT[19:0] register

2) Interrupt setting of the INT_PDM_DAT0/1/2 signal in the PDM receive initialization setting:

The interrupt setting for the INT_PDM_DAT0/1/2 signal is required in “Set Channel’s Data Buffer Control” in **Figure 8.7-17** of **8.7.4.1 Start Flow**.
As the receive buffer amount of PDM in this LSI is fixed at 64 stages, it is recommended to set PDDBCRCHn (n = 0, 1, 2).DATRITHR[2:0] = 4 (16 stages). Since a time difference may occur between the active timing of the INT_PDM_DAT0/1/2 signal and the request decision on the DMA master, the active timing of the INT_PDM_DAT0/1/2 signal must always be selected to be at least 2 stages.

8.7.4.8 Reset Clock Control

For the reset clock control, the following reset clock pins must be controlled.

Table 8.7-23 List of Reset Clocks

	Reset Signal	Clock Signal	Clock Source	Description
System 1	PRESETn	PCLK, PCLK_SFR	CPG unit (Divider clock for CPG PLL output)	For internal bus control circuit
System 2	CRESETn	CCLK	CPG unit (External system clock: 4.8 MHz fixed at 24 MHz divided by 5)	For external IF control circuit

The above reset clock pins are controlled by the CPG unit, and each reset clock can be controlled ON/OFF with the CPG register. When each reset is released, a feature is implemented on the CPG to stop the target clock before and after the reset is released.

For each system, reset and clock control should be performed according to the following procedure.

However, the reset and clock control between the two systems below are independent, and there are no restrictions on the order of settings between the systems.

Furthermore, when controlling the reset clock, the PDM unit should be stopped and accesses from related units should also be stopped.

1. Reset (PRESETn) and clock (PCLK, PCLK_SFR) for the internal bus control circuit:
   — PDM operation start condition (reset/clock supply): clock supply → reset release
   — PDM stop condition (reset/clock stop): reset enabled → clock stop
   — PDM reset (reset ON/OFF, clock supply continued): Reset can be enabled or disabled at any timing

2. Reset (CRESETn) and clock (CCLK) for the external IF control circuit:
   — PDM operation start condition (reset/clock supply): clock supply → reset release
   — PDM stop condition (reset/clock stop): reset enabled → clock stop
   — PDM reset (reset ON/OFF, clock supply continued): Reset can be enabled or disabled at any timing

As for PCLK/PCLK_SFR, the CPG unit can change the clock frequency variable (100 MHz, 50 MHz, 25 MHz, 12.5 MHz, 3.125 MHz). When lowering the PCLK/PCLK_SFR frequency, the performance of reading voice data from the PDM unit will be degraded. Use caution when lowering the frequency.

8.7.5 Interrupt

8.7.5.1 Interrupt List

Table 8.7-24 is a list of interrupt signals for this unit. For details on interrupts, refer to **4.6 Interrupt Controller**.

Table 8.7-24 PDM-IF Interrupt List

No.	Signal Name	Interrupt Name	Number of Signals	Pulse or Level	Related Register Bit Name		Function (Related Section)
					Common Register	Channel Register	
1	INT_PDM_DATn (n = 0, 1, 2)	Data reception interrupt CHn (n = 0, 1, 2)	3	Level	PDCSR.DRFn	PDSRCHn (n = 0, 1, 2).DRF	**8.7.3.7 Data Buffer**
2	INT_PDM_SDET	Sound detection interrupt (3-channel status flags are ORed.)	1	Level	PDCSR.SDFn	PDSRCHn (n = 0, 1, 2).SDF	**8.7.3.6 Sound Detection**
3	INT_PDM_ERRn (n = 0, 1, 2)	Error detection interrupt CHn (n = 0, 1, 2)	3	Level	PDCSR.EDFn	PDSRCHn (n = 0, 1, 2).SCDF	**8.7.3.8 Short-Circuit Detection**
						PDSRCHn (n = 0, 1 ,2).OVLDF	**8.7.3.9 Overvoltage Detection**
						PDSRCHn (n = 0, 1, 2).OVUDF	**8.7.3.9 Overvoltage Detection**
						PDSRCHn (n = 0, 1, 2) BFOWDF	**8.7.3.7 Data Buffer**

8.7.6 Usage Notes

8.7.6.1 Data Reception Interrupt Threshold Register Setting

A data reception interrupt INT_PDM_DATn is issued as a level output. Therefore, if the number of samples in buffer still exceeds the data reception interrupt threshold set in PDDBCRCHn.DATRITHR[2:0] when data reading of DMA finishes, the interrupt controller might not detect the edge of INT_PDM_DATn. The situation happens if data reading time exceeds sampling rate*1 x threshold. Set the appropriate threshold so that the interrupt controller can detect the interrupt.

In case the threshold cannot avoid the situation, read PDDBCRCHn.DATRITHR[2:0] and re-start DMA by software after reading the data.

Note 1. Sampling rate: Time to spend to store a sample in buffer.

8.7.6.2 Frequency Characteristics of Filters with Default Settings

The following figure and table show the frequency characteristics of the filters with the default settings*1 for reference information.

Note 1. SINC filter output sampling frequency = 32 kHz
PDM output sampling frequency = 16 kHz

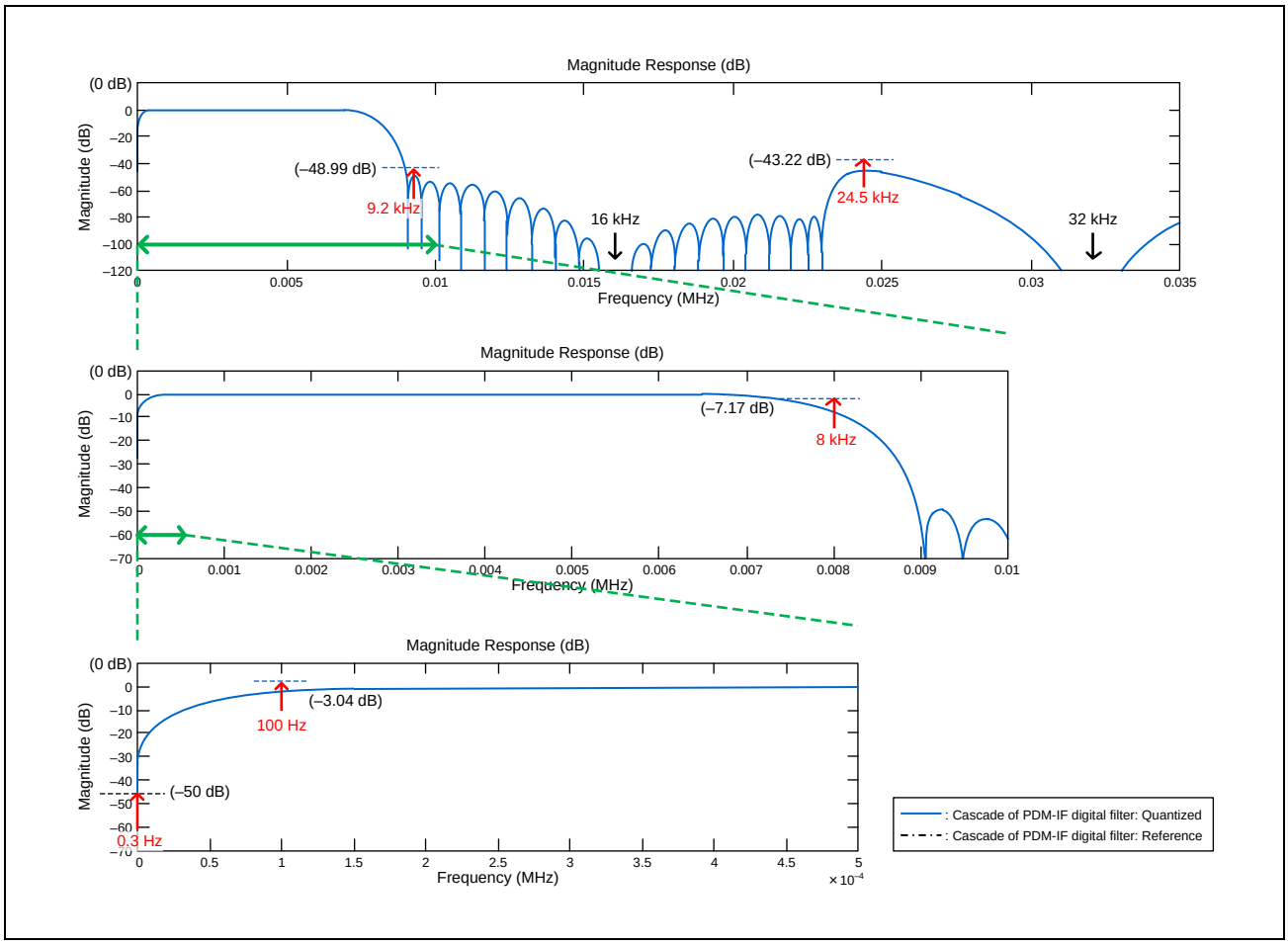


Figure 8.7-24 Frequency Characteristics of Filters with Default Settings

Table 8.7-25 Frequency Characteristics of Filters with Default Settings

Frequency (Hz)		Magnitude (dB)
Cut-off frequency of LPF	24.5 kHz	−43.22
	…	…
	9.2 kHz	−48.99
	8 kHz	−7.17
Passband	7 kHz	−0.39
	6 kHz	−0.07
	…	…
	150 Hz	−1.63
	100 Hz	−3.04
Cut-off frequency of HPF	50 Hz	−7.02
	0.3 Hz	−50.00

SECTION 9 IMAGE

9.1 Image Overview

This sections describes the Image unit of this LSI. For details, refer to the sections of each unit.

■ Camera Data Receive Unit (CRU) (See 9.2)

This LSI equips 1ch of CRU.

■ LCD Controller (LCDC) (See 9.4)

This LSI equips 2ch LCD Controllers.

■ MIPI DSI (See 9.5)

Conforms to the following standards.

- MIPI® Alliance Specification for Display Serial Interface Version 1.3.1
- MIPI® Alliance Specification for D-PHY Version 1.2

■ H.265/H.264 Multi Codec (VCD) (See 9.6)

■ 3D Graphics Engine (GE3D) (See 9.7)

This LSI includes the Arm® Mali™-G52 GPU as a 3D Graphics Engine (GE3D). This unit is an optional function.

■ Video Signal Processor (VSP) (See 9.9)

■ Frame Data Processor (FDP) (See 9.10)

■ Low Voltage Differential Signaling (LVDS) (See 9.11)

This LSI equips 2 LVDS interfaces.

9.1.1 Variation of Display Output

Figure 9.1-1 shows the display interface connection.

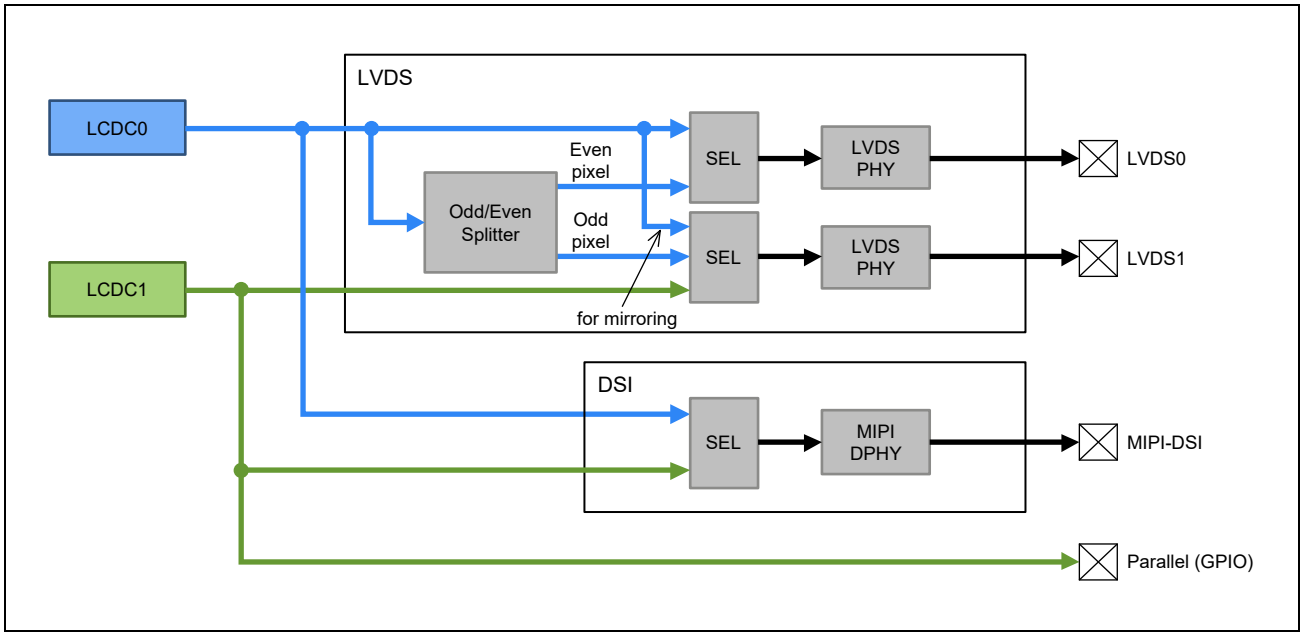


Figure 9.1-1 Display Interface Connection Diagram

This LSI has 2ch LCD controllers (LCDC0 and LCDC1), so it can output up to two different screens to LVDS0, LVDS1, Parallel and MIPI-DSI interfaces.

Table 9.1-1 shows the available vatiation of the display output.

Table 9.1-1 Variation of Display Output

Pattern	LVDS0	LVDS1	Parallel	MIPI-DSI	The Number of Output Displays
	Up to 1366×768, 60 fps Up to 1280×800, 60 fps	Up to 1366×768, 60 fps Up to 1280×800, 60 fps			
	Dual-link: Up to 1920×1200, 60 fps		Up to 1280×800, 60 fps	Up to 1920×1200, 60 fps	
Pattern 1-0	✓ (from LCDC0)	—	—	—	1
Pattern 1-1	—	✓ (from LCDC1)	—	—	
Pattern 1-2	—	—	✓ (from LCDC1)	—	
Pattern 1-3	—	—	—	✓ (from LCDC1)	
Pattern 1-4	—	—	—	✓ (from LCDC0)	
Pattern 1-5	✓ Dual-link (from LCDC0)		—	—	
Pattern 2-1	✓ (from LCDC0)	✓ (from LCDC1)	—	—	2
Pattern 2-2	✓ (from LCDC0)	—	✓ (from LCDC1)	—	
Pattern 2-3	✓ (from LCDC0)	—	—	✓ (from LCDC1)	
Pattern 2-4	✓ Dual-link (from LCDC0)		✓ (from LCDC1)	—	
Pattern 2-5	✓ Dual-link (from LCDC0)		—	✓ (from LCDC1)	
Pattern 2-6	—	—	✓ (from LCDC1)	✓ (from LCDC0)	

This LSI can also output the same screen from LCDC0 to LVDS0 and LVDS1 interfaces. (Mirroring output)

Table 9.1-2 shows the available vatiation of the mirroring display output.

Remark Connect the same specification display to LVDS0 and LVDS1 interfaces.

Table 9.1-2 Variation of Mirroring Display Output

Pattern	LVDS0 Up to 1366×768, 60 fps Up to 1280×800, 60 fps	LVDS1 Up to 1366×768, 60 fps Up to 1280×800, 60 fps	Parallel Up to 1280×800, 60 fps	MIPI-DSI Up to 1920×1200, 60 fps	The Number of Output Displays
Pattern 2-0	✓ (from LCDC0)	✓ (from LCDC0)	—	—	2
Pattern 3-0	✓ (from LCDC0)	✓ (from LCDC0)	✓ (from LCDC1)	—	3
Pattern 3-1	✓ (from LCDC0)	✓ (from LCDC0)	—	✓ (from LCDC1)	

SECTION 9 IMAGE

9.2 Camera Data Receiver Unit (CRU)

9.2.1 Overview

CRU has an MIPI CSI-2 input and then decodes the data type specified by the packet, handles conversion of the color space of pixels, LUT conversion, pixel format conversion, and image conversion such as demosaicing and obtaining statistics. After converting processed data to various pixel formats, it transfers the data to external memory.

CRU consists of the MIPI CSI2 block and an image converter. The main functions of these blocks are as follows.

MIPI CSI2

The MIPI CSI2 block receives signals compliant with the MIPI CSI-2 V2.1(DPHY V1.2) standard, extracts video signals from various packets, and sends data to the image converter, which is the next stage.

The details of its functions are described in **9.2.3 MIPI CSI2**.

Image Converter

The image converter receives video data from the MIPI CSI2 block, that is, from the previous stage, and handles image processing corresponding to the given data. Data to which image processing has been applied are initially stored in a FIFO buffer and then transferred to external memory.
The details of its functions are described in **9.2.4 Image Converter**.

9.2.2 Connected Units

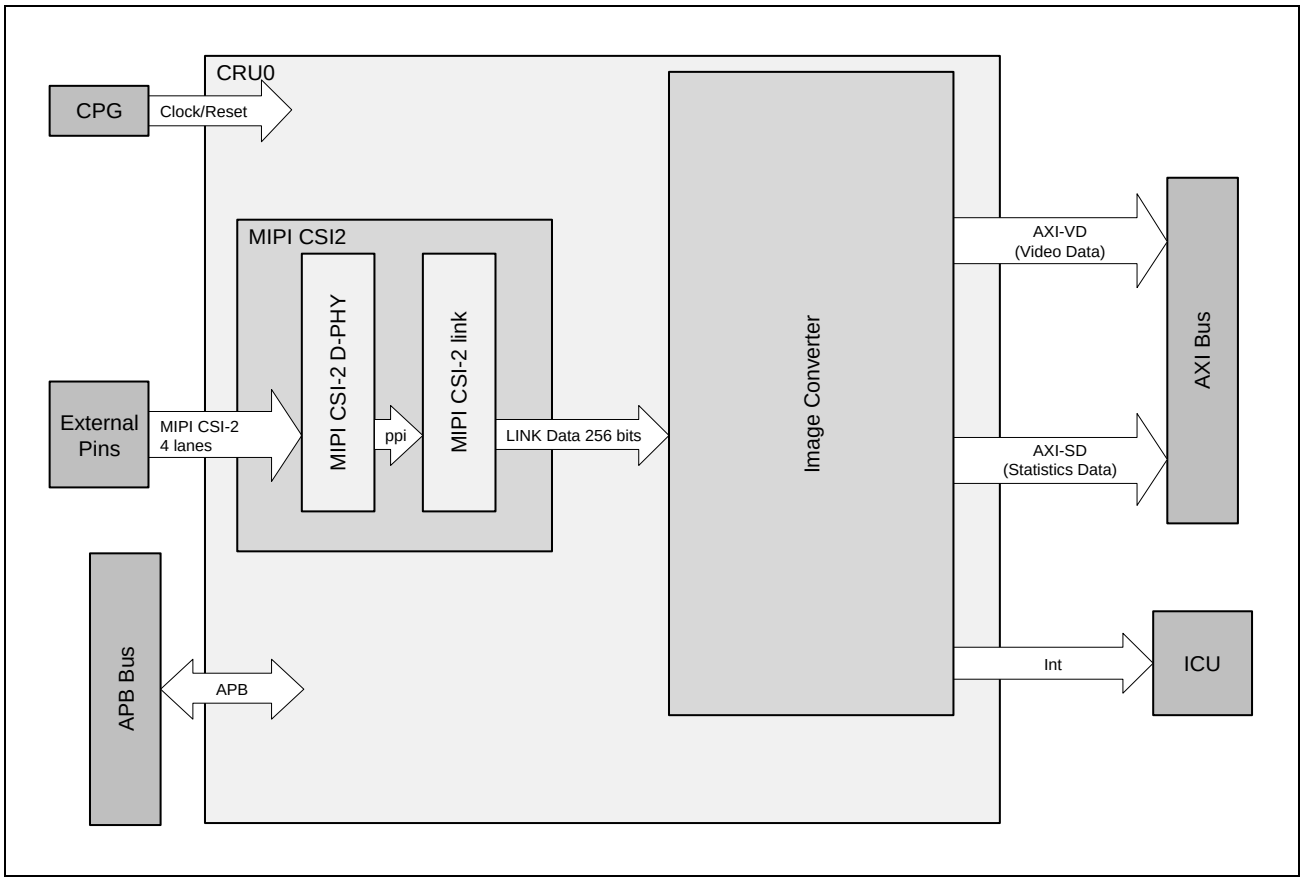


Figure 9.2-1 Block Diagram of CRU0

9.2.3 MIPI CSI2

9.2.3.1 Overview

MIPI CSI2 (camera serial interface 2) receives signals compliant with the DPHY standard from a camera module connected to external pins and decodes them as image data.

Data decoded by this module are sent to the image converter block as the next stage.

9.2.3.1.1 Features

This module has the following functions.

COMMON

1) Data input range

[Pixel size]

- Maximum number of pixels: 8.8 Mpixels; minimum number of pixels: QVGA (320 × 240) = 76.8 Kpixels
- Maximum number of effective pixels in the horizontal direction: 4096 pixels / 8192 pixels*1
- Maximum number of effective lines in the vertical direction: 4096 lines / 8192 lines*1

[Other than images]

- Data size range of input long packets: 1 to 24576 bytes

Note 1. 4096 pixels/lines when image processing and non-image processing are selected
8192 pixels/lines when full bypass is selected

2) Restrictions on the input bandwidth

Use the input bandwidth within the range which satisfies all conditions in **Table 9.2-1** and **Table 9.2-2**.

Table 9.2-1 Maximum Rate per Lane

	Transfer Rate per Lane (Mbps)		Transfer Rate for All Lanes (Mbps)	
Lane to be Used	MIN	MAX	MIN	MAX
1	80	2100	80	2100
2	80	2100	160	4200
4	80	2100	320	8400

Table 9.2-2 Maximum Transfer Rate per Format for Total of Four Lanes (1/2)

CRUnCTRL.ATH	Input Format	MIPI CSI-2 Transfer Rate (Mbps (max.))
0	YUV422 8-bit	8400
0	YUV422 10-bit	8400
0	RGB444	8400
0	RGB555	8400
0	RGB565	8400
0	RGB666	8400
0	RGB888	8400

Table 9.2-2 Maximum Transfer Rate per Format for Total of Four Lanes (2/2)

CRUnCTRL.ATH	Input Format	MIPI CSI-2 Transfer Rate (Mbps (max.))
0	RAW6	3751
0	RAW7	4359
0	RAW8	5040
0	RAW10	6203
0	RAW12	7331
0	RAW14	8400
0	RAW16	8400
0	RAW20	8400
1 (full bypass)	MIPI recommended memory storage format	8400

3) Restrictions on blanking

At least the time given below is required during the period from the last valid data on a line to the first valid data on the next line.

- When demosaicing is in use: (300 + H-size) × 1/vclk (MHz) or more
  e.g. when vclk = 630 MHz, for VGA (640x480): (300 + 640) × 1/630 (MHz) = approx. 1.50 µs or more
- When demosaicing is not in use: 300 × 1/vclk (MHz) or more
  e.g. when vclk = 630 MHz: 300 × 1/630 (MHz) = approx. 477 ns or more

4) Compliant with the specification for camera serial interface 2 (CSI-2)^{SM}

5) Receiver error detection and reporting

- D-PHY level errors
- Packet level errors
- Protocol decoding level errors

D-PHY

1) Support MIPI D-PHY V1.2 (80 Mbps to 2100 Mbps/lane)

2) Support 1, 2, and 4 lanes

3) Support for receiver deskewing by D-PHY

- In operation at over 1.5 Gbps, periodic deskewing is essential.
- The timing for periodic deskewing calibration: 2^{13} to 10 µs
- The timing for initial deskewing calibration: 2^{15} UI to 100 µs

4) D-PHY Data lane type: CIL-SFEN (forward ULPS only).

5) Low Power Mode Max Speed 10 Mbps

LINK

1) Support MIPI CSI-2 V2.1

2) Support data de-scrambling

3) Support latency reduction and transport efficiency (LRTE)

4) ECC 1-bit error correction and 2-bit error detection in packet headers

5) CRC checking for payload data

6) Support 16 virtual channels

Note: Although the MIPI CSI2 block can receive 16 virtual channels (VC0 to VC15), 4 virtual channels can be selected from VC0 to VC15 in the image converter. Refer to **9.2.4.1.1 Features** for more detail.

7) Number of stages of the FIFO for generic short packets: 16

8) Support data interleaving

- Data type interleaving
  - Packet-level interleaved data transmission
  - Frame-level interleaved data transmission
- Virtual channel identifier interleaving

9.2.3.1.2 Block Diagram

Figure 9.2-2 is a block diagram of MIPI CSI2.

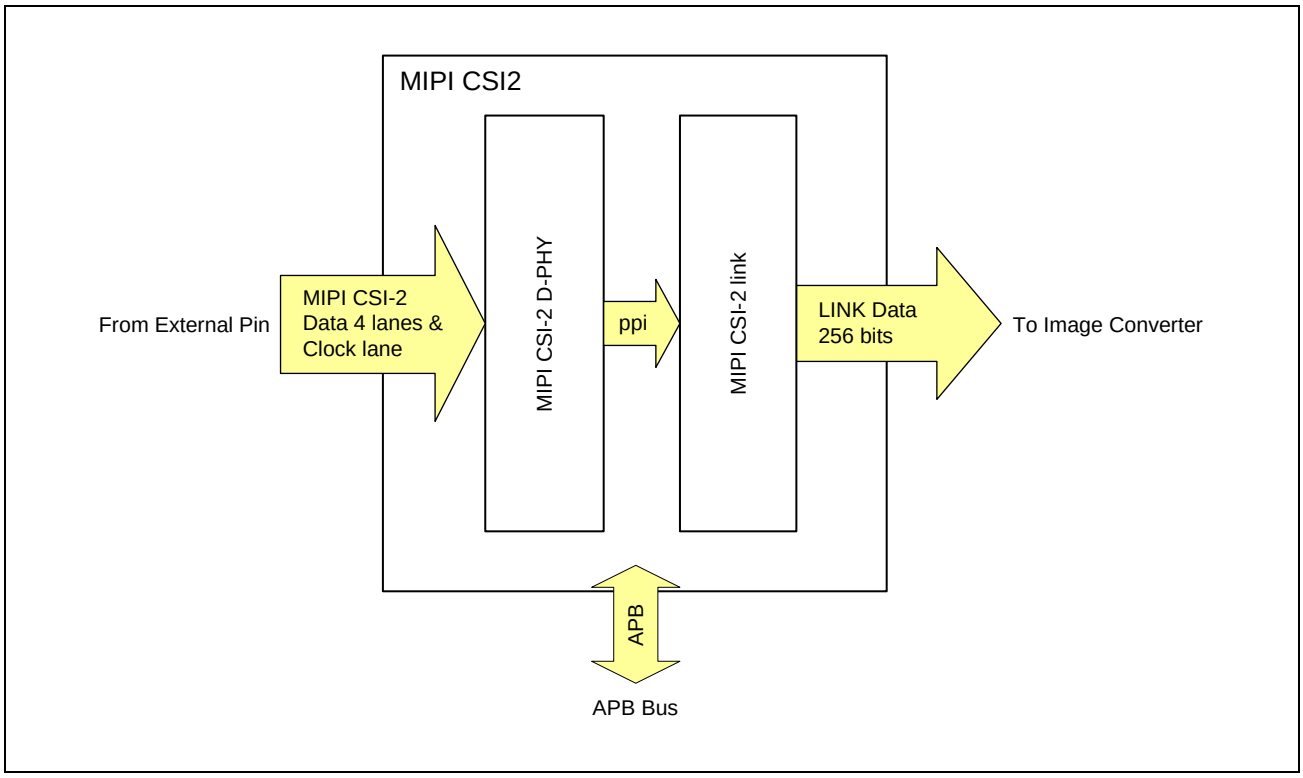


Figure 9.2-2 Block Diagram of MIPI-CSI2

9.2.3.1.3 External Pins

Table 9.2-3 lists the input/output pins of MIPI CSI2.

Table 9.2-3 MIPI CSI2 Input/Output Pins

Category	Pin Name	Input/Output	Description
DPHY Interface for MIPI CSI-2	CSI0_CLKP	Input	MIPI CSI-2 Rx clock input for CRU0 (positive)
	CSI0_CLKN	Input	MIPI CSI-2 Rx clock input for CRU0 (negative)
	CSI0_DATA0P	Input	MIPI CSI-2 Rx serial data input lane 0 for CRU0 (positive)
	CSI0_DATA0N	Input	MIPI CSI-2 Rx serial data input lane 0 for CRU0 (negative)
	CSI0_DATA1P	Input	MIPI CSI-2 Rx serial data input lane 1 for CRU0 (positive)
	CSI0_DATA1N	Input	MIPI CSI-2 Rx serial data input lane 1 for CRU0 (negative)
	CSI0_DATA2P	Input	MIPI CSI-2 Rx serial data input lane 2 for CRU0 (positive)
	CSI0_DATA2N	Input	MIPI CSI-2 Rx serial data input lane 2 for CRU0 (negative)
	CSI0_DATA3P	Input	MIPI CSI-2 Rx serial data input lane 3 for CRU0 (positive)
	CSI0_DATA3N	Input	MIPI CSI-2 Rx serial data input lane 3 for CRU0 (negative)

9.2.3.2 MIPI CSI2 Registers

The base addresses of the CRUm (m = 0) registers are shown below.

Base Address Name	Base Address
<CRU0_base>	0_1600_0000h (5600_0000h*1, 4600_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

9.2.3.2.1 List of Registers

(1) CRU LINK Registers [m = 0]

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Module Configuration Register	CRUm_CSI2nMCG	0010_0400h	0400h	32
Reserve	-	-	0404h to 040Fh	-
Module Control Register 0	CRUm_CSI2nMCT0	0200_0004h	0410h	32
Reserve	-	-	0414h to 0417h	-
Module Control Register 2	CRUm_CSI2nMCT2	0000_0000h	0418h	32
Module Control Register 3	CRUm_CSI2nMCT3	0000_0000h	041Ch	32
Reserve	-	-	0420h to 0427h	-
Reset Control Register	CRUm_CSI2nRTCT	0000_0000h	0428h	32
Reset Status Register	CRUm_CSI2nRTST	0000_0000h	042Ch	32
Reserve	-	-	0430h to 043Fh	-
EPD Option Control Register	CRUm_CSI2nEPCT	0000_0000h	0440h	32
Reserve	-	-	0444h to 044Fh	-
Module Interrupt Status Register	CRUm_CSI2nMIST	0000_0000h	0450h	32
Reserve	-	-	0454h to 045Fh	-
Receive Data Type Enable Low Register	CRUm_CSI2nDTEL	0000_000Fh	0460h	32
Receive Data Type Enable High Register	CRUm_CSI2nDTEH	0000_0000h	0464h	32
Reserve	-	-	0468h to 046Fh	-
Receive Status Register	CRUm_CSI2nRXST	0000_0000h	0470h	32
Receive Status Clear Register	CRUm_CSI2nRXSC	0000_0000h	0474h	32
Receive Interrupt Enable Register	CRUm_CSI2nRXIE	0000_0000h	0478h	32
Reserve	-	-	047Ch to 047Fh	-
Data Lane 0 Status Register	CRUm_CSI2nDLST0	0000_0000h	0480h	32
Data Lane 0 Status Clear Register	CRUm_CSI2nDLSC0	0000_0000h	0484h	32
Data Lane 0 Interrupt Enable Register	CRUm_CSI2nDLIE0	0000_0000h	0488h	32
Reserve	-	-	048Ch to 048Fh	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Data Lane 1 Status Register	CRUm_CSI2nDLST1	0000_0000h	0490h	32
Data Lane 1 Status Clear Register	CRUm_CSI2nDLSC1	0000_0000h	0494h	32
Data Lane 1 Interrupt Enable Register	CRUm_CSI2nDLIE1	0000_0000h	0498h	32
Reserve	-	-	049Ch to 049Fh	-
Data Lane 2 Status Register	CRUm_CSI2nDLST2	0000_0000h	04A0h	32
Data Lane 2 Status Clear Register	CRUm_CSI2nDLSC2	0000_0000h	04A4h	32
Data Lane 2 Interrupt Enable Register	CRUm_CSI2nDLIE2	0000_0000h	04A8h	32
Reserve	-	-	04ACh to 04AFh	-
Data Lane 3 Status Register	CRUm_CSI2nDLST3	0000_0000h	04B0h	32
Data Lane 3 Status Clear Register	CRUm_CSI2nDLSC3	0000_0000h	04B4h	32
Data Lane 3 Interrupt Enable Register	CRUm_CSI2nDLIE3	0000_0000h	04B8h	32
Reserve	-	-	04BCh to 04FFh	-
Virtual Channel 0 Status Register	CRUm_CSI2nVCST0	0000_0000h	0500h	32
Virtual Channel 0 Status Clear Register	CRUm_CSI2nVCSC0	0000_0000h	0504h	32
Virtual Channel 0 Interrupt Enable Register	CRUm_CSI2nVCIE0	0000_0000h	0508h	32
Reserve	-	-	050Ch to 050Fh	-
Virtual Channel 1 Status Register	CRUm_CSI2nVCST1	0000_0000h	0510h	32
Virtual Channel 1 Status Clear Register	CRUm_CSI2nVCSC1	0000_0000h	0514h	32
Virtual Channel 1 Interrupt Enable Register	CRUm_CSI2nVCIE1	0000_0000h	0518h	32
Reserve	-	-	051Ch to 051Fh	-
Virtual Channel 2 Status Register	CRUm_CSI2nVCST2	0000_0000h	0520h	32
Virtual Channel 2 Status Clear Register	CRUm_CSI2nVCSC2	0000_0000h	0524h	32
Virtual Channel 2 Interrupt Enable Register	CRUm_CSI2nVCIE2	0000_0000h	0528h	32
Reserve	-	-	052Ch to 052Fh	-
Virtual Channel 3 Status Register	CRUm_CSI2nVCST3	0000_0000h	0530h	32
Virtual Channel 3 Status Clear Register	CRUm_CSI2nVCSC3	0000_0000h	0534h	32
Virtual Channel 3 Interrupt Enable Register	CRUm_CSI2nVCIE3	0000_0000h	0538h	32
Reserve	-	-	053Ch to 053Fh	-
Virtual Channel 4 Status Register	CRUm_CSI2nVCST4	0000_0000h	0540h	32
Virtual Channel 4 Status Clear Register	CRUm_CSI2nVCSC4	0000_0000h	0544h	32
Virtual Channel 4 Interrupt Enable Register	CRUm_CSI2nVCIE4	0000_0000h	0548h	32
Reserve	-	-	054Ch to 054Fh	-
Virtual Channel 5 Status Register	CRUm_CSI2nVCST5	0000_0000h	0550h	32
Virtual Channel 5 Status Clear Register	CRUm_CSI2nVCSC5	0000_0000h	0554h	32
Virtual Channel 5 Interrupt Enable Register	CRUm_CSI2nVCIE5	0000_0000h	0558h	32
Reserve	-	-	055Ch to 055Fh	-
Virtual Channel 6 Status Register	CRUm_CSI2nVCST6	0000_0000h	0560h	32
Virtual Channel 6 Status Clear Register	CRUm_CSI2nVCSC6	0000_0000h	0564h	32
Virtual Channel 6 Interrupt Enable Register	CRUm_CSI2nVCIE6	0000_0000h	0568h	32
Reserve	-	-	056Ch to 056Fh	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Virtual Channel 7 Status Register	CRUm_CSI2nVCST7	0000_0000h	0570h	32
Virtual Channel 7 Status Clear Register	CRUm_CSI2nVCSC7	0000_0000h	0574h	32
Virtual Channel 7 Interrupt Enable Register	CRUm_CSI2nVCIE7	0000_0000h	0578h	32
Reserve	-	-	057Ch to 057Fh	-
Virtual Channel 8 Status Register	CRUm_CSI2nVCST8	0000_0000h	0580h	32
Virtual Channel 8 Status Clear Register	CRUm_CSI2nVCSC8	0000_0000h	0584h	32
Virtual Channel 8 Interrupt Enable Register	CRUm_CSI2nVCIE8	0000_0000h	0588h	32
Reserve	-	-	058Ch to 058Fh	-
Virtual Channel 9 Status Register	CRUm_CSI2nVCST9	0000_0000h	0590h	32
Virtual Channel 9 Status Clear Register	CRUm_CSI2nVCSC9	0000_0000h	0594h	32
Virtual Channel 9 Interrupt Enable Register	CRUm_CSI2nVCIE9	0000_0000h	0598h	32
Reserve	-	-	059Ch to 059Fh	-
Virtual Channel 10 Status Register	CRUm_CSI2nVCST10	0000_0000h	05A0h	32
Virtual Channel 10 Status Clear Register	CRUm_CSI2nVCSC10	0000_0000h	05A4h	32
Virtual Channel 10 Interrupt Enable Register	CRUm_CSI2nVCIE10	0000_0000h	05A8h	32
Reserve	-	-	05ACh to 05AFh	-
Virtual Channel 11 Status Register	CRUm_CSI2nVCST11	0000_0000h	05B0h	32
Virtual Channel 11 Status Clear Register	CRUm_CSI2nVCSC11	0000_0000h	05B4h	32
Virtual Channel 11 Interrupt Enable Register	CRUm_CSI2nVCIE11	0000_0000h	05B8h	32
Reserve	-	-	05BCh to 05BFh	-
Virtual Channel 12 Status Register	CRUm_CSI2nVCST12	0000_0000h	05C0h	32
Virtual Channel 12 Status Clear Register	CRUm_CSI2nVCSC12	0000_0000h	05C4h	32
Virtual Channel 12 Interrupt Enable Register	CRUm_CSI2nVCIE12	0000_0000h	05C8h	32
Reserve	-	-	05CCh to 05CFh	-
Virtual Channel 13 Status Register	CRUm_CSI2nVCST13	0000_0000h	05D0h	32
Virtual Channel 13 Status Clear Register	CRUm_CSI2nVCSC13	0000_0000h	05D4h	32
Virtual Channel 13 Interrupt Enable Register	CRUm_CSI2nVCIE13	0000_0000h	05D8h	32
Reserve	-	-	05DCh to 05DFh	-
Virtual Channel 14 Status Register	CRUm_CSI2nVCST14	0000_0000h	05E0h	32
Virtual Channel 14 Status Clear Register	CRUm_CSI2nVCSC14	0000_0000h	05E4h	32
Virtual Channel 14 Interrupt Enable Register	CRUm_CSI2nVCIE14	0000_0000h	05E8h	32
Reserve	-	-	05ECh to 05EFh	-
Virtual Channel 15 Status Register	CRUm_CSI2nVCST15	0000_0000h	05F0h	32
Virtual Channel 15 Status Clear Register	CRUm_CSI2nVCSC15	0000_0000h	05F4h	32
Virtual Channel 15 Interrupt Enable Register	CRUm_CSI2nVCIE15	0000_0000h	05F8h	32
Reserve	-	-	05FCh to 05FFh	-
Power Management Status Register	CRUm_CSI2nPMST	000F_4000h	0600h	32
Power Management Status Clear Register	CRUm_CSI2nPMSC	0000_0000h	0604h	32
Power Management Interrupt Enable Register	CRUm_CSI2nPMIE	0000_0000h	0608h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	060Ch to 067Fh	-
Generic Short Packet Control Register	CRUm_CSI2nGSCT	0001_0000h	0680h	32
Generic Short Packet Status Register	CRUm_CSI2nGSST	0000_0000h	0684h	32
Generic Short Packet Status Clear Register	CRUm_CSI2nGSSC	0000_0000h	0688h	32
Generic Short Packet Interrupt Enable Register	CRUm_CSI2nGSIE	0000_0000h	068Ch	32
Generic Short Packet Register	CRUm_CSI2nGSHT	0008_0000h	0690h	32
Generic Short Packet Information Update Register	CRUm_CSI2nGSIU	0000_0000h	0694h	32

(2) CRU PHY Registers [m = 0]

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Slave Timing Control Register	CRUm_S_TIMCTL	0000_0000h	081Ch	32
Reserve	-	-	0820h to 0833h	-
Slave D-PHY Control Register (MSB)	CRUm_S_DPHYCTL_MSB	0000_0090h	0834h	32
DP/DN Swap Control Register	CRUm_SWAPCTL	0000_0000h	0838h	32
Reserve	-	-	083Ch to 086Bh	-
RX Skew Calibration Results 0	CRUm_S_ATB_OUT_0	0000_0000h	086Ch	32
RX Skew Calibration Results 1	CRUm_S_ATB_OUT_1	0000_0000h	0870h	32
RX Skew Calibration Results 2	CRUm_S_ATB_OUT_2	0000_0000h	0874h	32
RX Skew Calibration Results 3	CRUm_S_ATB_OUT_3	0000_0000h	0878h	32

9.2.3.2.2 Register Descriptions

(1) Module Configuration Register (CRUm_CSI2nMCG) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0400h
Initial Value : 0010_0400h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	GSNM[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	SDLN[3:0]				-	-	-	-	VER[3:0]			
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	GSNM[7:0]	10h	R	Indicate the number of stages of the generic short packet FIFO (16 stages).
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	SDLN[3:0]	4h	R	Indicate the supported maximum number of data lanes. 4h: Operable with four lanes, two lanes or one lane.
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	VER[3:0]	0h	R	Indicate the version of this core.

(2) Module Control Register 0 (CRUm_CSI2nMCT0) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0410h

Initial Value : 0200_0004h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	LFSREN	ECCV13	-	-	-	-	RVMD	-	EDMD	ZLMD
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R0W	RW	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	VDLN[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25	LFSREN	1h	RW	Enables or disables de-scrambling. 0b: Disables de-scrambling. 1b: Enables de-scrambling.
24	ECCV13	0h	RW	CSI-2 specification-compliant mode for ECC checking 0b: 26 bits to be checked by ECC 1b: 24 bits to be checked by ECC It is prohibited to set this bit to 1b when receiving a packet whose virtual channel ID is from 4 to 15.
23 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	-	0h	R0W	The read value is always 0b. Do not change the value of this bit from 0h.
19	RVMD	0h	RW	For a packet whose data type is 38h to 3Fh (“Reserved” in "Table 10 Data Type Classes" of the CSI-2 specification), set the receive operation mode to 0. 0b: Discards the data (notifies an ErrID). 1b: Receives data as a long packet (outputs data to the image processing module when the corresponding bit in CSI2nDTEH is 1, or discards the data and notifies an ErrID when the corresponding bit in CSI2nDTEH is 0).
18	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	EDMD	0h	RW	Notifies ErrFrameData when an ECC 2-bit error or a packet of less than four bytes is received between FS and FE. 0b: Does not notify ErrFrameData. 1b: Notifies ErrFrameData.
16	ZLMD	0h	RW	Sets whether to output a long packet whose word count is 0 to the image processing module. Set this bit to 0. 0b: Output 1b: No output
15 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	VDLN[3:0]	4h	RW	Set the number of used data lanes. Change this field when CSI2nMCT3.RXEN = 0. A value equal to or less than the CSI2nMCG.SDLN value can be specified. 1h: Operation with one lane 2h: Operation with two lanes 4h: Operation with four lanes Others: Setting prohibited

(3) Module Control Register 2 (CRUm_CSI2nMCT2) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0418h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	FRRSKW[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	FRRCLK[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24 to 16	FRRSKW[8:0]	0h	RW	Internal setting parameter Set the following value by using the ROUNDDOWN function that rounds down the decimal places. (ROUNDDOWN((3 × vclk frequency/hsclk1 frequency), 0) + 1) where vclk = 630 MHz, hsclk1 = transfer rate/8 Example: Set 10h when the vclk frequency is 630 MHz and the transfer rate is 1 Gbps. ROUNDDOWN((3 × 630/(1000/8)), 0) + 1 = 16d
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8 to 0	FRRCLK[8:0]	0h	RW	Parameter for the IP to determine whether packets have been received if RxWordClkHS stops at the timing when the IP cannot recognize that RxValidHS is low. Set the following value by using the ROUNDDOWN function that rounds down the decimal places. (ROUNDDOWN((1.5 × vclk frequency/hsclk1 frequency), 0) + 1) where vclk = 630 MHz, hsclk1 = transfer rate/8 Example: Set 8h when the vclk frequency is 630 MHz and the transfer rate is 1 Gbps. ROUNDDOWN((1.5 × 630/(1000/8)), 0) + 1 = 8d

(4) Module Control Register 3 (CRUm_CSI2nMCT3) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 041Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RXEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	RXEN	0h	RW	Enables LINK data reception. 0b: Disables reception. 1b: Enables reception.

(5) Reset Control Register (CRUm_CSI2nRTCT) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0428h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	VSRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	VSRST	0h	R0W1	The read value is always 0b. Writing 1b to this bit resets the internal LINK signal by a software reset.

(6) Reset Status Register (CRUm_CSI2nRTST) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 042Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	VSRSTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	VSRSTS	0h	R	Indicates the execution status of the software reset generated by CSI2nRTCT.VSRST. 0b: Not during a reset 1b: During a reset

(7) EPD Option Control Register (CRUm_CSI2nEPCT) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0440h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EPDEN	SSP[14:0]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EPDOP	SLP[14:0]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	EPDEN	0h	RW	Enables the EPD. 0b: Disables the EPD. 1b: Enables the EPD.
30 to 16	SSP[14:0]	0h	RW	Specify the number of spacers inserted after a short packet. This field is valid when EPDEN = 1. The following shows the specifiable minimum number of spacers when EPDEN = 1. Setting a value less than the minimum value is prohibited. When EPDOP = 1b, the minimum spacer value is (1) The larger value of the following two values: • Minimum number of spacers due to the D-PHY setting • Minimum number of spacers due to the number of used lanes (2) Minimum number of spacers due to the D-PHY setting: The minimum value can be found by using the ROUNDUP function that rounds up the decimal places. ROUNDUP((105/UI[ns] + 20) / 8), 0) (3) Minimum number of spacers due to the number of used lanes: 1 lane: minimum spacer = 64 2 lanes: minimum spacer = 32 4 lanes: minimum spacer = 32
15	EPDOP	0h	RW	Selects an EPD option. Set this bit to 1b. 0b: D-PHY EPD Option 1 (0 cannot be set when EPDEN = 1) 1b: D-PHY EPD Option 2
14 to 0	SLP[14:0]	0h	RW	Specify the number of spacers inserted after a long packet. This field is valid when EPDEN = 1. When EPDEN = 1, the minimum number of spacers is the value described in the SSP field. Setting a value less than the minimum value is prohibited.

(8) Module Interrupt Status Register (CRUm_CSI2nMIST) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0450h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VC15S	VC14S	VC13S	VC12S	VC11S	VC10S	VC9S	VC8S	VC7S	VC6S	VC5S	VC4S	VC3S	VC2S	VC1S	VC0S
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	RXS	GSTS	PMS	-	-	-	-	DL3S	DL2S	DL1S	DL0S
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	VC15S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 15. Check the CSI2nVCST15 register for the details of the interrupt source.
30	VC14S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 14. Check the CSI2nVCST14 register for the details of the interrupt source.
29	VC13S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 13. Check the CSI2nVCST13 register for the details of the interrupt source.
28	VC12S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 12. Check the CSI2nVCST12 register for the details of the interrupt source.
27	VC11S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 11. Check the CSI2nVCST11 register for the details of the interrupt source.
26	VC10S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 10. Check the CSI2nVCST10 register for the details of the interrupt source.
25	VC9S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 9. Check the CSI2nVCST9 register for the details of the interrupt source.
24	VC8S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 8. Check the CSI2nVCST8 register for the details of the interrupt source.
23	VC7S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 7. Check the CSI2nVCST7 register for the details of the interrupt source.
22	VC6S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 6. Check the CSI2nVCST6 register for the details of the interrupt source.
21	VC5S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 5. Check the CSI2nVCST5 register for the details of the interrupt source.
20	VC4S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 4. Check the CSI2nVCST4 register for the details of the interrupt source.
19	VC3S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 3. Check the CSI2nVCST3 register for the details of the interrupt source.
18	VC2S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 2. Check the CSI2nVCST2 register for the details of the interrupt source.
17	VC1S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 1. Check the CSI2nVCST1 register for the details of the interrupt source.
16	VC0S	0h	R	When this bit is set to 1b, there is an interrupt source for virtual channel 0. Check the CSI2nVCST0 register for the details of the interrupt source.
15 to 11	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	RXS	0h	R	When this bit is set to 1b, there is an interrupt source for the reception in general. Check the CSI2nRXST register for the details of the interrupt source.
9	GSTS	0h	R	When this bit is set to 1b, there is an interrupt source for reception of generic short packets. Check the CSI2nGSST register for the details of the interrupt source.
8	PMS	0h	R	When this bit is set to 1b, there is an interrupt source for the power management. Check the CSI2nPMST register for the details of the interrupt source.
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
3	DL3S	0h	R	When this bit is set to 1b, there is an interrupt source for data lane 3. Check the CSI2nDLST3 register for the details of the interrupt source.
2	DL2S	0h	R	When this bit is set to 1b, there is an interrupt source for data lane 2. Check the CSI2nDLST2 register for the details of the interrupt source.
1	DL1S	0h	R	When this bit is set to 1b, there is an interrupt source for data lane 1. Check the CSI2nDLST1 register for the details of the interrupt source.
0	DL0S	0h	R	When this bit is set to 1b, there is an interrupt source for data lane 0. Check the CSI2nDLST0 register for the details of the interrupt source.

(9) Receive Data Type Enable Low Register (CRUm_CSI2nDTEL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0460h
Initial Value : 0000_000Fh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTEN[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTEN[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DTEN[31:0]	0000_000Fh	RW (R)	Receive data of a packet in which data type is one of the following. 0b: Does not receive data. 1b: Receives data. Note: After multiple data types are selected here, only one data type is selected by the INF[5:0] bits in the ICnIPMC_C0/1/2/3 register in the image processing module. See **Table 9.2-4** for details on setting this bit.

Table 9.2-4 Receive Data Type Enable Low Register (CSI2nDTEL)

Bit	Data Type	R/W	Description
31	YUV422 10-bit	RW	When this bit is set to 1b, data is output to the image processing module.
30	YUV422 8-bit	RW	
29	YUV420 10-bit	RW	
28	YUV420 8-bit	RW	
27	Reserved	RW	
26	Legacy YUV420 8-bit	RW	
25	YUV420 10-bit	RW	
24	YUV420 8-bit	RW	
23	Reserved	RW	
22	Generic long packet data type 4	RW	
21	Generic long packet data type 3	RW	
20	Generic long packet data type 2	RW	
19	Generic long packet data type 1	RW	
18	Embedded 8-bit non-image data	RW	
17	Blanking data	RW	
16	Null	RW	
15	Generic Short Packet Code 8	RW	Fixed to 1b for each bit.
14	Generic Short Packet Code 7	RW	When each bit is set to 1b and CSI2nGSCT.GFIF = 1, data is output to the generic short packet FIFO.
13	Generic Short Packet Code 6	RW	
12	Generic Short Packet Code 5	RW	
11	Generic Short Packet Code 4	RW	
10	Generic Short Packet Code 3	RW	
9	Generic Short Packet Code 2	RW	
8	Generic Short Packet Code 1	RW	
7 to 4	Reserved	R	Fixed to 0.
3 to 0	Reserved	R	Fixed to 1.

(10) Receive Data Type Enable High Register (CRUm_CSI2nDTEH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0464h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTEN[63:48]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTEN[47:32]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DTEN[63:32]	0000_0000h	RW	Receive data of a packet in which data type is one of the following. 0b: Does not receive data. 1b: Receives data. Note: See **Table 9.2-5** for details on setting this bit.

Table 9.2-5 Receive Data Type Enable High Register (CSI2nDTEH)

Bit	Data Type	R/W	Description
31	Reserved	RW	Changing the reserved bit from the initial value is prohibited.
30 to 24	Reserved	RW	When this bit is set to 1b and CSI2nMCT0.RVMD = 1, data is output to the image processing module.
23	User Defined 8-bit Data Type 8	RW	When this bit is set to 1b, data is output to the image processing module.
22	User Defined 8-bit Data Type 7	RW	
21	User Defined 8-bit Data Type 6	RW	
20	User Defined 8-bit Data Type 5	RW	
19	User Defined 8-bit Data Type 4	RW	
18	User Defined 8-bit Data Type 3	RW	
17	User Defined 8-bit Data Type 2	RW	
16	User Defined 8-bit Data Type 1	RW	
15	RAW20	RW	
14	RAW16	RW	
13	RAW14	RW	
12	RAW12	RW	
11	RAW10	RW	
10	RAW8	RW	
9	RAW7	RW	
8	RAW6	RW	
7 to 5	Reserved	RW	
4	RGB888	RW	
3	RGB666	RW	
2	RGB565	RW	
1	RGB555	RW	
0	RGB444	RW	

(11) Receive Status Register (CRUm_CSI2nRXST) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0470h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RACTDET	RACT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FRM15	FRM14	FRM13	FRM12	FRM11	FRM10	FRM9	FRM8	FRM7	FRM6	FRM5	FRM4	FRM3	FRM2	FRM1	FRM0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	RACTDET	0h	R	When this bit is set to 1b, CSI2nRXST.RACT = 1 has been detected. Write 1 to CSI2nRXSC.RACTDETC to clear this bit.
16	RACT	0h	R	Indicates that this IP is receiving packets.
15	FRM15	0h	R	Indicates that frames on virtual channel = 15 are being processed.
14	FRM14	0h	R	Indicates that frames on virtual channel = 14 are being processed.
13	FRM13	0h	R	Indicates that frames on virtual channel = 13 are being processed.
12	FRM12	0h	R	Indicates that frames on virtual channel = 12 are being processed.
11	FRM11	0h	R	Indicates that frames on virtual channel = 11 are being processed.
10	FRM10	0h	R	Indicates that frames on virtual channel = 10 are being processed.
9	FRM9	0h	R	Indicates that frames on virtual channel = 9 are being processed.
8	FRM8	0h	R	Indicates that frames on virtual channel = 8 are being processed.
7	FRM7	0h	R	Indicates that frames on virtual channel = 7 are being processed.
6	FRM6	0h	R	Indicates that frames on virtual channel = 6 are being processed.
5	FRM5	0h	R	Indicates that frames on virtual channel = 5 are being processed.
4	FRM4	0h	R	Indicates that frames on virtual channel = 4 are being processed.
3	FRM3	0h	R	Indicates that frames on virtual channel = 3 are being processed.
2	FRM2	0h	R	Indicates that frames on virtual channel = 2 are being processed.
1	FRM1	0h	R	Indicates that frames on virtual channel = 1 are being processed.
0	FRM0	0h	R	Indicates that frames on virtual channel = 0 are being processed.

(12) Receive Status Clear Register (CRUm_CSI2nRXSC) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0474h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RACTDETC	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	RACTDETC	0h	R0W1	The read value is always 0b. Write 1 to this bit to clear CSI2nRXST.RACTDET. Writing 0b to this bit has no effect.
16 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(13) Receive Interrupt Enable Register (CRUm_CSI2nRXIE) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0478h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RACTDETE	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	RACTDETE	0h	RW	Enables interrupts generated by CSI2nRXST.RACTDET. 0b: Does not assert csi2_int_rx when CSI2nRXST.RACTDET = 1. 1b: Asserts csi2_int_rx when CSI2nRXST.RACTDET = 1.
16 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(14) Data Lane (N) Status Register (CRUm_CSI2nDLST(N)) (m = 0)

(N) indicates a logical data lane number.
(N) = 0, 1, 2, or 3

Access Size : 32 bits
Address : <CRUm_base> + 0480h + (N) x 010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	ULP(N)	-	-	-	-	-	-	RUL(N)	EUL(N)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	EES(N)	ECT(N)	-	ESH(N)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	ULP(N)	0h	R	Indicates the RxUlpsEsc status of logical data lane (N).
23 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	RUL(N)	0h	R	When this bit is set to 1b, logical data lane (N) has transitioned to the ULPS receive mode (RxUlpsEsc). Write 1 to CSI2nDLSC(N).RULC(N) to clear this bit.
16	EUL(N)	0h	R	When this bit is set to 1b, logical data lane (N) has exited from the ULPS receive mode (RxUlpsEsc). Write 1 to CSI2nDLSC(N).EULC(N) to clear this bit.
15 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	EES(N)	0h	R	When this bit is set to 1b, an escape mode entry error (ErrESC) has occurred on logical data lane (N). Write 1 to CSI2nDLSC(N).EESC(N) to clear this bit.
2	ECT(N)	0h	R	When this bit is set to 1b, a control error (ErrControl) has occurred on logical data lane (N). Write 1 to CSI2nDLSC(N).ECTC(N) to clear this bit.
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	ESH(N)	0h	R	When this bit is set to 1b, an SoT error (ErrSotHS) has occurred on logical data lane (N). Write 1 to CSI2nDLSC(N).ESHC(N) to clear this bit.

(15) Data Lane (N) Status Clear Register (CRUm_CSI2nDLSC(N)) (m = 0)

(N) indicates a logical data lane number.
(N) = 0, 1, 2, or 3

Access Size : 32 bits
Address : <CRUm_base> + 0484h + (N) x 010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RULC (N)	EULC (N)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	EESC (N)	ECTC (N)	-	ESHC (N)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W1	R0W1	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	RULC(N)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nDLST(N).RUL(N). Writing 0b to this bit has no effect.
16	EULC(N)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nDLST(N).EUL(N). Writing 0b to this bit has no effect.
15 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	EESC(N)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nDLST(N).EES(N). Writing 0b to this bit has no effect.
2	ECTC(N)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nDLST(N).ECT(N). Writing 0b to this bit has no effect.
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	ESHC(N)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nDLST(N).ESH(N). Writing 0b to this bit has no effect.

(16) Data Lane (N) Interrupt Enable Register (CRUm_CSI2nDLIE(N)) (m = 0)

(N) indicates a logical data lane number.
(N) = 0, 1, 2, or 3

Access Size : 32 bits
Address : <CRUm_base> + 0488h + (N) x 010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RULE (N)	EULE (N)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	EESE (N)	ECTE (N)	-	ESHE (N)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	RULE(N)	0h	RW	Enables interrupts generated by CSI2nDLST(N).RUL(N). 0b: Does not assert csi2_int_dl when CSI2nDLST(N).RUL(N) = 1. 1b: Asserts csi2_int_dl when CSI2nDLST(N).RUL(N) = 1.
16	EULE(N)	0h	RW	Enables interrupts generated by CSI2nDLST(N).EUL(N). 0b: Does not assert csi2_int_dl when CSI2nDLST(N).EUL(N) = 1. 1b: Asserts csi2_int_dl when CSI2nDLST(N).EUL(N) = 1.
15 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	EESE(N)	0h	RW	Enables interrupts generated by CSI2nDLST(N).EES(N). 0b: Does not assert csi2_int_dl when CSI2nDLST(N).EES(N) = 1. 1b: Asserts csi2_int_dl when CSI2nDLST(N).EES(N) = 1.
2	ECTE(N)	0h	RW	Enables interrupts generated by CSI2nDLST(N).ECT(N). 0b: Does not assert csi2_int_dl when CSI2nDLST(N).ECT(N) = 1. 1b: Asserts csi2_int_dl when CSI2nDLST(N).ECT(N) = 1.
1	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	ESHE(N)	0h	RW	Enables interrupts generated by CSI2nDLST(N).ESH(N). 0b: Does not assert csi2_int_dl when CSI2nDLST(N).ESH(N) = 1. 1b: Asserts csi2_int_dl when CSI2nDLST(N).ESH(N) = 1.

(17) Virtual Channel (M) Status Register (CRUm_CSI2nVCST(M)) (m = 0)

(M) indicates a virtual channel number.
(M) = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, or 15

Access Size : 32 bits
Address : <CRUm_base> + 0500h + (M) x 010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	LER (M)	LSR (M)	FER (M)	FSR (M)	-	-	-	-	-	-	-	OVF (M)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	FRD (M)	FRS (M)	-	ECN (M)	ECC (M)	WCE (M)	IDE (M)	CRC (M)	ECD (M)	MLF (M)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27	LER(M)	0h	R	When this bit is set to 1b, a packet whose data type is "Line End Code" has been received. Write 1 to CSI2nVCSC(M).LERC(M) to clear this bit.
26	LSR(M)	0h	R	When this bit is set to 1b, a packet whose data type is "Line Start Code" has been received. Write 1 to CSI2nVCSC(M).LSRC(M) to clear this bit.
25	FER(M)	0h	R	When this bit is set to 1b, a packet whose data type is "Frame End Code" has been received. Write 1 to CSI2nVCSC(M).FERC(M) to clear this bit.
24	FSR(M)	0h	R	When this bit is set to 1b, a packet whose data type is "Frame Start Code" has been received. Write 1 to CSI2nVCSC(M).FSRC(M) to clear this bit.
23 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	OVF(M)	0h	R	When this bit is set to 1b, the generic short packets on virtual channel M have been discarded due to an overflow of the generic short packet FIFO. Write 1 to CSI2nVCSC(M).OVFC(M) to clear this bit.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	FRD(M)	0h	R	When this bit is set to 1b, ErrFrameData has been detected. This bit is set to 1 if any of the following errors occurs during the period between FS and FE: • ErrCRC • ErrWC • ECC 2-bit error (when CSI2nMCT0.EDMD = 1) • Reception of a packet of less than four bytes (when CSI2nMCT0.EDMD = 1) Note that this bit is set to 1 if virtual channel = 0 between FS and FE because the virtual channel for the packet with an ECC 2-bit error or the virtual channel which has received a packet of less than four bytes cannot be determined. Write 1 to CSI2nVCSC(M).FRDC(M) to clear this bit.
8	FRS(M)	0h	R	When this bit is set to 1b, ErrFrameSync has been detected. This bit is set to 1 if any of the following conditions occurs: • After an FS was received, another FS has been received on the same virtual channel without receiving an FE. • Although an FE was received, an FS has not been received on the same virtual channel. Write 1 to CSI2nVCSC(M).FRSC(M) to clear this bit.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6	ECN(M)	0h	R	When this bit is set to 1b, no error was found in ECC checking on the received packets. Write 1 to CSI2nVCSC(M).ECNC(M) to clear this bit.
5	ECC(M)	0h	R	When this bit is set to 1b, a 1-bit error was found in ECC checking on the received packets and the error was corrected. Write 1 to CSI2nVCSC(M).ECCC(M) to clear this bit.

Bit	Bit Name	Initial Value	R/W	Description
4	WCE(M)	0h	R	When this bit is set to 1b, the data payload length of the packet being shorter than the value indicated by WC has been detected. Write 1 to CSI2nVCSC(M).WCEC(M) to clear this bit.
3	IDE(M)	0h	R	When this bit is set to 1b, an ErrID was detected and the packet was discarded. This bit is set to 1 if any of the following conditions occurs: • A packet which is defined as "Reserved" in "Chapter 9 Low Level Protocol" of the CSI-2 specification has been received. - A packet whose data type is 04h to 07h has been received. - A packet whose data type is 38h to 3Fh has been received when CSI2nMCT0.RVMD = 0. • A generic short packet (data type = 08h to 0Fh) for which the corresponding bit in CSI2nDTEL.DTEN is 0 has been received. • A long packet (data type = 10h to 3Fh) for which the corresponding bit in CSI2nDTEL(H).DTEN is 0 has been received. Write 1 to CSI2nVCSC(M).IDEC(M) to clear this bit. Note that this bit may also be set to 1 if more spacers than the count specified in CSI2nEPCT.SSP/SLP are received when EPD option 2 is in use.
2	CRC(M)	0h	R	When this bit is set to 1b, a CRC error was detected in the received packet. Write 1 to CSI2nVCSC(M).CRCC(M) to clear this bit.
1	ECD(M)	0h	R	When this bit is set to 1b, a 2-bit error was found in ECC checking on the received packets. Because the virtual channel for the packet with an error cannot be determined, the ECD0 field is set to 1 for all virtual channels. To clear the ECD0 field for all virtual channels, write 1 to CSI2nVCSC(M).AECDC(M) for any virtual channel. To clear the ECD0 field of virtual channel M, write 1 to CSI2nVCSC(M).ECDC(M).
0	MLF(M)	0h	R	When this bit is set to 1b, a packet of less than 4 bytes has been received. Because the virtual channel for the packet with an error cannot be determined, the MLF0 field is set to 1 for all virtual channels. To clear the MLF0 field for all virtual channels, write 1 to CSI2nVCSC(M).AMLFC(M) for any virtual channel. To clear the MLF0 field of virtual channel 0, write 1 to CSI2nVCSC(M).MLFC(M).

(18) Virtual Channel (M) Status Clear Register (CRUm_CSI2nVCSC(M)) (m = 0)

(M) indicates a virtual channel number.
(M) = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, or 15

Access Size : 32 bits
Address : <CRUm_base> + 0504h + (M) x 010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	LERC (M)	LSRC (M)	FERC (M)	FSRC (M)	-	-	-	-	-	-	-	OVFC (M)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AECDC (M)	AMLFC (M)	-	-	-	-	FRDC (M)	FRSC (M)	-	ECNC (M)	ECCC (M)	WCEC (M)	IDEC (M)	CRCC (M)	ECDC (M)	MLFC (M)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W1	R0W1	R	R	R	R	R0W1	R0W1	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27	LERC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).LER(M). Writing 0b to this bit has no effect.
26	LSRC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).LSR(M). Writing 0b to this bit has no effect.
25	FERC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).FER(M). Writing 0b to this bit has no effect.
24	FSRC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).FSR(M). Writing 0b to this bit has no effect.
23 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	OVFC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).OVF(M). Writing 0b to this bit has no effect.
15	AECDC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).ECD(M) for all virtual channels. Writing 0b to this bit has no effect.
14	AMLFC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).MLF(M) for all virtual channels. Writing 0b to this bit has no effect.
13 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	FRDC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).FRD(M). Writing 0b to this bit has no effect.
8	FRSC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).FRS(M). Writing 0b to this bit has no effect.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6	ECNC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).ECN(M). Writing 0b to this bit has no effect.
5	ECCC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).ECC(M). Writing 0b to this bit has no effect.

Bit	Bit Name	Initial Value	R/W	Description
4	WCEC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear the CSI2nVCST(M).WCE(M). Writing 0b to this bit has no effect.
3	IDEC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).IDE(M). Writing 0b to this bit has no effect.
2	CRCC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).CRC(M). Writing 0b to this bit has no effect.
1	ECDC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).ECD(M). Writing 0b to this bit has no effect.
0	MLFC(M)	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nVCST(M).MLF(M). Writing 0b to this bit has no effect.

(19) Virtual Channel (M) Interrupt Enable Register (CRUm_CSI2nVCIE(M)) (m = 0)

(M) indicates a virtual channel number.
(M) = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, or 15

Access Size : 32 bits
Address : <CRUm_base> + 0508h + (M) x 010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	LERE (M)	LSRE (M)	FERE (M)	FSRE (M)	-	-	-	-	-	-	-	OVFE (M)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	FRDE (M)	FRSE (M)	-	ECNE (M)	ECCE (M)	WCEE (M)	IDEE (M)	CRCE (M)	ECDE (M)	MLFE (M)
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27	LERE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).LER(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).LER(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).LER(M) = 1.
26	LSRE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).LSR(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).LSR(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).LSR(M) = 1.
25	FERE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).FER(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).FER(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).FER(M) = 1.
24	FSRE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).FSR(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).FSR(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).FSR(M) = 1.
23 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	OVFE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).OVF(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).OVF(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).OVF(M) = 1.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	FRDE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).FRD(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).FRD(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).FRD(M) = 1.
8	FRSE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).FRS(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).FRS(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).FRS(M) = 1.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6	ECNE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).ECN(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).ECN(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).ECN(M) = 1.
5	ECCE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).ECC(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).ECC(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).ECC(M) = 1.
4	WCEE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).WCE(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).WCE(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).WCE(M) = 1.
3	IDEE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).IDE(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).IDE(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).IDE(M) = 1.

Bit	Bit Name	Initial Value	R/W	Description
2	CRCE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).CRC(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).CRC(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).CRC(M) = 1.
1	ECDE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).ECD(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).ECD(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).ECD(M) = 1.
0	MLFE(M)	0h	RW	Enables interrupts generated by CSI2nVCST(M).MLF(M). 0b: Does not assert csi2_int_vc when CSI2nVCST(M).MLF(M) = 1. 1b: Asserts csi2_int_vc when CSI2nVCST(M).MLF(M) = 1.

(20) Power Management Status Register (CRUm_CSI2nPMST) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0600h
Initial Value : 000F_4000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	DLUL[3:0]				-	-	-	-	DLSS[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLUL	CLSS	-	-	-	-	-	-	CUN	CUX	DUN	DUX	CSN	CSX	DSN	DSX
Initial Value	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27 to 24	DLUL[3:0]	0h	R	Indicate the RxUlpsEsc status of a logical data lane. Bit 0: RxUlpsEsc of logical lane 0 Bit 1: RxUlpsEsc of logical lane 1 Bit 2: RxUlpsEsc of logical lane 2 Bit 3: RxUlpsEsc of logical lane 3
23 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 16	DLSS[3:0]	Fh	R	Indicate the stop state of a logical data lane. Bit 0: Stop state of logical lane 0 Bit 1: Stop state of logical lane 1 Bit 2: Stop state of logical lane 2 Bit 3: Stop state of logical lane 3
15	CLUL	0h	R	Indicates the inverted state of RxUlpsClkNot of the PPI clock lane. 0b: Not in ULPS 1b: In ULPS
14	CLSS	1h	R	Indicates the stop state of the PPI clock lane. 0b: Not in the stop state 1b: In the stop state
13 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	CUN	0h	R	When this bit is set to 1b, the clock lane has transitioned to ULPS mode (inverse of RxUlpsClkNot). Write 1 to CSI2nPMSC.CUNC to clear this bit.
6	CUX	0h	R	When this bit is set to 1b, the clock lane has transitioned from ULPS mode (inverse of RxUlpsClkNot) to non-ULPS mode. Write 1 to CSI2nPMSC.CUXC to clear this bit.
5	DUN	0h	R	When this bit is set to 1b, all valid data lanes have transitioned to ULPS mode. Write 1 to CSI2nPMSC.DUNC to clear this bit.
4	DUX	0h	R	When this bit is set to 1b, all valid data lanes have transitioned from ULPS mode to any other state. Write 1 to CSI2nPMSC.DUXC to clear this bit.
3	CSN	0h	R	When this bit is set to 1b, the clock lane has transitioned to the stop state. Write 1 to CSI2nPMSC.CSNC to clear this bit.
2	CSX	0h	R	When this bit is set to 1b, the clock lane has transitioned from the stop state to a non-stop state. Write 1 to CSI2nPMSC.CSXC to clear this bit.
1	DSN	0h	R	When this bit is set to 1b, all valid data lanes have transitioned to the stop state. Write 1 to CSI2nPMSC.DSNC to clear this bit.
0	DSX	0h	R	When this bit is set to 1b, all valid data lanes have transitioned from the stop state to any other state. Write 1 to CSI2nPMSC.DSXC to clear this bit.

(21) Power Management Status Clear Register (CRUm_CSI2nPMSC) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0604h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CUNC	CUXC	DUNC	DUXC	CSNC	CSXC	DSNC	DSXC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	CUNC	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nPMST.CUN. Writing 0b to this bit has no effect.
6	CUXC	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nPMST.CUX. Writing 0b to this bit has no effect.
5	DUNC	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nPMST.DUN. Writing 0b to this bit has no effect.
4	DUXC	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nPMST.DUX. Writing 0b to this bit has no effect.
3	CSNC	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nPMST.CSN. Writing 0b to this bit has no effect.
2	CSXC	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nPMST.CSX. Writing 0b to this bit has no effect.
1	DSNC	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nPMST.DSN. Writing 0b to this bit has no effect.
0	DSXC	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nPMST.DSX. Writing 0b to this bit has no effect.

(22) Power Management Interrupt Enable Register (CRUm_CSI2nPMIE) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0608h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CUNE	CUXE	DUNE	DUXE	CSNE	CSXE	DSNE	DSXE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	CUNE	0h	RW	Enables interrupts generated by CSI2nPMST.CUN. 0b: Does not assert csi2_int_pm when CSI2nPMST.CUN = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.CUN = 1.
6	CUXE	0h	RW	Enables interrupts generated by CSI2nPMST.CUX. 0b: Does not assert csi2_int_pm when CSI2nPMST.CUX = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.CUX = 1.
5	DUNE	0h	RW	Enables interrupts generated by CSI2nPMST.DUN. 0b: Does not assert csi2_int_pm when CSI2nPMST.DUN = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.DUN = 1.
4	DUXE	0h	RW	Enables interrupts generated by CSI2nPMST.DUX. 0b: Does not assert csi2_int_pm when CSI2nPMST.DUX = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.DUX = 1.
3	CSNE	0h	RW	Enables interrupts generated by CSI2nPMST.CSN. 0b: Does not assert csi2_int_pm when CSI2nPMST.CSN = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.CSN = 1.
2	CSXE	0h	RW	Enables interrupts generated by CSI2nPMST.CSX. 0b: Does not assert csi2_int_pm when CSI2nPMST.CSX = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.CSX = 1.
1	DSNE	0h	RW	Enables interrupts generated by CSI2nPMST.DSN. 0b: Does not assert csi2_int_pm when CSI2nPMST.DSN = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.DSN = 1.
0	DSXE	0h	RW	Enables interrupts generated by CSI2nPMST.DSX. 0b: Does not assert csi2_int_pm when CSI2nPMST.DSX = 1. 1b: Asserts csi2_int_pm when CSI2nPMST.DSX = 1.

(23) Generic Short Packet Control Register (CRUm_CSI2nGSCT) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0680h
Initial Value : 0001_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	GFIF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	SHTH[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	-	0h	R0W	The read value is always 0b. Do not change the value of this bit from 0h.
16	GFIF	1h	RW	When this bit is set to 1b, the received generic short packet is stored in the generic short packet FIFO (GS FIFO).
15 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 0	SHTH[6:0]	0h	RW	Asserts CSI2nGSST.GTH while the number of packets stored in the generic short packet FIFO equals or exceeds the value of this field + 1. Set this field to 15 or less when using CSI2nGSST.GTH. Set the default value when not using CSI2nGSST.GTH.

(24) Generic Short Packet Status Register (CRUm_CSI2nGSST) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0684h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STRDS	GCD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PNUM[7:0]								-	-	-	GOV	-	-	GTH	GNE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	STRDS	0h	R	Indicates whether the received generic short packets can be stored in the generic short packet FIFO. 0b: Can be stored in the generic short packet FIFO. 1b: Cannot be stored in the generic short packet FIFO. This field indicates an inverted value of the store_en signal described in **9.2.3.3.1 Data Reception** with a delay for an internal latency. The previous value may be read depending on the signal change timing.
16	GCD	0h	R	When this bit is set to 1b, the GS FIFO is being cleared by CSI2nGSCT.GFCLR.
15 to 8	PNUM[7:0]	0h	R	Indicate the number of packets stored in the generic short packet FIFO.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	GOV	0h	R	When this bit is set to 1b, the generic short packet FIFO has overflowed. Write 1 to CSI2nGSSC.GOVC to clear this bit.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	GTH	0h	R	When this bit is set to 1b, the number of packets stored in the generic short packet FIFO equals or exceeds "CSI2nGSCT.SHTH + 1".
0	GNE	0h	R	When this bit is set to 1b, the generic short packet FIFO is not empty.

(25) Generic Short Packet Status Clear Register (CRUm_CSI2nGSSC) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0688h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	GOVC	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W1	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	GOVC	0h	R0W1	The read value is always 0b. Write 1b to this bit to clear CSI2nGSST.GOV. Writing 0b to this bit has no effect.
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(26) Generic Short Packet Interrupt Enable Register (CRUm_CSI2nGSIE) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 068Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	GOVE	-	-	GTHE	GNEE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	GOVE	0h	RW	Enables interrupts generated by CSI2nGSST.GOV. 0b: Does not assert csi2_int_gst when CSI2nGSST.GOV = 1. 1b: Asserts csi2_int_gst when CSI2nGSST.GOV = 1.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	GTHE	0h	RW	Enables interrupts generated by CSI2nGSST.GTH. 0b: Does not assert csi2_int_gst when CSI2nGSST.GTH = 1. 1b: Asserts csi2_int_gst when CSI2nGSST.GTH = 1.
0	GNEE	0h	RW	Enables interrupts generated by CSI2nGSST.GNE. 0b: Does not assert csi2_int_gst when CSI2nGSST.GNE = 1. 1b: Asserts csi2_int_gst when CSI2nGSST.GNE = 1.

(27) Generic Short Packet Register (CRUm_CSI2nGSHT) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0690h
Initial Value : 0008_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	SPVC[3:0]				-	-	DTYP[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPDT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27 to 24	SPVC[3:0]	0h	R	Indicate the virtual channel.
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21 to 16	DTYP[5:0]	8h	R	Indicate the data type. 08h-0Fh
15 to 0	SPDT[15:0]	0h	R	Indicate 16-bit user-defined data.

(28) Generic Short Packet Information Update Register (CRUm_CSI2nGSIU) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0694h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	GFEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	GFCLR	-	-	-	-	-	-	-	FINC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	R0W1

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	GFEN	0h	R0W1	The read value is always 0b. When the GS FIFO has overflowed, write 1b to this bit to enable storing of generic short packets in the GS FIFO which has been disabled.
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	GFCLR	0h	RW	Clears the GS FIFO. Writing 1 to this bit requests clearing of the FIFO. Writing 0 to this bit requests clearing of the FIFO to be released.
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	FINC	0h	R0W1	The read value is always 0b. Write 1b to this bit to update data of the packet indicated in the CSI2nGSHT register for the next packet. Writing 0b to this bit has no effect. Writing 1b to this bit while CSI2nGSST.PNUM = 0b is prohibited.

(29) Slave Timing Control Register (CRUm_S_TIMCTL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 081Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S_HSSETTLECTL[7:0]								-	-	-	-	-	-	S_CLKSETTLECTL[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 8	S_HSSETTLECTL[7:0]	0h	RW	Slave Data Lane Control Register for THS-SETTLE. Set the values according to **Table 9.2-6**.
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	S_CLKSETTLECTL[1:0]	0h	RW	Do not change the value of this bit from 0h.

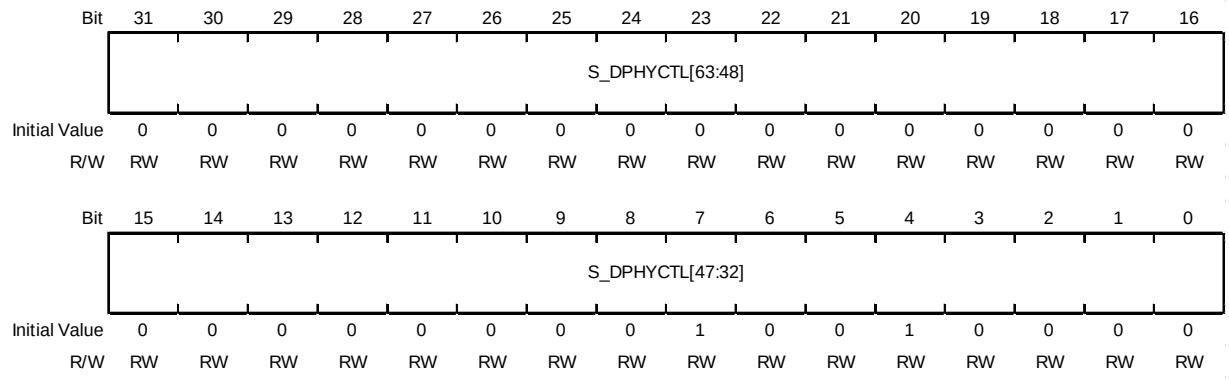
Table 9.2-6 S_TIMCTL.S_HSSETTLECTL[7:0] Register Settings

Transmission Rate TR (Mbps)	S_HSSETTLECTL[7:0]
2070 < TR ≤ 2100	2Eh
2030 < TR ≤ 2070	2Dh
1980 < TR ≤ 2030	2Ch
1940 < TR ≤ 1980	2Bh
1890 < TR ≤ 1940	2Ah
1850 < TR ≤ 1890	29h
1800 < TR ≤ 1850	28h
1760 < TR ≤ 1800	27h
1710 < TR ≤ 1760	26h
1670 < TR ≤ 1710	25h
1620 < TR ≤ 1670	24h
1580 < TR ≤ 1620	23h
1530 < TR ≤ 1580	22h
1490 < TR ≤ 1530	21h
1440 < TR ≤ 1490	20h
1400 < TR ≤ 1440	1Fh
1350 < TR ≤ 1400	1Eh
1310 < TR ≤ 1350	1Dh
1260 < TR ≤ 1310	1Ch
1220 < TR ≤ 1260	1Bh
1170 < TR ≤ 1220	1Ah
1120 < TR ≤ 1170	19h
1080 < TR ≤ 1120	18h

Transmission Rate TR (Mbps)	S_HSSETTLECTL[7:0]
1030 < TR ≤ 1080	17h
990 < TR ≤ 1030	16h
940 < TR ≤ 990	15h
900 < TR ≤ 940	14h
850 < TR ≤ 900	13h
810 < TR ≤ 850	12h
760 < TR ≤ 810	11h
720 < TR ≤ 760	10h
670 < TR ≤ 720	0Fh
630 < TR ≤ 670	0Eh
580 < TR ≤ 630	0Dh
540 < TR ≤ 580	0Ch
490 < TR ≤ 540	0Bh
450 < TR ≤ 490	0Ah
400 < TR ≤ 450	09h
360 < TR ≤ 400	08h
310 < TR ≤ 360	07h
270 < TR ≤ 310	06h
220 < TR ≤ 270	05h
180 < TR ≤ 220	04h
130 < TR ≤ 180	03h
90 < TR ≤ 130	02h
80 ≤ TR ≤ 90	01h

(30) Slave D-PHY Control Register (MSB) (CRUm_S_DPHYCTL_MSB) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0834h
Initial Value : 0000_0090h



Bit	Bit Name	Initial Value	R/W	Description
31 to 2	S_DPHYCTL [63:34]	24h	RW	Changing this bit from 24h to a different value is prohibited.
1	S_DPHYCTL [33]	0h	RW	DeSkew Calibration Function Enable/Disable Control • 0b: DeSkew Calibration Disable • 1b: DeSkew Calibration Enable
0	S_DPHYCTL [32]	0h	RW	Changing this bit from 0h to a different value is prohibited.

(31) DP/DN Swap Control Register (CRUm_SWAPCTL) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0838h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	S_DPDN_SWAP_DAT	S_DPDN_SWAP_CLK	-	-	M_DPDN_SWAP_DAT	M_DPDN_SWAP_CLK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	S_DPDN_SWAP_DAT	0h	RW	Slave PPI signal This signal is for crossing PCB trace lines of DP and DN when it connects the Slave lane to a Master lane on the PCB board. This signal swaps all the DP and DN signals of Slave when set to HIGH. Tie this signal to the ground when the connections are correct.
4	S_DPDN_SWAP_CLK	0h	RW	Slave PPI signal This signal is for crossing PCB trace lines of DP and DN when it connects the Slave lane to a Master lane on the PCB board. This signal swaps all the DP and DN signals of Slave when set to HIGH. Tie this signal to the ground when the connections are correct.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	M_DPDN_SWAP_DAT	0h	RW	Do not change the value of this bit from 0h.
0	M_DPDN_SWAP_CLK	0h	RW	Do not change the value of this bit from 0h.

(32) RX Skew Calibration Results Value n (CRUm_S_ATB_OUT_n) (m = 0, n = 0 to 3)

Access Size : 32 bits

Address : <CRUm_base> + 086Ch + n × 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	S_ATB_OUT_n	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read The written value will be ignored.
1	S_ATB_OUT_n	0h	R	The initial/periodic deskew calibration status for lane n (for debugging) 0b: Calibration in progress or calibration failed 1b: Calibration completed
0	-	0h	R	Reserved Whenever it is read, 0b is read The written value will be ignored.

9.2.3.3 Operation (Short Packet Reception)

The prefix “CRUm” of the register names is omitted in the following subsections.

9.2.3.3.1 Data Reception

(1) Common between short packets and long packets

Figure 9.2-3 shows the flow when receiving a packet whose data type is A and virtual channel is M (0, 1, 2, to 15).

This flow is common between short packets and long packets.

Upon receiving a packet, the LINK checks the length of the received packet.

If the length of the received packet is less than the packet header length (4 bytes), a virtual channel cannot be determined, resulting in setting CSI2nVCST(M).MLF(M) to 1b for all virtual channels. In addition, this packet is discarded because it is invalid.

When the length of the received packet is 4 bytes or more, ECC checking is performed. The check results are indicated in the CSI2nVCST(M) register.

- 2-bit error: Because the virtual channel of the packet with an error cannot be determined, CSI2nVCST(M).ECD(M) is set to 1b for all virtual channels.
  In addition, the packet is discarded because error correction cannot be performed.
- 1-bit error: CSI2nVCST(M).ECC(M) is set to 1b for the relevant virtual channel.
  Error correction proceeds because it is possible.
  No error: CSI2nVCST(M).ECN(M) is set to 1b for the relevant virtual channel.

If the ECC check results are “1-bit error” or “no error”, the data type field of the packet header is analyzed.

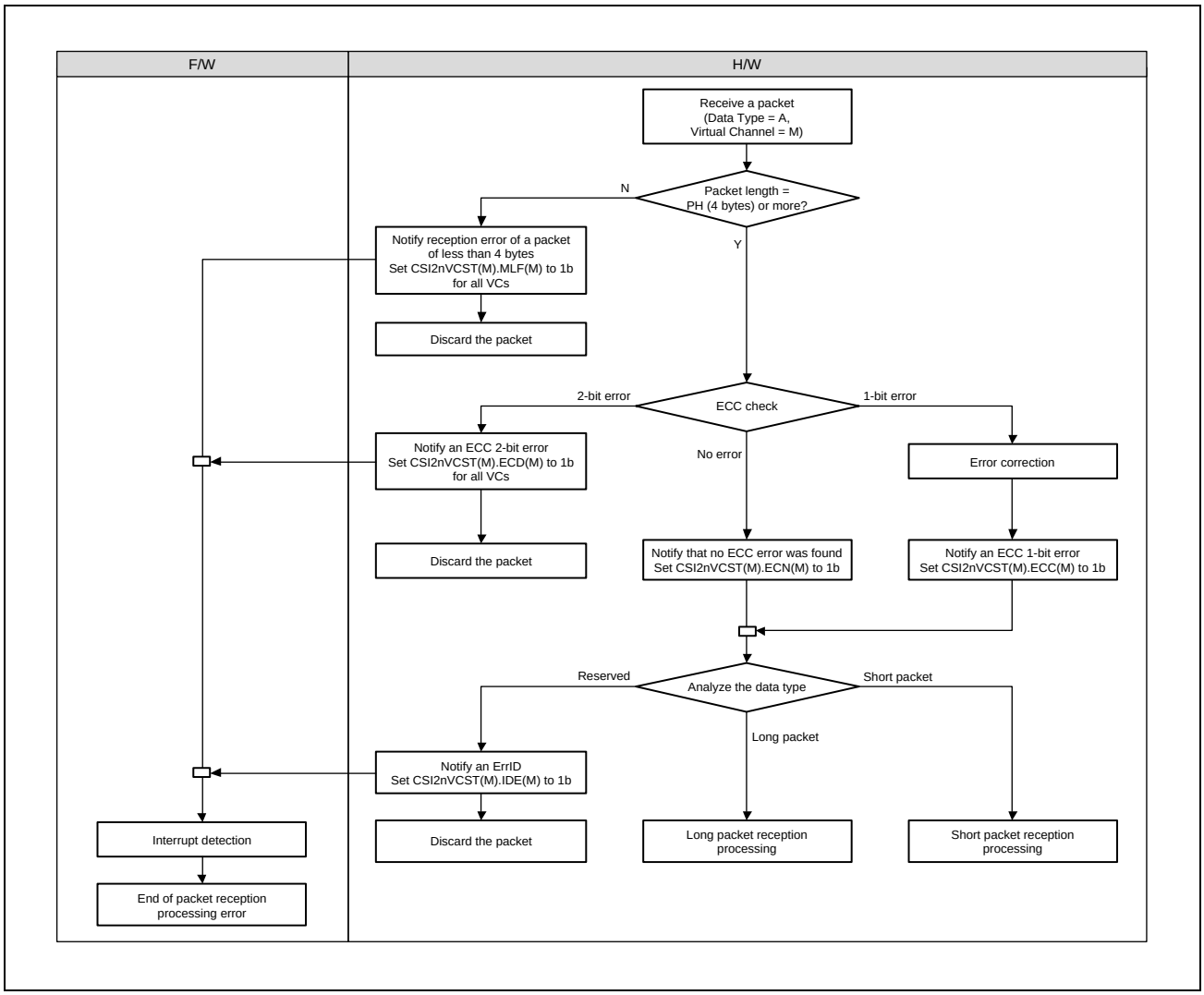


Figure 9.2-3 Packet Reception Processing

(2) Generic short packet data type code

Figure 9.2-4 shows the flow when receiving a packet whose data type is generic short packet code 1 to 8 and virtual channel is M.

When the corresponding bit in CSI2nDTEL.DTEN is set to 0b, CSI2nVCST(M).IDE(M)) is asserted and the packet is discarded.

When the corresponding bit in CSI2nDTEL.DTEN is set to 1b and CSI2nGSCT.GFIF is set to 1b, the generic short packet is output to the generic short packet FIFO. For details on the operation when using the generic short packet FIFO, see the flow in **Figure 9.2-4**.

The initial value of internal signal store_en which is described in the flow is 1. When store_en is set to 1, packets can be stored in the generic short packet FIFO. When it is set to 0, packets cannot be stored in the generic short packet FIFO.

- store_en is set to 0 when the generic short packet FIFO has overflowed.
- store_en is set to 1 when 1b is written to CSI2nGSIU.GFEN.

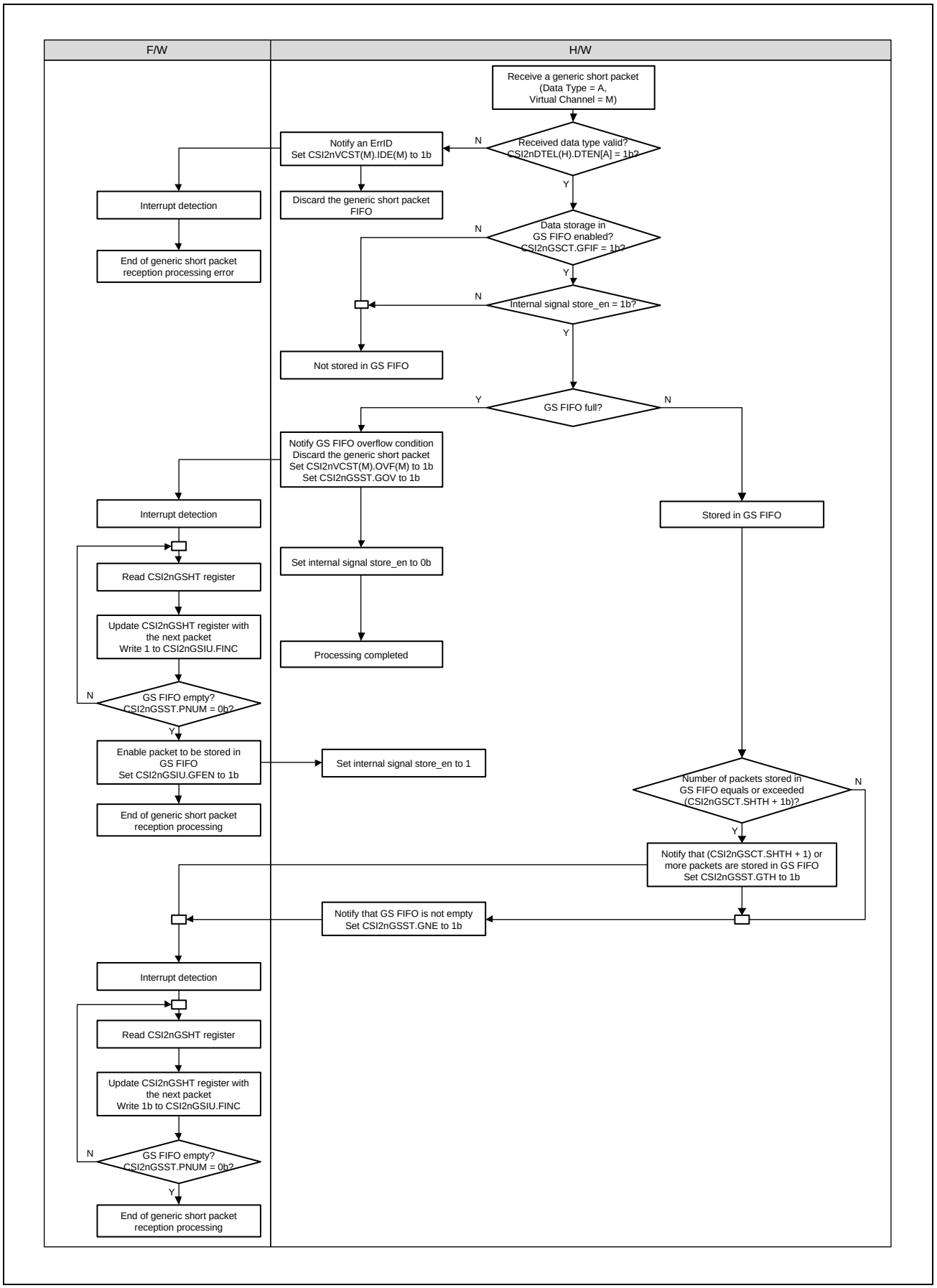


Figure 9.2-4 Generic Short Packet Reception Processing

(3) Clearing the generic short packet FIFO

Figure 9.2-5 shows how the generic short packet FIFO is cleared.

Writing 1b to CSI2nGSIU.GFCLR starts clearing the generic short packet FIFO.

After requesting to clear the generic short packet FIFO, check if CSI2nGSST.GCD is set to 1b to make sure that the FIFO is cleared. Then, writing 0b to CSI2nGSIU.GFCLR cancels the request for clearing the generic short packet FIFO. Check if CSI2nGSST.GCD is set to 0b to make sure that the request for clearing the generic short packet FIFO is canceled.

When CSI2nMCG.GSNM is set to 0b, it is prohibited to clear the generic short packet FIFO.

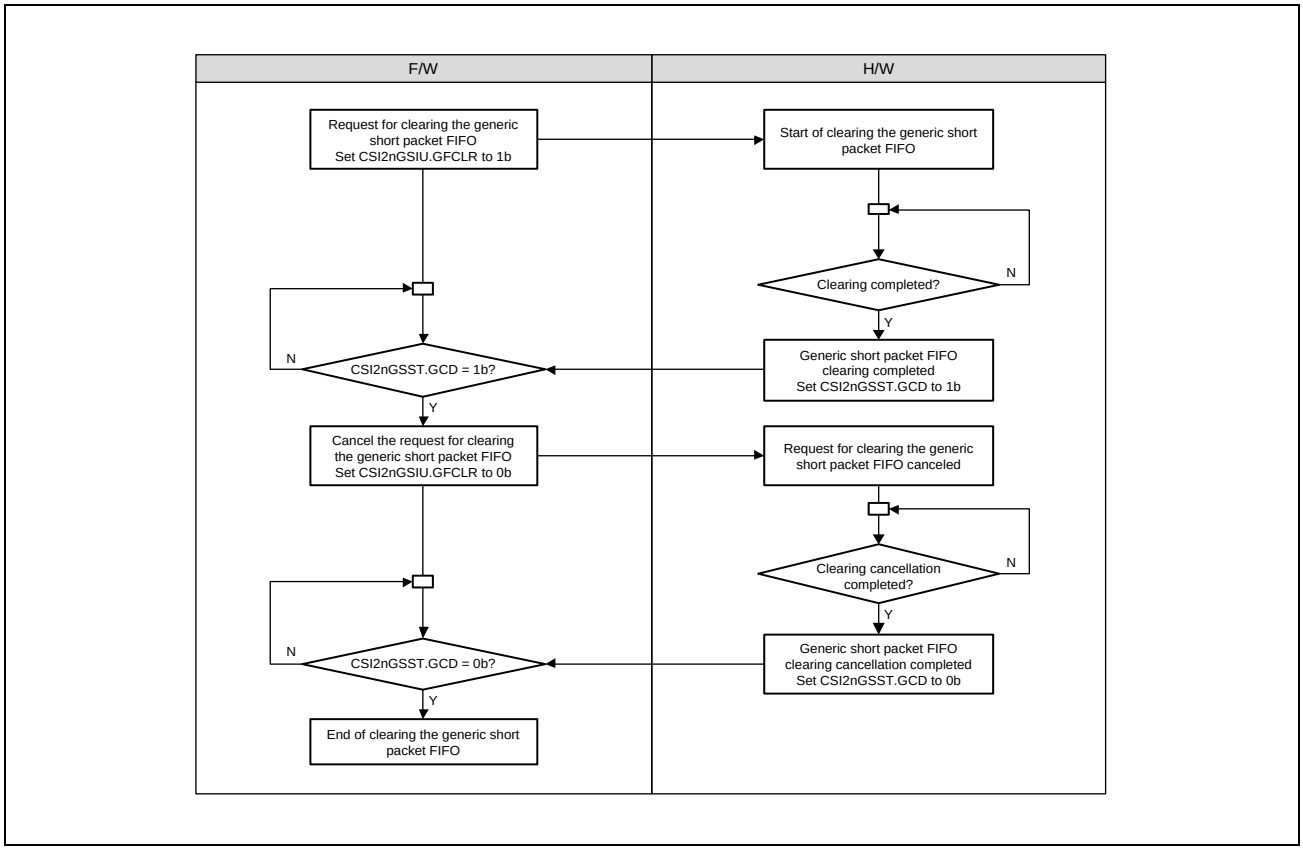


Figure 9.2-5 Clearing the Generic Short Packet FIFO

9.2.3.4 Interrupts

Figure 9.2-6 shows the structure of interrupt signals of MIPI CSI2. CRU0 has an interrupt. The conditions for interrupt generation, interrupt source registers, and interrupt enable registers are listed in **Table 9.2-7**, **Table 9.2-8**, and **Table 9.2-9**. The CSI2nMIST register indicates the interrupt output state. When an interrupt is generated, read the CSI2nMIST register to identify the interrupt source.

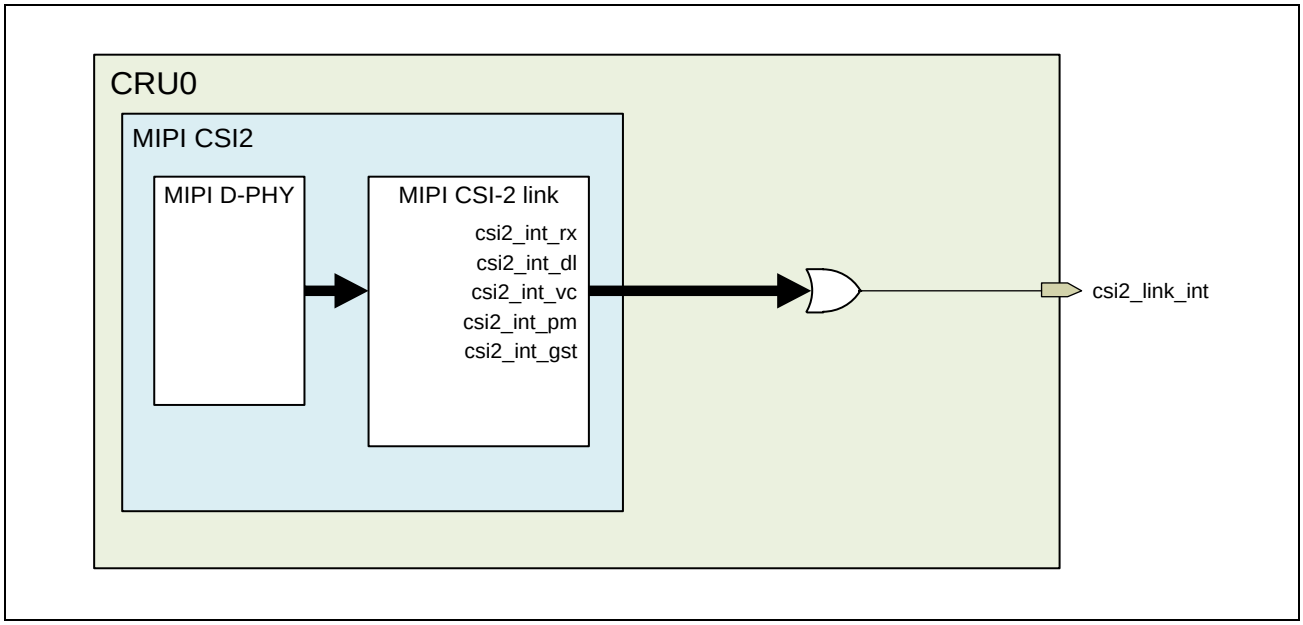


Figure 9.2-6 Interrupt Structure

Table 9.2-7 Interrupt States and Generation Conditions (1)

Interrupt State	Interrupt Source Register Field	Interrupt Enable Register Field	Generation Condition
CSI2nMIST.RXS	CSI2nRXST.RACTDET	CSI2nRXIE.RACTDETE	This IP received a packet.
CSI2nMIST.DLS0	CSI2nDLST0.ESH0	CSI2nDLIE0.ESHE0	ErrSotHS occurred on DL0.
	CSI2nDLST0.ECT0	CSI2nDLIE0.ECTE0	ErrControl occurred on DL0.
	CSI2nDLST0.EES0	CSI2nDLIE0.EESE0	ErrESC occurred on DL0.
	CSI2nDLST0.EUL0	CSI2nDLIE0.EULE0	DL0 exited from the ULPS.
	CSI2nDLST0.RUL0	CSI2nDLIE0.RULE0	DL0 transitioned to the ULPS.
CSI2nMIST.DLS1	CSI2nDLST1.ESH1	CSI2nDLIE1.ESHE1	ErrSotHS occurred on DL1.
	CSI2nDLST1.ECT1	CSI2nDLIE1.ECTE1	ErrControl occurred on DL1.
	CSI2nDLST1.EES1	CSI2nDLIE1.EESE1	ErrESC occurred on DL1.
	CSI2nDLST1.EUL1	CSI2nDLIE1.EULE1	DL1 exited from the ULPS.
	CSI2nDLST1.RUL1	CSI2nDLIE1.RULE1	DL1 transitioned to the ULPS.
CSI2nMIST.DLS2	CSI2nDLST2.ESH2	CSI2nDLIE2.ESHE2	ErrSotHS occurred on DL2.
	CSI2nDLST2.ECT2	CSI2nDLIE2.ECTE2	ErrControl occurred on DL2.
	CSI2nDLST2.EES2	CSI2nDLIE2.EESE2	ErrESC occurred on DL2.
	CSI2nDLST2.EUL2	CSI2nDLIE2.EULE2	DL2 exited from the ULPS.
	CSI2nDLST2.RUL2	CSI2nDLIE2.RULE2	DL2 transitioned to the ULPS.
CSI2nMIST.DLS3	CSI2nDLST3.ESH3	CSI2nDLIE3.ESHE3	ErrSotHS occurred on DL3.
	CSI2nDLST3.ECT3	CSI2nDLIE3.ECTE3	ErrControl occurred on DL3.
	CSI2nDLST3.EES3	CSI2nDLIE3.EESE3	ErrESC occurred on DL3.
	CSI2nDLST3.EUL3	CSI2nDLIE3.EULE3	DL3 exited from the ULPS.
	CSI2nDLST3.RUL3	CSI2nDLIE3.RULE3	DL3 transitioned to the ULPS.

Table 9.2-8 Interrupt States and Generation Conditions (2)

Interrupt State	Interrupt Source Register Field	Interrupt Enable Register Field	Generation Condition
CSI2nMIST.VCS0	CSI2nVCST0.MLF0	CSI2nVCIE0.MLFE0	Malformed packet was received on VC0.
	CSI2nVCST0.ECD0	CSI2nVCIE0.ECDE0	ErrEccDouble occurred on VC0.
	CSI2nVCST0.CRC0	CSI2nVCIE0.CRCE0	ErrCrc occurred on VC0.
	CSI2nVCST0.IDE0	CSI2nVCIE0.IDEE0	ErrID occurred on VC0.
	CSI2nVCST0.WCE0	CSI2nVCIE0.WCEE0	ErrWC occurred on VC0.
	CSI2nVCST0.ECC0	CSI2nVCIE0.ECCE0	ErrEccCorrected occurred on VC0.
	CSI2nVCST0.ECN0	CSI2nVCIE0.ECNE0	ErrEccNoError occurred on VC0.
	CSI2nVCST0.FRS0	CSI2nVCIE0.FRSE0	ErrFrameSync occurred on VC0.
	CSI2nVCST0.FRD0	CSI2nVCIE0.FRDE0	ErrFrameData occurred on VC0.
	CSI2nVCST0.OVF0	CSI2nVCIE0.OVFE0	Generic short packets on VC0 were discarded because GS FIFO overflowed.
	CSI2nVCST0.FSR0	CSI2nVCIE0.FSRE0	Frame start was received on VC0.
	CSI2nVCST0.FER0	CSI2nVCIE0.FERE0	Frame end was received on VC0.
	CSI2nVCST0.LSR0	CSI2nVCIE0.LSRE0	Line start was received on VC0.
	CSI2nVCST0.LER0	CSI2nVCIE0.LERE0	Line end was received on VC0.
CSI2nMIST.VCS1	CSI2nVCST1.MLF1	CSI2nVCIE1.MLFE1	Malformed packet was received on VC1.
	CSI2nVCST1.ECD1	CSI2nVCIE1.ECDE1	ErrEccDouble occurred on VC1.
	CSI2nVCST1.CRC1	CSI2nVCIE1.CRCE1	ErrCrc occurred on VC1.
	CSI2nVCST1.IDE1	CSI2nVCIE1.IDEE1	ErrID occurred on VC1.
	CSI2nVCST1.WCE1	CSI2nVCIE1.WCEE1	ErrWC occurred on VC1.
	CSI2nVCST1.ECC1	CSI2nVCIE1.ECCE1	ErrEccCorrected occurred on VC1.
	CSI2nVCST1.ECN1	CSI2nVCIE1.ECNE1	ErrEccNoError occurred on VC1.
	CSI2nVCST1.FRS1	CSI2nVCIE1.FRSE1	ErrFrameSync occurred on VC1.
	CSI2nVCST1.FRD1	CSI2nVCIE1.FRDE1	ErrFrameData occurred on VC1.
	CSI2nVCST1.OVF1	CSI2nVCIE1.OVFE1	Generic short packets on VC1 were discarded because GS FIFO overflowed.
	CSI2nVCST1.FSR1	CSI2nVCIE1.FSRE1	Frame start was received on VC1.
	CSI2nVCST1.FER1	CSI2nVCIE1.FERE1	Frame end was received on VC1.
	CSI2nVCST1.LSR1	CSI2nVCIE1.LSRE1	Line start was received on VC1.
	CSI2nVCST1.LER1	CSI2nVCIE1.LERE1	Line end was received on VC1.
…	…	…	…

Table 9.2-9 Interrupt States and Generation Conditions (3)

Interrupt State	Interrupt Source Register Field	Interrupt Enable Register Field	Generation Condition
CSI2nMIST.VCS15	CSI2nVCST15.MLF15	CSI2nVCIE15.MLFE15	Malformed packet was received on VC15.
	CSI2nVCST15ECD15	CSI2nVCIE15.ECDE15	ErrEccDouble occurred on VC15.
	CSI2nVCST15.CRC15	CSI2nVCIE15.CRCE15	ErrCrc occurred on VC15.
	CSI2nVCST15.IDE15	CSI2nVCIE15.IDEE15	ErrID occurred on VC15.
	CSI2nVCST15.WCE15	CSI2nVCIE15.WCEE15	ErrWC occurred on VC15.
	CSI2nVCST15.ECC15	CSI2nVCIE15.ECCE15	ErrEccCorrected occurred on VC15.
	CSI2nVCST15.ECN15	CSI2nVCIE15.ECNE15	ErrEccNoError occurred on VC15.
	CSI2nVCST15.FRS15	CSI2nVCIE15.FRSE15	ErrFrameSync occurred on VC15.
	CSI2nVCST15.FRD15	CSI2nVCIE15.FRDE15	ErrFrameData occurred on VC15.
	CSI2nVCST15.OVF15	CSI2nVCIE15.OVFE15	Generic short packets on VC15 were discarded because GS FIFO overflowed.
	CSI2nVCST15.FSR15	CSI2nVCIE15.FSRE15	Frame start was received on VC15.
	CSI2nVCST15.FER15	CSI2nVCIE15.FERE15	Frame end was received on VC15.
	CSI2nVCST15.LSR15	CSI2nVCIE15.LSRE15	Line start was received on VC15.
	CSI2nVCST15.LER15	CSI2nVCIE15.LERE15	Line end was received on VC15.
CSI2nMIST.PMS	CSI2nPMST.DSX	CSI2nPMIE.DSXE	All data lanes exited from the stop state.
	CSI2nPMST.DSN	CSI2nPMIE.DSNE	All data lanes transitioned to the stop state.
	CSI2nPMST.CSX	CSI2nPMIE.CSXE	The clock lane exited from the stop state.
	CSI2nPMST.CSN	CSI2nPMIE.CSNE	The clock lane transitioned to the stop state.
	CSI2nPMST.DUX	CSI2nPMIE.DUXE	All data lanes exited from the ULPS.
	CSI2nPMST.DUN	CSI2nPMIE.DUNE	All data lanes transitioned to the ULPS.
	CSI2nPMST.CUX	CSI2nPMIE.CUXE	The clock lane exited from the ULPS.
	CSI2nPMST.CUN	CSI2nPMIE.CUNE	The clock lane transitioned to the ULPS.
CSI2nMIST.GSTS	CSI2nGSST.GNE	CSI2nGSIE.GNEE	GS FIFO is not empty.
	CSI2nGSST.GTH	CSI2nGSIE.GTHE	Data stored in GS FIFO equals or exceeds the threshold.
	CSI2nGSST.GOV	CSI2nGSIE.GOVE	GS FIFO overflowed.

9.2.4 Image Converter

9.2.4.1 Overview

The image converter module is a data conversion module which handles image clipping, frame subsampling, demosaicing (only selectable for SVC0, and only for RAW input), linear matrix, statistics (only selectable for SVC0, and only for RAW input), color space conversion, LUT conversion, pixel format conversion, and so on. It has an MIPI CSI2 (**9.2.3 MIPI CSI2**) input as an image sensor interface and its processing is roughly divided into two.

- Video system (Image Process, Non Image Process, All Bypass): Data are initially stored in FIFO-VD and transferred to external memory through the internal AXI-VD bus.
- Statistics system: Data are initially stored in FIFO-SD and transferred to external memory through the internal AXI-SD bus.

The video system further has image processing, non-image processing and full bypass systems. Full bypass is used when images are not to be processed by the image converter. Image processing and non-image processing can be used simultaneously (however, the data type cannot be set simultaneously within a single VC). The image processing system handles image processing and the non-image processing system does not handle image processing. Therefore, the data type for image processing, such as RGB, YUV, or RAW, can be selected for image processing, while several data types which are not for image processing, such as embedded 8-bit non-image data and a user-defined 8-bit data type, can be selected for non-image processing. In full bypass (CRUnCTRL.ATH = 1b), image processing within the CRU is not used and data from the MIPI CSI-2 link (in the recommended memory storage format for MIPI CSI-2) are directly transferred to external memory (image processing and non-image processing data cannot be used with full bypass).

The statistics system only handles data processing to produce statistics.

For each MIPI CSI-2 input, up to four channels (SVC0 to 3) can be selected from among 16 virtual channels (VC0 to 15) and only one data type for image processing is processed per channel and several data types can be selected for non-image processing.

Additionally, when only SVC0 is used, some parameters are adjustable from frame to frame, and when demosaicing processing for multiple virtual channels (the image sizes, etc. are the same) is to proceed, control is applied with switching on frame-by-frame demarking by changing a register setting. However, this function cannot be used for sensors that have different VC for each line.

9.2.4.1.1 Features

The operating range of the image converter is as follows.

Image processing

- Range of effective pixels: QVGA (320 × 240) to 8.8 Mpixels
- Range of effective pixels in the horizontal direction: 320 to 4096 pixels
- Range of effective pixels in the vertical direction: 240 to 4096 lines
- Non-image processing and full bypass:
  The size range of application-specific payloads within input packets is from 1 to 24576 bytes.

Common

The maximum transfer rate at the time of MIPI CSI-2 input differs with the input format. For details, see **9.2.3.1.1 Features**.

- Input data format (type): See **9.2.4.3.1 Input Formats and Image Processing**.
  - Image processing:
    Only one data type from among YUV, RGB, RAW, etc. is selectable for each SVC.
  - Non-image processing:
    Several data types such as embedded 8-bit non-image data and user-defined 8-bit data are selectable for each SVC.

- Video output data format (type): Little endian is only supported.
  - 8-bit multiplexing of YCbCr422/YCbCr420 (with support for switching the order of Y and U/V)
  - YC separation of YCbCr422 and YCbCr420 (separation into Y and CbCr components)
  - Extraction of the Y component of YCbCr422
  - U and V in YUV422/YCbCr420 outputs are expressed in offset binary or as two’s complements.
  - RGB-888 (24 bits/pixel)
  - RGB-888 (32 bits/pixel)
  - ARGB-8888 (32 bits/pixel)
  - RAW6, 7, 8, 10, 12, 14, 16, 20
  - MIPI CSI-2 recommended memory storage format

- Demosaicing
  - RAW6, 7, 8, 10, 12, 14, 16, 20 to RGB888

- Color correction by linear matrix operations
  - Offset calculation included

- Color space conversion
  - YCbCr422 to RGB888
  - RGB888 to YCbCr422

- Image format conversion
  - YCbCr422 → YC separation (separation into Y and CbCr components)
  - YCbCr422 → Y component extraction
  - YCbCr422 → YCbCr420
  - RGB-888 → RGB-888 (32 bits/pixel)
  - RGB-888 → ARGB-8888 (32 bits/pixel) B first and A last format
  - RGB-888 → ARGB-8888 (32 bits/pixel) A first and B last format
  - RGB-565 → RGB-888 conversion (intermediate processing)
  - RGB-666 → RGB-888 conversion (intermediate processing)
  - RGB-555 → RGB-888 conversion (intermediate processing)
  - RGB-444 → RGB-888 conversion (intermediate processing)
  - RAW6, 7 → RAW8 conversion (intermediate processing)
  - RAW20: 16 higher-order bits are used for processing (demosaic, statis_cal) (intermediate processing)

- 4 channels (SVC0 to 3) from among virtual channels 0 to 15 can be selected for input (however, only SVC0 is usable for demosaicing and statistics)
- Conversion by a lookup table (LUT): 10-bit to 8-bit conversion
- Data clipping: Suppressing the level of YCbCr data. For details see **9.2.4.3.10 Data Clipping**.
- Image clipping
- Frame subsampling
- Bypassing of each function (not performing the function): Several data types selectable

- Pattern generator (for debugging)
  - Y, U, and V register values are output.

- Statistics (statistical processing operations: horizontal detection area: 320 to 4096 pixels; vertical detection area: 240 to 4096)
  - Accumulation in the (16×16, 32×32, 64×64, and 128×128) areas for each of R, G, and B from RAW6, 7, 8, 10, 12, 14, 16, and 20 data
  - Summation of absolute values for differences from adjacent pixel values of G from RAW6, 7, 8, 10, 12, 14, 16, and 20 data

- FIFO
  - Video system: 16 Kbytes
  - Statistics system: 2 Kbytes

- AXI master: 2 channels (video system, statistics system)

- Memory bank address settings
  - Support for 8 bank memory bank addresses (video system)
    - SVC offset address
    - Non-image processing offset address
    - Offset address when separation into Y and UV is selected
  - Support for 8 memory bank addresses (statistics system: SVC0)

- Memory addressing
  - Video: Stored with image striding
  - Statistics: Stored with consecutive writing line

- Appending of a header (data type, VC, word count) to the start of a line: turning this off or on is possible.
- Appending of a footer (256’d1/256’d0)

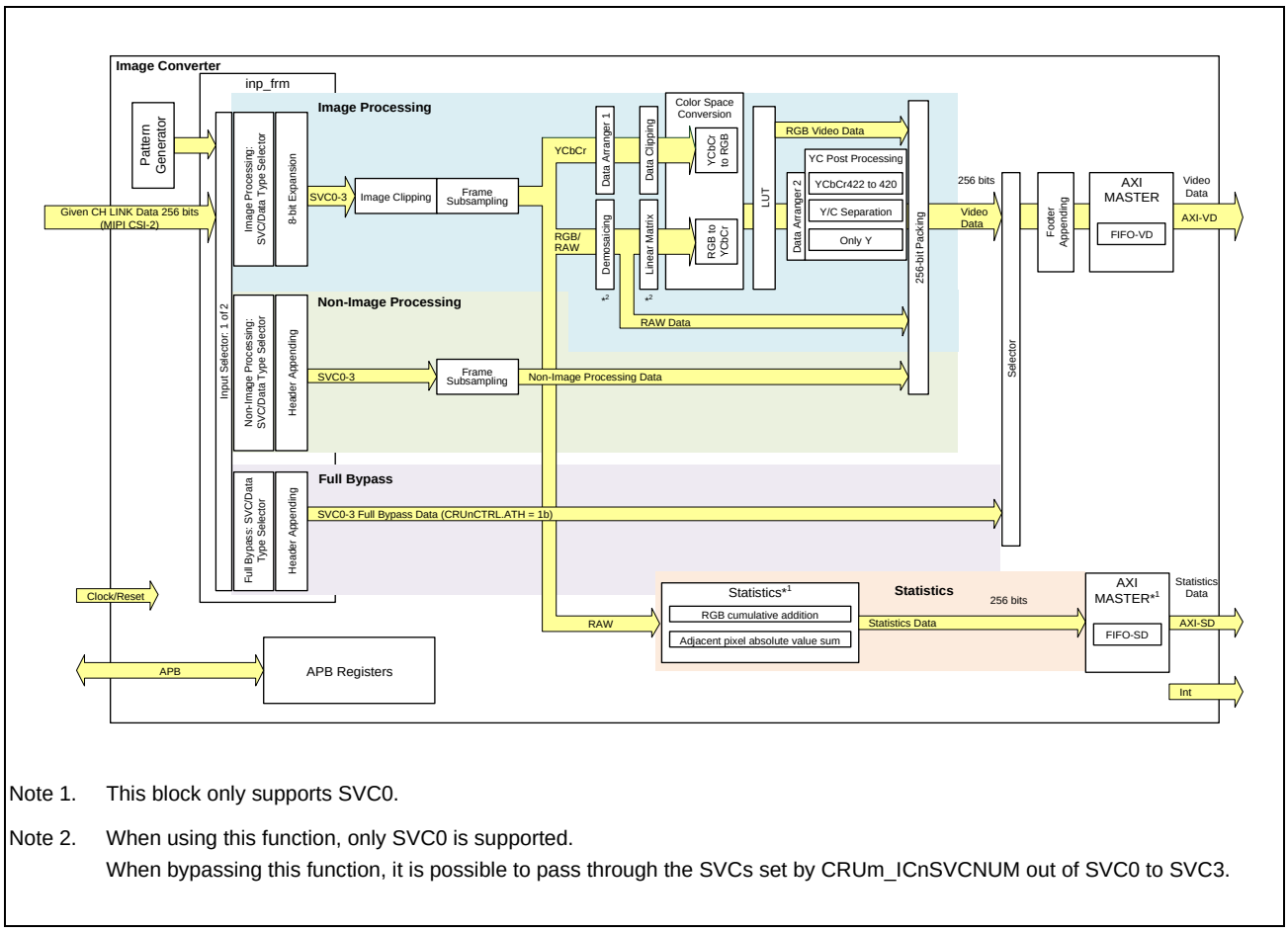
  A footer value (256’d1/256’d0) is inverted when writing to the same bank so that the CPU can judge how far data have been written.
  - Appending of a footer to the end of each line (video system)
  - Appending of a footer to the end of each frame (statistics system)

- APB slave: Only 32-bit access supported

- Interrupt sources
  - Scan line interrupt: 4 sources
  - Start-of-frame/end-of-frame interrupt: 8 sources
  - FIFO overflow interrupt (video system, statistics system): 2 sources
  - AXI SLVERR and DECERR error interrupts (video system, statistics system): 4 sources
  - AXI specified address transfer completed interrupt (video system: 5 addresses; statistics system: 2 addresses): 7 sources

9.2.4.1.2 Block Diagram

Figure 9.2-7 is a block diagram of the image converter.



Note 1. This block only supports SVC0.

Note 2. When using this function, only SVC0 is supported.
When bypassing this function, it is possible to pass through the SVCs set by CRUm_ICnSVCNUM out of SVC0 to SVC3.

Figure 9.2-7 Block Diagram of Image Converter

Pattern Generator

This function is used for debugging and generates image data for testing. The data types are limited to YCbCr422.

Input Selector (1 of 2)

A register setting selects data for input to the image converter from among the following three sources.

- Image data for testing
- Link data from this CRU

Video

- Image Processing
  - SVC/Data Type Selector
    A register setting selects one channel from among 16 VCs (MIPI CSI-2 virtual channels) for use by the image converter. Four channels are selectable and these are treated as SVC0 to 3.
    A register setting selects one data type for use in image processing. Four data types are selectable.

- 8-bit Expansion
  Input data less than 8 bits ((RGB444/RGB555/RGB565/RGB666, RAW6/7) are expanded to 8 bits (RGB888, RAW8).
- Image Clipping
  Input images are cropped to the size selected by register settings for the output of small images.
- Frame Subsampling
  Subsamples input image frames to lower the frame rate. This is used to reduce the bandwidth on the bus.
- Data Arrangers 1 and 2
  Converts the negative expression of the UV component of YUV (YCbC) data.
- Data Clipping
  Limits data values of YUV (YCbCr) to upper and lower limits. This is used when standards having upper and lower limits such as ITU-R BT.601 and 702 are applicable.
- Demosaicing
  Interpolates converted RGB data from RAW (Bayer-array) data.
- Linear Matrix
  Applies matrix operations to correct the colors of RGB data converted by demosaicing.
- Color Space Conversion
  Converts the color space from YCbCr to RGB and vice versa.
- LUT (Lookup Table)
  Converts 10-bit data after color space conversion (YCbCr to RGB or RGB to YCbCr) to a table of 8-bit entries. A basic bit rounding function is set, but the table can also be used for non-linear image correction such as gamma correction.
- YC Post Processing
  - YCbCr422 to 420
    Subsamples YCbCr422 data to convert it into YCbCr422.
  - Y/C Separation
    Sorts and separates YCbCr data into Y and C components.
  - Only Y
    Deletes the C components of YCbCr data so that only Y data remain.
- 256-bit Packing
  Packs data into 256-bit units to suit the bus width for transmission through the AXI.

- Non-Image Processing
  - SVC/Data Type Selector
    The same channels as those set as SVC0 to 3 described under “Image Processing” are used.
    A register setting selects several data types for use in non-image processing.
  - Header Appending
    Since non-image processing and full bypass involve the selection of several data types and storage of the data in a mixed state in the same SVC memory, a header is appended to the start of a line of non-image processing or full bypass data so that the data types and data lengths can be identified when read.
- Full Bypass
  - VC/Data Type Selector
    The same channels as those set as SVC0 to 3 described under “Image Processing” are used.
    A register setting selects several data types for use in full bypass.

- Footer Appending
  Appends 256’d0/d1 as a footer, which is inverted every time the memory bank is overwritten, so that the CPU can recognize the bank number to which the memory data have been written.
- AXI MASTER (AXI-VD)
  Video data are initially stored in a FIFO buffer and then written to the DRAM by the AXI master.

Statistics

- Statistics
  Acquires statistics for each block which is divided from a display for 3A (AE/AWB/AF).
- Footer Appending
  Appends 512 bits as a footer, which is inverted every time the memory bank is overwritten, so that the CPU can recognize the bank number to which the memory data have been written.
  The footer value is written alternately: (1) higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256’h1 and (2) higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256’h0.
- AXI MASTER (AXI-SD)
  Statistics data are initially stored in a FIFO buffer and then written to the DRAM by the AXI master.

APB Registers

Only 32-bit access is supported.

9.2.4.2 IMC Registers

See **9.2.3.2 MIPI CSI2 Registers** for the CRUm base addresses.

9.2.4.2.1 List of Registers [m = 0]

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
CRU Control Register	CRUm_CRUnCTRL	0000_0000h	0000h	32
CRU Interrupt Enable Register1	CRUm_CRUnIE1	0000_0000h	0004h	32
CRU Interrupt Enable Register2	CRUm_CRUnIE2	0000_0000h	0008h	32
CRU Interrupt Status Register1	CRUm_CRUnINTS1	0000_0000h	000Ch	32
CRU Interrupt Status Register2	CRUm_CRUnINTS2	0000_0000h	0010h	32
Reserve	-	-	0014h to 0017h	-
CRU Reset Register	CRUm_CRUnRST	0000_0000h	0018h	32
Reserve	-	-	001Ch to 002Bh	-
CRU General Read/Write Register	CRUm_CRUnCOM	0000_0000h	002Ch	32
Reserve	-	-	0030h to 003Fh	-
Memory Bank 1 Base Address Lower Register for Video Data	CRUm_AMnMB1ADDRL	0000_0000h	0040h	32
Memory Bank 1 Base Address Higher Register for Video Data	CRUm_AMnMB1ADDRH	0000_0000h	0044h	32
Memory Bank 2 Base Address Lower Register for Video Data	CRUm_AMnMB2ADDRL	0000_0000h	0048h	32
Memory Bank 2 Base Address Higher Register for Video Data	CRUm_AMnMB2ADDRH	0000_0000h	004Ch	32
Memory Bank 3 Base Address Lower Register for Video Data	CRUm_AMnMB3ADDRL	0000_0000h	0050h	32
Memory Bank 3 Base Address Higher Register for Video Data	CRUm_AMnMB3ADDRH	0000_0000h	0054h	32
Memory Bank 4 Base Address Lower Register for Video Data	CRUm_AMnMB4ADDRL	0000_0000h	0058h	32
Memory Bank 4 Base Address Higher Register for Video Data	CRUm_AMnMB4ADDRH	0000_0000h	005Ch	32
Memory Bank 5 Base Address Lower Register for Video Data	CRUm_AMnMB5ADDRL	0000_0000h	0060h	32
Memory Bank 5 Base Address Higher Register for Video Data	CRUm_AMnMB5ADDRH	0000_0000h	0064h	32
Memory Bank 6 Base Address Lower Register for Video Data	CRUm_AMnMB6ADDRL	0000_0000h	0068h	32
Memory Bank 6 Base Address Higher Register for Video Data	CRUm_AMnMB6ADDRH	0000_0000h	006Ch	32
Memory Bank 7 Base Address Lower Register for Video Data	CRUm_AMnMB7ADDRL	0000_0000h	0070h	32
Memory Bank 7 Base Address Higher Register for Video Data	CRUm_AMnMB7ADDRH	0000_0000h	0074h	32
Memory Bank 8 Base Address Lower Register for Video Data	CRUm_AMnMB8ADDRL	0000_0000h	0078h	32
Memory Bank 8 Base Address Higher Register for Video Data	CRUm_AMnMB8ADDRH	0000_0000h	007Ch	32
UV Data Address Offset Lower Register for Video Data	CRUm_AMnUVAOFL	0000_0000h	0080h	32
UV Data Address Offset Higher Register for Video Data	CRUm_AMnUVAOFH	0000_0000h	0084h	32
Memory Bank Enable Register for Video Data	CRUm_AMnMBVALID	0000_0001h	0088h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Memory Address Lower Status Register for Video Data	CRUm_AMnMADRSL	0000_0000h	008Ch	32
Memory Address Higher Status Register for Video Data	CRUm_AMnMADRSH	0000_0000h	0090h	32
Non Image Process Address Offset Lower Register	CRUm_AMnNIAOFL	0000_0000h	0094h	32
Non Image Process Address Offset Higher Register	CRUm_AMnNIAOFH	0000_0000h	0098h	32
SVC Data Address Offset Lower Register for Video Data	CRUm_AMnSVCAOFL	0000_0000h	009Ch	32
SVC Data Address Offset Higher Register for Video Data	CRUm_AMnSVCAOFH	0000_0000h	00A0h	32
Reserve	-	-	00A4h to 00B3h	-
AXI-VD Bus Transfer Completion Event Address 0 Lower Register	CRUm_AMnIVT0ADDRL	0000_0000h	00B4h	32
AXI-VD Bus Transfer Completion Event Address 0 Higher Register	CRUm_AMnIVT0ADDRH	0000_0000h	00B8h	32
AXI-VD Bus Transfer Completion Event Address 1 Lower Register	CRUm_AMnIVT1ADDRL	0000_0000h	00BCh	32
AXI-VD Bus Transfer Completion Event Address 1 Higher Register	CRUm_AMnIVT1ADDRH	0000_0000h	00C0h	32
AXI-VD Bus Transfer Completion Event Address 2 Lower Register	CRUm_AMnIVT2ADDRL	0000_0000h	00C4h	32
AXI-VD Bus Transfer Completion Event Address 2 Higher Register	CRUm_AMnIVT2ADDRH	0000_0000h	00C8h	32
AXI-VD Bus Transfer Completion Event Address 3 Lower Register	CRUm_AMnIVT3ADDRL	0000_0000h	00CCh	32
AXI-VD Bus Transfer Completion Event Address 3 Higher Register	CRUm_AMnIVT3ADDRH	0000_0000h	00D0h	32
AXI-VD Bus Transfer Completion Event Address 4 Lower Register	CRUm_AMnIVT4ADDRL	0000_0000h	00D4h	32
AXI-VD Bus Transfer Completion Event Address 4 Higher Register	CRUm_AMnIVT4ADDRH	0000_0000h	00D8h	32
Reserve	-	-	00DCh to 00EBh	-
AXI-VD Bus Master Transfer Setting Register	CRUm_AMnAXIATTR	0020_0150h	00ECh	32
AXI-VD Bus Master FIFO Setting Register	CRUm_AMnFIFO	0000_0001h	00F0h	32
AXI-VD Bus Master Transfer Resume Register	CRUm_AMnFIFOTRST	0000_0000h	00F4h	32
AXI-VD Bus Master FIFO Pointer Register	CRUm_AMnFIFOPNTR	0000_0000h	00F8h	32
Reserve	-	-	00FCh to 010Fh	-
AXI-VD Bus Master Transfer Stop Register	CRUm_AMnAXISTP	0000_0000h	0110h	32
AXI-VD Bus Master Transfer Stop Status Register	CRUm_AMnAXISTPACK	0000_0000h	0114h	32
Reserve	-	-	0118h to 0127h	-
Image Stride Setting Register	CRUm_AMnIS	0000_0000h	0128h	32
Reserve	-	-	012Ch to 013Bh	-
Memory Bank 1 Base Address Lower Register for CRU Statistics Data	CRUm_AMnSDMB1ADDRL	0000_0000h	013Ch	32
Memory Bank 1 Base Address Higher Register for CRU Statistics Data	CRUm_AMnSDMB1ADDRH	0000_0000h	0140h	32
Memory Bank 2 Base Address Lower Register for CRU Statistics Data	CRUm_AMnSDMB2ADDRL	0000_0000h	0144h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Memory Bank 2 Base Address Higher Register for CRU Statistics Data	CRUm_AMnSDMB2ADDRH	0000_0000h	0148h	32
Memory Bank 3 Base Address Lower Register for CRU Statistics Data	CRUm_AMnSDMB3ADDRL	0000_0000h	014Ch	32
Memory Bank 3 Base Address Higher Register for CRU Statistics Data	CRUm_AMnSDMB3ADDRH	0000_0000h	0150h	32
Memory Bank 4 Base Address Lower Register for CRU Statistics Data	CRUm_AMnSDMB4ADDRL	0000_0000h	0154h	32
Memory Bank 4 Base Address Higher Register for CRU Statistics Data	CRUm_AMnSDMB4ADDRH	0000_0000h	0158h	32
Memory Bank 5 Base Address Lower Register for CRU Statistics Data	CRUm_AMnSDMB5ADDRL	0000_0000h	015Ch	32
Memory Bank 5 Base Address Higher Register for CRU Statistics Data	CRUm_AMnSDMB5ADDRH	0000_0000h	0160h	32
Memory Bank 6 Base Address Lower Register for CRU Statistics Data	CRUm_AMnSDMB6ADDRL	0000_0000h	0164h	32
Memory Bank 6 Base Address Higher Register for CRU Statistics Data	CRUm_AMnSDMB6ADDRH	0000_0000h	0168h	32
Memory Bank 7 Base Address Lower Register for CRU Statistics Data	CRUm_AMnSDMB7ADDRL	0000_0000h	016Ch	32
Memory Bank 7 Base Address Higher Register for CRU Statistics Data	CRUm_AMnSDMB7ADDRH	0000_0000h	0170h	32
Memory Bank 8 Base Address Lower Register for CRU Statistics Data	CRUm_AMnSDMB8ADDRL	0000_0000h	0174h	32
Memory Bank 8 Base Address Higher Register for CRU Statistics Data	CRUm_AMnSDMB8ADDRH	0000_0000h	0178h	32
Reserve	-	-	017Ch to 018Bh	-
Memory Bank Enable Register for CRU Statistics Data	CRUm_AMnSDMBVALID	0000_0001h	018Ch	32
Memory Bank Status Register for CRU Statistics Data	CRUm_AMnSDMBS	0000_0000h	0190h	32
AXI-SD Bus Transfer Completion Event Address 0 Lower Register	CRUm_AMnSDIVT0ADDRL	0000_0000h	0194h	32
AXI-SD Bus Transfer Completion Event Address 0 Higher Register	CRUm_AMnSDIVT0ADDRH	0000_0000h	0198h	32
AXI-SD Bus Transfer Completion Event Address 1 Lower Register	CRUm_AMnSDIVT1ADDRL	0000_0000h	019Ch	32
AXI-SD Bus Transfer Completion Event Address 1 Higher Register	CRUm_AMnSDIVT1ADDRH	0000_0000h	01A0h	32
Reserve	-	-	01A4h to 01BFh	-
AXI-SD Bus Master FIFO Pointer Register	CRUm_AMnSDFIFOPNTR	0000_0000h	01C0h	32
Reserve	-	-	01C4h to 01D7h	-
AXI-SD Bus Master Transfer Stop Register	CRUm_AMnSDAXISTP	0000_0000h	01D8h	32
AXI-SD Bus Master Transfer Stop Status Register	CRUm_AMnSDAXISTPACK	0000_0000h	01DCh	32
Reserve	-	-	01E0h to 01EFh	-
CRU Image Converter Enable Register	CRUm_ICnEN	0000_0000h	01F0h	32
CRU Data Value Processing	CRUm_ICnDTVP	0000_0003h	01F4h	32
CRU SVC Number Register	CRUm_ICnSVCNUM	0000_0000h	01F8h	32
CRU VC Select Register	CRUm_ICnSVC	0000_3210h	01FCh	32
CRU Image Converter Main Control Register (SVC0)	CRUm_ICnIPMC_C0	001E_00FEh	0200h	32
CRU Non Image Process Data Type Code Select Lower Register (SVC0)	CRUm_ICnNIPDT_C0L	0000_0000h	0204h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
CRU Non Image Process Data Type Code Select Higher Register (SVC0)	CRUm_ICnNIPDT_C0H	0000_0000h	0208h	32
CRU Image Clipping Start Line Register (SVC0)	CRUm_ICnSLPrC_C0	0000_0000h	020Ch	32
CRU Image Clipping End Line Register (SVC0)	CRUm_ICnELPrC_C0	0000_0000h	0210h	32
CRU Image Clipping Start Pixel Register (SVC0)	CRUm_ICnSPPrC_C0	0000_0000h	0214h	32
CRU Image Clipping End Pixel Register (SVC0)	CRUm_ICnEPPrC_C0	0000_0000h	0218h	32
CRU Scan Line Interrupt Register (SVC0)	CRUm_ICnSI_C0	0000_0000h	021Ch	32
CRU Linear Matrix offset Register (SVC0 only)	CRUm_ICnLMXOF	0000_0000h	0220h	32
CRU Linear Matrix R coefficient 1 Register (SVC0 only)	CRUm_ICnLMXRC1	0000_0400h	0224h	32
CRU Linear Matrix R coefficient 2 Register (SVC0 only)	CRUm_ICnLMXRC2	0000_0000h	0228h	32
CRU Linear Matrix G coefficient 1 Register (SVC0 only)	CRUm_ICnLMXGC1	0000_0000h	022Ch	32
CRU Linear Matrix G coefficient 2 Register (SVC0 only)	CRUm_ICnLMXGC2	0000_0400h	0230h	32
CRU Linear Matrix B coefficient 1 Register (SVC0 only)	CRUm_ICnLMXBC1	0000_0000h	0234h	32
CRU Linear Matrix B coefficient 2 Register (SVC0 only)	CRUm_ICnLMXBC2	0400_0000h	0238h	32
CRU Statistics Control 1 Register (SVC0 only)	CRUm_ICnSTIC1	0000_0000h	023Ch	32
Reserve	-	-	0240h to 0243h	-
CRU Image Converter Register Setting Change Control Register	CRUm_ICnREGC	0000_0000h	0244h	32
Reserve	-	-	0248h to 0257h	-
CRU Image Converter Main Control Register (SVC1)	CRUm_ICnIPMC_C1	001E_0066h	0258h	32
CRU Image Converter Main Control Register (SVC2)	CRUm_ICnIPMC_C2	001E_0066h	025Ch	32
CRU Image Converter Main Control Register (SVC3)	CRUm_ICnIPMC_C3	001E_0066h	0260h	32
CRU Non Image Process Data Type Code Select Lower Register (SVC1)	CRUm_ICnNIPDT_C1L	0000_0000h	0264h	32
CRU Non Image Process Data Type Code Select Higher Register (SVC1)	CRUm_ICnNIPDT_C1H	0000_0000h	0268h	32
CRU Non Image Process Data Type Code Select Lower Register (SVC2)	CRUm_ICnNIPDT_C2L	0000_0000h	026Ch	32
CRU Non Image Process Data Type Code Select Higher Register (SVC2)	CRUm_ICnNIPDT_C2H	0000_0000h	0270h	32
CRU Non Image Process Data Type Code Select Lower Register (SVC3)	CRUm_ICnNIPDT_C3L	0000_0000h	0274h	32
CRU Non Image Process Data Type Code Select Higher Register (SVC3)	CRUm_ICnNIPDT_C3H	0000_0000h	0278h	32
Reserve	-	-	027Ch to 0288h	-
CRU Image Clipping Start Line Register (SVC1)	CRUm_ICnSLPrC_C1	0000_0000h	028Ch	32
CRU Image Clipping End Line Register (SVC1)	CRUm_ICnELPrC_C1	0000_0000h	0290h	32
CRU Image Clipping Start Pixel Register (SVC1)	CRUm_ICnSPPrC_C1	0000_0000h	0294h	32
CRU Image Clipping End Pixel Register (SVC1)	CRUm_ICnEPPrC_C1	0000_0000h	0298h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
CRU Scan Line Interrupt Register (SVC1)	CRUm_ICnSI_C1	0000_0000h	029Ch	32
CRU Image Clipping Start Line Register (SVC2)	CRUm_ICnSLPrC_C2	0000_0000h	02A0h	32
CRU Image Clipping End Line Register (SVC2)	CRUm_ICnELPrC_C2	0000_0000h	02A4h	32
CRU Image Clipping Start Pixel Register (SVC2)	CRUm_ICnSPPrC_C2	0000_0000h	02A8h	32
CRU Image Clipping End Pixel Register (SVC2)	CRUm_ICnEPPrC_C2	0000_0000h	02ACh	32
CRU Scan Line Interrupt Register (SVC2)	CRUm_ICnSI_C2	0000_0000h	02B0h	32
CRU Image Clipping Start Line Register (SVC3)	CRUm_ICnSLPrC_C3	0000_0000h	02B4h	32
CRU Image Clipping End Line Register (SVC3)	CRUm_ICnELPrC_C3	0000_0000h	02B8h	32
CRU Image Clipping Start Pixel Register (SVC3)	CRUm_ICnSPPrC_C3	0000_0000h	02BCh	32
CRU Image Clipping End Pixel Register (SVC3)	CRUm_ICnEPPrC_C3	0000_0000h	02C0h	32
CRU Scan Line Interrupt Register (SVC3)	CRUm_ICnSI_C3	0000_0000h	02C4h	32
Reserve	-	-	02C8h to 02D7h	-
CRU Module Status Register	CRUm_ICnMS	0000_0000h	02D8h	32
CRU Frame Subsampling Control Register	CRUm_ICnDEC	0000_0000h	02DCh	32
CRU Line Count Register (SVC0)	CRUm_ICnLC_C0	0000_0000h	02E0h	32
CRU Line Count Register (SVC1)	CRUm_ICnLC_C1	0000_0000h	02E4h	32
CRU Line Count Register (SVC2)	CRUm_ICnLC_C2	0000_0000h	02E8h	32
CRU Line Count Register (SVC3)	CRUm_ICnLC_C3	0000_0000h	02ECh	32
Reserve	-	-	02F0h to 02FFh	-
CRU Frame Subsampling Interrupt Control Register	CRUm_ICnINTCTRL	0000_0000h	0300h	32
CRU Output Image Format Register	CRUm_ICnDMR	0000_0000h	0304h	32
CRU YCbCr → RGB Color Space Conversion Coefficient 1 Register	CRUm_ICnCSCC1	0000_129Fh	0308h	32
CRU YCbCr → RGB Color Space Conversion Coefficient 2 Register	CRUm_ICnCSCC2	0100_0800h	030Ch	32
CRU YCbCr → RGB Color Space Conversion Coefficient 3 Register	CRUm_ICnCSCC3	1989_0D02h	0310h	32
CRU YCbCr → RGB Color Space Conversion Coefficient 4 Register	CRUm_ICnCSCC4	0645_2045h	0314h	32
CRU RGB → YCbCr Color Space Conversion Y Coefficient 1 Register	CRUm_ICnYCCR1	0000_0107h	0318h	32
CRU RGB → YCbCr Color Space Conversion Y Coefficient 2 Register	CRUm_ICnYCCR2	0064_0204h	031Ch	32
CRU RGB → YCbCr Color Space Conversion Y Coefficient 3 Register	CRUm_ICnYCCR3	0A00_0100h	0320h	32
CRU RGB → YCbCr Color Space Conversion Cb Coefficient 1 Register	CRUm_ICnCBCCR1	0000_1F68h	0324h	32
CRU RGB → YCbCr Color Space Conversion Cb Coefficient 2 Register	CRUm_ICnCBCCR2	01C2_1ED6h	0328h	32
CRU RGB → YCbCr Color Space Conversion Cb Coefficient 3 Register	CRUm_ICnCBCCR3	0A00_0800h	032Ch	32
CRU RGB → YCbCr Color Space Conversion Cr Coefficient 1 Register	CRUm_ICnCRCCR1	0000_01C2h	0330h	32
CRU RGB → YCbCr Color Space Conversion Cr Coefficient 2 Register	CRUm_ICnCRCCR2	1FB7_1E87h	0334h	32
CRU RGB → YCbCr Color Space Conversion Cr Coefficient 3 Register	CRUm_ICnCRCCR3	0A00_0800h	0338h	32
Reserve	-	-	033Ch to 034Bh	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
CRU Lookup Table Control Register	CRUm_ICnLUT	0000_0000h	034Ch	32
CRU Lookup Table Status Register	CRUm_ICnLUTS	0000_0000h	0350h	32
CRU Lookup Table Pointer Register	CRUm_ICnLUTP	0000_0000h	0354h	32
CRU Lookup Table Data Register	CRUm_ICnLUTD	0000_0000h	0358h	32
CRU Test Image Generation Control 1 Register	CRUm_ICnTICTRL1	0000_0000h	035Ch	32
CRU Test Image Generation Control 2 Register	CRUm_ICnTICTRL2	0000_0000h	0360h	32
CRU Test Image Size Setting 1 Register	CRUm_ICnTISIZE1	0000_0000h	0364h	32
CRU Test Image Size Setting 2 Register	CRUm_ICnTISIZE2	0000_0000h	0368h	32

9.2.4.2.2 Register Descriptions

(1) Bypass Setting for Each Function

Figure 9.2-8 shows where the bypass settings by the ICnIPMC_C0/1/2/3.*THR and CRUnCTRL.ATH registers work.

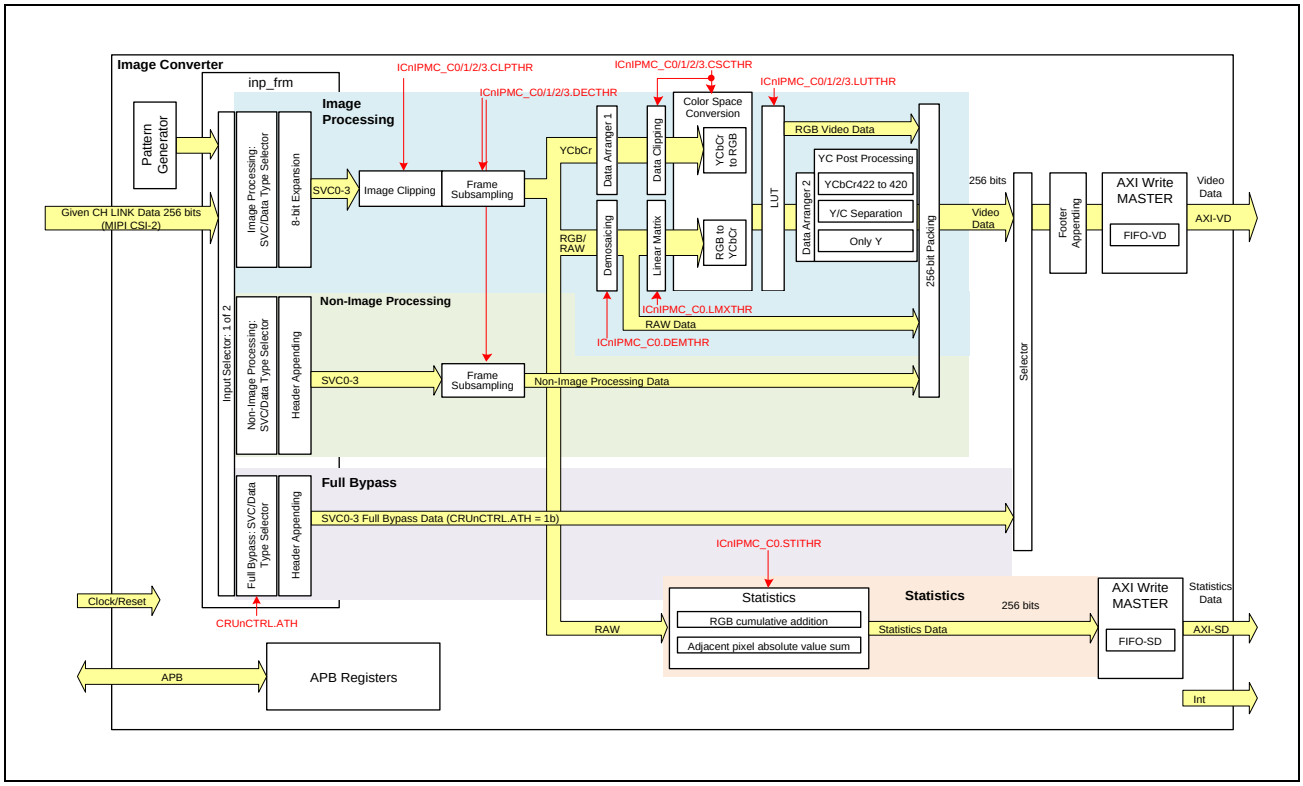


Figure 9.2-8 Bypass Setting for Each Function

(2) CRU Control Register (CRUm_CRUnCTRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ATH	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	ATH	0h	RW	Full bypass setting for the image converter 0b: Normal 1b: Full bypass is performed and the data from the LINK is output as is to the AXI-VD (the virtual channels and data types set by registers ICnSVC and ICnNIPDT_C0~3L/H are bypassed). Note: See **Figure 9.2-8**.
0	-	0h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.

(3) CRU Interrupt Enable Register1 (CRUm_CRUnIE1) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	SDDECEE	SDSLVEE	SDFOE	-	-	-	-	-	DECEE	SLVEE	FOE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.
30 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11	-	0h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.
10	SDDECEE	0h	RW	AXI-SD DECERR (statistics data-related) interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated. Note: DECERR occurs when the space where the forwarding address is not allowed is accessed. If an error occurs, check that the forwarding address is correct.
9	SDSLVEE	0h	RW	AXI-SD SLVERR (statistics data-related) interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated. Note: SLVERR occurs by the response from the other party as the destination for transfer from this bus. If this happens, check the destination settings.
8	SDFOE	0h	RW	FIFO-SD overflow (statistics data-related) interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	-	0h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.
2	DECEE	0h	RW	AXI-VD DECERR (video data-related) interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated. Note: DECERR occurs when the space where the forwarding address is not allowed is accessed. If an error occurs, check that the forwarding address is correct.
1	SLVEE	0h	RW	AXI-VD SLVERR (video data-related) interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated. Note: SLVERR occurs by the response from the other party as the destination for transfer from this bus. If this happens, check the destination settings.
0	FOE	0h	RW	FIFO-VD overflow (video data-related) interrupt enable control (for debugging) 0b: No interrupts are generated. 1b: Interrupts are generated.

(4) CRU Interrupt Enable Register2 (CRUm_CRUnIE2) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	SDADR1WE	SDADR0WE	-	-	-	VDADR4WE	VDADR3WE	VDADR2WE	VDADR1WE	VDADR0WE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	R	R	R	R	RW	RW	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	SL3E	FE3E	FS3E	SL2E	FE2E	FS2E	SL1E	FE1E	FS1E	SL0E	FE0E	FS0E
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.
30	-	0h	RW	Do not change the value of this bit from 0.
29 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25	SDADR1WE	0h	RW	AXI-SD bus write completion interrupt enable control (video data-related) for designated address 1 (35-bit memory address set by AMnSDIVT1ADDRL and AMnSDIVT1ADDRH) 0b: No interrupts are generated. 1b: Interrupts are generated.
24	SDADR0WE	0h	RW	AXI-SD bus write completion interrupt enable control(statistics data-related) for designated address 0 (35-bit memory address set by AMnSDIVT0ADDRL and AMnSDIVT0ADDRH) 0b: No interrupts are generated. 1b: Interrupts are generated.
23 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	VDADR4WE	0h	RW	AXI-VD bus write completion interrupt enable control (video data-related) for designated address 4 (35-bit memory address set by AMnIVT4ADDRL and AMnIVT4ADDRH) 0b: No interrupts are generated. 1b: Interrupts are generated.
19	VDADR3WE	0h	RW	AXI-VD bus write completion interrupt enable control (video data-related) for designated address 3 (35-bit memory address set by AMnIVT3ADDRL and AMnIVT3ADDRH) 0b: No interrupts are generated. 1b: Interrupts are generated.
18	VDADR2WE	0h	RW	AXI-VD bus write completion interrupt enable control (video data-related) for designated address 2 (35-bit memory address set by AMnIVT2ADDRL and AMnIVT2ADDRH) 0b: No interrupts are generated. 1b: Interrupts are generated.
17	VDADR1WE	0h	RW	AXI-VD bus write completion interrupt enable control (video data-related) for designated address 1 (35-bit memory address set by AMnIVT1ADDRL and AMnIVT1ADDRH) 0b: No interrupts are generated. 1b: Interrupts are generated.
16	VDADR0WE	0h	RW	AXI-VD bus write completion interrupt enable control (video data-related) for designated address 0 (35-bit memory address set by AMnIVT0ADDRL and AMnIVT0ADDRH) 0b: No interrupts are generated. 1b: Interrupts are generated.
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11	SL3E	0h	RW	SVC3 Scan line interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.
10	FE3E	0h	RW	SVC3 Frame end interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.

Bit	Bit Name	Initial Value	R/W	Description
9	FS3E	0h	RW	SVC3 Frame start interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.
8	SL2E	0h	RW	SVC2 Scan line interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.
7	FE2E	0h	RW	SVC2 Frame end interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.
6	FS2E	0h	RW	SVC2 Frame start interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.
5	SL1E	0h	RW	SVC1 Scan line interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.
4	FE1E	0h	RW	SVC1 Frame end interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.
3	FS1E	0h	RW	SVC1 Frame start interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.
2	SL0E	0h	RW	SVC0 Scan line interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.
1	FE0E	0h	RW	SVC0 Frame end interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.
0	FS0E	0h	RW	SVC0 Frame start interrupt enable control 0b: No interrupts are generated. 1b: Interrupts are generated.

(5) CRU Interrupt Status Register1 (CRUm_CRUnINTS1) (m = 0)

See **Figure 9.2-7** for where each bit of the interrupt is detected.

Access Size : 32 bits
Address : <CRUm_base> + 000Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	SDDECES	SDSLVES	SDFOS	-	-	-	-	-	DECES	SLVES	FOS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW1	RW1	RW1	R	R	R	R	RW	RW1	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 12	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value will be ignored.
10	SDDECES	0h	RW1	AXI-SD DECERR (statistics data-related) interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE1.SDDECEE = 1. Note: DECERR occurs when the space where the forwarding address is not allowed is accessed. If an error occurs, check that the forwarding address is correct. Note: This status is detected in the AXI Master of Statistics Data.
9	SDSLVES	0h	RW1	AXI-SD SLVERR (statistics data-related) interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE1.SDSLVEE = 1. Note: SLVERR occurs by the response from the other party as the destination for transfer from this bus. If this happens, check the destination settings. Note: This status is detected in the AXI Master of Statistics Data.
8	SDFOS	0h	RW1	FIFO-SD overflow (statistics data-related) interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE1.SDFOE = 1. Note: This status is detected in the AXI Master of Statistics Data.
7 to 4	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
3	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value will be ignored.
2	DECES	0h	RW1	AXI-VD DECERR (video data-related) interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE1.DECEE = 1. Note: DECERR occurs when the space where the forwarding address is not allowed is accessed. If an error occurs, check that the forwarding address is correct. Note: This status is detected in the AXI Master of Video Data.

Bit	Bit Name	Initial Value	R/W	Description
1	SLVES	0h	RW1	AXI-VD SLVERR (video data-related) interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit only is valid when CRUnIE1.SLVEE = 1. Note: SLVERR occurs by the response from the other party as the destination for transfer from this bus. If this happens, check the destination settings. Note: This status is detected in the AXI Master of Video Data.
0	FOS	0h	RW1	FIFO-VD overflow (video data-related) interrupt status (for debugging) Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE1.FOE = 1. Note: This status is detected in the AXI Master of Video Data.

(6) CRU Interrupt Status Register2 (CRUm_CRUnINTS2) (m = 0)

See **Figure 9.2-7** for where each bit of the interrupt is detected.

Access Size : 32 bits
Address : <CRUm_base> + 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	SDADR1WS	-	-	-	-	VDADR4WS	VDADR3WS	VDADR2WS	VDADR1WS	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	R	R	R	R	RW1	R	R	R	R	RW1	RW1	RW1	RW1	R

CRU	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	SL3S	FE3S	FS3S	SL2S	FE2S	FS2S	SL1S	FE1S	FS1S	SL0S	FE0S	FS0S
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1	RW1

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	0h	RW	Reserved Whenever it is read, 0 is read. The written value will be ignored.
29 to 26	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	SDADR1WS	0h	RW1	AXI-SD bus write completion interrupt status (statistics data-related) for designated address 1 (35 bits) Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.SDADR1WE = 1. Note: This status is detected in the AXI Master of Statistics Data.
24 to 21	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
20	VDADR4WS	0h	RW1	AXI-VD bus write completion interrupt status (video data-related) for designated address 4 (35 bits) Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.VDADR4WE = 1. Note: This status is detected in the AXI Master of Video Data.
19	VDADR3WS	0h	RW1	AXI-VD bus write completion interrupt status (video data-related) for designated address 3 (35 bits) Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.VDADR3WE = 1. Note: This status is detected in the AXI Master of Video Data.
18	VDADR2WS	0h	RW1	AXI-VD bus write completion interrupt status (video data-related) for designated address 2 (35 bits) Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.VDADR2WE = 1. Note: This status is detected in the AXI Master of Video Data.
17	VDADR1WS	0h	RW1	AXI-VD bus write completion interrupt status (video data-related) for designated address 1 (35 bits) Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.VDADR1WE = 1. Note: This status is detected in the AXI Master of Video Data.
16 to 12	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
11	SL3S	0h	RW1	SVC3 Scan line interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.SL3E = 1. Note: This status is detected in SVC3 Image Clipping.
10	FE3S	0h	RW1	SVC3 Frame end interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.FE3E = 1. Note: This status is detected in SVC3 Image Clipping.
9	FS3S	0h	RW1	SVC3 Frame start interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.FS3E = 1. Note: This status is detected in SVC3 Image Clipping.
8	SL2S	0h	RW1	SVC2 Scan line interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.SL2E = 1. Note: This status is detected in SVC2 Image Clipping.
7	FE2S	0h	RW1	SVC2 Frame end interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.FE2E = 1. Note: This status is detected in SVC2 Image Clipping.
6	FS2S	0h	RW1	SVC2 Frame start interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.FS2E = 1. Note: This status is detected in SVC2 Image Clipping.
5	SL1S	0h	RW1	SVC1 Scan line interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.SL1E = 1. Note: This status is detected in SVC1 Image Clipping.
4	FE1S	0h	RW1	SVC1 Frame end interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.FE1E = 1. Note: This status is detected in SVC1 Image Clipping.
3	FS1S	0h	RW1	SVC1 Frame start interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.FS1E = 1. Note: This status is detected in SVC1 Image Clipping.
2	SL0S	0h	RW1	SVC0 Scan line interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.SL0E = 1. Note: This status is detected in SVC0 Image Clipping.
1	FE0S	0h	RW1	SVC0 Frame end interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.FE0E = 1. Note: This status is detected in SVC0 Image Clipping.
0	FS0S	0h	RW1	SVC0 Frame start interrupt status Write 0: Invalid, 1: Clears the interrupt & status Read 0: No interrupts generated, 1: Interrupts generated Note: This bit is only valid when CRUnIE2.FS0E = 1. Note: This status is detected in SVC0 Image Clipping.

(7) CRU Reset Register (CRUm_CRUnRST) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0018h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	VRESETN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	VRESETN	0h	RW	0b: Resets the Image Converter module. 1b: Releases the Image Converter module from the reset state.

(8) CRU General Read/Write Register (CRUm_CRUnCOM) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 002Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	COMMON[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COMMON[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	COMMON[31:0]	0h	RW	General register The value written to this register can be read.

(9) Memory Bank 1 Base Address Lower Register for Video Data (CRUm_AMnMB1ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0040h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB1ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB1ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB1ADDRL [31:9]	0h	RW	Video data transfer destination base address 1 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(10) Memory Bank 1 Base Address Higher Register for Video Data (CRUm_AMnMB1ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0044h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	MB1ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	MB1ADDRH [34:32]	0h	RW	Video data transfer destination base address 1 (valid bits: [34:32]) (These bits cannot be modified while the AXI-VD is operating.)

(11) Memory Bank 2 Base Address Lower Register for Video Data (CRUm_AMnMB2ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0048h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB2ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB2ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB2ADDRL [31:9]	0h	RW	Video data transfer destination base address 2 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(12) Memory Bank 2 Base Address Higher Register for Video Data (CRUm_AMnMB2ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 004Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	MB2ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	MB2ADDRH [34:32]	0h	RW	Video data transfer destination base address 2 (valid bits: [34:32]) (These bits cannot be modified while the AXI-VD is operating.)

(13) Memory Bank 3 Base Address Lower Register for Video Data (CRUm_AMnMB3ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0050h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB3ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB3ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB3ADDRL [31:9]	0h	RW	Video data transfer destination base address 3 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(14) Memory Bank 3 Base Address Higher Register for Video Data (CRUm_AMnMB3ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0054h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	MB3ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	MB3ADDRH [34:32]	0h	RW	Video data transfer destination base address 3 (valid bits: [34:32]) (These bits cannot be modified while the AXI-VD is operating.)

(15) Memory Bank 4 Base Address Lower Register for Video Data (CRUm_AMnMB4ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0058h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB4ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB4ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB4ADDRL [31:9]	0h	RW	Video data transfer destination base address 4 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(16) Memory Bank 4 Base Address Higher Register for Video Data (CRUm_AMnMB4ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 005Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	MB4ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	MB4ADDRH [34:32]	0h	RW	Video data transfer destination base address 4 (valid bits: [34:32]) (These bits cannot be modified while the AXI-VD is operating.)

(17) Memory Bank 5 Base Address Lower Register for Video Data (CRUm_AMnMB5ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0060h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB5ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB5ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB5ADDRL [31:9]	0h	RW	Video data transfer destination base address 5 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(18) Memory Bank 5 Base Address Higher Register for Video Data (CRUm_AMnMB5ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0064h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	MB5ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	MB5ADDRH [34:32]	0h	RW	Video data transfer destination base address 5 (valid bits: [34:32]) (These bits cannot be modified while the AXI-VD is operating.)

(19) Memory Bank 6 Base Address Lower Register for Video Data (CRUm_AMnMB6ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0068h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB6ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB6ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB6ADDRL [31:9]	0h	RW	Video data transfer destination base address 6 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(20) Memory Bank 6 Base Address Higher Register for Video Data (CRUm_AMnMB6ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 006Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	MB6ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	MB6ADDRH [34:32]	0h	RW	Video data transfer destination base address 6 (valid bits: [34:32]) (These bits cannot be modified while the AXI-VD is operating.)

(21) Memory Bank 7 Base Address Lower Register for Video Data (CRUm_AMnMB7ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0070h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB7ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB7ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB7ADDRL [31:9]	0h	RW	Video data transfer destination base address 7 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(22) Memory Bank 7 Base Address Higher Register for Video Data (CRUm_AMnMB7ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0074h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	MB7ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	MB7ADDRH [34:32]	0h	RW	Video data transfer destination base address 7 (valid bits: [34:32]) (These bits cannot be modified while the AXI-VD is operating.)

(23) Memory Bank 8 Base Address Lower Register for Video Data (CRUm_AMnMB8ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0078h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB8ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB8ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	MB8ADDRL [31:9]	0h	RW	Video data transfer destination base address 8 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(24) Memory Bank 8 Base Address Higher Register for Video Data (CRUm_AMnMB8ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 007Ch
Initial Value : 0000_0000h

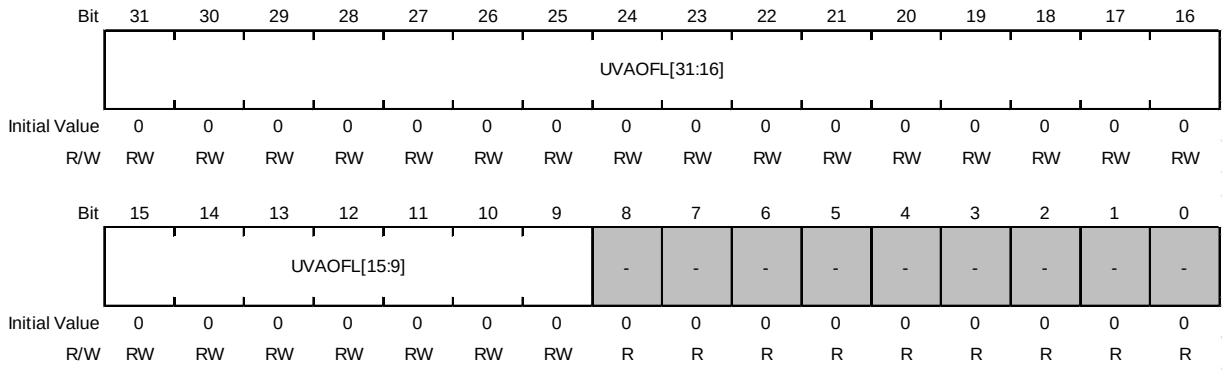
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	MB8ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	MB8ADDRH [34:32]	0h	RW	Video data transfer destination base address 8 (valid bits: [34:32]) (These bits cannot be modified while the AXI-VD is operating.)

(25) UV Data Address Offset Lower Register for Video Data (CRUm_AMnUVAOFL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0080h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 9	UVAOFL [31:9]	0h	RW	UV data address offset for image processing data YUV output (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(26) UV Data Address Offset Higher Register for Video Data (CRUm_AMnUVAOFH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0084h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	UVAOFH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	UVAOFH [34:32]	0h	RW	UV data address offset for image processing data YUV output (valid bits: [34:32]) (These bits cannot be modified while the AXI-VD is operating.)

(27) Memory Bank Enable Register for Video Data (CRUm_AMnMBVALID) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0088h
Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	MBVALID[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	MBVALID[7:0]	1h	RW	Video data transfer destination valid address (the details are described in **9.2.4.3.3 Transfer to DRAM**.) [0]: 0: MB1 invalid, 1: MB1 valid [1]: 0: MB2 invalid, 1: MB2 valid [2]: 0: MB3 invalid, 1: MB3 valid [3]: 0: MB4 invalid, 1: MB4 valid [4]: 0: MB5 invalid, 1: MB5 valid [5]: 0: MB6 invalid, 1: MB6 valid [6]: 0: MB7 invalid, 1: MB7 valid [7]: 0: MB8 invalid, 1: MB8 valid Note: Set the valid destination address from the LSB continuously. Data transferred from the LSB are written sequentially only to valid memory banks, and the processing is repeated from the start (LSB) when the last address is reached. (These bits cannot be modified while the AXI-VD is operating.)

(28) Memory Address Lower Status Register for Video Data (CRUm_AMnMADRSL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 008Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MADRSL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MADRSL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MADRSL[31:0]	0h	R	Memory address for video data (lower) Reads the memory address (lower) to which the current video data was written. Note: When this register is read, AMnMADRSH of the higher-order address also latches the address. Note: Be sure to read AMnMADRSH after AMnMADRSL has been read.

(29) Memory Address Higher Status Register for Video Data (CRUm_AMnMADRSH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0090h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	MADRSH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	MADRSH [34:32]	0h	R	Memory address for video data (higher) Read the memory address (higher) to which the current video data was written. Note: When the AMnMADRSL register is read, the higher address AMnMADRSH also latches the address. Note: Be sure to lead AMnMADRSH after AMnMADRSL lead.

(30) Non Image Process Address Offset Lower Register (CRUm_AMnNIAOFL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0094h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NIAOFL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NIAOFL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	NIAOFL[31:9]	0h	RW	Non-image processing address offset (lower) for image processing for video data (significant bits [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(31) Non Image Process Address Offset Higher Register (CRUm_AMnNIAOFH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0098h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	NIAOFH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	NIAOFH [34:32]	0h	RW	Non-image processing address offset (higher) for image processing for video data (valid bits [34:32]) (These bits cannot be modified while the AXI-VD is operating.)

(32) SVC Data Address Offset Lower Register for Video Data (CRUm_AMnSVCAOFL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 009Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SVCAOFL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SVCAOFL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SVCAOFL [31:9]	0h	RW	SVC data address offset (lower) for video data (valid bits [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(33) SVC Data Address Offset Higher Register for Video Data (CRUm_AMnSVCAOFH) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 00A0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SVCAOFH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SVCAOFH [34:32]	0h	RW	SVC data address offset (higher) for video data (valid bits [34:32]) (These bits cannot be modified while the AXI-VD is operating.)

(34) AXI-VD Bus Transfer Completion Event Address 0 Lower Register (CRUm_AMnIVT0ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 00B4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IVT0ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IVT0ADDRL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IVT0ADDRL [31:0]	0h	RW	AXI-VD bus transfer completion event address 0 for video data (lower) Note: The completion of transfer of video data to the address set in this register is notified by an interrupt.

(35) AXI-VD Bus Transfer Completion Event Address 0 Higher Register (CRUm_AMnIVT0ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 00B8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	IVT0ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	IVT0ADDRH [34:32]	0h	RW	AXI-VD bus transfer completion event address 0 for video data (higher) Note: The completion of transfer of video data to the address set in this register is notified by an interrupt.

(36) AXI-VD Bus Transfer Completion Event Address 1 Lower Register (CRUm_AMnIVT1ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 00BCh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IVT1ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IVT1ADDRL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IVT1ADDRL [31:0]	0h	RW	AXI-VD bus transfer completion event address 1 for video data (lower) Note: The completion of transfer of video data to the address set in this register is notified by an interrupt.

(37) AXI-VD Bus Transfer Completion Event Address 1 Higher Register (CRUm_AMnIVT1ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 00C0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	IVT1ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	IVT1ADDRH [34:32]	0h	RW	AXI-VD bus transfer completion event address 1 for video data (higher) Note: The completion of transfer of video data to the address set in this register is notified by an interrupt.

(38) AXI-VD Bus Transfer Completion Event Address 2 Lower Register (CRUm_AMnIVT2ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 00C4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IVT2ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IVT2ADDRL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IVT2ADDRL [31:0]	0h	RW	AXI-VD bus transfer completion event address 2 for video data (lower) Note: The completion of transfer of video data to the address set in this register is notified by an interrupt.

(39) AXI-VD Bus Transfer Completion Event Address 2 Higher Register (CRUm_AMnIVT2ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 00C8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	IVT2ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	IVT2ADDRH [34:32]	0h	RW	AXI-VD bus transfer completion event address 2 for video data (higher) Note: The completion of transfer of video data to the address set in this register is notified by an interrupt.

(40) AXI-VD Bus Transfer Completion Event Address 3 Lower Register (CRUm_AMnIVT3ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 00CCh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IVT3ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IVT3ADDRL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IVT3ADDRL [31:0]	0h	RW	AXI-VD bus transfer completion event address 3 for video data (lower) Note: The completion of transfer of video data to the address set in this register is notified by an interrupt.

(41) AXI-VD Bus Transfer Completion Event Address 3 Higher Register (CRUm_AMnIVT3ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 00D0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	IVT3ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	IVT3ADDRH [34:32]	0h	RW	AXI-VD bus transfer completion event address 3 for video data (higher) Note: The completion of transfer of video data to the address set in this register is notified by an interrupt.

(42) AXI-VD Bus Transfer Completion Event Address 4 Lower Register (CRUm_AMnIVT4ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 00D4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IVT4ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IVT4ADDRL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IVT4ADDRL [31:0]	0h	RW	AXI-VD bus transfer completion event address 4 for video data (lower) Note: The completion of transfer of video data to the address set in this register is notified by an interrupt.

(43) AXI-VD Bus Transfer Completion Event Address 4 Higher Register (CRUm_AMnIVT4ADDRH) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 00D8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	IVT4ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	IVT4ADDRH [34:32]	0h	RW	AXI-VD bus transfer completion event address 4 for video data (higher) Note: The completion of transfer of video data to the address set in this register is notified by an interrupt.

(44) AXI-VD Bus Master Transfer Setting Register (CRUm_AMnAXIATTR) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 00ECh
Initial Value : 0020_0150h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	AXILEN[3:0]			
Initial Value	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27 to 24	-	0h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.
23	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22 to 16	-	20h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9, 8	-	1h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 4	-	5h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.
3 to 0	AXILEN[3:0]	0h	RW	Maximum value setting of AXI-VD burst length for video data (changing the value during AXI-VD operation is prohibited) Note: If the transfer speed is given priority, set AXILEN[3:0] = Fh. When YUV is not separated (ICnDMR.YCMODE[2:0] ≠ 010b or 110b): 0h: Up to 1 burst (initial value) 1h: Up to 2 bursts 2h: Up to 3 bursts 3h: Up to 4 bursts 4h: Up to 5 bursts 5h: Up to 6 bursts ⋮ Fh: Up to 16 bursts (recommended value) When YUV is separated (ICnDMR.YCMODE[2:0] = 010b or 110b): 0h: Up to 1 burst 1h: Up to 1 burst 2h: Up to 2 bursts 3h: Up to 2 bursts 4h: Up to 3 bursts 5h: Up to 3 bursts ⋮ Eh: Up to 8 bursts Fh: Up to 8 bursts (recommended value)

(45) AXI-VD Bus Master FIFO Setting Register (m = 0) (CRUm_AMnFIFO)

Access Size : 32 bits

Address : <CRUm_base> + 00F0h

Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	RETSVC[2:0]			-	-	-	FIFOOVFREC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 4	RETSVC[2:0]	0h	RW	SVC number setting to resume when FIFO-VD overflow 000b : Resume from the beginning of SVC0 frame (initial value) 001b : Resume from the beginning of the SVC1 frame 010b : Resume from the beginning of the SVC2 frame 011b: Resume from the beginning of the SVC3 frame 1xxb : Resume from the VC first frame that came, without SVC designation
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	FIFOOVFREC	1h	RW	Mode setting when image data FIFO-VD overflows When a FIFO-VD overflows, the FIFO-VD is cleared and the AXI-VD master stops transferring to memory. There are two methods as shown below. 0b: Recovery mode: The AXI-VD Master performs the following operations after stopped. - Resume import from SVC selected by AMnFIFO.RETSVC[2:0] - Resume writing from the beginning of the overflow MB 1b: Stop mode (default): The FIFO remains stopped after the overflow occurs. [How to resume] When the AXI-VD master stops, it updates the FOS bit in the CRUnINTS1 register and reg_axi_mst_err_int Interrupts are generated. Set the AXI_TRANS_START bit of the AMnFIFOTRST register to 1 after detection. After this, the operation resumes with the same behavior as in recovery mode. Note 1: The image data being processed will be corrupted during FIFO overflow. FIFO overflow may be caused by bus congestion. Please take measures to alleviate bus congestion. Note 2: The data for the first horizontal line after FIFO-VD overflow return is not guaranteed. Note 3: If set Footer appending ON (ICnDTVP.FTON=1), only if no FIFO-VD overflow occurs. Set to Footer appending OFF (ICnDTVP.FTON=0), when FIFO-VD will overflow occurs.

(46) AXI-VD Bus Master Transfer Resume Register (m = 0) (CRUm_AMnFIFOTRST)

Access Size : 32 bits

Address : <CRUm_base> + 00F4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	AXI_TRANS_START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	AXI_TRANS_START	0h	RW1	AXI-VD transfer control for image data (valid when AMnFIFO.FIFOOVFREC = 1). 0b: Writing prohibited. Write 1 to this bit to recover the image-related AXI-VD (transfer resume) (This bit returns to 0 when transfer is resumed.)

(47) AXI-VD Bus Master FIFO Pointer Register (CRUm_AMnFIFOPNTR) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 00F8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIFORPNTR_B1[7:0]								FIFORPNTR_B0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FIFOWPNTR_B1[7:0]								FIFOWPNTR_B0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	FIFORPNTR_B1[7:0]	0h	R	FIFO-VD read pointer (bank 1) The read pointer and write pointer comparison allows checking if there are data in the FIFO. Bank 1 FIFO-VD read pointer [7:0] Real RAM address for FIFO-VD (Max.: FFh) Note : When (AMnFIFOPNTR.FIFOWPNTR_B0[7:0]≠ AMnFIFOPNTR.FIFORPNTR_B0[7:0])‖(AMnFIFOPNTR.FIFOWPNTR_B1[7:0]≠ AMnFIFOPNTR.FIFORPNTR_B1[7:0], data are present in the FIFO-VD.
23 to 16	FIFORPNTR_B0[7:0]	0h	R	FIFO-VD read pointer (bank 0) The read pointer and write pointer comparison allows checking if there are data in the FIFO. Bank 0 FIFO-VD read pointer [7:0] Real RAM address for FIFO-VD (Max.: FFh) Note: When (AMnFIFOPNTR.FIFOWPNTR_B0[7:0]≠ AMnFIFOPNTR.FIFORPNTR_B0[7:0])‖(AMnFIFOPNTR.FIFOWPNTR_B1[7:0]≠ AMnFIFOPNTR.FIFORPNTR_B1[7:0], data are present in the FIFO-VD.
15 to 8	FIFOWPNTR_B1[7:0]	0h	R	FIFO-VD write pointer (bank 1) The read pointer and write pointer comparison allows checking if there are data in the FIFO. Bank 1 FIFO-VD write pointer [7:0] Real RAM address for FIFO-VD (Max.: FFh) Note: When (AMnFIFOPNTR.FIFOWPNTR_B0[7:0]≠ AMnFIFOPNTR.FIFORPNTR_B0[7:0])‖(AMnFIFOPNTR.FIFOWPNTR_B1[7:0]≠ AMnFIFOPNTR.FIFORPNTR_B1[7:0], data are present in the FIFO-VD.
7 to 0	FIFOWPNTR_B0[7:0]	0h	R	FIFO-VD write pointer (bank 0) The read pointer and write pointer comparison allows checking if there are data in the FIFO. Bank 0 FIFO-VD write pointer [7:0] Real RAM address for FIFO-VD (Max.: FFh) Note: When (AMnFIFOPNTR.FIFOWPNTR_B0[7:0]≠ AMnFIFOPNTR.FIFORPNTR_B0[7:0])‖(AMnFIFOPNTR.FIFOWPNTR_B1[7:0]≠ AMnFIFOPNTR.FIFORPNTR_B1[7:0], data are present in the FIFO-VD.

(48) AXI-VD Bus Master Transfer Stop Register (CRUm_AMnAXISTP) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0110h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	AXI_STOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	AXI_STOP	0h	RW	AXI-VD bus transfer stop (used to stop the CRU receiving data. Follow the procedure for stopping reception described in **9.2.5.2 Stopping Reception for the MIPI CSI-2 Input.**) 0b: Does not stop the AXI-VD master transferring data. 1b: Stops the AXI-VD master transferring data.

(49) AXI-VD Bus Master Transfer Stop Status Register (CRUm_AMnAXISTPACK) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0114h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	AXI_STOP_ACK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	AXI_STOP_ACK	0h	R	AXI-VD bus transfer stop response (used to stop the CRU receiving data. Follow the procedure for stopping reception described in **9.2.5.2 Stopping Reception for the MIPI CSI-2 Input.**) 0b: The AXI-VD master is transferring data. 1b: The AXI-VD master has finished transferring data (valid when AXI_STOP = 1)

(50) Image Stride Setting Register (CRUm_AMnIS) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0128h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	IS[7:0]								-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14 to 7	IS[7:0]	0h	RW	Line start address offset setting for image stride You can set 128 to 32640 bytes (with a 128-byte step) by setting IS[7:0]. IS[7:0] = 00h: Setting prohibited IS[7:0] = 01h: 128 bytes IS[7:0] = 02h: 256 bytes IS[7:0] = 03h: 384 bytes IS[7:0] = 04h: 512 bytes ⋮ IS[7:0] = 80h: 16384 bytes ⋮ IS[7:0] = FFh: 32640 bytes Note: For output format RGB888 (24 bits), the setting must be in units of 384 bytes (multiples of 3 for IS [7:0]). Note: If the output format is other than RGB888 (24 bits), the setting must be in units of 128 bytes. Note: The setting must be at least the number of bytes per line. Note: When adding a header and footer, set the value including a header and footer. Note: Only data output from the video data bus are supported. Statistics data output is not supported.
6 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(51) Memory Bank 1 Base Address Lower Register for CRU Statistics Data (CRUm_AMnSDMB1ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 013Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB1ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB1ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB1ADDRL [31:9]	0h	RW	Statistics data transfer destination base address 1 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(52) Memory Bank 1 Base Address Higher Register for CRU Statistics Data (CRUm_AMnSDMB1ADDRH) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0140h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SDMB1ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDMB1ADDRH [34:32]	0h	RW	Statistics data transfer destination base address 1 (valid bits: [34:32]) (These bits cannot be modified while the AXI-SD is operating.)

(53) Memory Bank 2 Base Address Lower Register for CRU Statistics Data (CRUm_AMnSDMB2ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0144h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB2ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB2ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB2ADDRL [31:9]	0h	RW	Statistics data transfer destination base address 2 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(54) Memory Bank 2 Base Address Higher Register for CRU Statistics Data (CRUm_AMnSDMB2ADDRH) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0148h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SDMB2ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDMB2ADDRH [34:32]	0h	RW	Statistics data transfer destination base address 2 (valid bits: [34:32]) (These bits cannot be modified while the AXI-SD is operating.)

(55) Memory Bank 3 Base Address Lower Register for CRU Statistics Data (CRUm_AMnSDMB3ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 014Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB3ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB3ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB3ADDRL [31:9]	0h	RW	Statistics data transfer destination base address 3 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(56) Memory Bank 3 Base Address Higher Register for CRU Statistics Data (CRUm_AMnSDMB3ADDRH) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0150h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SDMB3ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDMB3ADDRH [34:32]	0h	RW	Statistics data transfer destination base address 3 (valid bits: [34:32]) (These bits cannot be modified while the AXI-SD is operating.)

(57) Memory Bank 4 Base Address Lower Register for CRU Statistics Data (CRUm_AMnSDMB4ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0154h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB4ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB4ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB4ADDRL [31:9]	0h	RW	Statistics data transfer destination base address 4 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(58) Memory Bank 4 Base Address Higher Register for CRU Statistics Data (CRUm_AMnSDMB4ADDRH) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0158h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SDMB4ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDMB4ADDRH [34:32]	0h	RW	Statistics data transfer destination base address 4 (valid bits: [34:32]) (These bits cannot be modified while the AXI-SD is operating.)

(59) Memory Bank 5 Base Address Lower Register for CRU Statistics Data (CRUm_AMnSDMB5ADDRL) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 015Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB5ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB5ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB5ADDRL [31:9]	0h	RW	Statistics data transfer destination base address 5 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(60) Memory Bank 5 Base Address Higher Register for CRU Statistics Data (CRUm_AMnSDMB5ADDRH) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0160h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SDMB5ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDMB5ADDRH [34:32]	0h	RW	Statistics data transfer destination base address 5 (valid bits: [34:32]) (These bits cannot be modified while the AXI-SD is operating.)

(61) Memory Bank 6 Base Address Lower Register for CRU Statistics Data (CRUm_AMnSDMB6ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0164h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB6ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB6ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB6ADDRL [31:9]	0h	RW	Statistics data transfer destination base address 6 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(62) Memory Bank 6 Base Address Higher Register for CRU Statistics Data (CRUm_AMnSDMB6ADDRH) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0168h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SDMB6ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDMB6ADDRH [34:32]	0h	RW	Statistics data transfer destination base address 6 (valid bits: [34:32]) (These bits cannot be modified while the AXI-SD is operating.)

(63) Memory Bank 7 Base Address Lower Register for CRU Statistics Data (CRUm_AMnSDMB7ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 016Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB7ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB7ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB7ADDRL [31:9]	0h	RW	Statistics data transfer destination base address 7 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(64) Memory Bank 7 Base Address Higher Register for CRU Statistics Data (CRUm_AMnSDMB7ADDRH) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0170h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SDMB7ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDMB7ADDRH [34:32]	0h	RW	Statistics data transfer destination base address 7 (valid bits: [34:32]) (These bits cannot be modified while the AXI-SD is operating.)

(65) Memory Bank 8 Base Address Lower Register for CRU Statistics Data (CRUm_AMnSDMB8ADDRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0174h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDMB8ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDMB8ADDRL[15:9]							-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	SDMB8ADDRL [31:9]	0h	RW	Statistics data transfer destination base address 8 (valid bits: [31:9]) Set it as the base address in units of 512 bytes. Note: The 9 lower-order bits cannot be written. (These bits cannot be modified while the AXI-VD is operating.)
8 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(66) Memory Bank 8 Base Address Higher Register for CRU Statistics Data (CRUm_AMnSDMB8ADDRH) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0178h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SDMB8ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDMB8ADDRH [34:32]	0h	RW	Statistics data transfer destination base address 8 (valid bits: [34:32]) (These bits cannot be modified while the AXI-SD is operating.)

(67) Memory Bank Enable Register for CRU Statistics Data (CRUm_AMnSDMBVALID) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 018Ch
Initial Value : 0000_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SDMBVALID[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	SDMBVALID [7:0]	1h	RW	Statistics data transfer destination valid address (the details are described in **9.2.4.3.3 Transfer to DRAM**.) [0]: 0: SDMB1 invalid, 1: SDMB1 valid [1]: 0: SDMB2 invalid, 1: SDMB2 valid [2]: 0: SDMB3 invalid, 1: SDMB3 valid [3]: 0: SDMB4 invalid, 1: SDMB4 valid [4]: 0: SDMB5 invalid, 1: SDMB5 valid [5]: 0: SDMB6 invalid, 1: SDMB6 valid [6]: 0: SDMB7 invalid, 1: SDMB7 valid [7]: 0: SDMB8 invalid, 1: SDMB8 valid Note: Set the valid destination address from the LSB continuously. Data transferred from the LSB are written sequentially only to valid memory banks, and the processing is repeated from the start (LSB) when the last address is reached. (These bits cannot be modified while the AXI-SD is operating.)

(68) Memory Bank Status Register for CRU Statistics Data (CRUm_AMnSDMBS) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0190h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SDMBSTS[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDMBSTS[2:0]	0h	R	Indicates which memory bank from SDMB1 to SDMB8 the statistics data are being transferred to. (0h: SDMB1, 1h: SDMB2 … 6h: SDMB7, 7h: SDMB8)

(69) AXI-SD Bus Transfer Completion Event Address 0 Lower Register (CRUm_AMnSDIVT0ADDRL) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0194h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDIVT0ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDIVT0ADDRL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SDIVT0ADDRL [31:0]	0h	RW	AXI-SD bus transfer completion event address 0 for statistics data (lower) Note: The completion of transfer of statistics data to the address set in this register is notified by an interrupt.

(70) AXI-SD Bus Transfer Completion Event Address 0 Higher Register (CRUm_AMnSDIVT0ADDRH) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0198h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SDIVT0ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDIVT0ADDRH [34:32]	0h	RW	AXI-SD bus transfer completion event address 0 for statistics data (higher) Note: The completion of transfer of statistics data to the address set in this register is notified by an interrupt.

(71) AXI-SD Bus Transfer Completion Event Address 1 Lower Register (CRUm_AMnSDIVT1ADDRL) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 019Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDIVT1ADDRL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDIVT1ADDRL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SDIVT1ADDRL [31:0]	0h	RW	AXI-SD bus transfer completion event address 1 for statistics data (lower) Note: The completion of transfer of statistics data to the address set in this register is notified by an interrupt.

(72) AXI-SD Bus Transfer Completion Event Address 1 Higher Register (CRUm_AMnSDIVT1ADDRH) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 01A0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	SDIVT1ADDRH[34:32]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	SDIVT1ADDRH [34:32]	0h	RW	AXI-SD bus transfer completion event address 1 for statistics data (higher) Note: The completion of transfer of statistics data to the address set in this register is notified by an interrupt.

(73) AXI-SD Bus Master FIFO Pointer Register (CRUm_AMnSDFIFOPNTR) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 01C0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	SDFIFORPNTR[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	SDFIFOWPNTR[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21 to 16	SDFIFORPNTR [5:0]	0h	R	FIFO-SD read pointer for statistics data The read pointer and write pointer comparison allows checking if there are data in the FIFO. FIFO-SD read pointer [5:0] Real RAM address for FIFO-SD (Max.: 3Fh) Note: When (AMnSDFIFOPNTR.SDFIFOWPNTR[5:0]≠AMnSDFIFOPNTR.SDFIFORPNTR[5:0]), data are present in the FIFO-SD.
15 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5 to 0	SDFIFOWPNTR [5:0]	0h	R	FIFO-SD write pointer for statistics data The read pointer and write pointer comparison allows checking if there are data in the FIFO. Bank 1 FIFO-SD write pointer [5:0] Real RAM address for FIFO-SD (Max.: 3Fh) Note: When (AMnSDFIFOPNTR.SDFIFOWPNTR[5:0]≠AMnSDFIFOPNTR.SDFIFORPNTR[5:0]), data are present in the FIFO-SD.

(74) AXI-SD Bus Master Transfer Stop Register (CRUm_AMnSDAXISTP) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 01D8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SD_AXI_STOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	SD_AXI_STOP	0h	RW	Statistics data AXI-SD transfer stop (used to stop the CRU receiving data. Follow the procedure for stopping reception described in **9.2.5.2 Stopping Reception for the MIPI CSI-2 Input**.) 0b: Does not stop the AXI-SD master transferring data. 1b: Stops the AXI-SD master transferring data.

(75) AXI-SD Bus Master Transfer Stop Status Register (CRUm_AMnSDAXISTPACK) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 01DCh

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SD_AXI_STOP_ACK
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	SD_AXI_STOP_ACK	0h	R	Statistics data AXI-SD transfer stop response (used to stop the CRU receiving data. Follow the procedure for stopping reception described in **9.2.5.2 Stopping Reception for the MIPI CSI-2 Input**.) 0b: The AXI-SD master is transferring data. 1b: The AXI-SD master has finished transferring data (valid when AXI_STOP = 1)

(76) CRU Image Converter Enable Register (CRUm_ICnEN) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 01F0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ICEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	ICEN	0h	RW	Enables the operation of the Image Converter module. (Enable or disable the operation according to **9.2.5 Operation**.) 0b: Disables the Image Converter module. 1b: Enables the Image Converter module. Note: If this bit is set to 0, a frame start is not detected. If this bit is changed from 1 to 0 while a frame is being processed, the data at that time is processed as the frame end and then the next frames are not processed.

(77) CRU Data Value Processing (CRUm_ICnDTVP) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 01F4h
Initial Value : 0000_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	FTON	HDON	-	IBINSEL	CLP[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	FTON	0h	RW	Footer appending ON/OFF 0b: Do not append a footer 1b: Append a footer Note: The footers of both video data and statistics data are turned on or off. Note: For the video data footer, 256 bits (256'h1/256'h0) are written at the end of each line. Initially, 1h is written as a footer. The next time the memory bank is overwritten after one round of writing, 0h is written as the footer. After that, 1h and 0h are written alternately every time the memory bank is overwritten. Note: For the footer of statistics data, 512 bits ({higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256'h1} or 512 bits {higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256'h0}) are written at the end of each frame unit. The higher-order 256 bits of 512 bits are reserved (indefinite value), so skip reading them. Initially, 512 bits {higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256'h1} are written as a footer. Next, when the memory bank is overwritten after one round of writing, 512 bits {higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256'h0} are written as a footer. After that, 512 bits {higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256'h1} and 512 bits {higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256'h0} are written alternately each time the memory bank is overwritten.
4	HDON	0h	RW	Header appending ON/OFF ON/OFF selection for whether to append a header to the start of a line for transfer data from the AXI-VD bus to DRAM 0: Do not append a header 1: Append a header Note: The data type, byte number, and virtual channel number are added to the header (256 bits). Note: See **9.2.4.3.3 (5) Frame Structure**.
3	-	0h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.
2	IBINSEL	0h	RW	Data Arranger 1 UV(CbCr) binary type selection for YUV422(YCbCr) data 0b: Offset binary (straight binary) 1b: Two's complement binary Note: Valid when the input is YUV422/420.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	CLP[1:0]	3h	RW	Data clipping setting for the YCbCr422 (YUV422) format To make the input data compliant with the ITU-R BT.601 standard, specify whether to clip the values below 16 and above 240 (when YCbCr is 8 bits). When CRUm_ICnIPMC_C0/X.CSCTHR is set to 1, CLP[1:0] are set to 00b. 00b: No clipping for Y When CbCr is 8 bits: Values below 16 and above 240 are clipped for CbCr When CbCr is 10 bits: Values below 64 and above 960 are clipped for CbCr 01b: When YCbCr is 8 bits: Values below 16 and above 240 are clipped for Y : Values below 16 and above 240 are clipped for CbCr When YCbCr is 10 bits: Values below 64 and above 960 are clipped for Y : Values below 64 and above 960 are clipped for CbCr 10b: No clipping for Y When CbCr is 8 bits: Values below 16 are increased to 128 and values above 240 are clipped to 128 for CbCr When CbCr is 10 bits: Values below 64 are increased to 512 and values above 960 are clipped to 512 for CbCr 11b: No clipping for Y and CbCr (initial value) Note: See **9.2.4.3.10 Data Clipping**.

(78) CRU SVC Number Register (CRUm_ICnSVCNUM) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 01F8h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SVCNUM[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	SVCNUM[1:0]	0h	RW	Set how many channels are used from among SVC0 to 3. 00b: Only SVC0 enabled; SVC1 to 3 disabled (initial value) 01b: SVC0 and 1 enabled; SVC2 and 3 disabled 10b: SVC0 to 2 enabled; SVC3 disabled 11b: SVC0 to 3 enabled Note: When changing the frame synchronization register, only the setting of ICnSVCNUM.SVCNUM [1:0] = 00 can be used. Note: Select virtual channels for SVC0 to 3 by the ICnSVC register setting. Note: When using demosaicing, linear matrix, or statistics, the setting of this must be 00b.

(79) CRU VC Select Register (CRUm_ICnSVC) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 01FCh
Initial Value : 0000_3210h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SVC3[3:0]				SVC2[3:0]				SVC1[3:0]				SVC0[3:0]			
Initial Value	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 12	SVC3[3:0]	3h	RW	Selected Virtual Channel 3 (SVC3) The maximum number of channels that can be used by the image converter is 4 (SVC0~3). Assign one of VC0~15 to SVC3. 0h: Virtual channel 0 selected 1h: Virtual channel 1 selected 2h: Virtual channel 2 selected : Eh: Virtual channel 14 selected Fh: Virtual channel 15 selected Note: Do not set the same value as other SVC0/1/2[3:0].
11 to 8	SVC2[3:0]	2h	RW	Selected Virtual Channel 2 (SVC2) The maximum number of channels that can be used by the image converter is 4 (SVC0~3). Assign one of VC0~15 to SVC2. 0h: Virtual channel 0 selected 1h: Virtual channel 1 selected 2h: Virtual channel 2 selected : Eh: Virtual channel 14 selected Fh: Virtual channel 15 selected Note: Do not set the same value as other SVC0/1/3[3:0].
7 to 4	SVC1[3:0]	1h	RW	Selected Virtual Channel 1 (SVC1) The maximum number of channels that can be used by the image converter is 4 (SVC0~3). Assign one of VC0~15 to SVC1. 0h: Virtual channel 0 selected 1h: Virtual channel 1 selected 2h: Virtual channel 2 selected : Eh: Virtual channel 14 selected Fh: Virtual channel 15 selected Note: Do not set the same value as other SVC0/2/3[3:0].
3 to 0	SVC0[3:0]	0h	RW	Selected Virtual Channel 0 (SVC0) The maximum number of channels that can be used by the image converter is 4 (SVC0~3). Assign one of VC0~15 to SVC0. 0h: Virtual channel 0 selected 1h: Virtual channel 1 selected 2h: Virtual channel 2 selected : Eh: Virtual channel 14 selected Fh: Virtual channel 15 selected Note: Do not set the same value as other SVC1/2/3[3:0].

(80) CRU Image Converter Main Control Register (SVC0) (CRUm_ICnIPMC_C0) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0200h
Initial Value : 001E_00FEh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	RAWSTTYP [1:0]		-	-	INF[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
R/W	R	R	R	R	R	R	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	STITHR	LUTTHR	CSCTHR	LMXTHR	DEMTHR	CLPTHR	DECTHR	-
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25, 24	RAWSTTYP [1:0]	0h	RW	Specify the color component of the first pixel of the first line of RAW data for SVC0 00b: R (RGrRGr...) 01b: Gr (GrRGrR...) 10b: Gb (GbBGbB...) 11b: B (BGbBGb...)
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21 to 16	INF[5:0]	1Eh	RW	Specify the MIPI input image formats for SVC0 to be processed by the image converter module. For details about the formats, see Data Type Codes in the MIPI CSI-2 V2.1 specification. Of the formats specified here, only one format is processed. Note 1. Initial value: YUV422 8-bit Note 2. The following values can be set. Other settings are prohibited. 13h to 16h: Generic Long Packet Data Type 18h to 1Ah, 1Ch to 1Fh: YUV Data 20h to 24h: RGB Data 28h to 2Fh: RAW Data 30h to 37h: User-Defined Byte-based Data The data types specified here must be set in CSI2nDTEL.DTEN and CSI2nDTEH.DTEN of MIPI CSI2.
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	STITHR	1h	RW	Bypassing statistics processing for SVC0 (the details are described in **9.2.4.3.18 Statistics**.) 0b: Statistics processing is performed. 1b: Statistics processing is not performed (bypassed). Note: The setting 0b only applies for the corresponding formats shown in **9.2.4.3.1 Input Formats and Image Processing**.
6	LUTTHR	1h	RW	Bypassing LUT-based image color conversion for SVC0 Regardless of whether the setting of this bit is 0b or 1b, the output of the LUT is 8 bits for 10-bit input. (the details are described in **9.2.4.3.14 LUT (Lookup Table)**.) 0b: Image color conversion is performed. 1b: Image color conversion is not performed (bypassed). Note: The setting 0b only applies for the corresponding formats shown in **9.2.4.3.1 Input Formats and Image Processing**.
5	CSCTHR	1h	RW	Bypassing color space conversion and Data clipping for SVC0 (YCbCr(YUV) → RGB, RGB → YCbCr(YUV)) (the details are described in **9.2.4.3.13 Color Space Conversion**.) 0b: Color space conversion and Data clipping are performed. 1b: Color space conversion and Data clipping are not performed (bypassed). Note 1. The setting 0b only applies for the corresponding formats shown in **9.2.4.3.1 Input Formats and Image Processing**. Note 2. Even when this bit is 1, the data is expanded from 8 bits to 10 bits.

Bit	Bit Name	Initial Value	R/W	Description
4	LMXTHR	1h	RW	Bypassing liner matrix processing for SVC0 (the details are described in **9.2.4.3.12 Liner Matrix**.) 0b: Liner matrix processing is performed. 1b: Liner matrix processing is not performed (bypassed). Note: The setting 0b only applies for the corresponding formats shown in **9.2.4.3.1 Input Formats and Image Processing**.
3	DEMTHR	1h	RW	Bypassing demosaicing of RAW data for SVC0 (the details are described in **9.2.4.3.11 Demosaicing**.) 0b: Demosaicing is performed. 1b: Demosaicing is not performed (bypassed). Note: The setting 0b only applies for the corresponding formats shown in **9.2.4.3.1 Input Formats and Image Processing**.
2	CLPTHR	1h	RW	Bypassing image clipping for SVC0 (the details are described in **9.2.4.3.7 Image Clipping**.) 0b: Image clipping is performed. 1b: Image clipping is not performed (bypassed). Note: The setting 0b only applies for the corresponding formats shown in **9.2.4.3.1 Input Formats and Image Processing**.
1	DECTHR	1h	RW	Bypassing frame subsampling for SVC0 (the details are described in **9.2.4.3.8 Frame Subsampling**.) 0b: Subsampling is performed. 1b: Subsampling is not performed (bypassed). Note: Image processing, non-image processing, and statistics data are supported. Note: The setting 0b only applies for the corresponding formats shown in **9.2.4.3.1 Input Formats and Image Processing**.
0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(81) CRU Image Converter Main Control Register (SVCx) (CRUm_ICnIPMC_Cx) (m = 0)

Note: x = 1 to 3 CRU Non Image Process Data Type Code Select Lower Register

Access Size : 32 bits

Address : <CRUm_base> + 0258h (x = 1)
<CRUm_base> + 025Ch (x = 2)
<CRUm_base> + 0260h (x = 3)

Initial Value : 001E_0066h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	INF[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
R/W	R	R	R	R	R	R	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	LUTTHR	CSCTHR	-	-	CLPTHR	DECTHR	-
Initial Value	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	R	R	RW	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25, 24	-	All 0	RW	Reserved These bits are read as 0b. The write value should always be 0b.
23, 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21 to 16	INF[5:0]	1Eh	RW	Specify the MIPI input image formats for SVCx to be processed by the image converter module. For details about the formats, see Data Type Codes in the MIPI CSI-2 V2.1 specification. Of the formats specified here, only one format is processed. Note 1. Initial value: YUV422 8-bit Note 2. The following values can be set. Other settings are prohibited. 13h to 16h: Generic Long Packet Data Type 18h to 1Ah, 1Ch to 1Fh: YUV Data 20h to 24h: RGB Data 28h to 2Fh: RAW Data 30h to 37h: User-Defined Byte-based Data The data types specified here must be set in CSI2nDTEL.DTEN and CSI2nDTEH.DTEN of MIPI CSI2.
15 to 7	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6	LUTTHR	1h	RW	Bypassing LUT-based image color conversion for SVCx (the details are described in **9.2.4.3.14 LUT (Lookup Table)**.) Regardless of whether the setting of this bit is 0b or 1b, the output of the LUT is 8 bits for 10-bit input. 0b: Image color conversion is performed. 1b: Image color conversion is not performed (bypassed). Note: The setting 0b only applies for the corresponding formats shown in **9.2.4.3.1 Input Formats and Image Processing**.
5	CSCTHR	1h	RW	Bypassing color space conversion and Data clipping for SVCx (YCbCr(YUV) → RGB, RGB → YCbCr(YUV)) (the details are described in **9.2.4.3.13 Color Space Conversion**.) 0b: Color space conversion and Data clipping are performed. 1b: Color space conversion and Data clipping are not performed (bypassed). Note 1. The setting 0b only applies for the corresponding formats shown in **9.2.4.3.1 Input Formats and Image Processing**. Note 2. Even when this bit is 1, the data is expanded from 8 bits to 10 bits.
4, 3	-	All 0	RW	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
2	CLPTHR	1h	RW	Bypassing image clipping for SVCx (the details are described in **9.2.4.3.7 Image Clipping**.) 0b: Image clipping is performed. 1b: Image clipping is not performed (bypassed). Note: The setting 0b only applies for the corresponding formats shown in **9.2.4.3.1 Input Formats and Image Processing**.
1	DECTHR	1h	RW	Bypassing frame subsampling for SVCx (the details are described in **9.2.4.3.8 Frame Subsampling**.) 0b: Subsampling is performed. 1b: Subsampling is not performed (bypassed). Note: Image processing, non-image processing, and statistics data are supported. Note: The setting 0b only applies for the corresponding formats shown in **9.2.4.3.1 Input Formats and Image Processing**.
0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(82) CRU Non Image Process Data Type Code Select Lower Register (SVCx) (CRUm_ICnNIPDT_CxL) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0204h (x = 0)
<CRUm_base> + 0264h (x = 1)
<CRUm_base> + 026Ch (x = 2)
<CRUm_base> + 0274h (x = 3)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NIPDT_CxL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

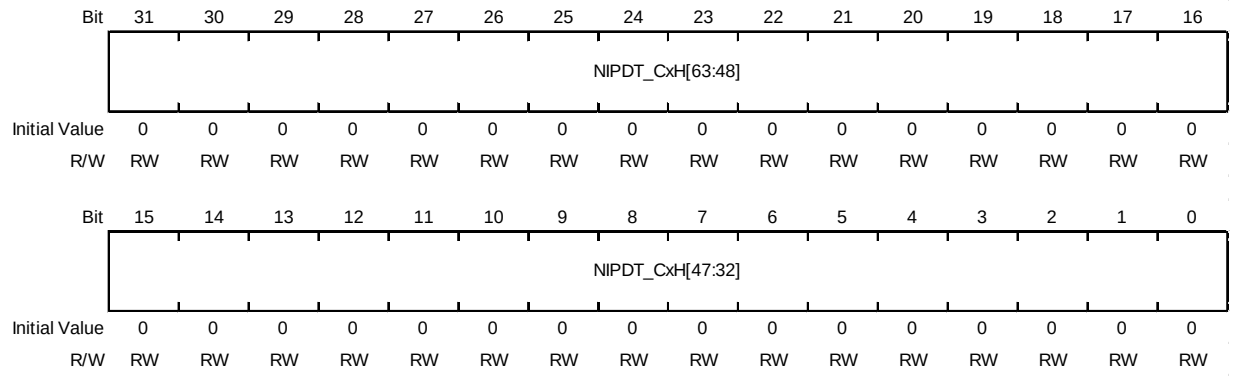
Bit	Bit Name	Initial Value	R/W	Description
31 to 16	NIPDT_CxL [31:16]	0h	RW	Non-image processing/full-bypass data type code selection (lower) selected by SVCx Each bit number in NIPDT_CxL [31:16] matches the decimal value of the data type number specified in MIPI CSI-2 v2.1. Select whether to send the data type corresponding to that number. NIPDT_CxL Corresponding data types [16] 10h [17] 11h ⋮ ⋮ [31] 1Fh When each bit is 0b: Do not send data of the corresponding data type as SVCx. When each bit is 1b: Data of the corresponding data type is transmitted as SVCx.
15 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(83) CRU Non Image Process Data Type Code Select Higher Register (SVCx) (CRUm_ICnNIPDT_CxH) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0208h (x = 0)
<CRUm_base> + 0268h (x = 1)
<CRUm_base> + 0270h (x = 2)
<CRUm_base> + 0278h (x = 3)

Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NIPDT_CxH [63:32]	0h	RW	Non-image processing/full-bypass data type code selection (higher) selected by SVCx Each bit number in NIPDT_CxH [63:32] matches the decimal value of the data type number specified in MIPI CSI-2 v2.1. Select whether to send the data type corresponding to that number. NIPDT_CxH Corresponding data types [32] 20h [33] 21h ⋮ ⋮ [63] 3Fh When each bit is 0b: Do not send data of the corresponding data type as SVCx. When each bit is 1b: Data of the corresponding data type is transmitted as SVCx.

(84) CRU Image Clipping Start Line Register (SVCx) (CRUm_ICnSLPrC_Cx) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 020Ch (x = 0)
<CRUm_base> + 028Ch (x = 1)
<CRUm_base> + 02A0h (x = 2)
<CRUm_base> + 02B4h (x = 3)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SLPrC[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	SLPrC[12:0]	0h	RW	Start line for SVCx in the vertical direction when clipping the input image to the specified area. Set a value of 1 or greater because the frame start is processed as the first line. The image is output from the specified line. (If 0 is specified, the same operation as for setting of 1 is performed.) Note: Set a value such that 1 ≤ ICnSLPrC_Cx.SLPrC ≤ ICnELPrC_Cx.ELPrC. Note: Only image processing data are supported.

(85) CRU Image Clipping End Line Register (SVCx) (CRUm_ICnELPrC_Cx) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0210h (x = 0)
<CRUm_base> + 0290h (x = 1)
<CRUm_base> + 02A4h (x = 2)
<CRUm_base> + 02B8h (x = 3)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	ELPrC[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	ELPrC[12:0]	0h	RW	End line for SVCx in the vertical direction when clipping the input image to the specified area. Set a value of 1 or greater because the frame start is processed as the first line. The image is output up to the specified line. Note: Set a value such that 1 ≤ ICnSLPrC_Cx.SLPrC ≤ ICnELPrC_Cx.ELPrC. Note: Only image processing data are supported.

(86) CRU Image Clipping Start Pixel Register (SVCx) (CRUm_ICnSPPrC_Cx) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0214h (x = 0)
<CRUm_base> + 0294h (x = 1)
<CRUm_base> + 02A8h (x = 2)
<CRUm_base> + 02BCh (x = 3)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SPPrC[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	SPPrC[12:0]	0h	RW	Start pixel for SVCx in the horizontal direction when clipping the input image to the specified area. Set a value of 1 or greater because the line start is processed as the first pixel. The image is output from the specified pixel data. (If 0 is specified, the same operation as for setting of 1 is performed.) Note 1. Set a value such that 1 ≤ ICnSPPrC_Cx.SPPrC ≤ (ICnEPPrC_Cx.EPPrC+1-320). Note 2. Specify an odd value when the input format is YUV422 (ICnIPMC_Cx.INF[5:0] = 1Eh to 1Fh). If an even number of 2 or greater is specified, the CRU operates as if an odd number (the specified number - 1) is specified. (Example: When 34 is specified, the data is output from the 33rd pixel.) Note 3. Even when the input format is RGB, etc., set a value that satisfies the following formula so that a line consists of an even number of pixels if the RGBtoYCbCr output is YUV422/420. ICnEPPrC_Cx.EPPrC - ICnSPPrC_Cx.SPPrC = 2N - 1 (N = 160, 161, 162,...) Note: Only image processing data are supported.

(87) CRU Image Clipping End Pixel Register (SVCx) (CRUm_ICnEPPrC_Cx) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0218h (x = 0)
<CRUm_base> + 0298h (x = 1)
<CRUm_base> + 02ACh (x = 2)
<CRUm_base> + 02C0h (x = 3)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	EPPrC[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	EPPrC[12:0]	0h	RW	End pixel for SVCx in the horizontal direction when clipping the input image to the specified area. Set a value of 1 or greater because the line start is processed as the first pixel. The image is output up to the specified image data. Note 1. Set a value such that 1 ≤ ICnSPPrC_Cx.SPPrC ≤ (ICnEPPrC_Cx.EPPrC + 1 - 320). Note 2. Specify an even value when the input format is YUV422 (ICnIPMC_Cx.INF[5:0] = 1Eh to 1Fh). If an odd number is specified, the CRU operates as if an even number (the specified number + 1) is specified. (Example: When 127 is specified, the data is output up to the 128th pixel.) Note 3. Even when the input format is RGB, etc., set a value that satisfies the following formula so that a line consists of an even number of pixels if the RGBtoYCbCr output is YUV422/420. ICnEPPrC_Cx.EPPrC - ICnSPPrC_Cx.SPPrC = 2N - 1 (N = 160, 161, 162,...) Note: Only image processing data are supported.

(88) CRU Scan Line Interrupt Register (SVCx) (CRUm_ICnSI_Cx) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 021Ch (x = 0)
<CRUm_base> + 029Ch (x = 1)
<CRUm_base> + 02B0h (x = 2)
<CRUm_base> + 02C4h (x = 3)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SI[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	SI[12:0]	0h	RW	Line specification for generating a scan line interrupt (for SVCx) When the line number being processed in a frame of the Image Converter module matches the line setting value specified by this bit, an image_conv_int interrupt can be generated. Setting range: 1h to 1000h Note: By using the ICnINTCTRL.DECINTE bit, whether to generate interrupts in subsampled frames can be specified. Note: Only image processing data are supported.

(89) CRU Linear Matrix offset Register (SVC0 only) (CRUm_ICnLMXOF) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0220h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	BOF[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GOF[7:0]								ROF[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	BOF[7:0]	0h	RW	B offset value of linear matrix calculation for SVC0 (bit7: sign) Note: Setting range: -128 (80h) to 127 (7Fh) Note: Only image processing data are supported.
15 to 8	GOF[7:0]	0h	RW	G offset value of linear matrix calculation for SVC0 (bit7: sign) Note: Setting range: -128 (80h) to 127 (7Fh) Note: Only image processing data are supported.
7 to 0	ROF[7:0]	0h	RW	R offset value of linear matrix calculation for SVC0 (bit7: sign) Note: Setting range: -128 (80h) to 127 (7Fh) Note: Only image processing data are supported.

(90) CRU Linear Matrix R coefficient 1 Register (SVC0 only) (CRUm_ICnLMXRC1) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0224h
Initial Value : 0000_0400h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	RR[12:0]												
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	RR[12:0]	400h	RW	Coefficient R of R data for SVC0 linear matrix calculation (bit12: sign) Note: Specify a signed 13-bit integer by multiplying the desired coefficient value (-4 to approx. 4) by 1024 Setting range: -4096 (1000h) to 4095 (0FFFh) Note: Only image processing data are supported.

(91) CRU Linear Matrix R coefficient 2 Register (SVC0 only) (CRUm_ICnLMXRC2) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0228h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	RB[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	RG[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	RB[12:0]	0h	RW	Coefficient B of R data for SVC0 linear matrix calculation (bit12: sign) Note: Specify a signed 13-bit integer by multiplying the desired coefficient value (-4 to approx. 4) by 1024 Setting range: -4096 (1000h) to 4095 (0FFFh) Note: Only image processing data are supported.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	RG[12:0]	0h	RW	Coefficient G of R data for SVC0 linear matrix calculation (bit12: sign) Note: Specify a signed 13-bit integer by multiplying the desired coefficient value (-4 to approx. 4) by 1024 Setting range: -4096 (1000h) to 4095 (0FFFh) Note: Only image processing data are supported.

(92) CRU Linear Matrix G coefficient 1 Register (SVC0 only) (CRUm_ICnLMXGC1) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 022Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	GR[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	GR[12:0]	0h	RW	Coefficient R of G data for SVC0 linear matrix calculation (bit12: sign) Note: Specify a signed 13-bit integer by multiplying the desired coefficient value (-4 to approx. 4) by 1024 Setting range: -4096 (1000h) to 4095 (0FFFh) Note: Only image processing data are supported.

(93) CRU Linear Matrix G coefficient 2 Register (SVC0 only) (CRUm_ICnLMXGC2) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0230h
Initial Value : 0000_0400h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	GB[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	GG[12:0]												
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	GB[12:0]	0h	RW	Coefficient B of G data for SVC0 linear matrix calculation (bit12: sign) Note: Specify a signed 13-bit integer by multiplying the desired coefficient value (-4 to approx. 4) by 1024 Setting range: -4096 (1000h) to 4095 (0FFFh) Note: Only image processing data are supported.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	GG[12:0]	400h	RW	Coefficient G of G data for SVC0 linear matrix calculation (bit12: sign) Note: Specify a signed 13-bit integer by multiplying the desired coefficient value (-4 to approx. 4) by 1024 Setting range: -4096 (1000h) to 4095 (0FFFh) Note: Only image processing data are supported.

(94) CRU Linear Matrix B coefficient 1 Register (SVC0 only) (CRUm_ICnLMXBC1) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0234h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	BR[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	BR[12:0]	0h	RW	Coefficient R of B data for SVC0 linear matrix calculation (bit12: sign) Note: Specify a signed 13-bit integer by multiplying the desired coefficient value (-4 to approx. 4) by 1024 Setting range: -4096 (1000h) to 4095 (0FFFh) Note: Only image processing data are supported.

(95) CRU Linear Matrix B coefficient 2 Register (SVC0 only) (CRUm_ICnLMXBC2) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0238h

Initial Value : 0400_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	BB[12:0]												
Initial Value	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	BG[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	BB[12:0]	400h	RW	Coefficient B of B data for SVC0 linear matrix calculation (bit12: sign) Note: Specify a signed 13-bit integer by multiplying the desired coefficient value (-4 to approx. 4) by 1024 Setting range: -4096 (1000h) to 4095 (0FFFh) Note: Only image processing data are supported.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	BG[12:0]	0h	RW	Coefficient G of B data for SVC0 linear matrix calculation (bit12: sign) Note: Specify a signed 13-bit integer by multiplying the desired coefficient value (-4 to approx. 4) by 1024 Setting range: -4096 (1000h) to 4095 (0FFFh) Note: Only image processing data are supported.

(96) CRU Statistics Control 1 Register (SVC0 only) (CRUm_ICnSTIC1) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 023Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	STSADPOS[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STUNIT[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19 to 16	STSADPOS [3:0]	0h	RW	Input data bit position selection (0-8) for the sum of absolute values for adjacent pixels for statistics processing of SVC0 Use the 8 lower-order bits from the MSB-"n" bit for each type of RAW. Example: For RAW 16, use input data [15-n:8-n]. Note: Comply with the restrictions described in ICnSTIC1.STUNIT. Note: Only image processing data are supported.
15 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	STUNIT[1:0]	0h	RW	Selecting an arithmetic processing unit for SVC0 statistics processing 0: 16x16 pixels 1: 32x32 pixels 2: 64x64 pixels 3: 128x128 pixels Note: Only image processing data are supported.

(97) CRU Image Converter Register Setting Change Control Register (CRUm_ICnREGC) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0244h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	REFEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	REFEN	0h	RW1	Allows register setting changes to be reflected. By using this register for an MIPI CSI-2 input, the registers at offset addresses 01FCh to 023Ch can be changed through frame synchronization during operation. (Allow or disallow this according to **9.2.5.3 Frame Synchronization Settings**.) 0b: Does not allow register setting changes to be reflected. 1b: Allows register setting changes to be reflected. Note: If this bit is set to 1 while the Image Converter module is operating (ICnEN.ICEN = 1), the register setting values (at offset addresses 01FCh to 023Ch) are reflected at a frame start, and then the bit is cleared to 0 automatically. Note: If the Image Converter module is stopped (ICnEN.ICEN = 0), the register setting values can be reflected irrespective of this register. Note that registers other than the above cannot be changed during operation. Note: Only image processing data are supported.

(98) CRU Module Status Register (CRUm_ICnMS) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 02D8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	IA3	AV3	CA3	-	IA2	AV2	CA2	-	IA1	AV1	CA1	-	IA0	AV0	CA0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14	IA3	0h	R	Indicates whether the Image Converter block (the module converting the video data input up to the pre-FIFO-VD/SD stage) is processing data. (for SVC3) 0b: Video/statistics data have not been received. 1b: Video/statistics data are being converted.
13	AV3	0h	R	Indicates whether processing is performed within the image-clipped area. (for SVC3) (For details, see **9.2.4.3.7 Image Clipping**.) 0b: Processing is performed outside the image-clipped area. 1b: Processing is performed within the image-clipped area. Note: Only image processing data are supported.
12	CA3	0h	R	Indicates whether the frame is subsampled. (for SVC3) (For details, see **9.2.4.3.8 Frame Subsampling**.) 0b: A subsampled frame is being processed. 1b: A frame which is not subsampled is being processed. Note: This bit is 0 when CRUnRST.VRESETN is 0. This bit is set to 1 two cycles of PCLK after CRUnRST.VRESETN has been changed from 0 to 1. Note: Only image processing data are supported.
11	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
10	IA2	0h	R	Indicates whether the Image Converter block (the module converting the video data input up to the pre-FIFO-VD/SD stage) is processing data. (for SVC2) 0b: Video/statistics data have not been received. 1b: Video/statistics data are being converted.
9	AV2	0h	R	Indicates whether processing is performed within the image-clipped area. (for SVC2) (For details, see **9.2.4.3.7 Image Clipping**.) 0b: Processing is performed outside the image-clipped area. 1b: Processing is performed within the image-clipped area. Note: Only image processing data are supported.
8	CA2	0h	R	Indicates whether the frame is subsampled. (for SVC2) (For details, see **9.2.4.3.8 Frame Subsampling**.) 0b: A subsampled frame is being processed. 1b: A frame which is not subsampled is being processed. Note: This bit is 0 when CRUnRST.VRESETN is 0. This bit is set to 1 two cycles of PCLK after CRUnRST.VRESETN has been changed from 0 to 1.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6	IA1	0h	R	Indicates whether the Image Converter block (the module converting the video data input up to the pre-FIFO-VD/SD stage) is processing data. (for SVC1) 0b: Video/statistics data have not been received. 1b: Video/statistics data are being converted. Note: Only image processing data are supported.

Bit	Bit Name	Initial Value	R/W	Description
5	AV1	0h	R	Indicates whether processing is performed within the image-clipped area. (for SVC1) (For details, see **9.2.4.3.7 Image Clipping**.) 0b: Processing is performed outside the image-clipped area. 1b: Processing is performed within the image-clipped area.
4	CA1	0h	R	Indicates whether the frame is subsampled. (for SVC1) (For details, see **9.2.4.3.8 Frame Subsampling**.) 0b: A subsampled frame is being processed. 1b: A frame which is not subsampled is being processed. Note: This bit is 0 when CRUnRST.VRESETN is 0. This bit is set to 1 two cycles of PCLK after CRUnRST.VRESETN has been changed from 0 to 1.
3	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	IA0	0h	R	Indicates whether the Image Converter block (the module converting the video data input up to the pre-FIFO-VD/SD stage) is processing data. (for SVC0) 0b: Video/statistics data have not been received. 1b: Video/statistics data are being converted.
1	AV0	0h	R	Indicates whether processing is performed within the image-clipped area. (for SVC0) (For details, see **9.2.4.3.7 Image Clipping**.) 0b: Processing is performed outside the image-clipped area. 1b: Processing is performed within the image-clipped area. Note: Only image processing data are supported.
0	CA0	0h	R	Indicates whether the frame is subsampled. (for SVC0) (For details, see **9.2.4.3.8 Frame Subsampling**.) 0b: A subsampled frame is being processed. 1b: A frame which is not subsampled is being processed. Note: This bit is 0 when CRUnRST.VRESETN is 0. This bit is set to 1 two cycles of PCLK after CRUnRST.VRESETN has been changed from 0 to 1.

(99) CRU Frame Subsampling Control Register (CRUm_ICnDEC) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 02DCh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	FRMDEC[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	FRMDEC[3:0]	0h	RW	Subsampling interval in frame subsampling (For details, see **9.2.4.3.8 Frame Subsampling**.) 0h: All images are passed. 1h: One pass every two frames 2h: One pass every three frames Omitted 14h: One pass every 15 frames 15h: One pass every 16 frames Note: Image processing and non-image processing and statistics data are supported.

(100) CRU Line Count Register (SVCx) (CRUm_ICnLC_Cx) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 02E0h (x = 0)
<CRUm_base> + 02E4h (x = 1)
<CRUm_base> + 02E8h (x = 2)
<CRUm_base> + 02ECh (x = 3)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	LC[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	LC[12:0]	0h	R	Indicate the line counter value before image clipping and subsampling by the Image Converter module. (for SVCx) Note: Only image processing data are supported. Note: The maximum line value is 1000h.

(101) CRU Frame Subsampling Interrupt Control Register (CRUm_ICnINTCTRL) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0300h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DECINTE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	DECINTE	0h	RW	Specifies whether an interrupt (CRUnINTS2.FS0/1/2/3S, CRUnINTS2.FE0/1/2/3S, or CRUnINTS2.SL0/1/2/3S) is generated for a subsampled frame. 0b: No interrupt is generated for a subsampled frame. (Initial value) 1b: An interrupt is generated for a subsampled frame. Note: Image processing and non-image processing and statistics data are supported. Note: The maximum line value is 1000h.

(102) CRU Output Image Format Register (CRUm_ICnDMR) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0304h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	A8BIT[7:0]								-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	OBINSEL	-	YCMODE[2:0]			-	-	RGBMODE[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	RW	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	A8BIT[7:0]	0h	RW	Alpha bit for AXI-VD bus Alpha value in the ARGB8888 format output Note: The default value is 0h, so it is transparent. Note: This bit is valid when ICnDMR.RGBMODE[1:0]=10b,11b.
23 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	OBINSEL	0h	RW	Data Arranger 2 UV(CbCr) binary type selection for YUV422(YCbCr) data 0b: Offset binary (straight binary) 1b: Two's complement binary Note: Valid when the output is YUV422/420.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 4	YCMODE[2:0]	0h	RW	YC output data format selection for AXI-VD bus 000b: YCbCr422, 8-bit multiplexed (YUYV format (Y 1st)) 001b: YCbCr422, 8-bit multiplexed (UYVY format (Cb 1st)) 010b: YC separation from YCbCr422 011b: Y component extraction from YCbCr422 100b: YCbCr420, 8-bit multiplexed (YUYV format (Y 1st)) 101b: YCbCr420, 8-bit multiplexed (UYVY format (Cb 1st)) 110b : YC Separation of YCbCr420 *The CbCr(UV) side is Image Stride (AMnIS) ×2. 111b : Y component extraction of YCbCr420 (equivalent to YUV422)
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	RGBMODE[1:0]	0h	RW	RGB output data format selection for AXI-VD bus 00b: RGB888 (24-bit) 01b: RGB888 (32-bit) (The eight higher-order bits are padded with 00h.) 10b: ARGB888 (32-bit, B First A Last) 11b: ARGB888 (32-bit, A First B Last)

(103) CRU YCbCr → RGB Color Space Conversion Coefficient 1 Register (CRUm_ICnCSCC1) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0308h
Initial Value : 0000_129Fh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	YMUL[13:0]													
Initial Value	0	0	0	1	0	0	1	0	1	0	0	1	1	1	1	1
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 0	YMUL[13:0]	129Fh	RW	Y data multiplication coefficient These bits specify a Y data multiplication coefficient in YCbCr → RGB color space conversion. (Initial value: 1.164) Specify an unsigned 14-bit integer which is the desired coefficient value multiplied by 4096.

(104) CRU YCbCr → RGB Color Space Conversion Coefficient 2 Register (CRUm_ICnCSCC2) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 030Ch
Initial Value : 0100_0800h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	YSUB[11:0]											
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	CSUB[11:0]											
Initial Value	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27 to 16	YSUB[11:0]	100h	RW	Y data subtraction coefficient These bits specify a Y data subtraction coefficient in YCbCr → RGB color space conversion. Specify an unsigned 12-bit integer. Calculate on 12 bits extended from YCbCr422 8-/10-bit.
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	CSUB[11:0]	800h	RW	CbCr data subtraction coefficient These bits specify a Cb and Cr data subtraction coefficient in YCbCr → RGB color space conversion. Specify an unsigned 12-bit integer. Calculate on 12 bits extended from YCbCr422 8-/10-bit.

(105) CRU YCbCr → RGB Color Space Conversion Coefficient 3 Register (CRUm_ICnCSCC3) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0310h
Initial Value : 1989_0D02h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	RCRMUL[13:0]													
Initial Value	0	0	0	1	1	0	0	1	1	0	0	0	1	0	0	1
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	GCRMUL[13:0]													
Initial Value	0	0	0	0	1	1	0	1	0	0	0	0	0	0	1	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29 to 16	RCRMUL[13:0]	1989h	RW	Cr multiplication coefficient in R data calculation These bits specify a Cr multiplication coefficient for the R data equation in YCbCr → RGB color space conversion. (Initial value: 1.596) Specify an unsigned 14-bit integer which is the desired coefficient value multiplied by 4096.
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 0	GCRMUL[13:0]	0D02h	RW	Cr multiplication coefficient in G data calculation These bits specify a Cr multiplication coefficient for the G data equation in YCbCr → RGB color space conversion. (Initial value: 0.813) Specify an unsigned 14-bit integer which is the desired coefficient value multiplied by 4096.

(106) CRU YCbCr → RGB Color Space Conversion Coefficient 4 Register (CRUm_ICnCSCC4) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0314h
Initial Value : 0645_2045h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	GCBMUL[13:0]													
Initial Value	0	0	0	0	0	1	1	0	0	1	0	0	0	1	0	1
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	BCBMUL[13:0]													
Initial Value	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29 to 16	GCBMUL[13:0]	0645h	RW	Cb multiplication coefficient in G data calculation These bits specify a Cb multiplication coefficient for the G data equation in YCbCr → RGB color space conversion. (Initial value: 0.392) Specify an unsigned 14-bit integer which is the desired coefficient value multiplied by 4096.
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 0	BCBMUL[13:0]	2045h	RW	Cb multiplication coefficient in B data calculation These bits specify a Cb multiplication coefficient for the B data equation in YCbCr → RGB color space conversion. (Initial value: 2.017) Specify an unsigned 14-bit integer which is the desired coefficient value multiplied by 4096.

(107) CRU RGB → YCbCr Color Space Conversion Y Coefficient 1 Register (CRUm_ICnYCCR1) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0318h

Initial Value : 0000_0107h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	YCLRP[12:0]												
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	YCLRP[12:0]	0107h	RW	R multiplication coefficient in Y calculation (R multiplication coefficient for the Y data equation in RGB → YCbCr color space conversion)

(108) CRU RGB → YCbCr Color Space Conversion Y Coefficient 2 Register (CRUm_ICnYCCR2) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 031Ch

Initial Value : 0064_0204h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	YCLBP[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	YCLGP[12:0]												
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	YCLBP[12:0]	0064h	RW	B multiplication coefficient in Y calculation (B multiplication coefficient for the Y data equation in RGB → YCbCr color space conversion)
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	YCLGP[12:0]	0204h	RW	G multiplication coefficient in Y calculation (G multiplication coefficient for the Y data equation in RGB → YCbCr color space conversion)

(109) CRU RGB → YCbCr Color Space Conversion Y Coefficient 3 Register (CRUm_ICnYCCR3) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0320h

Initial Value : 0A00_0100h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	YCLSFT[4:0]					-	-	-	-	-	-	-	YCLCEN
Initial Value	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	YCLAP[11:0]											
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.
30, 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 24	YCLSFT[4:0]	0Ah	RW	Amount of downward shift in Y calculation (specification of the amount of downward shift for Y calculation in RGB → YCbCr color space conversion)
23 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	YCLCEN	0h	RW	Y Calculation Data Clip Enable (Data Clipping Process Control for Y Data Calculation in RGB → YCbCr Color Space Conversion) 0b: Data clipping is not performed 1b: Data clipping is performed
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	YCLAP[11:0]	100h	RW	Value added for normalization of Y calculation data (specification of Y data addition constant in RGB → YCbCr color space conversion)

(110) CRU RGB → YCbCr Color Space Conversion Cb Coefficient 1 Register (CRUm_ICnCBCCR1) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0324h

Initial Value : 0000_1F68h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	CBCLRP[12:0]												
Initial Value	0	0	0	1	1	1	1	1	0	1	1	0	1	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	CBCLRP[12:0]	1F68h	RW	R multiplication coefficient in Cb calculation (R multiplication coefficient for the Cb data equation in RGB → YCbCr color space conversion)

(111) CRU RGB → YCbCr Color Space Conversion Cb Coefficient 2 Register (CRUm_ICnCBCCR2) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0328h

Initial Value : 01C2_1ED6h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	CBCLBP[12:0]												
Initial Value	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	CBCLGP[12:0]												
Initial Value	0	0	0	1	1	1	1	0	1	1	0	1	0	1	1	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	CBCLBP[12:0]	01C2h	RW	B multiplication coefficient in Cb calculation (B multiplication coefficient for the Cb data equation in RGB → YCbCr color space conversion)
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	CBCLGP[12:0]	1ED6h	RW	G multiplication coefficient in Cb calculation (G multiplication coefficient for the Cb data equation in RGB → YCbCr color space conversion)

(112) CRU RGB → YCbCr Color Space Conversion Cb Coefficient 3 Register (CRUm_ICnCBCCR3) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 032Ch
Initial Value : 0A00_0800h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	CBCLSFT[4:0]					-	-	-	-	-	-	-	CBCLCEN
Initial Value	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	CBCLAP[11:0]											
Initial Value	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.
30, 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 24	CBCLSFT[4:0]	0Ah	RW	Amount of downward shift in Cb calculation (specification of the amount of downward shift for Cb calculation in RGB → YCbCr color space conversion)
23 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	CBCLCEN	0h	RW	Cb Calculation Data Clip Enable (Data Clipping Process Control for Cb Data Calculation in RGB → YCbCr Color Space Conversion) 0: Data clipping is not performed 1: Data clipping is performed
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	CBCLAP[11:0]	800h	RW	Value added for normalization of Cb calculation data (specification of Cb data addition constant in RGB → YCbCr color space conversion)

(113) CRU RGB → YCbCr Color Space Conversion Cr Coefficient 1 Register (CRUm_ICnCRCCR1) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0330h

Initial Value : 0000_01C2h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	CRCLRP[12:0]												
Initial Value	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	CRCLRP[12:0]	01C2h	RW	R multiplication coefficient in Cr calculation (R multiplication coefficient for the Cr data equation in RGB → YCbCr color space conversion)

(114) CRU RGB → YCbCr Color Space Conversion Cr Coefficient 2 Register (CRUm_ICnCRCCR2) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0334h

Initial Value : 1FB7_1E87h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	CRCLBP[12:0]												
Initial Value	0	0	0	1	1	1	1	1	1	0	1	1	0	1	1	1
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	CRCLGP[12:0]												
Initial Value	0	0	0	1	1	1	1	0	1	0	0	0	0	1	1	1
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	CRCLBP[12:0]	1FB7h	RW	B multiplication coefficient in Cr calculation (B multiplication coefficient for the Cr data equation in RGB → YCbCr color space conversion)
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	CRCLGP[12:0]	1E87h	RW	G multiplication coefficient in Cr calculation (G multiplication coefficient for the Cr data equation in RGB → YCbCr color space conversion)

(115) CRU RGB → YCbCr Color Space Conversion Cr Coefficient 3 Register (CRUm_ICnCRCCR3) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0338h

Initial Value : 0A00_0800h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	CRCLSFT[4:0]					-	-	-	-	-	-	-	CRCLCEN
Initial Value	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	CRCLAP[11:0]											
Initial Value	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	RW	Reserved A register that can be read and written. Do not change the value of this bit from the initial value.
30, 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 24	CRCLSFT[4:0]	0Ah	RW	Amount of downward shift in Cr calculation (specification of the amount of downward shift for Cr calculation in RGB → YCbCr color space conversion)
23 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	CRCLCEN	0h	RW	Cr Calculation Data Clip Enable (Data Clipping Process Control for Cr Data Calculation in RGB → YCbCr Color Space Conversion) 0: Data clipping is not performed 1: Data clipping is performed
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	CRCLAP[11:0]	800h	RW	Value added for normalization of Cr calculation data (specification of Cr data addition constant in RGB → YCbCr color space conversion)

(116) CRU Lookup Table Control Register (CRUm_ICnLUT) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 034Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LUTWR	LUTSEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	LUTWR	0h	RW	Enables APB read/write access to the lookup table (LUT). 0b: Disables APB access (set this bit to 0 in operation with the normal path). 1b: Enables APB access (normal access is unavailable in this case). Note: See **Figure 9.2-57 Storing LUT Data** for the LUT setting procedure.
0	LUTSEL	0h	RW	Specifies the access path of the lookup table (LUT). 0b: Normal path (image conversion by using the LUT) 1b: APB I/F (read/write from registers) Used to write (and confirm) data to the LUT in initial setting. Set this bit to 0 after finishing writing and reading data. Note that the LUT data is retained even after a reset. Note: See **Figure 9.2-57 Storing LUT Data** for the LUT setting procedure.

(117) CRU Lookup Table Status Register (CRUm_ICnLUTS) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0350h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LUTREGSEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	LUTREGSEL	0h	R	Indicates that the lookup table (LUT)is writable and readable from the APB. The LUTWR register value is reflected in this register after a while. Access the lookup table after confirming this bit. 0b: Neither writable nor readable. 1b: Writable and readable. Note: See **Figure 9.2-57 Storing LUT Data** for the LUT setting procedure.

(118) CRU Lookup Table Pointer Register (CRUm_ICnLUTP) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0354h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	LTCRPR[9:0]										LTCBPR[9:6]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTCBPR[5:0]						LTYPR[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29 to 20	LTCRPR[9:0]	0h	RW	Lookup table (LUT) Cr/B pointer Specify the current pointer (access point) to the lookup table Cr/B in APB access. The pointer is incremented automatically when ICnLUTD.LTCRDT[7:0] are written. It returns to 0 upon an overflow. It is not incremented when reading these bits. Note: See **Figure 9.2-57 Storing LUT Data** for the LUT setting procedure.
19 to 10	LTCBPR[9:0]	0h	RW	Lookup table (LUT) Cb/G pointer Specify the current pointer (access point) to lookup table Cb/G in APB access. The pointer is incremented automatically when ICnLUTD.LTCBDT[7:0] are written. It returns to 0 upon an overflow. It is not incremented when reading these bits. Note: See **Figure 9.2-57 Storing LUT Data** for the LUT setting procedure.
9 to 0	LTYPR[9:0]	0h	RW	Lookup table (LUT) Y/R pointer Specify the current pointer (access point) to lookup table Y/R in APB access. The pointer is incremented automatically when ICnLUTD.LTYDT[7:0] are written. It returns to 0 upon an overflow. It is not incremented when reading these bits. Note: See **Figure 9.2-57 Storing LUT Data** for the LUT setting procedure.

(119) CRU Lookup Table Data Register (CRUm_ICnLUTD) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0358h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	LTCRDT[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTCBDT[7:0]								LTYDT[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	LTCRDT[7:0]	0h	RW	Lookup table (LUT) Cr/B data Read and write data to the location indicated by the ICnLUTP.LTCRPR[9:0] pointer in lookup table Cr/B through the APB interface.
15 to 8	LTCBDT[7:0]	0h	RW	Lookup table (LUT) Cb/G data Read and write data to the location indicated by the ICnLUTP.LTCBPR[9:0] pointer in lookup table Cb/G through the APB interface.
7 to 0	LTYDT[7:0]	0h	RW	Lookup table (LUT) Y/R data Read and write data to the location indicated by the ICnLUTP.LTYPR[9:0] pointer in lookup table Y/R through the APB interface.

(120) CRU Test Image Generation Control 1 Register (CRUm_ICnTICTRL1) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 035Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	TIRATE[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIPTNV1[3:0]				TIPTNU1[3:0]				TIPTNY1[3:0]				-	TIROTSEL	TIMODE	TIEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
20 to 16	TIRATE[4:0]	0h	RW	Specify the test image generation rate. 16 pixels / (18 + TIRATE) cycles It can be used in the following ranges. 1) When ICnDMR.YCMODE = 010b or 110b or ICnDMR.RGBMODE = 00b 3.291 Gbps(ICnTICTRL1.TTIRATE[4:0] = Fh) to 8.488 Gbps(ICnTICTRL1.TTIRATE[4:0] = 1h) 2) When ICnDMR.YCMODE ≠ 010b, ICnDMR.YCMODE ≠ 110b and ICnDMR.RGBMODE ≠ 00b 3.291 Gbps(ICnTICTRL1.TTIRATE[4:0] = Fh) to 8.960 Gbps(ICnTICTRL1.TTIRATE[4:0] = 0h)
15 to 12	TIPTNV1[3:0]	0h	RW	Specify the pixel (V) counter bits for the test image pattern. 0000b: Selects bits [3:0]. 0001b: Selects bits [4:1]. 0010b: Selects bits [5:2]. Omitted 1100b: Selects bits [15:12]. 1101b to 1111b: Setting prohibited
11 to 8	TIPTNU1[3:0]	0h	RW	Specify the pixel (U) counter bits for the test image pattern. 0000b: Selects bits [3:0]. 0001b: Selects bits [4:1]. 0010b: Selects bits [5:2]. Omitted 1100b: Selects bits [15:12]. 1101b to 1111b: Setting prohibited
7 to 4	TIPTNY1[3:0]	0h	RW	Specify the pixel (Y) counter bits for the test image pattern. 0000b: Selects bits [3:0]. 0001b: Selects bits [4:1]. 0010b: Selects bits [5:2]. Omitted 1100b: Selects bits [15:12]. 1101b to 1111b: Setting prohibited
3	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.
2	TIROTSEL	0h	RW	Image processing/non-image processing route selection for test image output 0b: Input for image processing 1b: Input for non-image processing
1	TIMODE	0h	RW	Specifies a test image generation pattern. 0b: Selects a counter value for YUV output. 1b: Same-color YUV output for the entire screen

Bit	Bit Name	Initial Value	R/W	Description
0	TIEN	0h	RW	Enables generation of a test image pattern. This test image works with VC0 only. 0b: Does not generate a test image pattern (processes the data type from the LINK). 1b: Generates a test image pattern (process test image pattern data). Note: When this bit is set to 1b, test data is forcibly selected regardless of CRUnCTRL.VINSEL.

(121) CRU Test Image Generation Control 2 Register (CRUm_ICnTICTRL2) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0360h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	TIPTNV2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TIPTNU2[7:0]								TIPTNY2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	TIPTNV2[7:0]	0h	RW	Specify V for the entire-screen same-color test image pattern. Note: This bit is valid when ICnTICTRL1.TIMODE=1b.
15 to 8	TIPTNU2[7:0]	0h	RW	Specify U for the entire-screen same-color test image pattern. Note: This bit is valid when ICnTICTRL1.TIMODE=1b.
7 to 0	TIPTNY2[7:0]	0h	RW	Specify Y for the entire-screen same-color test image pattern. Note: This bit is valid when ICnTICTRL1.TIMODE=1b.

(122) CRU Test Image Size Setting 1 Register (CRUm_ICnTISIZE1) (m = 0)

Access Size : 32 bits

Address : <CRUm_base> + 0364h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	TIPPL[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	TIPPL[11:0]	0h	RW	Number of valid pixels per line in the test image pattern Note 1. If a value from 0 to 15 is specified, operation is for the setting of 16. Note 2. The four lower-order bits are invalid.

(123) CRU Test Image Size Setting 2 Register (CRUm_ICnTISIZE2) (m = 0)

Access Size : 32 bits
Address : <CRUm_base> + 0368h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	TIM[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	TIN[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27 to 16	TIM[11:0]	0h	RW	Invalid period (number of lines) per frame in the test image pattern Note: If 0 is specified, operation is for the setting of 1.
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	TIN[11:0]	0h	RW	Valid period (number of lines) per frame in the test image pattern Note: If 0 is specified, operation is for the setting of 1.

9.2.4.3 Function Description

The prefix “CRUm (m = 0)” of the register names is omitted in the following subsections.

9.2.4.3.1 Input Formats and Image Processing

Table 9.2-10, **Table 9.2-11**, and **Table 9.2-12** list the input formats in which images can be processed by this module and the corresponding available forms of image processing.

Table 9.2-10 Input Formats and Corresponding Forms of Image Processing (for Image Processing) (1/3)

| Input Format | | | Processing Indicator | | | | | | | | | | | | | | | | Interrupt | | | Output Format | | | | | | | | | | | | | | | |

Table 9.2-10 Input Formats and Corresponding Forms of Image Processing (for Image Processing) (2/3)

| Input Format | | | Processing Indicator | | | | | | | | | | | | | | | | | | Interrupt | | | Output Format | | | | | | | | | | | | | | | | | |

Table 9.2-10 Input Formats and Corresponding Forms of Image Processing (for Image Processing) (3/3)

Input Format			Processing Indicator																Interrupt			Output Format																				
					COMMON					RAW			YUV					RGB				YCbCr 422			YCb Cr 420		RGB			RAW												
Data Type	Data Type Classes	Data Type Name	Full Bypass (ICnCTRL.ATH = 1)	8-bit Expansion (RGB888, RAW8)	Header Appending	Footer Appending	Frame Subsampling	Image Clipping	LUT	Statistics	Demosaicing	Linear Matrix	Data Arranger 1	Data Clipping	YCbCr to RGB	YCbCr422 to 420	Data Arranger 2	RGB to YCbCr	Frame Start/End, Scan line	ERROR (DECERR/SLVERR/FIFO Overflow)	AXI Specified Address Forwarding Completed	8-bit multiplexed	YC separation	Y component extraction	8-bit multiplexed	YC separation	RGB888 (24 bits)	RGB888 (32 bits)	ARGB888 (32 bits)	RAW6	RAW7	RAW8	RAW10	RAW12	RAW14	RAW16	RAW20	MIPI Recommended Memory Storage Format
30h	User defined byte-based data	User defined 8-bit data type 1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
31h		User defined 8-bit data type 2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
32h		User defined 8-bit data type 3	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
33h		User defined 8-bit data type 4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
34h		User defined 8-bit data type 5	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
35h		User defined 8-bit data type 6	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
36h		User defined 8-bit data type 7	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
37h		User defined 8-bit data type 8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
38h to 3Fh	Reserved	Reserved	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Pattern generator (ICnTICTRL1.TIEN register)																																							
1b	Test pattern	YCbCr422 8-bit	—	—	—	✓	✓	✓	✓	—	—	—	✓	✓	✓	✓	✓	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—

Note: ✓: Supports SVC0 to SVC3
⊙: Only SVC0 is supported
—: Not supported

Table 9.2-11 Input Formats and Corresponding Forms of Image Processing (for Non-Image Processing) (1/3)

Input Format			Processing Indicator																	Interrupt			Output Format																				
			COMMON						RAW			YUV						RGB				YCbCr 422			YCb Cr 420		RGB			RAW													
Data Type	Data Type Classes	Data Type Name	Full Bypass (ICnCTRL.ATH = 1)	8-bit Expansion (RGB888, RAW8)	Header Appending	Footer Appending	Frame Subsampling	Image Clipping	LUT	Statistics	Demosaicing	Linear Matrix	Data Arranger 1	Data Clipping	YCbCr to RGB	YCbCr422 to 420	Data Arranger 2	RGB to YCbCr	Frame Start/End, Scan line	ERROR (DECERR/SLVERR/FIFO Overflow)	AXI Specified Address Forwarding Completed	8-bit multiplexed	YC separation	Y component extraction	8-bit multiplexed	YC separation	RGB888 (24 bits)	RGB888 (32 bits)	ARGB888 (32 bits)	RAW6	RAW7	RAW8	RAW10	RAW12	RAW14	RAW16	RAW20	MIPI Recommended Memory Storage Format
Input settings ({ICnNIPDT_C0/1/2/3H.NIPDT_C0/1/2/3H[63:32],ICnNIPDT_C0/1/2/3L.NIPDT_C0/1/2/3L[31:16]} register)																																									
00h	Synchronization short packet data types	Frame start code	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
01h		Frame end code	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
02h		Line start code	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
03h		Line end code	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
04h to 07h		—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
08h	Generic short packet data types	Generic short packet code 1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
09h		Generic short packet code 2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0Ah		Generic short packet code 3	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0Bh		Generic short packet code 4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0Ch		Generic short packet code 5	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0Dh		Generic short packet code 6	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0Eh		Generic short packet code 7	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
0Fh		Generic short packet code 8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
10h	Generic long packet data types	Null	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
11h		Blanking data	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
12h		Embedded 8-bit non image data	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
13h		Generic long packet data type 1	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
14h		Generic long packet data type 2	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
15h		Generic long packet data type 3	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
16h		Generic long packet data type 4	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
17h		Reserved	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Table 9.2-11 Input Formats and Corresponding Forms of Image Processing (for Non-Image Processing) (2/3)

Input Format			Processing Indicator																Interrupt		Output Format																				
					COMMON					RAW			YUV						RGB				YCbCr 422			YCb Cr 420		RGB			RAW										MIPI Recommended Memory Storage Format
Data Type	Data Type Classes	Data Type Name	Full Bypass (ICnCTRL.ATH = 1)	8-bit Expansion (RGB888, RAW8)	Header Appending	Footer Appending	Frame Subsampling	Image Clipping	LUT	Statistics	Demosaicing	Linear Matrix	Data Arranger 1	Data Clipping	YCbCr to RGB	YCbCr422 to 420	Data Arranger 2	RGB to YCbCr	Frame Start/End, Scan line	ERROR (DECERR/SLVERR/FIFO Overflow)	AXI Specified Address Forwarding Completed	8-bit multiplexed	YC separation	Y component extraction	8-bit multiplexed	YC separation	RGB888 (24 bits)	RGB888 (32 bits)	ARGB888 (32 bits)	RAW6	RAW7	RAW8	RAW10	RAW12	RAW14	RAW16	RAW20	
18h	YUV data	YUV420 8-bit	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
19h		YUV420 10-bit	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
1Ah		Legacy YUV420 8-bit	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
1Bh		Reserved	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
1Ch		YUV420 8-bit (chroma shifted pixel sampling)	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
1Dh		YUV420 10-bit (chroma shifted pixel sampling)	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
1Eh		YUV422 8-bit	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
1Fh		YUV422 10-bit	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
20h	RGB data	RGB444	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
21h		RGB555	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
22h		RGB565	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
23h		RGB666	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
24h		RGB888	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
25h to 27h		Reserved	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
28h	RAW data	RAW6	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
29h		RAW7	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
2Ah		RAW8	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
2Bh		RAW10	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
2Ch		RAW12	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
2Dh		RAW14	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
2Eh		RAW16	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
2Fh		RAW20	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
30h	User defined byte-based data	User defined 8-bit data type 1	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
31h		User defined 8-bit data type 2	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
32h		User defined 8-bit data type 3	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
33h		User defined 8-bit data type 4	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
34h		User defined 8-bit data type 5	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
35h		User defined 8-bit data type 6	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
36h		User defined 8-bit data type 7	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
37h		User defined 8-bit data type 8	✓	—	✓	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓

Table 9.2-11 Input Formats and Corresponding Forms of Image Processing (for Non-Image Processing) (3/3)

Input Format			Processing Indicator																Interrupt			Output Format																					
					COMMON					RAW			YUV						RGB				YCbCr 422			YCb Cr 420		RGB			RAW												
Data Type	Data Type Classes	Data Type Name	Full Bypass (ICnCTRL.ATH = 1)	8-bit Expansion (RGB888, RAW8)	Header Appending	Footer Appending	Frame Subsampling	Image Clipping	LUT	Statistics	Demosaicing	Linear Matrix	Data Arranger 1	Data Clipping	YCbCr to RGB	YCbCr422 to 420	Data Arranger 2	RGB to YCbCr	Frame Start/End, Scan line	ERROR (DECERR/SLVERR/FIFO Overflow)	AXI Specified Address Forwarding Completed	8-bit multiplexed	YC separation	Y component extraction	8-bit multiplexed	YC separation	RGB888 (24 bits)	RGB888 (32 bits)	ARGB888 (32 bits)	RAW6	RAW7	RAW8	RAW10	RAW12	RAW14	RAW16	RAW20	MIPI Recommended Memory Storage Format
38h to 3Fh	Reserved	Reserved	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Pattern generator (ICnTICTRL1.TIEN register)																																									
1b	Test pattern	YCbCr422 8-bit	✓	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	✓	✓	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	✓

Note: ✓: Supports SVC0 to SVC3
—: Not supported

Table 9.2-12 Input Formats and Corresponding Forms of Image Processing (for Statistics)

Input Format			Full Bypass (ICnCTRL.ATH = 1)	Processing Indicator															Interrupt			Output Format
				COMMON					RAW			YUV					RGB					
Data Type	Data Type Classes	Data Type Name		8-bit Expansion (RGB888,RAW8)	Header Appending	Footer Appending	Frame Subsampling	Image Clipping	LUT	Statistics	Demosaicing	Linear Matrix	Data Arranger 1	Data Clipping	YCbCr to RGB	YCbCr422 to 420	Data Arranger 2	RGB to YCbCr	Frame Start/End, Scan line	ERROR (DECERR/ SLVERR/FIFO Overflow)	AXI Specified Address Forwarding Completed	Statistics Data
Input settings (ICnIPMC_C0/1/2/3.INF[5:0] register)																						
00h to 27h		Not supports	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
28h	RAW data	RAW6	—	⊙	—	⊙	⊙	⊙	—	⊙	—	—	—	—	—	—	—	—	⊙	⊙	⊙	⊙
29h		RAW7	—	⊙	—	⊙	⊙	⊙	—	⊙	—	—	—	—	—	—	—	—	⊙	⊙	⊙	⊙
2Ah		RAW8	—	—	—	⊙	⊙	⊙	—	⊙	—	—	—	—	—	—	—	—	⊙	⊙	⊙	⊙
2Bh		RAW10	—	—	—	⊙	⊙	⊙	—	⊙	—	—	—	—	—	—	—	—	⊙	⊙	⊙	⊙
2Ch		RAW12	—	—	—	⊙	⊙	⊙	—	⊙	—	—	—	—	—	—	—	—	⊙	⊙	⊙	⊙
2Dh		RAW14	—	—	—	⊙	⊙	⊙	—	⊙	—	—	—	—	—	—	—	—	⊙	⊙	⊙	⊙
2Eh		RAW16	—	—	—	⊙	⊙	⊙	—	⊙	—	—	—	—	—	—	—	—	⊙	⊙	⊙	⊙
2Fh		RAW20	—	—	—	⊙	⊙	⊙	—	⊙	—	—	—	—	—	—	—	—	⊙	⊙	⊙	⊙
30h to 3Fh		Not supports	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Pattern generator (ICnTICTRL1.TIEN register)																						
1b	Test pattern	YCbCr422 8-bit	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Note: ✓: Supports SVC0 to SVC3
⊙: Only SVC0 is supported
—: Not supported

9.2.4.3.2 Output Formats

Table 9.2-13 and **Table 9.2-14** list the video output formats and the statistics output format, respectively.

Table 9.2-13 Video Output Formats (1) (1/2)

Format		Line*4	AMnUVAOFH/L address offset*5	Registers	Data[255:0]															
					byte 31	byte 30	byte 29	byte 28	byte 27	byte 26	byte 25	byte 24	byte 23	byte 22	byte 21	byte 20	byte 19	byte 18	byte 17	byte 16
YUV (YCbCr) 422	Cb 1st	Both	Non-use	ICnDMR.YCMODE = 001b CRUnCTRL.ATH = 0	Y15 [7:0]	Cr14 [7:0]	Y14 [7:0]	Cb14 [7:0]	Y13 [7:0]	Cr12 [7:0]	Y12 [7:0]	Cb12 [7:0]	Y11 [7:0]	Cr10 [7:0]	Y10 [7:0]	Cb10 [7:0]	Y9 [7:0]	Cr8 [7:0]	Y8 [7:0]	Cb8 [7:0]
	Y 1st	Both	Non-use	ICnDMR.YCMODE = 000b CRUnCTRL.ATH = 0	Cr14 [7:0]	Y15 [7:0]	Cb14 [7:0]	Y14 [7:0]	Cr12 [7:0]	Y13 [7:0]	Cb12 [7:0]	Y12 [7:0]	Cr10 [7:0]	Y11 [7:0]	Cb10 [7:0]	Y10 [7:0]	Cr8 [7:0]	Y9 [7:0]	Cb8 [7:0]	Y8 [7:0]
	YC separation	Both	Non-use	ICnDMR.YCMODE = 010b CRUnCTRL.ATH = 0	Y31 [7:0]	Y30 [7:0]	Y29 [7:0]	Y28 [7:0]	Y27 [7:0]	Y26 [7:0]	Y25 [7:0]	Y24 [7:0]	Y23 [7:0]	Y22 [7:0]	Y21 [7:0]	Y20 [7:0]	Y19 [7:0]	Y18 [7:0]	Y17 [7:0]	Y16 [7:0]
		Both	Use		Cr30 [7:0]	Cb30 [7:0]	Cr28 [7:0]	Cb28 [7:0]	Cr26 [7:0]	Cb26 [7:0]	Cr24 [7:0]	Cb24 [7:0]	Cr22 [7:0]	Cb22 [7:0]	Cr20 [7:0]	Cb20 [7:0]	Cr18 [7:0]	Cb18 [7:0]	Cr16 [7:0]	Cb16 [7:0]
	Y component extraction	Both	Non-use	ICnDMR.YCMODE = 011b CRUnCTRL.ATH = 0	Y31 [7:0]	Y30 [7:0]	Y29 [7:0]	Y28 [7:0]	Y27 [7:0]	Y26 [7:0]	Y25 [7:0]	Y24 [7:0]	Y23 [7:0]	Y22 [7:0]	Y21 [7:0]	Y20 [7:0]	Y19 [7:0]	Y18 [7:0]	Y17 [7:0]	Y16 [7:0]
YUV (YCbCr) 420	Cb 1st	Odd	Non-use	ICnDMR.YCMODE = 101b CRUnCTRL.ATH = 0	Y15 [7:0]	Cr14 [7:0]	Y14 [7:0]	Cb14 [7:0]	Y13 [7:0]	Cr12 [7:0]	Y12 [7:0]	Cb12 [7:0]	Y11 [7:0]	Cr10 [7:0]	Y10 [7:0]	Cb10 [7:0]	Y9 [7:0]	Cr8 [7:0]	Y8 [7:0]	Cb8 [7:0]
		Even	Non-use		Y31 [7:0]	Y30 [7:0]	Y29 [7:0]	Y28 [7:0]	Y27 [7:0]	Y26 [7:0]	Y25 [7:0]	Y24 [7:0]	Y23 [7:0]	Y22 [7:0]	Y21 [7:0]	Y20 [7:0]	Y19 [7:0]	Y18 [7:0]	Y17 [7:0]	Y16 [7:0]
	Y 1st	Odd	Non-use	ICnDMR.YCMODE = 100b/111b CRUnCTRL.ATH = 0	Cr14 [7:0]	Y15 [7:0]	Cb14 [7:0]	Y14 [7:0]	Cr12 [7:0]	Y13 [7:0]	Cb12 [7:0]	Y12 [7:0]	Cr10 [7:0]	Y11 [7:0]	Cb10 [7:0]	Y10 [7:0]	Cr8 [7:0]	Y9 [7:0]	Cb8 [7:0]	Y8 [7:0]
		Even	Non-use		Y31 [7:0]	Y30 [7:0]	Y29 [7:0]	Y28 [7:0]	Y27 [7:0]	Y26 [7:0]	Y25 [7:0]	Y24 [7:0]	Y23 [7:0]	Y22 [7:0]	Y21 [7:0]	Y20 [7:0]	Y19 [7:0]	Y18 [7:0]	Y17 [7:0]	Y16 [7:0]
	YC separation*3	Both	Non-use	ICnDMR.YCMODE = 110b CRUnCTRL.ATH = 0	Y31 [7:0]	Y30 [7:0]	Y29 [7:0]	Y28 [7:0]	Y27 [7:0]	Y26 [7:0]	Y25 [7:0]	Y24 [7:0]	Y23 [7:0]	Y22 [7:0]	Y21 [7:0]	Y20 [7:0]	Y19 [7:0]	Y18 [7:0]	Y17 [7:0]	Y16 [7:0]
		Both	Use		Cr30 [7:0]	Cb30 [7:0]	Cr28 [7:0]	Cb28 [7:0]	Cr26 [7:0]	Cb26 [7:0]	Cr24 [7:0]	Cb24 [7:0]	Cr22 [7:0]	Cb22 [7:0]	Cr20 [7:0]	Cb20 [7:0]	Cr18 [7:0]	Cb18 [7:0]	Cr16 [7:0]	Cb16 [7:0]
RGB-888	24 bit/pixel	Both	Non-use	ICnDMR.RGBMODE = 00b CRUnCTRL.ATH=0	G10 [7:0]	B10 [7:0]	R9 [7:0]	G9 [7:0]	B9 [7:0]	R8 [7:0]	G8 [7:0]	B8 [7:0]	R7 [7:0]	G7 [7:0]	B7 [7:0]	R6 [7:0]	G6 [7:0]	B6 [7:0]	R5 [7:0]	G5 [7:0]
	32 bit/pixel	Both	Non-use	ICnDMR.RGBMODE = 01b CRUnCTRL.ATH = 0	00h	R7 [7:0]	G7 [7:0]	B7 [7:0]	00h	R6 [7:0]	G6 [7:0]	B6 [7:0]	00h	R5 [7:0]	G5 [7:0]	B5 [7:0]	00h	R4 [7:0]	G4 [7:0]	B4 [7:0]
ARGB-8888	B 1st	Both	Non-use	ICnDMR.RGBMODE = 10b CRUnCTRL.ATH = 0	A7 [7:0]	R7 [7:0]	G7 [7:0]	B7 [7:0]	A8 [7:0]	R6 [7:0]	G6 [7:0]	B6 [7:0]	A5 [7:0]	R5 [7:0]	G5 [7:0]	B5 [7:0]	A4 [7:0]	R4 [7:0]	G4 [7:0]	B4 [7:0]
	A 1st	Both	Non-use	ICnDMR.RGBMODE = 11b CRUnCTRL.ATH = 0	B7 [7:0]	G7 [7:0]	R7 [7:0]	A7 [7:0]	B6 [7:0]	G6 [7:0]	R6 [7:0]	A6 [7:0]	B5 [7:0]	G5 [7:0]	R5 [7:0]	A5 [7:0]	B4 [7:0]	G4 [7:0]	R4 [7:0]	A4 [7:0]
RAW	RAW6	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 28h ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	{Xh, P39[5:0], P38[5:0], P37[5:0], P36[5:0], P35[5:0], P34[5:0], P33[5:0], P32[5:0], P31[5:0], P30[5:0]}								{Xh, P29[5:0], P28[5:0], P27[5:0], P26[5:0], P25[5:0], P24[5:0], P23[5:0], P22[5:0], P21[5:0], P20[5:0]}							
	RAW7	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 29h ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	{Xh, P35[6:0], P34[6:0], P33[6:0], P32[6:0], P31[6:0], P30[6:0], P29[6:0], P28[6:0], P27[6:0]}								{Xh, P26[6:0], P25[6:0], P24[6:0], P23[6:0], P22[6:0], P21[6:0], P20[6:0], P19[6:0], P18[6:0]}							
	RAW8	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 2Ah ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	P31 [7:0]	P30 [7:0]	P29 [7:0]	P28 [7:0]	P27 [7:0]	P26 [7:0]	P25 [7:0]	P24 [7:0]	P23 [7:0]	P22 [7:0]	P21 [7:0]	P20 [7:0]	P19 [7:0]	P18 [7:0]	P17 [7:0]	P16 [7:0]
	RAW10	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 2Bh ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	{Xh, P23[9:0], P22[9:0], P21[9:0], P20[9:0], P19[9:0], P18[9:0]}								{Xh, P17[9:0], P16[9:0], P15[9:0], P14[9:0], P13[9:0], P12[9:0]}							
	RAW12	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 2Ch ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	{Xh, P19[11:0], P18[11:0], P17[11:0], P16[11:0], P15[11:0]}								{Xh, P14[11:0], P13[11:0], P12[11:0], P11[11:0], P10[11:0]}							

Table 9.2-13 Video Output Formats (1) (2/2)

Format		Line*4	AMnUVAOFH/L address offset*5	Registers	Data[255:0]															
					byte 31	byte 30	byte 29	byte 28	byte 27	byte 26	byte 25	byte 24	byte 23	byte 22	byte 21	byte 20	byte 19	byte 18	byte 17	byte 16
RAW	RAW14	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 2Dh ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	00h	{P15[13:0], P14[13:0], P13[13:0], P12[13:0]}							00h	{P11[13:0], P10[13:0], P9[13:0], P8[13:0]}						
	RAW16	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 2Eh ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	P14[15:0]		P13[15:0]		P12[15:0]		P11[15:0]		P10[15:0]		P9[15:0]		P8[15:0]		P7[15:0]	
	RAW20	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 2Fh ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	{Xh, P11[19:0], P10[19:0], P9[19:0]}								{Xh, P8[19:0], P7[19:0], P6[19:0]}							
MIPI recommended memory storage format (full bypass)*2		Both	Non-use	CRUnCTRL.ATH = 1	Byte 32 [7:0]*1	Byte 31 [7:0]*1	Byte 30 [7:0]*1	Byte 29 [7:0]*1	Byte 28 [7:0]*1	Byte 27 [7:0]*1	Byte 26 [7:0]*1	Byte 25 [7:0]*1	Byte 24 [7:0]*1	Byte 23 [7:0]*1	Byte 22 [7:0]*1	Byte 21 [7:0]*1	Byte 20 [7:0]*1	Byte 19 [7:0]*1	Byte 18 [7:0]*1	Byte 17 [7:0]*1

Note 1. Data from MIPI CSI-2 LINK is output in 8-bit units. However, it is output in the data array of the Recommended Memory Storage of the MIPI CSI-2 V2.1 standard.

Note 2. Only data types marked with ✓ in “MIPI Recommended Memory Storage Format” in **Table 9.2-11** are output in this format. Otherwise, the data will be output in one of the above formats.

Note 3. The CbCr(U/V) side is Image Stride(AMnIS) × 2. See **Image Stride** in **9.2.4.3.3(5)(a) Video data frame** for details.

Note 4. The entries in this column have the following meanings.
“Odd” means the image has an odd number of horizontal lines.
“Even” means the image has an even number of horizontal lines.
“Both” means the image has an odd and even number of horizontal lines.

Note 5. When transferring data to memory, there are cases where Cb and Cr data are transferred to one area as Y, or data is divided into two independent areas.
The entries in this column represent the following.
Non-use: Cb and Cr data are transferred to the same area as Y when transferred to memory.
Use: When Cb and Cr data are transferred to memory, they are transferred to an area separate from Y. This separate area is transferred to the area specified in the AMnUVAOFH/L register for the Y data storage destination.

Table 9.2-14 Video Output Formats (2) (1/2)

Format		Line*4	AMnUVAOFH/L address offset*5	Registers	Data[255:0]															
					byte 15	byte 14	byte 13	byte 12	byte 11	byte 10	byte 9	byte 8	byte 7	byte 6	byte 5	byte 4	byte 3	byte 2	byte 1	byte 0
YUV (YCbCr) 422	Cb 1st	Both	Non-use	ICnDMR.YCMODE = 001b CRUnCTRL.ATH = 0	Y7 [7:0]	Cr6 [7:0]	Y6 [7:0]	Cb6 [7:0]	Y5 [7:0]	Cr4 [7:0]	Y4 [7:0]	Cb4 [7:0]	Y3 [7:0]	Cr2 [7:0]	Y2 [7:0]	Cb2 [7:0]	Y1 [7:0]	Cr0 [7:0]	Y0 [7:0]	Cb0 [7:0]
	Y 1st	Both	Non-use	ICnDMR.YCMODE = 000b CRUnCTRL.ATH=0	Cr6 [7:0]	Y7 [7:0]	Cb6 [7:0]	Y6 [7:0]	Cr4 [7:0]	Y5 [7:0]	Cb4 [7:0]	Y4 [7:0]	Cr2 [7:0]	Y3 [7:0]	Cb2 [7:0]	Y2 [7:0]	Cr0 [7:0]	Y1 [7:0]	Cb0 [7:0]	Y0 [7:0]
	YC separation	Both	Non-use	ICnDMR.YCMODE = 010b CRUnCTRL.ATH = 0	Y15 [7:0]	Y14 [7:0]	Y13 [7:0]	Y12 [7:0]	Y11 [7:0]	Y10 [7:0]	Y9 [7:0]	Y8 [7:0]	Y7 [7:0]	Y6 [7:0]	Y5 [7:0]	Y4 [7:0]	Y3 [7:0]	Y2 [7:0]	Y1 [7:0]	Y0 [7:0]
		Both	Use		Cr14 [7:0]	Cb14 [7:0]	Cr12 [7:0]	Cb12 [7:0]	Cr10 [7:0]	Cb10 [7:0]	Cr8 [7:0]	Cb8 [7:0]	Cr6 [7:0]	Cb6 [7:0]	Cr4 [7:0]	Cb4 [7:0]	Cr2 [7:0]	Cb2 [7:0]	Cr0 [7:0]	Cb0 [7:0]
	Y component extraction	Both	Non-use	ICnDMR.YCMODE = 011b CRUnCTRL.ATH = 0	Y15 [7:0]	Y14 [7:0]	Y13 [7:0]	Y12 [7:0]	Y11 [7:0]	Y10 [7:0]	Y9 [7:0]	Y8 [7:0]	Y7 [7:0]	Y6 [7:0]	Y5 [7:0]	Y4 [7:0]	Y3 [7:0]	Y2 [7:0]	Y1 [7:0]	Y0 [7:0]
YUV (YCbCr) 420	Cb 1st	Odd	Non-use	ICnDMR.YCMODE = 101b CRUnCTRL.ATH = 0	Y7 [7:0]	Cr6 [7:0]	Y6 [7:0]	Cb6 [7:0]	Y5 [7:0]	Cr4 [7:0]	Y4 [7:0]	Cb4 [7:0]	Y3 [7:0]	Cr2 [7:0]	Y2 [7:0]	Cb2 [7:0]	Y1 [7:0]	Cr0 [7:0]	Y0 [7:0]	Cb0 [7:0]
		Even	Non-use		Y15 [7:0]	Y14 [7:0]	Y13 [7:0]	Y12 [7:0]	Y11 [7:0]	Y10 [7:0]	Y9 [7:0]	Y8 [7:0]	Y7 [7:0]	Y6 [7:0]	Y5 [7:0]	Y4 [7:0]	Y3 [7:0]	Y2 [7:0]	Y1 [7:0]	Y0 [7:0]
	Y 1st	Odd	Non-use	ICnDMR.YCMODE = 100b/111b CRUnCTRL.ATH = 0	Cr6 [7:0]	Y7 [7:0]	Cb6 [7:0]	Y6 [7:0]	Cr4 [7:0]	Y5 [7:0]	Cb4 [7:0]	Y4 [7:0]	Cr2 [7:0]	Y3 [7:0]	Cb2 [7:0]	Y2 [7:0]	Cr0 [7:0]	Y1 [7:0]	Cb0 [7:0]	Y0 [7:0]
		Even	Non-use		Y15 [7:0]	Y14 [7:0]	Y13 [7:0]	Y12 [7:0]	Y11 [7:0]	Y10 [7:0]	Y9 [7:0]	Y8 [7:0]	Y7 [7:0]	Y6 [7:0]	Y5 [7:0]	Y4 [7:0]	Y3 [7:0]	Y2 [7:0]	Y1 [7:0]	Y0 [7:0]
	YC separation*3	Both	Non-use	ICnDMR.YCMODE = 110b CRUnCTRL.ATH = 0	Y15 [7:0]	Y14 [7:0]	Y13 [7:0]	Y12 [7:0]	Y11 [7:0]	Y10 [7:0]	Y9 [7:0]	Y8 [7:0]	Y7 [7:0]	Y6 [7:0]	Y5 [7:0]	Y4 [7:0]	Y3 [7:0]	Y2 [7:0]	Y1 [7:0]	Y0 [7:0]
		Both	Use		Cr14 [7:0]	Cb14 [7:0]	Cr12 [7:0]	Cb12 [7:0]	Cr10 [7:0]	Cb10 [7:0]	Cr8 [7:0]	Cb8 [7:0]	Cr6 [7:0]	Cb6 [7:0]	Cr4 [7:0]	Cb4 [7:0]	Cr2 [7:0]	Cb2 [7:0]	Cr0 [7:0]	Cb0 [7:0]
RGB-888	24 bit/pixel	Both	Non-use	ICnDMR.RGBMODE = 00b CRUnCTRL.ATH = 0	B5 [7:0]	R4 [7:0]	G4 [7:0]	B4 [7:0]	R3 [7:0]	G3 [7:0]	B3 [7:0]	R2 [7:0]	G2 [7:0]	B2 [7:0]	R1 [7:0]	G1 [7:0]	B1 [7:0]	R0 [7:0]	G0 [7:0]	B0 [7:0]
	32 bit/pixel	Both	Non-use	ICnDMR.RGBMODE = 01b CRUnCTRL.ATH = 0	00h	R3 [7:0]	G3 [7:0]	B3 [7:0]	00h	R2 [7:0]	G2 [7:0]	B2 [7:0]	00h	R1 [7:0]	G1 [7:0]	B1 [7:0]	00h	R0 [7:0]	G0 [7:0]	B0 [7:0]
ARGB-8888	B 1st	Both	Non-use	ICnDMR.RGBMODE = 10b CRUnCTRL.ATH = 0	A3 [7:0]	R3 [7:0]	G3 [7:0]	B3 [7:0]	A2 [7:0]	R2 [7:0]	G2 [7:0]	B2 [7:0]	A1 [7:0]	R1 [7:0]	G1 [7:0]	B1 [7:0]	A0 [7:0]	R0 [7:0]	G0 [7:0]	B0 [7:0]
	A 1st	Both	Non-use	ICnDMR.RGBMODE = 11b CRUnCTRL.ATH = 0	B3 [7:0]	G3 [7:0]	R3 [7:0]	A3 [7:0]	B2 [7:0]	G2 [7:0]	R2 [7:0]	A2 [7:0]	B1 [7:0]	G1 [7:0]	R1 [7:0]	A1 [7:0]	B0 [7:0]	G0 [7:0]	R0 [7:0]	A0 [7:0]
RAW	RAW6	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 28h ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	{Xh, P19[5:0], P18[5:0], P17[5:0], P16[5:0], P15[5:0], P14[5:0], P13[5:0], P12[5:0], P11[5:0], P10[5:0]}								{Xh, P9[5:0], P8[5:0], P7[5:0], P6[5:0], P5[5:0], P4[5:0], P3[5:0], P2[5:0], P1[5:0], P0[5:0]}							
	RAW7	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 29h ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	{Xh, P17[6:0], P16[6:0], P15[6:0], P14[6:0], P13[6:0], P12[6:0], P11[6:0], P10[6:0], P9[6:0]}								{Xh, P8[6:0], P7[6:0], P6[6:0], P5[6:0], P4[6:0], P3[6:0], P2[6:0], P1[6:0], P0[6:0]}							
	RAW8	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 2Ah ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	P15 [7:0]	P14 [7:0]	P13 [7:0]	P12 [7:0]	P11 [7:0]	P10 [7:0]	P9 [7:0]	P8 [7:0]	P7 [7:0]	P6 [7:0]	P5 [7:0]	P4 [7:0]	P3 [7:0]	P2 [7:0]	P1 [7:0]	P0 [7:0]
	RAW10	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 2Bh ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	{Xh, P11[9:0], P10[9:0], P9[9:0], P8[9:0], P7[9:0], P6[9:0]}								{Xh, P5[9:0], P4[9:0], P3[9:0], P2[9:0], P1[9:0], P0[9:0]}							
	RAW12	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 2Ch ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	{Xh, P9[11:0], P8[11:0], P7[11:0], P6[11:0], P5[11:0]}								{Xh, P4[11:0], P3[11:0], P2[11:0], P1[11:0], P0[11:0]}							

Table 9.2-14 Video Output Formats (2) (2/2)

Format		Line*4	AMnUVAOFH/L address offset*5	Registers	Data[255:0]															
					byte 15	byte 14	byte 13	byte 12	byte 11	byte 10	byte 9	byte 8	byte 7	byte 6	byte 5	byte 4	byte 3	byte 2	byte 1	byte 0
RAW	RAW14	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 2Dh ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	00h	{P7[13:0], P6[13:0], P5[13:0], P4[13:0]}							00h	{P3[13:0], P2[13:0], P1[13:0], P0[13:0]}						
	RAW16	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 2Eh ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	P7[15:0]		P6[15:0]		P5[15:0]		P4[15:0]		P3[15:0]		P2[15:0]		P1[15:0]		P0[15:0]	
	RAW20	Both	Non-use	ICnIPMC_C0/1/2/3.INF = 2Fh ICnIPMC_C0/1/2/3.DEMTHR = 1 CRUnCTRL.ATH = 0	{Xh, P5[19:0], P4[19:0], P3[19:0]}								{Xh, P2[19:0], P1[19:0], P0[19:0]}							
MIPI recommended memory storage format (full bypass)*2		Both	Non-use	CRUnCTRL.ATH = 1	Byte 16 [7:0]*1	Byte 15 [7:0]*1	Byte 14 [7:0]*1	Byte 13 [7:0]*1	Byte 12 [7:0]*1	Byte 11 [7:0]*1	Byte 10 [7:0]*1	Byte 9 [7:0]*1	Byte 8 [7:0]*1	Byte 7 [7:0]*1	Byte 6 [7:0]*1	Byte 5 [7:0]*1	Byte 4 [7:0]*1	Byte 3 [7:0]*1	Byte 2 [7:0]*1	Byte 1 [7:0]*1

Note 1. Data from MIPI CSI-2 LINK is output in 8-bit units. However, it is output in the data array of the Recommended Memory Storage of the MIPI CSI-2 V2.1 standard.

Note 2. Only data types marked with ✓ in “MIPI Recommended Memory Storage Format” in **Table 9.2-11** are output in this format. Otherwise, the data will be output in one of the above formats.

Note 3. The CbCr(U/V) side is Image Stride(AMnIS) × 2. See **Image Stride** in **9.2.4.3.3(5)(a) Video data frame** for details.

Note 4. The entries in this column have the following meanings.
“Odd” means the image has an odd number of horizontal lines.
“Even” means the image has an even number of horizontal lines.
“Both” means the image has an odd and even number of horizontal lines.

Note 5. When transferring data to memory, there are cases where Cb and Cr data are transferred to one area as Y, or data is divided into two independent areas.
The entries in this column represent the following.
Non-use: Cb and Cr data are transferred to the same area as Y when transferred to memory.
Use: When Cb and Cr data are transferred to memory, they are transferred to an area separate from Y. This separate area is transferred to the area specified in the AMnUVAOFH/L register for the Y data storage destination.

Table 9.2-15 Statistics Output Format

Format	Data[255:0]															
	byte31	byte30	byte29	byte28	byte27	byte26	byte25	byte24	byte23	byte22	byte21	byte20	byte19	byte18	byte17	byte16
Statistics data	In-block sum of absolute differ-ence data	In-block accumulated data			In-block sum of absolute differ-ence data	In-block accumulated data			In-block sum of absolute differ-ence data	In-block accumulated data			In-block sum of absolute differ-ence data	In-block accumulated data		
	D_G7 [7:0]	S_R7 [7:0]	S_G7 [7:0]	S_B7 [7:0]	D_G6 [7:0]	S_R6 [7:0]	S_G6 [7:0]	S_B6 [7:0]	D_G5 [7:0]	S_R5 [7:0]	S_G5 [7:0]	S_B5 [7:0]	D_G4 [7:0]	S_R4 [7:0]	S_G4 [7:0]	S_B4 [7:0]

Format	Data[255:0]															
	byte15	byte14	byte13	byte12	byte11	byte10	byte9	byte8	byte7	byte6	byte5	byte4	byte3	byte2	byte1	byte0
Statistics data	In-block sum of absolute differ-ence data	In-block accumulated data			In-block sum of absolute differ-ence data	In-block accumulated data			In-block sum of absolute differ-ence data	In-block accumulated data			In-block sum of absolute differ-ence data	In-block accumulated data		
	D_G3 [7:0]	S_R3 [7:0]	S_G3 [7:0]	S_B3 [7:0]	D_G2 [7:0]	S_R2 [7:0]	S_G2 [7:0]	S_B2 [7:0]	D_G1 [7:0]	S_R1 [7:0]	S_G1 [7:0]	S_B1 [7:0]	D_G0 [7:0]	S_R0 [7:0]	S_G0 [7:0]	S_B0 [7:0]

9.2.4.3.3 Transfer to DRAM

In data transfer from the CRU to the DRAM, data are transferred frame by frame separately based on the concept of memory banks. The memory banks can be set to operate with from 1 to 8 planes. As the transfer destination address for each memory bank, the transfer destination start address of the memory bank must be set such that it does not overlap with memory for the previous frames of data. Operation takes a cyclic form whereby the frame data returns to 1 when it reaches the last plane (see **Figure 9.2-9**). Accordingly, data on the DRAM must have been taken for next processing before overwriting. Set the number of memory banks so that this can be done before overwriting.

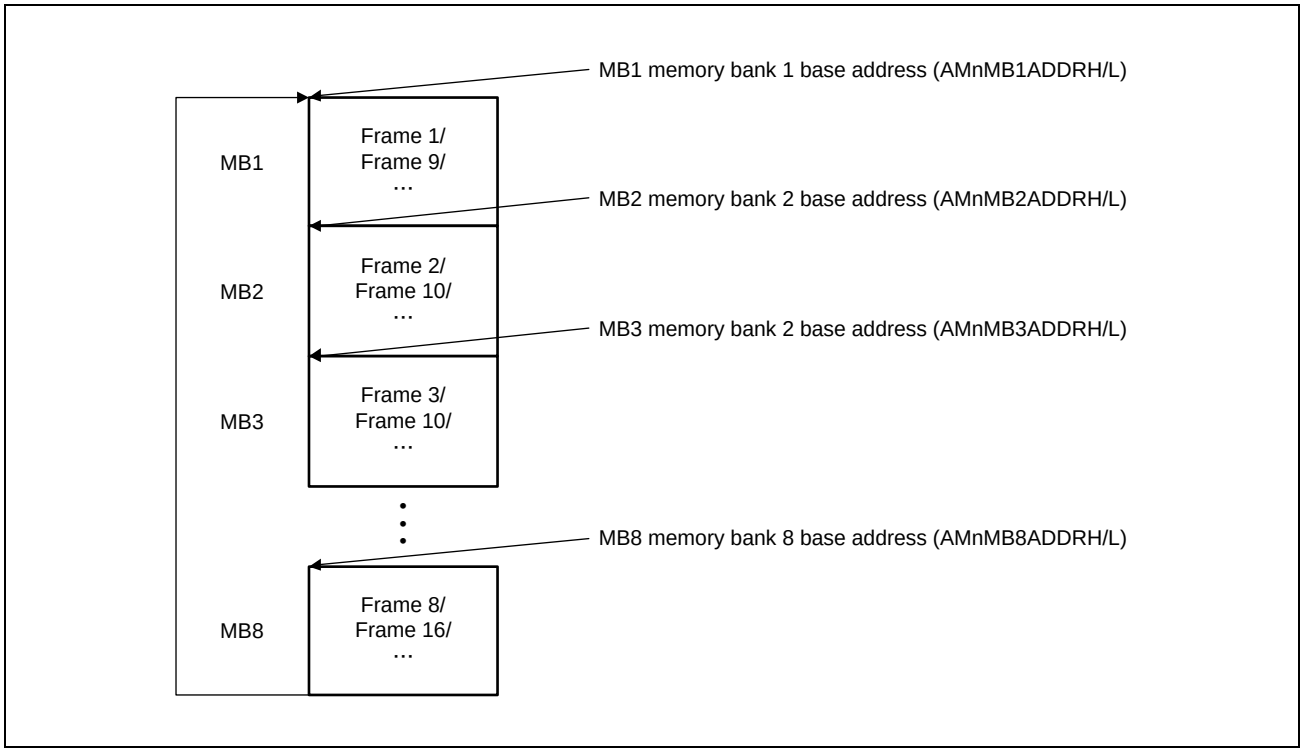


Figure 9.2-9 Image of Data Transfer to DRAM (Example: Eight Destinations in DRAM for the Transfer of Image-Processing Data)

Table 9.2-16 lists the memory bank setting addresses required for the DRAM. In this module, the transfer destination start addresses for the transfer of image data can be specified by setting the offsets listed below to proceed with transfer of up to four planes for SVC, transfer of the UV plane at the time of Y/UV separation, and transfer of non-image processing data or full-bypass data.

Furthermore, the transfer destination start addresses can be specified independently of those for statistics data that are transferred through the statistics bus.

Table 9.2-16 Memory Bank Setting Addresses

x = 0 to 7 (number of memory banks); y = 0 to 3 (number of SVC channels)

Transfer Bus	Processing	Transfer Destination Start Address	Description	Related Section
Video data bus	Image processing data	AMnMBxADDRH/L + AMnSVCAOFH/Lxy	Image data transfer destination. Transfer is possible for SVC0 to 3.	**(1) Video Data Transfer Destination Address (Video Data Storage Destination)**
		AMnMBxADDRH/L + AMnSVCAOFH/Lxy + AMnUVAOFH/L	Transfer destination of the CbCr (UV) plane at the time of YCbCb separation in image data. Transfer is possible for SVC0 to 3.	**(2) Video Data Transfer Destination Address (Storage in Y/UV Separation)**
	Non-Image processing data/ full bypass	AMnMBxADDRH/L + AMnSVCAOFH/Lxy + MnNIAOFH/L	Non-image data/full-bypass transfer destination. Transfer is possible for SVC0 to 3.	**(3) Video Data Transfer Destination Address (Image Processing/Non-Image Processing, 4-ch. SVC, Y/UV Separated Data)**
Statistics data bus	Statistics processing	AMnSDMBxADDRH/L	Statistics data transfer destination. Transfer is only possible for SVC0.	**(4) Statistics Data Transfer Destination Address (Statistics Data Storage Destination)**

Note: x represents the number of planes of each memory bank and it can be set from one to eight planes by the AMnMBVALID and AMnSDMBVALID registers (the number differs between video and statistics).
y represents the channel number of SVC and it can be set from SVC0 to SVC3 by the ICnSVCNUM register.

Data are transferred to the address obtained by adding the given offset to AMnMBxADDRH/L, the image processing data transfer destination start address for SVC0.

1) SVC OFFSET

SVCs from SVC0 to 3 are supported. Data are transferred to the address obtained by adding the offset of SVC1 to 3 to the image data transfer destination start address for SVC0.

SVC1: AMnMBxADDRH/L + AMnSVCAOFH/L
SVC2: AMnMBxADDRH/L + AMnSVCAOFH/L × 2
SVC3: AMnMBxADDRH/L + AMnSVCAOFH/L × 3

2) UV OFFSET

In the separate transfer of Y and CbCr (UV), the CbCr (UV) plane is transferred to the address obtained by adding AMnUVAOFH/L to AMnMBxADDRH/L, the image data transfer destination start address for SVC0.

UV plane for SVC0: AMnMBxADDRH/L + AMnUVAOFH/L
UV plane for SVC1: AMnMBxADDRH/L + AMnSVCAOFH/L + AMnUVAOFH/L
UV plane for SVC2: AMnMBxADDRH/L + AMnSVCAOFH/L × 2 + AMnUVAOFH/L
UV plane for SVC3: AMnMBxADDRH/L + AMnSVCAOFH/L × 3 + AMnUVAOFH/L

3) Non-Image Processing Data

Data are transferred to the address obtained by adding AMnNIAOFH/L to AMnMBxADDRH/L, the image data transfer destination start address for SVC0.

Non-image data plane for SVC0: AMnMBxADDRH/L + AMnNIAOFH/L
Non-image data plane for SVC1: AMnMBxADDRH/L + AMnSVCAOFH/L + AMnNIAOFH/L
Non-image data plane for SVC2: AMnMBxADDRH/L + AMnSVCAOFH/L × 2 + AMnNIAOFH/L
Non-image data plane for SVC3: AMnMBxADDRH/L + AMnSVCAOFH/L × 3 + AMnNIAOFH/L

4) Statistics Data

Statistics data are transferred through the static data (SD) bus, which is independent of the video data bus. Therefore, the destination addresses for these data are defined in the following form, regardless of the address settings for image data. In addition, such data are only transferred to the following set locations, since they are only supported for SVC0.

AMnSDMBxADDRH/L

(1) Video Data Transfer Destination Address (Video Data Storage Destination)

- The start address of each memory bank as the video data transfer destination is determined by the AMnMB1 to 8ADDRL and AMnMB1 to 8ADDRH registers. Valid AMnMB1ADDRH/L, AMnMB2ADDRH/L, … registers are switched by AMnMBVALID. 1 to 8 can be set for one step of memory-bank transfer (examples for settings of 8, 4, and 1 are shown in **(a) AMnMBVALID[7:0] = 1111_1111b (eight transfer destinations)**, **(b) AMnMBVALID[7:0] = 0000_1111b (four transfer destinations)**, and **(c) AMnMBVALID[7:0] = 0000_0001b (one transfer destination)**, respectively).

Set the following offset addresses as required based on the above. Set the following offset addresses as required based on the above description.

- Set the SVC offset addresses, AMnSVCAOFL and AMnSVCAOFH, when SVC1 to 3 are used.
- Set the non-image processing offset addresses, AMnNIAOFL and AMnNIAOFH, when non-image processing/full bypass is selected.
- Set the YC offset addresses, AMnUVAOFL and AMnUVAOFH, when YC separation (ICnDMR.YCMODE[2:0] = 010b/110b) is selected.

(a) AMnMBVALID[7:0] = 1111_1111b (eight transfer destinations)

Order of transfer destinations: MB1 → MB2 → …. → MB8 → MB1 → MB2 → …

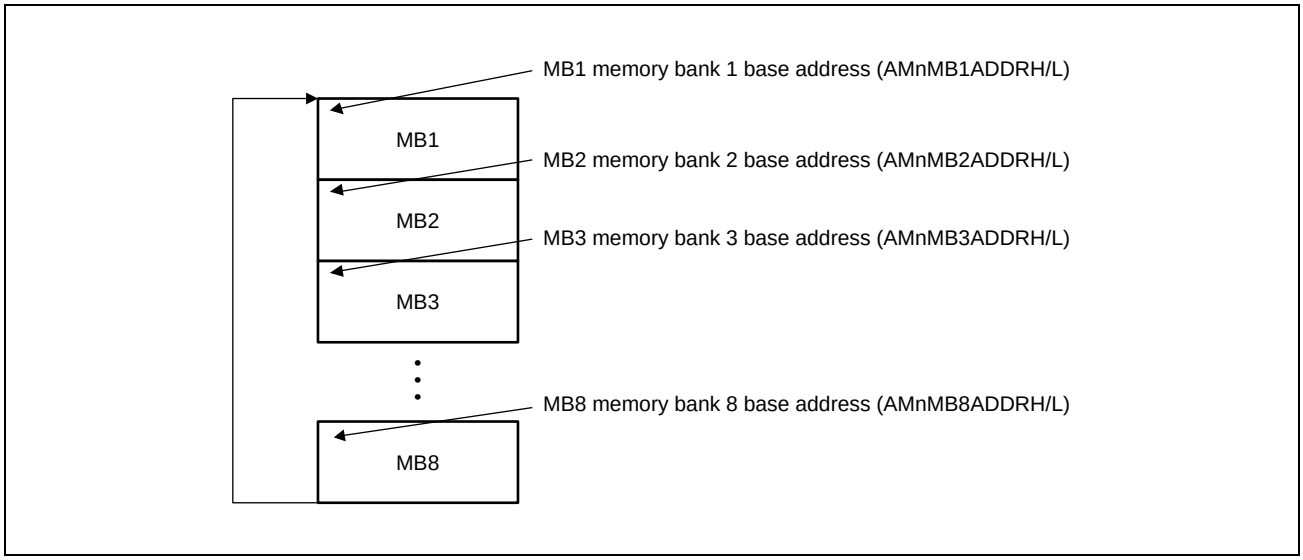


Figure 9.2-10 Image of Memory Bank (Eight Transfer Destinations)

(b) AMnMBVALID[7:0] = 0000_1111b (four transfer destinations)

Order of transfer destinations: MB1 → MB2 → MB3 → MB4 → MB1 → MB2 → …

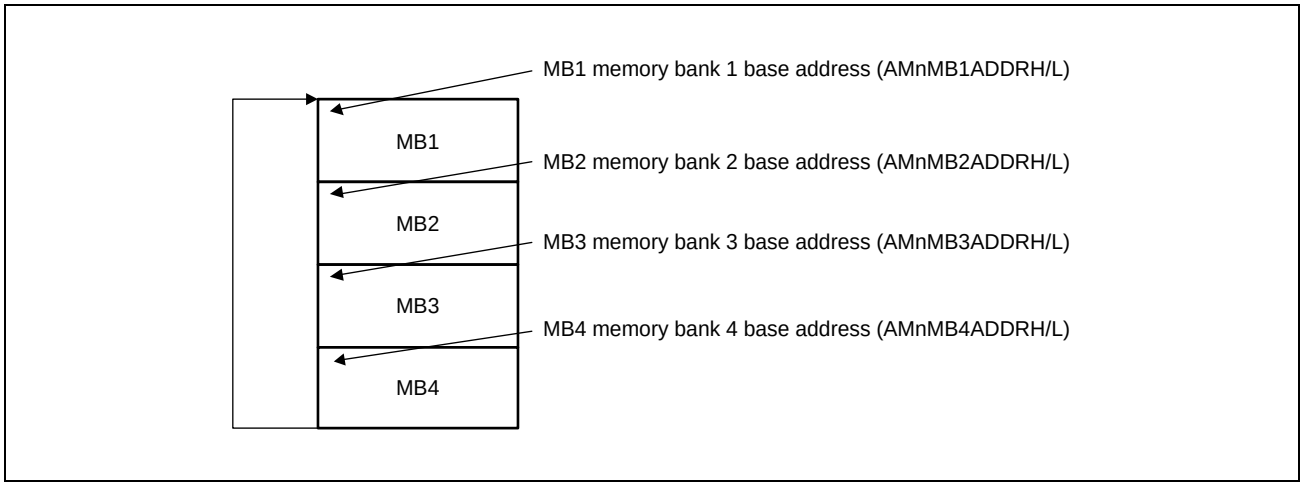


Figure 9.2-11 Image of Memory Bank (Four Transfer Destinations)

(c) AMnMBVALID[7:0] = 0000_0001b (one transfer destination)

Order of transfer destinations: MB1 → MB1 → MB1 → …

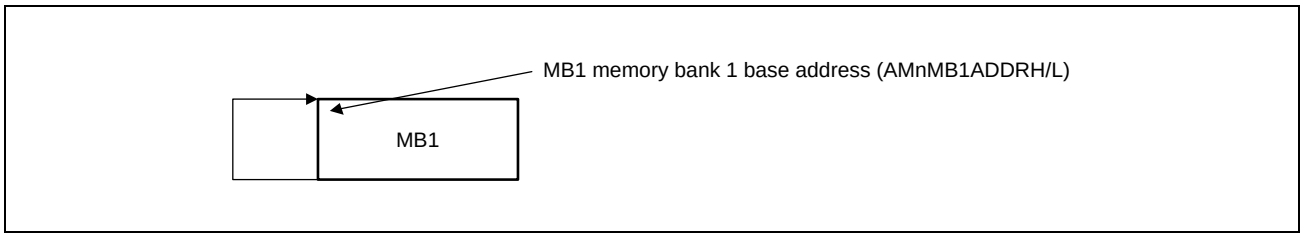


Figure 9.2-12 Image of Memory Bank (One Transfer Destination)

(2) Video Data Transfer Destination Address (Storage in Y/UV Separation)

The order of storing data in the MB1 transfer destination start address (AMnMB1ADDRH/L) in the case of Y-first (YUYV..) is shown below.

(a) When YUV is not separated (ICnDMR.YCMODE[2:0] = 000b to 001b, 100b to 101b)

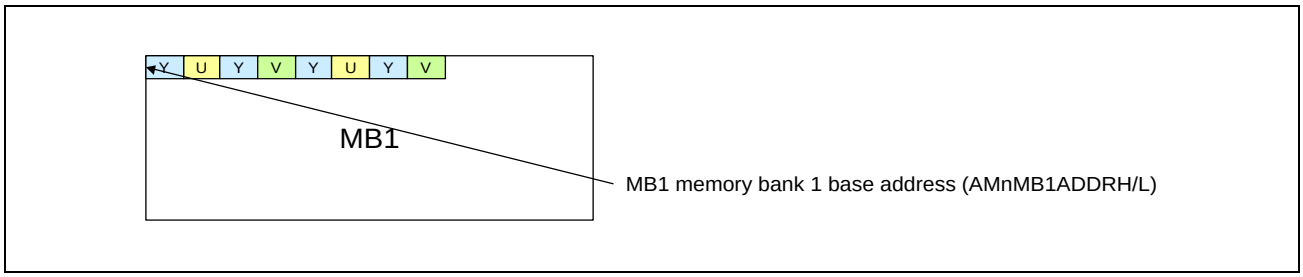


Figure 9.2-13 Image of Data Storage (in the case of no separation of YUV)

(b) When YUV is separated (ICnDMR.YCMODE[2:0] = 010b)

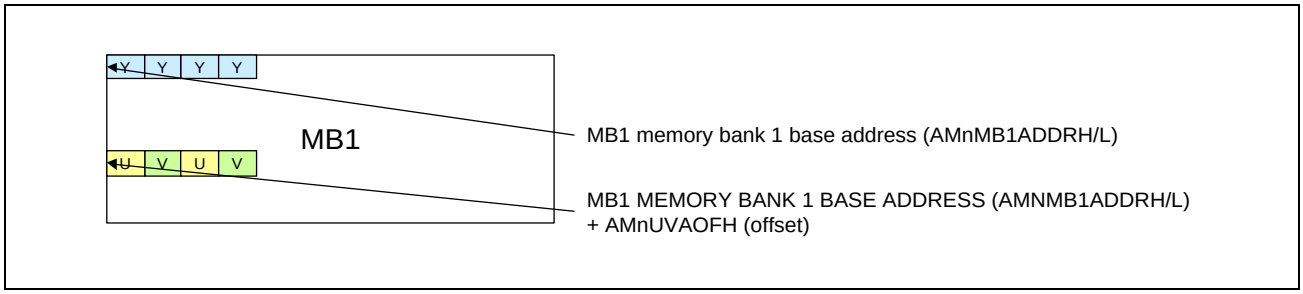


Figure 9.2-14 Image of Data Storage (in the case of separation of YUV)

(c) When only Y is separated (ICnDMR.YCMODE[2:0] = 011b)

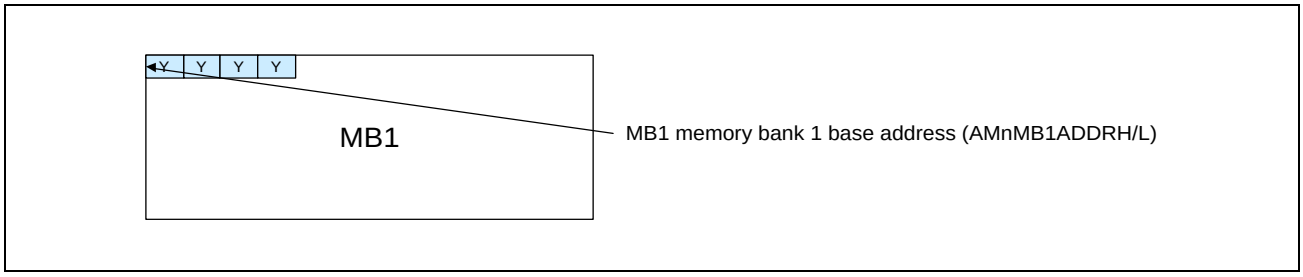


Figure 9.2-15 Image of Data Storage (in the case of only Y)

(3) Video Data Transfer Destination Address (Image Processing/Non-Image Processing, 4-ch. SVC, Y/UV Separated Data)

Figure 9.2-16 shows an image of memory to illustrate how image data of image processing/non-image processing, 4-ch. SVC, and Y/UV separated data are stored in memory.

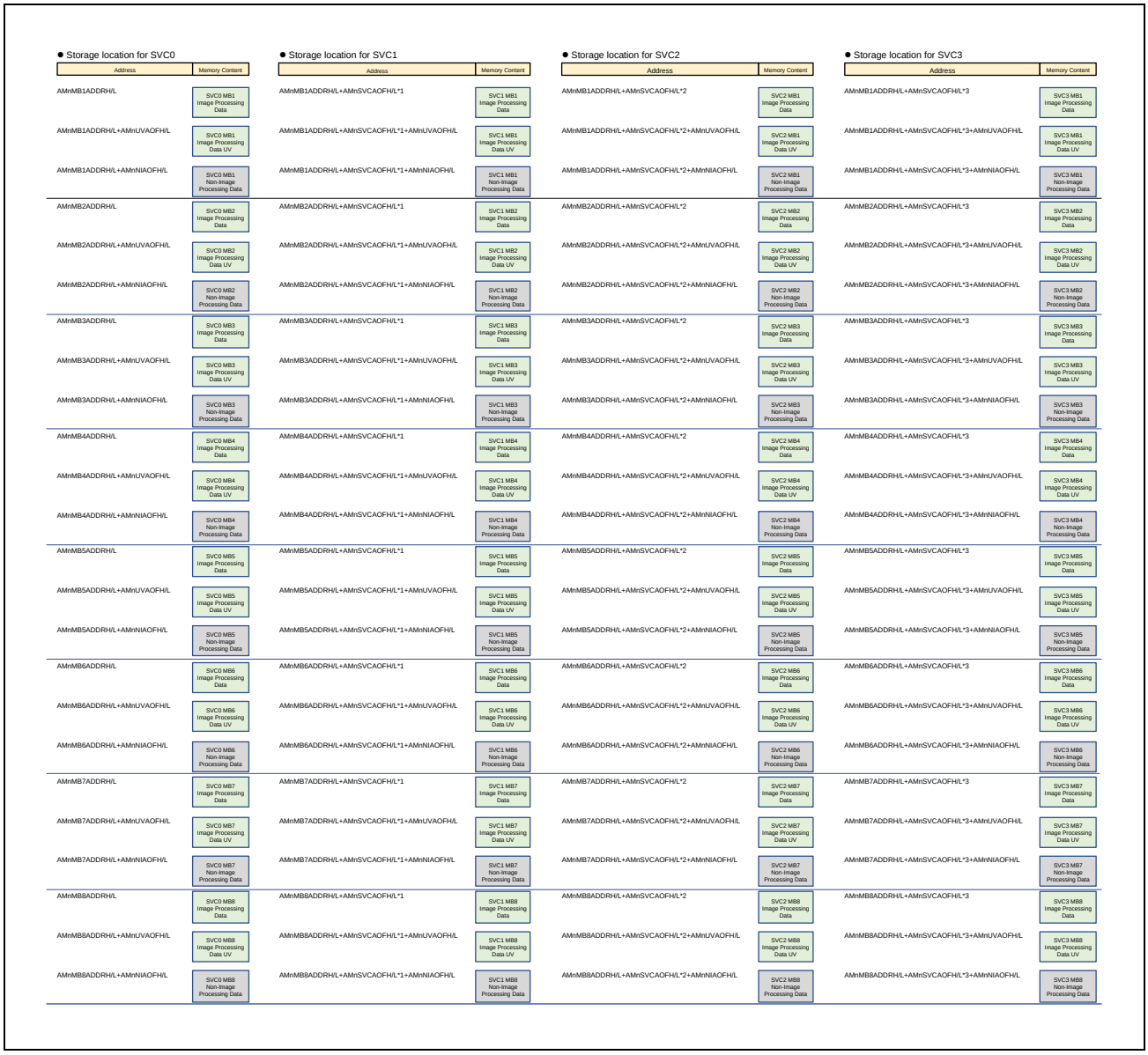


Figure 9.2-16 Image of Data Storage (Image Processing/Non-Image Processing, 4-ch. SVC, and Y/UV Separated Data)

(a) Example of Division by VC and by MB

Figure 9.2-17 shows an image of memory when data are to be stored separately by VC and by memory bank.

1) Memory map example 1 (example of division by VC)

AMnMB1ADDRH/L	SVC0 MB1 Image Processing Data
AMnMB1ADDRH/L+AMnUVAOFH/L	SVC0 MB1 Image Processing Data UV
AMnMB1ADDRH/L+AMnNIAOFH/L	SVC0 MB1 Non-Image Processing Data
AMnMB2ADDRH/L	SVC0 MB2 Image Processing Data
AMnMB2ADDRH/L+AAMnUVAOFH/L	SVC0 MB2 Image Processing Data UV
AMnMB2ADDRH/L+AMnNIAOFH/L	SVC0 MB2 Non-Image Processing Data
AMnMB3ADDRH/L	SVC0 MB3 Image Processing Data
AMnMB3ADDRH/L+AMnUVAOFH/L	SVC0 MB3 Image Processing Data UV
AMnMB3ADDRH/L+AMnNIAOFH/L	SVC0 MB3 Non-Image Processing Data
AMnMB4ADDRH/L	SVC0 MB4 Image Processing Data
AMnMB4ADDRH/L+AMnUVAOFH/L	SVC0 MB4 Image Processing Data UV
AMnMB4ADDRH/L+AMnNIAOFH/L	SVC0 MB4 Non-Image Processing Data
AMnMB5ADDRH/L	SVC0 MB5 Image Processing Data
AMnMB5ADDRH/L+AMnUVAOFH/L	SVC0 MB5 Image Processing Data UV
AMnMB5ADDRH/L+AMnNIAOFH/L	SVC0 MB5 Non-Image Processing Data
AMnMB6ADDRH/L	SVC0 MB6 Image Processing Data
AMnMB6ADDRH/L+AMnUVAOFH/L	SVC0 MB6 Image Processing Data UV
AMnMB6ADDRH/L+AMnNIAOFH/L	SVC0 MB6 Non-Image Processing Data
AMnMB7ADDRH/L	SVC0 MB7 Image Processing Data
AMnMB7ADDRH/L+AMnUVAOFH/L	SVC0 MB7 Image Processing Data UV
AMnMB7ADDRH/L+AMnNIAOFH/L	SVC0 MB7 Non-Image Processing Data
AMnMB8ADDRH/L	SVC0 MB8 Image Processing Data
AMnMB8ADDRH/L+AMnUVAOFH/L	SVC0 MB8 Image Processing Data UV
AMnMB8ADDRH/L+AMnNIAOFH/L	SVC0 MB8 Non-Image Processing Data
⋮	⋮
AMnMB1ADDRH/L+AMnSVCAOFH/L×3	SVC3 MB1 Image Processing Data
AMnMB1ADDRH/L+AMnSVCAOFH/L×3+AMnUVAOFH/L	SVC3 MB1 Image Processing Data UV
AMnMB1ADDRH/L+AMnSVCAOFH/L×3+AMnNIAOFH/L	SVC3 MB1 Non-Image Processing Data
AMnMB2ADDRH/L+AMnSVCAOFH/L×3	SVC3 MB2 Image Processing Data
AMnMB2ADDRH/L+AMnSVCAOFH/L×3+AMnUVAOFH/L	SVC3 MB2 Image Processing Data UV
AMnMB2ADDRH/L+AMnSVCAOFH/L×3+AMnNIAOFH/L	SVC3 MB2 Non-Image Processing Data
AMnMB3ADDRH/L+AMnSVCAOFH/L×3	SVC3 MB3 Image Processing Data
AMnMB3ADDRH/L+AMnSVCAOFH/L×3+AMnUVAOFH/L	SVC3 MB3 Image Processing Data UV
AMnMB3ADDRH/L+AMnSVCAOFH/L×3+AMnNIAOFH/L	SVC3 MB3 Non-Image Processing Data
AMnMB4ADDRH/L+AMnSVCAOFH/L×3	SVC3 MB4 Image Processing Data
AMnMB4ADDRH/L+AMnSVCAOFH/L×3+AMnUVAOFH/L	SVC3 MB4 Image Processing Data UV
AMnMB4ADDRH/L+AMnSVCAOFH/L×3+AMnNIAOFH/L	SVC3 MB4 Non-Image Processing Data
AMnMB5ADDRH/L+AMnSVCAOFH/L×3	SVC3 MB5 Image Processing Data
AMnMB5ADDRH/L+AMnSVCAOFH/L×3+AMnUVAOFH/L	SVC3 MB5 Image Processing Data UV
AMnMB5ADDRH/L+AMnSVCAOFH/L×3+AMnNIAOFH/L	SVC3 MB5 Non-Image Processing Data
AMnMB6ADDRH/L+AMnSVCAOFH/L×3	SVC3 MB6 Image Processing Data
AMnMB6ADDRH/L+AMnSVCAOFH/L×3+AMnUVAOFH/L	SVC3 MB6 Image Processing Data UV
AMnMB6ADDRH/L+AMnSVCAOFH/L×3+AMnNIAOFH/L	SVC3 MB6 Non-Image Processing Data
AMnMB7ADDRH/L+AMnSVCAOFH/L×3	SVC3 MB7 Image Processing Data
AMnMB7ADDRH/L+AMnSVCAOFH/L×3+AMnUVAOFH/L	SVC3 MB7 Image Processing Data UV
AMnMB7ADDRH/L+AMnSVCAOFH/L×3+AMnNIAOFH/L	SVC3 MB7 Non-Image Processing Data
AMnMB8ADDRH/L+AMnSVCAOFH/L×3	SVC3 MB8 Image Processing Data
AMnMB8ADDRH/L+AMnSVCAOFH/L×3+AMnUVAOFH/L	SVC3 MB8 Image Processing Data UV
AMnMB8ADDRH/L+AMnSVCAOFH/L×3+AMnNIAOFH/L	SVC3 MB8 Non-Image Processing Data

Figure 9.2-17 Image of Data Storage (Example of Division by VC)

2) Memory map example 2 (example of division by MB)

AMnMB1ADDRH/L	SVC0 MB1 Image Processing Data
AMnMB1ADDRH/L+AMnUVAOFH/L	SVC0 MB1 Image Processing Data UV
AMnMB1ADDRH/L+AMnNIAOFH/L	SVC0 MB1 Non-Image Processing Data
AMnMB1ADDRH/L+AMnSVCAOFH/L×1	SVC1 MB1 Image Processing Data
AMnMB1ADDRH/L+AMnSVCAOFH/L×1+AMnUVAOFH/L	SVC1 MB1 Image Processing Data UV
AMnMB1ADDRH/L+AMnSVCAOFH/L×1+AMnNIAOFH/L	SVC1 MB1 Non-Image Processing Data
AMnMB1ADDRH/L+AMnSVCAOFH/L×2	SVC2 MB1 Image Processing Data
AMnMB1ADDRH/L+AMnSVCAOFH/L×2+AMnUVAOFH/L	SVC2 MB1 Image Processing Data UV
AMnMB1ADDRH/L+AMnSVCAOFH/L×2+AMnNIAOFH/L	SVC2 MB1 Non-Image Processing Data
AMnMB1ADDRH/L+AMnSVCAOFH/L×3	SVC3 MB1 Image Processing Data
AMnMB1ADDRH/L+AMnSVCAOFH/L×3+AMnUVAOFH/L	SVC3 MB1 Image Processing Data UV
AMnMB1ADDRH/L+AMnSVCAOFH/L×3+AMnNIAOFH/L	SVC3 MB1 Non-Image Processing Data
AMnMB2ADDRH/L	SVC0 MB2 Image Processing Data
AMnMB2ADDRH/L+AMnUVAOFH/L	SVC0 MB2 Image Processing Data UV
AMnMB2ADDRH/L+AMnNIAOFH/L	SVC0 MB2 Non-Image Processing Data
AMnMB2ADDRH/L+AMnSVCAOFH/L×1	SVC1 MB2 Image Processing Data
AMnMB2ADDRH/L+AMnSVCAOFH/L×1+AMnUVAOFH/L	SVC1 MB2 Image Processing Data UV
AMnMB2ADDRH/L+AMnSVCAOFH/L×1+AMnNIAOFH/L	SVC1 MB2 Non-Image Processing Data
AMnMB2ADDRH/L+AMnSVCAOFH/L×2	SVC2 MB2 Image Processing Data
AMnMB2ADDRH/L+AMnSVCAOFH/L×2+AMnUVAOFH/L	SVC2 MB2 Image Processing Data UV
AMnMB2ADDRH/L+AMnSVCAOFH/L×2+AMnNIAOFH/L	SVC2 MB2 Non-Image Processing Data
AMnMB2ADDRH/L+AMnSVCAOFH/L×3	SVC3 MB2 Image Processing Data
AMnMB2ADDRH/L+AMnSVCAOFH/L×3+AMnUVAOFH/L	SVC3 MB2 Image Processing Data UV
AMnMB2ADDRH/L+AMnSVCAOFH/L×3+AMnNIAOFH/L	SVC3 MB2 Non-Image Processing Data
⋮	⋮
AMnMB7ADDRH/L	SVC0 MB7 Image Processing Data
AMnMB7ADDRH/L+AMnUVAOFH/L	SVC0 MB7 Image Processing Data UV
AMnMB7ADDRH/L+AMnNIAOFH/L	SVC0 MB7 Non-Image Processing Data
AMnMB7ADDRH/L+AMnSVCAOFH/L×1	SVC1 MB7 Image Processing Data
AMnMB7ADDRH/L+AMnSVCAOFH/L×1+AMnUVAOFH/L	SVC1 MB7 Image Processing Data UV
AMnMB7ADDRH/L+AMnSVCAOFH/L×1+AMnNIAOFH/L	SVC1 MB7 Non-Image Processing Data
AMnMB7ADDRH/L+AMnSVCAOFH/L×2	SVC2 MB7 Image Processing Data
AMnMB7ADDRH/L+AMnSVCAOFH/L×2+AMnUVAOFH/L	SVC2 MB7 Image Processing Data UV
AMnMB7ADDRH/L+AMnSVCAOFH/L×2+AMnNIAOFH/L	SVC2 MB7 Non-Image Processing Data
AMnMB7ADDRH/L+AMnSVCAOFH/L×3	SVC3 MB7 Image Processing Data
AMnMB7ADDRH/L+AMnSVCAOFH/L×3+AMnUVAOFH/L	SVC3 MB7 Image Processing Data UV
AMnMB7ADDRH/L+AMnSVCAOFH/L×3+AMnNIAOFH/L	SVC3 MB7 Non-Image Processing Data
AMnMB8ADDRH/L	SVC0 MB8 Image Processing Data
AMnMB8ADDRH/L+AMnUVAOFH/L	SVC0 MB8 Image Processing Data UV
AMnMB8ADDRH/L+AMnNIAOFH/L	SVC0 MB8 Non-Image Processing Data
AMnMB8ADDRH/L+AMnSVCAOFH/L×1	SVC1 MB8 Image Processing Data
AMnMB8ADDRH/L+AMnSVCAOFH/L×1+AMnUVAOFH/L	SVC1 MB8 Image Processing Data UV
AMnMB8ADDRH/L+AMnSVCAOFH/L×1+AMnNIAOFH/L	SVC1 MB8 Non-Image Processing Data
AMnMB8ADDRH/L+AMnSVCAOFH/L×2	SVC2 MB8 Image Processing Data
AMnMB8ADDRH/L+AMnSVCAOFH/L×2+AMnUVAOFH/L	SVC2 MB8 Image Processing Data UV
AMnMB8ADDRH/L+AMnSVCAOFH/L×2+AMnNIAOFH/L	SVC2 MB8 Non-Image Processing Data
AMnMB8ADDRH/L+AMnSVCAOFH/L×3	SVC3 MB8 Image Processing Data
AMnMB8ADDRH/L+AMnSVCAOFH/L×3+AMnUVAOFH/L	SVC3 MB8 Image Processing Data UV
AMnMB8ADDRH/L+AMnSVCAOFH/L×3+AMnNIAOFH/L	SVC3 MB8 Non-Image Processing Data

Figure 9.2-18 Image of Data Storage (Example of Division by MB)

(4) Statistics Data Transfer Destination Address (Statistics Data Storage Destination)

The results of statistics processing are transferred from the statistics data bus to external memory. Since this statistics function is limited to SVC0, the external transfer destination start address is determined by AMnSDMB1 to 8ADDRL and AMnSDMB1 to 8ADDRH.

The number of planes of the memory bank is determined by AMnSDMBVALID. The transfer start address (statistics data storage destination) is determined by the MnSDMB1 to 8ADDRL and AMnSDMB1 to 8ADDRH registers.

The valid AMnSDMB1ADDR, AMnSDMB2ADDR, …registers are switched by AMnSDMBVALID.

(a) AMnSDMBVALID[7:0] = 1111_1111b (eight transfer destinations)

- Order of transfer destinations: SDMB1 → SDMB2 → … → SDMB8 → SDMB1 → SDMB2 → …

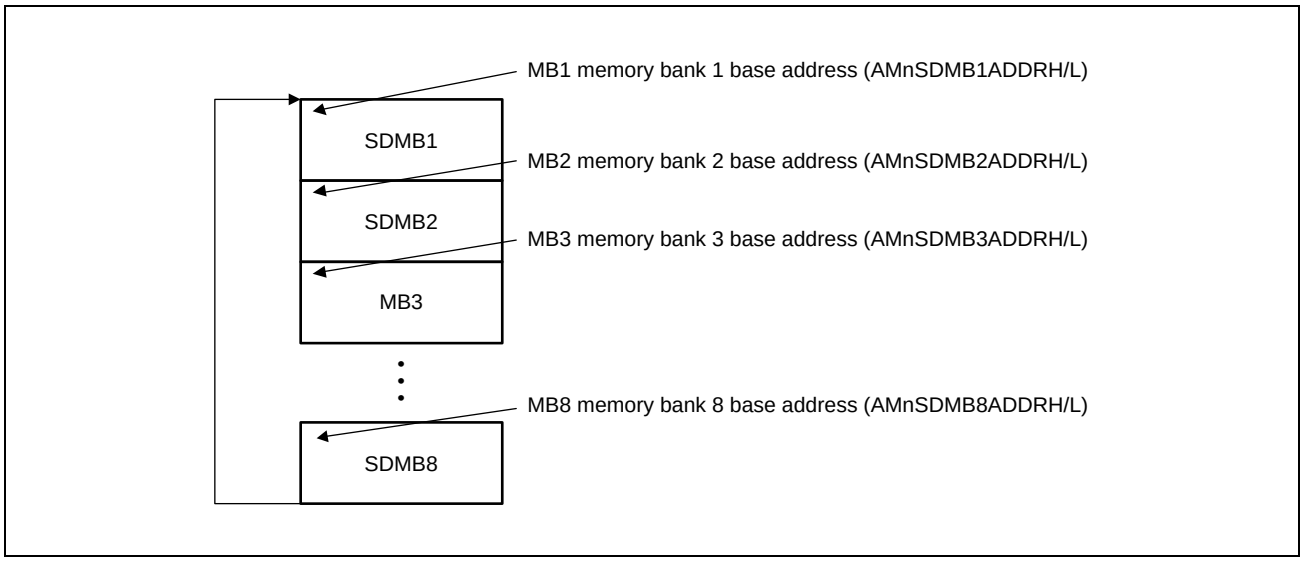


Figure 9.2-19 Image of Memory Bank (Eight Transfer Destinations)

(5) Frame Structure

(a) Video data frame

Image Stride

The image stride function is always handled for image processing and non-image processing from the video data system.

Figure 9.2-20 through **Figure 9.2-22** show the image of the storage of data with the image stride function. The image stride function involves writing from the memory write start address of each line in memory in units of address with the value set by the AMnIS (128-byte Step) register. The end of the line for writing is determined in multiples of 256 bits (32 bytes), so take this into account when setting the address. Furthermore, if a header and footer are to be appended, make the setting in consideration of that amount of data.

The following are the descriptions of storage in memory for each output format.

TYPE1: When the output format is YUV without separation (ICnDMR.YCMODE[2:0] = 000b to 001b, 011b to 101b,111b), RGB, RAW or MIPI Recommended Memory Storage Format.

As shown in **Figure 9.2-20**, the first address of memory write for each horizontal line is written to memory in address units.

TYPE2: When the output format is YUV422 (ICnDMR.YCMODE[2:0] = 010b)

As shown in **Figure 9.2-21**, the first address of memory write for each horizontal line is written to memory in address units.

TYPE3: When the output format is YUV420 (ICnDMR.YCMODE[2:0] = 110b)

As shown in **Figure 9.2-22**, Y writes the first address of memory write for each line to memory in address units of the value set in the AMnIS register.

Note that UV writes the first address of each line to memory in units of twice the value set in the AMnIS register.

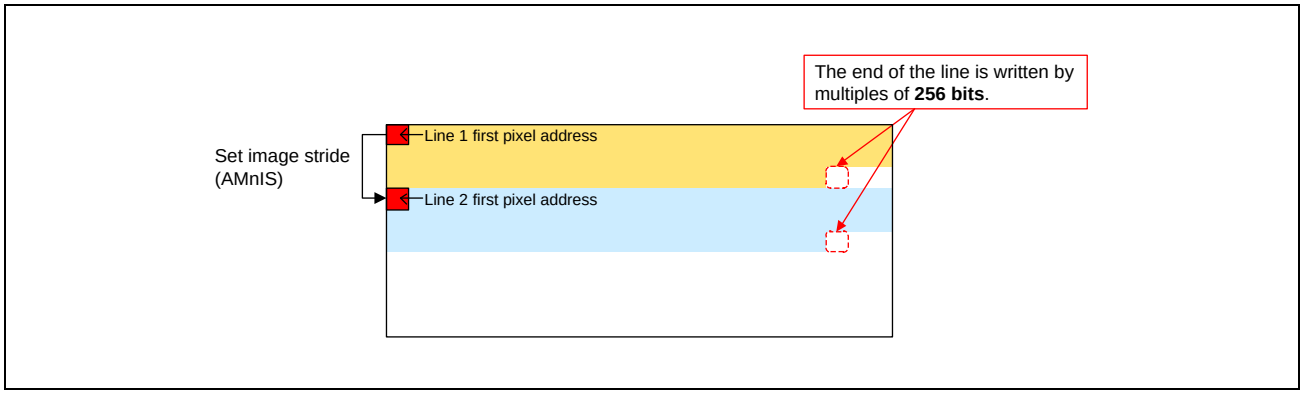


Figure 9.2-20 Image of Data Storage with Image Stride Function (TYPE1)

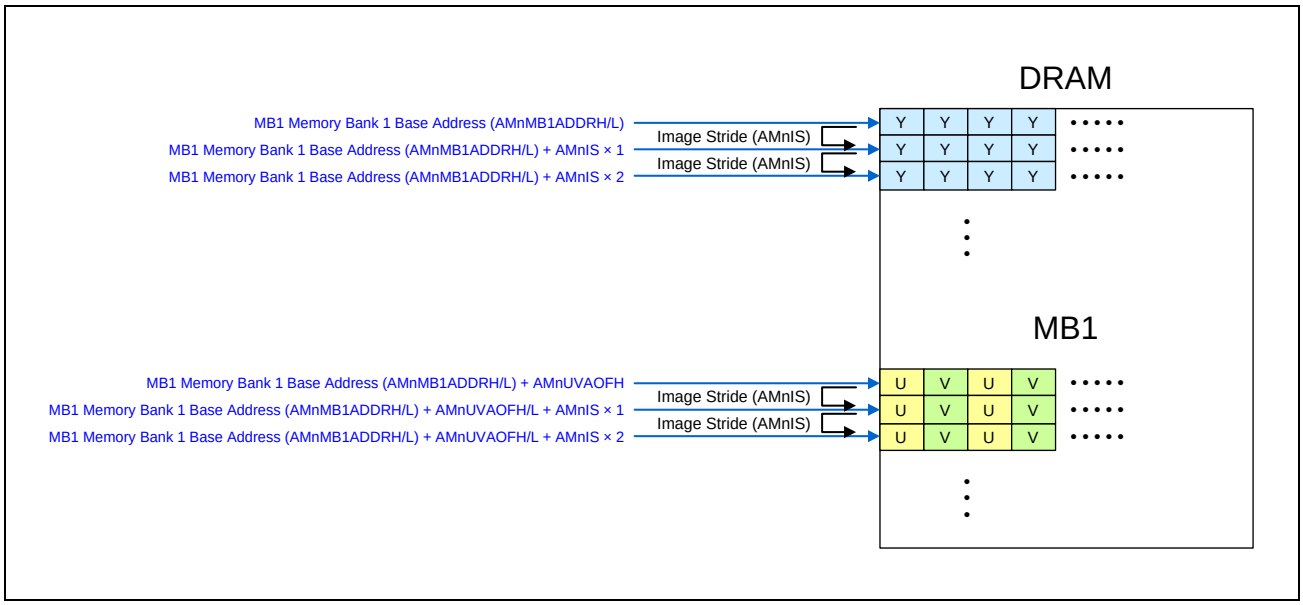


Figure 9.2-21 Image of Data Storage with Image Stride Function (TYPE2)

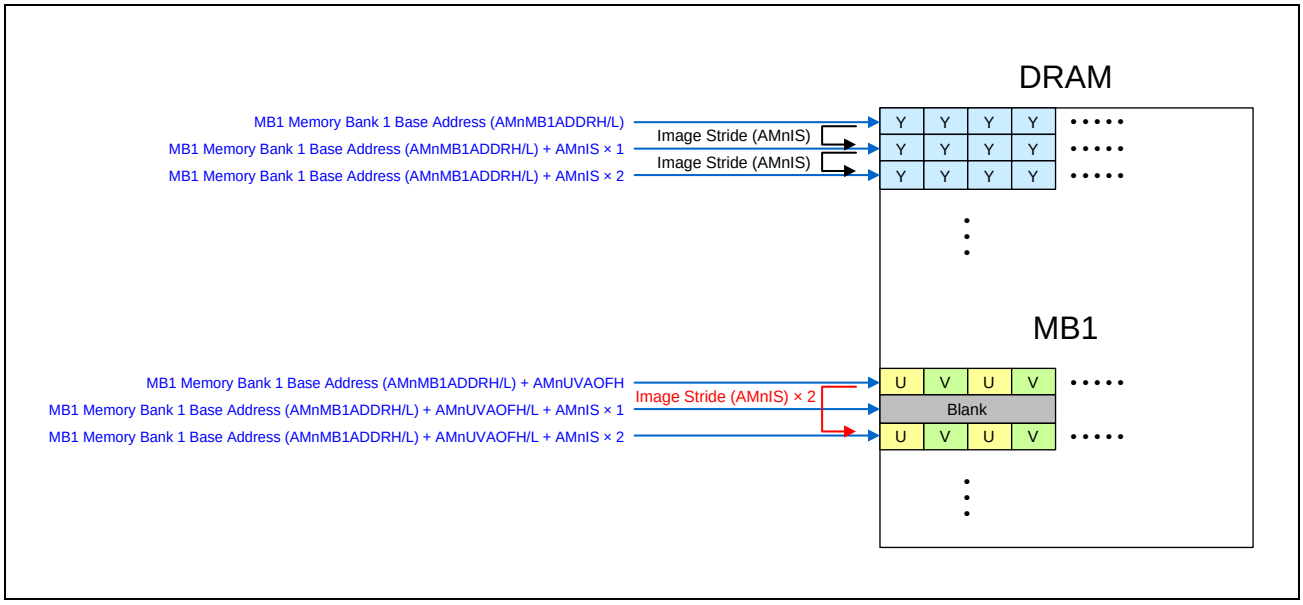


Figure 9.2-22 Image of Data Storage with Image Stride Function (TYPE3)

Image Processing Data Frame

Figure 9.2-23 shows the structure of the image processing data frame.

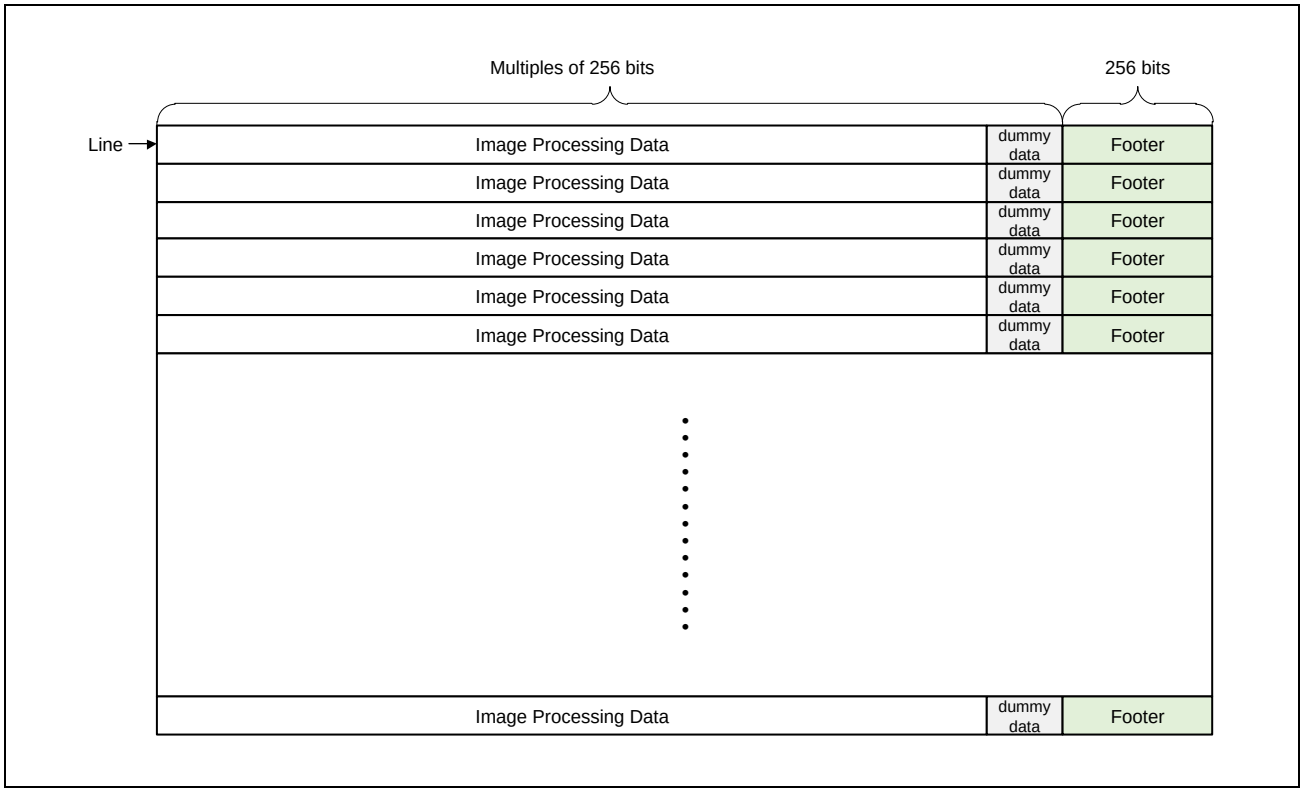


Figure 9.2-23 Image Processing Data Frame Structure

1) Dummy data

Transfer from this module to external memory proceeds in units of 256 bits (32 bytes). Therefore, only the amount of dummy data which is required to reach a 32-byte boundary is transferred. When processing images, ignore the dummy data.

2) Footer

Whether or not to insert footer with the image data can be selected by using the ICnDTVP.FTON register. For the data format of 256 bits of footer data, see **Footer Appending Function**. The value of this footer is updated alternately every time data at the target address are updated. The state of progress can be checked by monitoring which horizontal line has been updated.

This module has functionality for checking the address for transfer from the AXI bus via a register or using an interrupt to indicate the transfer of data to the specified address. Actually, whether or not data have been written to the DRAM can be checked by inspecting the footer to confirm the updating of data.

Non-Image Processing Data/Full-Bypass Data Frame

Figure 9.2-24 shows the structure of the non-image processing data/full-bypass data frame.

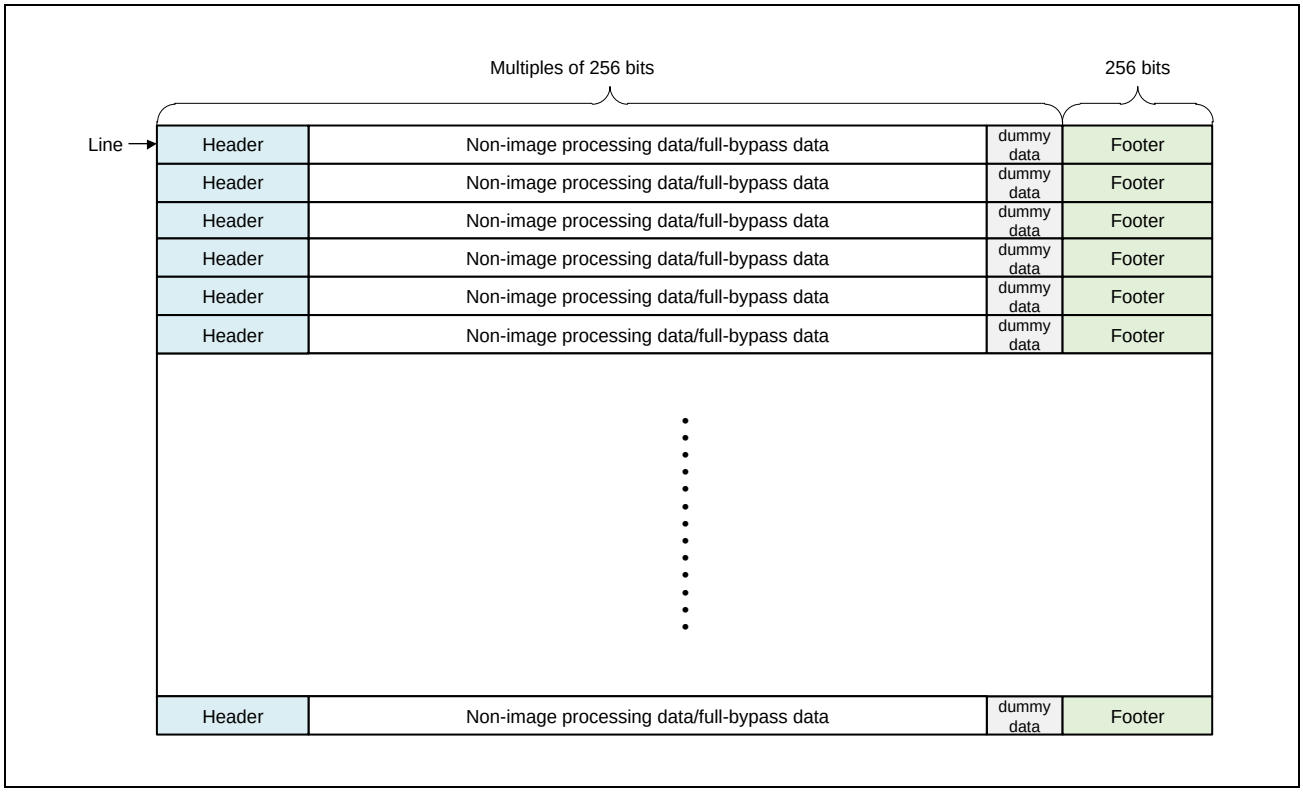


Figure 9.2-24 Non-Image Processing Data/Full-Bypass Data Frame Structure

A difference from an image processing frame is that a header can be inserted for these frame types.

Insertion of the header can be set by using the ICnDTVP.HDON register. For the format of the header, see **Header Appending Function (for Non-Image Processing Data/Full-Bypass Data)**.

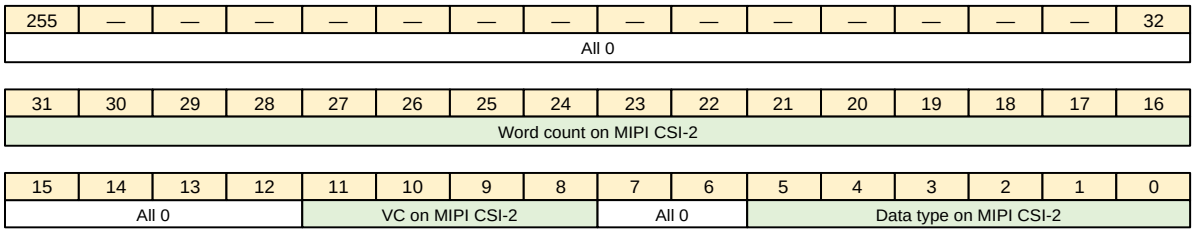
Since there are several data types for non-image processing and full-bypass data whereas image processing data are of one data type, the purpose of the header is to allow the CRU to identify the data type in line units. If the CRU has recognized the data type beforehand or a single data type is selected, headers need not be inserted.

Header Appending Function (for Non-Image Processing Data/Full-Bypass Data)

The data types for non-image processing and full-bypass are selectable from among several data types and the data are stored in a state such that they are mixed in the same SVC memory area, so a header can be appended at the start of each line for non-image processing or full-bypass data so that the data type and data length can be identified when read. Since this header appending function is not required for a sensor that can identify the data types by their order, etc., it can be switched on or off.

However, when the pattern generator is selected for output, a header will not be appended regardless of whether the setting for header appending is on or off.

Figure 9.2-25 shows the contents of the header.



Note: Since “Word count” is the word count value based on MIPI CSI-2, it does not include data such as headers and footers which have been appended by the CRU.

Figure 9.2-25 Header Contents

Footer Appending Function

This function writes footer data to memory so that the user can see up to which horizontal line data has been written in the memory bank set by AMnMB1-8ADDRH/L (for video data).

This function can be switched on or off with ICnDTVP.FTON.

For the video data footer, 256 bits (256’h1/256’h0) are written at the end of each line unit.

At first, 256’h1 is written as a footer. The next time the memory bank is overwritten after round of writing, 256’h0 is written as a footer. Subsequently, 256’h1 and 256’h0 are written alternately each time the memory bank is overwritten.

Accordingly, by checking the value of the footer, it is possible to prevent erroneous use of data before it is overwritten.

A 256-bit footer (256’d1/256’d0) is appended to the end of each line unit from the image system (image processing, non-image processing, or full bypass) and to the end of each frame unit from the statistics system.

(b) Statistics data frame

Figure 9.2-26 shows the structure of the image processing data frame.

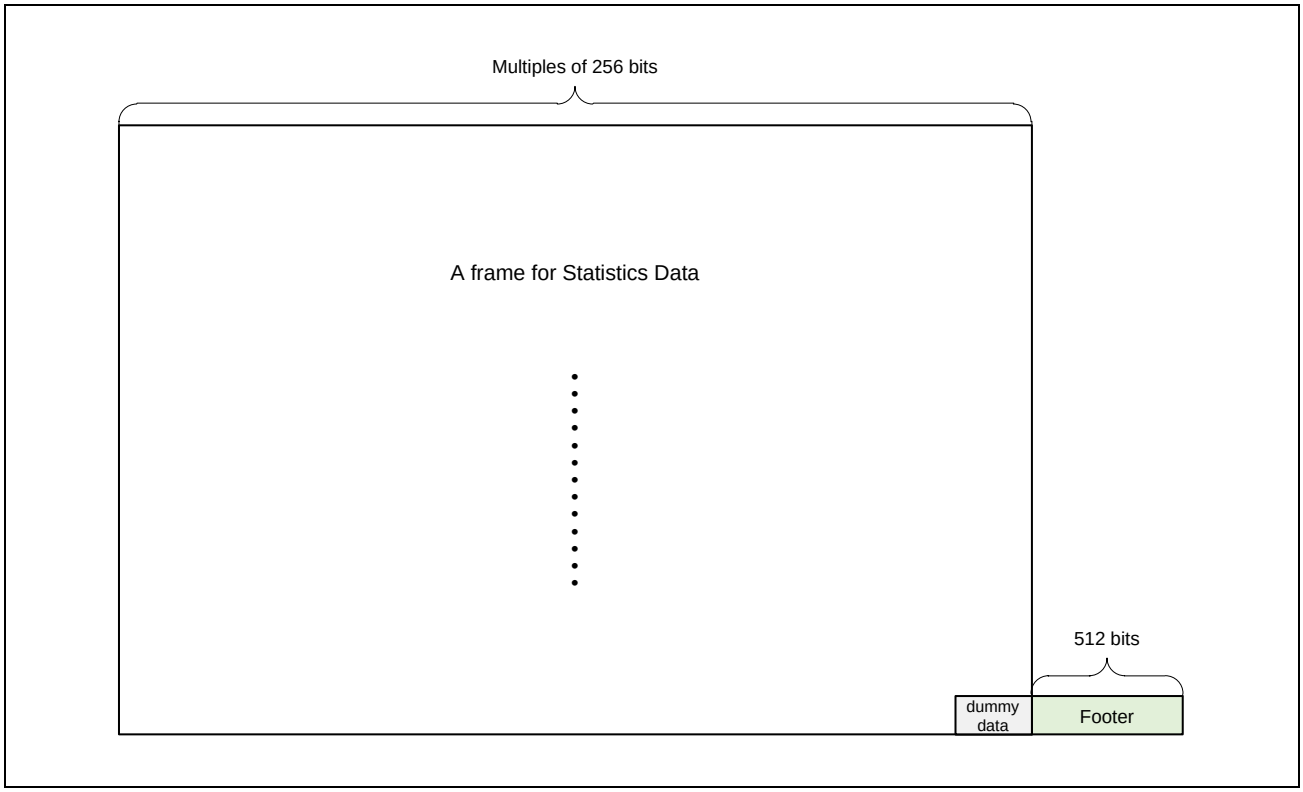


Figure 9.2-26 Statistics Data Frame Structure

Two differences between statistics frames and frames from image processing and non-image processing are that statistics frames have no concept of image striding and that footers are added in frame units. This is because statistics are processed in block units so that line-by-line processing is not possible.

Processing of the Last Line in a Frame (for Statistics Data)

Figure 9.2-27 shows an image of storing frames of statistics data. Statistics data are written to memory with no intervals in the data to reflect line transitions. The end of the frame for writing is determined in multiples of 256 bits (32 bytes), so take this into account when setting the address.

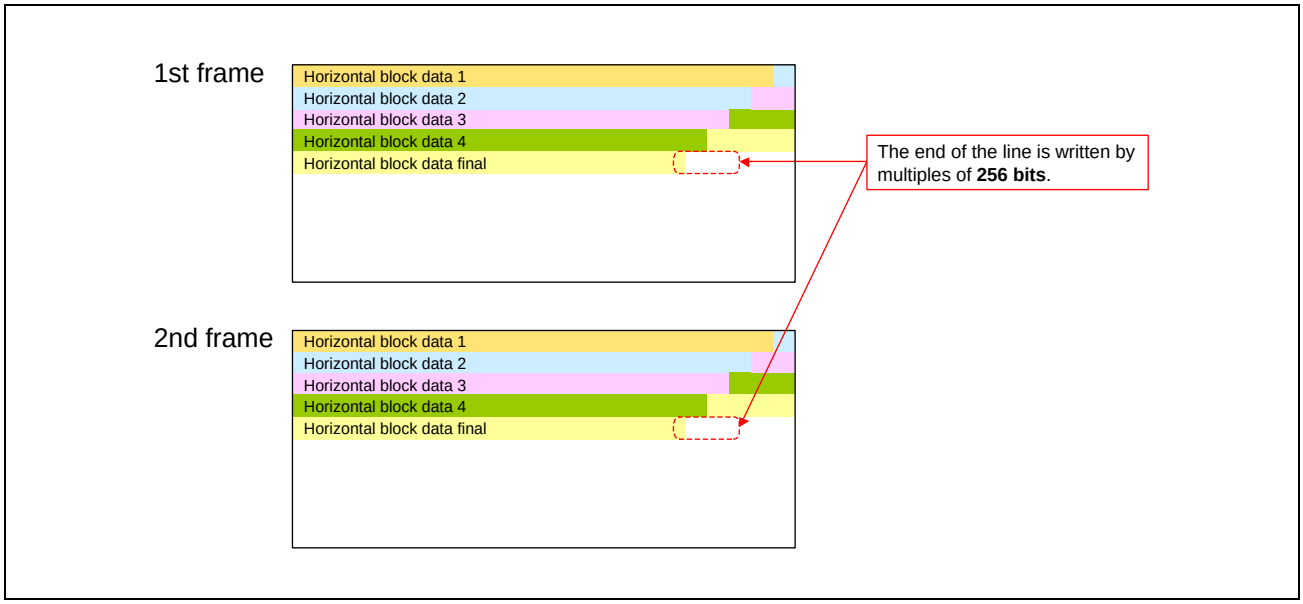


Figure 9.2-27 Image of Storing Frames of Statistics Data

Footer Appending Function

This function writes a footer to the end of each memory so that the user can see which bank number of the memory bank set by MnSDMB1-8ADDRH/L (for statistics data) has been written to the memory.

This function can be switched on or off with ICnDTVP.FTON.

A 512-bit ({higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256’h1} or 512-bit {higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256’h0}) footer of statistics data is written at the end of each frame unit.

Ignore the higher-order 256 bits of 512 bits because they are reserved (indefinite value).

At first, 512 bits {higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256’h1} are written as a footer.

The next time the memory bank is overwritten after one round of writing, 512 bits {higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256’h0} are written as a footer.

After that, 512 bits {higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256’h1} and 512 bits {higher-order 256 bits: reserved (indefinite value); lower-order 256 bits: 256’h0} are alternately written each time overwriting is done.

Therefore, by checking the footer value as the video data footer, it is possible to prevent the data from being inadvertently used before overwriting.

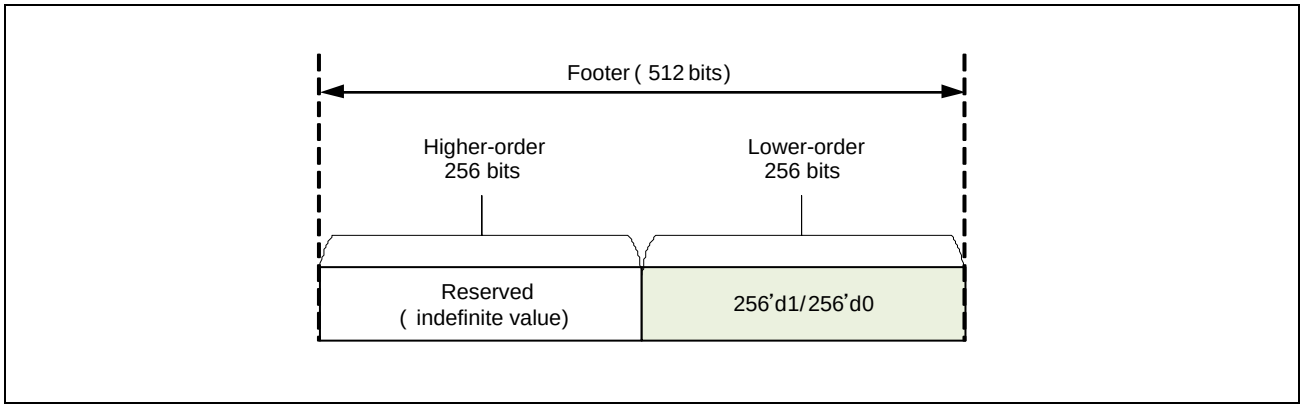


Figure 9.2-28 Statistics Data Footer Structure

9.2.4.3.4 Pattern Generator (for Debugging)

A test image (YUV422 only) can be generated for use as an image converter input by setting the following registers. This function is used for debugging to identify where the problem is originated (in the sensor or in the internal settings). Two modes are implemented for this pattern (monochrome or pattern). This pattern generator operates as VC0. You can select the route of image processing/non-image processing. See **9.2.5 Operation** for the setting procedure.

TIEN = ICnTICTRL1.TIEN: Enables generation of a test image pattern.
TIMODE = ICnTICTRL1.TIMODE: Selects a test image generation pattern (monochrome or pattern).

TIROTSEL = ICnTICTRL1.TIROTSEL: Selects the route of image processing/non-image processing.

TIPTNY1[3:0] = ICnTICTRL1.TIPTNY1[3:0]: Bit settings for the pixel (Y) counter
TIPTNU1[3:0] =ICnTICTRL1.TIPTNU1[3:0]: Bit settings for the pixel (U) counter
TIPTNV1[3:0] = ICnTICTRL1.TIPTNV1[3:0]: Bit settings for the pixel (V) counter

TIRATE[4:0] = ICnTICTRL1.TIRATE[4:0]: Specify a test image generation rate.

TIPTNY2[7:0] = ICnTICTRL2.TIPTNY2[7:0]: Specify Y for the entire-screen same-color test image pattern.
TIPTNU2[7:0] = ICnTICTRL2.TIPTNU2[7:0]: Specify U for the entire-screen same-color test image pattern.
TIPTNV2[7:0] = ICnTICTRL2.TIPTNV2[7:0]: Specify V for the entire-screen same-color test image pattern.

TIPPL[11:0] = ICnTISIZE1.TIPPL[11:0]: Number of valid pixels per line in the test image pattern
TIN[11:0] = ICnTISIZE2.TIN[11:0]: Valid period (number of lines) per frame in the test image pattern
TIM[11:0] = ICnTISIZE2.TIM[11:0]: Invalid period (number of lines) per frame in the test image pattern

(1) Image and Settings (Common to Monochrome and Pattern Generation)

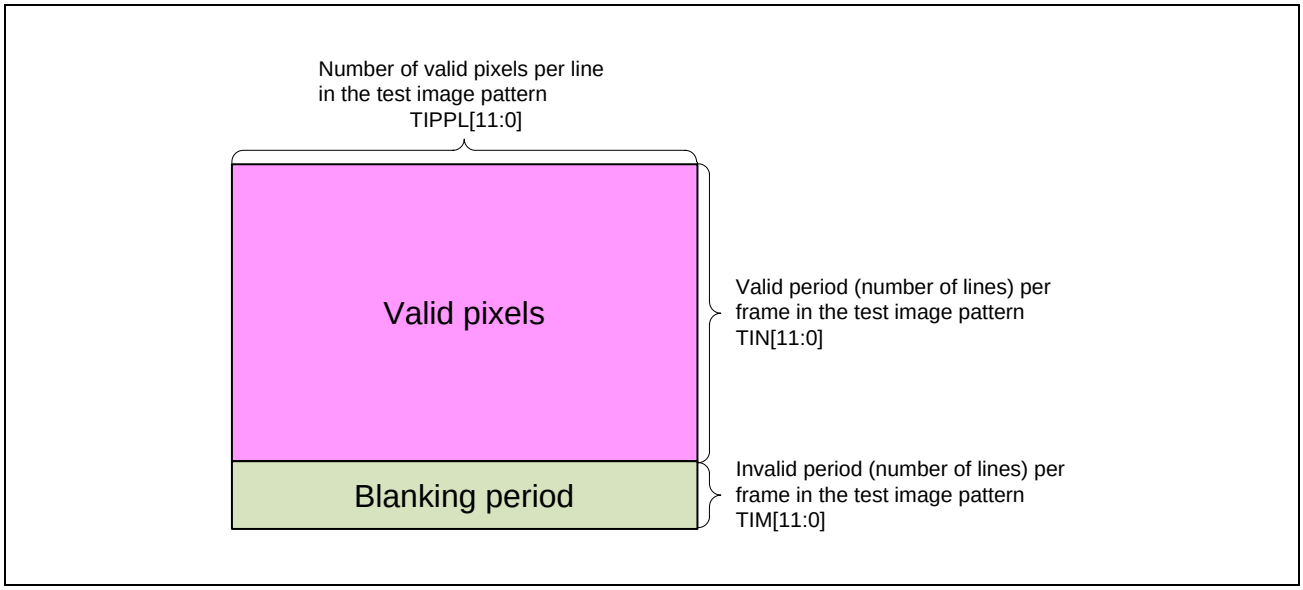


Figure 9.2-29 Image and Settings

(2) Timing Generation and Settings (Common to Monochrome and Pattern Generation)

(Bits TIRATE[4:0] allow the test image generation rate to be set in the range between 3.291 Gbps and 8.960 Gbps with VCLK at 630 MHz.)

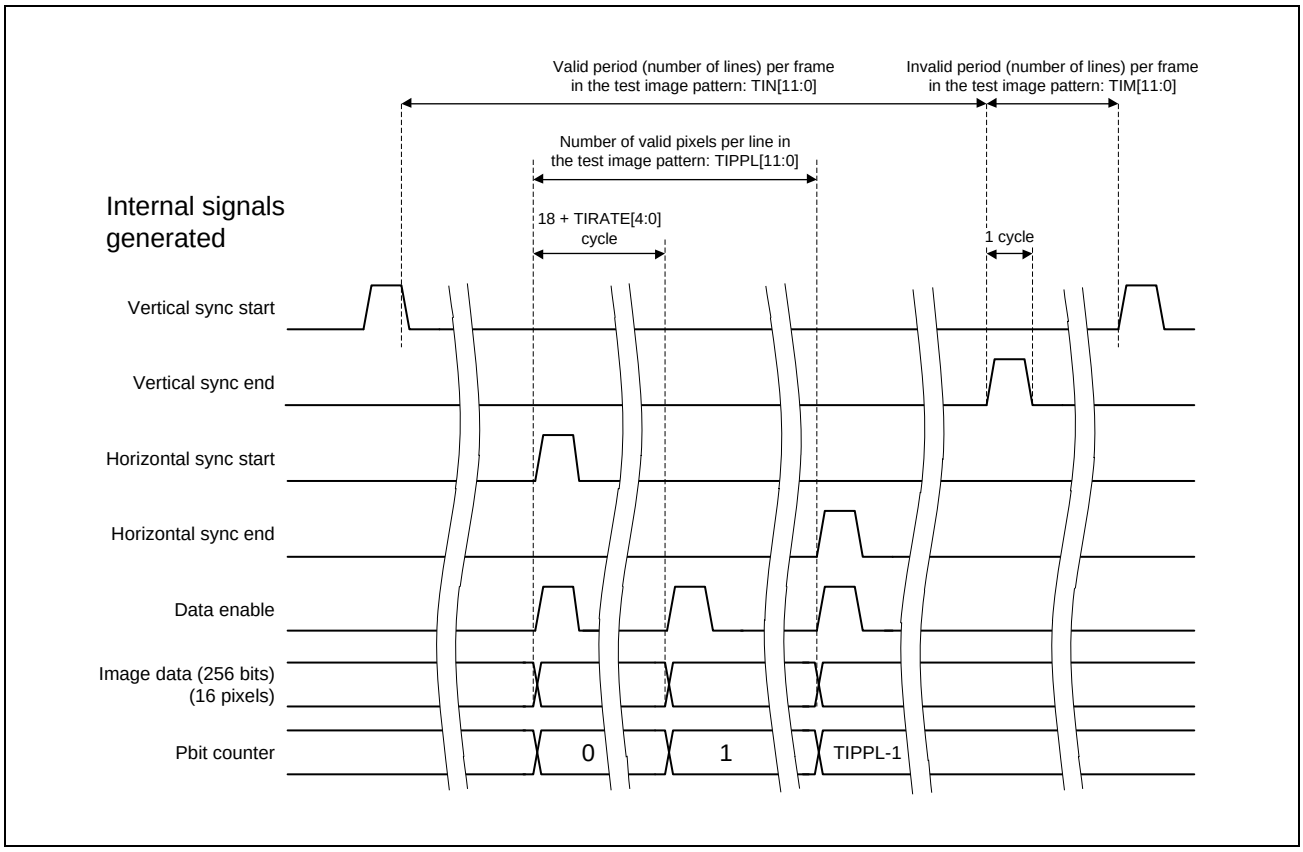
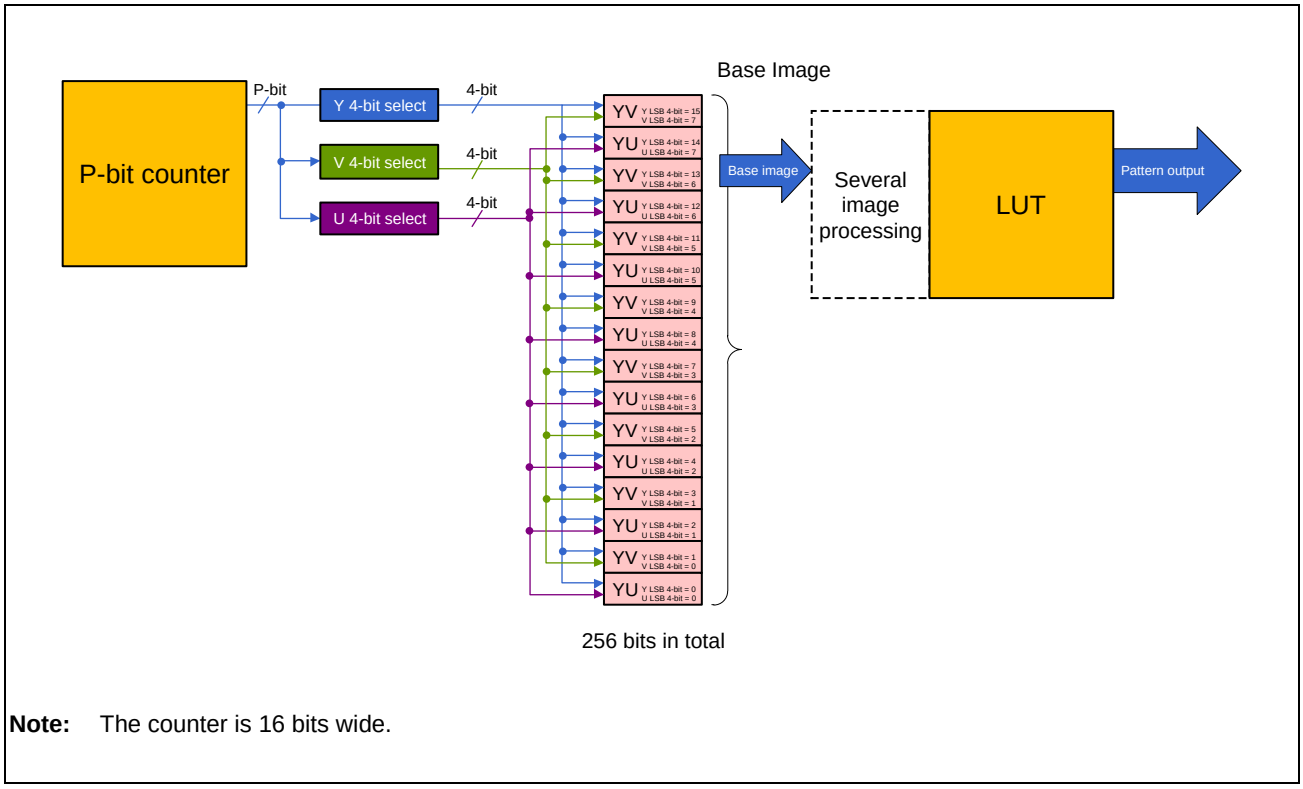


Figure 9.2-30 Timing Generation and Settings

(3) Data Generation Circuit (in the Case of Pattern Generation)

Image YUV422 8-bit data of 256 bits (16 pixels) is generated as follows (other formats cannot be generated). The 16-bit Pbit counter is incremented in 18 + TIRATE[4:0] (cycles at 630 MHz). The bits of this Pbit counter are selected by the TIPTNV1[3:0], TIPTNU1[3:0], and TIPTNY1[3:0] bits, and then they are packed and sent as follows:



Note: The counter is 16 bits wide.

Figure 9.2-31 Data Generation Circuit (in the Case of Pattern Generation)

(4) Data Generation Circuit (in the Case of Monochrome Generation)

Image YUV422 8-bit data of 256 bits (16 pixels) is generated in monochrome as follows (other formats cannot be generated). The values set by the TIPTNV2[7:0], TIPTNU2[7:0], and TIPTNY2[7:0] bits are packed and sent as follows:

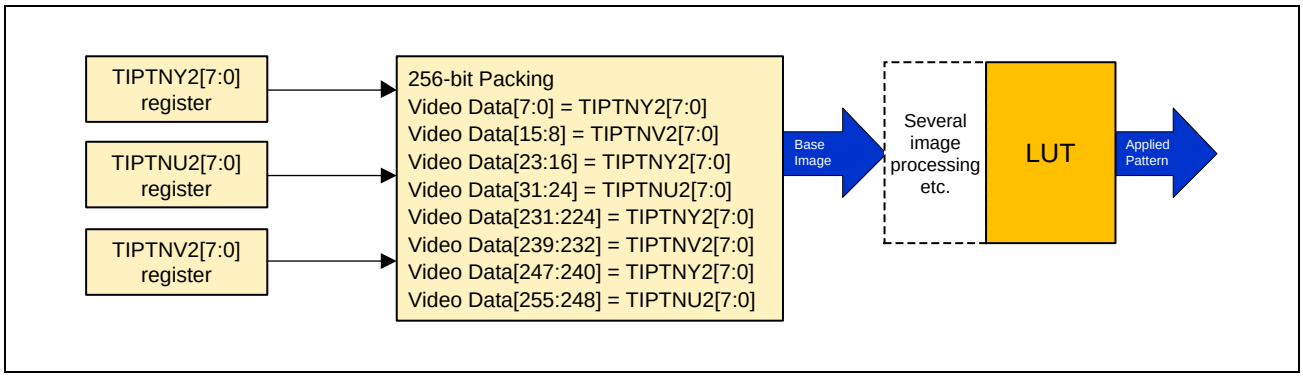


Figure 9.2-32 Data Generation Circuit (in the Case of Monochrome Generation)

(5) Generated Image Data Example (in the Case of Pattern Generation)

VGA (640 × 480): TIEN = 1b, TIMODE = 0b, TIPTNY1[3:0] = 0000b, TIPTNU1[3:0] = 0100b, TIPTNV1[3:0] = 1000b (Offset binary: ICnIPMC_C0/1/2/3.IBINSEL = 0b, ICnDMR.OBINSEL = 0b)

Table 9.2-17 Expected Values

No	Y	U	V
0	00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
16	10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, 1E, 1F,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
32	20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D, 2E, 2F,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
48	30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 3A, 3B, 3C, 3D, 3E, 3F,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
64	40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 4A, 4B, 4C, 4D, 4E, 4F,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
80	50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 5A, 5B, 5C, 5D, 5E, 5F,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
96	60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 6A, 6B, 6C, 6D, 6E, 6F,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
112	70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 7A, 7B, 7C, 7D, 7E, 7F,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
128	80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 8A, 8B, 8C, 8D, 8E, 8F,	10, 11, 12, 13, 14, 15, 16, 17, 10, 11, 12, 13, 14, 15, 16, 17,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
144	90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 9A, 9B, 9C, 9D, 9E, 9F,	10, 11, 12, 13, 14, 15, 16, 17, 10, 11, 12, 13, 14, 15, 16, 17,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
160	A0, A1, A2, A3, A4, A5, A6, A7, A8, A9, AA, AB, AC, AD, AE, AF,	10, 11, 12, 13, 14, 15, 16, 17, 10, 11, 12, 13, 14, 15, 16, 17,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
176	B0, B1, B2, B3, B4, B5, B6, B7, B8, B9, BA, BB, BC, BD, BE, BF,	10, 11, 12, 13, 14, 15, 16, 17, 10, 11, 12, 13, 14, 15, 16, 17,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
192	C0, C1, C2, C3, C4, C5, C6, C7, C8, C9, CA, CB, CC, CD, CE, CF,	10, 11, 12, 13, 14, 15, 16, 17, 10, 11, 12, 13, 14, 15, 16, 17,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
208	D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, DA, DB, DC, DD, DE, DF,	10, 11, 12, 13, 14, 15, 16, 17, 10, 11, 12, 13, 14, 15, 16, 17,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
224	E0, E1, E2, E3, E4, E5, E6, E7, E8, E9, EA, EB, EC, ED, EE, EF,	10, 11, 12, 13, 14, 15, 16, 17, 10, 11, 12, 13, 14, 15, 16, 17,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
240	F0, F1, F2, F3, F4, F5, F6, F7, F8, F9, FA, FB, FC, FD, FE, FF,	10, 11, 12, 13, 14, 15, 16, 17, 10, 11, 12, 13, 14, 15, 16, 17,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
256	00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F,	20, 21, 22, 23, 24, 25, 26, 27, 20, 21, 22, 23, 24, 25, 26, 27,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
272	10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, 1E, 1F,	20, 21, 22, 23, 24, 25, 26, 27, 20, 21, 22, 23, 24, 25, 26, 27,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,
288	20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D, 2E, 2F,	20, 21, 22, 23, 24, 25, 26, 27, 20, 21, 22, 23, 24, 25, 26, 27,	00, 01, 02, 03, 04, 05, 06, 07, 00, 01, 02, 03, 04, 05, 06, 07,

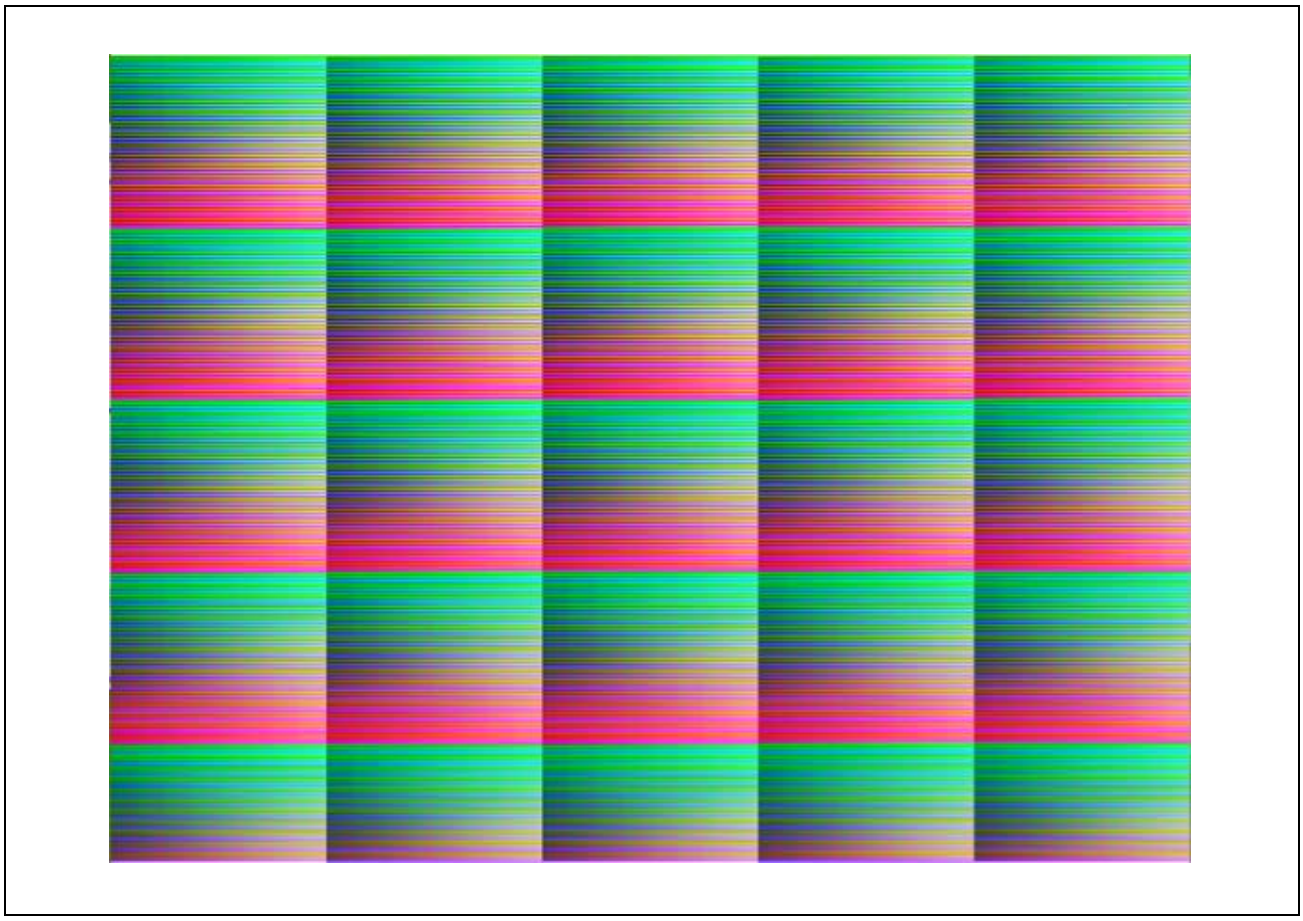


Figure 9.2-33 Generated Image Data Example (in the Case of Pattern Generation)

(6) Generated Image Data Example (in the Case of Monochrome Generation)

VGA (640 × 480): TIEN = 1b, TIMODE = 1b, TIPTNY2[7:0] = 52h, TIPTNU2[7:0] = 5Ah, TIPTNV2[3:0] = F0h (Offset binary: ICnIPMC_C0/1/2/3.IBINSEL = 0b, ICnDMR.OBINSEL = 0b)

Table 9.2-18 Expected Values

	Y	U	V
Fixed value	52h	5Ah	F0h

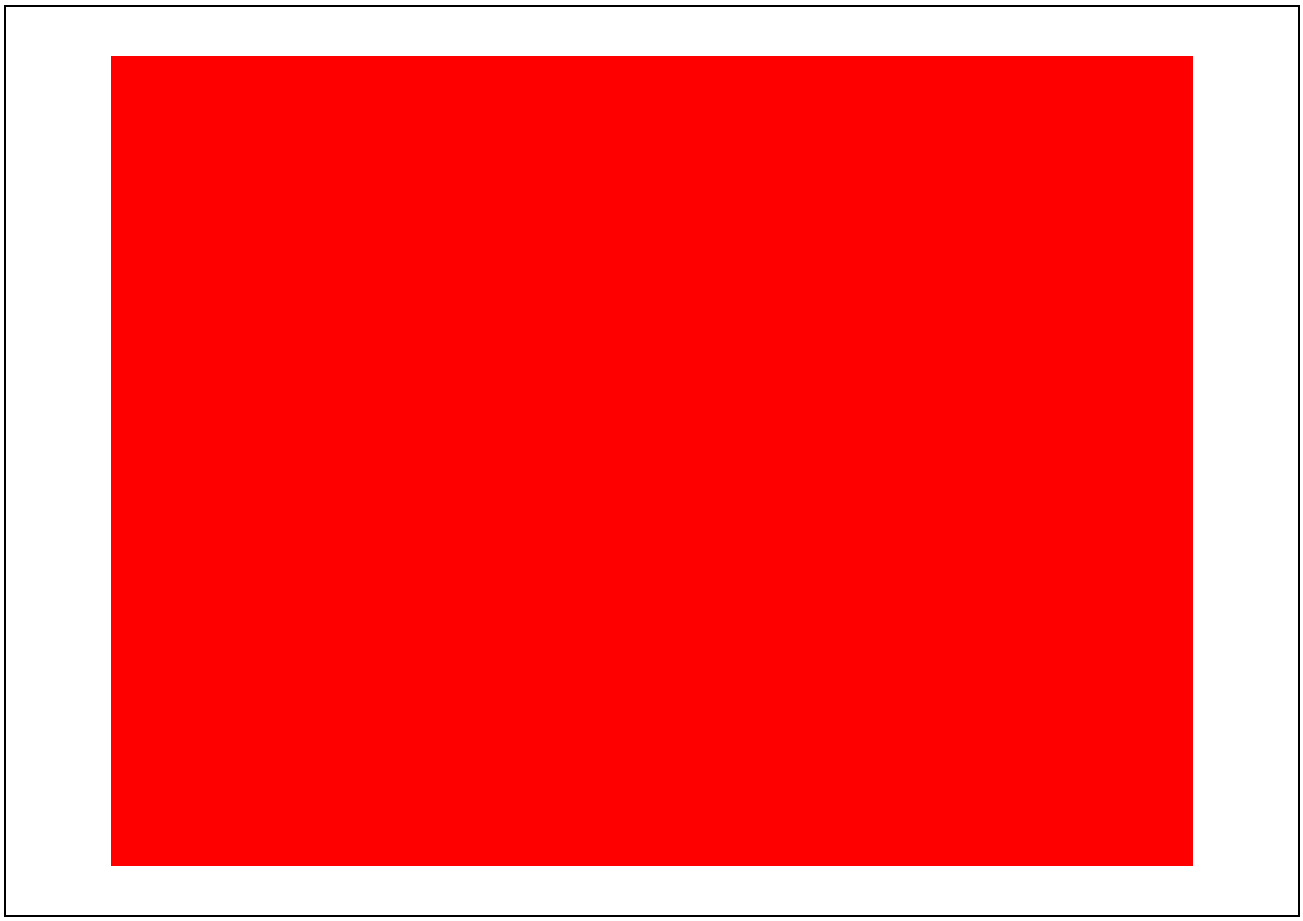


Figure 9.2-34 Generated Image Data Example (in the Case of Monochrome Generation)

9.2.4.3.5 VC/Data Type Selector (for Image Processing)

Virtual channels from VC0 to VC15 are supported for video input data input from the MIPI CSI2 block. On the other hand, the number of channels which can be handed in image processing is up to four. This block handles assignment of the selected virtual channels (SVCs).

Only one data type can be handled by each SVC and the data types are selected in the ICnIPMC_C0 to 3.INF[5:0] registers.

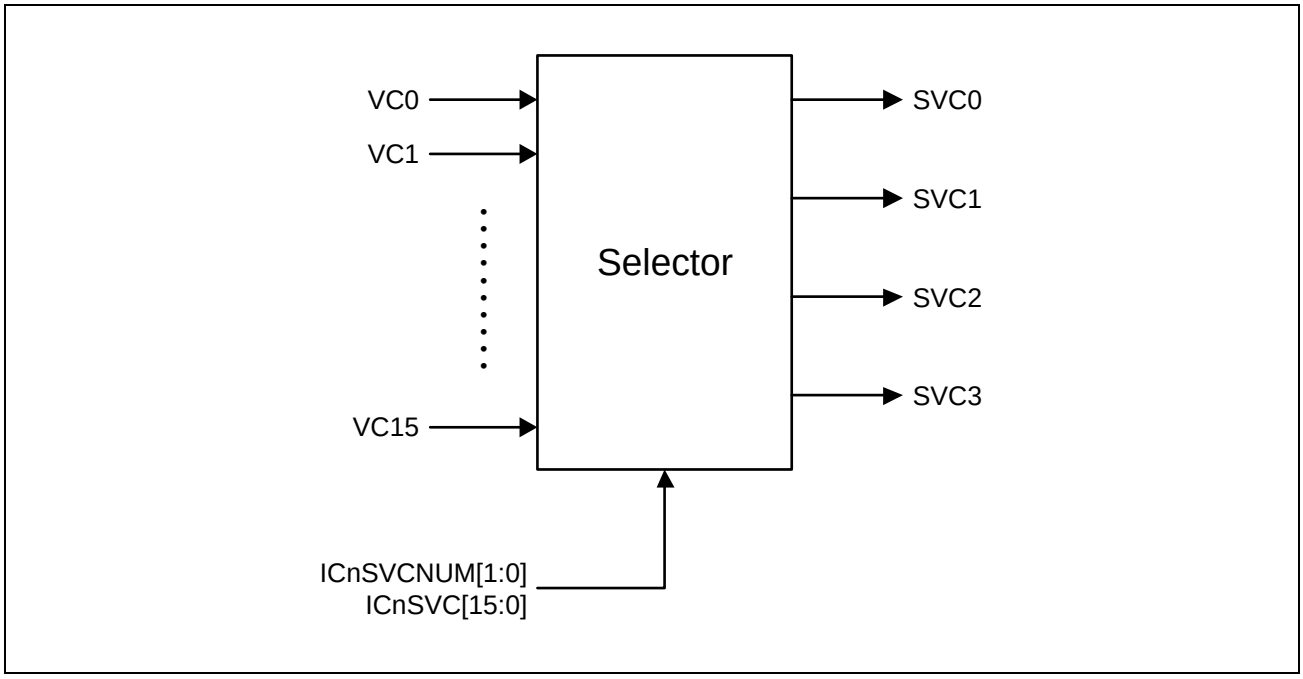


Figure 9.2-35 VC Selector Image

9.2.4.3.6 8-bit Expansion

(1) RGB444/RGB555/RGB565/RGB666 to RGB888

This block handles internal processing of RGB with 8 bits for each color. Inputs with fewer color bits are expanded to 8 bits. The conversion of inputs in the RGB444 and RGB555 and in the RGB565 and RGB666 formats to RGB888 is shown in **Table 9.2-19** and **Table 9.2-20**, respectively.

Table 9.2-19 RGB444/RGB555 to RGB888 Conversion

RGB444	RGB888
R[3:0]	{R[3:0],R[3:0]}
G[3:0]	{G[3:0],G[3:0]}
B[3:0]	{B[3:0],B[3:0]}

RGB555	RGB888
R[4:0]	{R[4:0],R[4:2]}
G[4:0]	{G[4:0],G[4:2]}
B[4:0]	{B[4:0],B[4:2]}

Table 9.2-20 RGB565/RGB666 to RGB888 Conversion

RGB565	RGB888
R[4:0]	{R[4:0],R[4:2]}
G[5:0]	{G[5:0],G[5:4]}
B[4:0]	{B[4:0],B[4:2]}

RGB666	RGB888
R[5:0]	{R[5:0],R[5:4]}
G[5:0]	{G[5:0],G[5:4]}
B[5:0]	{B[5:0],B[5:4]}

(2) RAW6/7 to RAW8

This block handles internal processing of RAW with 8 bits for each color. Inputs with fewer color bits are expanded to 8 bits. The conversion of inputs in the RAW6 and RAW7 formats to RGB888 is shown in **Table 9.2-21**.

Table 9.2-21 RAW6/7 to RAW8 Conversion

RAW6	RAW8
RAW[5:0]	{RAW[5:0],RAW[5:4]}

RAW7	RAW8
RAW[6:0]	{RAW[6:0],RAW[6]}

9.2.4.3.7 Image Clipping

Ranges for the clipping of input images are controlled by register settings. Specifically, the ranges of clipping for each SVC are set through the following register settings.

(Horizontal direction)

- Image clipping start pixel registers: ICnSPPrC_C0~3.SPPrC[12:0]
- Image clipping end pixel registers: ICnEPPrC_C0~3.EPPrC[12:0]

(Vertical direction)

- Image clipping start line registers: ICnSLPrC_C0~3.SLPrC[12:0]
- Image clipping end line registers: ICnELPrC_C0~3.ELPrC[12:0]

While the part of the image with the size after clipping (the gray part below) is being processed, the value of register ICnMS.AV0/1/2/3 = 1b. For the rest of the image, ICnMS.AV0/1/2/3 = 0b. When subsampling is done, operation proceeds in the same way for the subsampled images.

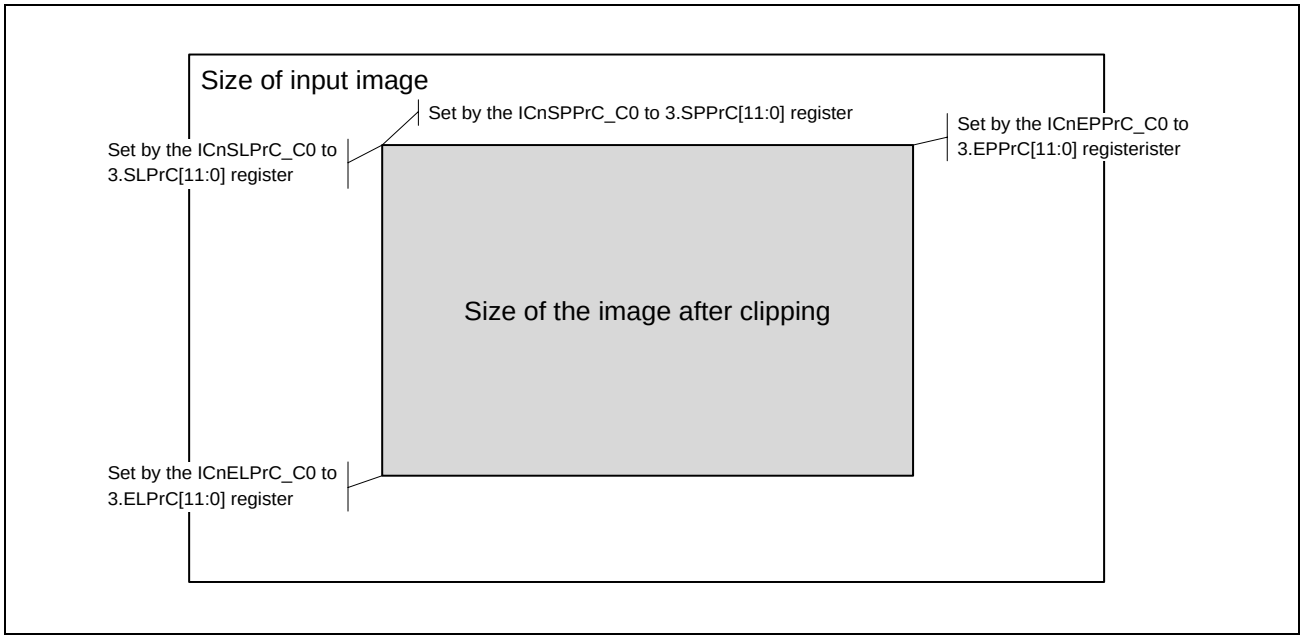


Figure 9.2-36 Image of Image Clipping

9.2.4.3.8 Frame Subsampling

Image frames are subsampled with the use of the registers which set input images. The settings of the following bit fields select the number of frames to be transferred in each round of transfer.

- Frame subsampling control register: ICnDEC.FRMDEC[3:0] (common to all SVCs)

Operation when the setting of ICnDEC.FRMDEC[3:0] = 2h is as follows:

Images are transferred in units of (the above register value + 1) frames. The state of ICnMS.CA0/1/2/3 is 1b when an image which is not to be eliminated by subsampling is being processed, as shown below.

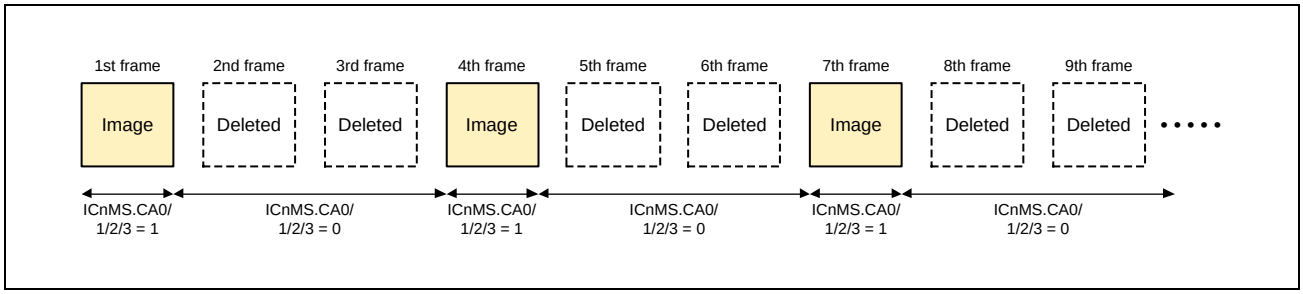


Figure 9.2-37 Frame Subsampling when ICnDEC.FRMDEC[3:0] = 2h

9.2.4.3.9 Data Arrangers 1 and 2

When image data are input and output, UV values expressed in negative form in YUV (YCbCr) data are converted. The data arranger 1 input side and data arranger 2 output side settings can be made by using the ICnDTVP.IBINSEL and ICnDMR.OBINSEL registers, respectively.

Settings for data arranger 1 must suit the specifications of the sensor. Data arranger 1 supports 8 or 10 bits depending on the number of bits of input data. Settings for data arranger 2 must suit the specifications of the destination for output. Data arranger 2 supports 8 bits depending on the output data. YUV of MIPI CSI-2 is usually used in offset binary. However, this depends on the sensor, and it may be output as two's complements. **Table 9.2-22** shows an example of processing when the input is 8 bits. Data arranger 1 also has 10-bit input, and when 10-bit input is used, it can be converted to 10-bit offset binary/2's complement as well as 8-bit.

Table 9.2-22 Input Formats and Corresponding Forms of Image Processing (at 8-bit input)

UV Value (Hex)	Offset Binary (Dec) Register Settings: Data Arranger 1: ICnDTVP.IBINSEL = 0b Data Arranger 2: ICnDMR. OBINSEL =0b	Two's Complements (Dec) Register Settings: Data Arranger 1: ICnDTVP.IBINSEL = 1b Data Arranger 2: ICnDMR. OBINSEL = 1b
FFh	127	−1
FEh	126	−2
…	…	…
81h	1	−127
80h	0	−128
7Fh	−1	127
…	…	…
01h	−127	1
00h	−128	0

9.2.4.3.10 Data Clipping

Data clipping is performed on the YUV (YCbCr) format. To make input data for RGB conversion compliant with the ITU-R BT.601 standard, whether or not to clip values less than or equal to 16 and greater than or equal to 240 in 8-bit encoding (or less than or equal to 64 and greater than or equal to 960 in 10-bit encoding) is specified. This setting is made by using the ICnDTVP.CLP[1:0] register.

9.2.4.3.11 Demosaicing

This function is only valid on input data of the RAW type. Its operation is on the assumption that the RAW data are Bayer data and generates pixel data that are interpolated from the RAW data.

The demosaicing function uses a bilinear method. Of the peripheral 3 × 3 pixels at the position to be calculated, 2 or 4 pixels are used for interpolation processing. The interpolation formula is given below.

The following describes the method of calculating the RGB components (R22, Gb22/Gr22, B22) of the pixels at the center position.

In the case of RAW10/12/14/16/20, the eight higher-order bits are used for the calculation.

(1) RAW20 to RAW16

For a RAW20 input, the higher-order bits of RAW20 are extracted and the lower-order bits are truncated in preprocessing for demosaicing and statistics.

RAW20 → RAW8 (RAW[19:4])

(2) Calculating the Gb22 data portion

Figure 9.2-38 shows the method of interpolation calculation for pixel data (R/G/B) of Gb22.

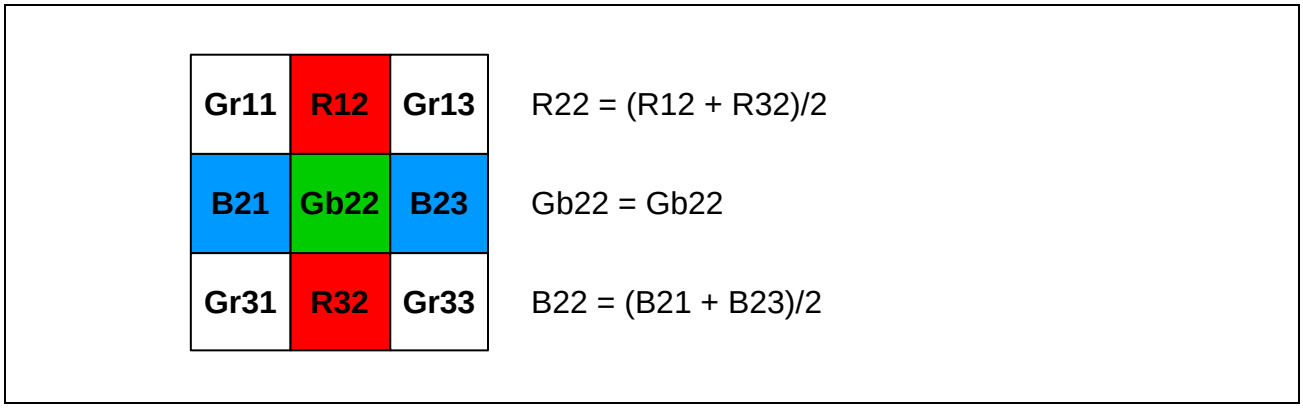


Figure 9.2-38 Calculation of the Gb22 Data Portion

(3) Calculating the B22 data portion

Figure 9.2-39 shows the method of interpolation calculation for pixel data (R/G/B) of B22.

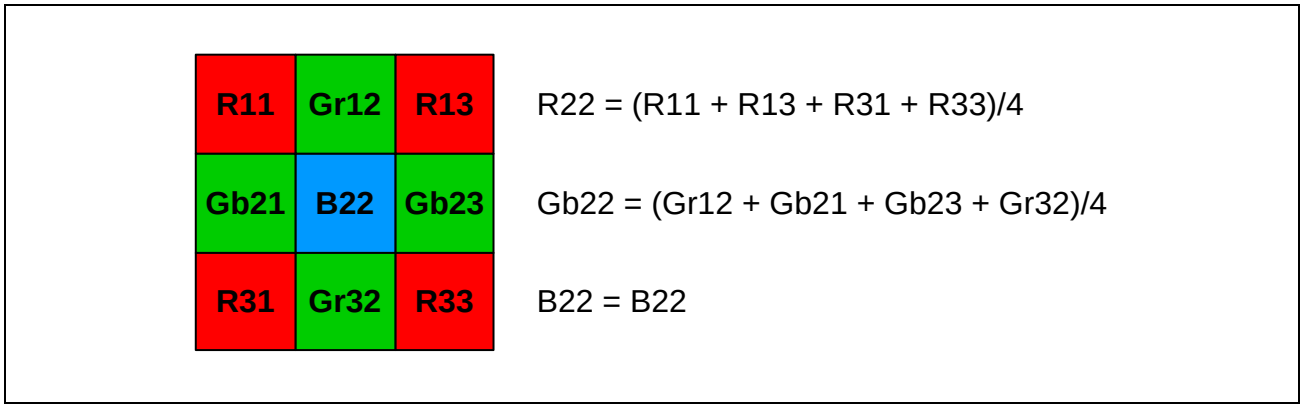


Figure 9.2-39 Calculation of the B22 Data Portion

(4) Calculating the R22 data portion

Figure 9.2-40 shows the method of interpolation calculation for pixel data (R/G/B) of R22.

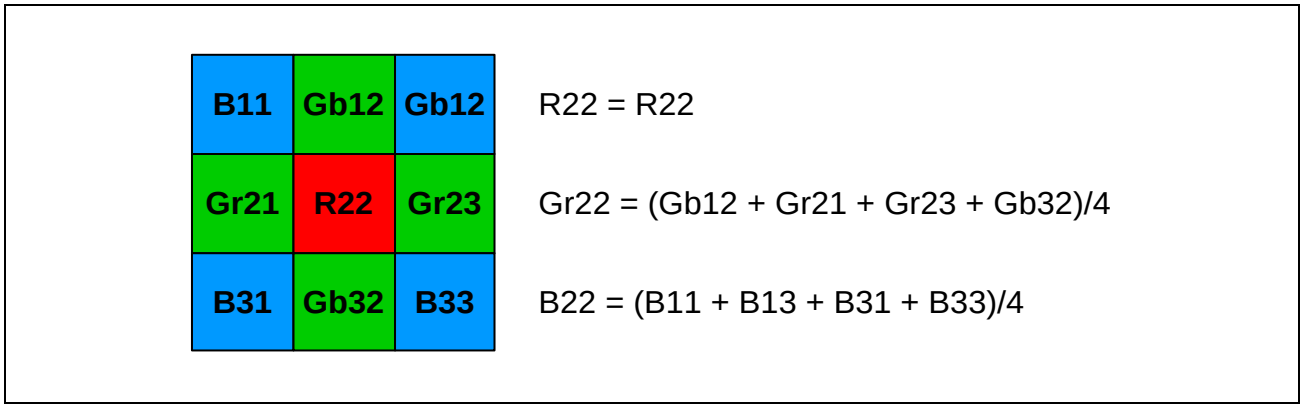


Figure 9.2-40 Calculation of the R22 Data Portion

(5) Calculating the Gr22 data portion

Figure 9.2-41 shows the method of interpolation calculation for pixel data (R/G/B) of G22.

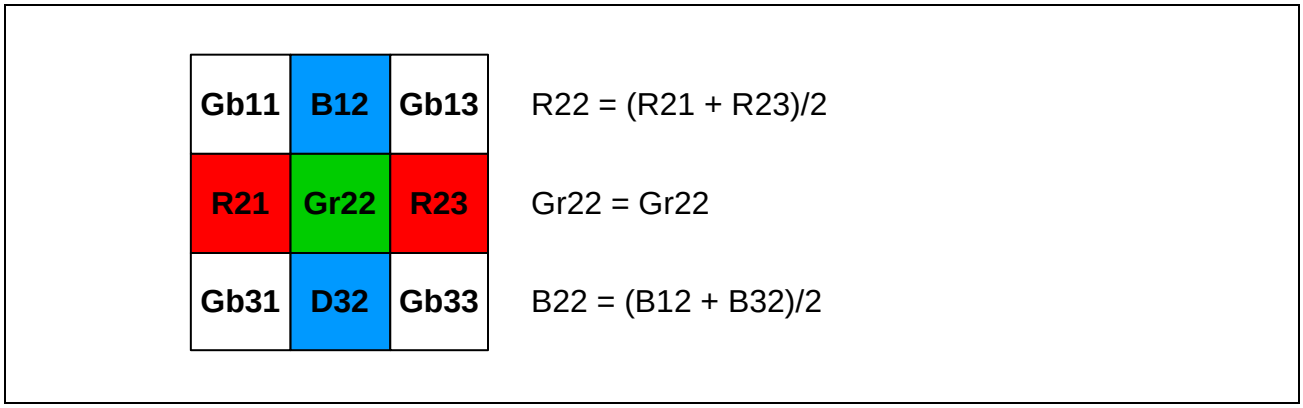


Figure 9.2-41 Calculation of the Gr22 Data Portion

9.2.4.3.12 Linear Matrix

Linear matrix processing involves calculation of the matrix of R', G', and B' (8 bits per color) by using the registers listed below to set the following coefficients for R, G, and B (8 bits per color) after demosaicing as the previous stage.

Gof = Register ICnLMXOF.GOF[7:0]
Rof = Register ICnLMXOF.ROF[7:0]
Bof= Register ICnLMXOF.BOF[7:0]

Rr = Register ICnLMXRC1.RR[12:0]
Rb = Register ICnLMXRC2.RB[12:0]
Rg = Register ICnLMXRC2.RG[12:0]

Gr = Register ICnLMXGC1.GR[12:0]
Gb = Register ICnLMXGC2.GB[12:0]
Gg = Register ICnLMXGC2.GG[12:0]

Br = Register ICnLMXBC1.BR[12:0]
Bb = Register ICnLMXBC2.BB[12:0]
Bg = Register ICnLMXBC2.BG[12:0]

$$R' = \frac{Rr \cdot (R + Rof)^{*1} + Rg \cdot (G + Gof)^{*1} + Rb \cdot (B + Bof)^{*1}}{1024}$$

$$G' = \frac{Gr \cdot (R + Rof)^{*1} + Gg \cdot (G + Gof)^{*1} + Gb \cdot (B + Bof)^{*1}}{1024}$$

$$B' = \frac{Br \cdot (R + Rof)^{*1} + Bg \cdot (G + Gof)^{*1} + Bb \cdot (B + Bof)^{*1}}{1024}$$

Note 1. When the result of addition is a negative value, it is converted to 0.

9.2.4.3.13 Color Space Conversion

(1) RGB to YCbCr

RGB (8/10 bits) data to which image clipping, frame subsampling, or demosaicing has been applied are expanded to 12 bits and then converted to 10-bit YCbCr. Calculations equivalent to those specified in ITU-R BT.601 or ITU-R BT.709 are performed by setting coefficients through registers.

The calculations proceed according to the coefficients and expressions set in the following registers.

YCLRP = Register ICnYCCR1.YCLRP[12:0]
YCLBP = Register ICnYCCR2.YCLBP[12:0]
YCLGP = Register ICnYCCR2.YCLGP[12:0]
YCLSFT = Register ICnYCCR3.YCLSFT[4:0]
YCLAP = Register ICnYCCR3.YCLAP[11:0]

CBCLRP = Register ICnCBCCR1.CBCLRP[12:0]
CBCLBP = Register ICnCBCCR2.CBCLBP[12:0]
CBCLGP = Register ICnCBCCR2.CBCLGP[12:0]
CBCLSFT = Register ICnCBCCR3.CBCLSFT[4:0]
CBCLAP = Register ICnCBCCR3.CBCLAP[11:0]

CRCLRP = Register ICnCRCCR1.CRCLRP[12:0]
CRCLBP = Register ICnCRCCR2.CRCLBP[12:0]
CRCLGP = Register ICnCRCCR2.CRCLGP[12:0]
CRCLSFT = Register ICnCRCCR3.CRCLSFT[4:0]
CRCLAP = Register ICnCRCCR3.CRCLAP[11:0]

$$Y = \left(\frac{YCLRP \times R + YCLGP \times G + YCLBP \times B}{2^{YCLSFT}}\right) + YCLAP$$

$$Cb = \left(\frac{CBCLRP \times R + CBCLGP \times G + CBCLBP \times B}{2^{CBCLSFT}}\right) + CBCLAP$$

$$Cr = \left(\frac{CRCLRP \times R + CRCLGP \times G + CRCLBP \times B}{2^{CRCLSFT}}\right) + CRCLAP$$

For ITU-R BT.601 (8 bits): (Same as the default values)

$Y = 0.257 \times R + 0.504 \times G + 0.098 \times B + 16$
$Cb = -0.148 \times R - 0.291 \times G + 0.439 \times B + 128$
$Cr = 0.439 \times R - 0.368 \times G - 0.071 \times B + 128$

[Register settings]

YCLRP = 0107h
YCLBP = 0064h
YCLGP = 0204h
YCLSFT = 0Ah
YCLAP = 100h

CBCLRP = 1F68h
CBCLBP = 01C2h
CBCLGP = 1ED6h

CBCLSFT = 0Ah
CBCLAP = 800h

CRCLRP = 01C2h
CRCLBP = 1FB7h
CRCLGP = 1E87h
CRCLSFT = 0Ah
CRCLAP = 800h

(2) YCbCr to RGB

YCbCr (8/10 bits) data which have been arranged and clipped are expanded to 12 bits (by appending zeros as the lower-order bits) and then converted to 10-bit RGB. Calculations equivalent to those specified in ITU-R BT.601 or ITU-R BT.709 are performed by setting coefficients through registers.

The calculations proceed according to the coefficients and expressions set in the following registers.

YMUL = Register ICnCSCC1.YMUL[13:0]
YSUB = RegisterICnCSCC2.YSUB[11:0]
RCRMUL = Register ICnCSCC2.CSUB[11:0]
GCRMUL = Register ICnCSCC3.RCRMUL[13:0]
GCBMUL = Register ICnCSCC3.GCRMUL[13:0]
BCBMUL = Register ICnCSCC4.GCBMUL[13:0]
CSUB = Register ICnCSCC4.BCBMUL[13:0]

$$R = \frac{(YMUL \times (Y - YSUB)^{*1} + RCRMUL \times (Cr - CSUB)}{4096}$$

$$G = \frac{YMUL \times (Y - YSUB)^{*1} - GCRMUL \times (Cr - CSUB) - GCBMUL \times (Cb - CSUB)}{4096}$$

$$B = \frac{YMUL \times (Y - YSUB)^{*1} + BCBMUL \times (Cb - CSUB)}{4096}$$

Note 1. A negative value is rounded up to 0.

For ITU-R BT.601 (8 bits): (Same as the default values)

R = 1.164 × (Y − 16) + 1.596 × (Cr − 128)
G = 1.164 × (Y − 16) − 0.813 × (Cr − 128) − 0.392 × (Cb − 128)
B = 1.164 × (Y − 16) + 2.017 × (Cb − 128)

[Register settings]

YMUL = 129Fh
YSUB = 100h
RCRMUL = 1989h
GCRMUL = 0D02h
GCBMUL = 0645h
BCBMUL = 2045h
CSUB = 800h

9.2.4.3.14 LUT (Lookup Table)

This is used to convert 10-bit data after color space conversion (YCbCr to RGB or RGB to YCbCr) into 8-bit data.

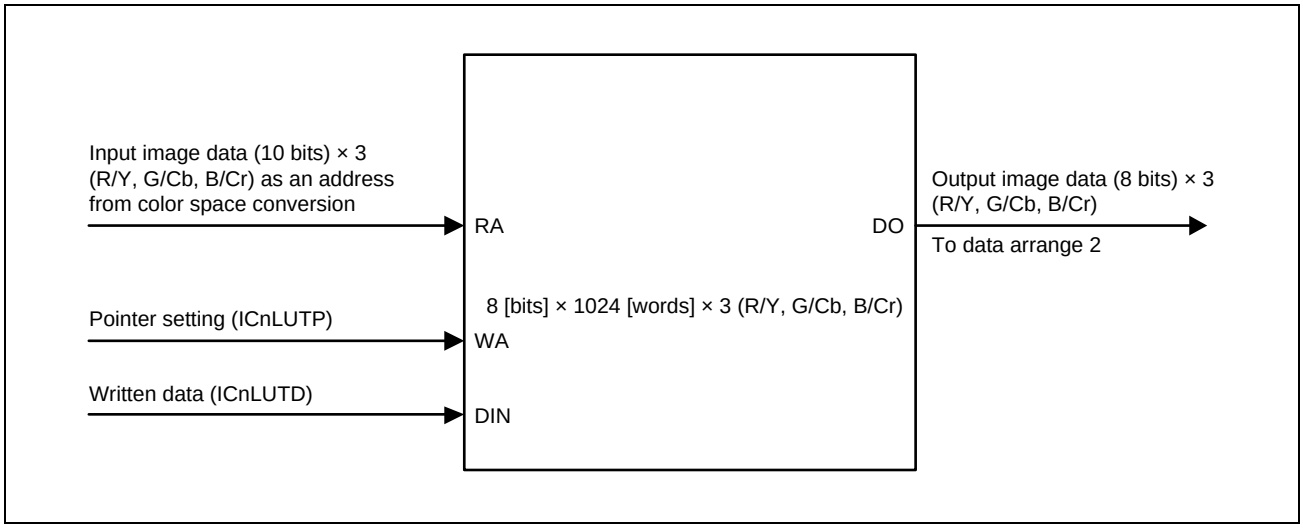


Figure 9.2-42 Lookup Table (LUT)

(1) Pointer Access

The pointer is incremented every time the ICnLUTD register is written.

The incremented value can be read from the ICnLUTP register. When the ICnLUTD register is read, the pointer is not incremented. The pointer must be set immediately before the register is read.

Follow the procedure below to access the lookup table. See **9.2.5.1 Starting Reception for the MIPI CSI-2 Input**.

The lookup table cannot be accessed from the CPU in normal mode.

(2) Usage Notes on LUT

1. Set the lookup table only after the CRU has stopped its operation.
2. Rewrite the entire lookup table before use because the initial values are not guaranteed after the power is turned on.
3. Data to be set in the lookup table must be restricted to the range that is compatible with the input bit width.

9.2.4.3.15 YC Post Processing

(1) YCbCr422 to 420

This function thins out the even-line CbCr data for the image processed in the previous stage and outputs only Y. Averaging or any other calculations are not performed. This conversion can be set by ICnDMR.YCMODE[2:0]=100b/101b/110b.

(2) Y/C Separation

This function separates the Y and C (CbCr) components for storage in two separate memory banks of the DRAM. For details, see **9.2.4.3.3 Transfer to DRAM**.

(3) Y Only

This function changes the method of storage in the DRAM. For details, see **9.2.4.3.3 Transfer to DRAM**.

9.2.4.3.16 VC/Data Type Selector (for Non-Image Processing)

When non-image processing is selected, set the target SVC and data types. When selecting the number of channels of SVC to be used, use the ICnSVCNUM.SVCNUM[1:0] register to set how many channels are used from among SVC0 to 3. Selection of SVCs with the ICnSVC.SVC0~3[3:0] registers determines the allocation of which channels are to be selected from among VC0 to VC15 for SVC0 to 3.

In data type selection, the required type can be selected with the ICnNIPDT_C0~3H/L registers and selection of several types is also possible (see **Figure 9.2-43**).

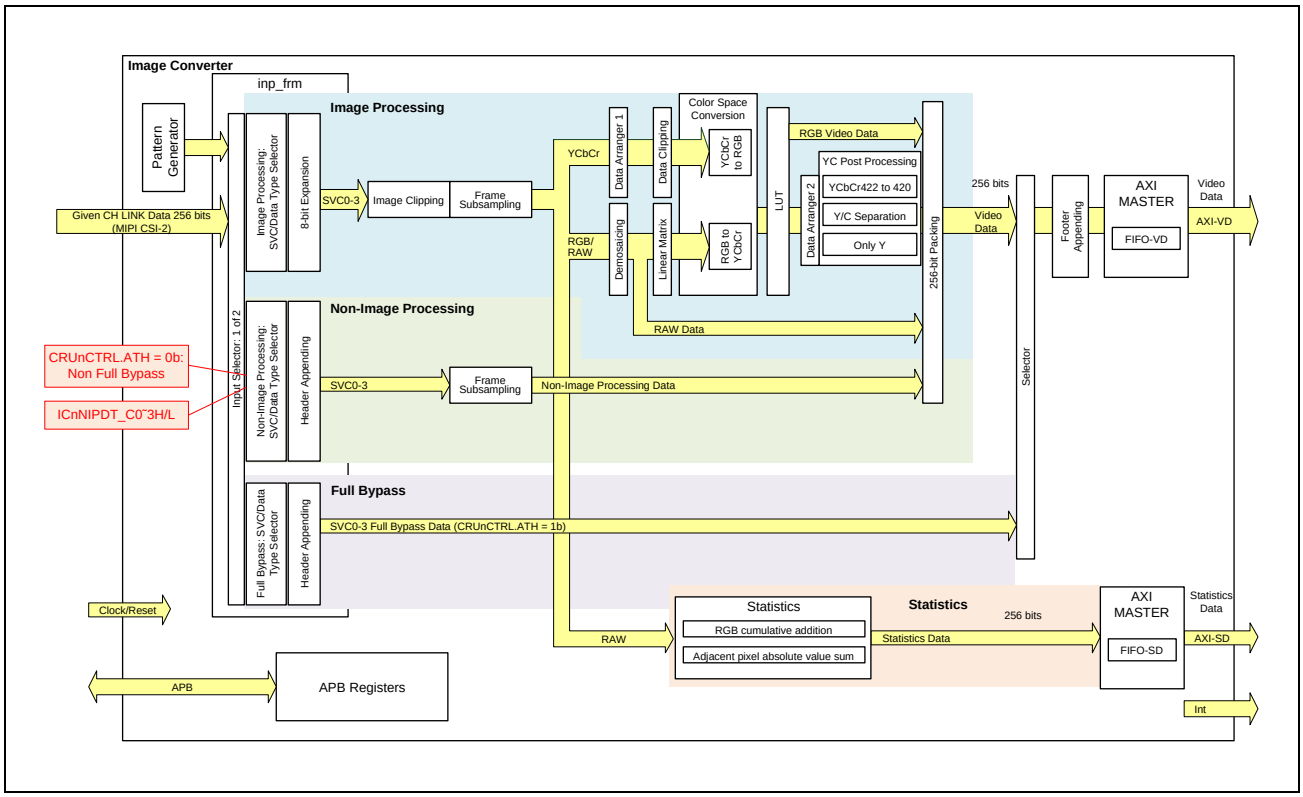


Figure 9.2-43 Example of VC Specification and Selection of Several Data Types for Non-Image Processing

9.2.4.3.17 Full Bypass function

Figure 9.2-44 shows an example of the settings for full bypass. These settings are used when you want to store images in memory without processing them within the CRU (for example, when you want to process images with the unit).

The full bypass function handles the transfer of data of the data type selected for the specified single VC from the video data bus to memory. Data are not corrected at all and are transferred in the recommended memory storage format for MIPI CSI-2.

The data types are not limited to one and several types can be selected by the settings of ICnNIPDT_C0~3H/L.

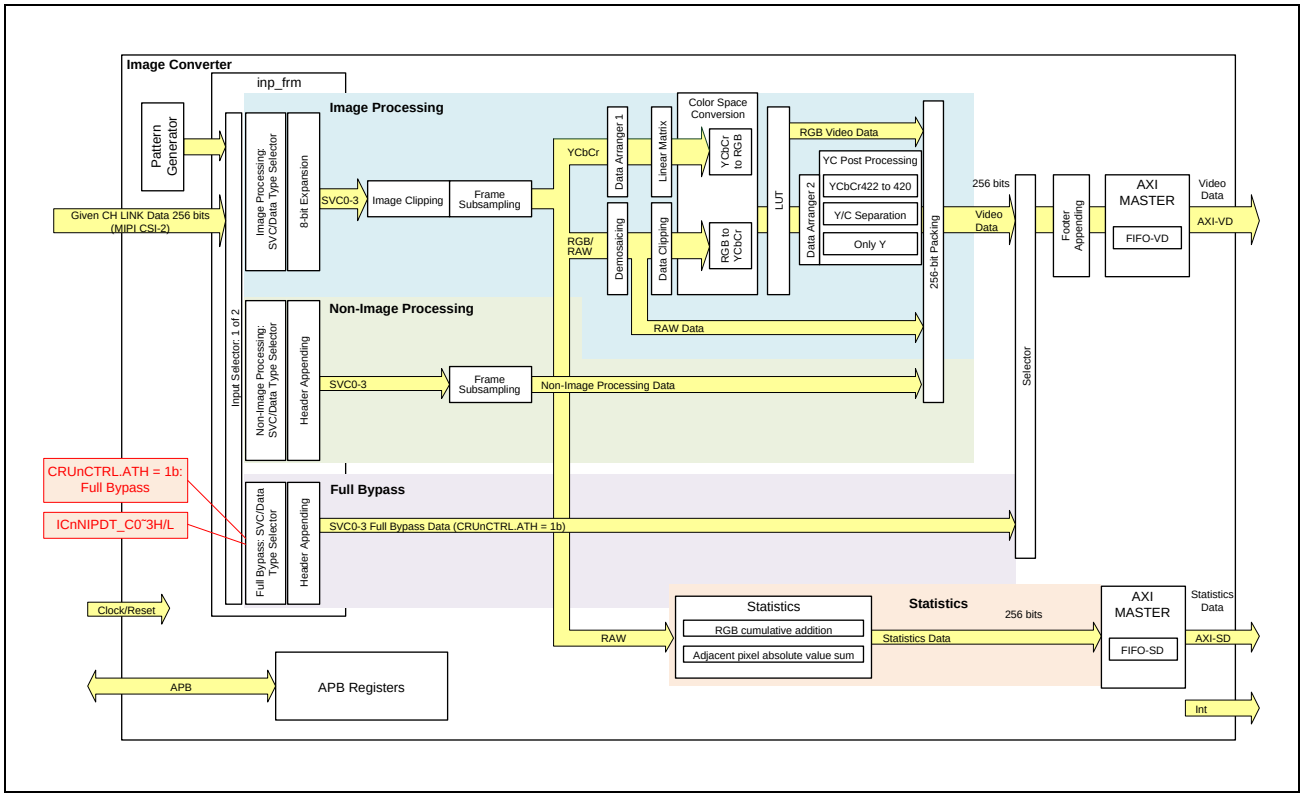


Figure 9.2-44 Example of Settings for Full Bypass

9.2.4.3.18 Statistics

This function is limited to a RAW data input through SVC0. This module divides one image vertically and horizontally into blocks (see **Figure 9.2-45**). The statistical information within this block is transferred to memory and the image is adjusted by weighting the information of every block by the CPU and providing feedback to the external video state.

The number of block divisions can be determined by the setting of register ICnSTIC1.STUNIT[1:0].

The statistical information of each block has two types and their respective purposes are as follows.

1) In-block accumulation

Accumulation of each color component, R, G, and B, is performed within each block.

On the CPU side, the weighting is applied per block based on the information from each block, and the obtained evaluation values can be used for exposure control, etc. from the next frame.

The weighting could in general be center-weighted allocation, weighted allocation according to the subject, and so on. If the subject is judged to be dark, exposure control is applied to widen the aperture, and if the subject is judged to be bright, the aperture is made narrower.

Additionally, each color component among the R, G, and B components is adjusted and the latter stage performs operations in such a way that the B component is reduced, for example, in video which appears bluish in white regions.

This adjustment of each color includes changing the coefficients of the sensor or the linear matrix as the latter stage.

2) In-block sum of absolute difference values

Focusing on the G component within the block, the overall value for the differences from the adjacent parts of the image is extracted and accumulation is performed within the block.

On the CPU side, the weighting is applied per block based on the information from each block, and the obtained evaluation values can be used for focus adjustment, etc. from the next frame.

The weighing could in general be center-weighted allocation, weighted allocation according to the subject, and so on. If the obtained evaluation value is compared with the previous frame and if the amount continues to be reduced, the evaluation value is increased to adjust the focus, for example, by adjusting the distance between the lens and the sensor toward the opposite direction.

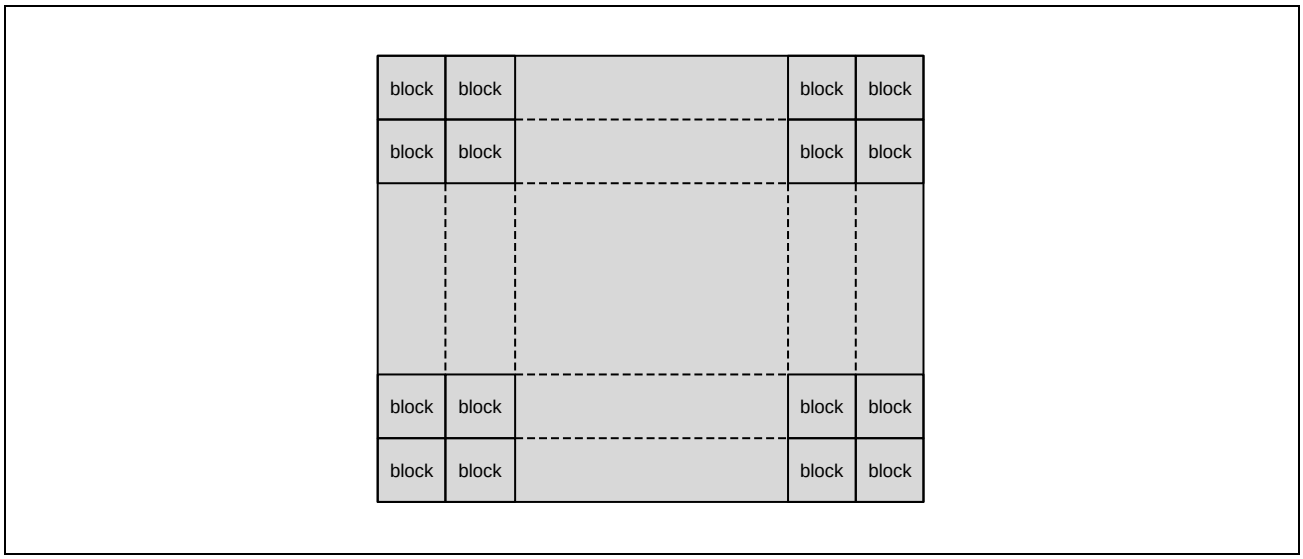


Figure 9.2-45 Image of Dividing an Image into Blocks Vertically and Horizontally

Figure 9.2-46 is a block diagram of statistics. The result of AE/AWB statistical calculations for per block is 8 bits × 3 = 3 bytes, respectively, and that of the AF statistical calculations is 1 byte for G only, so a total of 4 bytes of data are output.

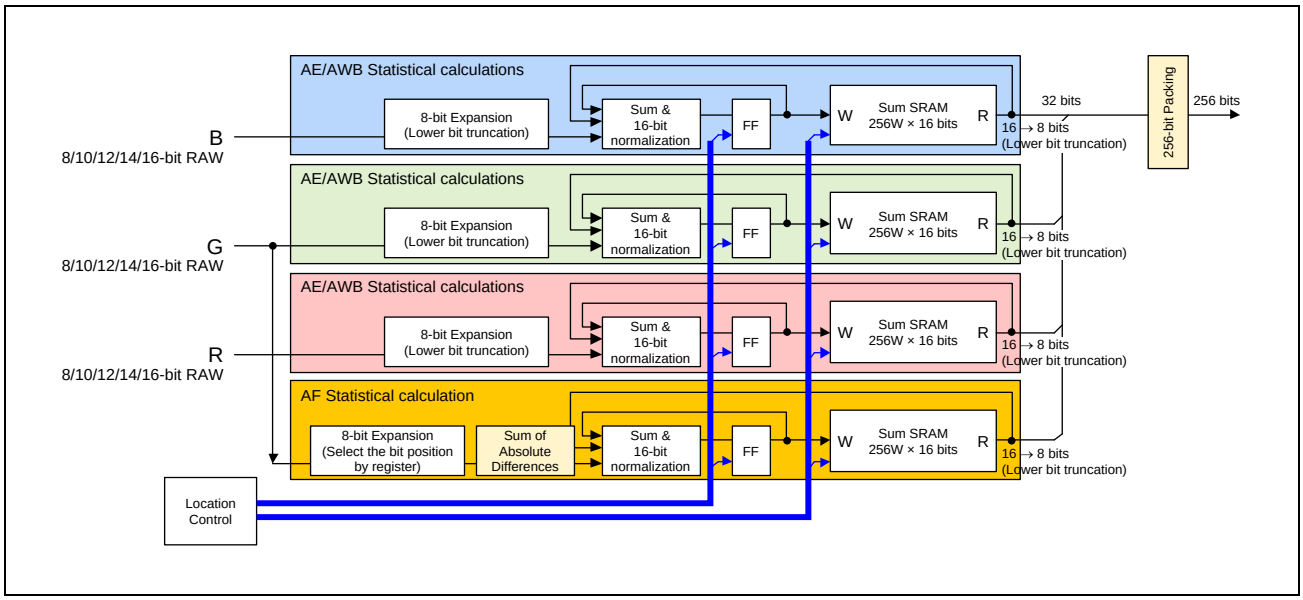


Figure 9.2-46 Statistics Block Diagram

(1) RGB cumulative addition

The STUNIT[1:0] bit fields in each of the ICnSTIC1 registers are used to select the pixel (block) sizes for statistical calculations from among four values, 16×16, 32×32, 64×64, and 128×128. The target pixels when the setting is 16×16 are described in **(a) Target pixels when the block size is selected as 16×16 (ICnSTIC1.STUNIT[1:0] = 0b)** and the methods of calculation for each block-size setting are described in **(b) Methods of calculation for each block-size setting**.

(a) Target pixels when the block size is selected as 16×16 (ICnSTIC1.STUNIT[1:0] = 0b)

G component: 128 pixels are present in the area of 16×16
R and B components: 32 pixels are present in the area of 16×16

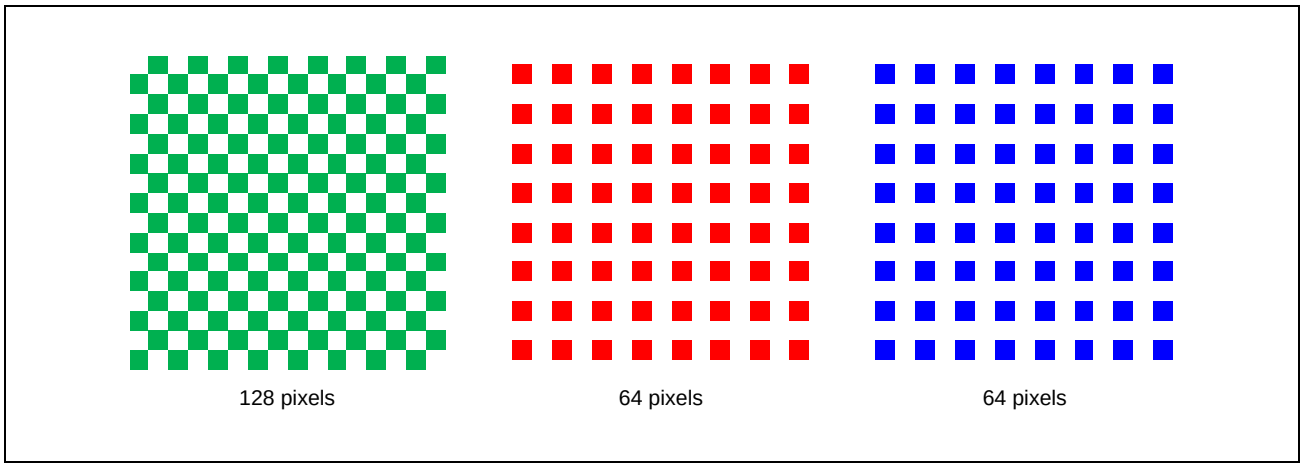


Figure 9.2-47 Target Pixels when the Block Size is Selected as 16×16

(b) Methods of calculation for each block-size setting

[G component]

- When the block size is 16×16: 8 bits × 128 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 32×32: 8 bits × 512 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 64×64: 8 bits × 2048 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 128×128: 8 bits × 8192 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.

[R and B components]

- When the block size is 16×16: 8 bits × 64 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 32×32: 8 bits × 256 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 64×64: 8 bits × 1024 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 128×128: 8 bits × 4096 pixels are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.

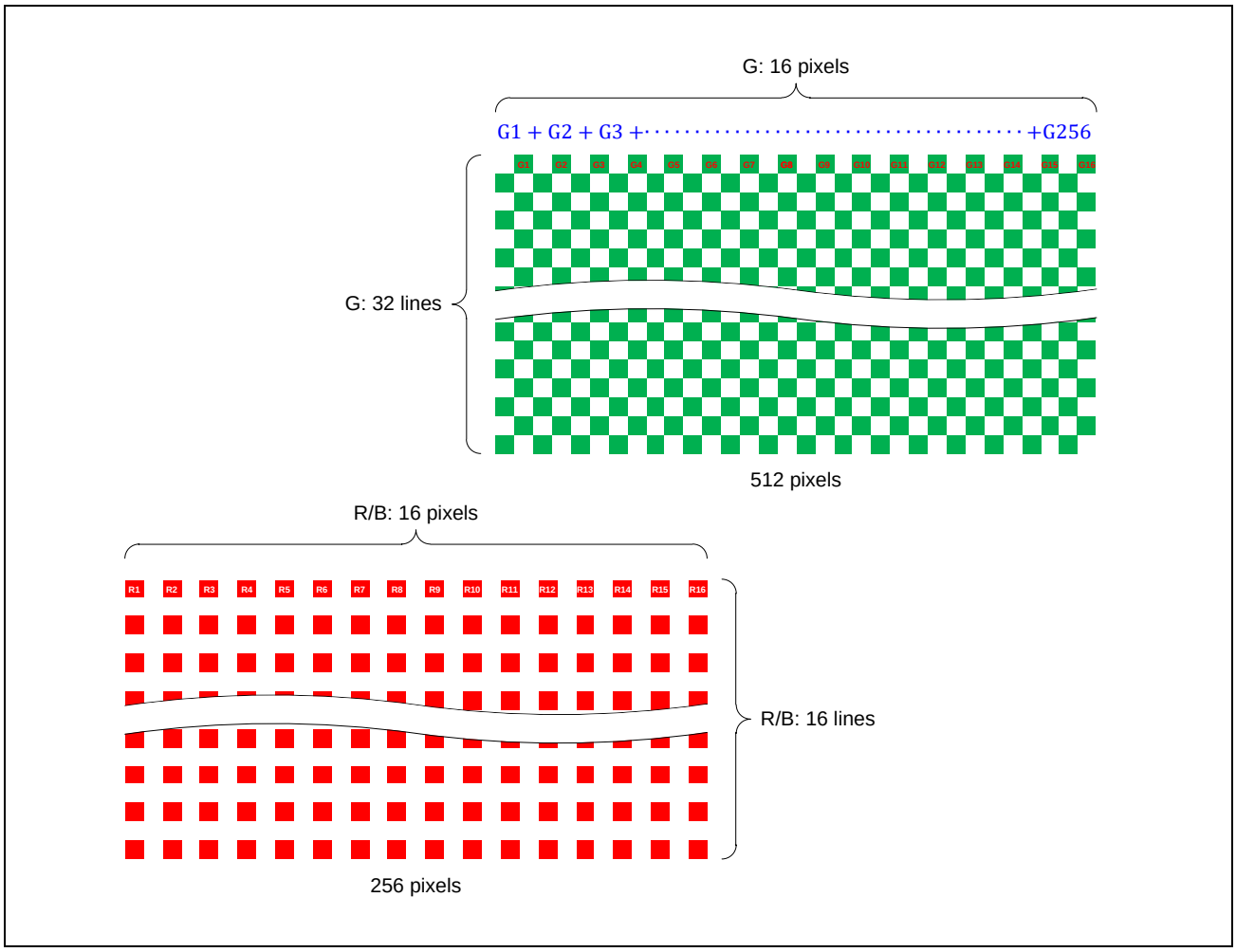


Figure 9.2-48 Illustration Diagram of Accumulation within the Block when the Block Size is 32×32

(2) Sum of absolute values for adjacent pixels

Set the area for which the sum of absolute values for adjacent pixels is taken with the use of register ICnSTIC1.STUNIT[1:0] to make the calculations in the following ways.

- **In the case of 16×16 pixel unit operations (ICnSTIC1.STUNIT[1:0] = 0b):**
  - For 8-bit data: Up to 16 bits
  - For 10- to 16-bit data: A sum-of-absolute-differences (SAD) operation is performed with the eight higher-order bits (default).
  - Addition can also be performed with the eight lower-order bits by shifting the bits by setting register ICnSTIC1.STSADPOS[3:0].

- **Methods of calculation for the setting area:**

[G component calculation]

- When the block size is 16×16: Adjacent differences × 112 are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 32×32: Adjacent differences × 480 are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 64×64: Adjacent differences × 1984 are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.
- When the block size is 128×128: Adjacent differences × 8064 are accumulated: The eight higher-order bits are output after accumulation and normalization with a 16-bit precision.

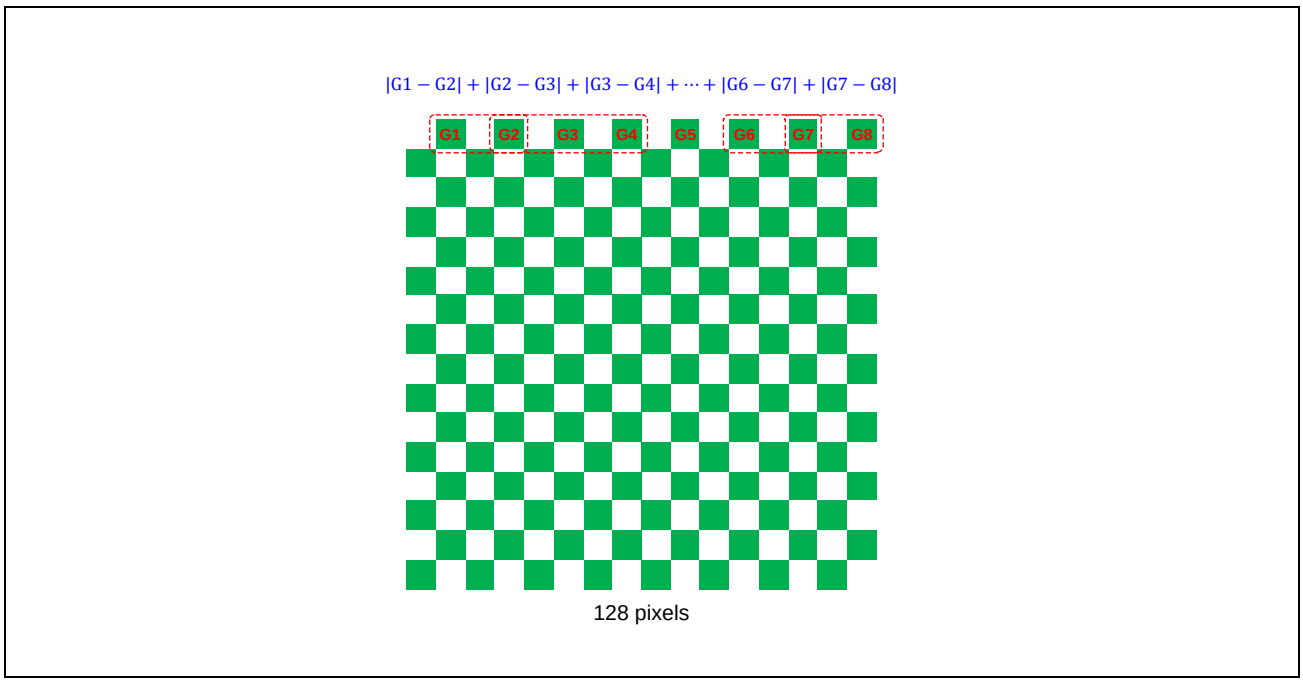


Figure 9.2-49 Illustration Diagram of Sum of Absolute Values for Adjacent Pixels when the Block Size is 16×16

(3) Statistics data generated image

The following describes the amounts of data in the operations when a 1920×1080 pixel area is set as 16×16 blocks and as 128×128 blocks for the statistics function. The numerals in the figures below represent the block numbers. In processing of the right end or bottom of the image, if the input image size is greater than the size specified as the number of blocks in the horizontal direction, that difference does not fall within the target range for arithmetic processing. This also applies in the vertical direction.

- **In the case of operations for a 1920×1080 pixel area as 16×16 blocks:**

As shown in **Figure 9.2-50**, the calculated values for each of the blocks are transferred sequentially. A total of 31 Kbytes of data are transferred for one frame. For the format of data for transmission, see **Table 9.2-15**.

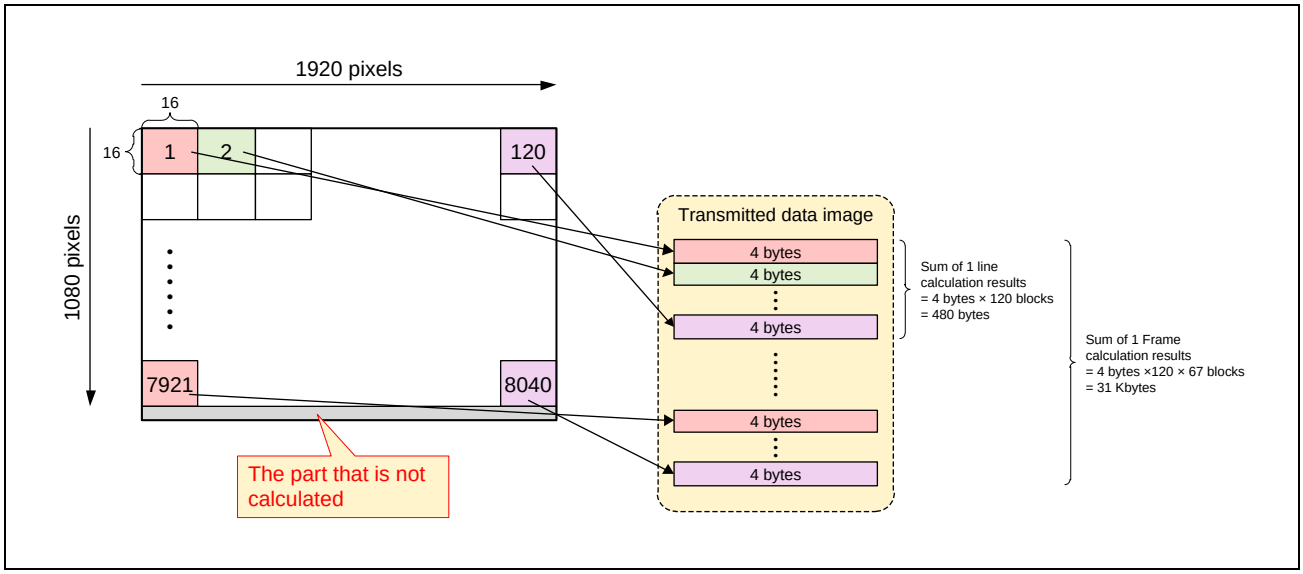


Figure 9.2-50 Illustration Diagram of the Amounts of Data in the Operations when a 1920×1080 Pixel Area is Set as 16×16 Blocks

● **In the case of operations for a 1920×1080 pixel area as 128×128 blocks:**

The calculated values for the corresponding blocks are sequentially transferred. A total of 480 bytes of data are transferred for one frame.

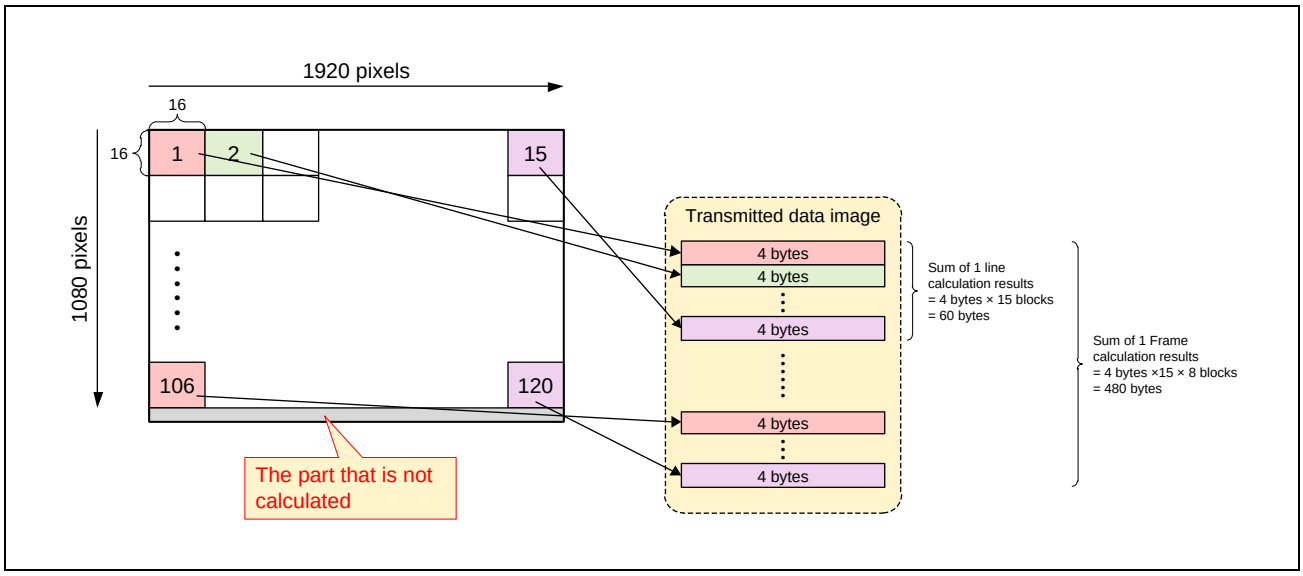


Figure 9.2-51 Illustration Diagram of the Amounts of Data in the Operations when a 1920×1080 Pixel Area is Set as 16×16 Blocks

Confirm the completion of transfer for one frame by setting the specified address interrupt described in **9.2.4.4.2 Designated address forwarding completion interrupt**.

9.2.4.4 Interrupts

9.2.4.4.1 Interrupt list

Table 9.2-23 lists the interrupt sources and registers, and the interrupt signals to the interrupt controller.

Table 9.2-23 Interrupt Sources and Registers, and Interrupt Signals (1/2) [n = 0]

Internal Signal Name	Kind	Item	Interrupt Source Status/Clear	Interrupt Enable Register Field	Generated Blocks	Conditions for Occurrence
axi_mst_err_int*4	Level	FIFO-VD overflow (video data) (for debugging)	CRUnINTS1.FOS	CRUnIE1.FOE	FIFO-VD	When Video data FIFO-VD overflows.
		SLVERR error (video data)	CRUnINTS1.SLVES	CRUnIE1.SLVEE	AXI-VD	When Video Data AXI-VD SLVERR error occurs.
		DECERR error (video data)	CRUnINTS1.DECES	CRUnIE1.DECEE	AXI-VD	When Video Data AXI-VD SLVERR error occurs.
		FIFO-SD overflow*1 (statistics data) (for debugging)	CRUnINTS1.SDFOS	CRUnIE1.SDFOE	FIFO-SD	When Statistics Data FIFO-SD overflows.
		SLVERR error*1 (statistics data)	CRUnINTS1.SDSLVES	CRUnIE1.SDSLVEE	AXI-SD	When the SLVERR error of Statistics Data AXI-SD occurs.
		DECERR error*1 (statistics data)	CRUnINTS1.SDDECES	CRUnIE1.SDDECEE	AXI-SD	When the Statistics Data AXI-SD DECERR error.
image_conv_int*2	Level	SVC0 frame start	CRUnINTS2.FS0S	CRUnIE2.FS0E	Image Clipping	When SVC0 frame starts.*3
	Level	SVC0 frame end	CRUnINTS2.FE0S	CRUnIE2.FE0E	Image Clipping	When SVC0 frame ends.*3
	Level	SVC0 scan line	CRUnINTS2.SL0S	CRUnIE2.SL0E	Image Clipping	When the SVC0 register (ICnSI_C0.SI[11:0]) matches the set number of lines.*3
	Level	SVC1 frame start	CRUnINTS2.FS1S	CRUnIE2.FS1E	Image Clipping	When SVC1 frame starts.*3
	Level	SVC1 frame end	CRUnINTS2.FE1S	CRUnIE2.FE1E	Image Clipping	When SVC1 frame ends.*3
	Level	SVC1 scan line	CRUnINTS2.SL1S	CRUnIE2.SL1E	Image Clipping	When the SVC1 register (ICnSI_C1.SI[11:0]) matches the set number of lines.*3
	Level	SVC2 frame start	CRUnINTS2.FS2S	CRUnIE2.FS2E	Image Clipping	When SVC2 frame starts.*3
	Level	SVC2 frame end	CRUnINTS2.FE2S	CRUnIE2.FE2E	Image Clipping	When SVC2 frame ends.*3
	Level	SVC2 scan line	CRUnINTS2.SL2S	CRUnIE2.SL2E	Image Clipping	When the SVC2 register (ICnSI_C2.SI[11:0]) matches the set number of lines.*3
	Level	SVC3 frame start	CRUnINTS2.FS3S	CRUnIE2.FS3E	Image Clipping	When SVC3 frame starts.*3
	Level	SVC3 frame end	CRUnINTS2.FE3S	CRUnIE2.FE3E	Image Clipping	When SVC3 frame ends.*3
	Level	SVC3 scan line	CRUnINTS2.SL3S	CRUnIE2.SL3E	Image Clipping	When the SVC3 register (ICnSI_C3.SI[11:0]) matches the set number of lines.*3
cru_vd_addr_wend	Edge	Specified address 0 write completed (video data)	—	CRUnIE2.VDADR0WE	AXI-VD	When AXI-VD bus write to the AXI-VD Bus Transfer Completion Event Address 0 Higher/Lower Register (AMnIVT0ADDRH/L) address is completed.
cru_sd_addr_wend	Edge	Specified address 0 write completed*1 (statistics data)	—	CRUnIE2.SDADR0WE	AXI-SD	When AXI-SD bus write to the AXI-SD Bus Transfer Completion Event Address 0 Higher/Lower Register (AMnSDIVT0ADDRH/L) address is completed.

Table 9.2-23 Interrupt Sources and Registers, and Interrupt Signals (2/2) [n = 0]

Internal Signal Name	Kind	Item	Interrupt Factor Status/Clear	Interrupt Enable Register Field	Generation Blocks	Conditions of Occurrence
cru_vsd_addr_wend	Level	Specified address 1 write completed (video data)	CRUnINTS2. VDADR1WS	CRUnIE2.VDADR1WE	AXI-VD	When AXI-VD bus write to the AXI-VD Bus Transfer Completion Event Address 1 Higher/Lower Register (AMnIVT1ADDRH/L) address is completed.
		Specified address 2 write completed (video data)	CRUnINTS2. VDADR2WS	CRUnIE2.VDADR2WE	AXI-VD	When AXI-VD bus write to the AXI-VD Bus Transfer Completion Event Address 2 Higher/Lower Register (AMnIVT2ADDRH/L) address is completed.
		Specified address 3 write completed (video data)	CRUnINTS2. VDADR3WS	CRUnIE2.VDADR3WE	AXI-VD	When AXI-VD bus write to the AXI-VD Bus Transfer Completion Event Address 3 Higher/Lower Register (AMnIVT3ADDRH/L) address is completed.
		Specified address 4 write completed (video data)	CRUnINTS2. VDADR4WS	CRUnIE2.VDADR4WE	AXI-VD	When AXI-VD bus write to the AXI-VD Bus Transfer Completion Event Address 4 Higher/Lower Register (AMnIVT4ADDRH/L) address is completed.
		Specified address 1 write completed*1 (statistics data)	CRUnINTS2. SDADR1WS	CRUnIE2.SDADR1WE	AXI-SD	When AXI-SD bus write to the AXI-SD Bus Transfer Completion Event Address 1 Higher/Lower Register (AMnISDIVT1ADDRH/L) address is completed.

Note 1. An interrupt from SVC is detected in image processing. It is not detected in non-image processing/full bypass.

Note 2. The ICnINTCTRL.DECINT register allows you to set whether or not interrupts are issued for subsampled frames.

Note 3. If an error interrupt occurs, a reset is required after the CRU has been stopped.

9.2.4.4.2 Designated address forwarding completion interrupt

This interrupt function is used to notify the CPU of the timing of fetching video and statistics data by using an interrupt.

Set the desired addresses in the DRAM as the destinations for data transfer by using the AMnIVT0/1/2/3/4ADDRH/L (for video data) and AMnSDIVT0ADDRH/L (for statistics data) registers. Up to four addresses can be set for video data and up to two addresses can be set for statistics data. These must be set according to the number of memory banks and the speed of reading by the CPU. An interrupt is generated upon the completion of writing to the address set by the AXI master (AXI-VD, AXI-SD).

Figure 9.2-52 shows an example when the number of memory banks is 4 for video data and the last addresses of memory banks 2 and 4 are set as the completion addresses.

The CPU reads data from the initial data in bank 1 in response to the first interrupt as a trigger and then reads up to the end of bank 2. When the next interrupt arrives, the CPU reads data from the initial data in bank 3 and then reads up to the end of bank 4. This is thereafter repeated and consecutive data can thus be read.

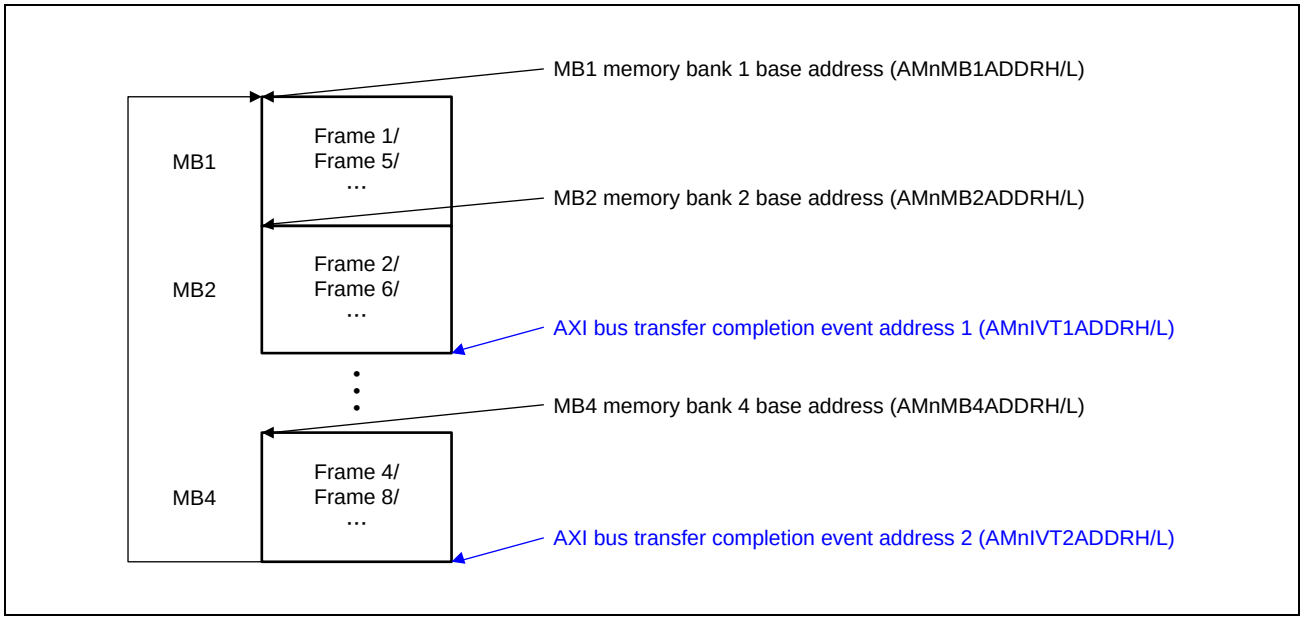


Figure 9.2-52 Example when the Number of Memory Banks is 4 for Video Data and the Last Addresses of Memory Banks 2 and 4 are Set as the Completion Addresses

9.2.4.5 Usage Notes

Use of this CRU requires caution on the following points.

(1) The following functions can only be used with SVC0 (that is, they cannot be used with SVC1 to SVC3).
- Demosaicing
- Statistics
- Changing the frame synchronization

(2) The same data (the same data type for the same VC) cannot be set for image processing and non-image processing. For example, making the following register settings is prohibited.

(Case1) This is an example of setting the same VC data type in Image Process to SVC0 and SVC1.
- SVC0: VC0, YUV422 8-bit: ICnSVC.SVC0[3:0] = 0h, ICnIPMC_C0.INF[5:0] = 1Eh
- SVC1: VC0, YUV422 8-bit: ICnSVC.SVC1[3:0] = 0h, ICnIPMC_C1.INF[5:0] = 1Eh

(Case2) This is an example of setting the same VC data type in Non Image Process to SVC0 and SVC1.
- SVC0: VC0, YUV422 8-bit: ICnSVC.SVC0[3:0] = 0h, ICnNIPDT_C0L.INF[5:0] =1b
- SVC1: VC0, YUV422 8-bit: ICnSVC.SVC1[3:0] = 0h, ICnNIPDT_C1L.INF[5:0] = 1b

(Case3) This is an example of setting the same VC data type in Image Process and Non Image Process on the same SVC0.
- SVC0: VC0, YUV422 8-bit: ICnSVC.SVC0[3:0] = 0h, ICnIPMC_C0.INF[5:0] = 1Eh
- SVC1: VC0, YUV422 8-bit: ICnNIPDT_C0L[30]=1b

(3) Interlacing is not supported.

(4) The timing for periodic deskewing calibration for the DPHY is 2^13 to 10 µs. Be sure to use a sensor that conforms to this specification.

(5) When FIFO-VD and FIFO-SD are in use, make sure that they do not overflow. In the event that an overflow occurs during operation with footers appended, the footer value following the overflow is not guaranteed. In such cases, reset the CRU after it has been stopped.

(6) An interrupt will not occur unless the addresses that are set for the specified address 0 to 4 write completed interrupt are set as addresses to which video or statistics data are to be written, so take care on this point. For example, an interrupt will not occur if video or statistics data are not written to the address which has been set for the specified address 0 to 4 write completed interrupt and the address jumps to the start of the next line due to the image-stride setting.

(7) For the order of the startup of the sensor and CRU, be sure to initialize the CRU before sending data from the sensor, as stated in **9.2.5.1 Starting Reception for the MIPI CSI-2 Input**. If this order is not observed, the CRU may receive incomplete packets of MIPI CSI-2 signals and various errors may occur.

(8) Parameter setting of the SVCs are as follows.

Table 9.2-24 Parameter Setting of the SVCs

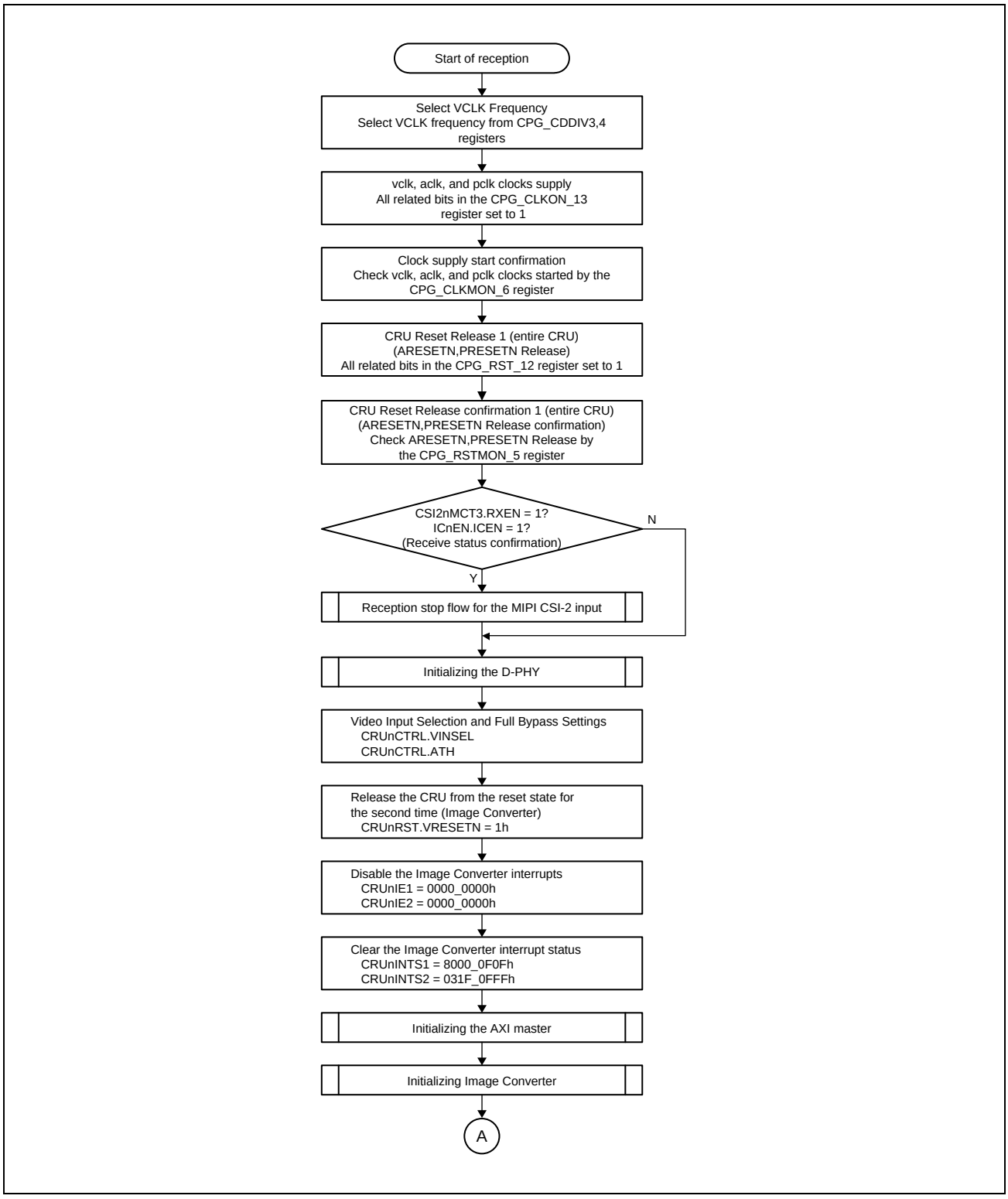
Function	SVC0	SVC1	SVC2	SVC3	Note
Image Clipping	(A)	(A)	(A)	(A)	—
Frame Subsampling	(B)	(B)	(B)	(B)	—
Data Arrange 1/2	(B)	(B)	(B)	(B)	—
Data Clip	(B)	(B)	(B)	(B)	—
Color Space Conversion	(B)	(B)	(B)	(B)	—
Demosaic	(C)	(D)	(D)	(D)	When any of the Demosaic, Linear Matrix, and Statistic functions are used, SVC1 to SVC3 cannot be used for other functions.
Linear Matrix	(C)	(D)	(D)	(D)	
Statistics	(C)	(D)	(D)	(D)	
LUT	(B)	(B)	(B)	(B)	—
YCbCr422 to a420	(A)	(A)	(A)	(A)	—
Y/C Separation	(A)	(A)	(A)	(A)	—
Only Y	(A)	(A)	(A)	(A)	—

Note: (A): Can be set independently of other SVCs
(B): Parameter setting common to SVC0 to SVC3
(C): Functions only with SVC0
(D): Cannot be set

(9) Image Stride when output format is YUV420
Note that when the output format is set to YUV420 (ICnDMR.YCMODE[2:0] = 110b), the U/V side is written to RAM as Image Stride (AMnIS) × 2.
See **Image Stride** in **9.2.4.3.3(5)(a) Video data frame** for details.

9.2.5 Operation

9.2.5.1 Starting Reception for the MIPI CSI-2 Input



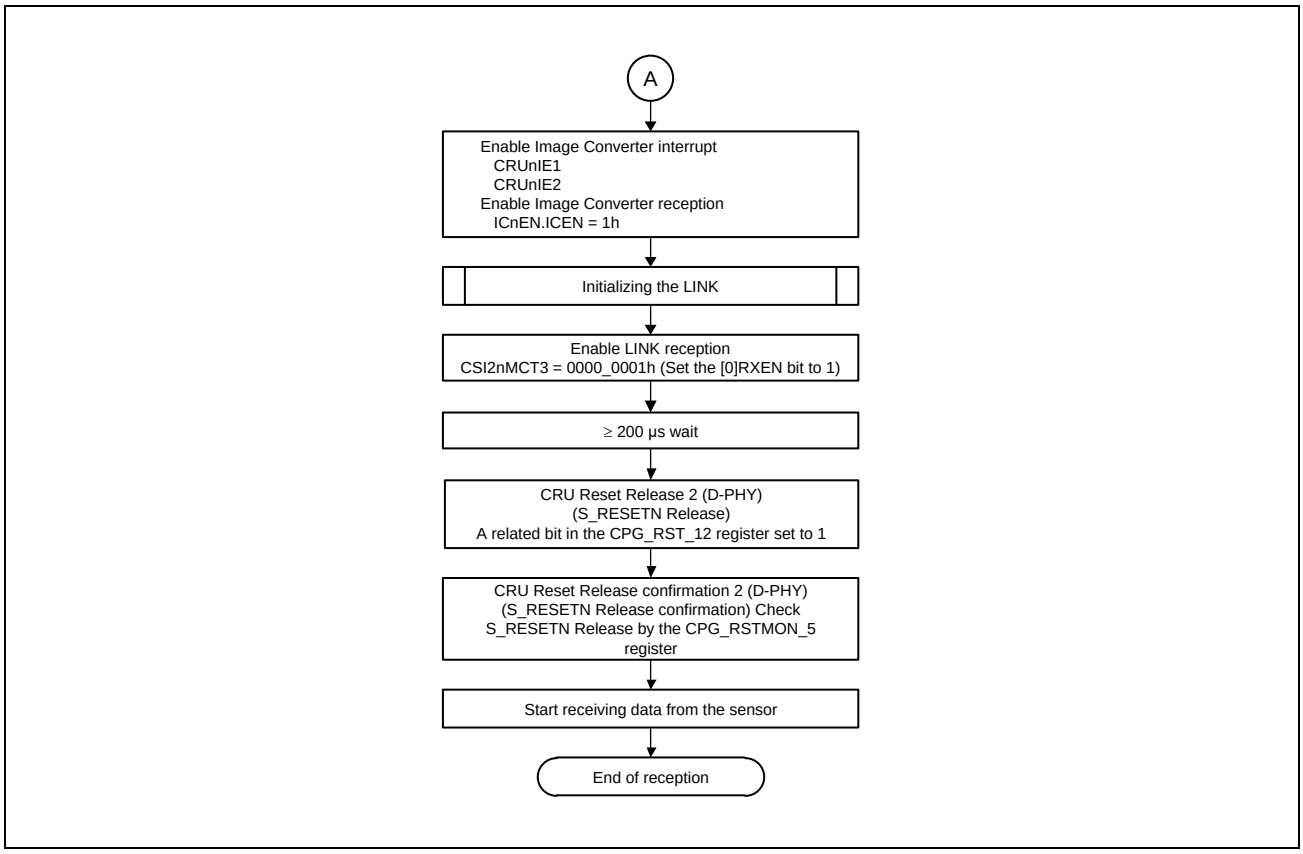


Figure 9.2-53 Reception Start Flow for the MIPI CSI-2 Input

9.2.5.1.1 Initializing the D-PHY

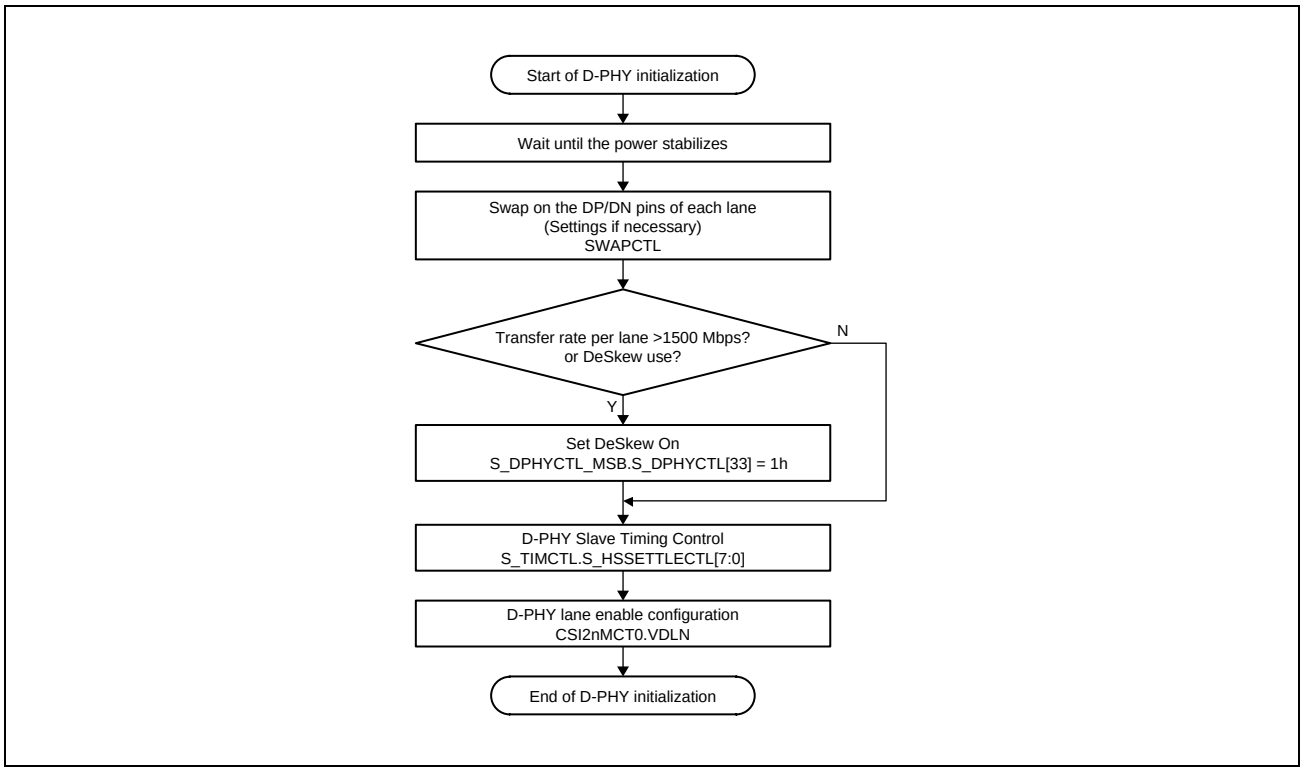


Figure 9.2-54 Initializing the D-PHY

9.2.5.1.2 Initializing the AXI Master

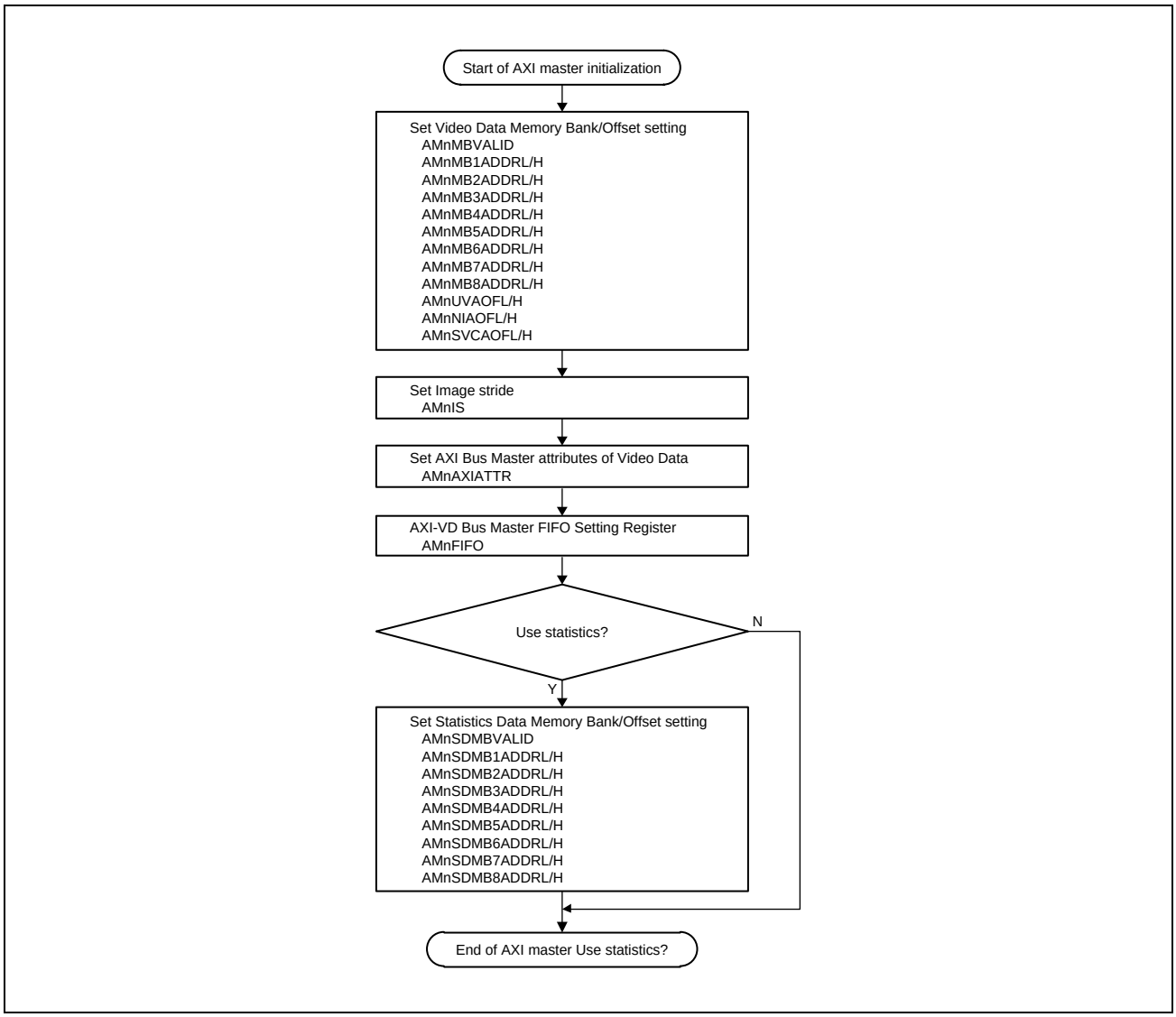
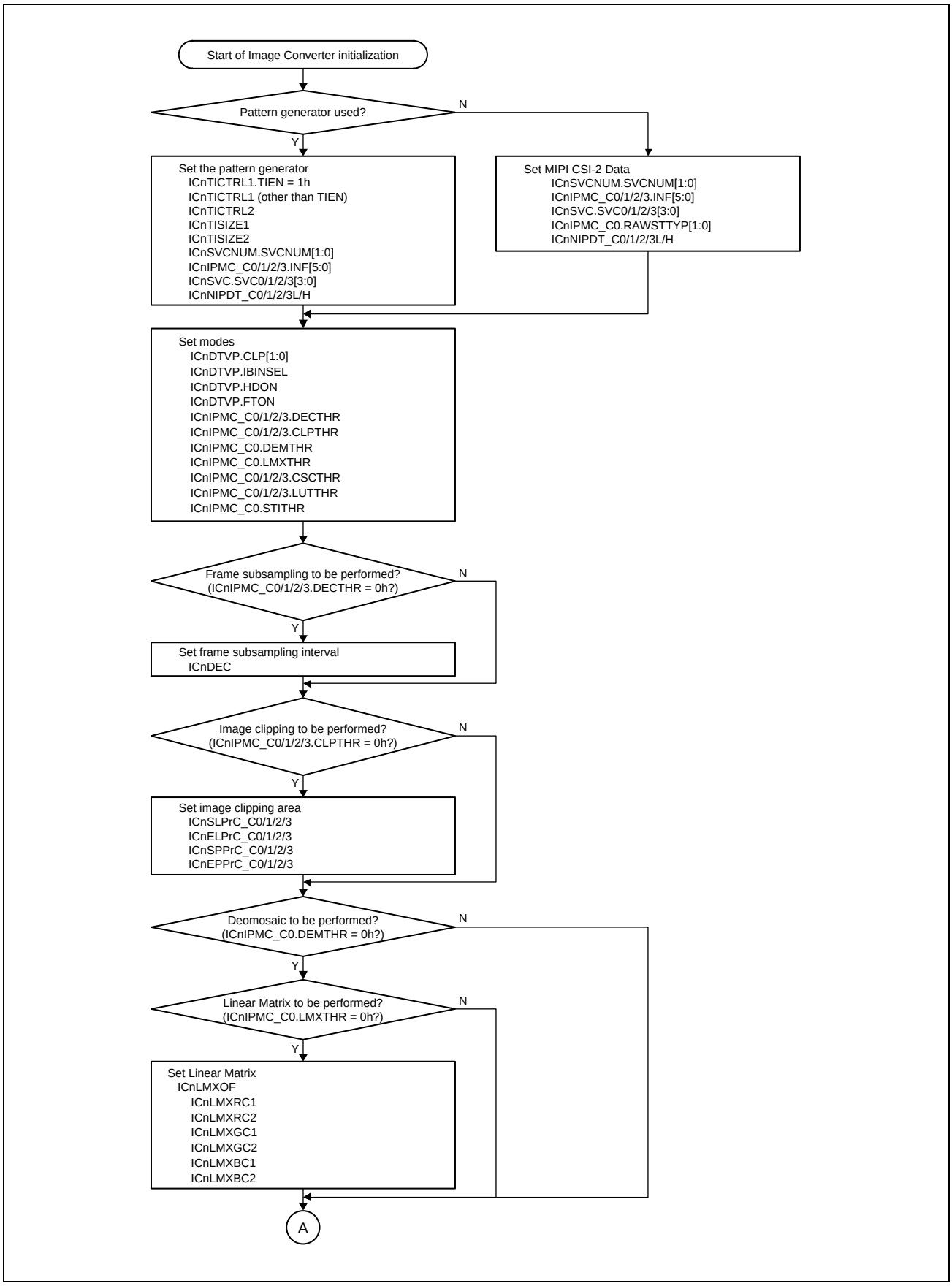


Figure 9.2-55 Initializing the AXI Master

9.2.5.1.3 Initializing the Image Converter



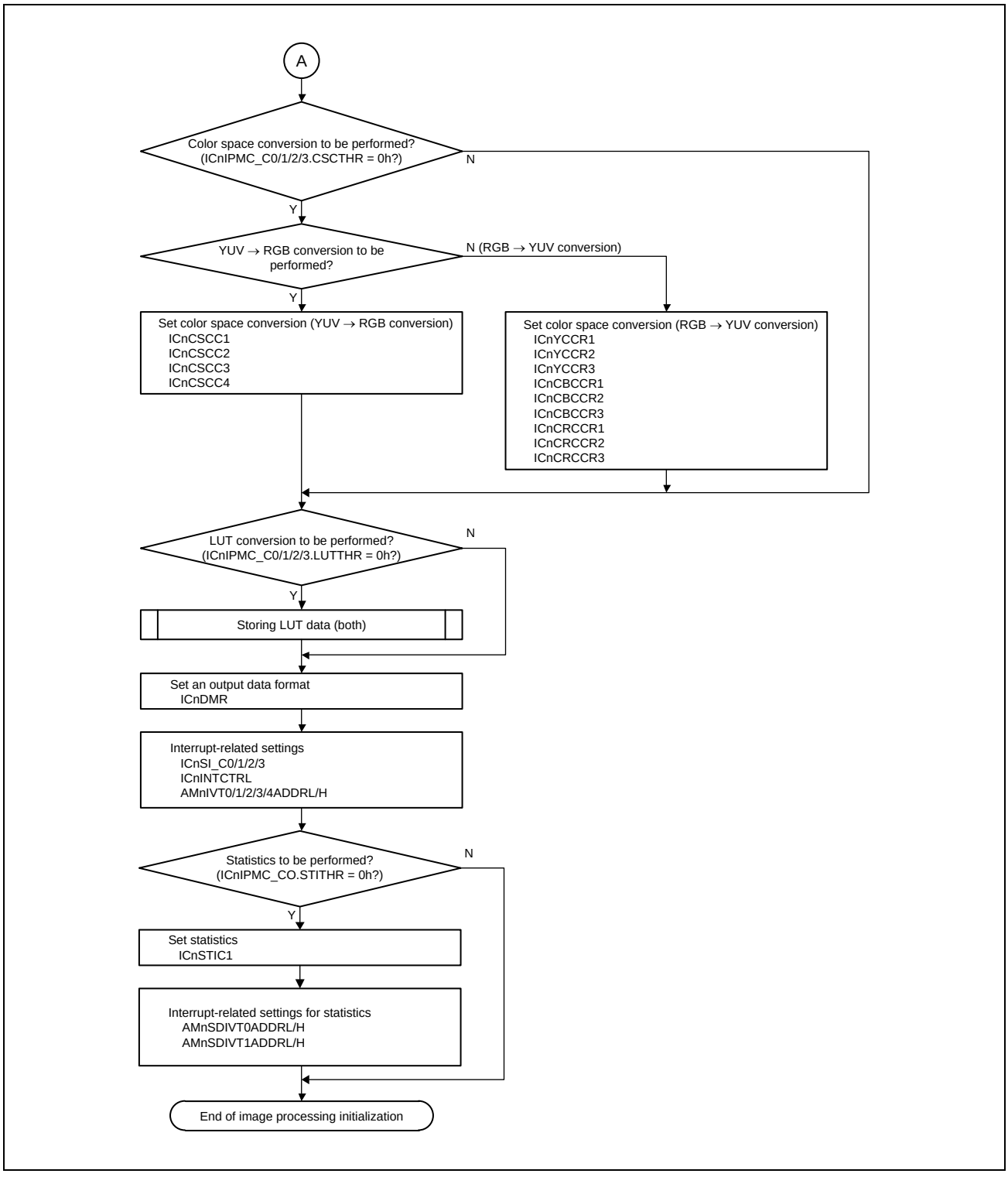


Figure 9.2-56 Initializing Image Processing

9.2.5.1.4 Storing LUT Data

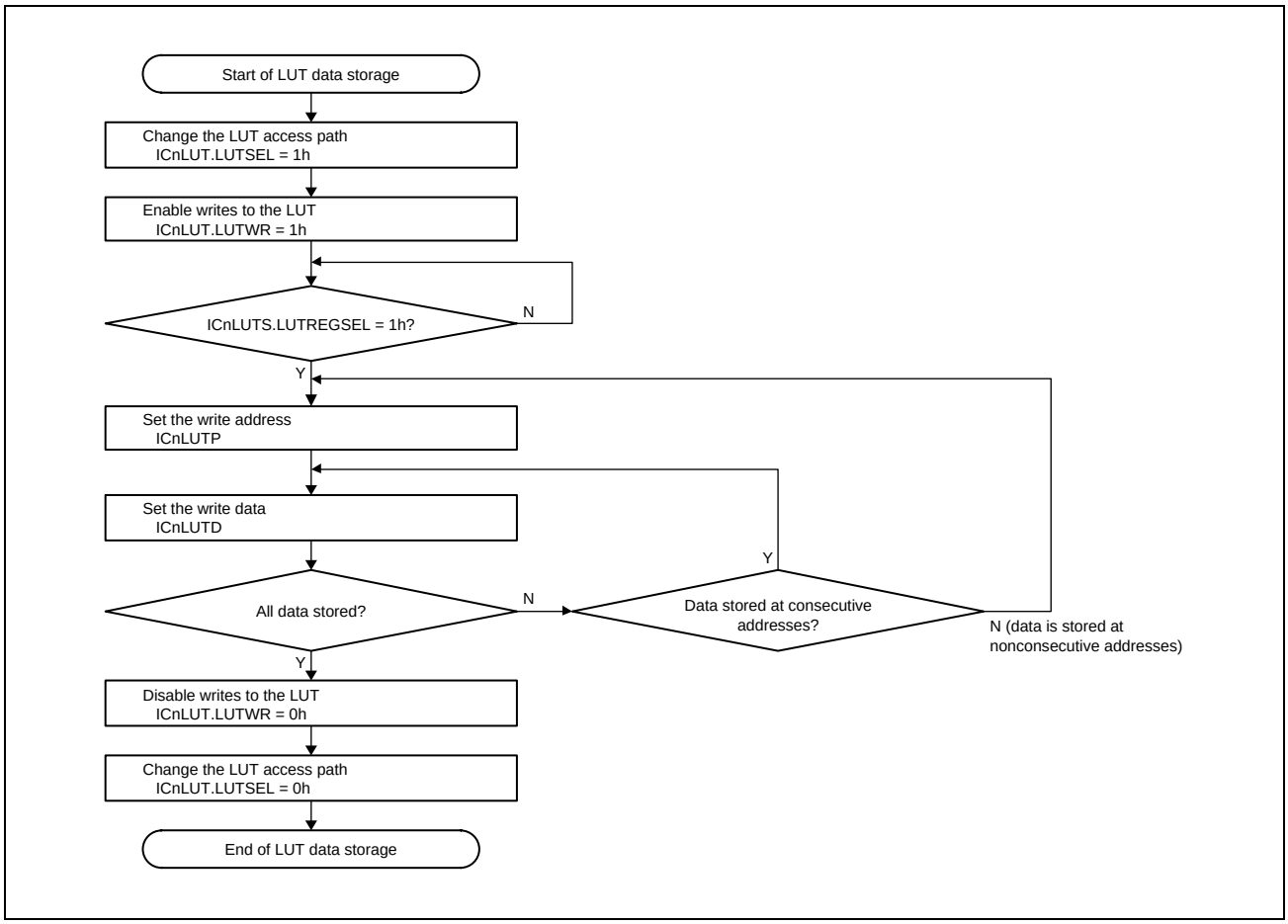


Figure 9.2-57 Storing LUT Data

9.2.5.1.5 Initializing the LINK

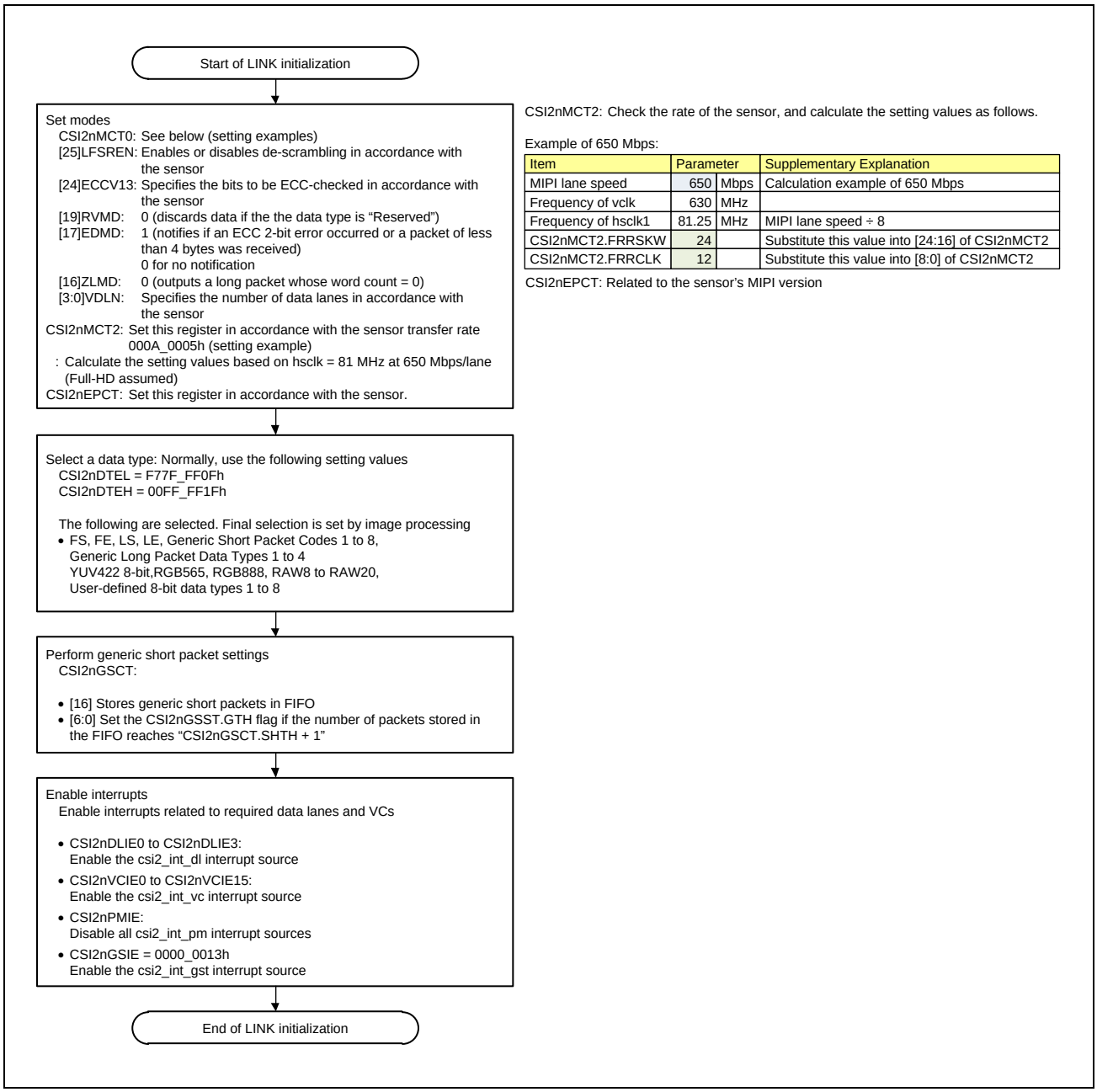


Figure 9.2-58 Initializing the LINK

9.2.5.2 Stopping Reception for the MIPI CSI-2 Input

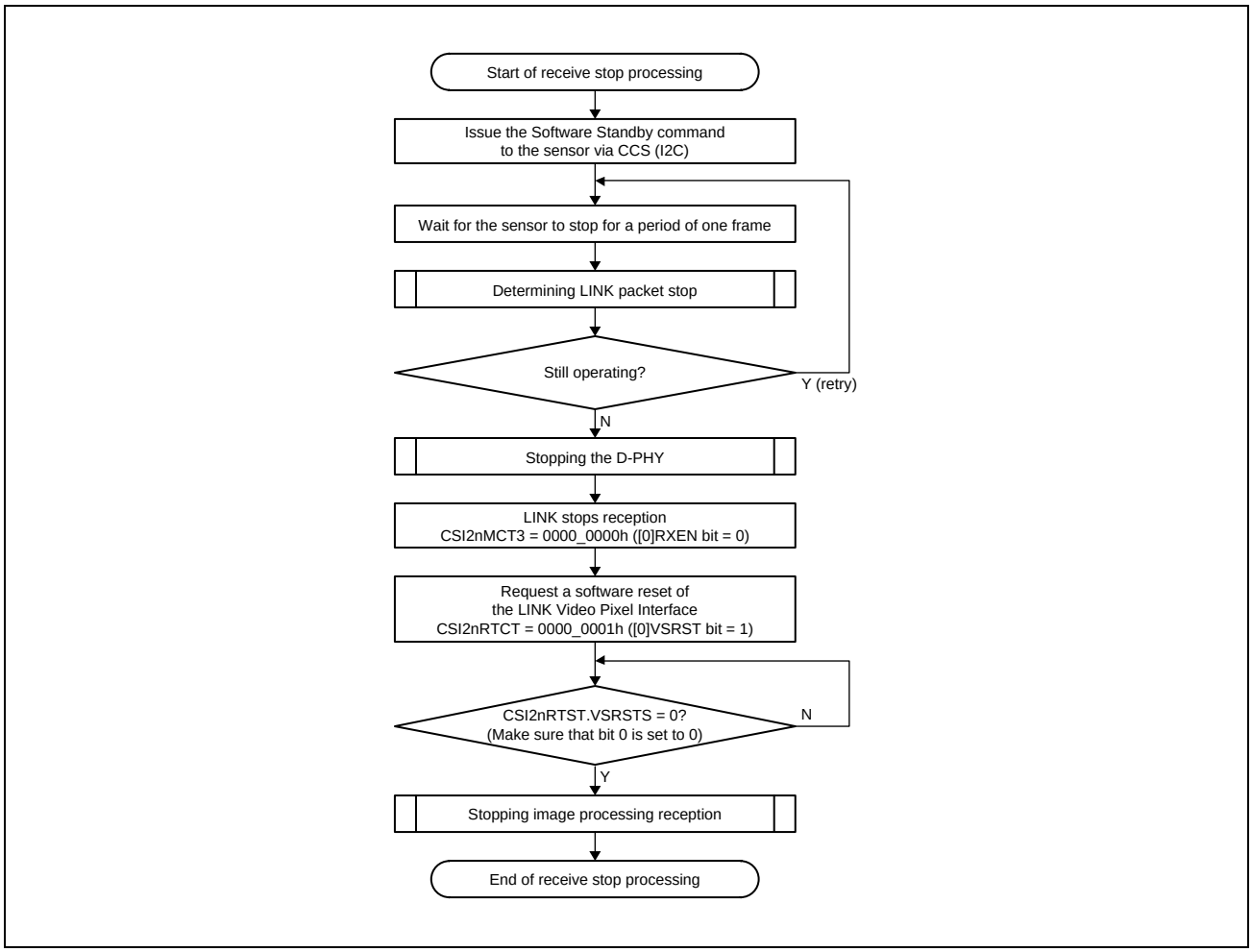


Figure 9.2-59 Flow of Stopping Reception for the MIPI CSI-2 Input

9.2.5.2.1 Determining LINK Packet Stop

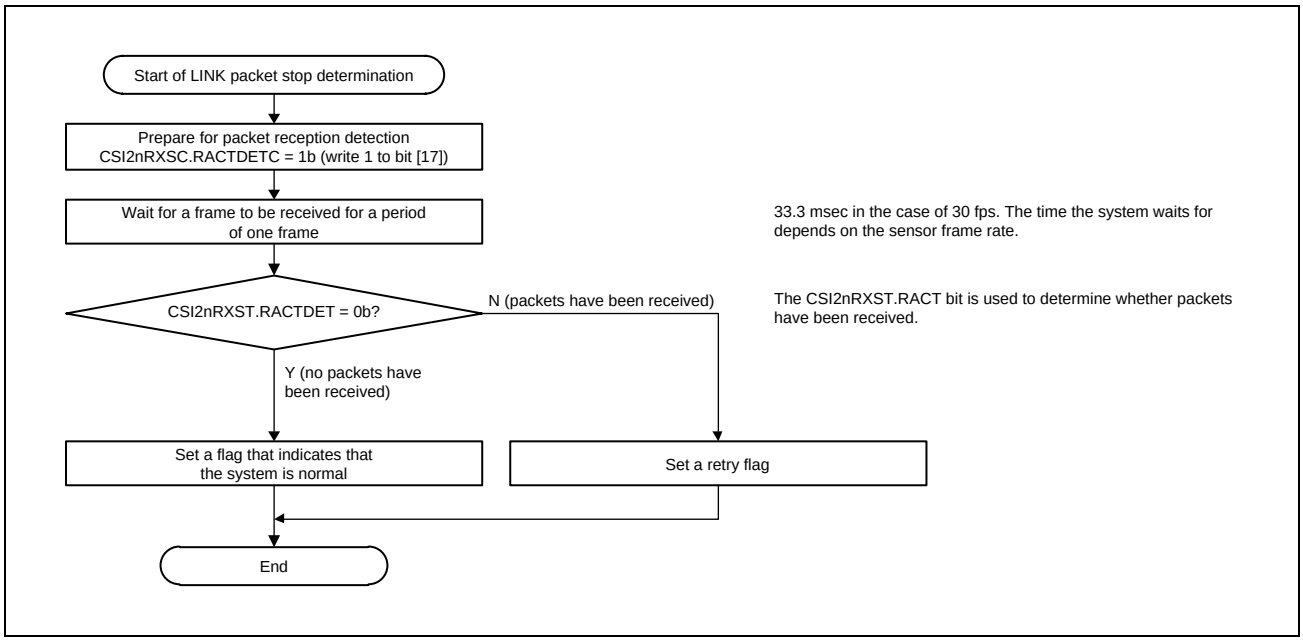
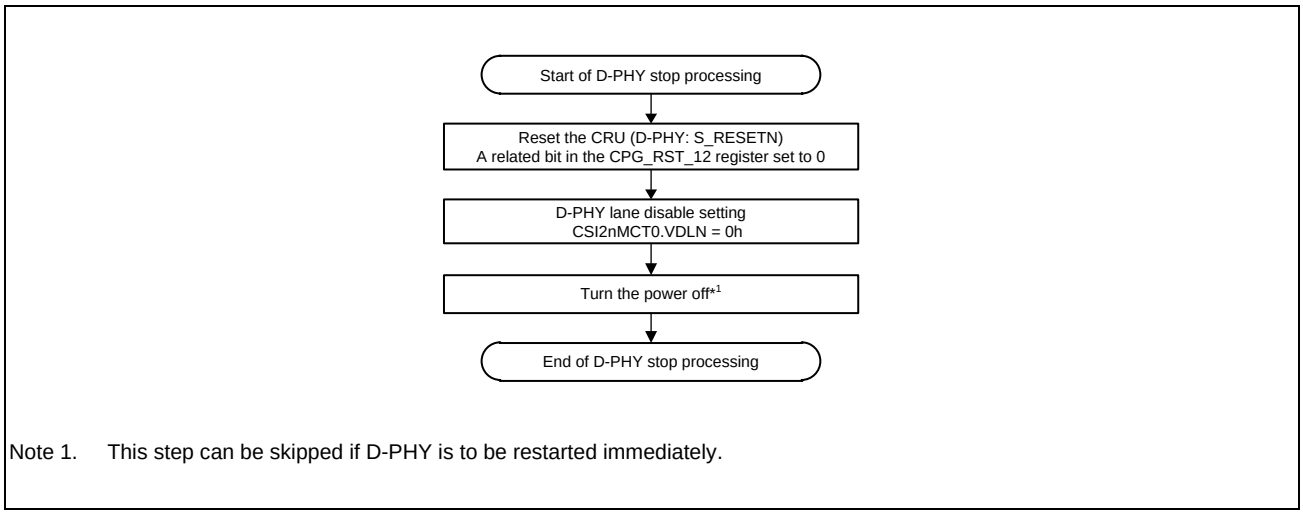


Figure 9.2-60 Determining LINK Packet Stop

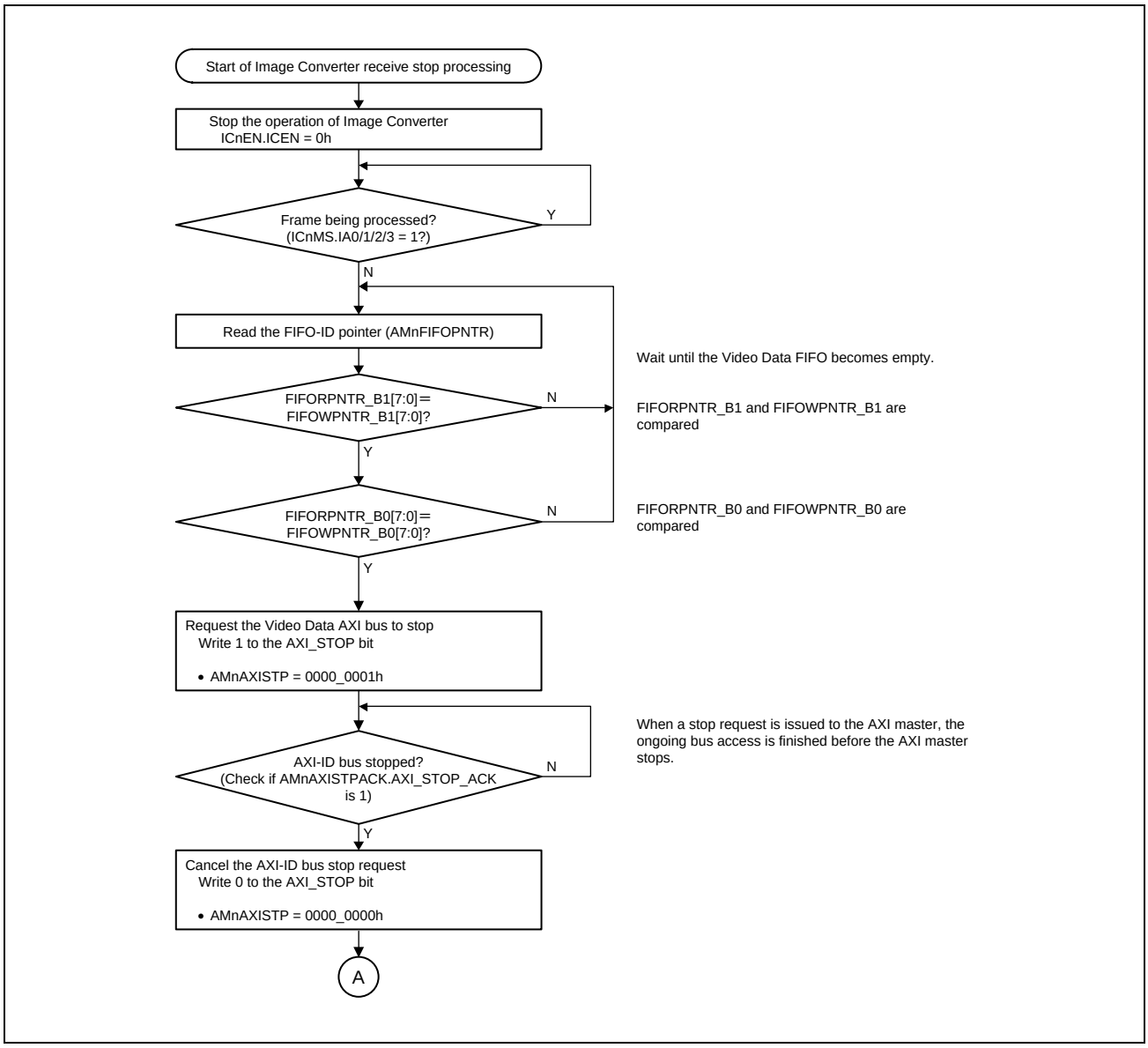
9.2.5.2.2 Stopping the D-PHY



Note 1. This step can be skipped if D-PHY is to be restarted immediately.

Figure 9.2-61 Stopping the D-PHY

9.2.5.2.3 Stopping Image Converter Reception



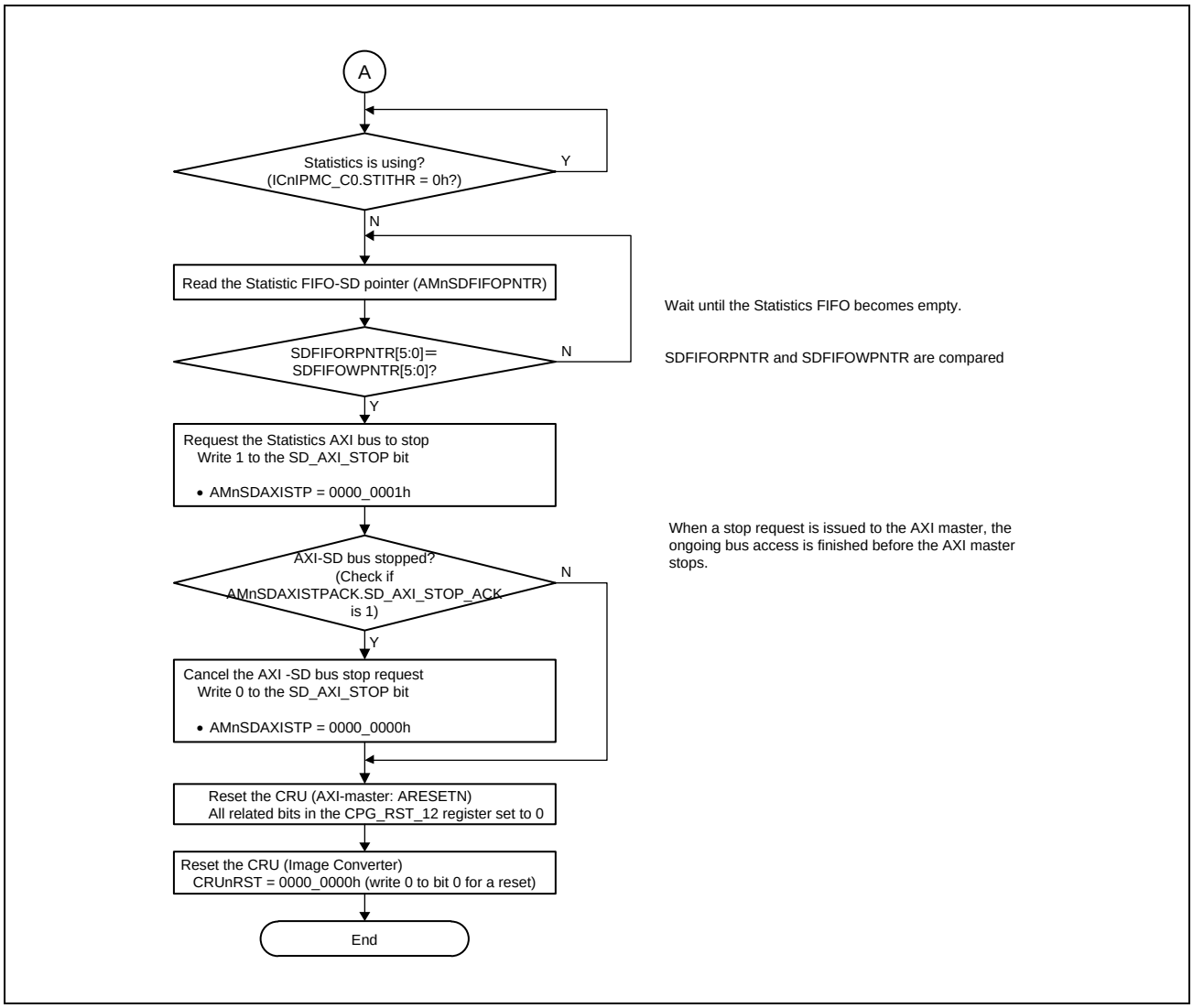


Figure 9.2-62 Stopping Image Processing Reception

9.2.5.3 Frame Synchronization Settings

While the CRU is receiving data using SVC0, the following registers can only be changed with frame synchronization:

- ICnSVC.SVC0[3:0]
- ICnIPMC_C0.STITHR
- ICnNIPDT_C0L
- ICnNIPDT_C0H
- ICnSLPrC_C0
- ICnELPrC_C0
- ICnSPPrC_C0
- ICnEPPrC_C0
- ICnSI_C0
- ICnLMXOF
- ICnLMXRC1
- ICnLMXRC2
- ICnLMXGC1
- ICnLMXGC2
- ICnLMXBC1
- ICnLMXBC2
- ICnSTIC1

9.2.5.3.1 Changing Frame Synchronization Settings (by Setting Registers)

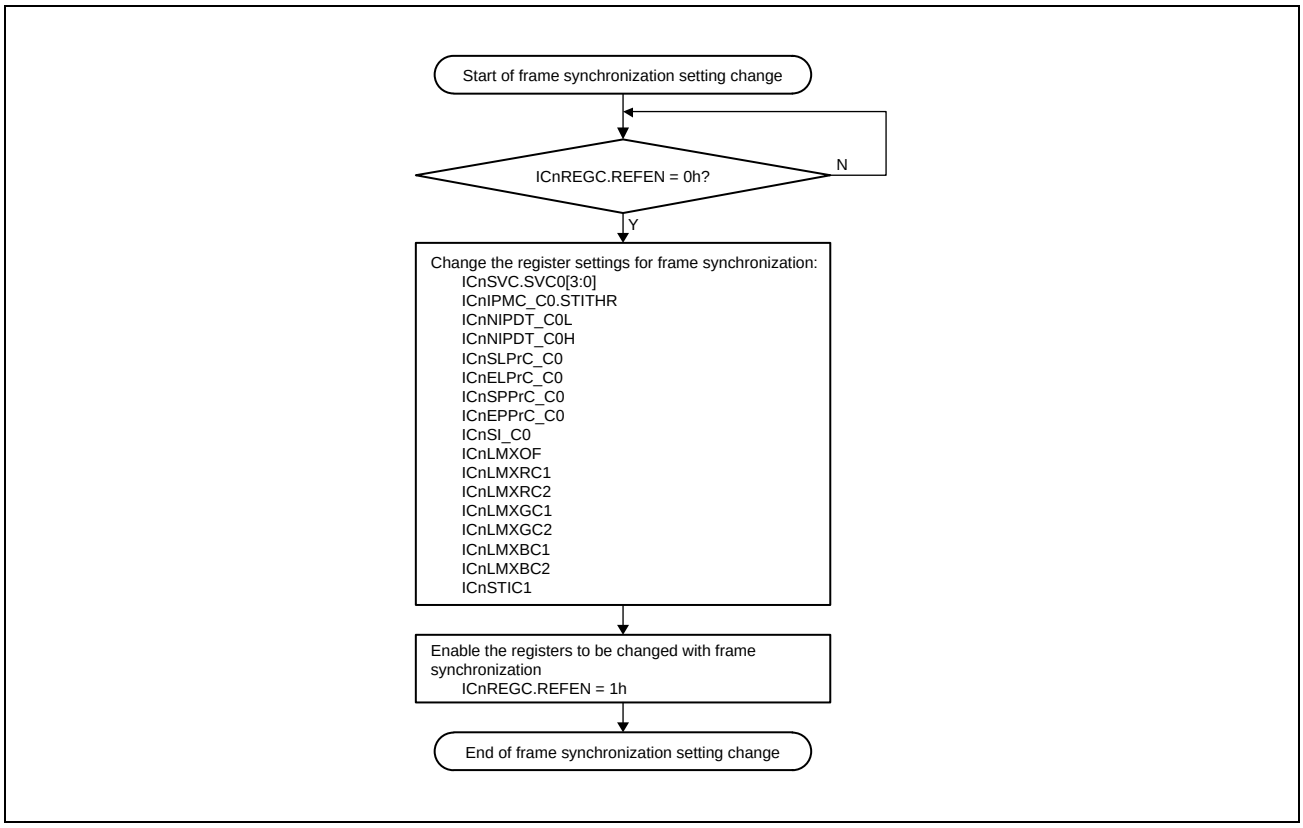
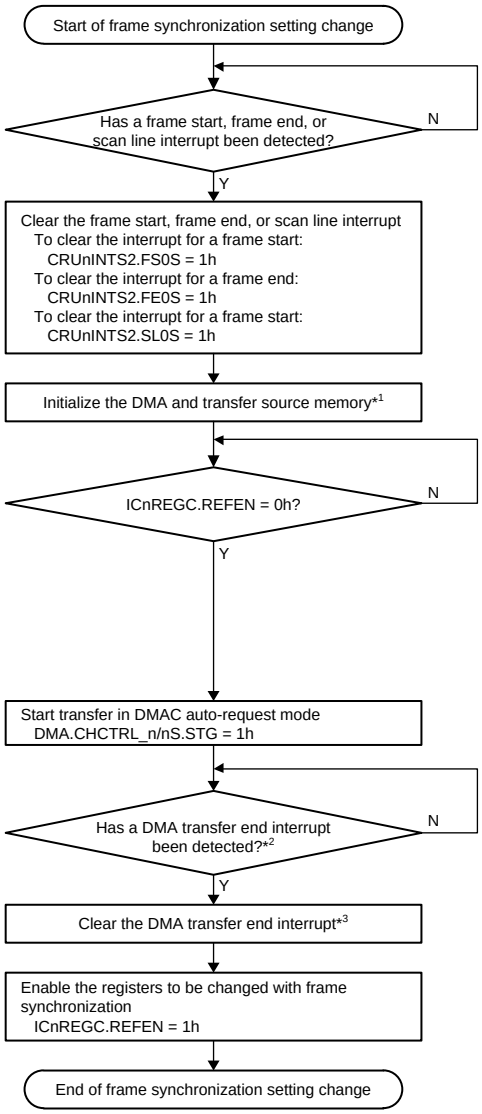


Figure 9.2-63 Flow for Changing the Frame Synchronization Settings (by Setting Registers)

9.2.5.3.2 Changing Frame Synchronization Settings (by Using the DMAC)



NOTES:

1. This flow for changing the frame synchronization settings is only valid for the MIPI CSI-2 input. It cannot be used for the parallel input.
2. In the initial settings when the CRU starts, interrupts for a frame start, frame end and scan line must be set.

   To set an interrupt for a frame start:
   CRUnIE2.FS0E = 1h

   To set an interrupt for a frame end:
   CRUnIE2.FE0E = 1h

   To set an interrupt for a frame start:
   CRUnIE2.SL0E = 1h

   In addition, CRU and DMA interrupts must be enabled by seeing the section about the interrupt controller.

Note 1. Perform initialization by seeing the section about DMA.

- Note that the following register settings are required:
  DMA.CHCFG_n/nS.DAD = 0h: Increment
  DMA.CHCFG_n/nS.SAD = 0h: Increment
  DMA.CHCFG_n/nS.DDS = 3h: 32 bits
- Set the DMA transfer size to 32 bits for both transfer destination and transfer source
- In transfer source memory initialization, store the values that set the following register: (Start address: Up to 68 bytes in ICnSVC)

  ICnSVC.SVC0[3:0]
  ICnIPMC_C0.STITHR
  ICnNIPDT_C0L
  ICnNIPDT_C0H
  ICnSLPrC_C0
  ICnELPrC_C0
  ICnSPPrC_C0
  ICnEPPrC_C0
  ICnSI_C0
  ICnLMXOF
  ICnLMXRC1
  ICnLMXRC2
  ICnLMXGC1
  ICnLMXGC2
  ICnLMXBC1
  ICnLMXBC2
  ICnSTIC1

Note 2. A setting is required in advance (see the sections about DMA and interrupt controller).

Note 3. Clear the interrupt by seeing the section about DMA and interrupt controller.

Figure 9.2-64 Flow for Changing the Frame Synchronization Settings (by Using the DMAC)

9.2.5.4 Starting Reception for the Pattern Generator Input

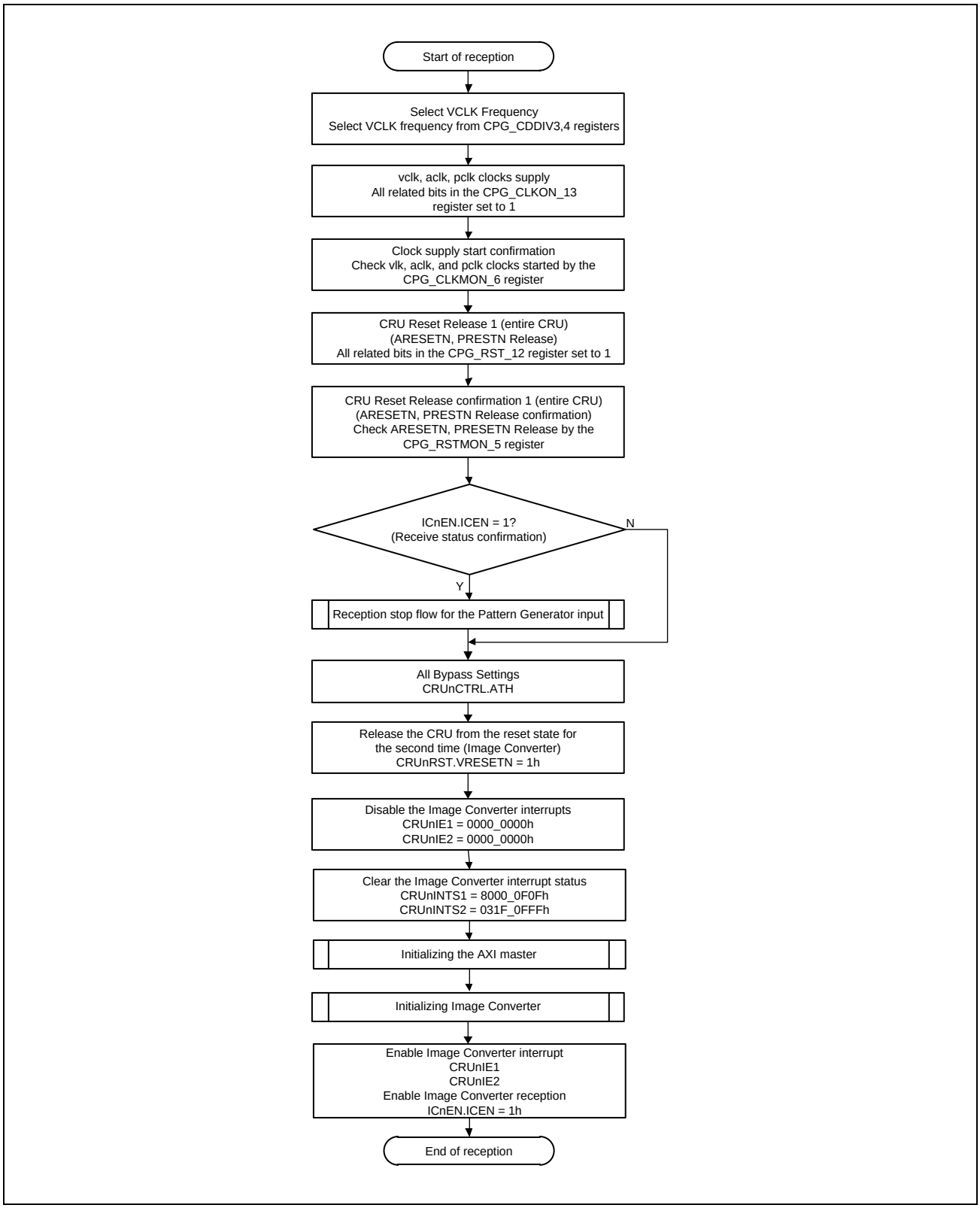


Figure 9.2-65 Reception Start Flow for the Pattern Generator Input

9.2.5.5 Stopping Reception for the Pattern Generator Input

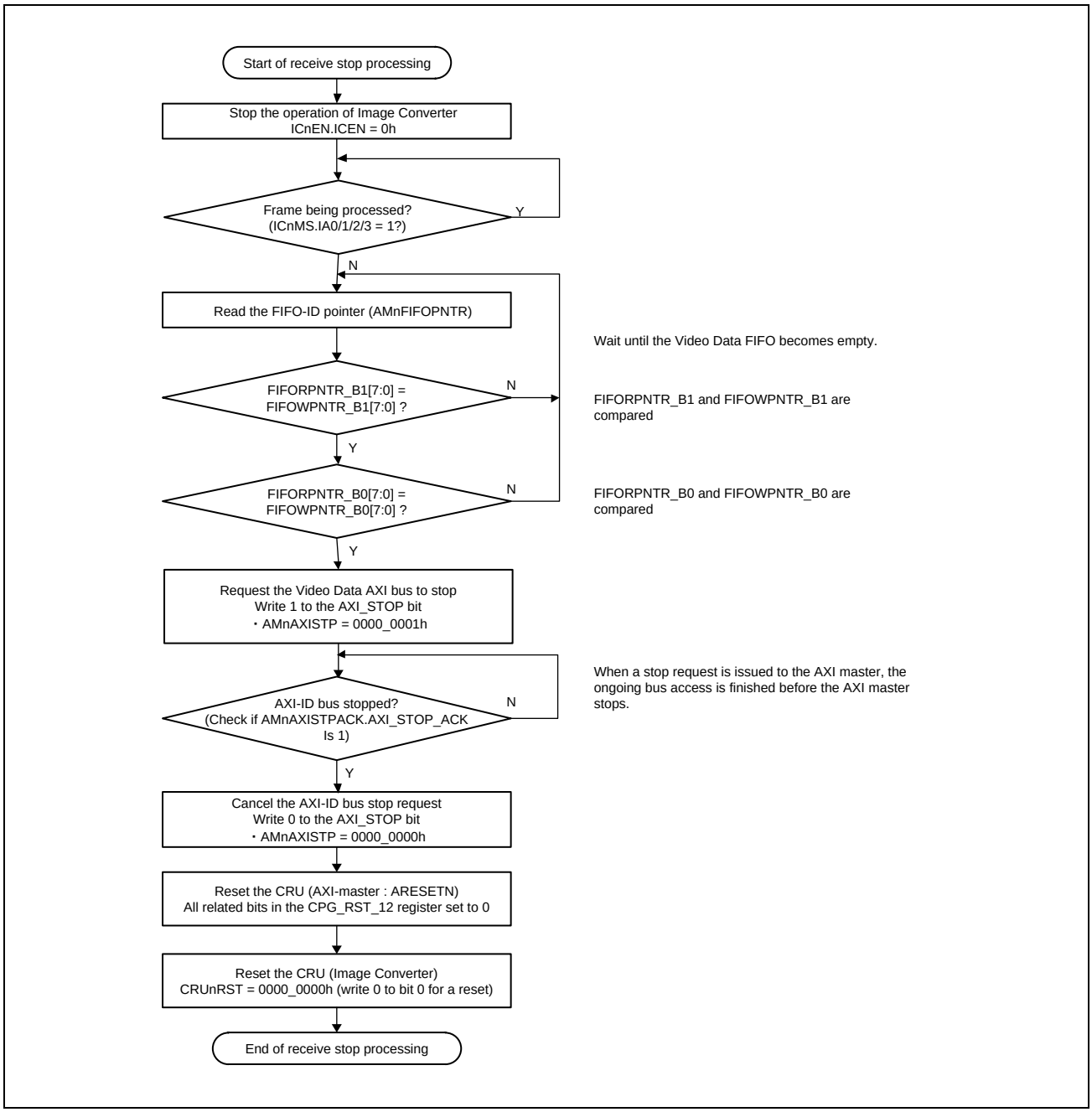


Figure 9.2-66 Flow of Stopping Reception for the Pattern Generator Input

SECTION 9 IMAGE

9.4 LCD Controller (LCDC)

9.4.1 Overview

This chapter describes the features of the LCDC unit of this LSI.

This unit is an LCD controller and it is composed of Frame Compression Processor (FCPVD), Video Signal Processor (VSPD), and Display Unit (DU).

This LSI has two these units, each unit is called as LCDC0 and LCDC1, respectively. LCDC0 is connected to LVDS (single or dual ch) and DSI. LCDC1 is connected to LVDS (single ch), DSI, and GPIO (Parallel I/F). The video data, which is output by this unit, is input to DSI, LVDS, and GPIO, and DSI converts to MIPI DSI (Display Serial Interface), LVDS converts to single channel or dual channel LVDS (Low Voltage Differential Signaling), GPIO converts only the IO voltage, and output.

9.4.1.1 Features

The following is key features of this unit.

- FCPVD
  - Supports out-of-order for the whole outstanding transactions
  - Read linear addressing image data
  - Read display list data
  - Write image data

- VSPD
  - Supports various data formats and conversion
    - Supports YCbCr444/422/420, RGB, α RGB, α plane
    - Color space conversion and changes to the number of colors by dithering
    - Color keying
    - Supports combination between pixel alpha and global alpha
    - Supports generating pre multiplied alpha
  - Video processing
    - Blending of two picture layers and raster operations (ROPs)
    - Clipping
    - 1D look up table
    - Vertical flipping in case of output to memory
  - Direct connection to display module
    - Supports up to 1920 pixels in horizontal direction

- Writing back image data which is transferred to Display Unit (DU) to memory

- DU
  - Supports Display Parallel Interface (DPI), and inputs the video data to DSI and/or LVDS
  - Display timing master
    - Generates video timings (Front porch, Back porch, Sync active, Active video area)
    - Selects the polarity of output DCLK, HSYNC, VSYNC, and DE
  - Supports Progressive (Non-interlace)
  - Not supports Interlace
  - Input data format (from VSPD): RGB888, RGB666 (not supports dithering of RGB565)
  - Output data format: same as Input data format
  - Supports up to 187.5 MHz of the pixel clock frequency
  - Examples of the supported resolutions is below if using LVDS dual ch or DSI:
    - RGB888, 1920 pixels × 1200 lines, 60 fps
    - RGB888, 1920 pixels × 1080 lines, 60 fps (FHD)
    - RGB888, 1280 pixels × 1024 lines, 120 fps
  - Examples of the supported resolutions is below if using LVDS single ch or Parallel I/F:
    - RGB888, 1366 pixels × 768 lines, 60 fps (WXGA)
    - RGB888, 1280 pixels × 800 lines, 60 fps (WXGA)
    - RGB888, 800 pixels × 480 lines, 60 fps (WVGA)
    - RGB888, 640 pixels × 480 lines, 60 fps (VGA)

9.4.1.2 Block Diagram

Figure 9.4-1 and **Figure 9.4-2** show block diagrams.

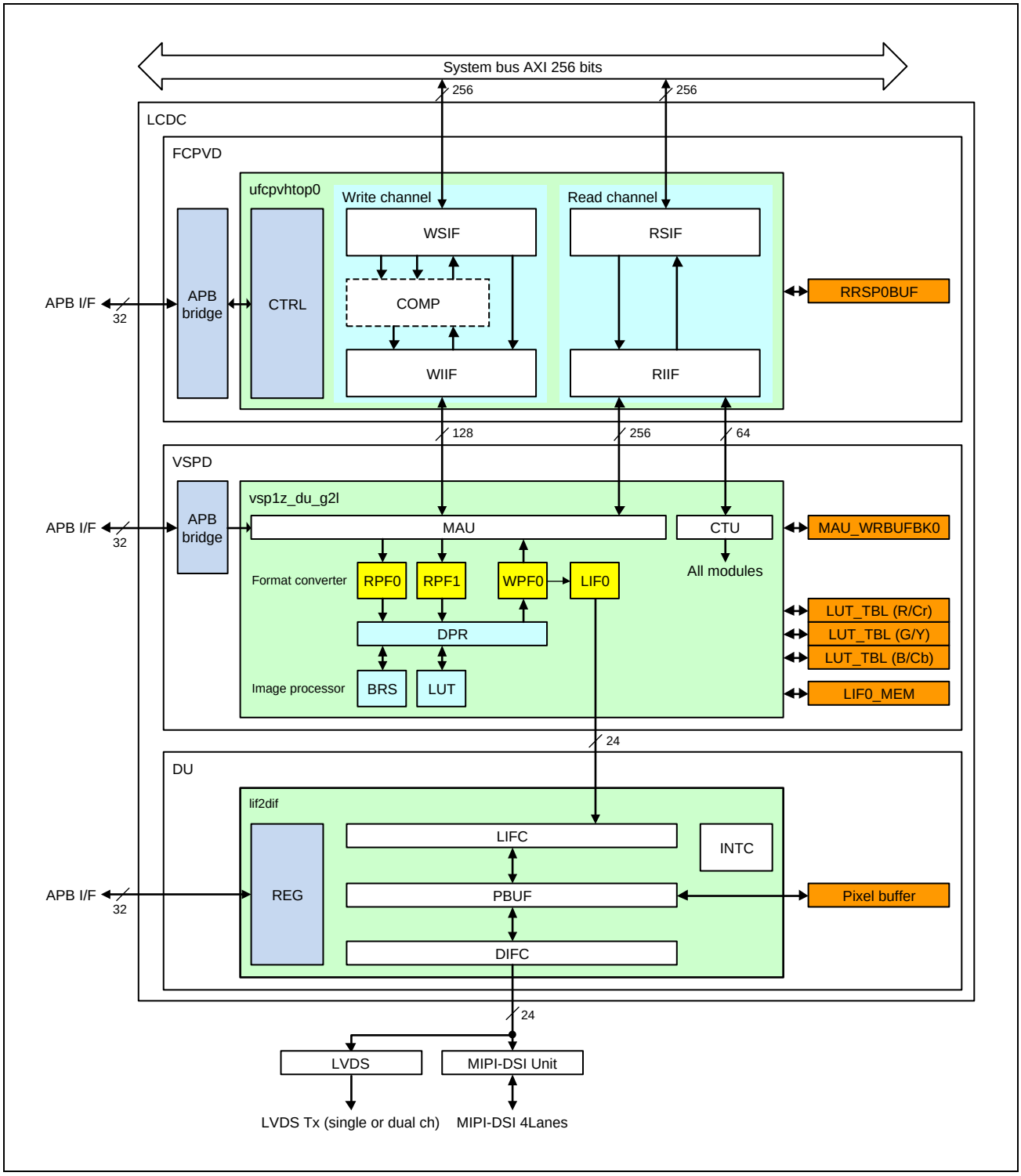


Figure 9.4-1 Block Diagram (LCDC0)

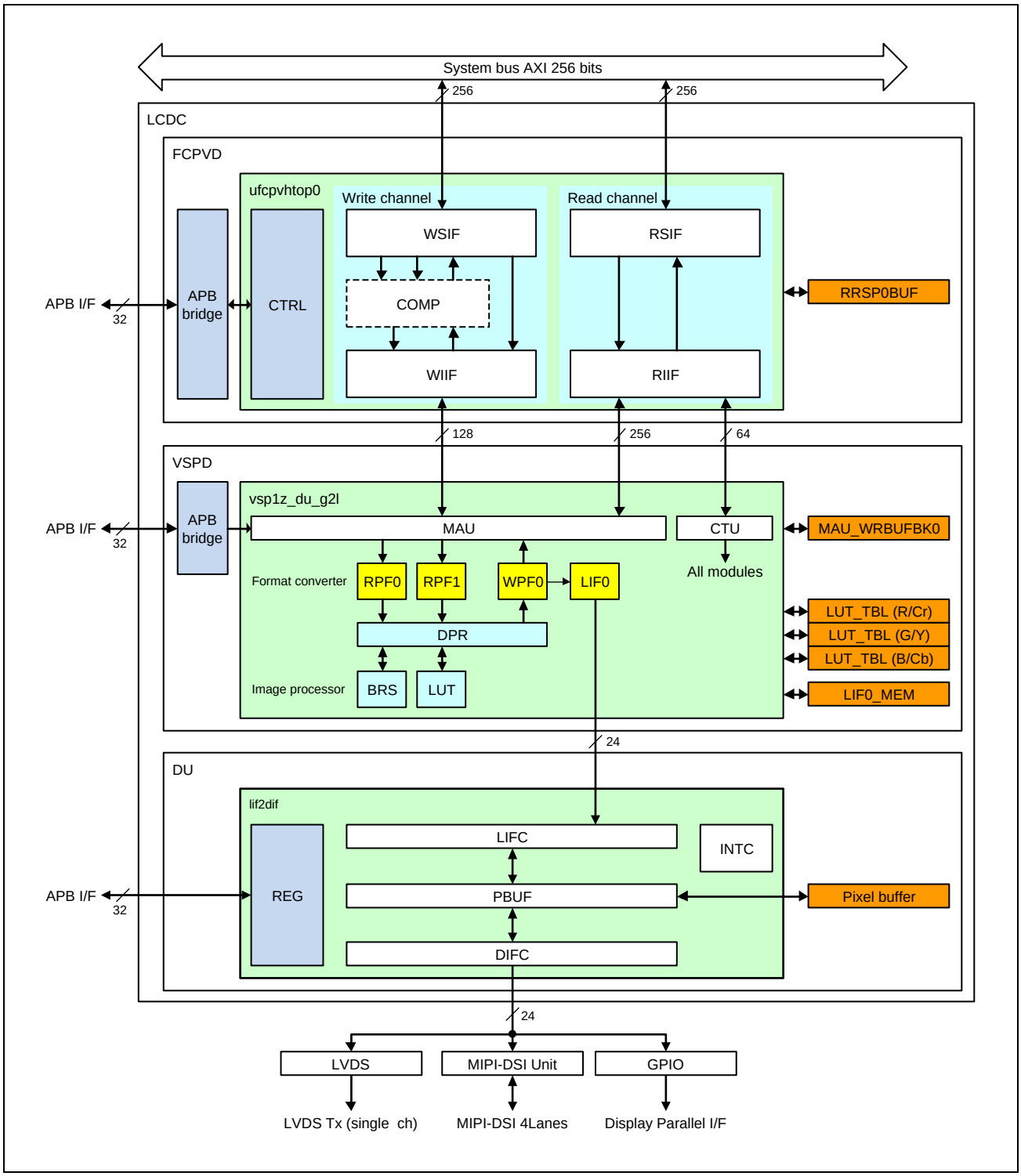


Figure 9.4-2 Block Diagram (LCDC1)

9.4.1.2.1 FCPVD

FCPVD has the following sub modules.

Table 9.4-1 Sub Modules in FCPVD

Abbreviation	Description
CTRL	FCPVD controller
WIIF	VSPD write channel interface
WSIF	System AXI write channel interface
RIIF	VSPD read channel interface
RSIF	System AXI read channel interface
COMP	Compression module (This is a dummy module)

9.4.1.2.2 VSPD

VSPD has the following sub modules.

(1) Memory Access Unit (MAU)

The VSPD applies processing to the image data stored in the external memory and writes the resultant data back to the external memory. The data transfer between the external memory and VSPD necessary for this operation is done by the MAU, which works as the bus master, according to the register settings. The MAU executes this data transfer between the external memory and VSPD.

(2) Command Transfer Unit (CTU)

The VSPD can directly read register parameters for image processing by display lists stored in external memory. The CTU module is a bus interface and controls display lists when the CTU reads display lists as a bus master.

(3) Read Pixel Formatter (RPF)

The RPF reads image data from the external memory through the MAU, unpacks data according to the specified format, converts the color space, converts the number of colors, executes color keying, ROP operation, Multiply-alpha and OSD processing, and outputs the resultant data to the DPR. The RPF has an input format unpacking unit, a 1-bit mask generator, a raster operation unit (ROP unit), a color keying unit, a color space converter and multiply-alpha. **Figure 9.4-3** show the processing flow in the RPF.

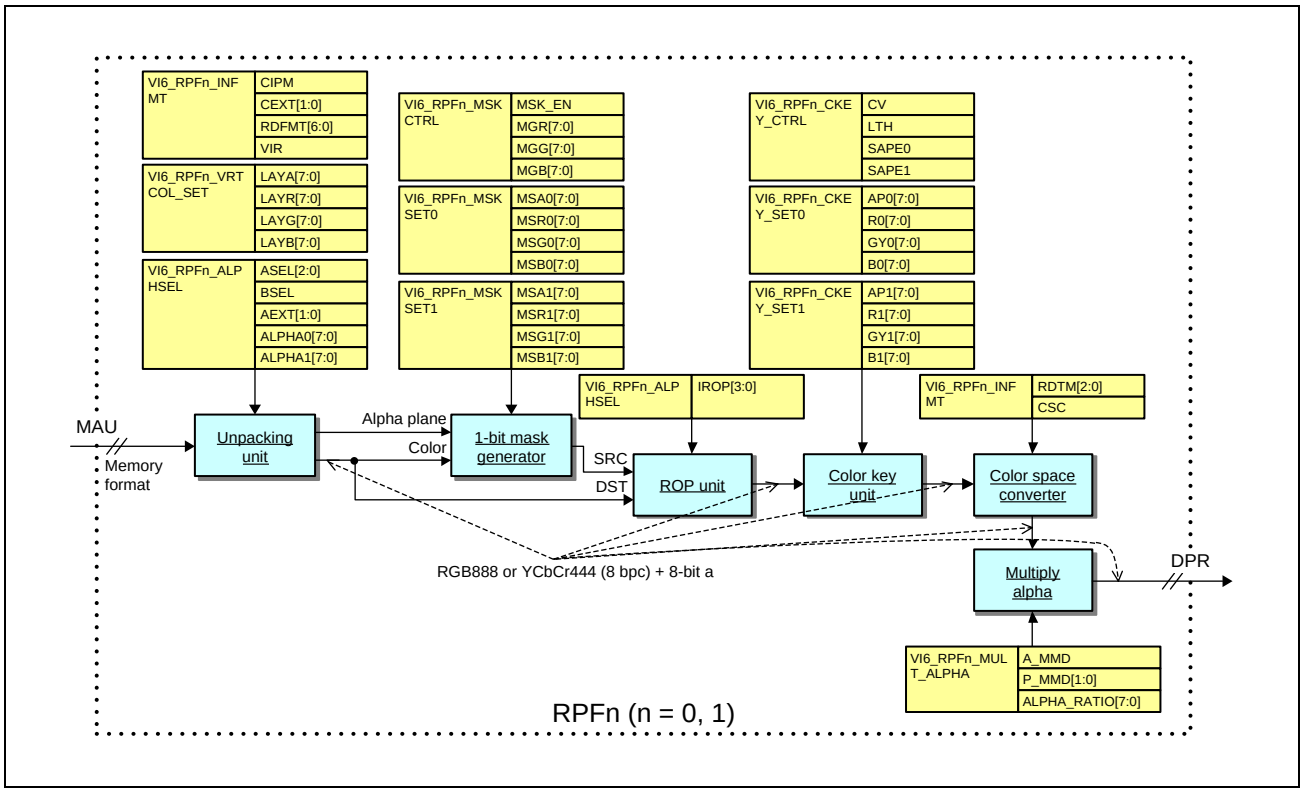


Figure 9.4-3 RPF Processing Flow

The input format unpacking unit expand the image data input from the MAU into the image format for internal processing (YCbCr444 8bpc or RGB888 8bpc), and the 1-bit mask generator generates a 1-bit image mask from the image data expanded through the unpacking unit. Alternately, a 1-bit image mask can be generated from the alpha plane that is different from the picture plane read through the MAU.

The raster operation unit (ROP unit) executes raster operation between the data from the 1-bit mask generator and the image data expanded from the input format, and the color keying unit applies color replacement and specifies the transparent color for the image data input from the ROP unit. The color space converter converts the color space (RGB-YCbCr) of the image data input from the color keying unit as necessary. The multiply-alpha unit multiplies pixel-alpha by specified alpha or/and multiplies image data by pixel-alpha or/and specified alpha.

The VSPD provides maximum two RPF modules (RPF0 to RPF1).

(4) Data Path Router (DPR)

The DPR controls the data paths among RPFs, function modules, and WPFs. The DPR selects one of the images input from RPFs, outputs it to a function module (BRS or LUT), and selects one of WPFs as the destination where the image data processed in the function module will be output. Before output to the WPF, the output from each function module can be input to another function module, which enables multiple image processing functions to be executed continuously without involving the external memory.

(5) Look Up Table (LUT)

This is a 1D-LUT that converts each of three color components by using a lookup table. The LUT is connected to the DPR and can be used for gamma correction, negative-positive conversion, posterization, and binarization through desired tone curve settings.

The LUT supports local area processing for Y component. The LUT can apply 1D look up table for Y component per area where a image is divided into multiple area specified by registers and command list.

(6) Blend ROP Sub Unit (BRS)

The BRS is a module connected to the DPR, which executes the image blending processing and ROP operation. The BRS has two blend/ROP operation units (blend/ROP unit m, m = A to B), a blend/ROP input switch (SEL) for selecting the input to these operation units, and a divider for normalization (div unit).

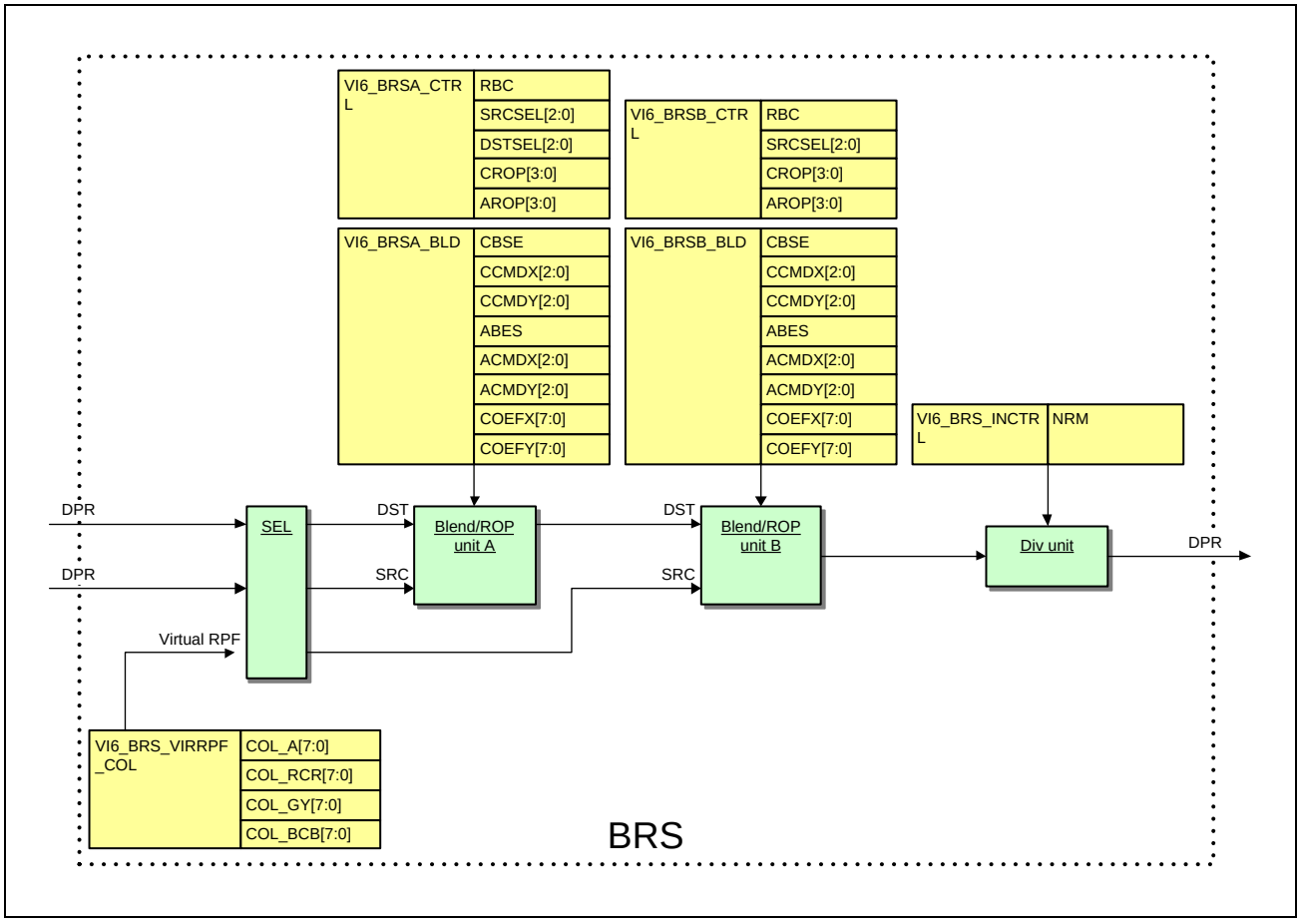


Figure 9.4-4 BRS Processing Flow

Each of the two blend-ROP operation units (blend/ROP unit m) receives the output from the SEL or blend/ROP Unit m, and executes blending or raster operation (ROP) of images.

The divider for normalization (div unit) divides the pixel value by the α value.

(7) Write Pixel Formatter (WPF)

The WPF is an output module that receives 32bits image data (YCbCr444 or RGB888 + 8-bitα) from the DPR, converts the color space, number of colors, and format of the data, and outputs the results of VSPD image processing to external memory through the MAU. The WPF is mainly configured from a color space converter and an output format converter (the packing unit).

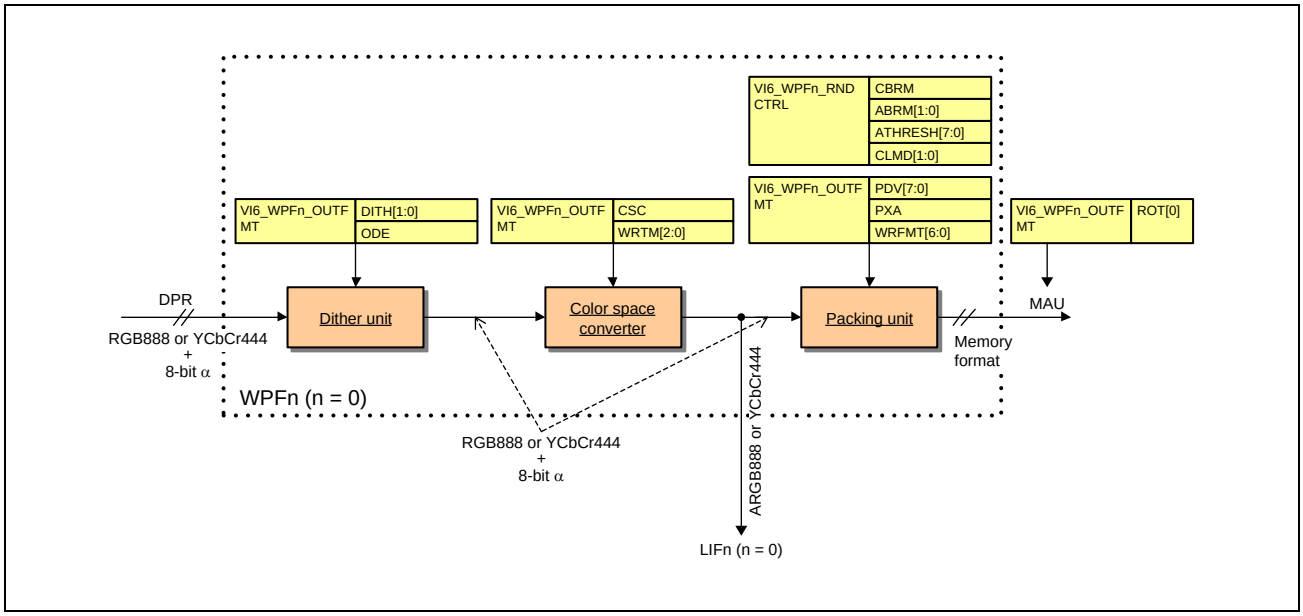


Figure 9.4-5 WPF Processing Flow

The color space converter converts the color space between RGB and YCbCr, and the packing unit converts the format into the picture plane storing format. The VSPD provides one WPF (WPF0). The WPF has the output to LIF module after the color space conversion for transferring the video data to the display module. Image data to LIF is 24bits (YCbCr444 or RGB888) format, and its' color space is after color space conversion.

(8) Display Unit Interface (LIF)

The LIF module is used for transferring image data to the display module. The input port of the LIFn (n = 0) module is connected to WPFn (n = 0), and the output port of the LIF module is connected to DU (Display Unit). Data flow in LIF is shown in **Figure 9.4-31**.

(9) Detail Function

Table 9.4-2 Detail Function of VSPD (1/4)

• Image Data Transfer Function			
Bus interface	Protocol		AXI 256 bits through FCPVD
	Data alignment	Conversion method	Byte, Word, LW, or LLW data swapping
		Channel	Data alignment can be specified separately for each input/output channel
Image memory	Input	Address setting	1-byte units
	Output	Address setting	1-byte units
		Memory area	Images can be written to the same memory area where the master layer is stored. Note the restrictions shown in **Table 9.4-38**
Tile transfer mode			Supported by RPF0 to RPF1
YCbCr memory storage format			Interleaved, planar, or semi-planar
• Display list/Extended Display List Transfer Function			
Bus interface	Protocol		AXI 256 bits through FCPVD
	Data alignment	Conversion method	Byte, Word, LW data swapping
		Channel	Data alignment can be specified separately for display list and each entry of extended display list
Data memory	Load	Address setting	8-byte units
	Store	—	Not available

Table 9.4-2 Detail Function of VSPD (2/4)

• Read Pixel Formatter (RPF)			
Number of channels			Two channels (RPF0 to RPF1)
Input color bit-depth		Color	2- to 8-bit
		Alpha	1-/2-/4-/8-bit
Operation bit-depth		Color	8-bit
		Alpha	8-bit
Image format	Input	RGB	RGB888, RGB565, RGB666, αRGB8888, αRGB4444, αRGB1555 α plane (8 bpp, 1 bpp)
		YCbCr	• YCbCr4:4:4 (8 bpc) Planar/Semi planar/Interleaved • YCbCr4:2:2 (8 bpc) Planar/Semi planar/Interleaved • YCbCr4:2:0 (8 bpc) Planar/Semi planar/Interleaved α plane (8 bpp, 1 bpp)
		Maximum size	1920 × 1200 pixels, 1200 × 1920 pixels The internal data path modules have separate restrictions on the maximum image size. For details, refer to **9.4.6.3 Input Image Size**.
		Minimum size	1 × 1 pixel The internal data path modules have separate restrictions on the minimum image size. For details, refer to **9.4.6.3 Input Image Size**.
		Size setting unit	• YCbCr420: 2-pixel units both horizontally and vertically. • YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. In other formats, the size can be set in 1-pixel units.
Color keying	Color replacement	Compared data	RGB or Y (8 bpc)
		Replaced data	αRGB or αYCbCr (8 bpc for R, G, B, Y, Cb, Cr, α)
		Comparison Mode	Matched color mode
		Input source	RPF0 to RPF1
	Transparent color	Compared data	RGB or Y (8 bpc)
		Replaced data	α 8 bpp
		Comparison Mode	Matched color mode, Luma threshold mode
		Input source	RPF0 to RPF1
Raster operation	ROP2 (within input channels)	Operator	16 types (OpenGL2.0 is supported)
		Sources of operation	ROP2 operation between the 1-bpp α plane and RGB/YCbCr data in RPF0 to RPF1. Note that 1-bpp α is converted to αRGB or αYCbCr4:4:4. (8 bpc)
Color space conversion	RGB to YCbCr	Conversion expression	• RGB (0, $2^n - 1$) to BT601 (2^{n-4}, $235\times2^{n-8}$/$240\times2^{n-8}$) • RGB (0, $2^n - 1$) to BT709 (2^{n-4}, $235\times2^{n-8}$/$240\times2^{n-8}$) • RGB (0, $2^n - 1$) to BT601 (0, $2^n - 1$) • RGB (2^{n-4}, $235\times2^{n-8}$) to BT709 (2^{n-4}, $235\times2^{n-8}$/$240\times2^{n-8}$) (n = 8)
		Target of conversion	RPF0 to RPF1
	YCbCr to RGB	Conversion expression	• BT.601 (2^{n-4}, $235\times2^{n-8}$/$240\times2^{n-8}$) to RGB (0, $2^n - 1$) • BT.709 (2^{n-4}, $235\times2^{n-8}$/$240\times2^{n-8}$) to RGB (0, $2^n - 1$) • BT.601 (0, $2^n - 1$) to RGB (0, $2^n - 1$) • BT.709 (2^{n-4}, $235\times2^{n-8}$/$240\times2^{n-8}$) to RGB (2^{n-4}, $235\times2^{n-8}$) (n = 8)
		Target of conversion	RPF0 to RPF1

Table 9.4-2 Detail Function of VSPD (3/4)

• Read Pixel Formatter (RPF)			
Changing number of colors	Bit extension	Extended bits	Input bitdepth (2bpc to 6bpc) to operation bitdepth 8bpc
		Target format	RGB666, RGB565, RGB555, RGB444, or RGB332
		Bit reduction method	Padded with 0. Copied from the most significant bits.
		Target of conversion	RPF0 to RPF1
	YCbCr444 generation	Vertical	CbCr copying
		Horizontal	copying or interpolation.
α bit count conversion	Bit extension	Extended bits	Input bitdepth 1bpc to operation bitdepth 8bpc
		Method	Padded with 0. Copied from the most significant bits.
Multiply-alpha function	Fade-alpha		RPF0 to RPF1 Available for straight pixel and pre-multiplied pixel.
	Generate pre-multiplied alpha		RPF0 to RPF1
Virtual display	RPF0 to RPF1	Color format	αRGB8888 or αYCbCr4:4:4 single-color
		Display size	Same as the size of the input channel
	Virtual RPF	Color format	αRGB8888 or αYCbCr4:4:4 single-color
		Display size	Maximum: 1920 × 1200 pixels, 1200 × 1920 pixels Minimum: 4 × 4 pixels
• Write Pixel Formatter (WPF)			
Number of channels			One channel (WPF0)
Image format	Output	RGB	RGB332, RGB444, RGB565, RGB666, RGB888, αRGB8666, αRGB8888, αRGB4444, αRGB1555
		YCbCr	YCbCr4:4:4, YCbCr4:2:2, YCbCr4:2:0
		Maximum size	1920 × 1200 pixels, 1200 × 1920 pixels The internal data path modules have separate restrictions on the maximum image size. For details, refer to **9.4.6.4 Output Image Size**.
		Minimum size	1 × 1 pixel The internal data path modules have separate restrictions on the minimum image size. For details, refer to **9.4.6.4 Output Image Size**.
		Size setting unit	• YCbCr420: 2-pixel units both horizontally and vertically. • YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. In other formats, the size can be set in 1-pixel units.
Color space conversion	RGB to YCbCr	Conversion expression	• RGB (0, 255) to BT.601 (16, 235/240) • RGB (0, 255) to BT.709 (16, 235/240) • RGB (0, 255) to BT.601 (0, 255) • RGB (16, 235) to BT.709 (16, 235/240)
		Target of conversion	WPF0
	YCbCr to RGB	Conversion expression	• BT.601 (16, 235/240) to RGB (0, 255) • BT.709 (16, 235/240) to RGB (0, 255) • BT.601 (0, 255) to RGB (0, 255) • BT.709 (16, 235/240) to RGB (16, 235)
		Target of conversion	WPF0
Changing number of colors	Output	Reducing RGB color depth	Dithering, lower-order bit truncation, or rounding
		YCbCr422/420	CbCr skipping or CbCr vertical skipping and horizontal skipping
α bit count conversion	Output	Bit reduction	Truncation, rounding, or comparison with threshold (for 1 bpp)

Table 9.4-2 Detail Function of VSPD (4/4)

• Image Compositing			
α blending	Input α value selection	RGB	Pixel α, fixed α value, α plane, or 1-bit α converted from the color specified for pixels
		YCbCr	α plane, fixed α value, or 1-bit α converted from the color specified for pixels
	α blending expression	Layer A: Upper layer Layer B: Lower layer	xA + yB, xA − yB Coefficients x and y should be selected from the following. Fixed α value, (α for layer A), (1 − α for layer A), (α for layer B), (1 − α for layer B)
	Output α value selection	RGB	Fixed α value, x (α for layer A) + y (α for layer B), x (α for layer A) – y (α for layer B) Coefficients x and y should be selected from the following. Fixed α value, (α for layer A), (1 − α for layer A), (α for layer B), (1 − α for layer B)
	Blending layers	Number of layers	Two layers selected from RPF0 to RPF1 and video processing function output, and virtual RPF; three layers in total
		Order of layers	The order of three layers selected from RPF0 to RPF1, virtual RPF, and video processing function output can be changed as desired.
	α plane	Format	8 bpp or 1 bpp (α value can be specified through register)
		Input source	RPF0 to RPF1
	Fixed α value	Format	8 bpp
		Input source	RPF0 to RPF1, virtual RPF, or video processing function output
Raster operation	ROP2 (between input channels)	Operator	16 types (OpenGL2.0 is supported)
		Sources of operation	RPF0 to RPF1, virtual RPF, and video processing function output
		Operation control	RGB/YCbCr and α are operated separately.
	ROP3 (between input channels)	Operator	256-type ROP3 is available by combining ROP2 operations
		Sources of operation	RPF0 to RPF1, virtual RPF, and video processing function output
		Operation control	RGB/YCbCr and α are operated separately.
• Color Adjustment Function			
1D-LUT	LUT configuration		Independent R/Y, G/Cb, and B/Cr. 256 entries each
• Direct connection to Display module			
Display I/F	Destination		DU
	Direct connection		Can transfer data without going through external memory to DU
Line padding	Padding cycles		1 to 32 cycles
	Padding pattern		Arbitrarily setting up to 32 cycles

9.4.1.2.3 DU

DU has the following sub modules.

(1) REG

The REG controls registers for DU. The DU has APB4 interface, its transaction is transfer with no wait states.

(2) LIFC

The LIFC is the interface block from/to VSPD. The LIFC receives image data from VSPD, sends to PBUF in a pixel unit. The LIFC continues to receive the image data from VSPD unless FIFO will be full.

(3) PBUF

The PBUF is the asynchronous FIFO which convert clock domain of image data from System Bus Clock to Video Clock. The PBUF monitors the FIFO underflow.

(4) DIFC

The DIFC is the display timing master. The DIFC controls the polarity and the timing of the Video Output signals. Controllable timings are the followings.

- Hfront: Horizontal front porch (pixels)
- Hback: Horizontal back porch (pixels)
- Hsync: Horizontal sync active (pixels)
- Hactive: Horizontal active area (pixels)
- Vfront: Vertical front porch (lines)
- Vback: Vertical back porch (lines)
- Vsync: Vertical sync active (lines)
- Vactive: Vertical active area (lines)

(5) INTC

The INTC generates the interrupt when PBUF will detect a FIFO underflow. The interrupt is level signal.

9.4.1.3 Clock and Interrupt signal

The following clocks are supplied to the LCDC.

Table 9.4-3 Clock List

Clock Name	Description	Frequency (MHz)
LCDC_0_clk_a	AXI clock.	400/200/100/50/13
LCDC_0_clk_p	APB clock.	100
LCDC_0_clk_d	Video clock.	5.44 to 187.5
LCDC_1_clk_a	AXI clock.	400/200/100/50/13
LCDC_1_clk_p	APB clock.	100
LCDC_1_clk_d	Video clock.	5.44 to 187.5

The LCDC transmits the following interrupt signals to the interrupt controller.

Table 9.4-4 Interrupt List

Signal Name	Interrupt Condition	Interrupt Source Register
VSPD_INT	Frame End	VI6_WPF0_IRQ_STA.FRE
	Display List Frame End	VI6_WPF0_IRQ_STA.DFE
	DU Connection UnderRun Error	VI6_WPF0_IRQ_STA.UND
DU_INT	PBUF FIFO Underflow	DU_MSR0.ST_PB_RUF

Note: Refer to **(13) WPF0 Interrupt Status Register (LCDC_VSPD_VI6_WPF0_IRQ_STA)** and **9.4.4.2.1 DU Module Control Register 0 (LCDC_DU_MCR0)** for details.

9.4.2 FCPVD Registers

The FCPVD contains the registers which can directly be read and written to by the host CPU. The host can access any long word (32 bits) in the registers. Therefore, read-modify-write is needed to change partial bits in the 32-bit register.

Never access the registers when the FCPVD is in the module standby mode (with the operation clock placed in hold status).

Table 9.4-5 FCPVD Register Base Address

Base Register Name	LCDC Ch.	Base Address
<LCDC0_fcpvd_base>	LCDC0	0_1647_0000h (5647_0000h*1, 4647_0000h*2)
<LCDC1_fcpvd_base>	LCDC1	0_164A_0000h (564A_0000h*1, 464A_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

9.4.2.1 List of FCPVD Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
FCPV version control register	LCDC_FCPVD_FCP_VCR	0000_0109h	0000h	32

9.4.2.2 FCPVD Register Description

The prefix (LCDC_FCPVD_) of the register names is omitted in this and subsequent sections.

9.4.2.2.1 FCPVD Version Control Register (LCDC_FCPVD_FCP_VCR)

FCP_VCR is the version control register of the FCPVD. The value read is fixed.

Access Size : 32 bits
Address : <LCDC_fcpvd_base> + 0000h
Initial Value : 0000_0109h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATEGORY[7:0]								REVISION[7:0]							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 8	CATEGORY [7:0]	1h	R	FCP use case category.
7 to 0	REVISION[7:0]	9h	R	Version of LSI product 09h: FCP revision is the Simple LCDC

9.4.3 VSPD Registers

In the following, “n” in a register name or a offset address indicates an integer, and the range of value n is defined when necessary. For RPFn and WPFn, when the range of value n is not defined, RPFn (n = 0, 1) and WPFn (n = 0) are assumed.

Table 9.4-6 VSPD Register Base Address

Base Register Name	LCDC Ch.	Base Address
<LCDC0_vspd_base>	LCDC0	0_1648_0000h (5648_0000h*1, 4648_0000h*2)
<LCDC1_vspd_base>	LCDC1	0_164B_0000h (564B_0000h*1, 464B_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

9.4.3.1 List of VSPD Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
VSPD Start Register 0	LCDC_VSPD_VI6_CMD0	0000_0000h	0000h	32
Reserve	-	-	0004h to 000Fh	-
Clock Control Register 0	LCDC_VSPD_VI6_CLK_CTRL0	0000_0000h	0010h	32
Clock Control Register 1	LCDC_VSPD_VI6_CLK_CTRL1	0000_0000h	0014h	32
Dynamic Clock Stop Control Register	LCDC_VSPD_VI6_CLK_DCSWT	0000_0000h	0018h	32
Dynamic Clock Stop Disable Register 0	LCDC_VSPD_VI6_CLK_DCSM0	0000_0000h	001Ch	32
Dynamic Clock Stop Disable Register 1	LCDC_VSPD_VI6_CLK_DCSM1	0000_0000h	0020h	32
Reserve	-	-	0024h to 0027h	-
Software Reset Register	LCDC_VSPD_VI6_SRESET	0000_0000h	0028h	32
Module Reset Enable Register 0	LCDC_VSPD_VI6_MRESET_ENB 0	0000_0000h	002Ch	32
Module Reset Enable Register 1	LCDC_VSPD_VI6_MRESET_ENB 1	0000_0000h	0030h	32
Module Reset Issuing Register	LCDC_VSPD_VI6_MRESET	0000_0000h	0034h	32
Operating Status Register	LCDC_VSPD_VI6_STATUS	0000_0000h	0038h	32
Reserve	-	-	003Ch to 0047h	-
WPF0 Interrupt Enable Register	LCDC_VSPD_VI6_WPF0_IRQ_EN B	0000_0000h	0048h	32
WPF0 Interrupt Status Register	LCDC_VSPD_VI6_WPF0_IRQ_ST A	0000_0000h	004Ch	32
Reserve	-	-	0050h to 0077h	-
Display-0 Interrupt Enable Register	LCDC_VSPD_VI6_DISP0_IRQ_EN B	0000_0000h	0078h	32
Display-0 Interrupt Status Register	LCDC_VSPD_VI6_DISP0_IRQ_ST A	0000_0000h	007Ch	32
Reserve	-	-	0080h to 00FFh	-
Display List Control Register	LCDC_VSPD_VI6_DL_CTRL	0000_0000h	0100h	32
Reserve	-	-	0108h to 0113h	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Display List-0 Data Swapping Register	LCDC_VSPD_VI6_DL_SWAP0	0000_0000h	0114h	32
Reserve	-	-	0118h to 011Bh	-
Extended Display List-0 Control Register	LCDC_VSPD_VI6_DL_EXT_CTRL0	0000_0000h	011Ch	32
Display List Body Size Register 0	LCDC_VSPD_VI6_DL_BODY_SIZE0	0000_0000h	0120h	32
Display List-0 Header Address Register	LCDC_VSPD_VI6_DL_HDR_ADDR0	0000_0000h	0104h	32
Reserve	-	-	0124h to 012Fh	-
Display List-0 Header Reference Address Register	LCDC_VSPD_VI6_DL_HDR_REF_ADDR0	0000_0000h	0130h	32
Reserve	-	-	0134h to 0157h	-
Display List-0 Wake Up Counter Register	LCDC_VSPD_VI6_DL_WUPCNT0	0000_0000h	0158h	32
Reserve	-	-	015Ch to 02FFh	-
RPF0 Basic Read Size Register	LCDC_VSPD_VI6_RPF0_SRC_BSIZE	0000_0000h	0300h	32
RPF0 Extended Read Size Register	LCDC_VSPD_VI6_RPF0_SRC_ESIZE	0000_0000h	0304h	32
RPF0 Input Format Register	LCDC_VSPD_VI6_RPF0_INFMT	0000_0000h	0308h	32
RPF0 Data Swapping Register	LCDC_VSPD_VI6_RPF0_DSWAP	0000_0000h	030Ch	32
RPF0 Display Location Register	LCDC_VSPD_VI6_RPF0_LOC	0000_0000h	0310h	32
RPF0 a Plane Selection Control Register	LCDC_VSPD_VI6_RPF0_ALPH_SEL	0000_0000h	0314h	32
RPF0 Virtual Plane Color Information Register	LCDC_VSPD_VI6_RPF0_VRTCOL_SET	0000_0000h	0318h	32
RPF0 Mask Control Register	LCDC_VSPD_VI6_RPF0_MSKCTRL	0000_0000h	031Ch	32
RPF0 IROP-SRC Input Value Register 0	LCDC_VSPD_VI6_RPF0_MSKSET0	0000_0000h	0320h	32
RPF0 IROP-SRC Input Value Register 1	LCDC_VSPD_VI6_RPF0_MSKSET1	0000_0000h	0324h	32
RPF0 Color Keying Control Register	LCDC_VSPD_VI6_RPF0_CKEY_CTRL	0000_0000h	0328h	32
RPF0 Color Keying Color Setting Register 0	LCDC_VSPD_VI6_RPF0_CKEY_SET0	0000_0000h	032Ch	32
RPF0 Color Keying Color Setting Register 1	LCDC_VSPD_VI6_RPF0_CKEY_SET1	0000_0000h	0330h	32
RPF0 Source Picture Memory Stride Setting Register	LCDC_VSPD_VI6_RPF0_SRCM_PSTRIDE	0000_0000h	0334h	32
RPF0 Source α Memory Stride Setting Register	LCDC_VSPD_VI6_RPF0_SRCM_ASTRIDE	0000_0000h	0338h	32
RPF0 Source Y/RGB Address Register	LCDC_VSPD_VI6_RPF0_SRCM_ADDR_Y	0000_0000h	033Ch	32
RPF0 Source Chroma Address Register 0	LCDC_VSPD_VI6_RPF0_SRCM_ADDR_C0	0000_0000h	0340h	32
RPF0 Source Chroma Address Register 1	LCDC_VSPD_VI6_RPF0_SRCM_ADDR_C1	0000_0000h	0344h	32
RPF0 Source α Address Register	LCDC_VSPD_VI6_RPF0_SRCM_ADDR_AI	0000_0000h	0348h	32
Reserve	-	-	034Ch to 034Fh	-
RPF0 Bus Access Control Register	LCDC_VSPD_VI6_RPF0_BAC	0000_0000h	0350h	32
Reserve	-	-	0354h to 036Bh	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
RPF0 Multiple Alpha Control Register	LCDC_VSPD_VI6_RPF0_MULT_ALPH	0000_0000h	036Ch	32
Reserve	-	-	0370h to 03FFh	-
RPF1 Basic Read Size Register	LCDC_VSPD_VI6_RPF1_SRC_BSIZE	0000_0000h	0400h	32
RPF1 Extended Read Size Register	LCDC_VSPD_VI6_RPF1_SRC_ESIZE	0000_0000h	0404h	32
RPF1 Input Format Register	LCDC_VSPD_VI6_RPF1_INFMT	0000_0000h	0408h	32
RPF1 Data Swapping Register	LCDC_VSPD_VI6_RPF1_DSWAP	0000_0000h	040Ch	32
RPF1 Display Location Register	LCDC_VSPD_VI6_RPF1_LOC	0000_0000h	0410h	32
RPF1 a Plane Selection Control Register	LCDC_VSPD_VI6_RPF1_ALPH_SEL	0000_0000h	0414h	32
RPF1 Virtual Plane Color Information Register	LCDC_VSPD_VI6_RPF1_VRTCOL_SET	0000_0000h	0418h	32
RPF1 Mask Control Register	LCDC_VSPD_VI6_RPF1_MSKCTRL	0000_0000h	041Ch	32
RPF1 IROP-SRC Input Value Register 0	LCDC_VSPD_VI6_RPF1_MSKSET0	0000_0000h	0420h	32
RPF1 IROP-SRC Input Value Register 1	LCDC_VSPD_VI6_RPF1_MSKSET1	0000_0000h	0424h	32
RPF1 Color Keying Control Register	LCDC_VSPD_VI6_RPF1_CKEY_CTRL	0000_0000h	0428h	32
RPF1 Color Keying Color Setting Register 0	LCDC_VSPD_VI6_RPF1_CKEY_SET0	0000_0000h	042Ch	32
RPF1 Color Keying Color Setting Register 1	LCDC_VSPD_VI6_RPF1_CKEY_SET1	0000_0000h	0430h	32
RPF1 Source Picture Memory Stride Setting Register	LCDC_VSPD_VI6_RPF1_SRCM_PSTRIDE	0000_0000h	0434h	32
RPF1 Source α Memory Stride Setting Register	LCDC_VSPD_VI6_RPF1_SRCM_ASTRIDE	0000_0000h	0438h	32
RPF1 Source Y/RGB Address Register	LCDC_VSPD_VI6_RPF1_SRCM_ADDR_Y	0000_0000h	043Ch	32
RPF1 Source Chroma Address Register 0	LCDC_VSPD_VI6_RPF1_SRCM_ADDR_C0	0000_0000h	0440h	32
RPF1 Source Chroma Address Register 1	LCDC_VSPD_VI6_RPF1_SRCM_ADDR_C1	0000_0000h	0444h	32
RPF1 Source α Address Register	LCDC_VSPD_VI6_RPF1_SRCM_ADDR_AI	0000_0000h	0448h	32
Reserve	-	-	044Ch to 044Fh	-
RPF1 Bus Access Control Register	LCDC_VSPD_VI6_RPF1_BAC	0000_0000h	0450h	32
Reserve	-	-	0454h to 046Bh	-
RPF1 Multiple Alpha Control Register	LCDC_VSPD_VI6_RPF1_MULT_ALPH	0000_0000h	046Ch	32
Reserve	-	-	0470h to 0FFFh	-
WPF0-Source-RPF Register	LCDC_VSPD_VI6_WPF0_SRCRPF	0000_0000h	1000h	32
WPF0 Horizontal Input Size Clipping Register	LCDC_VSPD_VI6_WPF0_HSZCLIP	0000_0000h	1004h	32
WPF0 Vertical Input Size Clipping Register	LCDC_VSPD_VI6_WPF0_VSZCLIP	0000_0000h	1008h	32
WPF0 Output Format Register	LCDC_VSPD_VI6_WPF0_OUTFMT	0000_0000h	100Ch	32
WPF0 Data Swapping Register	LCDC_VSPD_VI6_WPF0_DSWAP	0000_0000h	1010h	32
WPF0 Rounding Control Register	LCDC_VSPD_VI6_WPF0_RNDCTRL	0000_0000h	1014h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	1018h to 101Bh	-
WPF0 Destination Y Plane Memory Stride Register	LCDC_VSPD_VI6_WPF0_DSTM_STRIDE_Y	0000_0000h	101Ch	32
WPF0 Destination C Plane Memory Stride Register	LCDC_VSPD_VI6_WPF0_DSTM_STRIDE_C	0000_0000h	1020h	32
WPF0 Destination Y/RGB Address Register	LCDC_VSPD_VI6_WPF0_DSTM_ADDR_Y	0000_0000h	1024h	32
WPF0 Destination Chroma Address Register 0	LCDC_VSPD_VI6_WPF0_DSTM_ADDR_C0	0000_0000h	1028h	32
WPF0 Destination Chroma Address Register 1	LCDC_VSPD_VI6_WPF0_DSTM_ADDR_C1	0000_0000h	102Ch	32
Reserve	-	-	1030h to 1033h	-
WPF0 LIF Write Back Control Register	LCDC_VSPD_VI6_WPF0_WRBCK_CTRL	0000_0000h	1034h	32
Reserve	-	-	1038h to 1FFFh	-
RPF0 Routing Register	LCDC_VSPD_VI6_DPR_RPF0_ROUTE	0000_0000h	2000h	32
RPF1 Routing Register	LCDC_VSPD_VI6_DPR_RPF1_ROUTE	0000_0000h	2004h	32
Reserve	-	-	2008h to 2013h	-
WPF0 Timing Control Register	LCDC_VSPD_VI6_DPR_WPF0_FPORCH	0000_0000h	2014h	32
Reserve	-	-	2018h to 203Bh	-
LUT Routing Register	LCDC_VSPD_VI6_DPR_LUT_ROUTE	0000_0000h	203Ch	32
Reserve	-	-	2040h to 204Fh	-
BRS Routing Register	LCDC_VSPD_VI6_DPR_BRS_ROUTE	0000_0000h	2050h	32
Reserve	-	-	2054h to 27FFh	-
LUT Control Register	LCDC_VSPD_VI6_LUT_CTRL	0000_0000h	2800h	32
Reserve	-	-	2804h to 38FFh	-
BRS Input Control Register	LCDC_VSPD_VI6_BRS_INCTRL	0000_0000h	3900h	32
Size Register of BRS Input Virtual RPF	LCDC_VSPD_VI6_BRS_VIRRPF_SIZE	0000_0000h	3904h	32
Display Location Register of BRS Input Virtual RPF	LCDC_VSPD_VI6_BRS_VIRRPF_LOC	0000_0000h	3908h	32
Color Information Register of BRS Input Virtual RPF	LCDC_VSPD_VI6_BRS_VIRRPF_COL	0000_0000h	390Ch	32
BRS Control Register A	LCDC_VSPD_VI6_BRSA_CTRL	0000_0000h	3910h	32
BRS Blend Control Register A	LCDC_VSPD_VI6_BRSA_BLD	0000_0000h	3914h	32
BRS Control Register B	LCDC_VSPD_VI6_BRSB_CTRL	0000_0000h	3918h	32
BRS Blend Control Register B	LCDC_VSPD_VI6_BRSB_BLD	0000_0000h	391Ch	32
Reserve	-	-	3920h to 3AFFh	-
LIF0 Control Register	LCDC_VSPD_VI6_LIF0_CTRL	0000_0000h	3B00h	32
LIF0 Clock Stop Buffer Control Register	LCDC_VSPD_VI6_LIF0_CSBTH	0000_0000h	3B04h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	3B08h to 3B0Bh	-
LIF0 Buffer Attribute Register	LCDC_VSPD_VI6_LIF0_LBA	0000_0000h	3B0Ch	32
Reserve	-	-	3B10h to 3B2Fh	-
LIF0 Padding Line Cycle Register	LCDC_VSPD_VI6_LIF0_PADLN_CYC	0000_0000h	3B30h	32
LIF0 Padding Line Pattern Register	LCDC_VSPD_VI6_LIF0_PADLN_PT	0000_0000h	3B34h	32
LIF0 Padding Line Value Register	LCDC_VSPD_VI6_LIF0_PADLN_VAL	0000_0000h	3B38h	32
LIF0 Padding Line Size Register	LCDC_VSPD_VI6_LIF0_PADLN_SIZE	0000_0000h	3B3Ch	32

Note: Read only bits or write only bits or read/write bits are mixed in the registers.

9.4.3.2 VSPD Register Description

The prefix (LCDC_VSPD_) of the register names is omitted in this and subsequent sections.

9.4.3.2.1 Notational Conventions for Registers and Bit Fields

This section uses the following notational conventions for the VSPD registers and bit fields.

1. The names of registers and bits are written in uppercase.
2. A bit or bit field in a register is indicated as [register name.bit name]. For example, the STRCMD bit in the VI6_CMD0 register is indicated as VI6_CMD0.STRCMD.
3. Lowercase “n” in a register name or a bit name indicates an integer and the range of value n is defined when necessary. For RPFn, and WPFn, when the range of value n is not defined, RPFn (n = 0, 1), WPFn (n = 0) are assumed.
4. In each subsection for register description in **9.4.3.2 VSPD Register Description**, when only a bit name is written without showing its register name, the bit is in the register described in that subsection.
5. A wildcard (*) indicates any characters in a name and represents all registers or bits that match the specified first part of a name. For example, when there are two registers VI6_RPF_SRC_BSIZE and VI6_RPF_SRC_ESIZE, VI6_RPF_SRC_* indicates both registers.

9.4.3.2.2 Register Classification

The VSPD registers are arranged in the following order; the general control registers to display list control registers control operation of the entire VSPD, and the other registers control each image processing and specify parameters for the processing. The functions of the registers are described in this order starting from **9.4.3.2.4 General Control Registers**.

1. General control registers (VI6_CMDn (n = 0, 1), VI6_SRESET, VI6_STATUS, VI6_WPFn_IRQ_* (n = 0))
2. Display list control registers (VI6_DL_*)
3. RPF control registers (VI6_RPFn_* (n = 0, 1))
4. WPF control registers (VI6_WPFn_* (n = 0))
5. DPR control registers (VI6_DPR_*)
6. LUT control register (VI6_LUT_CTRL)
7. BRS control registers (VI6_BRS_*)
8. LIFn control registers (VI6_LIFn_* (n = 0))

The VSPD has two RPF channels and the register configuration is the same for all of RPFn (n = 0, 1). However, some bit fields have restrictions in certain RPFs. These restrictions are included in the description of the corresponding bit fields and registers. Likewise, the register configuration is the same for all WPFn (n = 0), but some bit fields have restrictions in certain WPFs; the restrictions are included in the description of the corresponding bit fields and registers.

For each register address, refer to **9.4.3.1 List of VSPD Registers**.

9.4.3.2.3 Restrictions on Access to Registers and Lookup Tables

The VSPD has control registers and lookup tables. All VSPD registers are writable and readable by only 32 bits unit. To write partial bits in each register, read-modify-write is needed. When accessing the addresses where these registers and lookup tables are allocated, the following restrictions should be satisfied. If any restriction is violated, the VSPD will not operate correctly.

1. For the read-only bits and reserved bits in all VSPD registers, writing 1 is prohibited unless otherwise specified.
2. Addresses undefined in **9.4.3.1 List of VSPD Registers** are reserved areas and write access is prohibited in these areas.
3. For all registers and lookup tables, except VI6_CMDn, VI6_SRESET, VI6_*IRQ* and two plane registers such as VI6_DL_HDR_ADDRn, VI6_DL_BODY_SIZE0, modifying register values during operation of the module is prohibited. Modify registers while the corresponding module is stopped. For the operating status of the target module, refer to **9.4.5.4 Concept of VSPD Operation Starting and Stopping**.
4. There are three types about General control registers (**9.4.3.2.4 General Control Registers**) and Display List Control Registers (**9.4.3.2.5 Display List Control Registers**) as below.
   - Controlling WPF0 (ex. VI6_CMD0, VI6_SRESET.SRST0)
   - Common setting of WPF0 (ex. VI6_DL_CTRL.AR_WAIT [15:0])

Table 9.4-7 Correspondence between Modules and Register Names

Module Name	Register Name
RPFn (n = 0, 1)	VI6_RPFn_*
WPFn (n = 0)	VI6_WPFn_*
DPR	VI6_DPR_*
LUT	VI6_LUT_CTRL
BRS	VI6_BRS_*
LIFn (n = 0)	VI6_LIFn_*

9.4.3.2.4 General Control Registers

(1) VSPD Start Register 0 (LCDC_VSPD_VI6_CMD0)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 0000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	UPDHDR	-	-	-	STRCMD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	UPDHDR	0h	RW	Reserved state of updating Display List Header (DLH) address. UPDHDR can be negated by software reset (VI6_SRESET.SRST0). [Write] 0b: NOP 1b: Do not set 1b to this bit [Read] 0b: Not reserved to update DLH address. 1b: Reserved to update DLH address. Set 0b to this bit when STRCMD bit is set to 1b.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	STRCMD	0h	RW	Start reservation of WPF VI6_CMD0.STRCMD controls WPF0. Writing 1b to this bit starts WPF0 in VSPD. Set this bit for activation only after all register settings in each output channel have been completed. If WPF0 is idle, WPF0 starts right after this bit is set to 1b. If WPF0 is active, writing 1b to this bit reserves starting WPF0 operation. Wait to set 1b to VI6_CMD0.STRCMD until this bit is read as 0b. VI6_CMD0.STRCMD can be negated by software reset (VI6_SRESET.SRST0). [Write] 0b: Start reservation of WPF0 is canceled. 1b: Start reservation of WPF0 is set. [Read] 0b: Starting VSPD is not reserved. 1b: Starting VSPD is reserved.

The basic concept of image processing operation started by activating the VSPD is shown in **Figure 9.4-6**. The actual data input/output is executed by the MAU, which is the bus interface module, as described in **9.4.1.2 Block Diagram**, but conceptually the RPF works as the data entry point to the VSPD and the WPF works as the data exit point. To process images through the VSPD, the RPF (entrance) and WPF (exit) should be connected and a data path from the RPF to the WPF should be formed.

To connect RPFn to WPFn, specify RPFn as the source RPF for WPFn in VI6_WPFn_SRCRPF, which is a register for WPFn (refer to **(1) WPF0-Source-RPF Register (LCDC_VSPD_VI6_WPF0_SRCRPF)**). This setting determines that RPFn will be started when WPFn is started through VI6_CMDn.

A data path to execute desired image processing should then be formed between the RPF (entrance) and WPF (exit). To form a data path, connect the necessary function modules in the VSPD between RPF and WPF. This function is

provided by the DPR; specify the information for each module connection in data path routing registers VI6_DPR_*_ROUTE (refer to **9.4.3.2.8 DPR Control Registers**).

After a data path is formed (RPFn → WPFn) as described above, starting output module WPFn in the VSPD through VI6_CMDn starts all function modules connected to WPFn and the desired image processing is executed. According to this design concept, starting a WPF module means starting the VSPD.

There are two types of data path configuration in the VSPD; one is “a single input module to a single output module” as shown in **Figure 9.4-6** (A), and the other is “multiple input modules to a single output module” as shown in **Figure 9.4-6** (B).

Figure 9.4-6 (A) shows an example of a configuration where modules with single input and single output are implemented through the DPR.

Figure 9.4-6 (B) shows another configuration example where the module with multiple input and single output is implemented through the DPR.

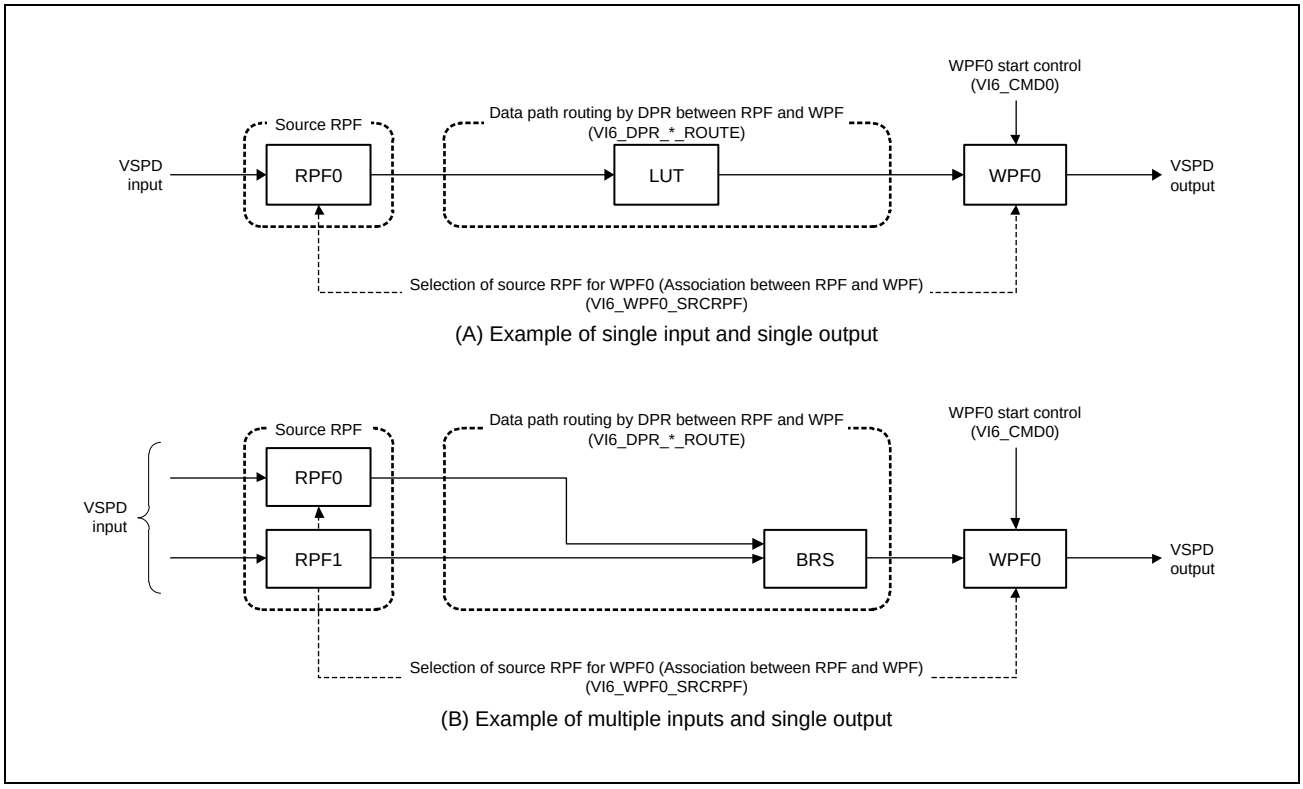


Figure 9.4-6 Basic Concept of VSPD Startup

(2) Clock Control Register 0 (LCDC_VSPD_VI6_CLK_CTRL0)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	GCS0	-	-	-	-	-	-	-	-	-	-	-	GCS1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	GCS2[3:0]				-	-	-	GCS3[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	GCS0	0h	RW	Clock Control Setting0
27 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	GCS1	0h	RW	Clock Control Setting1
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	GCS2[3:0]	0h	RW	Clock Control Setting2
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4 to 0	GCS3[4:0]	00h	RW	Clock Control Setting3

VSPD can stop its operating clock for reducing power consumption.

To enable clock stop function, set following registers:

- VI6_CLK_CTRL0 = 0000_0000h
- VI6_CLK_CTRL1 = 0000_0000h
- VI6_CLK_DCSWT = 0000_0808h
- VI6_CLK_DCSM0 = 0000_0000h
- VI6_CLK_DCSM1 = 0000_0000h

To disable clock stop function, set following registers:

- VI6_CLK_CTRL0 = 1001_0F1Fh
- VI6_CLK_CTRL1 = FF10_FFFFh
- VI6_CLK_DCSWT = 0033_0808h
- VI6_CLK_DCSM0 = 1FFF_0F1Fh
- VI6_CLK_DCSM1 = FF10_FFFFh

When the clock gating is enabled, the supply of the clock signal is stopped for the module which is not operated. If the bit field of this register is changed from 0 to 1, the supply of the clock signal for the module is immediately started. If

the bit field of the register is changed from 1 to 0, the clock supply is immediately stopped. Therefore, please do not change this register field from 1 to 0 in operation because the VSPD may be stalled.

(3) Clock Control Register 1 (LCDC_VSPD_VI6_CLK_CTRL1)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 0014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GCS4[7:0]								-	-	-	GCS6	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	RW	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GCS5[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GCS4[7:0]	00h	RW	Clock Control Setting4
23 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	GCS6	0h	RW	Clock Control Setting6
19 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	GCS5[15:0]	0000h	RW	Clock Control Setting5

See **(2) Clock Control Register 0 (LCDC_VSPD_VI6_CLK_CTRL0)** for detail.

(4) Dynamic Clock Stop Control Register (LCDC_VSPD_VI6_CLK_DCSWT)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 0018h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	DCC0[1:0]		-	-	DCC1	DCC2
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSTPW[7:0]								CSTRW[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
21, 20	DCC0[1:0]	0h	RW	Dynamic Clock Control Setting0
19, 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	DCC1	0h	RW	Dynamic Clock Control Setting1
16	DCC2	0h	RW	Dynamic Clock Control Setting2
15 to 8	CSTPW[7:0]	00h	RW	Dynamic Clock Stop Control 1 Always specify 8.
7 to 0	CSTRW[7:0]	00h	RW	Dynamic Clock Stop Control 2 Always specify 8.

See **(2) Clock Control Register 0 (LCDC_VSPD_VI6_CLK_CTRL0)** for detail.

(5) Dynamic Clock Stop Disable Register 0 (LCDC_VSPD_VI6_CLK_DCSM0)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 001Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	DCD0[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	DCD1[3:0]				-	-	-	DCD2[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	DCD0[12:0]	0000h	RW	Dynamic Clock Stop Disable Setting0
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	DCD1[3:0]	0h	RW	Dynamic Clock Stop Disable Setting1
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4 to 0	DCD2[4:0]	00h	RW	Dynamic Clock Stop Disable Setting2

See **(2) Clock Control Register 0 (LCDC_VSPD_VI6_CLK_CTRL0)** for detail.

(6) Dynamic Clock Stop Disable Register 1 (LCDC_VSPD_VI6_CLK_DCSM1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCD3[7:0]								-	-	-	DCD5	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	RW	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCD4[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DCD3[7:0]	00h	RW	Dynamic Clock Stop Disable Setting3
23 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	DCD5	0h	RW	Dynamic Clock Stop Disable Setting5
19 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	DCD4[15:0]	0000h	RW	Dynamic Clock Stop Disable Setting4

See **(2) Clock Control Register 0 (LCDC_VSPD_VI6_CLK_CTRL0)** for detail.

(7) Software Reset Register (LCDC_VSPD_VI6_SRESET)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 0028h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SRST0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	SRST0	0h	RW	WPF0 Software Reset Writing 1b to this bit aborts the current processing in WPF0 (the partially-completed image undergoing processing is output). The period until this software reset processing is completed depends on the bus state. When this reset processing is completed, the VI6_WPF0_IRQ_STA.FRE interrupt source bit is set to 1b; when the FRE interrupt is enabled, the FRE end interrupt is output to notify the end of the reset processing. This bit is always read as 0b. 0b: NOP 1b: WPF0 software reset*[1]

Note 1. Applying a software reset to each WPF has the following restrictions.

1. A software reset can be applied to only one of WPF0 through single write access to VI6_SRESET.
2. After a software reset is issued, no more software reset can be issued to another WPF until the issued software reset processing is completed
3. The end of software reset processing is notified through the FRE bit in VI6_WPFn_IRQ_STA, but the software reset issued while WPF is stopped is ignored as NOP. As it takes a while until the reset is actually issued after the reset bit is set, the VSPD may complete operation before the reset is actually issued. In this case, no interrupt is output for the software reset that is issued after the VSPD completes operation.
4. If a software reset is issued during downloading of a display list, the downloading processing is not aborted. After the end of downloading that is in progress when a software reset is issued, a frame end interrupt is output.

(8) Module Reset Enable Register 0 (LCDC_VSPD_VI6_MRESET_ENB0)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 002Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	MRSTE0[1:0]		-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	MRSTE1[3:0]				-	-	-	MRSTE2[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29, 28	MRSTE0[1:0]	0h	RW	Module Reset Enable 0
27 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	MRSTE1[3:0]	0h	RW	Module Reset Enable 1
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4 to 0	MRSTE2[4:0]	00h	RW	Module Reset Enable 2

This register is for purpose of h/w debugging.

(9) Module Reset Enable Register 1 (LCDC_VSPD_VI6_MRESET_ENB1)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 0030h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MRSTE3[7:0]								-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MRSTE4[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	MRSTE3[7:0]	00h	RW	Module Reset Enable 3
23 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	MRSTE4[15:0]	0000h	RW	Module Reset Enable 4

This register is for purpose of h/w debugging.

(10) Module Reset Issuing Register (LCDC_VSPD_VI6_MRESET)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0034h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	MRST	0h	RW	Module Reset Assertion

This register is for purpose of h/w debugging.

(11) Operating Status Register (LCDC_VSPD_VI6_STATUS)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0038h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	FLDST0	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	SYS0_ACT	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	FLDST0	0h	R	Field status of previous frame of WPF0. 0b: Previous finished frame is TOP field. 1b: Previous finished frame is BOT field. This bit can be referred in case of AUTO-FLD or AUTO-DISP. This bit is changed at the timing of VI6_WPF0_IRQ_STA.FRE.
27 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	SYS0_ACT	0h	R	WPF0 Operating Status Each bit indicates the operating or stopped state of control channel n (WPF0). 0b: WPF0 is stopped. 1b: WPF0 is operating.
7 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(12) WPF0 Interrupt Enable Register (LCDC_VSPD_VI6_WPF0_IRQ_ENB)

Each bit controls the interrupt enable of the corresponding interrupt source.

0: Interrupt disabled
1: Interrupt enabled

Access Size : 32 bits
Address : <LCDC_vspd_base> + 0048h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	UNDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DFEE	FREE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	UNDE	0h	RW	Interrupt Enable for WPF0 Underrun in case of DU connection.
15 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	DFEE	0h	RW	Interrupt Enable for WPF0 Display List Frame End
0	FREE	0h	RW	Interrupt Enable for WPF0 Frame End

Each bit in VI6_WPFn_IRQ_STA is set to 1 when the corresponding interrupt source is generated. VI6_WPFn_IRQ_ENB specifies whether to output an interrupt signal for the generated source. When an interrupt is disabled in this register, no interrupt signal is generated even when the corresponding bit in VI6_WPFn_IRQ_STA is set to 1. When an interrupt is enabled in this register, an interrupt signal is output when the corresponding bit in VI6_WPFn_IRQ_STA is set to 1.

(13) WPF0 Interrupt Status Register (LCDC_VSPD_VI6_WPF0_IRQ_STA)

The read value from each bit is the status of the interrupt source, and the write access to each bit controls the interrupt status.

[Read Access] Interrupt Status

0: No interrupt
1: Interrupt activated

[Write Access] Interrupt Clear

0: The interrupt status is cleared to 0
1: Hold the interrupt status value

Access Size : 32 bits
Address : <LCDC_vspd_base> + 004Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	UND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DFE	FRE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	UND	0h	RW	Interrupt Status and Clear for WPF0 Underrun in case of DU connection. This interrupt source bit is set to 1b when data underrun occurs in case of DU connection. Timing that this bit is set to 1b is at end of 1 frame.
15 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	DFE	0h	RW	Interrupt Status and Clear for WPF0 Display List Frame End This interrupt source bit is set to 1b when VSPD completes one-frame processing while the current frame in enable value stored in the display list header is 1 (refer to **9.4.5.4 Concept of VSPD Operation Starting and Stopping**). When display lists are not used, this bit is not used. In this case, clear VI6_WPF0_IRQ_ENB.DFEE to 0b to mask the interrupt generation by this interrupt source. This bit can hold the most recent of two times of interrupt status at the maximum. When writing 0b to this bit, the oldest interrupt status is cleared. And the status bit becomes 0b after all interrupt statuses are cleared.
0	FRE	0h	RW	Interrupt Status and Clear for WPF0 Frame End This interrupt source bit is set to 1b when VSPD completes one-frame processing. This bit is also set to 1b when one-frame processing using a display list is completed. The interrupt status is set to 1b by any of the following conditions. a) Processing one frame is finished normally b) Software reset is issued during VSPD is processing. c) One frame's data from DU is displayed while VSPD couldn't transfer one frame data to DU. (in case of Linked with DU) This bit can hold the most recent of two times of interrupt status at the maximum. When writing 0b to this bit, the oldest interrupt status is cleared. And the status bit becomes 0b after all interrupt statuses are cleared.

VI6_WPFn_IRQ_STA indicates the state of the interrupt sources generated in the VSPD. Whether to output a VSPD interrupt when an interrupt source is generated and the corresponding bit is set to 1 is determined by the corresponding bit setting in VI6_WPFn_IRQ_ENB. While an interrupt is disabled in VI6_WPFn_IRQ_ENB, the VSPD does not output an interrupt signal even when an interrupt source is generated, but the source flag in this register is set to 1.

Note that the interrupt source bits in this register cannot be cleared by write access using a display list.

(14) Display-0 Interrupt Enable Register (LCDC_VSPD_VI6_DISP0_IRQ_ENB)

Each bit controls the interrupt enable of the corresponding interrupt source.

0: Interrupt disabled
1: Interrupt enabled

Access Size : 32 bits
Address : <LCDC_vspd_base> + 0078h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	DSTE	-	-	MAEE	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	RW	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	DSTE	0h	RW	Interrupt Enable for Display Start
7 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	MAEE	0h	RW	Interrupt Enable for Display Read Data End
4 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(15) Display-0 Interrupt Status Register (LCDC_VSPD_VI6_DISP0_IRQ_STA)

The read value from each bit is the status of the interrupt source, and the write access to each bit controls the interrupt status.

[Read Access] Interrupt Status

0: No interrupt
1: Interrupt activated

[Write Access] Interrupt Clear

0: The interrupt status is cleared to 0
1: Hold the interrupt status value

Access Size : 32 bits
Address : <LCDC_vspd_base> + 007Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	DST	-	-	MAE	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	RW	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	DST	0h	RW	Interrupt Status and Clear for Display Start This bit is set to 1b when LIF module transfers the first data to the display module at the beginning of each frame. The timing depends on the output buffer status of LIF module.
7 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	MAE	0h	RW	Interrupt Status and Clear for Display Read Data End This bit is set to 1b when all RPFs transfer the last data of the frame to LIF module. The RPF module which is not used by LIF module does not affect this bit.
4 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.4.3.2.5 Display List Control Registers

(1) Display List Control Register (LCDC_VSPD_VI6_DL_CTRL)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0100h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AR_WAIT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	DC2	-	-	-	DC1	-	-	-	DLE1	RLM0	CFM0	NH0	DLE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	RW	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	AR_WAIT[15:0]	0000h	RW	Display List Control Setting Always specify 256.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12	DC2	0h	RW	This bit doesn't affect anything to VSPD (NOP)
11 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	DC1	0h	RW	This bit doesn't affect anything to VSPD (NOP)
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	DLE1	0h	RW	This bit doesn't affect anything to VSPD (NOP)
3	RLM0	0h	RW	Loading two plane Registers Mode for WPF0. 0b: Reserved status (VI6_CMD0.UPDHDR) is accepted by next frame auto start. Two plane registers are downloaded by next_frame_start also. 1b: Reserved status (VI6_CMD0.UPDHDR) is not accepted by next frame auto start. pNextHeader in previous frame's DLH is used for loading DisplayList. Two plane registers are not downloaded by next_frame_start also.
2	CFM0	0h	RW	Continuous Frame Mode for Header-less Display List for WPF0 This bit determines whether the next frame is automatically started or not. When the updated flag of the display list, VI6_DL_BODY_SIZE0.UPD0, is not updated, the display list of the next frame is not transferred and the same register values are used for the next frame. When the value of VI6_DL_BODY_SIZE0.UPD0 is updated, the new display list is transferred. 0b: Stopped at the end of every frame 1b: The next frame is automatically started
1	NH0	0h	RW	Header-less Display List Mode This bit is used for specifying the header-less display list mode. In case of header-less mode, the number of the display lists is 1. The address of the display body is set in VI6_DL_HDR_ADDR0 register, and the body size is set in VI6_DL_BODY_SIZE0 register. When this bit is changed, make sure that VSPD is stopped. And also make sure the following before starting VSPD. - Header Address (VI6_DL_HDR_ADDR0) - Body Size (VI6_DL_BODY_SIZE0) in case of header-less mode 0b: Use Display List Header (Normal DL Mode) 1b: Do not use Display List Header (Header-less Mode) Note: When DLE0 bit is 0b, set 0b to NH0 bit.

Bit	Bit Name	Initial Value	R/W	Description
0	DLE0	0h	RW	Display List Enable/Disable for WPF0 Enables or disables the WPF0 display list function. When the display list function is enabled through this bit, all WPF processing channels work in display list mode. When using display lists, note the restrictions in **9.4.3.2.3 Restrictions on Access to Registers and Lookup Tables**. 0b: The display list function is disabled 1b: The display list function is enabled

(2) Display List-0 Header Address Register (LCDC_VSPD_VI6_DL_HDR_ADDR0)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0104h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DL_HDR_PTR-0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DL_HDR_PTR-0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DL_HDR_PTR-0[31:0]	0000_0000h	RW	Display List-0 Header Address These bits specify the address of the display list header to be read for display list-n in 16-byte units (the lower-order four bits are read-only). When WPF0 is first started in display list mode, the display list header is loaded from the address specified in this register. After loading of the header is completed, the register value of the display list address is updated to the next header address stored in the loaded header to prepare for loading of the next display list header. After that, this header address updating is repeated. A value from 0000_0000h to FFFF_FFF0h can be specified.

(3) Display List-0 Data Swapping Register (LCDC_VSPD_VI6_DL_SWAP0)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 0114h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	LWS	WDS	BTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

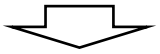
Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	LWS	0h	RW	Display List Data Swapping in long word Units The effect of this bit setting is defined in **Table 9.4-8**. 0b: Data swapping in long word (32-bit) units is disabled 1b: Data swapping in long word (32-bit) units is enabled
1	WDS	0h	RW	Display List Data Swapping in Word Units The effect of this bit setting is defined in **Table 9.4-8**. 0b: Data swapping in word (16-bit) units is disabled 1b: Data swapping in word (16-bit) units is enabled
0	BTS	0h	RW	Display List Data Swapping in Byte Units The effect of this bit setting is defined in **Table 9.4-8**. 0b: Data swapping in byte (8-bit) units is disabled 1b: Data swapping in byte (8-bit) units is enabled

Table 9.4-8 shows the data order before and after swapping according to the long word, word, and byte swapping settings.

When data order in memory for each format is the same as **Table 9.4-34**, set 111b to {LWS, WDS, BTS}. If data order is not the same as the definition, change data order within 8byte unit by these bits as shown in **Table 9.4-8**.

Table 9.4-8 Changing Data Order According to Display List Swap Register

Data order in memory											
Byte address	8n+0	8n+1	8n+2	8n+3	8n+4	8n+5	8n+6	8n+7	*_LWS	*_WDS	*_BTS
Data	0	1	2	3	4	5	6	7	1	1	1
	1	0	3	2	5	4	7	6	1	1	0
	2	3	0	1	6	7	4	5	1	0	1
	3	2	1	0	7	6	5	4	1	0	0
	4	5	6	7	0	1	2	3	0	1	1
	5	4	7	6	1	0	3	2	0	1	0
	6	7	4	5	2	3	0	1	0	0	1
	7	6	5	4	3	2	1	0	0	0	0



Data order defined in **Table 9.4-34**								
Byte address	8n+0	8n+1	8n+2	8n+3	8n+4	8n+5	8n+6	8n+7
Data	0	1	2	3	4	5	6	7

(4) Extended Display List-0 Control Register (LCDC_VSPD_VI6_DL_EXT_CTRL0)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 011Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	NWE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	POLINT[5:0]						-	-	DLPRI	EXPRI	-	-	-	EXT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	NWE	0h	RW	No Wait for Polling When this bit is set to 1b, the polling condition for extended display lists is always assumed to be true.
15 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 8	POLINT[5:0]	00h	RW	Extended Display List Command Control Always specify 2h.
7 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5	DLPRI	0h	RW	Display List Control 0 Always specify 1b.
4	EXPRI	0h	RW	Display List Control 1 Always specify 0b.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	EXT	0h	RW	Extended Display List for WPF0 Enables or disables the extended display list function. When extended display lists are used, the display list header size is 96 bytes; when they are not used, the header size is 80 bytes. 0b: No extended display lists are used 1b: Extended display lists are used [Note] When using extended display lists, be sure to also use normal display list mode (VI6_DL_CTRL.DLE0); executing only extended display lists is not possible. When the header-less display list mode is activated, this bit should be set to 0. The extended display list cannot be used with the header-less display list mode. Extended Display List is available only for VSPD/LIF0 to realized AUTO-FLD or AUTO-DISP.

(5) Display List Body Size Register-0 (LCDC_VSPD_VI6_DL_BODY_SIZE0)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 0120h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	UPD0	-	-	-	-	-	-	-	BS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BS0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	UPD0	0h	RW	Update Flag This bit controls the download of the display list at the next downloading timing in case that the header-less display list mode is used. Set 1b to this bit when display list need to be downloaded for next frame. This bit is automatically cleared to 0b after VI6_DL_HDR_ADDRn and VI6_DL_BODY_SIZE0.BS0 is downloaded in H/W side. When this bit is set to 1b, the value of VI6_DL_HDR_ADDR0 and VI6_DL_BODY_SIZE0.BS0 should not be changed. 0b: Updating display list for the next frame is not reserved 1b: Updating display list for the next frame is reserved
23 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16 to 0	BS0[16:0]	00000h	RW	Header-less Display List Body Size (WPF0) These bits are used for specifying the body size of the display list in case of header-less display list mode. The unit of the size is byte. The value should be set in multiples of 8.

See **9.4.5.8.2 Controlling two register planes using display lists** for detail.

(6) Display List-0 Header Reference Address Register (LCDC_VSPD_VI6_DL_HDR_REF_ADDR0)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0130h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DL_HDR_REF_PTR-0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DL_HDR_REF_PTR-0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DL_HDR_REF_ PTR-0[31:0]	00000000h	R	Display List-n reference Header Address Following value is read out from the VI6_DL_HDR_REF_ADDR0 for each period. (1) When H/W is reading display list from external memory, header address of the display list referred by VSPD-H/W is read out. (2) When H/W is not reading display list from external memory, the value of VI6_DL_HDR_ADDR0 is read out. For details, refer to **9.4.5.8.1 Operation flow of VSPD and DU**.

(7) Display List-0 Wake Up Counter Register (VI6_DL_WUPCNT0)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0158h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.4.3.2.6 RPF Control Registers

(1) RPFn Basic Read Size Registers (LCDC_VSPD_VI6_RPFn_SRC_BSIZE) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0300h (n = 0)
<LCDC_vspd_base> + 0400h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	BHSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	BVSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	BHSIZE[12:0]	0000h	RW	Horizontal Size of RPF Basic Read Area These bits specify the horizontal size of the basic area to be read from the external RAM by the RPFn. When the input format is YCbCr4:2:2 or YCbCr4:2:0, specify the size in 2-pixel units. A value from 1 to 1920 can be specified. Specify a value equal to or smaller than the extended read size (VI6_RPFn_SRC_ESIZE.EHSIZE).
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	BVSIZE[12:0]	0000h	RW	Vertical Size of RPF Basic Read Area These bits specify the vertical size of the basic area to be read from the external RAM by the RPFn. When the input format is YCbCr4:2:0, specify the size in 2-pixel units. A value from 1 to 1920 can be specified. Specify a value equal to or smaller than the extended read size (VI6_RPFn_SRC_ESIZE.EVSIZE).

Figure 9.4-7 shows the relationship between the basic read size and extended read size. The RPF reads data from the source memory area specified by the basic read size. The RPF repeats reading the basic read area in the horizontal and vertical directions up to the extended read size and sends the read data to the processing modules in the VSPD.

For basic read size reading, the reading start address, called the RPFn source image storing address, should be specified in VI6_RPFn_SRCM_ADDR_*. In the memory area where the basic read area image is stored, the distance (number of bytes) between addresses for lines n and n + 1 of two-dimensional image data, called the memory stride, should be specified in VI6_RPFn_SRCM_PSTRIDE.

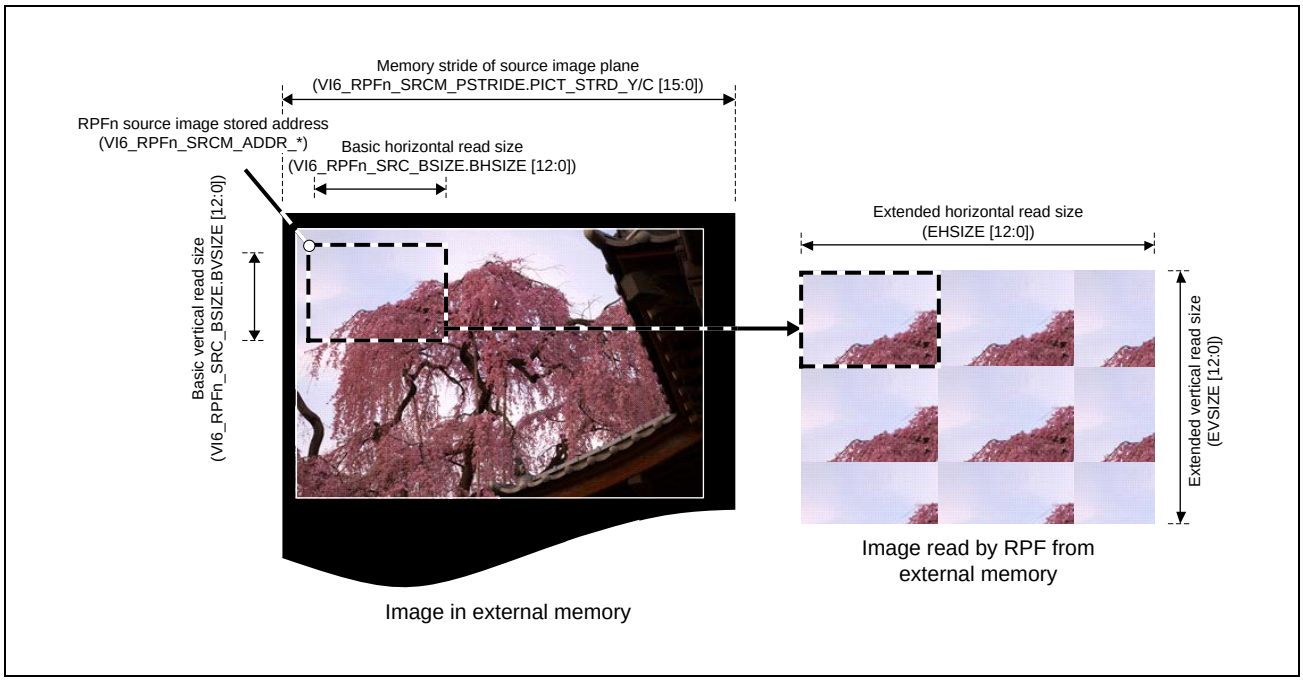


Figure 9.4-7 Relationship between Basic Read Size and Extended Read Size

Refer also to the following sections.

- **(2) RPFn Extended Read Size Registers (LCDC_VSPD_VI6_RPFn_SRC_ESIZE)**
- **(13) RPFn Source Picture Memory Stride Setting Registers (LCDC_VSPD_VI6_RPFn_SRCM_PSTRIDE)**
- **(15) RPFn Source Y/RGB Address Registers (LCDC_VSPD_VI6_RPFn_SRCM_ADDR_Y)**
- **(16) RPFn Source Chroma Address Registers 0 (LCDC_VSPD_VI6_RPFn_SRCM_ADDR_C0)**
- **(17) RPFn Source Chroma Address Registers 1 (LCDC_VSPD_VI6_RPFn_SRCM_ADDR_C1)**

(2) RPFn Extended Read Size Registers (LCDC_VSPD_VI6_RPFn_SRC_ESIZE)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0304h (n = 0)
<LCDC_vspd_base> + 0404h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	EHSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	EVSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	EHSIZE[12:0]	0000h	RW	RPF Extended Horizontal Read Size These bits specify the horizontal size of the extended read area to which the RPFn reads data from the external RAM. As shown in **Figure 9.4-7**, the basic read area image is repeatedly placed in the extended read area; in the EHSIZE bits, specify a value not smaller than the horizontal size of the basic read area. When the input format is YCbCr4:2:2 or YCbCr4:2:0, specify the size in 2-pixel units (an even value). A value from 1 to 1920 can be specified.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	EVSIZE[12:0]	0000h	RW	RPF Extended Vertical Read Size These bits specify the vertical size of the extended read area to which the RPFn reads data from the external RAM. As shown in **Figure 9.4-7**, the basic read area image is repeatedly placed in the extended read area; in the EVSIZE bits, specify a value not smaller than the vertical size of the basic read area. When the input format is YCbCr4:2:0, specify the size in 2-pixel units (an even value). A value from 1 to 1920 can be specified.

VI6_RPFn_SRC_ESIZE specifies the extended size for RPFn. The extended horizontal and vertical sizes should be equal to or greater than the basic sizes specified in VI6_RPFn_SRC_BSIZE. The RPF internal data processing described later and image processing described in **9.4.3.2.8 DPR Control Registers** and later sections are all applied to the image in the extended read size shown on the right side in **Figure 9.4-7**.

(3) RPFn Input Format Registers (LCDC_VSPD_VI6_RPFn_INFMT) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0308h (n = 0)
<LCDC_vspd_base> + 0408h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	VIR	-	-	-	-	-	-	-	-	-	-	-	CIPM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPYCS	SPUVS	CEXT[1:0]		RDTM[2:0]			CSC	-	RDFMT[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	VIR	0h	RW	Virtual Input Enable Enables or disables the virtual input function of the RPFn. The image to be processed by the RPFn is usually read from the external memory by the MAU. Instead of this input, the virtual input function generates a single-color image within the RPFn and sends it to the modules in VSPD. When the virtual input function is enabled, the fixed value specified in VI6_RPFn_VRTCOL_SET is used as the input to the RPFn. While the virtual input function is enabled, data is not read from the external memory; that is, the α plane is not read and the IROP calculation thus cannot be executed. In this case, set VI6_RPFn_ALPH_SEL.ASEL to 4. Neither the color space conversion through CSC nor the color keying described in **9.4.3.2.6 (11) RPFn Color Keying Control Registers (VI6_RPFn_CKEY_CTRL)** can be used. 0b: RPFn uses general input. 1b: RPFn uses virtual input.
27 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	CIPM	0h	RW	Horizontal Chrominance Interpolation Method Setting Image data is processed in the YCbCr444 format inside VSPD in case of YCbCr color space. When the chrominance format of the input image is YCbCr422 or YCbCr420, data is up-sampled as shown in **Figure 9.4-8** for internal processing. This bit specifies the method of up-sampling for this purpose. 0b: The nearest-neighbor method is used for horizontal chrominance interpolation. 1b: The bilinear method is used for horizontal chrominance interpolation.
15	SPYCS	0h	RW	RPF Input Mode Setting 1 When the input format is YUY2, set this bit to 1b and set the RDFMT bits to 71 (47h). When the input format is YVYU, set this bit and the SPUVS bit to 1b and set the RDFMT bits to 71 (47h). In other cases, set this bit to 0b.
14	SPUVS	0h	RW	RPF Input Mode Setting 2 When the input format is NV61, set this bit to 1b and set the RDFMT bits to 65 (41h). When the input format is NV21, set this bit to 1b and set the RDFMT bits to 66 (42h). When the input format is YVYU, set this bit and the SPYCS bit to 1b and set the RDFMT bits to 71 (47h). In other cases, set this bit to 0b.
13, 12	CEXT[1:0]	0h	RW	Lower-Bit Color Data Extension Method Setting When an RGB input format where each color component is expressed in less than 8 bits are selected from **Table 9.4-10** through the RDFMT bits, VSPD internally extends each color component to 8 bits before using the data. These bits select this extension method. 00b: Lower-order bits of color data are extended with 0. 01b: Upper-order bits of color data are copied to the lower -order bits. 10b: Lower-order bits of color data are extended with 0. The maximum value is limited to FFh. 11b: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
11 to 9	RDTM[2:0]	0h	RW	CSC Conversion Expression Setting These bits select the expression used for color space conversion. The conversion direction is RGB → YCbCr when RGB is selected through the RDFMT bits; the direction is YCbCr → RGB when YCbCr is selected. 0h: BT.601 YCbCr [16,235/240] ↔ RGB [0,255] 1h: BT.601 YCbCr [0,255] ↔ RGB [0,255] 2h: BT.709 YCbCr [16,235/240] ↔ RGB [0,255] 3h: BT.709 YCbCr [16,235/240] ↔ RGB [16,235] 4h to 7h: Setting prohibited
8	CSC	0h	RW	Color Space Conversion Enable Enables or disables color space conversion between YCbCr and RGB to be executed in RPFn. The characteristics of color space conversion are determined by the RDTM bit setting.*1 When using the virtual input (VIR = 1b), specify 0b. 0b: Color space conversion is disabled. 1b: Color space conversion is enabled.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 0	RDFMT[6:0]	00h	RW	RPF Input Image Format Setting These bits select the format of the image input from the external RAM to the RPFn. Select a value corresponding to the desired format from those shown in **Table 9.4-10**, **Table 9.4-11**, and **Table 9.4-12**. When the virtual input function is used (VIR = 1b), the color information for the virtual input should be specified in VI6_RPFn_VRTCOL_SET. If this color information is in the RGB format, set the RDFMT bits to 19. If the color information is in the YCbCr format, set these bits to 64. [Note] Number of input pixels When YCbCr4:2:2 is selected through the RDFMT bits, the horizontal size of the input image should be specified in 2-pixel units. When YCbCr4:2:0 is selected, the vertical and horizontal sizes should be specified in 2-pixel units. Observe these restrictions when specifying the image size in VI6_RPFn_SRC_BSIZE and VI6_RPFn_SRC_ESIZE.

Note 1. Note on color space settings

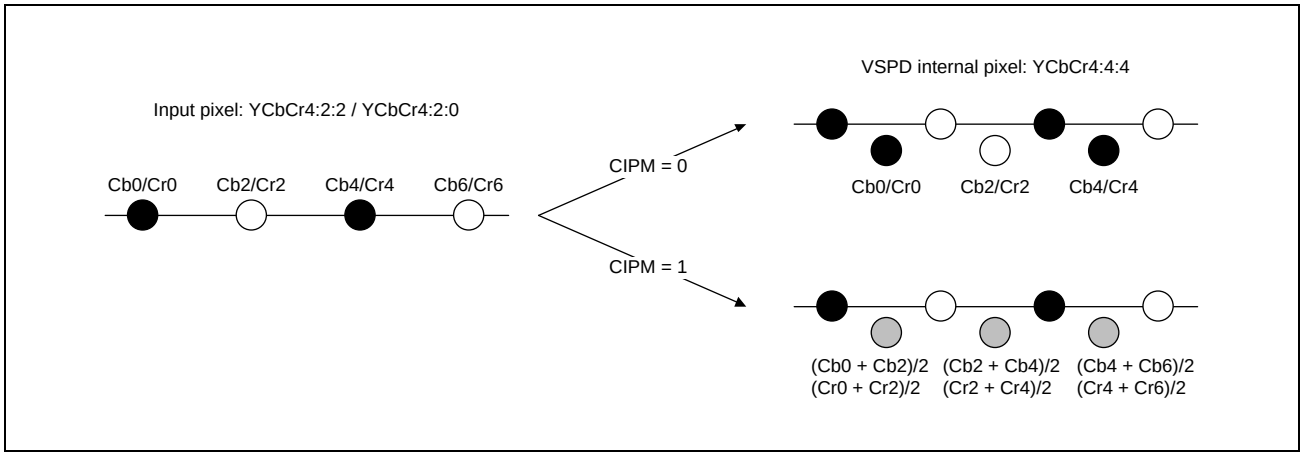


Figure 9.4-8 Chrominance Interpolation Methods Selectable through CIPM Setting

The color space for the image output from the RPF to VSPD internal modules is determined by the combination of the color space for the image input to the RPF, which is selected through the VI6_RPFn_INFMT.RDFMT setting, and the enabled or disabled state of the color space conversion function, which is selected through the VI6_RPFn_INFMT.CSC setting (**Table 9.4-9**). For example, when the image input to the RPF is in the YCbCr format, the RPF outputs data to VSPD internal modules in the YCbCr format if color space conversion is disabled through the CSC bit, and the RPF outputs data in the RGB format if color space conversion is enabled. When the image input to the RPF is in the RGB format, the relationship between the output format and the color space conversion setting is the opposite of the YCbCr case. For some VSPD internal modules, the YCbCr format is recommended for image processing because of the characteristics of the processing, or the same color space needs to be specified between multiple RPF outputs. In these

cases, set VI6_RPFn_INFMT.RDFMT and VI6_RPFn_INFMT.CSC appropriately so that the RPFs can output the required color space according to the color space conditions described above.

Table 9.4-9 RPFn Input Color Space and Output Color Space

RPFn Input Color Space (VI6_RPFn_INFMT. RDFMT)		Color Space Conversion Setting (VI6_RPFn_INFMT.CSC)		RPFn Output Color Space
RGB	(00h to 3Fh)*1	Disabled	(0)*1	RGB
		Enabled	(1)*1	YCbCr
YCbCr	(40h to 7Fh)*1	Disabled	(0)*1	YCbCr
		Enabled	(1)*1	RGB

Note 1. Value specified in the register

A color space conversion function equivalent to that in the RPFn is also provided by the WPF. As shown in **Table 9.4-9**, the color space (YCbCr or RGB) output from the RPF becomes the input format for the WPF. Here, the color space of the output image obtained by the color space conversion function of the WPF must match the color space of the format specified through VI6_WPFn_OUTFMT.WRFMT.

Figure 9.4-9 shows the relationship between the input/output format and color space. The input color space for the RPF is determined when the input image format for the RPF is specified through the RDFMT bits. The color space for the image output from the RPF to subsequent VSPD internal modules depends on the combination of the RPF input format and CSC (color space conversion function) enabled or disabled state in the RPF as shown in **Table 9.4-9** and **Figure 9.4-9**. The user should first determine whether the image processing in the VSPD is done in YCbCr or RGB, and then specify the RPF input format and CSC enabled or disabled state to obtain the desired color space. The color space of the RPF output image is also that of the WPF input image; the color space of the data output from the WPF to the outside of VSPD depends on the combination of the WPF input color space and the enabled or disabled state of the CSC implemented in the WPF as shown in **Figure 9.4-9**. The color space of the WPF output image must match that of the WPF output format (determined by VI6_WPFn_OUTFMT.WRFMT). For example, in the flow shown in **Figure 9.4-9**, YCbCr should not be specified as the WPF output format regardless of the fact that the color space of the WPF output image is in RGB format.

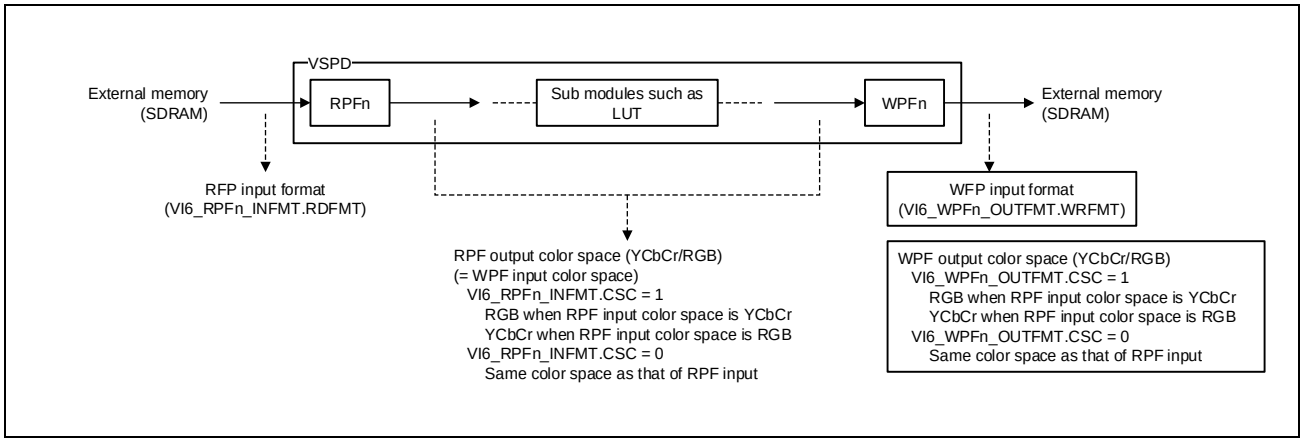


Figure 9.4-9 Relationship between Input/output Format and Color Space

Table 9.4-10 Packed Formats for RPF Input (1/2)

RDFMT[6:0]	Bit per pixel	Phase	upper row - address / bottom row - bit field																													
			n								n+1								n+2								n+3							
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
00h	8	—	R0	R0	R0	G0	G0	G0	B0	B0	R1	R1	R1	G1	G1	G1	B1	B1	R2	R2	R2	G2	G2	G2	B2	B2	R3	R3	R3	G3	G3	G3	B3	B3
01h	12	—					R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0					R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
02h			R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0					R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1				
03h	—	—	Reserved								Reserved								Reserved								Reserved							
04h	15	—		R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0		R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
05h			R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0		R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	
06h	16	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
07h	18	—	A	A	A	A	A	A	A	A							R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
08h			R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0							A	A	A	A	A	A	A	A
09h									R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	A	A	A	A	A	A	A	A
0Ah			A	A	A	A	A	A	A	A	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0						
0Bh			A	A	A	A	A	A	A	A			R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0
0Ch					R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0	A	A	A	A	A	A	A	A
0Dh			A	A	A	A	A	A	A	A	R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0		
0Eh			R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0			A	A	A	A	A	A	A	A
0Fh		0							R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0							R1	R1
		1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1							R2	R2	R2	R2	R2	R2	G2	G2	G2	G2
		2	G2	G2	B2	B2	B2	B2	B2	B2							R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3
10h		0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0							R1	R1	R1	R1	R1	R1	G1	G1
		1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1							R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	B2	B2	B2	B2
		2	B2	B2							R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3						
11h		0			R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0			R1	R1	R1	R1	R1	R1
		1			G1	G1	G1	G1	G1	G1			B1	B1	B1	B1	B1	B1			R2	R2	R2	R2	R2	R2			G2	G2	G2	G2	G2	G2
		2			B2	B2	B2	B2	B2	B2			R3	R3	R3	R3	R3	R3			G3	G3	G3	G3	G3	G3			B3	B3	B3	B3	B3	B3
12h		0	R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0			R1	R1	R1	R1	R1	R1		
		1	G1	G1	G1	G1	G1	G1			B1	B1	B1	B1	B1	B1			R2	R2	R2	R2	R2	R2			G2	G2	G2	G2	G2	G2		
		2	B2	B2	B2	B2	B2	B2			R3	R3	R3	R3	R3	R3			G3	G3	G3	G3	G3	G3			B3	B3	B3	B3	B3	B3		
13h	24	—	A	A	A	A	A	A	A	A	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
14h			R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	A	A	A	A	A	A	A	A
15h		0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
		1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
		2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
16h	18	—								R0	R0	R0	R0	R0	R0	G0	G0	G0								G0	G0	G0	B0	B0	B0	B0	B0	B0
17h																	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
18h	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1
		1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2
		2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3
19h	12	—	A	A	A	A	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	A	A	A	A	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
1Ah		—	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	A	A	A	A	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	A	A	A	A
1Bh	15	—	A	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	A	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
1Ch		—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	A	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	A
1Dh	12	—	A	A	A	A	B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	A	A	A	A	B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1
1Eh		—	B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	A	A	A	A	B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1	A	A	A	A
1Fh	15	—	A	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	A	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1
20h		—	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	A	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	A
21h	18	0			B0	B0	B0	B0	B0	B0			G0	G0	G0	G0	G0	G0			R0	R0	R0	R0	R0	R0			B1	B1	B1	B1	B1	B1
		1			G1	G1	G1	G1	G1	G1			R1	R1	R1	R1	R1	R1			B2	B2	B2	B2	B2	B2			G2	G2	G2	G2	G2	G2
		2			R2	R2	R2	R2	R2	R2			B3	B3	B3	B3	B3	B3			G3	G3	G3	G3	G3	G3			R3	R3	R3	R3	R3	R3
22h	24	—	A	A	A	A	A	A	A	A	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0
23h	16	—																	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0
24h to 2Fh	—	—	Reserved								Reserved								Reserved								Reserved							

Table 9.4-10 Packed Formats for RPF Input (2/2)

RDFMT[6:0]	Bit per pixel	Phase	upper row - address / bottom row - bit field																									
			n								n+1								n+2								n+3							
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
30h	*1	*1	*1								*1								*1								*1							
31h to 3Fh	—	—	Reserved								Reserved								Reserved								Reserved							

Note 1. When the RDFMT[6:0] bits are set to 30h, refer to **Table 9.4-11** for the additional settings and the packed formats

Table 9.4-11 Packed formats of RGB10, RGB10A2, A2RGB10

RDFMT[6:0]	Format name	BYPP_M1[2:0] CPOS[31:0] CLEN[31:0]	upper row - address / bottom row - bit field																									
			n								n+1								n+2								n+3							
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
30h	RGB10	3h 000A1400h 0A0A0A00h	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	B0	B0		
30h	RGB10A2	3h 000A141Eh 0A0A0A02h	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	B0	B0	A0	A0
30h	A2RGB10	3h 020C1600h 0A0A0A02h	A0	A0	R0	R0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	B0	B0

Table 9.4-12 Packed YCbCr Formats for RPF Input

RDFMT[6:0]	Packed YCbCr Input Format	Reference
40h	YCbCr4:4:4 semi-planar	**Figure 9.4-10***4
41h	YCbCr4:2:2 semi-planar (NV16, NV61*1)	
42h	YCbCr4:2:0 semi-planar (NV12, NV21*1)*5	
43h to 45h	Reserved	—
46h	YCbCr4:4:4 interleaved	**Figure 9.4-11***4
47h	YCbCr4:2:2 interleaved type 0 (UYVY, YUY2*2, YVYU*3)	
48h	YCbCr4:2:2 interleaved type 1	
49h	YCbCr4:2:0 interleaved*6	
4Ah	YCbCr4:4:4 planar	**Figure 9.4-12***4
4Bh	YCbCr4:2:2 planar (YV16)	
4Ch	YCbCr4:2:0 planar (YV12, I420)*5	
4Dh to 7Eh	Reserved	—
7Fh	Reserved	—

Note 1. When the input format is NV61 or NV21, set the SPUVS bit to 1.

Note 2. When the input format is YUY2, set the SPYCS bit to 1.

Note 3. When the input format is YVYU, set the SPUVS bit to 1 and SPYCS bit to 1.

Note 4. **Figure 9.4-13** shows the definition of memory address for each pixel in **Figure 9.4-10** to **Figure 9.4-12**.

Note 5. Each line of C plane is read twice, so byte/pixel of YCbCr420 is 2 byte/pixel (same as YCbCr422).

Note 6. Each line of plane is read twice, so byte/pixel of YCbCr420ITL is 3 byte/pixel (same as YCbCr444ITL).

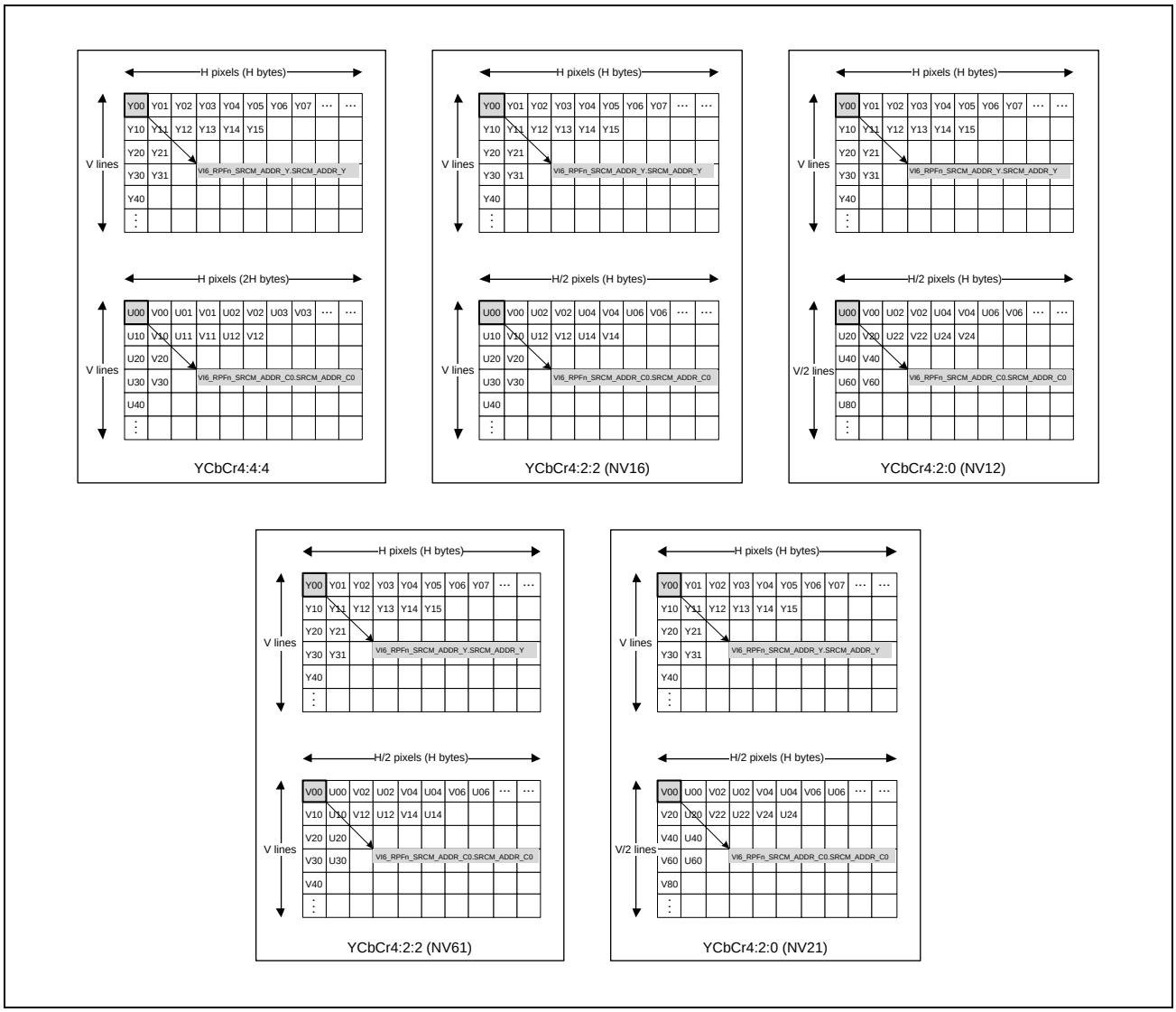


Figure 9.4-10 YCbCr Semi-Planar Formats*1

Note 1. This figure is for 8 bpc.

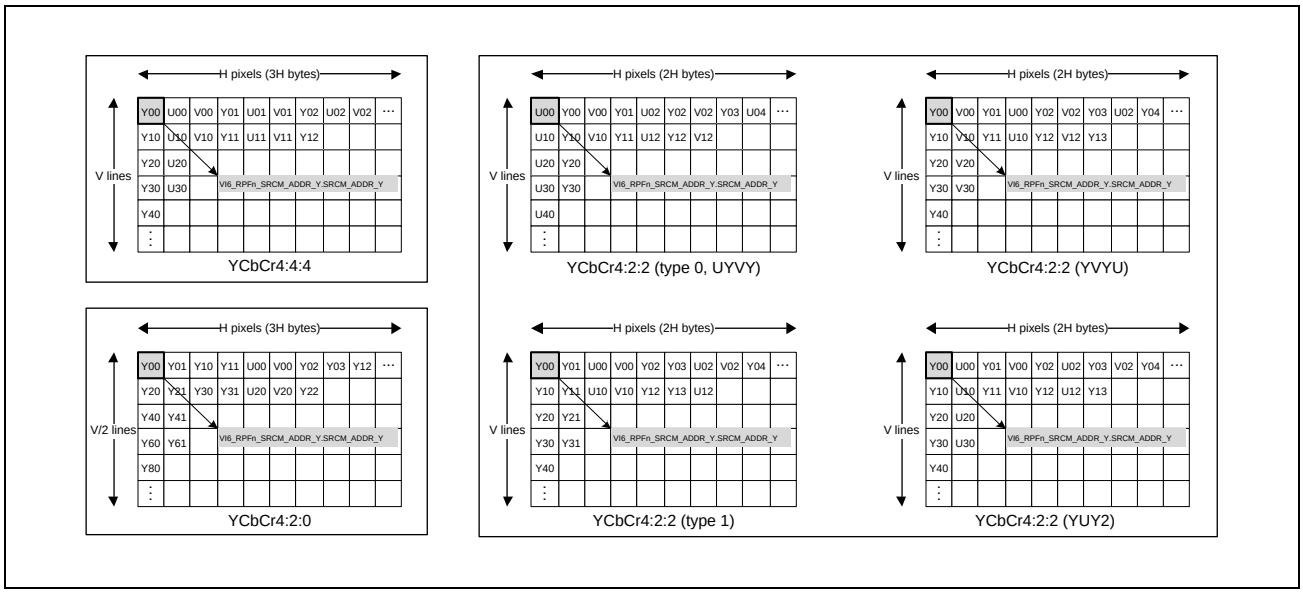


Figure 9.4-11 YCbCr Interleaved Formats*1

Note 1. This figure is for 8 bpc.

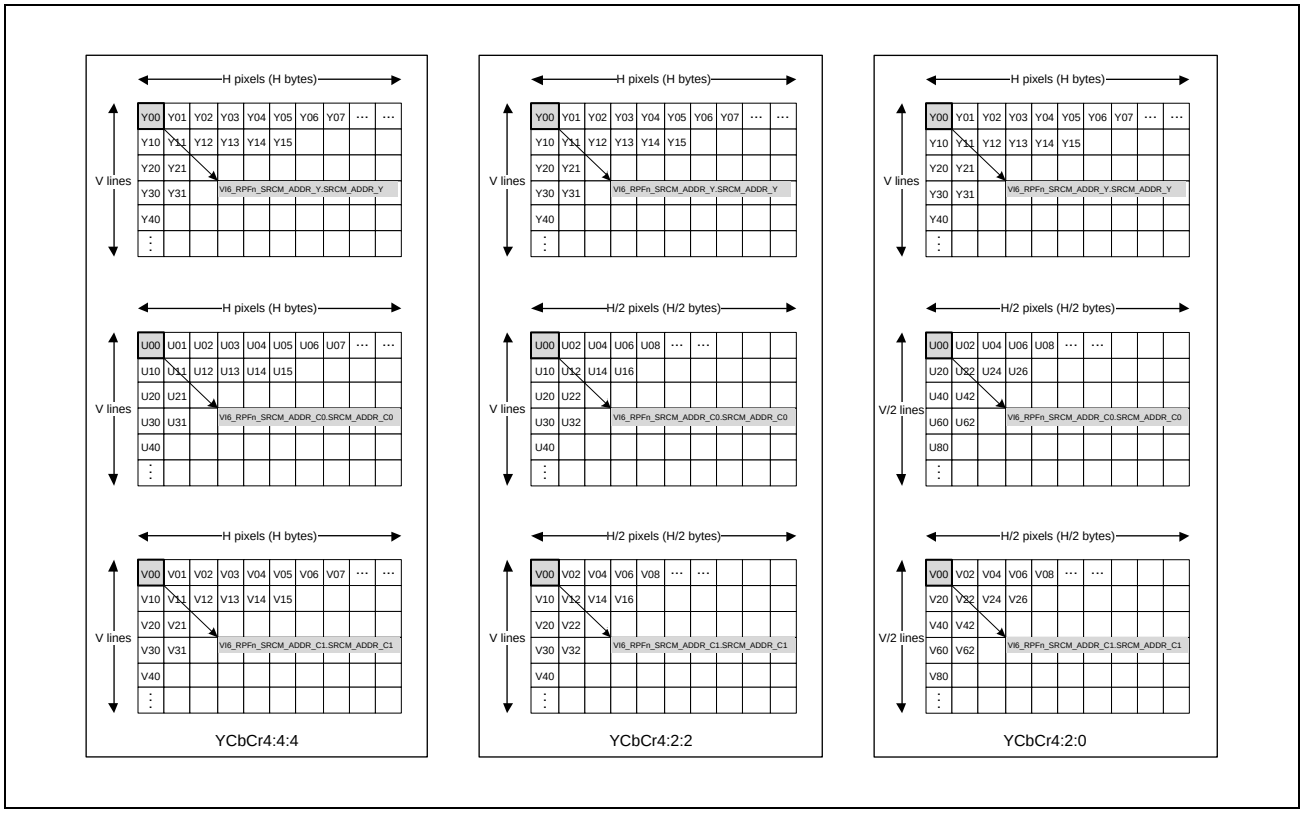


Figure 9.4-12 YCbCr Planar Formats*1

Note 1. This figure is for 8 bpc.

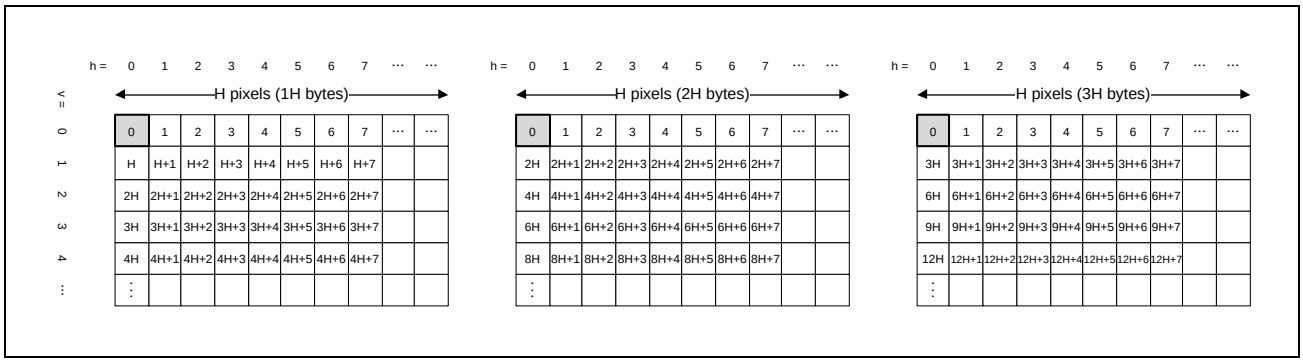


Figure 9.4-13 Memory Address for the Corresponding Pixel's Position

Legend in **Figure 9.4-13**

- H is horizontal image size in pixel unit.
- 1H, 2H, and 3H correspond to the case where 1 pixel has 1 byte, 2 bytes, and 3 bytes of data, respectively.

(4) RPFn Data Swapping Registers (LCDC_VSPD_VI6_RPFn_DSWAP) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 030Ch (n = 0)
<LCDC_vspd_base> + 040Ch (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	A_LLS	A_LWS	A_WDS	A_BTS	-	-	-	-	P_LLS	P_LWS	P_WDS	P_BTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11	A_LLS	0h	RW	α Plane Data Swapping in LONG LWORD Units The effect of this bit setting is defined in **Table 9.4-14**. 0b: Data swapping in LONG LWORD (64-bit) units is disabled 1b: Data swapping in LONG LWORD (64-bit) units is enabled
10	A_LWS	0h	RW	α Plane Data Swapping in long word Units The effect of this bit setting is defined in **Table 9.4-14**. 0b: Data swapping in long word (32-bit) units is disabled 1b: Data swapping in long word (32-bit) units is enabled
9	A_WDS	0h	RW	α Plane Data Swapping in Word Units The effect of this bit setting is defined in **Table 9.4-14**. 0b: Data swapping in Word (16-bit) units is disabled 1b: Data swapping in Word (16-bit) units is enabled
8	A_BTS	0h	RW	α Plane Data Swapping in Byte Units The effect of this bit setting is defined in **Table 9.4-14**. 0b: Data swapping in Byte (8-bit) units is disabled 1b: Data swapping in Byte (8-bit) units is enabled
7 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	P_LLS	0h	RW	Picture Plane Data Swapping in LONG LWORD Units The effect of this bit setting is defined in **Table 9.4-14**. 0b: Data swapping in LONG LWORD (64-bit) units is disabled 1b: Data swapping in LONG LWORD (64-bit) units is enabled This register is available for only Luma plane (Y) in the case input data format is YUV Planar or Semi-Planar and input bit-depth is 2bypc (IPBD_Y != 0 or IPBD_C != 0), and available for both Luma plane (Y) and Chroma plane (CbCr/U/V) in other cases.
2	P_LWS	0h	RW	Picture Plane Data Swapping in long word Units The effect of this bit setting is defined in **Table 9.4-14**. 0b: Data swapping in long word (32-bit) units is disabled 1b: Data swapping in long word (32-bit) units is enabled This register is available for only Luma plane (Y) in the case input data format is YUV Planar or Semi-Planar and input bit-depth is 2bypc (IPBD_Y != 0 or IPBD_C != 0), and available for both Luma plane (Y) and Chroma plane (CbCr/U/V) in other cases.
1	P_WDS	0h	RW	Picture Plane Data Swapping in Word Units The effect of this bit setting is defined in **Table 9.4-14**. 0b: Data swapping in Word (16-bit) units is disabled 1b: Data swapping in Word (16-bit) units is enabled This register is available for only Luma plane (Y) in the case input data format is YUV Planar or Semi-Planar and input bit-depth is 2bypc (IPBD_Y != 0 or IPBD_C != 0), and available for both Luma plane (Y) and Chroma plane (CbCr/U/V) in other cases.

Bit	Bit Name	Initial Value	R/W	Description
0	P_BTS	0h	RW	Picture Plane Data Swapping in Byte Units The effect of this bit setting is defined in **Table 9.4-14**. 0b: Data swapping in Byte (8-bit) units is disabled 1b: Data swapping in Byte (8-bit) units is enabled This register is available for only Luma plane (Y) in the case input data format is YUV Planar or Semi-Planar and input bit-depth is 2bypc (IPBD_Y != 0 or IPBD_C != 0), and available for both Luma plane (Y) and Chroma plane (CbCr/U/V) in other cases.

When the virtual input function of the RPFn is used (VI6_RPFn_INFMT.VIR = 1), this register setting is ignored. Swapping of RPF input data can be specified separately for the α plane and picture plane.

Table 9.4-13 shows the data order before and after swapping according to the long long word, long word, word, and byte swapping settings.

When data order in memory for each format is the same as **Table 9.4-10** for RGB format and **Figure 9.4-10** to **Figure 9.4-13** for YCbCr format, set 1111b to {*_LLS, *_LWS, *_WDS, *_BTS}. If data order is not the same as the definition, change data order within 16byte unit by these bits as shown in **Table 9.4-13**.

Table 9.4-13 Changing Data Order According to Swap Register

Data order in memory																				
Byte address	16n+0	16n+1	16n+2	16n+3	16n+4	16n+5	16n+6	16n+7	16n+8	16n+9	16n+10	16n+11	16n+12	16n+13	16n+14	16n+15	*_LLS	*_LWS	*_WDS	*_BTS
Data	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	1	1	1	1
	1	0	3	2	5	4	7	6	9	8	11	10	13	12	15	14	1	1	1	0
	2	3	0	1	6	7	4	5	10	11	8	9	14	15	12	13	1	1	0	1
	3	2	1	0	7	6	5	4	11	10	9	8	15	14	13	12	1	1	0	0
	4	5	6	7	0	1	2	3	12	13	14	15	8	9	10	11	1	0	1	1
	5	4	7	6	1	0	3	2	13	12	15	14	9	8	11	10	1	0	1	0
	6	7	4	5	2	3	0	1	14	15	12	13	10	11	8	9	1	0	0	1
	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	1	0	0	0
	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	0	1	1	1
	9	8	11	10	13	12	15	14	1	0	3	2	5	4	7	6	0	1	1	0
	10	11	8	9	14	15	12	13	2	3	0	1	6	7	4	5	0	1	0	1
	11	10	9	8	15	14	13	12	3	2	1	0	7	6	5	4	0	1	0	0
	12	13	14	15	8	9	10	11	4	5	6	7	0	1	2	3	0	0	1	1
	13	12	15	14	9	8	11	10	5	4	7	6	1	0	3	2	0	0	1	0
	14	15	12	13	10	11	8	9	6	7	4	5	2	3	0	1	0	0	0	1
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	0	0	0	0

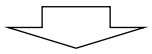


Table 9.4-10 for RGB format and **Figure 9.4-10** to **Figure 9.4-13** for YCbCr format																
Byte address	16n+0	16n+1	16n+2	16n+3	16n+4	16n+5	16n+6	16n+7	16n+8	16n+9	16n+10	16n+11	16n+12	16n+13	16n+14	16n+15
Data	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

(5) RPFn Display Location Registers (LCDC_VSPD_VI6_RPFn_LOC) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0310h (n = 0)
<LCDC_vspd_base> + 0410h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	HCOORD[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VCOORD[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	HCOORD[12:0]	0000h	RW	Horizontal Coordinate of Sublayer Display Location on Master Layer These bits specify the left-end location of the sublayer displayed by the RPFn and the subsequent module connected through the DPR. Specify the horizontal coordinate of the location in pixel units with the left-end pixel of the master layer set at coordinate 0. When the RPFn is the master layer, set these bits to 0b. If the sublayer extends beyond the master layer according to the HCOORD setting, the extended section is cut off at the right end of the master layer. Even in this case, however, a bus transfer that is unnecessary for output image generation is executed since the whole sublayer data is read from the external memory. Appropriate coordinate setting is required so that the sublayer does not extend beyond the right end of the master layer. A value from 0 to 1919 can be specified.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	VCOORD[12:0]	0000h	RW	Vertical Coordinate of Sublayer Display Location on Master Layer These bits specify the top-end location of the sublayer displayed by the RPFn and the subsequent module connected through the DPR. Specify the vertical coordinate of the location in pixel units with the top-end pixel of the master layer set at coordinate 0. When the RPFn is the master layer, set these bits to 0b. If the sublayer extends beyond the master layer according to the VCOORD setting, the extended section is cut off at the bottom end of the master layer. Even in this case, however, a bus transfer that is unnecessary for output image generation is executed since the whole sublayer data is read from the external memory. Appropriate coordinate setting is required so that the sublayer does not extend beyond the bottom end of the master layer. A value from 0 to 1919 can be specified.

Figure 9.4-14 shows an example of RPF1 offsets with respect to master layer RPF0. Although this figure only shows sublayers RPF1, specify offsets for all RPFs other than the master layer in the same way as shown in this example.

Whether an RPFn is the master layer or a sublayer is determined through the selection of the source RPF for WPFn (the VI6_WPFn_SRCRPF setting). For details, refer to **(1) WPF0-Source-RPF Register (LCDC_VSPD_VI6_WPF0_SRCRPF)**.

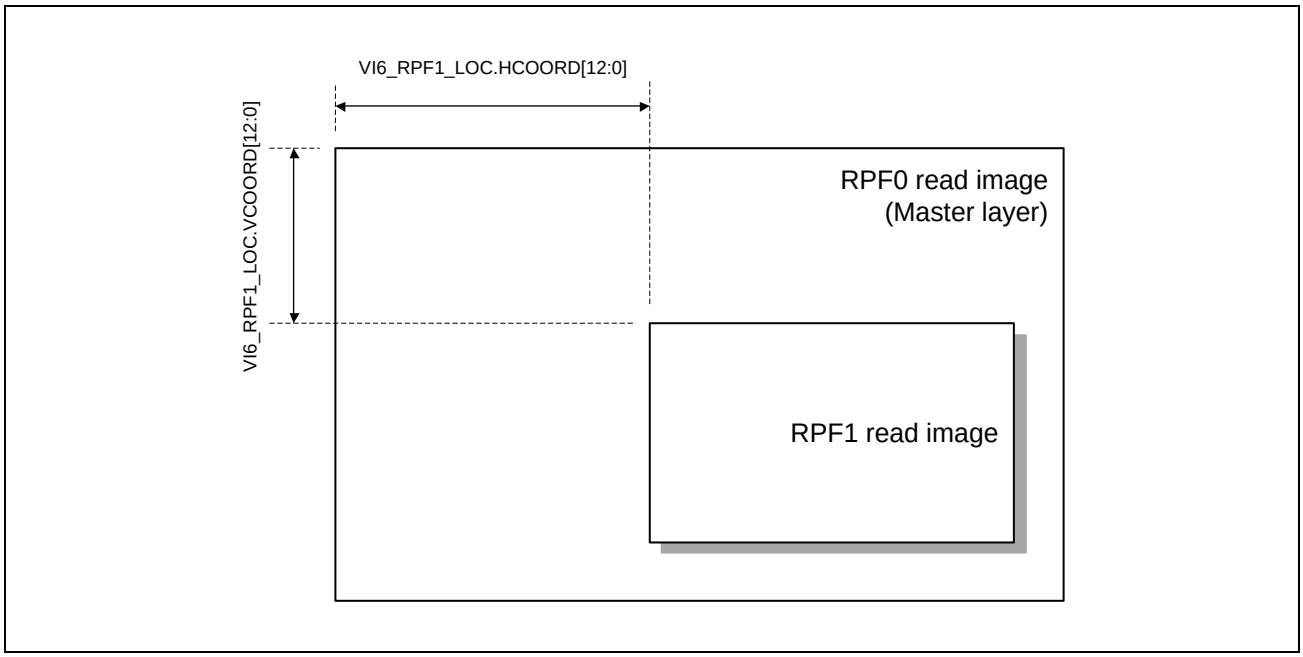


Figure 9.4-14 RPF1 Offsets from Master Layer

(6) RPFn α Plane Selection Control Registers (LCDC_VSPD_VI6_RPFn_ALPH_SEL) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0314h (n = 0)
<LCDC_vspd_base> + 0414h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	ASEL[2:0]			IROP[3:0]				BSEL	-	-	-	AEXT[1:0]		-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	RW	RW	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALPHA1[7:0]								ALPHA0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 28	ASEL[2:0]	0h	RW	α Format and Processing Method Select These bits select how to handle the α value to be used. The RPF handles two types of α value; 8-bit and 1-bit values. When a 1-bit α value is used, VSPD assumes that the 1-bpp α value for each pixel is stored in the order from MSB to LSB in each byte (big endian). The α value is used as either transparency information or mask information. Transparency information is included in the α plane read from external memory when the ASEL bits are set to 1 or 3 and in the α value stored in the packed RGB bit field when these bits are set to 0 or 2. The α value as transparency information is sent as the destination value to the IROP as shown in **Figure 9.4-15** and then output to the subsequent modules. The output α value is used, for example, for blending in the BRS. The α value as mask information is used for IROP operation in the RPF. The mask information is included in the α plane read from external RAM when the ASEL bits are set to 0 or 2 and the source value is used in IROP operation (IROP setting other than 0, 5, 10, or 15). This α value is sent as the source value to the IROP as shown in **Figure 9.4-15**. Note that the α value selected through the ASEL bits has a lower priority than the VI6_RPFn_CKEY_SET*.AP* value replaced through the color keying function. When the color keying function is used, the α value may be replaced with the VI6_RPFn_CKEY_SET*.AP* value regardless of the ASEL bit setting. When using the virtual input function (VI6_RPFn_INFMT.VIR = 1b), specify 4. 0h: 1, 4, or 8-bit packed α + plane α (IROP ! = 0, 5, 10, 15) The α bit field in 1, 4, or 8-bit packed α is handled as transparency information. Be sure to specify the packed format that includes α through VI6_RPFn_INFMT.RDFMT. When VI6_RPFn_MSKCTRL.MSK_EN is 0b and the IROP bit value is not 0, 5, 10, or 15, the α plane should be read as mask information. Specify the number of α data bits (BSEL) stored in the α plane and the α plane read start address (VI6_RPFn_SRCM_ADDR_AI). When the IROP bits are set to 0, 5, 10, or 15, the α plane is not read. 1h: 8-bit plane α The 8-bit α plane is read from external RAM as transparency information. When the packed RGB format has a bit field for α, the information in the α bit field is discarded. The α plane read start address (VI6_RPFn_SRCM_ADDR_AI) should be specified. The α value goes through the 8-bit transparent α generator shown in **Figure 9.4-15** without change. When VI6_RPFn_MSKCTRL.MSK_EN is 0b, IROP operation cannot be executed; set the IROP bits to 0 in this case. When VI6_RPFn_MSKCTRL.MSK_EN is 1b. IROP operation can be executed.

Bit	Bit Name	Initial Value	R/W	Description
				2h: 1-bit packed α + plane α (IROP ! = 0, 5, 10, 15) The 1-bit packed α input is converted by the 8-bit transparent α generator shown in **Figure 9.4-15** according to the ALPHA0/1 setting into the 8-bit α value as transparency information. Select the packed input format that includes a 1-bit α field. When VI6_RPFn_MSKCTRL.MSK_EN is 0b and the IROP value is not 0, 5, 10, or 15, the α plane should be read as mask information. Specify the number of α data bits (BSEL) stored in the α plane and the α plane read start address (VI6_RPFn_SRCM_ADDR_AI). When the IROP bits are set to 0, 5, 10, or 15, the α plane is not read. 3h: 1-bit plane α The 1-bit α plane is read from external RAM and converted by the 8-bit transparent α generator shown in **Figure 9.4-15**. according to the ALPHA0/1 setting into the 8-bit α value as transparency information. When the packed RGB format has a bit field for α, the information in the α bit field is discarded. The α plane read start address (VI6_RPFn_SRCM_ADDR_AI) should be specified. When VI6_RPFn_MSKCTRL.MSK_EN is 0b, IROP operation cannot be executed; set the IROP bits to 0b in this case. When VI6_RPFn_MSKCTRL.MSK_EN is 1b. IROP operation can be executed. 4h: Fixed α The fixed α value (VI6_RPFn_VRTCOL_SET.LAYA value) is output from the RPF. IROP operation cannot be executed; set the IROP bits to 0 in this case. 5 to 7: Setting prohibited.
27 to 24	IROP[3:0]	0h	RW	IROP Operation Setting These bits specify the operator to be executed in the IROP operation unit shown in **Figure 9.4-15**. The source (S) for the IROP operation is the pixel data and α data specified in the VI6_RPFn_MSKSET0 or VI6_RPFn_MSKSET1 IROP input value register, which is selected according to the value (0 or 1) generated by the 1-bit mask generator. The destination (D) is the image data (RGB/YCbCr) and 8-bit α data output from the unpack/OSD processor. IROP operation is applied both for the image data and α data between the source and destination data. If these bits are set to the operation that involves the source (S) (IROP setting other than 0, 5, 10, or 15) while VI6_RPFn_MSKCTRL.MSK_EN is 0b, the α plane is read from the external RAM to be used for the α value for IROP operation; specify the α plane read start address (VI6_RPFn_SRCM_ADDR_A). When the virtual input function is used (VI6_RPFn_INFMT.VIR = 1b), IROP operation is not available; set these bits to 0000b. 0000b: NOP(D) 0001b: AND(S & D) 0010b: AND_REVERSE(S & ~D) 0011b: COPY(S) 0100b: AND_INVERTED(~S & D) 0101b: CLEAR(0) 0110b: XOR(S ^ D) 0111b: OR(S | D) 1000b: NOR(~(S | D)) 1001b: EQUIV(~(S ^ D)) 1010b: INVERT(~D) 1011b: OR_REVERSE(S | ~D) 1100b: COPY_INVERTED(~S) 1101b: OR_INVERTED(~S | D) 1110b: NAND(~(S & D)) 1111b: SET(all 1)
23	BSEL	0h	RW	α Bit Count Conversion Selection for 1-Bit Mask Generator Specifies the number of bits in the α plane to be read as mask information from the external RAM. The α value in mask information is used for the source (S) in IROP. When α plane data is 8 bits, it is converted to one bit through the 1-bit mask generator shown in **Figure 9.4-15**. Note that this bit setting is valid when the ASEL bits are set to 0 or 2 and VI6_RPFn_MSKCTRL.MSK_EN is set to 0b. In other cases, this bit setting has no effect. 0b: 8-bit α is converted to 1-bit α through the 1-bit mask generator. When the 8-bit α value input to the RPF is not 0, it is converted to 1b; when the value is 0, it is converted to 0b. 1b: α value goes through the 1-bit mask generator. The 1-bit α value input to the RPF is output through the 1-bit mask generator without change.
22 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
19, 18	AEXT[1:0]	0h	RW	Lower-Bit α Value Extension Method Set These bits specify the method for extending the input α data to 8 bits through the unpack processing. 00b: The lower-order bits of α value are extended with 0. 01b: The upper-order bits of α value are copied to the lower-order bits. 10b: The lower-order bits of α value are extended with 0. The maximum value is limited to FFh. 11b: Setting prohibited
17, 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 8	ALPHA1[7:0]	00h	RW	8-Bit α Value Output when 1-Bit α Value is 1 These bits specify the 8-bit α value to be output when 1-bit α data is input and the α value input to the 8-bit transparent α generator shown in **Figure 9.4-15** is 1b. This setting is valid when the ASEL bits are set to 010b or 011b. A value from 0 to 255 can be specified.
7 to 0	ALPHA0[7:0]	00h	RW	8-Bit α Value Output when 1-Bit α Value is 0 These bits specify the 8-bit α value to be output when 1-bit α data is input and the α value input to the 8-bit transparent α generator shown in **Figure 9.4-15** is 0b. This setting is valid when the ASEL bits are set to 010b or 011b. A value from 0 to 255 can be specified.

Figure 9.4-15 shows the relationship between the α selector, IROP operation unit, color keying unit, and related registers. The IROP operation unit receives two inputs, source and destination. The image data input from the external memory is processed through the unpack processor and 8-bit transparent α generator and then input to the IROP operation unit as destination data. The α plane data input from the external memory is sent to the 8-bit transparent α generator when the ASEL bits are set to 1 or 3, or sent to the 1-bit mask α generator when the ASEL bits are set to 0 or 2. For the pixel data and 8-bit α value on the source side of the IROP operation unit, either the VI6_RPFn_MSKSET0 value or VI6_RPFn_MSKSET1 values will be selected according to the 1-bit α value output by the 1-bit mask α generator.

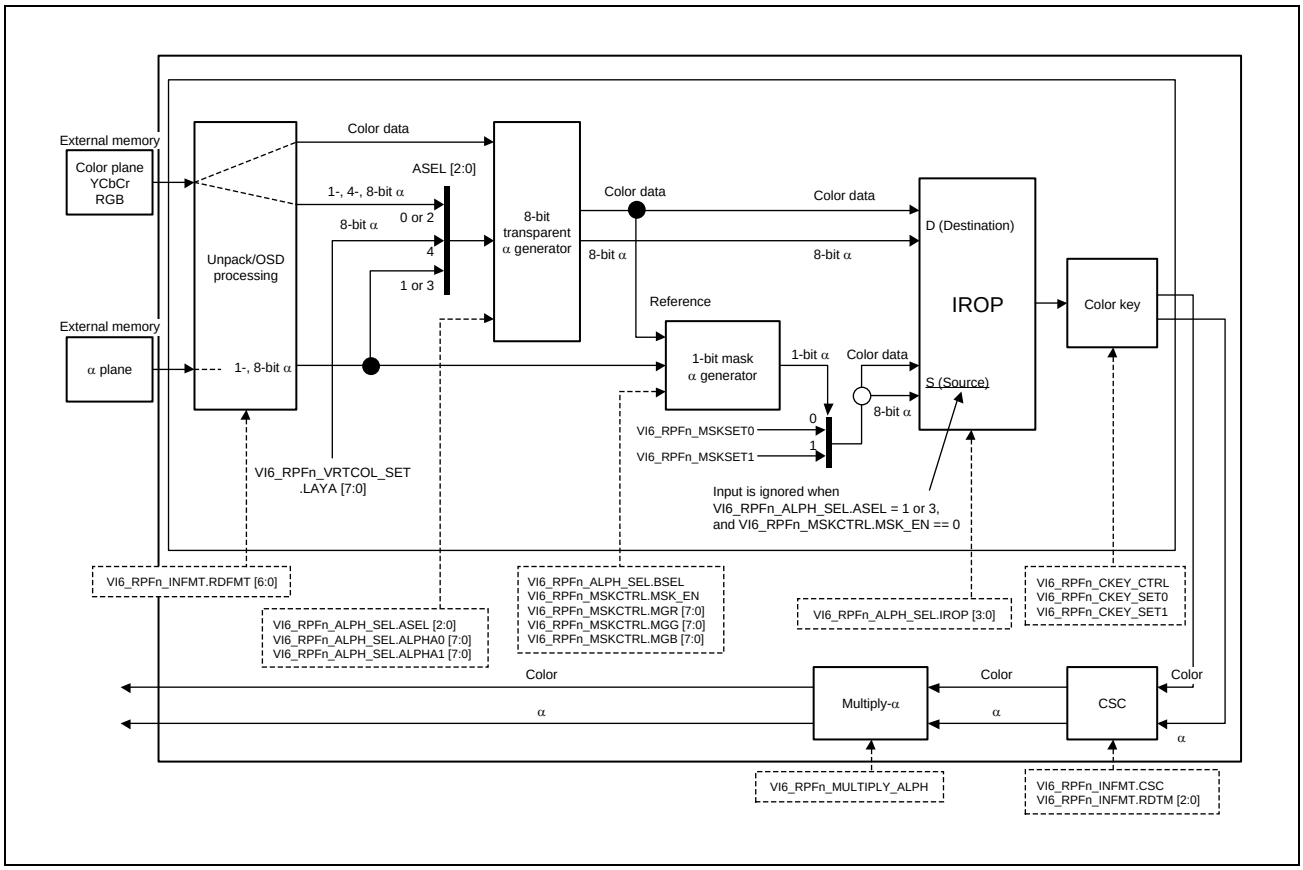


Figure 9.4-15 Configuration of alpha Selector and IROP Operation Unit in RPF

The following describes the function of each block shown in **Figure 9.4-15**. Read the following while referring to the figure as necessary.

Unpack/OSD processor:

Unpacks each component and α value of the image data according to the packed format specified in VI6_RPFn_INFMT.RDFMT.

8-bit transparent α generator:

Converts the input α value into 8-bit α when the input α is four bits or one bit.

When VI6_RPFn_ALPH_SEL.ASEL are set to 0 (8-, 4-, or 1-bit packed α), this generator outputs the input α value without change if the α bit field in the packed α data is 8 bits; if the α bit field is less than 8 bits, it is converted to an 8-bit α value by extending the LSB side according to the VI6_RPFn_ALPH_SEL.AEXT setting.

When VI6_RPFn_ALPH_SEL.ASEL are set to 1 (8-bit plane α) or 4 (fixed α), this generator outputs the input 8-bit plane α without change. If a packed α value is included in RGB data, it is discarded.

When VI6_RPFn_ALPH_SEL.ASEL are set to 2 (1-bit packed α) or 3 (8-bit α generated from 1-bit plane α), an 8-bit α value is generated by using the VI6_RPFn_ALPH_SEL.ALPHA0 [7:0] value when the input 1-bit α value is 0 or by using the VI6_RPFn_ALPH_SEL.ALPHA1 [7:0] value when the input 1-bit α value is 1. When VI6_RPFn_ALPH_SEL.ASEL is set to 3, a packed α value that is included in RGB data is discarded.

1-bit mask α generator:

Generates 1-bit α data from the input 8-bit α data or pixel data. When the input α data is one bit, this generator outputs it without change.

When VI6_RPFn_ALPH_SEL.ASEL are set to 0 (8-, 4-, or 1-bit packed α) or 2 (plane α) and VI6_RPFn_MSKCTRL.MSK_EN is set to 0, the α plane read from the external memory to be used in IROP is converted to 1-bit α data when necessary. When the α plane data read from the external RAM is 8 bits (BSEL = 0), if the value is 0, a 1-bit α value of 0b is generated; if the value is not 0, a 1-bit α value of 1b is generated. When the α plane data is one bit (BSEL = 1), this generator outputs it without change.

When the value of the 1-bit α generated by the 1-bit mask α generator is 0b, the 8-bit α and pixel data specified in VI6_RPFn_MSKSET0 are output as the source. When the generated 1-bit α value is 1b, the 8-bit α and pixel data specified in VI6_RPFn_MSKSET1 are output as the source.

As shown in **Figure 9.4-15**, when VI6_RPFn_ALPH_SEL.ASEL are set to 1 (8-bit plane α) or 3 (1-bit plane α), the α plane read from the external RAM is sent to the 8-bit transparent α generator as transparency information. When VI6_RPFn_MSKCTRL.MSK_EN is set to 0, the 1-bit α for masking is generated according to the input α plane (refer to **(8) RPFn Mask Control Registers (LCDC_VSPD_VI6_RPFn_MSKCTRL)**), but the 1-bit mask α generator does not refer to the input α plane because it is input to the 8-bit transparent α generator as transparency information. Accordingly, the 1-bit mask α generator does not generate a 1-bit α for masking and the data on the source side becomes invalid; that is, IROP operation cannot be executed. Set the IROP bits to 0 in this case. In contrast, when VI6_RPFn_MSKCTRL.MSK_EN is set to 1, the 1-bit mask α generator creates α data for masking according to the pixel data instead of the input α plane data, and IROP operation can be executed in this case.

IROP operation unit:

Executes ROP operation according to the opcode specified in VI6_RPFn_ALPH_SEL.IROP. For ROP operation (other than NOP), valid values should be input both for the source and destination. As described in the above (description of the 1-bit mask α generator), when VI6_RPFn_ALPH_SEL.ASEL are set to 1 or 3 and VI6_RPFn_MSKCTRL.MSK_EN is set to 0, the source data for the IROP operation unit is treated as invalid; set VI6_RPFn_ALPH_SEL.IROP to 0 (NOP). When VI6_RPFn_ALPH_SEL.ASEL are set to 4, a fixed α value is output from the RPF and IROP operation is not available. In the same way as the above case, set VI6_RPFn_ALPH_SEL.IROP to 0 (NOP).

To specify a valid source value for the IROP operation unit and execute IROP operation (specify an opcode other than NOP in the IROP bits), specify register values as shown in **Table 9.4-14**. Where the source input state is indicated as “Valid” in the table, IROP operation can be executed. In the cases where IROP operation is not available, set the IROP bits to 0 (NOP).

Table 9.4-14 Source Input State in IROP Operation Unit

VI6_RPFn_ALPH_SEL.ASEL[2:0]		VI6_RPFn_MSKCTRL.MSK_EN		
		0 (Source data is generated according to input α plane)		1 (Source data is generated according to the destination-side pixel data)
000b	(1-, 4-, or 8-bit packed α + plane α)	Valid	(α plane input)	Valid
001b	(8-bit α plane)	Invalid	(IROP operation is not available; α plane is output to the subsequent modules behind RPF)	Valid
010b	(8-bit α generated from 1-bit packed α + plane α)	Valid	(α plane input)	Valid
011b	(8-bit α generated from 1-bit plane α)	Invalid	(IROP operation is not available; α plane is output to the subsequent modules behind RPF)	Valid
100b	(Fixed α)	Invalid (IROP operation is not available; fixed α is output to the subsequent modules behind RPF)		

For the handling of the α values shown in **Figure 9.4-15** and **Table 9.4-14**, the relationship between the RPF input format and RPF output α value is shown in **Table 9.4-15**. Where only bit names are shown in the table, the bits are in VI6_RPFn_ALPH_SEL described in this section.

Table 9.4-15 α Value Selected and Output according to ASEL Bits in Each Input Format

ASEL Setting		α Value Output for Each Input Format	
		RGB	YCbCr
000b	(8-, 4-, or 1-bit packed α is input)	1-, 4-, or 8-bit pixel α	FFh*1
001b	(8-bit plane α is input)	8-bit α plane	8-bit α plane
010b	(8-bit α is generated from the 1-bit packed α input)	ALPHA0 or ALPHA1 setting	FFh*1
011b	(8-bit α is generated from the 1-bit plane α input)	ALPHA0 or ALPHA1 setting	ALPHA0 or ALPHA1 setting
100b	(Fixed α is output)	VI6_RPFn_VRTCOL_SET.LAYA setting	

Note 1. Fixed value FFh is output because packed α is not included in YCbCr.

(7) RPFn Virtual Plane Color Information Registers (LCDC_VSPD_VI6_RPFn_VRTCOL_SET) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0318h (n = 0)
<LCDC_vspd_base> + 0418h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LAYA[7:0]								LAYR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LAYG[7:0]								LAYB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	LAYA[7:0]	00h	RW	Virtual-Input Fixed α Value These bits specify the fixed α value generated in the RPFn when the virtual input function is enabled through the VI6_RPFn_INFMT.VIR setting. When the virtual input function is disabled (VI6_RPFn_INFMT.VIR = 0b), these bits are used to specify the fixed α value to be output from the RPF while VI6_RPFn_ALPH_SEL.ASEL are set to 4. A value from 0 to 255 can be specified.
23 to 16	LAYR[7:0]	00h	RW	Virtual-Input Fixed R/Cr Component Value These bits specify the fixed R or Cr value generated in the RPFn when the virtual input function is enabled through the VI6_RPFn_INFMT.VIR setting. When VI6_RPFn_INFMT.RDFMT is set to 19, the value specified in these bits is used as the R value. When VI6_RPFn_INFMT.RDFMT is set to 64, the value specified in these bits is used as the Cr value. A value from 0 to 255 can be specified.
15 to 8	LAYG[7:0]	00h	RW	Virtual-Input Fixed G/Y Component Value These bits specify the fixed G or Y value generated in the RPFn when the virtual input function is enabled through the VI6_RPFn_INFMT.VIR setting. When VI6_RPFn_INFMT.RDFMT is set to 19, the value specified in these bits is used as the G value. When VI6_RPFn_INFMT.RDFMT is set to 64, the value specified in these bits is used as the Y value. A value from 0 to 255 can be specified.
7 to 0	LAYB[7:0]	00h	RW	Virtual-Input Fixed B/Cb Component Value These bits specify the fixed B or Cb value generated in the RPFn when the virtual input function is enabled through the VI6_RPFn_INFMT.VIR setting. When VI6_RPFn_INFMT.RDFMT is set to 19, the value specified in these bits is used as the B value. When VI6_RPFn_INFMT.RDFMT is set to 64, the value specified in these bits is used as the Cb value. A value from 0 to 255 can be specified.

(8) RPFn Mask Control Registers (LCDC_VSPD_VI6_RPFn_MSKCTRL) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 031Ch (n = 0)
<LCDC_vspd_base> + 041Ch (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	MSK_EN	MGR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MGG[7:0]								MGB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
24	MSK_EN	0h	RW	Mask Generation Specification Specifies the method of α value generation in the 1-bit mask α generator shown in **Figure 9.4-15**. 0b: A 1-bit mask value is generated according to the input α plane value. When the input α is in the 1-bit format (VI6_RPFn_ALPH_SEL.BSEL = 1b), the 1-bit mask value is output without change. When the input α is in the 8-bit format (VI6_RPFn_ALPH_SEL.BSEL = 0b), the 1-bit mask value is 0 if the α value is 00h; otherwise, the 1-bit mask value is 1. 1b: The R/Cr, G/Y, and B/Cb components of the image input to the destination side of the IROP operation unit are compared with the values specified in the MGR, MGG, and MGB bits, respectively. When all values match, 1 is output as the 1-bit mask value, and in other cases, 0 is output. When the generated 1-bit mask data is not used, set VI6_RPFn_ALPH_SEL.IROP to 0b.
23 to 16	MGR[7:0]	00h	RW	R/Cr Comparison Value for 1-Bit α Generation These bits specify the R/Cr value to be compared for 1-bit α generation by using the pixel data on the destination side. When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify an R value for comparison. When YCbCr is specified, specify a Cr value for comparison. This setting is ignored when the MSK_EN bit is set 0b. A value from 0 to 255 can be specified.
15 to 8	MGG[7:0]	00h	RW	G/Y Comparison Value for 1-Bit α Generation These bits specify the G/Y value to be compared for 1-bit α generation by using the pixel data on the destination side. When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a G value for comparison. When YCbCr is specified, specify a Y value for comparison. This setting is ignored when MSK_EN is set to 0b. A value from 0 to 255 can be specified.
7 to 0	MGB[7:0]	00h	RW	B/Cb Comparison Value for 1-Bit α Generation These bits specify the B/Cb value to be compared for 1-bit α generation by using the pixel data on the destination side. When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a B value for comparison. When YCbCr is specified, specify a Cb value for comparison. This setting is ignored when MSK_EN is set to 0b. A value from 0 to 255 can be specified.

(9) RPFn IROP-SRC Input Value Registers 0 (LCDC_VSPD_VI6_RPFn_MSKSET0) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0320h (n = 0)
<LCDC_vspd_base> + 0420h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSA0[7:0]								MSR0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSG0[7:0]								MSB0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	MSA0[7:0]	00h	RW	IROP-Source Input α Value when 1-Bit α is 0 These bits specify the 8-bit α value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (**Figure 9.4-15**). A value from 0 to 255 can be specified.
23 to 16	MSR0[7:0]	00h	RW	IROP-Source Input R/Cr Value when 1-Bit α is 0 These bits specify the R/Cr value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (**Figure 9.4-15**). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify an R component value. When YCbCr is specified, specify a Cr component value. A value from 0 to 255 can be specified.
15 to 8	MSG0[7:0]	00h	RW	IROP-Source Input G/Y Value when 1-Bit α is 0 These bits specify the G/Y value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (**Figure 9.4-15**). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a G component value. When YCbCr is specified, specify a Y component value. A value from 0 to 255 can be specified.
7 to 0	MSB0[7:0]	00h	RW	IROP-Source Input B/Cb Value when 1-Bit α is 0 These bits specify the B/Cb value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (**Figure 9.4-15**). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a B component value. When YCbCr is specified, specify a Cb component value. A value from 0 to 255 can be specified.

(10) RPFn IROP-SRC Input Value Registers 1 (LCDC_VSPD_VI6_RPFn_MSKSET1) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0324h (n = 0)
<LCDC_vspd_base> + 0424h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSA1[7:0]								MSR1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSG1[7:0]								MSB1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	MSA1[7:0]	00h	RW	IROP-Source Input α Value when 1-Bit α is 1 These bits specify the 8-bit α value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1. A value from 0 to 255 can be specified **(Figure 9.4-15)**.
23 to 16	MSR1[7:0]	00h	RW	IROP-Source Input R/Cr Value when 1-Bit α is 1 These bits specify the R/Cr value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1 (**Figure 9.4-15**). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify an R component value. When YCbCr is specified, specify a Cr component value. A value from 0 to 255 can be specified.
15 to 8	MSG1[7:0]	00h	RW	IROP-Source Input G/Y Value when 1-Bit α is 1 These bits specify the G/Y value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1 (**Figure 9.4-15**). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a G component value. When YCbCr is specified, specify a Y component value. A value from 0 to 255 can be specified.
7 to 0	MSB1[7:0]	00h	RW	IROP-Source Input B/Cb Value when 1-Bit α is 1 These bits specify the B/Cb value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1 (**Figure 9.4-15**). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a B component value. When YCbCr is specified, specify a Cb component value. A value from 0 to 255 can be specified.

(11) RPFn Color Keying Control Registers (LCDC_VSPD_VI6_RPFn_CKEY_CTRL) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0328h (n = 0)
<LCDC_vspd_base> + 0428h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	LTH	-	-	-	CV	-	-	SAPE1	SAPE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	LTH	0h	RW	Transparent color-luma Threshold Mode Enable/Disable This bit enables or disables transparent color-luma threshold mode for the color keying module. In transparent color-luma threshold mode, color information 0 (GY0) specified in VI6_RPFn_CKEY_SET0 register is compared with the input luma values. When the input data is in YCbCr format, and if input Y value is equal to or smaller than specified color information (VI6_RPFn_CKEY_SET0.GY0), the input α value is replaced with the value specified in VI6_RPFn_CKEY_SET0.AP0. This bit is available only when the input data is in YCbCr format. When the input data is in an RGB format, set this bit to 0b. This bit setting is valid only when the CV bit is set to 0b; it is ignored when the CV bit is set to 1b. When using the virtual input function (VI6_RPFn_INFMT.VIR = 1b), set this bit to 0b. 0b: Luma threshold mode is disabled (Matched color mode) 1b: Luma threshold mode is enabled
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	CV	0h	RW	Color Replacement Control This bit controls the color replacement function in the color keying module shown in **Figure 9.4-3**. When an RGB format is specified as the color space of the RPFn input data through VI6_RPFn_INFMT.RDFMT, and if all components of an input pixel match the color components specified in VI6_RPFn_CKEY_SET0, the color replacement function replaces the values of the input α and all RGB components with the α and color components specified in VI6_RPFn_CKEY_SET1. When a YCbCr format is specified as the color space of the RPFn input data through VI6_RPFn_INFMT.RDFMT, only the Y data is compared; if the luminance component of an input pixel matches the value specified in VI6_RPFn_CKEY_SET0.GY0, the color replacement function replaces the values of the input α and all YCbCr components with the α and color components specified in VI6_RPFn_CKEY_SET1. When the CV bit is set to 1, the color replacement function is enabled. When using the virtual input function (VI6_RPFn_INFMT.VIR = 1), set this bit to 0. 0b: Color replacement function is disabled (transparent color mode). 1b: Color replacement function is enabled.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
1	SAPE1	0h	RW	Comparison Color Data Setting 1 Enable/Disable This bit enables or disables comparison color data setting 1 in the transparent color mode for the color keying module. This bit setting is valid only when the CV bit is set to 0; it is ignored when the CV bit is set to 1. In transparent color mode, color information 1 (VI6_RPFn_CKEY_SET1.R1/GY1/B1) specified in VI6_RPFn_CKEY_SET1 is compared with the input component values. When the input data is in an RGB format, and if all input values match the specified color information, the input α value is replaced with the value specified in VI6_RPFn_CKEY_SET1.AP. When the input data is in YCbCr format, only the Y data is compared. When using the virtual input function (VI6_RPFn_INFMT.VIR = 1), set this bit to 0. 0b: Comparison color data setting 1 is disabled. 1b: Comparison color data setting 1 is enabled.
0	SAPE0	0h	RW	Comparison Color Data Setting 0 Enable/Disable This bit enables or disables comparison color data setting 0 in the transparent color mode for the color keying module. This bit setting is valid only when the CV bit is set to 0b; it is ignored when the CV bit is set to 1b. In transparent color mode, color information 0 (VI6_RPFn_CKEY_SET0.R0/GY0/B0) specified in VI6_RPFn_CKEY_SET0 is compared with the input component values. When the input data is in an RGB format, and if all input values match the specified color information, the input α value is replaced with the value specified in VI6_RPFn_CKEY_SET0.AP. When the input data is in YCbCr format, only the Y data is compared. When using the virtual input function (VI6_RPFn_INFMT.VIR = 1b), clear this bit to 0b. 0b: Comparison color data setting 0 is disabled. 1b: Comparison color data setting 0 is enabled.

(12) RPFn Color Keying Color Setting Registers-m (LCDC_VSPD_VI6_RPFn_CKEY_SETm) (n = 0, 1) (m = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 032Ch + m x 0004h (n = 0)
<LCDC_vspd_base> + 042Ch + m x 0004h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AP0[7:0]								R0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GY0[7:0]								B0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	AP0[7:0]	00h	RW	α Data in Color Keying Color Information-1 - In transparent color-matched color mode for color keying When the input data matches color setting-0 (R0, GY0, and B0) for transparent color comparison in the color keying module, the input α value is replaced with the value specified in these bits. Specify the α value to replace the input value. - In transparent color-luma threshold mode for color keying When the input data is in YCbCr format, and if input Y value is equal to or smaller than GY0, the input α value is replaced with the value specified in this bit field (AP0). Specify the α value in this bit field (AP0) to replace the input value. - In color replacement mode for color keying These bits are not used in this mode. Clear them to 0. α value replacement through these bits in transparent color mode for color keying takes priority over the α value selected through the VI6_RPFn_ALPH_SEL.ASEL setting. A value from 0 to 255 can be specified.
23 to 16	R0[7:0]	00h	RW	R*[1]/Cr Component Data in Color Keying Color Information-1 - In transparent color-matched color mode for color keying Specify the R component value for comparison enabled through the VI6_RPFn_CKEY_CTRL.SAPE0 setting. When the RPFn input is in YCbCr format, the color keying module does not compare the Cr component, and the setting of these bits is ignored. - In transparent color-luma threshold mode for color keying The color keying module does not refer this bit field, and the setting of these bits is ignored. - In color replacement mode for color keying Specify the R component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified.
15 to 8	GY0[7:0]	00h	RW	G*[1]/Y Component Data in Color Keying Color Information-1 - In transparent color-matched color mode for color keying Specify the G/Y component value for comparison enabled through the VI6_RPFn_CKEY_CTRL.SAPE0 setting. - In transparent color-luma threshold mode for color keying Specify the Y component value in the GY0 to be compared. - In color replacement mode for color keying Specify the G/Y component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	B0[7:0]	00h	RW	B*[1]/Cb Component Data in Color Keying Color Information-1 - In transparent color-matched color mode for color keying Specify the B component value for comparison enabled through the VI6_RPFn_CKEY_CTRL.SAPE0 setting. When the RPFn input is in YCbCr format, the color keying module does not compare the Cb component, and the setting of these bits is ignored. - In transparent color-luma threshold mode for color keying The color keying module does not refer this bit field, and the setting of these bits is ignored. - In color replacement mode for color keying Specify the B component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified.

Note 1. When comparison data is specified in an RGB format, if a packed format is selected for RGB input and each of the RGB components is not 8 bits, the lower-order bits of input data are extended as specified through VI6_RPFn_INFMT.CEXT before comparison. The RGB components to be compared with the input should also be extended in the same way and the extended values should be specified in this register.

(13) RPFn Source Picture Memory Stride Setting Registers (LCDC_VSPD_VI6_RPFn_SRCM_PSTRIDE) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0334h (n = 0)
<LCDC_vspd_base> + 0434h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PICT_STRD_Y[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PICT_STRD_C[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	PICT_STRD_Y [15:0]	0000h	RW	Memory Stride of Source Picture Y/RGB Plane These bits specify in 1-byte units the memory stride of the source picture Y/RGB plane read by the RPFn. A value from 0 to 65535 can be specified. Refer to **Figure 9.4-16** for settings.
15 to 0	PICT_STRD_C [15:0]	0000h	RW	Memory Stride of Source Picture C Plane These bits specify in 1-byte units the memory stride of the source picture C plane read by the RPFn. When an RGB-format picture is read, these bits do not need to be set. A value from 0 to 65535 can be specified. Refer to **Figure 9.4-16** for settings. In the YCbCr planar format, this setting is used as the memory stride of the Cb and Cr planes.

This register specifies the memory stride of the picture planes in the source area as shown in **Figure 9.4-16**. The memory stride of the α plane should be specified through VI6_RPFn_SRCM_ASTRIDE.ALPH_STRD. When the RPF input is in an RGB format, only the RGB plane is read; when the input is in YCbCr format, the Y and C planes are read as shown in **Figure 9.4-16**. When the α plane is used, the α plane is also read. According to the image format and the necessity of the α plane, specify the necessary addresses where the source image is stored (VI6_RPFn_SRCM_ADDR_Y, VI6_RPFn_SRCM_ADDR_C, and VI6_RPFn_SRCM_ADDR_AI).

Whether the α plane needs to be read is determined according to the α plane selection method and IROP operation type. For details, refer to **(6) RPFn α Plane Selection Control Registers (LCDC_VSPD_VI6_RPFn_ALPH_SEL)**.

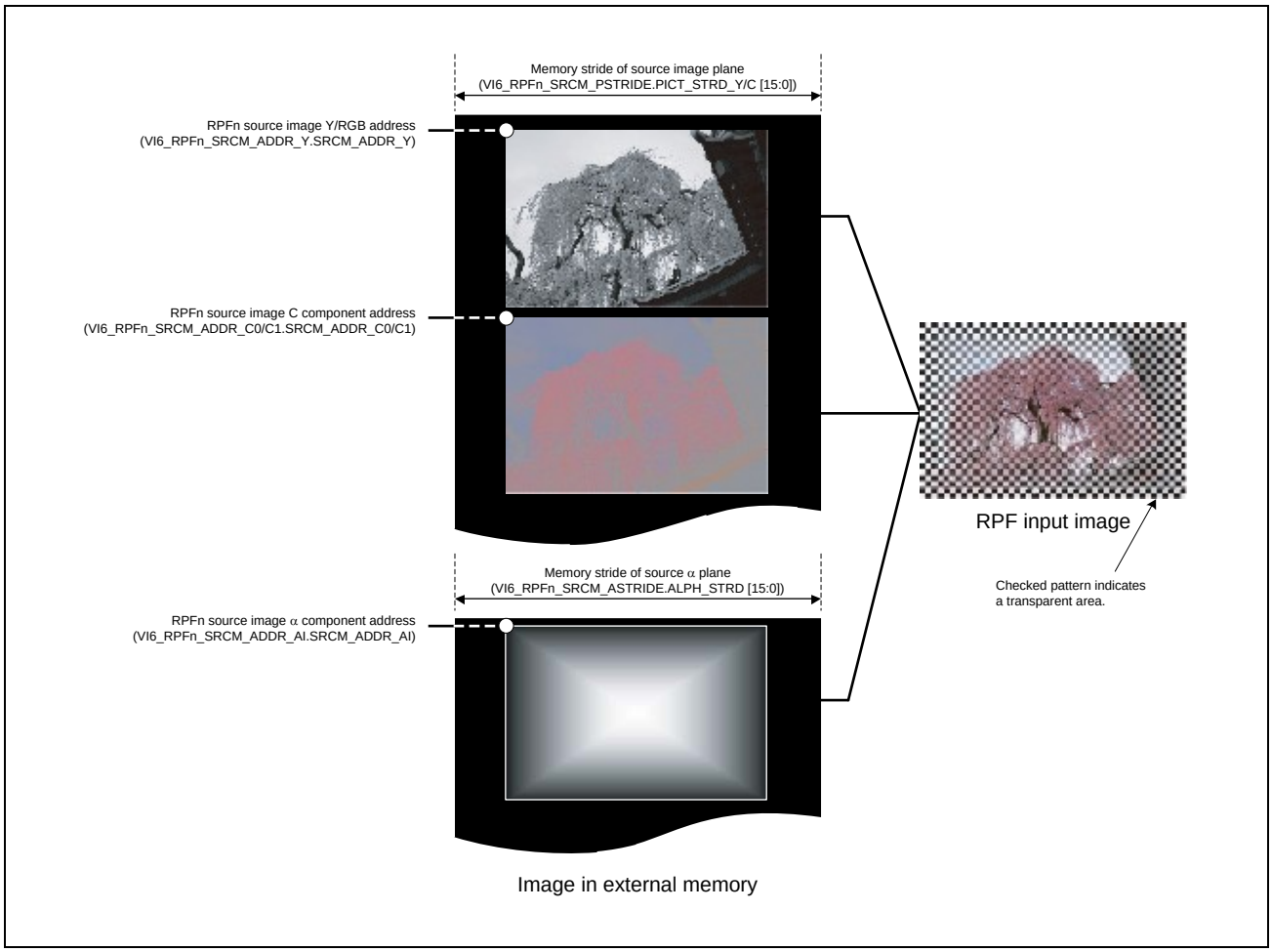


Figure 9.4-16 Reading an Image from RPFn Source Area

(14) RPFn Source α Memory Stride Setting Registers (LCDC_VSPD_VI6_RPFn_SRCM_ASTRIDE) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0338h (n = 0)
<LCDC_vspd_base> + 0438h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALPH_STRD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	ALPH_STRD [15:0]	0000h	RW	Memory Stride of Source α Plane These bits specify in 1-byte units the memory stride of the source α plane read by the RPFn. A value from 0 to 65535 can be specified. Refer to **Figure 9.4-16** for settings.

(15) RPFn Source Y/RGB Address Registers (LCDC_VSPD_VI6_RPFn_SRCM_ADDR_Y) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 033Ch (n = 0)
<LCDC_vspd_base> + 043Ch (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCM_ADDR_Y[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCM_ADDR_Y[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRCM_ADDR_Y[31:0]	0000_0000h	RW	Source Image Y/RGB Plane Storing Address These bits specify in 1-byte units the start address of the source image Y plane and packed RGB plane read by the RPFn. A value from 0000_0000h to FFFF_FFFFh can be specified. Refer to **Figure 9.4-16** in **(13) RPFn Source Picture Memory Stride Setting Registers (VI6_RPFn_SRCM_PSTRIDE)** for settings.

(16) RPFn Source Chroma Address Registers 0 (LCDC_VSPD_VI6_RPFn_SRCM_ADDR_C0) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0340h (n = 0)
<LCDC_vspd_base> + 0440h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCM_ADDR_C0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCM_ADDR_C0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRCM_ADDR_C0[31:0]	0000_0000h	RW	Source Image C Plane Storing Address 0 These bits specify in 1-byte units the start address of the source image C plane read by the RPFn. Here, the C plane indicates the combined CbCr plane when a semi-planar format is selected from the packed YCbCr formats shown in **Figure 9.4-16** or the Cb plane when a planar format is selected. When an interleaved format is selected or the RPF input is in an RGB format, this setting is not used. A value from 0000_0000h to FFFF_FFFFh can be specified. Refer to **Figure 9.4-16** in **(13) RPFn Source Picture Memory Stride Setting Registers (VI6_RPFn_SRCM_PSTRIDE)** for settings.

(17) RPFn Source Chroma Address Registers 1 (LCDC_VSPD_VI6_RPFn_SRCM_ADDR_C1) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0344h (n = 0)
<LCDC_vspd_base> + 0444h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCM_ADDR_C1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCM_ADDR_C1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRCM_ADDR_C1[31:0]	0000_0000h	RW	Source Image C Plane Storing Address 1 These bits specify in 1-byte units the start address of the Cr plane when a planar YCbCr format shown in **Figure 9.4-16** is read by the RPFn. This setting is not used when the RPF input is in YCbCr format that is not a planar format or in an RGB format. A value from 0000_0000h to FFFF_FFFFh can be specified. Refer to **Figure 9.4-16** in **(13) RPFn Source Picture Memory Stride Setting Registers (VI6_RPFn_SRCM_PSTRIDE)** for settings.

(18) RPFn Source α Address Registers (LCDC_VSPD_VI6_RPFn_SRCM_ADDR_AI) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0348h (n = 0)
<LCDC_vspd_base> + 0448h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCM_ADDR_AI[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCM_ADDR_AI[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRCM_ADDR_AI[31:0]	0000_0000h	RW	Source Image α Plane Storing Address These bits specify in 1-byte units the start address of the α plane of the source image read by the RPFn. Specify in the same way as the start address of the picture plane. When the α plane is not read from the source area, these bits do not need to be set. A value from 0000_0000h to FFFF_FFFFh can be specified. Refer to **Figure 9.4-16** in **(13) RPFn Source Picture Memory Stride Setting Registers (VI6_RPFn_SRCM_PSTRIDE)** for settings.

(19) RPFn Bus Access Control Registers (LCDC_VSPD_VI6_RPFn_BAC) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 0350h (n = 0)
<LCDC_vspd_base> + 0450h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	B512
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	B512	0h	RW	Burst access in 512 pixels' enable 0b: burst access in 256 pixels 1b: burst access in 512 pixels Always set 0b to this bit
15 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(20) RPFn Multiple Alpha Control Register (LCDC_VSPD_VI6_RPFn_MULT_ALPH) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 036Ch (n = 0)
<LCDC_vspd_base> + 046Ch (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	A_MMD	-	-	P_MMD[1:0]		ALPHA_RATIO[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12	A_MMD	0h	RW	0b: Alpha data go through multiply-alpha unit. 1b: Multiply-alpha unit multiplies alpha data by specified alpha (ALPHA_RATIO[7:0]. When output format from CSC unit is YCbCr, set 0b to this bit.
11, 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9, 8	P_MMD[1:0]	0h	RW	0h: Image data go through multiply-alpha unit. 1h: Multiply-alpha unit multiplies image data by specified alpha (ALPHA_RATIO[7:0]). 2h: Multiply-alpha unit multiplies image data by alpha data. 3h: Multiply-alpha unit multiplies image data by alpha data and specified alpha (ALPHA_RATIO[7:0]. When output format from CSC unit is YCbCr, set 0 to this bit.
7 to 0	ALPHA_RATIO [7:0]	00h	RW	Multiply alpha value

- ALPin: Input Alpha data to Multiply-alpha unit
- PIXin(R): Input R data to Multiply-alpha unit
- PIXin(G): Input G data to Multiply-alpha unit
- PIXin(B): Input B data to Multiply-alpha unit
- ALPout: Output Alpha data from Multiply-alpha unit
- PIXout(R): Output R data from Multiply-alpha
- PIXout(G): Output G data from Multiply-alpha
- PIXout(B): Output B data from Multiply-alpha

Table 9.4-16 Expression of Output Alpha Data from Multiply-Alpha Unit

A_MMD	ALPHA_RATIO[7:0]	Expression
0	Don't care	ALPout = ALPin
1	Not 255	ALPout = ALPin × ALPHA_RATIO / 256
	255	ALPout = ALPin

Table 9.4-17 Expression of Output Pixel Data from Multiply-Alpha Unit

P_MMD[1:0]	ALPHA_RATIO[7:0]	ALPin	Expression
0	Don't care	Don't care	PIXout(R) = PIXin(R) PIXout(G) = PIXin(G) PIXout(B) = PIXin(B)
1	Not 255	Don't care	PIXout(R) = PIXin(R) × ALPHA_RATIO / 256 PIXout(G) = PIXin(G) × ALPHA_RATIO / 256 PIXout(B) = PIXin(B) × ALPHA_RATIO / 256
	255	Don't care	PIXout(R) = PIXin(R) PIXout(G) = PIXin(G) PIXout(B) = PIXin(B)
2	Don't care	Not 255	PIXout(R) = PIXin(R) × ALPin / 256 PIXout(G) = PIXin(G) × ALPin / 256 PIXout(B) = PIXin(B) × ALPin / 256
	Don't care	255	PIXout(R) = PIXin(R) PIXout(G) = PIXin(G) PIXout(B) = PIXin(B)
3	Not 255	Not 255	PIXout(R) = PIXin(R) × ALPin × ALPHA_RATIO / 256 / 256 PIXout(G) = PIXin(G) × ALPin × ALPHA_RATIO / 256 / 256 PIXout(B) = PIXin(B) × ALPin × ALPHA_RATIO / 256 / 256
	255	Not 255	PIXout(R) = PIXin(R) × ALPin / 256 PIXout(G) = PIXin(G) × ALPin / 256 PIXout(B) = PIXin(B) × ALPin / 256
	Not 255	255	PIXout(R) = PIXin(R) × ALPHA_RATIO / 256 PIXout(G) = PIXin(G) × ALPHA_RATIO / 256 PIXout(B) = PIXin(B) × ALPHA_RATIO / 256
	255	255	PIXout(R) = PIXin(R) PIXout(G) = PIXin(G) PIXout(B) = PIXin(B)

9.4.3.2.7 WPF Control Registers

(1) WPF0-Source-RPF Register (LCDC_VSPD_VI6_WPF0_SRCRPF)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 1000h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	VIR_ACT2[1:0]		-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	RPF1_ACT [1:0]		RPF0_ACT [1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25, 24	VIR_ACT2[1:0]	0h	RW	Virtual RPF Start Enable in BRS These bits enable start of the virtual RPF in the BRS as the source RPF for the WPF0 when the WPF0 is started. For details of the virtual RPF, refer to the following. **9.4.3.2.10 (1) BRS Input Control Register (VI6_BRS_INCTRL)** **9.4.3.2.10 (2) Size Register of BRS Input Virtual RPF (VI6_BRS_VIRRPF_SIZE)** **9.4.3.2.10 (3) Display Location Register of BRS Input Virtual RPF (VI6_BRS_VIRRPF_LOC)** **9.4.3.2.10 (4) Color Information Register of BRS Input Virtual RPF (VI6_BRS_VIRRPF_COL)** Note that the virtual RPF is in the BRS as shown in **Figure 9.4-29** and there are no register bits for DPR setting related to the virtual RPF. 0h: The virtual RPF in the BRS is not started. 1h: The virtual RPF in the BRS is started as a sublayer source RPF for the WPF0. 2h: The virtual RPF in the BRS is started as the master-layer source RPF for the WPF0. 3h: Setting prohibited
23 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3, 2	RPF1_ACT[1:0]	0h	RW	RPF1 Start Enable These bits enable start of RPF1 as the source RPF for the WPF0 when the WPF0 is started. When RPF1 is not started by the WPF0, set the VI6_DPR_RPF1_ROUTE.RT_RPF1 bits to 63. 0h: RPF1 is not started. 1h: RPF1 is started as a sublayer source RPF for the WPF0. 2h: RPF1 is started as the master-layer source RPF for the WPF0. 3h: Setting prohibited
1, 0	RPF0_ACT[1:0]	0h	RW	RPF0 Start Enable These bits enable start of RPFn as the source RPF for the WPF0 when the WPF0 is started. When RPF0 is not started by the WPF0, set the VI6_DPR_RPF0_ROUTE.RT_RPF0 bits to 63. 0h: RPF0 is not started. 1h: RPF0 is started as a sublayer source RPF for the WPF0. 2h: RPF0 is started as the master-layer source RPF for the WPF0. 3h: Setting prohibited

When the WPFn is started through the VSPD start register n (VI6_CMDn: n = 0), the RPF and virtual RPF in the BRS specified as the source RPF in this register are also started to supply data to the VSPD internal modules.

Note the following when specifying the source RPF.

(1) When blending or ROP operation is applied to multiple images through the BRS, multiple source RPFs are necessary for one WPF. When multiple source RPFs are used, images should be classified into a master layer and sublayers; assign one of the source RPFs as the master-layer source RPF and other RPFs as sublayer source RPFs. Do not assign all RPFs as sublayer source RPFs (VI6_WPF1_SRCRPF = 00000005h) or two or more RPFs as the master-layer source RPF (VI6_WPF0_SRCRPF = 0000000Ah) (such settings are prohibited).

(2) When the BRS is not used, there should be only one source RPF for one WPF. In this case, the source RPF should be assigned as the master-layer source RPF.

(2) WPF0 Horizontal Input Size Clipping Register (LCDC_VSPD_VI6_WPF0_HSZCLIP)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 1004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	HCEN	-	-	-	-	HCL_OFST[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	HCL_SIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	HCEN	0h	RW	Horizontal Size Clipping Enable/Disable Enables or disables clipping of the horizontal size of the WPF0 input image. 0b: Horizontal size clipping is disabled 1b: Horizontal size clipping is enabled
27 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	HCL_OFST[7:0]	00h	RW	Horizontal Size Clipping Offset Value Setting These bits specify the offset size (pixels) from the left end of the image in horizontal size clipping when the HCEN bit is 1b (**Figure 9.4-17**). The left side of the image input to the WPF is cut off for the size specified in these bits. When the HCEN bit is 0b, this setting is ignored. A value from 0 to 255 can be specified. (HCL_OFST + HCL_SIZE) should not exceed the horizontal size of the WPF input. If the setting shown in the bottom example in **Figure 9.4-17** is made, VSPD does not operate correctly.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	HCL_SIZE [12:0]	0000h	RW	Horizontal Clipping Size Setting When the HCEN bit is 1b, these bits specify the clipping size for horizontal clipping processing. Through this processing, the area of the horizontal size specified through the HCL_SIZE bits starting from the offset position specified through the HCL_OFST bits is determined as the valid image area. Accordingly, the right-side pixels beyond the (HCL_OFST + HCL_SIZE) size in the WPF0 input image are discarded. When the HCEN bit is 0b, this setting is ignored. A value from 1 to 1920 can be specified. (HCL_OFST + HCL_SIZE) should not exceed the horizontal size of the WPF input. If the setting shown in the bottom example in **Figure 9.4-17** is made, VSPD does not operate correctly. When the WPF0 output format is YCbCr4:2:2 or YCbCr4:2:0, specify an even value in these bits.

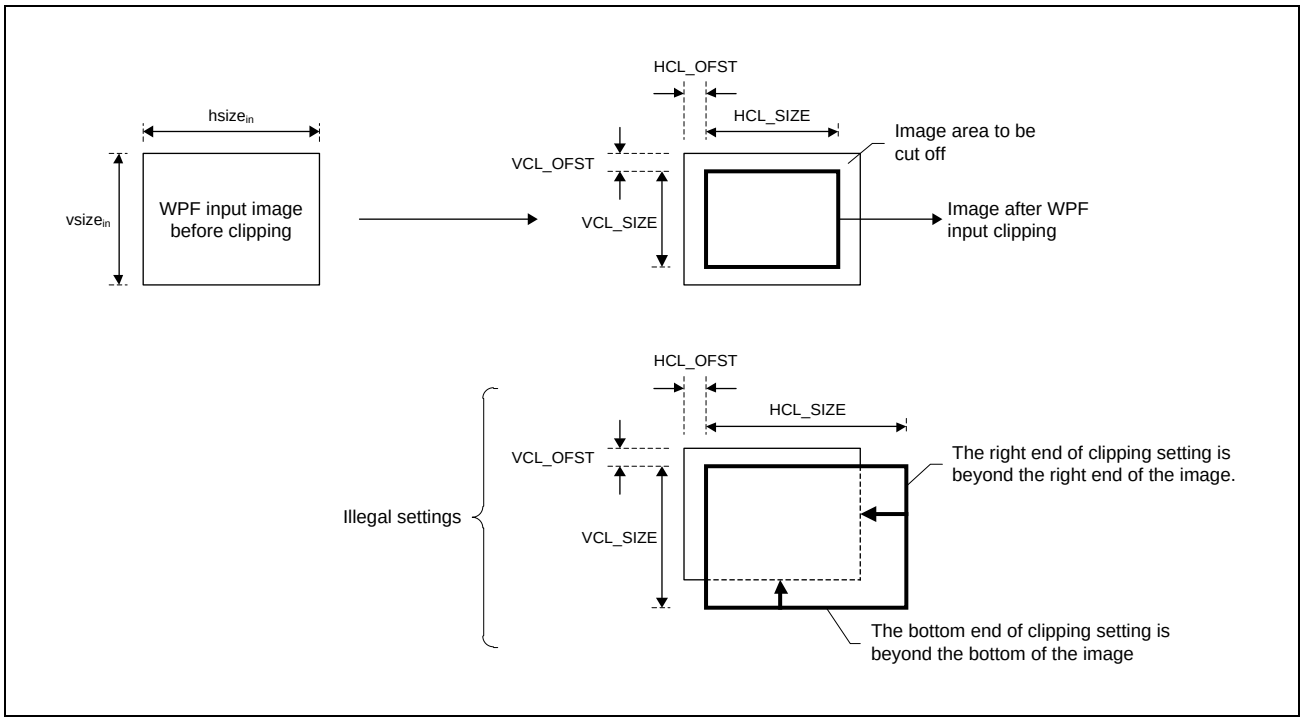


Figure 9.4-17 Image Clipping in WPF Input Section

(3) WPF0 Vertical Input Size Clipping Register (LCDC_VSPD_VI6_WPF0_VSZCLIP)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 1008h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	VCEN	-	-	-	-	VCL_OFST[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VCL_SIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	VCEN	0h	RW	Vertical Size Clipping Enable/Disable Enables or disables clipping of the vertical size of the WPF0 input image. 0b: Vertical size clipping is disabled 1b: Vertical size clipping is enabled
27 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	VCL_OFST [7:0]	00h	RW	Vertical Size Clipping Offset Value Setting These bits specify the offset size (pixels) from the top end of the image in vertical size clipping when the VCEN bit is 1 (**Figure 9.4-17**). The top of the image input to the WPF is cut off for the size specified in these bits. When the VCEN bit is 0b, this setting is ignored. A value from 0 to 255 can be specified. (VCL_OFST + VCL_SIZE) should not exceed the vertical size of the WPF input. If the setting shown in the bottom example in **Figure 9.4-17** is made, VSPD does not operate correctly.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	VCL_SIZE [12:0]	0000h	RW	Vertical Clipping Size Setting When the VCEN bit is 1b, these bits specify the clipping size for vertical clipping processing. Through this processing, the area of the vertical size specified through the VCL_SIZE bits starting from the offset position specified through the VCL_OFST bits is determined as the valid image area. Accordingly, the bottom pixels beyond the (VCL_OFST + VCL_SIZE) size in the WPF0 input image are discarded. When the VCEN bit is 0b, this setting is ignored. A value from 1 to 1920 can be specified. (VCL_OFST + VCL_SIZE) should not exceed the vertical size of the WPF input. If the setting shown in the bottom example in **Figure 9.4-17** is made, VSPD does not operate correctly. When the WPF0 output format is YCbCr4:2:0, specify an even value in these bits.

(4) WPF0 Output Format Register (LCDC_VSPD_VI6_WPF0_OUTFMT)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 100Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PDV [7:0]								PXA	ODE	-	-	-	ROT[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPYCS	SPUVS	DITH[1:0]		WRTM[2:0]			CSC	-	WRFMT[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	PDV [7:0]	00h	RW	PAD Value in Output Packed Data These bits specify the value to be stored in the bit field indicated as PAD or P in the output formats shown in **Table 9.4-18**. To store this value in PAD, specify 0 in the PXA bit. A value from 0 to 255 can be specified.
23	PXA	0h	RW	PAD Data Select Selects the value to be stored in the bit field indicated as PAD or P in the packed RGB output formats shown in **Table 9.4-19**. Both the value specified in the PDV bits and the α data input from the DPR to WPF are 8 bits, but some of the PAD and P bit fields shown in **Table 9.4-18** are four bits or one bit. When the target bit field is not 8 bits, the number of bits in the PDV value and the α data input from the DPR to WPF is reduced according to the VI6_WPF0_RNDCTRL.ABRM setting. For bit count reduction, refer to **Figure 9.4-18** and the description of VI6_WPF0_RNDCTRL.ABRM. 0b: The value specified in the PDV bits is stored in the PAD shown in **Table 9.4-18**. 1b: The α value output from DPR in pixel units is stored in the PAD shown in **Table 9.4-18**.
22	ODE	0h	RW	Ordered Dither (mode A) Enable/Disable 0b: Ordered dither (mode A) is disabled. 1b: Ordered dither (mode A) is enabled. When the output format specified through the WRFMT bits is YCbCr, specify 0b in this bit. And when VI6_WPF0_OUTFMT.CSC is set to 1b, specify 0b in this bit even in the case that the output format specified through the WRFMT bits is RGB. Ordered dither is available only for 18bpp. So, when ODE bit is equal to 1b, set WRFMT at 18bpp format. When ODE bit is equal to 0, WPF dither method is specified by DITH[1:0] in the register Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.
21 to 19	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	ROT[2:0]	0h	RW	Rotation Processing Select These bits select the flipping processing to be applied to the WPF0 output image. **Figure 9.4-19** shows the correspondence between the original image and the flipping result according to each setting. 0b: No flipping 1b: Vertical flipping Note that the destination address setting should be changed according to the setting of these bits. For details, refer to **9.4.3.2.7 (9) WPFn Destination Y/RGB Address Registers (VI6_WPFn_DSTM_ADDR_Y)**. When the LIF module is used (VI6_LIF_CTRL.LIF_EN = 1b), set 0b to ROT[2:0].
15	SPYCS	0h	RW	WPF Output Mode Setting 1 When the output format is YUY2, set this bit to 1b and set the WRFMT bits to 71 (47h). When the output format is YVYU, set this bit and the SPUVS bit to 1b and set the WRFMT bits to 71 (47h). In other cases, set this bit to 0b.

Bit	Bit Name	Initial Value	R/W	Description
14	SPUVS	0h	RW	WPF Output Mode Setting 2 When the output format is NV61, set this bit to 1b and set the WRFMT bits to 65 (41h). When the output format is NV21, set this bit to 1b and set the WRFMT bits to 66 (42h). When the output format is YVYU, set this bit and the SPYCS bit to 1b and set the WRFMT bits to 71 (47h). In other cases, set this bit to 0b.
13, 12	DITH[1:0]	0h	RW	Ordered Dither (mode B) Enable/Disable When the output format specified through the WRFMT bits is RGB with 18 bpp (260000 colors) or less, the color reduction processing is applied to match the number of colors. The color reduction processing may generate the artifacts of pseudo gradation, which can be suppressed through dithering. The DITH bits enable or disable dithering during color reduction. When the output format specified through the WRFMT bits is YCbCr, specify 0 in these bits. And when VI6_WPF0_OUTFMT.CSC is set to 1, specify 0 in these bits even in the case that the output format specified through the WRFMT bits is RGB. 0h: Dithering (mode B) is disabled 3h: Dithering (mode B) is enabled 1, 2h: Setting prohibited When ODE bit in the register is 1b, set this bit to 0h.
11 to 9	WRTM[2:0]	0h	RW	CSC Conversion Expression Setting These bits select the expression for color space conversion. The conversion direction is RGB to YCbCr when the format specified in the WRFMT bits is RGB, or YCbCr to RGB when the format is YCbCr. 0h: BT.601 YCbCr [16,235/240] ↔ RGB [0,255] 1h: BT.601 YCbCr [0,255] ↔ RGB [0,255] 2h: BT.709 YCbCr [16,235/240] ↔ RGB [0,255] 3h: BT.709 YCbCr [16,235/240] ↔ RGB [16,235] 4h to 7h: Setting prohibited
8	CSC	0h	RW	Color Space Conversion Setting Enables or disables YCbCr ↔ RGB color space conversion to be executed in the WPF0. The characteristics of color space conversion are determined by the WRTM setting. There are some points to be noted about the relationship between the CSC setting and output format (WRFMT). For details refer to (*1) in **9.4.3.2.6 (3) RPFn Input Format Registers (VI6_RPFn_INFMT)**. 0b: Color space is not converted. 1b: Color space is converted.
7	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
6 to 0	WRFMT[6:0]	00h	RW	WPF Output Image Format Setting These bits select the format of the image output from the WPF0 to the external memory from among those listed in **Table 9.4-18** and **Table 9.4-19**. [Note] Number of output pixels When YCbCr4:2:2 is specified through WRFMT, the horizontal size of the output image should be a multiple of 2 pixels. When YCbCr4:2:0 is specified, the vertical and horizontal sizes of the output image should be multiples of 2 pixels. Specify an appropriate data flow of the source RPF → DPR → target WPF so that the size of the image input to the target WPF satisfies the above restrictions. In particular, when the data flow includes a module or a function that modifies (up-scales, down-scales, or clips) the image size, take special care about the module or function settings. [Note] Output lines in YCbCr4:2:0 In the YCbCr4:2:0 output format, the number of chrominance lines in the vertical direction is one-half the number of luminance lines. For this reason, the WPF outputs only even-numbered chrominance lines (lines 0, 2, 4, 6, ...) (conversion from (A) to (B) in. When vertical flipping is also specified through the ROT bits, the flipping processing is executed last and the chrominance line locations are inverted (lines 1, 3, 5, 7, …) in the output image ((C) in **Figure 9.4-20**). [Note] Down sampling of CbCr in horizontal direction in YCbCr4:2:0 or YCbCr4:2:2 In the YCbCr4:2:2 or YCbCr4:2:0 output format, method of down sample of Cb/Cr in horizontal direction is average of neighbor two pixels.

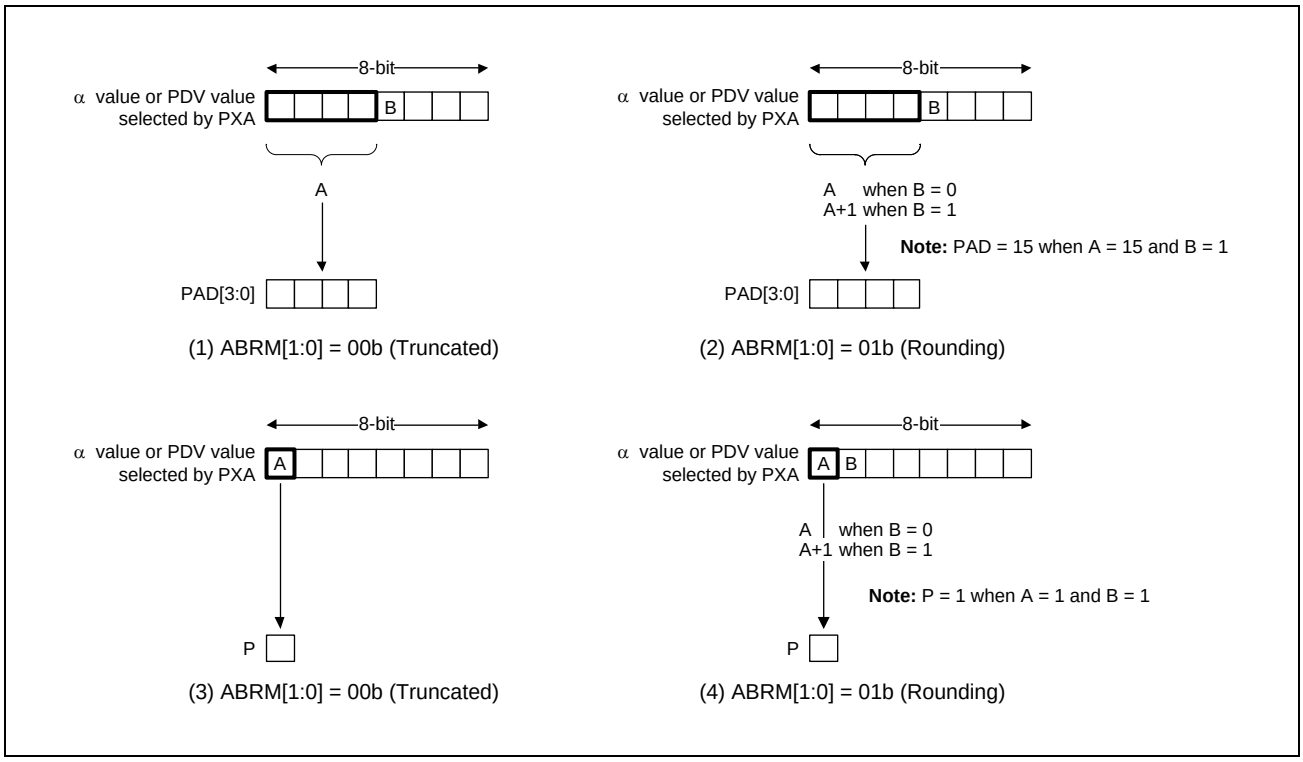
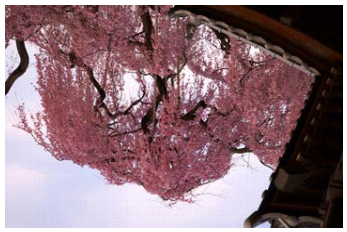


Figure 9.4-18 Selection of PAD Value and Reduction of Bit Count through PXA Setting

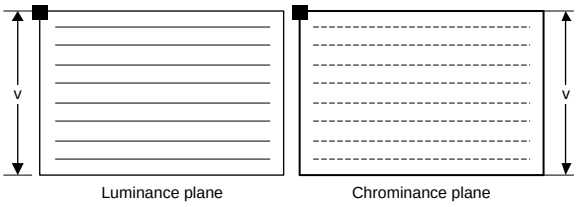
Without flipping (ROT = 0)

Vertical flipping

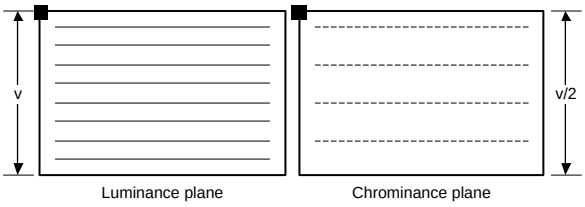


Vertical flipping (ROT = 1)

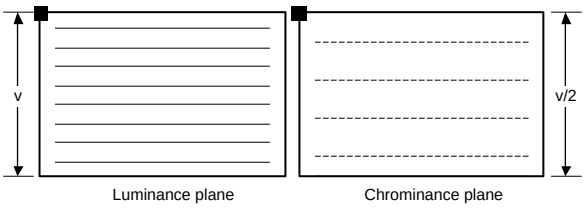
Figure 9.4-19 Correspondence between Original Image and Flipping Result according to ROT Setting



(A) Before conversion to YCbCr4:2:0



(B) After conversion to YCbCr4:2:0 without vertical flipping



(C) After conversion to YCbCr4:2:0 with vertical flipping

Figure 9.4-20 Chrominance Output Lines in YCbCr4:2:0 and Vertical Flipping Result

Table 9.4-18 Packed RGB Formats for WPF Output

WRFMT[6:0]	Bit per pixel	Phase	upper row - address / bottom row - bit field																													
			n								n+1								n+2								n+3							
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
00h	8	—	R0	R0	R0	G0	G0	G0	B0	B0	R1	R1	R1	G1	G1	G1	B1	B1	R2	R2	R2	G2	G2	G2	B2	B2	R3	R3	R3	G3	G3	G3	B3	B3
01h	12	—	0	0	0	0	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
02h			R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	0	0	0	0
03h	—	—	Reserved								Reserved								Reserved								Reserved							
04h	15	—	0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
05h			R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	0
06h	16	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
07h	18	—	PAD								0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
08h			R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	PAD							
09h			0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	PAD							
0Ah			PAD								R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0
0Bh			PAD								0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0
0Ch			0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	PAD							
0Dh			PAD								R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0
0Eh			R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	PAD							
0Fh		0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	R1	R1
		1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	0	0	0	0	0	0	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2
		2	G2	G2	B2	B2	B2	B2	B2	B2	0	0	0	0	0	0	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3
10h		0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	R1	R1	R1	R1	R1	R1	G1	G1
		1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	0	0	0	0	0	0	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	B2	B2	B2	B2
		2	B2	B2	0	0	0	0	0	0	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	0	0	0	0	0	0
11h		0	0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1
		1	0	0	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2
		2	0	0	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3
12h		0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1	0	0
		1	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2	0	0
		2	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3	0	0
13h	24	—	PAD								R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
14h			R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	PAD							
15h		0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
		1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
		2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
16h	18	—	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	0	0	0	0	0	0	0	G0	G0	B0	B0	B0	B0	B0	B0
17h			0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
18h	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1
		1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2
		2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3
19h	12	—	PAD				R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	PAD				R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
1Ah		—	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	PAD				R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	PAD			
1Bh	15	—	P	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	P	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
1Ch		—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	P	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	P
1Dh	12	—	PAD				B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	PAD				B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1
1Eh		—	B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	PAD				B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1	PAD			
1Fh	15	—	P	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	P	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1
20h		—	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	P	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	P
21h	18	0	0	0	B0	B0	B0	B0	B0	B0	0	0	G0	G0	G0	G0	G0	G0	0	0	R0	R0	R0	R0	R0	R0	0	0	B1	B1	B1	B1	B1	B1
		1	0	0	G1	G1	G1	G1	G1	G1	0	0	R1	R1	R1	R1	R1	R1	0	0	B2	B2	B2	B2	B2	B2	0	0	G2	G2	G2	G2	G2	G2
		2	0	0	R2	R2	R2	R2	R2	R2	0	0	B3	B3	B3	B3	B3	B3	0	0	G3	G3	G3	G3	G3	G3	0	0	R3	R3	R3	R3	R3	R3
22h	24	—	PAD								B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0
23h	16	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0
24h to 3Fh	—	—	Reserved								Reserved								Reserved								Reserved							

Table 9.4-19 Packed YCbCr Formats for WPF Output

WRFMT[6:0]	Packed YCbCr Output Format	Reference
40h	YCbCr4:4:4 semi-planar	**Figure 9.4-10***4
41h	YCbCr4:2:2 semi-planar (NV16, NV61*1)	
42h	YCbCr4:2:0 semi-planar (NV12, NV21*1)	
43h to 45h	Reserved	—
46h	YCbCr4:4:4 interleaved	**Figure 9.4-11***4
47h	YCbCr4:2:2 interleaved type 0 (UYVY, YUY2*2, YVYU*3)	
48h	YCbCr4:2:2 interleaved type 1	
49h	YCbCr4:2:0 interleaved*5	
4Ah	YCbCr4:4:4 planar	**Figure 9.4-12***4
4Bh	YCbCr4:2:2 planar (YV16)	
4Ch	YCbCr4:2:0 planar (YV12, I420)	
4Dh to 7Fh	Reserved	—

Note 1. When the output format is NV61 or NV21, set SPUVS (bit 14) to 1.

Note 2. When the output format is YUY2, set SPYCS (bit 15) to 1.

Note 3. When the output format is YVYU, set SPUVS (bit 14) to 1 and SPYCS (bit 15) to 1.

Note 4. **Figure 9.4-13** shows the definition of memory address for each pixel in **Figure 9.4-10** to **Figure 9.4-12**.

Note 5. Each line of plane is written twice, so byte/pixel of YCbCr420ITL is 3 byte/pixel (same as YCbCr444ITL).

For details of each YCbCr format, refer to **Figure 9.4-10** to **Figure 9.4-13**. In these figures, registers for the RPF are indicated; read them as registers for the WPF as follows.

(RPF registers in the figures)	→	(Corresponding WPF registers)
VI6_RPFn_SRCM_ADDR_Y.SRCM_ADDR_Y	→	VI6_WPFn_DSTM_ADDR_Y.DSTM_ADDR_Y
VI6_RPFn_SRCM_ADDR_C0.SRCM_ADDR_C0	→	VI6_WPFn_DSTM_ADDR_C0.DSTM_ADDR_C0
VI6_RPFn_SRCM_ADDR_C1.SRCM_ADDR_C1	→	VI6_WPFn_DSTM_ADDR_C1.DSTM_ADDR_C1

(5) WPF0 Data Swapping Register (LCDC_VSPD_VI6_WPF0_DSWAP)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 1010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	P_LLS	P_LWS	P_WDS	P_BTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	P_LLS	0h	RW	WPF Output Data Swapping in LONG LWORD Units The effect of this bit setting is the same as data swapping in the RPF; refer to **Table 9.4-13**. 0b: Data swapping in LONG LWORD (64-bit) units is disabled 1b: Data swapping in LONG LWORD (64-bit) units is enabled
2	P_LWS	0h	RW	WPF Output Data Swapping in long word Units The effect of this bit setting is the same as data swapping in the RPF; refer to **Table 9.4-13**. 0b: Data swapping in long word (32-bit) units is disabled 1b: Data swapping in long word (32-bit) units is enabled
1	P_WDS	0h	RW	WPF Output Data Swapping in Word Units The effect of this bit setting is the same as data swapping in the RPF; refer to **Table 9.4-13**. 0b: Data swapping in word (16-bit) units is disabled 1b: Data swapping in word (16-bit) units is enabled
0	P_BTS	0h	RW	WPF Output Data Swapping in Byte Units The effect of this bit setting is the same as data swapping in the RPF; refer to **Table 9.4-13**. 0b: Data swapping in byte (8-bit) units is disabled 1b: Data swapping in byte (8-bit) units is enabled

Table 9.4-13 shows the data order before and after swapping according to the long long word, long word, word, and byte swapping settings.

When data order in memory for each format is the same as in **Table 9.4-18** for RGB format and **Figure 9.4-10** to **Figure 9.4-13** for YCbCr format, set 1111b to {*_LLS, *_LWS, *_WDS, *_BTS}. If data order is not the same as the definition, change data order within 16byte unit by these bits as shown in **Table 9.4-13**.

(6) WPF0 Rounding Control Register (LCDC_VSPD_VI6_WPF0_RNDCTRL)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 1014h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	CBRM	-	-	ABRM[1:0]		ATHRESH[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	CLMD[1:0]		-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	CBRM	0h	RW	Bit Count Reduction Method Selection for Data Storage in Packed RGB This bit specifies the method for reducing the number of bits when data is stored in the bit fields indicated as R, G, and B in **Table 9.4-18** and the target bit fields are not 8 bits. 0b: Bit count conversion: The lower-order bits are truncated 1b: Bit count conversion: Rounding (rounding off)
27 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25, 24	ABRM[1:0]	0h	RW	Bit Count Reduction Method Selection for Data Storage in PAD These bits specify the method for reducing the number of bits when the data selected through VI6_WPF0_OUTFMT.PXA is stored in the bit fields indicated as PAD or P in **Table 9.4-18** and the target bit field is four bits or one bit. A value of 10b can be specified only when the packed RGB format specified through VI6_WPF0_OUTFMT.WRFMT includes a 1-bit P field. In this case, when the data selected through VI6_WPF0_OUTFMT.PXA is greater than the ATHRESH value, 1 is stored in the P field; when the selected data is not greater than the ATHRESH value, 0 is stored. 00b: Bit count conversion: The lower-order bits are truncated 01b: Bit count conversion: Rounding (rounding off) 10b: Bit count conversion: Comparison with the threshold value (this setting is allowed only when the storage field is one bit) 11b: Setting prohibited
23 to 16	ATHRESH[7:0]	00h	RW	Threshold for Conversion to 1-Bit α Data These bits specify the threshold value used for conversion from 8-bit α data to one bit when the ABRM bits are set to 10b. When the 8-bit α value before bit count reduction is equal to or smaller than the ATHRESH value, 0 is stored as the reduced 1-bit α data. In other cases, 1 is stored as the 1-bit α data. A value from 0 to 255 can be specified.
15 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13, 12	CLMD[1:0]	0h	RW	Color Data Clipping These bits specify the method for clipping the YCbCr color data output from the WPF. When RGB color data is output from the WPF, specify 0b in these bits. 00b: Output value is not clipped (0-255) 01b: Output value is clipped: YCbCr mode 1 (16-235 (Y), 16-240 (Cb/Cr)) 10b: Output value is clipped: YCbCr mode 2 (Y/Cb/Cr = 1-254) 11b: Setting prohibited
11 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(7) WPF0 Destination Y Plane Memory Stride Register (LCDC_VSPD_VI6_WPF0_DSTM_STRIDE_Y)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 101Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PICT_STRD_Y[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	PICT_STRD_Y [15:0]	0000h	RW	Memory Stride of Destination Picture Y/RGB Plane These bits specify in 1-byte units the memory stride of the destination picture in the external memory to be written to by the WPF0 as shown in **Figure 9.4-21**. A value from 0000h to FFFFh can be specified.

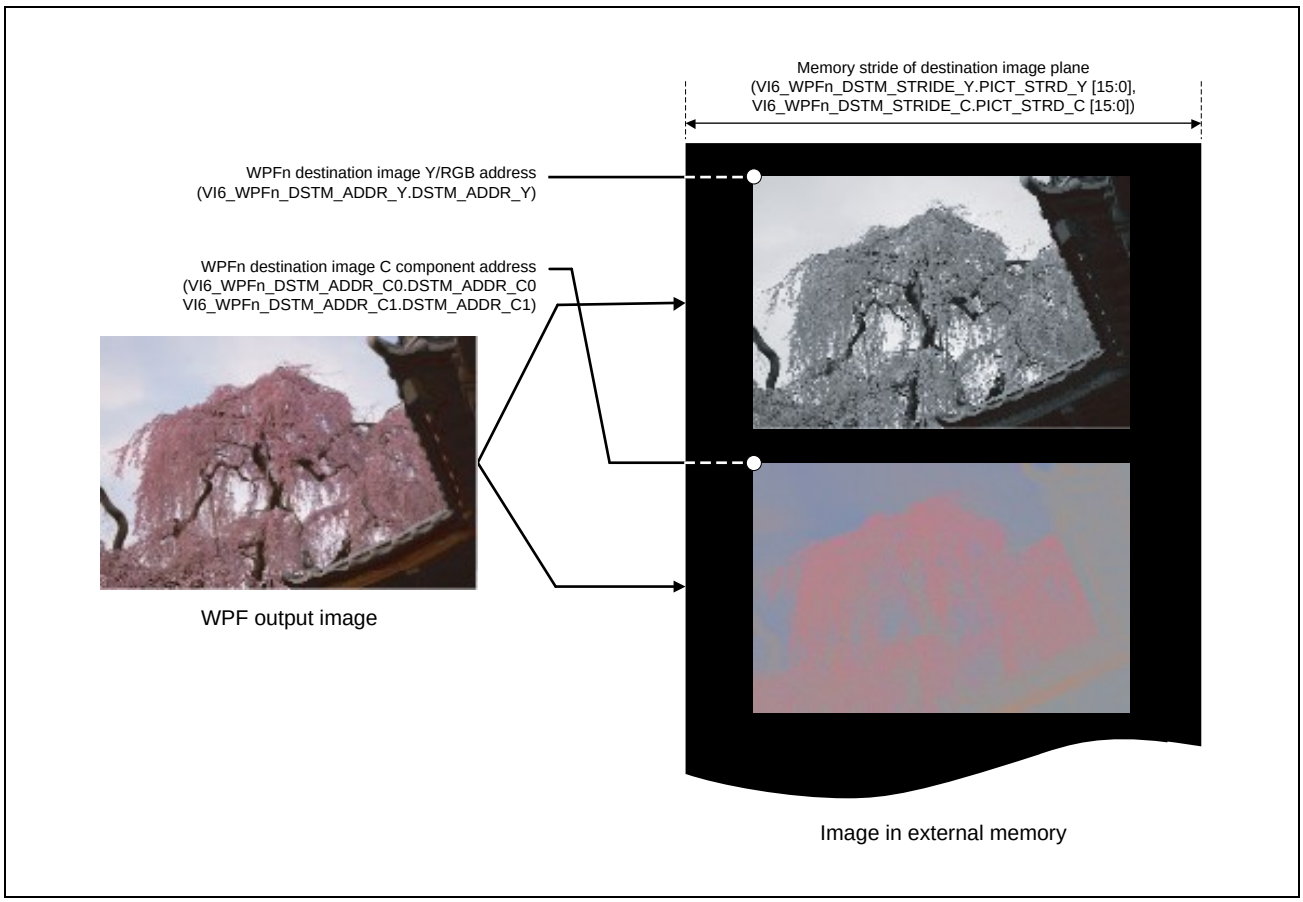


Figure 9.4-21 Writing Image Data to Destination Area in WPFn

(8) WPF0 Destination C Plane Memory Stride Register (LCDC_VSPD_VI6_WPF0_DSTM_STRIDE_C)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 1020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PICT_STRD_C[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	PICT_STRD_C [15:0]	0000h	RW	Memory Stride of Destination Picture C Plane These bits specify in 1-byte units the memory stride for the C plane of the destination picture in the external memory to be written to by the WPF0 as shown in **Figure 9.4-21**. When the WPF0 outputs images in an RGB format, this setting is not used. When the WPF0 outputs images in YCbCr planar format, this setting is applied to both the Cb and Cr planes. A value from 0000h to FFFFh can be specified.

(9) WPF0 Destination Y/RGB Address Register (LCDC_VSPD_VI6_WPF0_DSTM_ADDR_Y)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 1024h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSTM_ADDR_Y[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSTM_ADDR_Y[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSTM_ADDR_Y [31:0]	0000_0000h	RW	Destination Image Y/RGB Plane Storing Address These bits specify in 1-byte units the address for storing the destination-image Y plane or packed RGB plane to be written to by the WPF0 in the method described later. A value from 0000_0000h to FFFF_FFFFh can be specified.

[Destination Address Specification Method]

When flipping is not used, the start address of a frame (address FHA shown in **Figure 9.4-22**) should be specified as the destination address. When flipping is used, the destination address is not the frame start address (FHA); one of addresses A0 and A2 shown in **Figure 9.4-22** should be selected according to the combination of desired flipping (VI6_WPFn_OUTFMT.ROT setting).

To strictly define locations A0 and A2, let the horizontal size of the output image be H, the vertical size of the output image be V, and the memory stride (VI6_WPFn_DSTM_STRIDE_Y/C setting) be S as shown in **Figure 9.4-22**. Calculate the destination address (one of A0 and A2) using the formula shown in **Table 9.4-20** and specify it in the destination address storing register.

The values of variables L in **Table 9.4-20** depend on the other register settings and luminance and chrominance components. These values should be obtained by referring to **Table 9.4-21** when calculating the address to be specified in VI6_WPFn_DSTM_ADDR_Y, or **Table 9.4-22** when calculating the address to be specified in VI6_WPFn_DSTM_ADDR_C0 or VI6_WPFn_DSTM_ADDR_C1.

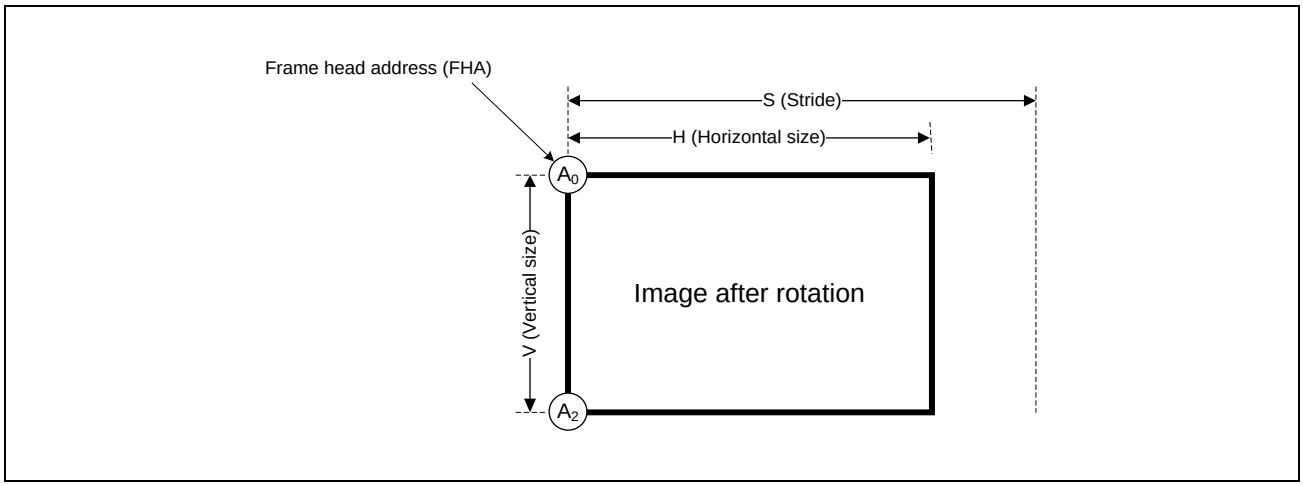


Figure 9.4-22 Location of Destination Address to be Specified for Flipping

Table 9.4-20 Destination Address A_0, A_1, A_2, and A3 Calculation Formulas

VI6_WPFn_OUTFMT.ROT Setting	Formula for Calculating Address to be Set in VI6_WPFn_DSTM_ADDR_Y, VI6_WPFn_DSTM_ADDR_C0, and VI6_WPFn_DSTM_ADDR_C1
0	A_0 = FHA
1	A_2 = FHA + (V×L – 1) × S

Table 9.4-21 Value of L according to VI6_WPFn_OUTFMT.WRFMT Setting (for RGB and Luminance Y Address Calculation)

VI6_WPFn_OUTFMT.WRFMT	L
73	0.5
0 to 2, 4 to 35, 64 to 66, 70 to 72, or 74 to 76	1

Table 9.4-22 Value of L according to VI6_WPFn_OUTFMT.WRFMT Setting (for Chrominance C0 and C1 Address Calculation)

VI6_WPFn_OUTFMT.WRFMT	L
70 to 73	Not defined
66, 76	0.5
64 to 65, 74 to 75	1

(10) WPF0 Destination Chroma Address Register 0 (LCDC_VSPD_VI6_WPF0_DSTM_ADDR_C0)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 1028h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSTM_ADDR_C0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSTM_ADDR_C0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSTM_ADDR_C0[31:0]	0000_0000h	RW	Destination Image C Plane Storing Address 0 These bits specify in 1-byte units the address for storing the destination-image C plane to be written to by the WPF0. Refer to the description of VI6_WPF0_DSTM_ADDR_Y for settings. Here, the C plane indicates the combined CbCr plane when a semi-planar format is selected from the packed YCbCr formats shown in **Table 9.4-19** or the Cb plane when a planar format is selected. When an interleaved format is selected or the output is in an RGB format, this setting is not used. A value from 0000_0000h to FFFF_FFFFh can be specified. Refer to **Figure 9.4-22** in **9.4.3.2.7 (9) WPFn Destination Y/RGB Address Registers (VI6_WPFn_DSTM_ADDR_Y)** for settings.

(11) WPF0 Destination Chroma Address Register 1 (LCDC_VSPD_VI6_WPF0_DSTM_ADDR_C1)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 102Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSTM_ADDR_C1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSTM_ADDR_C1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSTM_ADDR_C1[31:0]	0000_0000h	RW	Destination Image C Plane Storing Address 1 These bits specify in 1-byte units the address for storing the Cr plane when the WPF0 outputs images to the external memory in YCbCr planar format shown in **Table 9.4-19**. Refer to the description of VI6_WPF0_DSTM_ADDR_Y for settings. This setting is not used when the WPF outputs in YCbCr format that is not a planar format or in an RGB format. A value from 0000_0000h to FFFF_FFFFh can be specified. Refer to **Figure 9.4-22** in **9.4.3.2.7 (9) WPFn Destination Y/RGB Address Registers (VI6_WPFn_DSTM_ADDR_Y)** for settings.

(12) WPF0 LIF Write Back Control Register (LCDC_VSPD_VI6_WPF0_WRBCK_CTRL)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 1034h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WBMD[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	WBMD[1:0]	0h	RW	Display Data Write Back Control This bit is used for selecting the write back mode when the value of VI6_LIF0_CTRL.LIF_EN bit is set to 1b. 0b: Write Back Disabled 1b: Write Back Enabled

9.4.3.2.8 DPR Control Registers

(1) Concept of DPR Settings

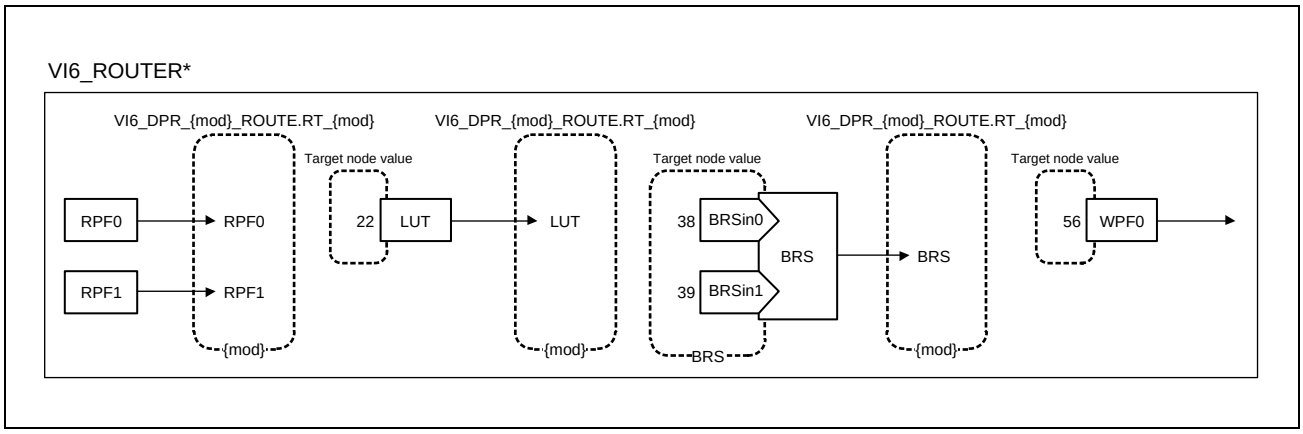


Figure 9.4-23 Node Register Names and Target Node Values on Data Path Router

In the VSPD internal data path, the order of processes can be specified as desired. The module for performing each process has a unique node value. Set each bit field in VI6_DPR_*_ROUTE to an appropriate node value shown in **Figure 9.4-23** to specify the target node to be connected behind each module.

For DPR settings, all of the following restrictions should be observed. If any of them is violated, even the WPF paths operating correctly at that time will be affected as well as the WPF paths connected through the DPR, and correct operation will not be guaranteed.

1. Specify 63 for the output node values of all RPFs and processing modules that are not used in the DPR. Here, make sure that no module is connected to a module for which 63 is specified as the node value. Note: VI6_DPR_{mod}_ROUTE.RT are not implemented in VSPD which doesn't have {mod}. See **Figure 9.4-1** whether VSPD has {mod} or not.
2. When specifying a value other than 63 for an output node value in the DPR, make sure that valid inputs (RPF0 to RPF1 or virtual RPF) and a target WPF are determined.
3. Only one module can be connected to each module; specifying the same target node value for two or more modules is prohibited.
4. Desired modules can be connected between each RPF and BRS input port, but all RPFs specified as the sources for a BRS input port should have the same target WPF.
5. Make appropriate routing or RPF register settings so that the color space formats (RGB/YCbCr) for all BRS input ports are the same.
6. Do not connect the output of any module as the input to the same module (in the BRS case, any input port) even when there is another module between the output and input (creating a loop is prohibited).
7. Each node can be used only once throughout all paths from RPFn to WPFn. When a module shown in **Figure 9.4-23** is assigned in one RPF to WPF path, it cannot be used in another RPF to WPF path.
8. While a WPF is operating, modifying the DPR connection settings in VI6_DPR_*_ROUTE is prohibited for modules used by the WPF but allowed for modules not used by the WPF. Be careful not to accidentally modify the settings of the modules included in the WPF path that is operating.
9. Be careful to do the following setting when BRS is used.
   VI6_DPR_BRS_ROUTE.BRSSEL = 1

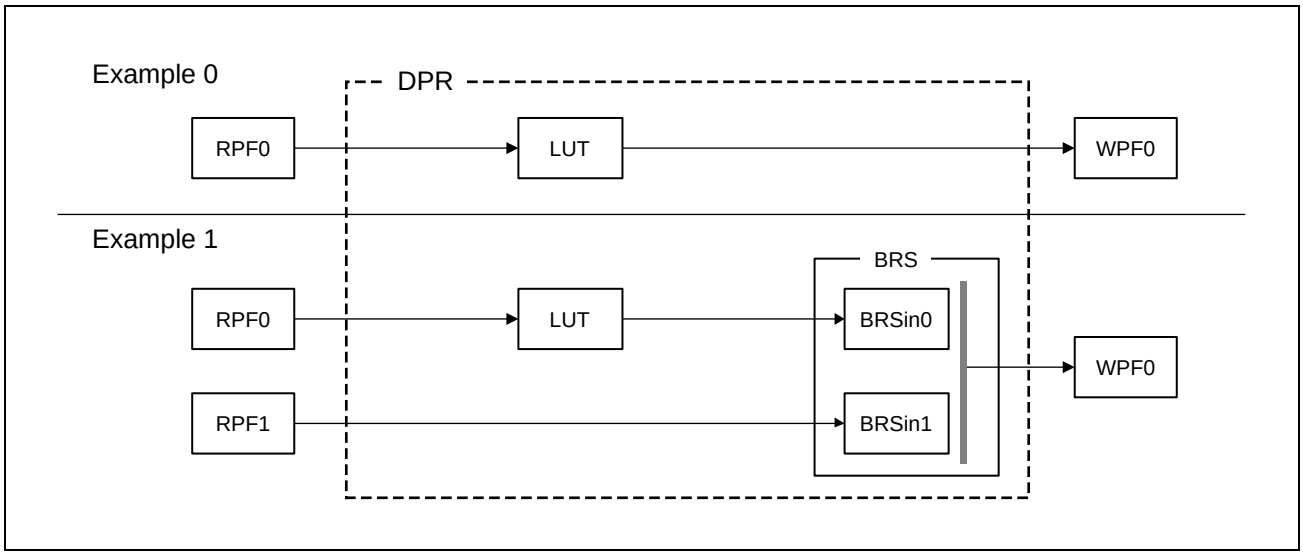


Figure 9.4-24 Examples of Internal Data Path Routing

Figure 9.4-24 shows examples of internal data path routing. Example 0 is WPF0 processing (RPF0 is the source RPF), and example 1 is also WPF0 (RPF0 to RPF1 are the source RPFs). Each example has the configuration shown in **Figure 9.4-24**. Example 0 performs LUT (e.g. γ correction) processing. Example 1 performs LUT processing (e.g. γ correction), for input 0 (RPF0) and then applies blending or raster operation between the resultant data and input data 1 (RPF1). The VI6_ DPR_*_ROUTE settings for these examples are shown in **Table 9.4-23**. The bit fields for the modules that are not used in the examples should be set to 63.

Table 9.4-23 VI6_DPR_*_ROUTE Register Settings in Connection Examples Shown in **Figure 9.4-24**. Although both examples show in **Figure 9.4-24** include operation in image processing modules, connect the RPF to the WPF directly when only image format conversion or packed format conversion is required.

Table 9.4-23 Examples of Internal Data Path Routing

	Register Name	Setting	
Example 0	VI6_DPR_RPF0_ROUTE.RT_RPF0	22	(To LUT)
	VI6_DPR_RPF1_ROUTE.RT_RPF1	63	(UNUSED)
	VI6_DPR_LUT_ROUTE.RT	56	(To WPF0)
	VI6_DPR_BRS_ROUTE.RT	63	(UNUSED)
Example 1	VI6_DPR_RPF0_ROUTE.RT_RPF0	22	(To LUT)
	VI6_DPR_RPF1_ROUTE.RT_RPF1	39	(To BRSin1)
	VI6_DPR_LUT_ROUTE.RT	38	(To BRSin0)
	VI6_DPR_BRS_ROUTE.RT	56	(To WPF0)

(2) RPFn Routing Register (LCDC_VSPD_VI6_DPR_RPFn_ROUTE) (n = 0, 1)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 2000h (n = 0)
<LCDC_vspd_base> + 2004h (n = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	RT_RPFn[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5 to 0	RT_RPFn[5:0]	00h	RW	RPFn Target Node Value These bits specify the target node value for RPFn. When using RPFn, refer to **Figure 9.4-23** for settings. When RPFn is not started through the VI6_WPF0_SRCRPF setting, specify 63.

(3) WPF0 Timing Control Register (LCDC_VSPD_VI6_DPR_WPF0_FPORCH)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 2014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	FP_WPFn[5:0]						-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 8	FP_WPFn[5:0]	00h	RW	WPFn Internal Operation Timing Setting Specify 5.
7 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

(4) LUT Routing Register (LCDC_VSPD_VI6_DPR_LUT_ROUTE)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 203Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	FXA[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	FP[5:0]						-	-	RT[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 16	FXA[7:0]	00h	RW	Fixed α Output Value for LUT The LUT does not support input/output of the α value. The α value input to the LUT is discarded, and the fixed α value specified in these bits is always output from the LUT. A value from 0 to 255 can be specified.
15, 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 8	FP[5:0]	00h	RW	LUT Internal Operation Timing Setting Specify 0.
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5 to 0	RT[5:0]	00h	RW	LUT Target Node Value These bits specify the target node value for the LUT. When using the LUT, refer to **Figure 9.4-23** for settings. When not using the LUT, specify 63.

(5) BRS Routing Register (LCDC_VSPD_VI6_DPR_BRS_ROUTE)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 2050h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	BRSSEL	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	FP[5:0]						-	-	RT[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	BRSSEL	0h	RW	Always specify 1.
27 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 8	FP[5:0]	00h	RW	BRS Internal Operation Timing Setting Specify 0.
7, 6	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
5 to 0	RT[5:0]	00h	RW	BRS Target Node Value These bits specify the target node value for the BRS. When using the BRS, refer to **Figure 9.4-23** for settings. When not using the BRS, specify 63.

9.4.3.2.9 LUT Control Register

(1) LUT Control Register (LCDC_VSPD_VI6_LUT_CTRL)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 2800h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LUT_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	LUT_EN	0h	RW	1D-LUT Enable/Disable Enables or disables the 1D-LUT function by the LUT. When the 1D-LUT is used, the color component curve information needs to be set separately in the LUT table. For the LUT table settings, refer to **9.4.4.7.1 LUT**. 0b: 1D-LUT function is disabled 1b: 1D-LUT function is enabled

In the LUT, various image processing, such as curves with high operation load (e.g., γ correction), negative-positive conversion, and gain adjustment of images, can be achieved by the data replacement processing by the 1D-LUT. As shown in **Figure 9.4-25**, the LUT replaces each component of the input pixel data using the set replacement table of 256 entries. For example, if the LUT is set as in **Figure 9.4-26**, when there is an input of 150, the data stored in address 150 of the 1D-LUT is read and output as the LUT output. **Figure 9.4-26** shows a case in which the input and output become equal for convenience in explaining.

In the LUT settings shown in **Figure 9.4-27**, the input bits are reversed. This has the effect of negative-positive flipping. In **Figure 9.4-28**, γ correction (γ = 1.8 is shown as an example) is possible. As described above, information to be set in the LUT indicates LUT processing characteristics. If the same value is set for each component in the LUT, an equal effect can be obtained for each component of the input image when it is processed. If the LUT is set with different characteristics for each component, the processing characteristics can be changed for each component.

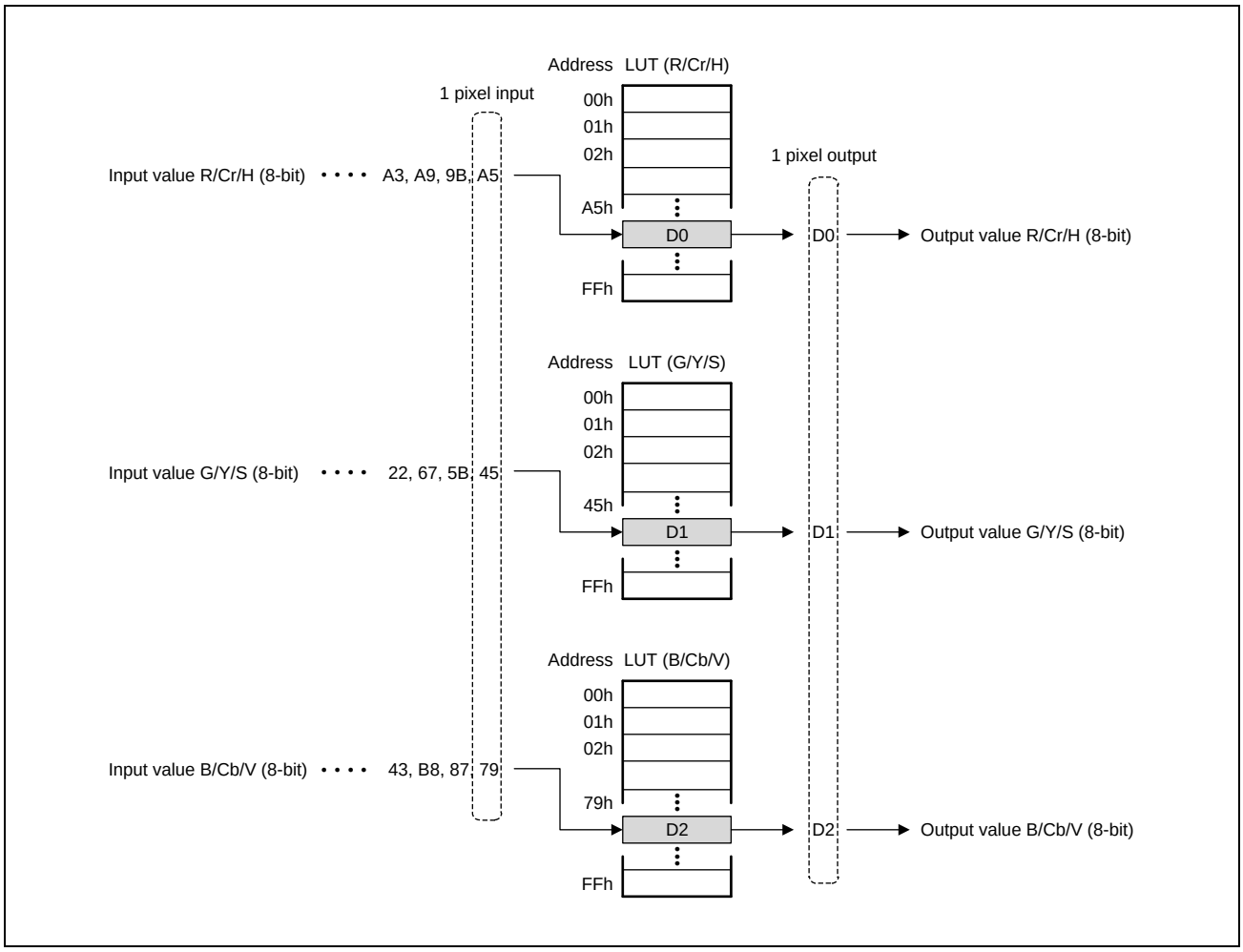


Figure 9.4-25 Relationship between Input and Output for 1D-LUT Table

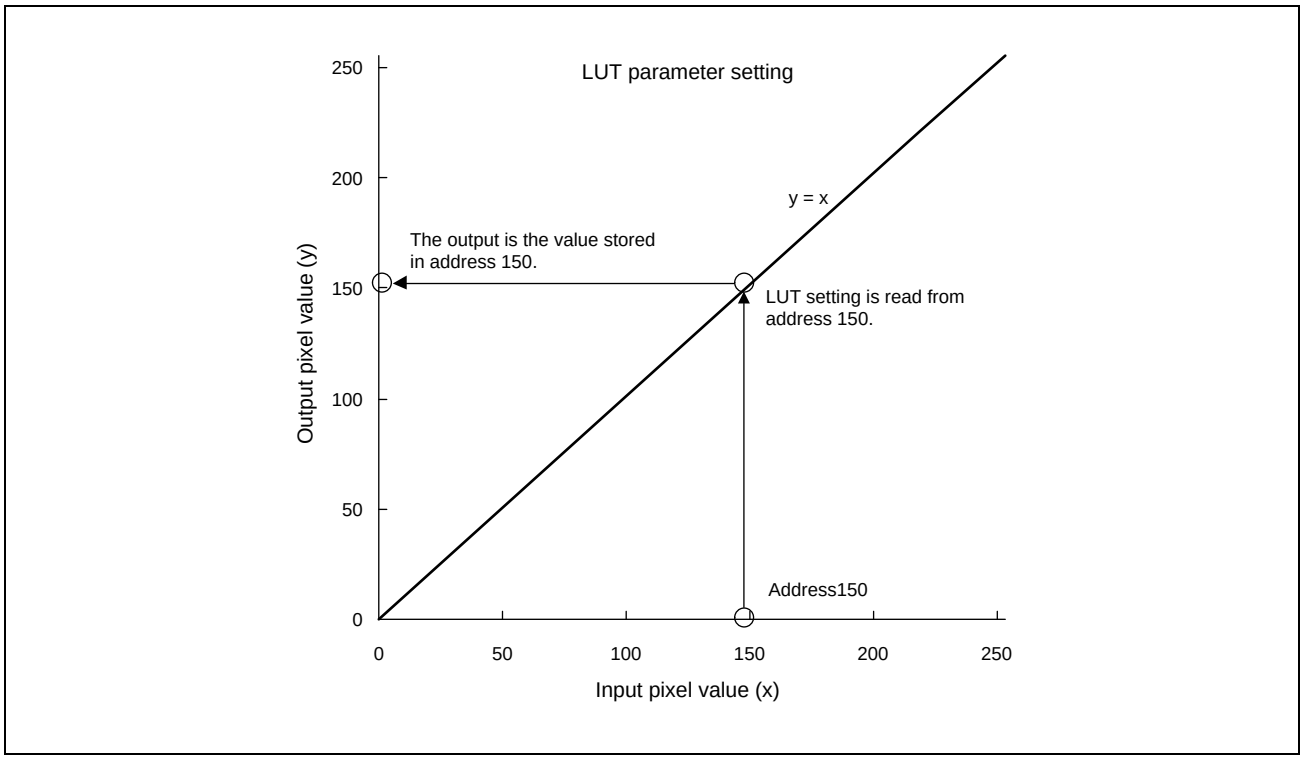


Figure 9.4-26 Setting Example in Which Output Becomes Equal to Input

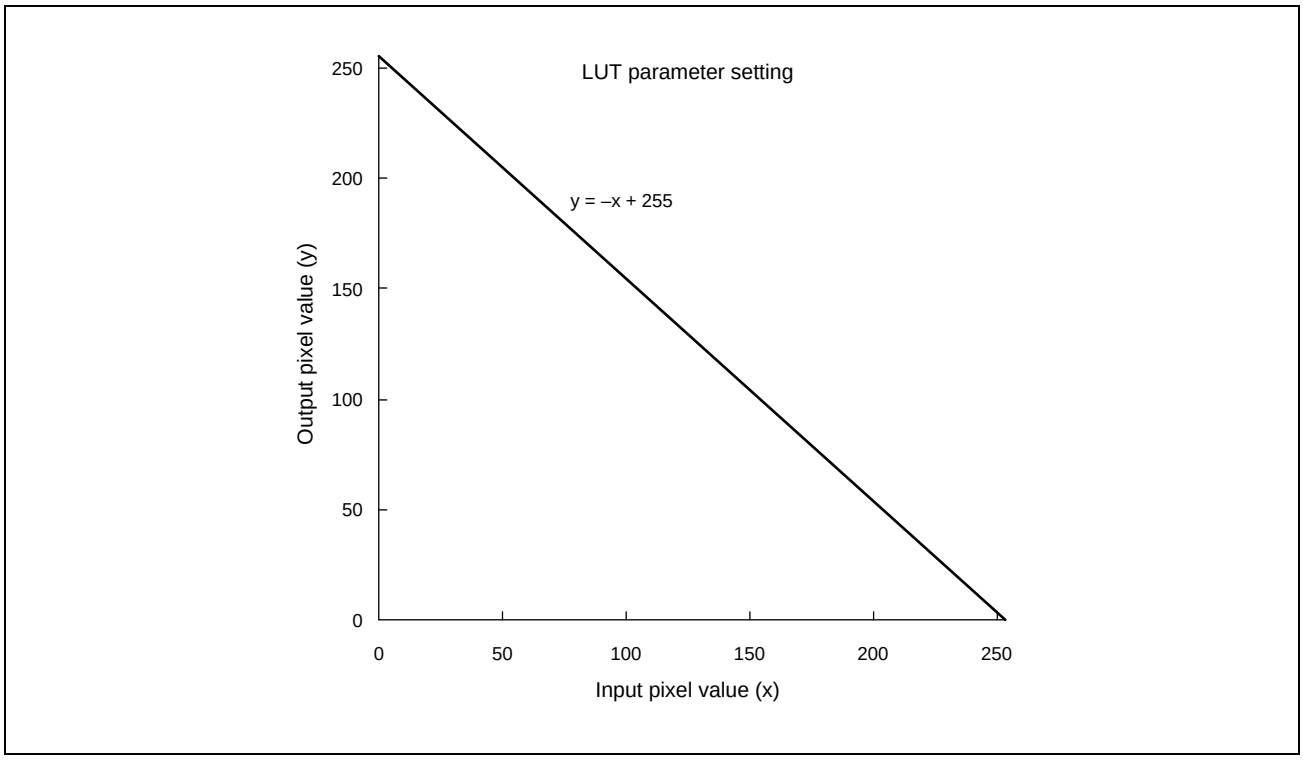


Figure 9.4-27 Setting Example of Negative-positive Conversion

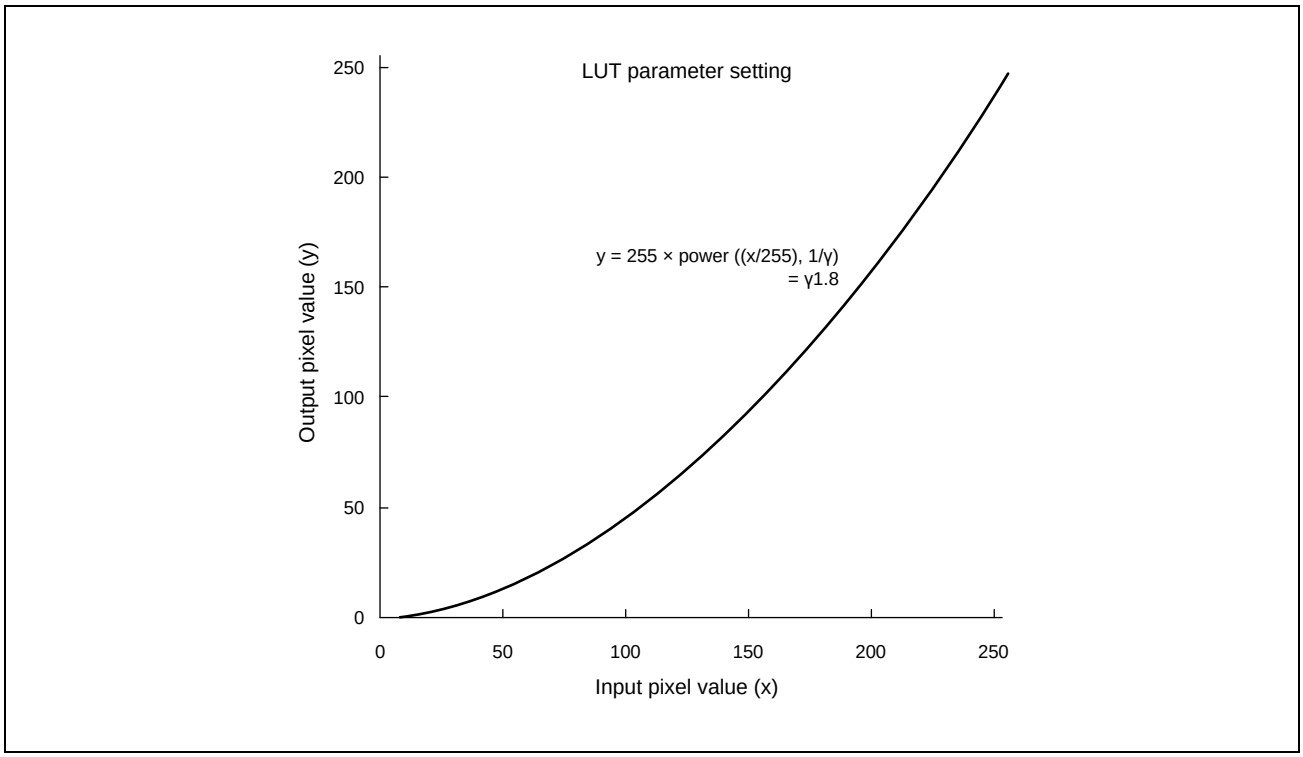


Figure 9.4-28 Setting Example of γ Correction

9.4.3.2.10 BRS Control Registers

(1) BRS Input Control Register (LCDC_VSPD_VI6_BRS_INCTRL)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 3900h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	NRM	-	-	-	-	-	-	-	-	-	-	D1ON	D0ON
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ODE1	DITH1[2:0]			ODE0	DITH0 [2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	NRM	0h	RW	Color Data Normalization Enables or disables division by the α value of the color data in BRS blending operation. This is used when converting the RGB color data format to which the α value is multiplied (pre-multiplied color) into the RGB color data format to which the α value is not multiplied (non pre-multiplied color). Do not use this for the YCbCr format. 0b: Divider (DIV unit in **Figure 9.4-29**) does not divide the color value by α 1b: Divider (DIV unit in **Figure 9.4-29**) divides the color value by α
27 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	D1ON	0h	RW	Ordered dither (mode B) Enable of BRS Input 1 Enables or disables dithering (color reduction) of BRS input 1 (BRSin1 in **Figure 9.4-29**). 0b: Dithering (mode B) of BRSin1 is disabled 1b: Dithering (mode B) of BRSin1 is enabled When ODE1 in this bit is set to 1, set 0 to this bit.
16	D0ON	0h	RW	Ordered dither (mode B) Enable of BRS Input 0 Enables or disables dithering (color reduction) of BRS input 0 (BRSin0 in **Figure 9.4-29**). 0b: Dithering (mode B) of BRSin0 is disabled 1b: Dithering (mode B) of BRSin0 is enabled When ODE0 in this bit is set to 1, set 0 to this bit.
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7	ODE1	0h	RW	Ordered Dither (mode A) of CH1 Input to BRS Enable/Disable 0b: Ordered dither (mode A) of BRSin1 is disabled. 1b: Ordered dither (mode A) of BRSin1 is enabled. Ordered dither is available only for 18bpp. So, when ODE1 bit is equal to 1, set DITH1 = 1. When ODE1 bit is equal to 0, BRSin1 dither method is specified by D1ON in the register Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.
6 to 4	DITH1[2:0]	0h	RW	Dithering of CH1 Input to BRS These bits specify the number of colors for pixels after dithering (color reduction) when dithering (color reduction) for pixel information is enabled through the D1ON bit. When dithering (color reduction) for pixel information is disabled, specify 0h in these bits. 0h: Dithering of BRSin1 input image is disabled 1h: Dithering of BRSin1 input image at 18 bpp (RGB666: 260000 colors) 2h: Dithering of BRSin1 input image at 16 bpp (RGB565: 65535 colors) 3h: Dithering of BRSin1 input image at 15 bpp (RGB555: 32768 colors) 4h: Dithering of BRSin1 input image at 12 bpp (RGB444: 4096 colors) 5h: Dithering of BRSin1 input image at 8 bpp (RGB332: 256 colors) 6, 7h: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
3	ODE0	0h	RW	Ordered Dither (mode A) of CH0 Input to BRS Enable/Disable 0b: Ordered dither (mode A) of BRSin0 is disabled. 1b: Ordered dither (mode A) of BRSin0 is enabled. Ordered dither is available only for 18bpp. So, when ODE0 bit is equal to 1b, set DITH0 = 1b. When ODE0 bit is equal to 0, BRSin0 dither method is specified by D0ON in the register Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.
2 to 0	DITH0 [2:0]	0h	RW	Dithering of CH0 Input to BRS These bits specify how to perform dithering of the CH0 input to the BRS. The setting method is the same as that for the DITH1 bits. Read the description of the DITH1 bits with BRSin0 and D0ON replacing BRSin1 and D1ON, respectively.

Figure 9.4-29 shows the configuration of the BRS. For the BRS inputs, there are two inputs from the DPR and one internal input as a virtual RPF. BRSin0 to BRSin1 are input ports that have the target node values shown in **Figure 9.4-23**, and they can be connected to any module on the DPR. The same color space (YCbCr or RGB) has to be used for the two inputs from the DPR to the BRS.

The virtual RPF inside the BRS is an input unit not connected to the DPR. It is called the "virtual RPF" because it outputs images internally created by the BRS. Starting of the virtual RPF is controlled by VI6_WPFn_SRCRPF.VIR_ACT2, and the single-color data created at the virtual RPF can be used for blending or raster operation (ROP) with data from the other input units BRSin0 to BRSin1. The color space for the single color to be set for the virtual RPF needs to match the color space of the two inputs from the DPR to the BRS. For this setting method, see **(4) Color Information Register of BRS Input Virtual RPF (LCDC_VSPD_VI6_BRS_VIRRPF_COL)**.

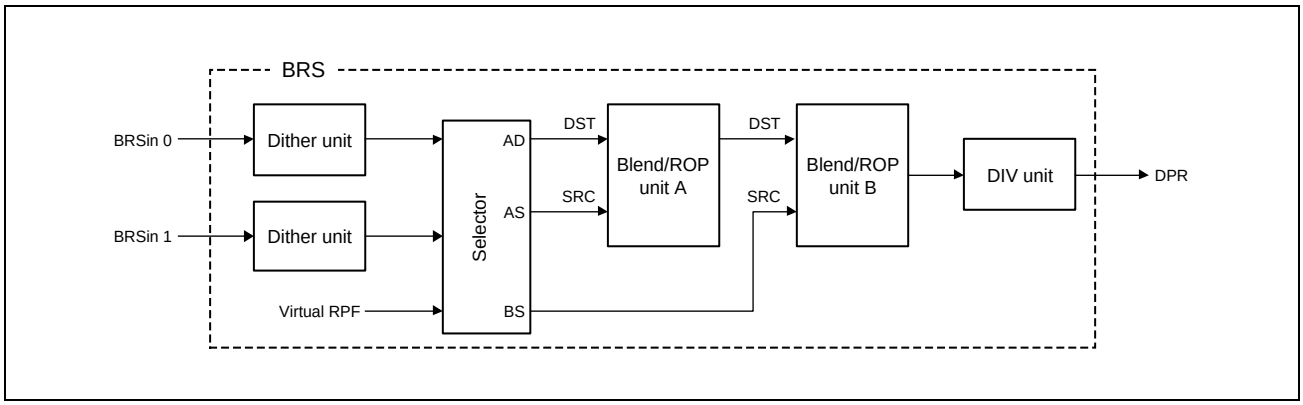


Figure 9.4-29 BRS Configuration

The selector in **Figure 9.4-29** is used to select the SRC and DST inputs to blending/ROP units A to B from BRSin0 to BRSin1 which are inputs from the DPR and the virtual RPF. The SRC and DST input sources for blending/ROP units A to B are either uniquely determined based on the configuration shown in **Figure 9.4-29** or selected as desired by registers. The input sources that can be arbitrarily selected by registers are AD, AS and BS, which correspond to the registers shown in **Table 9.4-24**.

Table 9.4-24 Correspondence between Selector Output Destinations and Register Bits for BRS

Selector Output	Output Destination	Register Bits
AD	Blending/ROP unit A - DST	VI6_BRSA_CTRL.DSTSEL
AS	Blending/ROP unit A - SRC	VI6_BRSA_CTRL.SRCSEL
BS	Blending/ROP unit B - SRC	VI6_BRSB_CTRL.SRCSEL

(2) Size Register of BRS Input Virtual RPF (LCDC_VSPD_VI6_BRS_VIRRPF_SIZE)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 3904h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	VIR_HSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VIR_VSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	VIR_HSIZE [12:0]	0000h	RW	Virtual RPF Horizontal Size These bits set the horizontal size of an image from the virtual RPF shown in **Figure 9.4-29**. A value from 1 to 1920 can be specified.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	VIR_VSIZE [12:0]	0000h	RW	Virtual RPF Vertical Size These bits set the vertical size of an image from the virtual RPF shown in **Figure 9.4-29**. A value from 1 to 1920 can be specified.

The virtual RPF has only a function to output a fixed α value and a fixed pixel value. The virtual RPF can internally create a single-color image without accessing external memory via the MAU. Same as images from the other BRS input ports, a sublayer can be blended on an image created in this manner with the image used as the background (master layer). In turn, when using the image as a sublayer, it can be drawn on the master layer as a window.

(3) Display Location Register of BRS Input Virtual RPF (LCDC_VSPD_VI6_BRS_VIRRPF_LOC)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 3908h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	HCOORD[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VCOORD[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	HCOORD [12:0]	0000h	RW	Horizontal Coordinate of Virtual RPF Location on Master Layer These bits specify the horizontal coordinate of where to locate the left-edge pixel of the virtual RPF's layer, with the left-edge pixel of the master layer set at coordinate 0. This setting should be made in pixel units. A value from 0 to 1919 can be specified. When the virtual RPF is specified as the master layer by VI6_WPF0_SRCRPF.VIR_ACT2, set these bits to 0b.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	VCOORD[12:0]	0000h	RW	Vertical Coordinate of Virtual RPF Location on Master Layer These bits specify the vertical coordinate of where to locate the top-edge pixel of the virtual RPF's layer, with the top-edge pixel of the master layer set at coordinate 0. This setting should be made in pixel units. A value from 0 to 1919 can be specified. When the virtual RPF is specified as the master layer by VI6_WPF0_SRCRPF.VIR_ACT2, set these bits to 0b.

(4) Color Information Register of BRS Input Virtual RPF (LCDC_VSPD_VI6_BRS_VIRRPF_COL)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 390Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	COL_A[7:0]								COL_RCR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COL_GY[7:0]								COL_BCB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	COL_A[7:0]	00h	RW	Fixed α of Virtual RPF These bits set the fixed α value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified.
23 to 16	COL_RCR[7:0]	00h	RW	Fixed R/Cr of Virtual RPF These bits set the fixed R/Cr value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified.
15 to 8	COL_GY[7:0]	00h	RW	Fixed G/Y of Virtual RPF These bits set the fixed G/Y value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified.
7 to 0	COL_BCB[7:0]	00h	RW	Fixed B/Cb of Virtual RPF These bits set the fixed B/Cb value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified.

The transparency information and color information of the single color that is created by the virtual RPF are set in the bits of this register. As described earlier, the color information is set for the YCbCr or RGB color space. The color space to be set in this register depends on the register settings of the environment and other modules to which the BRS is connected by the DPR. Two cases can be considered. Since the α value (COL_A) is transparency information and irrelevant to the concept of color space, the same setting is made for either the YCbCr or RGB color space.

[Case 1: When an input other than the virtual RPF is used]

When the source RPF is connected to any one of the BRS input ports (BRSin0 to BRSin1) other than the virtual RPF and valid data is being supplied, the same color space data as the color space for the BRS inputs should be set in this register as the color space for the virtual RPF's color information. This is based on the restriction of "all BRS inputs must have the same color space", as described in **(1) Concept of DPR Settings** or **(1) BRS Input Control Register (LCDC_VSPD_VI6_BRS_INCTRL)**.

[Case 2: When only the virtual RPF is used]

When only the virtual RPF is used as the source RPF of WPFn, RPFn is not connected to the BRS, as shown in **Figure 9.4-30**. Thus, there is no color space for another input that the color space for the virtual RPF has to follow, as in case 1. This means that the color space of the data output by the BRS is determined by the WPF setting.

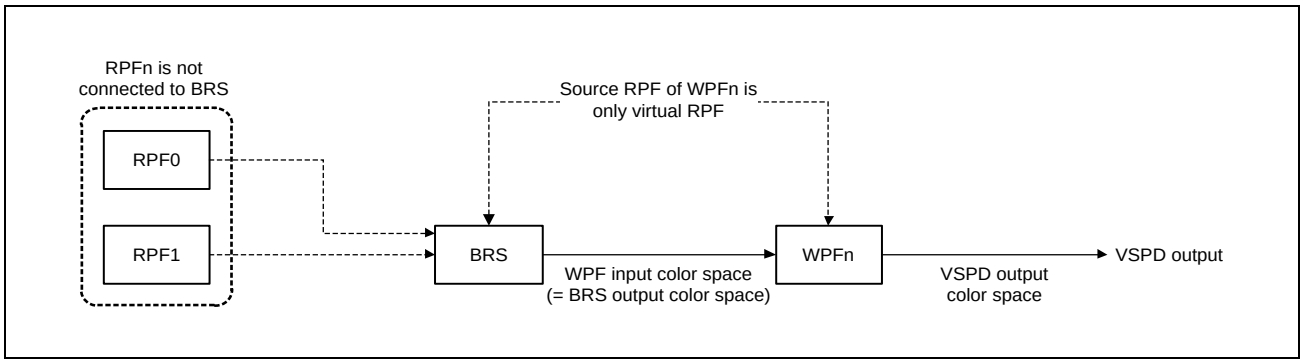


Figure 9.4-30 Relationship between DPR Connection and Color Space When Only Virtual RPF is Used

Figure 9.4-30, the output color space of the VSPD (= output color space of WPFn) is determined by VI6_WPFn_OUTFMT.WRFMT. When bit 6 in VI6_WPFn_OUTFMT.WRFMT (WRFMT [6]) is 0, the color space is RGB, while when it is 1, the color space is YCbCr. Next, the WPF input color space (= BRS output color space) is determined by the relationship between the WPF output color space and VI6_WPFn_OUTFMT.CSC. When VI6_WPFn_OUTFMT.CSC is 0, the WPF output color space and WPF input color space (= BRS output color space) are the same. When VI6_WPFn_OUTFMT.CSC is 1, the WPF output color space and WPF input color space (= BRS output color space) are the opposite. This relationship is summarized in **Table 9.4-25**.

The color space for the virtual RPF's color information should be set in this register according to the "BRS output color space" shown in **Table 9.4-25**.

Table 9.4-25 Relationship between WPF Output Color Space and BRS Output Color Space

VI6_WPFn_OUTFMT Register Bit Settings				BRS Output Color Space (= WPF Input Color Space)
Bit 6 in WRFMT		CSC		
0	(WPF output is RGB)	0	(YCbCr to RGB conversion is disabled)	RGB
0	(WPF output is RGB)	1	(YCbCr to RGB conversion is enabled)	YCbCr
1	(WPF output is YCbCr)	0	(RGB to YCbCr conversion is disabled)	YCbCr
1	(WPF output is YCbCr)	1	(RGB to YCbCr conversion is enabled)	RGB

(5) BRS Control Register A (LCDC_VSPD_VI6_BRSA_CTRL)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 3910h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RBC	-	-	-	-	-	-	-	-	DSTSEL[2:0]			-	SRCSEL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	RW	RW	RW	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CROP[3:0]				AROP[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	RBC	0h	RW	Operation Type of Blending/ROP Unit A Specifies the operation type for blending/ROP unit A shown in **Figure 9.4-29**. 0b: ROP (raster operation) 1b: Blending operation
30 to 23	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22 to 20	DSTSEL[2:0]	0h	RW	Input Selection for DST Side of Blending/ROP Unit A These bits select the input for the DST side of blending/ROP unit A shown in **Figure 9.4-29**. These bits specify the connection between the BRS input port and the DST separately from the setting of connections between other modules and the BRS input port through the DPR. 000b: BRS input 0 (BRSin0) is input to DST 001b: BRS input 1 (BRSin1) is input to DST 100b: Virtual RPF is input to DST 010b, 011b, 101b to 111b: Setting prohibited [Note] The DSTSEL bits for blending/ROP unit B are reserved. The write value should always be 0b.
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	SRCSEL[2:0]	0h	RW	Input Selection for SRC Side of Blending/ROP Unit A These bits select the input for the SRC side of blending/ROP unit A shown in **Figure 9.4-29**. These bits specify the connection between the BRS input port and the SRC separately from the setting of connections between other modules and the BRS input port through the DPR. 000b: BRS input 0 (BRSin0) is input to SRC 001b: BRS input 1 (BRSin1) is input to SRC 100b: Virtual RPF is input to SRC 010b, 011b, 101b to 111b: Setting prohibited
15 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 4	CROP[3:0]	0h	RW	Color Data ROP Operator These bits select the ROP operator of the color data in blending/ROP unit A. Select the opcode for ROP operation from **Table 9.4-26**.
3 to 0	AROP[3:0]	0h	RW	α Data ROP Operator These bits select the ROP operator of the α data in blending/ROP unit A. Select the opcode for ROP operation from **Table 9.4-26**.

Table 9.4-26 ROP Operator of Blending/ROP Unit m (m = A, B)

Opcode	Operator
0000b	NOP(D)
0001b	AND(S & D)
0010b	AND_REVERSE(S & ~D)
0011b	COPY(S)
0100b	AND_INVERTED(~S & D)
0101b	CLEAR(0)
0110b	XOR(S ^ D)
0111b	OR(S | D)
1000b	NOR(~(S | D))
1001b	EQUIV(~(S ^ D))
1010b	INVERT(~D)
1011b	OR_REVERSE(S | ~D)
1100b	COPY_INVERTED(~S)
1101b	OR_INVERTED(~S | D)
1110b	NAND(~(S & D))
1111b	SET(all1)

(6) BRS Blend Control Register A (LCDC_VSPD_VI6_BRSA_BLD)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 3914h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CBES	CCMDX[2:0]			-	CCMDY[2:0]			ABES	ACMDX[2:0]			-	ACMDY[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COEFX[7:0]								COEFY[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CBES	0h	RW	Blending Expression Selection Selects the blending expression of the color data in the BRS (VI6_BRSA_CTRL.RBC = 1b). Blending coefficients are specified by the CCMDX and CCMDY bits. 0b: CCMDX × (DST color data) + CCMDY × (SRC color data) 1b: CCMDX × (DST color data) - CCMDY × (SRC color data)
30 to 28	CCMDX[2:0]	0h	RW	Blending Coefficient X Selection These bits specify coefficient X used in the blending expression determined by the CBES bit. 000b: DST α data is used as blending coefficient X 001b: 255 - (DST α data) is used as blending coefficient X 010b: SRC α data is used as blending coefficient X 011b: 255 - (SRC α data) is used as blending coefficient X 100b: Fixed α value 0 (COEFX setting)
27	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
26 to 24	CCMDY[2:0]	0h	RW	Blending Coefficient Y Selection These bits specify coefficient Y used in the blending expression determined by the CBES bit. 000b: DST α data is used as blending coefficient Y 001b: 255 - (DST α data) is used as blending coefficient Y 010b: SRC α data is used as blending coefficient Y 011b: 255 - (SRC α data) is used as blending coefficient Y 100b: Fixed α value 1 (COEFY setting)
23	ABES	0h	RW	Blending α Creation Expression Specifies the expression for creating α data after blending by blending/ROP unit m (m = A, B). α creation coefficients are specified by the ACMDX and ACMDY bits. 0b: ACMDX × (DST α data) + ACMDY × (SRC α data) 1b: ACMDX × (DST α data) - ACMDY × (SRC α data)
22 to 20	ACMDX[2:0]	0h	RW	α Creation Coefficient X These bits specify α creation coefficient X used in the α creation expression determined by the ABES bit. 000b: (α creation coefficient X) = (DST α data) 001b: (α creation coefficient X) = 255 - (DST α data) 010b: (α creation coefficient X) = (SRC α data) 011b: (α creation coefficient X) = 255 - (SRC α data) 100b: (α creation coefficient X) = Fixed α value 0 (COEFX setting) 101b to 111b: Setting prohibited
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
18 to 16	ACMDY[2:0]	0h	RW	α Creation Coefficient Y These bits specify α creation coefficient Y used in the α creation expression determined by the ABES bit. 000b: (α creation coefficient Y) = (DST α data) 001b: (α creation coefficient Y) = 255 - (DST α data) 010b: (α creation coefficient Y) = (SRC α data) 011b: (α creation coefficient Y) = 255 - (SRC α data) 100b: (α creation coefficient Y) = Fixed α value 1 (COEFY setting) 101b to 111b: Setting prohibited
15 to 8	COEFX[7:0]	00h	RW	Fixed α Value 0 These bits specify fixed α value 0 used when the CCMDX or ACMDX bits are set to 100b. A value from 00h to FFh can be specified.
7 to 0	COEFY[7:0]	00h	RW	Fixed α Value 1 These bits specify fixed α value 1 used when the CCMDY or ACMDY bits are set to 100b. A value from 00h to FFh can be specified.

(7) BRS Control Register B (LCDC_VSPD_VI6_BRSB_CTRL)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 3918h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RBC	-	-	-	-	-	-	-	-	-	-	-	-	SRCSEL[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CROP[3:0]				AROP[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	RBC	0h	RW	Operation Type of Blending/ROP Unit B Specifies the operation type for blending/ROP unit B shown in **Figure 9.4-29**. 0b: ROP (raster operation) 1b: Blending operation
30 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18 to 16	SRCSEL[2:0]	0h	RW	Input Selection for SRC Side of Blending/ROP Unit B These bits select the input for the SRC side of blending/ROP unit A shown in **Figure 9.4-29**. These bits specify the connection between the BRS input port and the SRC separately from the setting of connections between other modules and the BRS input port through the DPR. 000b: BRS input 0 (BRSin0) is input to SRC 001b, 010b, 011b, 100b, 101b to 111b: Setting prohibited
7 to 4	CROP[3:0]	0h	RW	Color Data ROP Operator These bits select the ROP operator of the color data in blending/ROP unit B. Select the opcode for ROP operation from **Table 9.4-26**.
3 to 0	AROP[3:0]	0h	RW	α Data ROP Operator These bits select the ROP operator of the α data in blending/ROP unit B. Select the opcode for ROP operation from **Table 9.4-26**.

(8) BRS Blend Control Register B (LCDC_VSPD_VI6_BRSB_BLD)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 391Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CBES	CCMDX[2:0]			-	CCMDY[2:0]			ABES	ACMDX[2:0]			-	ACMDY[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COEFX[7:0]								COEFY[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	CBES	0h	RW	Blending Expression Selection Selects the blending expression of the color data in the BRS (VI6_BRSB_CTRL.RBC = 1). Blending coefficients are specified by the CCMDX and CCMDY bits. 0b: CCMDX × (DST color data) + CCMDY × (SRC color data) 1b: CCMDX × (DST color data) - CCMDY × (SRC color data)
30 to 28	CCMDX[2:0]	0h	RW	Blending Coefficient X Selection These bits specify coefficient X used in the blending expression determined by the CBES bit. 000b: DST α data is used as blending coefficient X 001b: 255 - (DST α data) is used as blending coefficient X 010b: SRC α data is used as blending coefficient X 011b: 255 - (SRC α data) is used as blending coefficient X 100b: Fixed α value 0 (COEFX setting)
27	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
26 to 24	CCMDY[2:0]	0h	RW	Blending Coefficient Y Selection These bits specify coefficient Y used in the blending expression determined by the CBES bit. 000b: DST α data is used as blending coefficient Y 001b: 255 - (DST α data) is used as blending coefficient Y 010b: SRC α data is used as blending coefficient Y 011b: 255 - (SRC α data) is used as blending coefficient Y 100b: Fixed α value 1 (COEFY setting)
23	ABES	0h	RW	Blending α Creation Expression Specifies the expression for creating α data after blending by blending/ROP unit m (m = A, B). α creation coefficients are specified by the ACMDX and ACMDY bits. 0b: ACMDX × (DST α data) + ACMDY × (SRC α data) 1b: ACMDX × (DST α data) - ACMDY × (SRC α data)
22 to 20	ACMDX[2:0]	0h	RW	α Creation Coefficient X These bits specify α creation coefficient X used in the α creation expression determined by the ABES bit. 000b: (α creation coefficient X) = (DST α data) 001b: (α creation coefficient X) = 255 - (DST α data) 010b: (α creation coefficient X) = (SRC α data) 011b: (α creation coefficient X) = 255 - (SRC α data) 100b: (α creation coefficient X) = Fixed α value 0 (COEFX setting) 101b to 111b: Setting prohibited
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
18 to 16	ACMDY[2:0]	0h	RW	α Creation Coefficient Y These bits specify α creation coefficient Y used in the α creation expression determined by the ABES bit. 000b: (α creation coefficient Y) = (DST α data) 001b: (α creation coefficient Y) = 255 - (DST α data) 010b: (α creation coefficient Y) = (SRC α data) 011b: (α creation coefficient Y) = 255 - (SRC α data) 100b: (α creation coefficient Y) = Fixed α value 1 (COEFY setting) 101b to 111b: Setting prohibited
15 to 8	COEFX[7:0]	00h	RW	Fixed α Value 0 These bits specify fixed α value 0 used when the CCMDX or ACMDX bits are set to 100b. A value from 00h to FFh can be specified.
7 to 0	COEFY[7:0]	00h	RW	Fixed α Value 1 These bits specify fixed α value 1 used when the CCMDY or ACMDY bits are set to 100b. A value from 00h to FFh can be specified.

9.4.3.2.11 LIF Control Registers

(1) LIF0 Control Register (LCDC_VSPD_VI6_LIF0_CTRL)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 3B00h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	OBTH [11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	PADL	-	-	-	CFMT	-	-	REQSEL	LIF_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27 to 16	OBTH [11:0]	000h	RW	Buffer Threshold for Start Ready Notification to Display Module Always specify 1500, when LIF_EN is set to 1b.
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	PADL	0h	RW	Enable/Disable of padding with dummy lines to output-image to the DU. Enables or disables of padding with dummy lines to output-image to the DU. 0b: Padding with dummy lines to output image to DU is disabled. 1b: Padding with dummy lines to output image to DU is enabled.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	CFMT	0h	RW	Chroma Format This bit selects the output format from the LIF module to the display module. When RGB format is used, this bit shall be set to 0b. 0b: YCbCr444 or RGB Format 1b: YCbCr422 Format [Note] The DU cannot receive YCbCr422 format. Therefore, when CFMT is 1, YCbCr444 data which information contents is equal to YCbCr422 is output to DU as shown in **Figure 9.4-31**.
3, 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	REQSEL	0h	RW	External Display Module Selection 0b: This value is setting prohibited when 1b is set to LIF_EN 1b: DU is selected as the destination external display module.
0	LIF_EN	0h	RW	Enable/Disable of Data Output to External Display Module Enables or disables data output from the LIF to the external display module (DU). 0b: Data output to the external display module is disabled. 1b: Data output to the external display module is enabled.

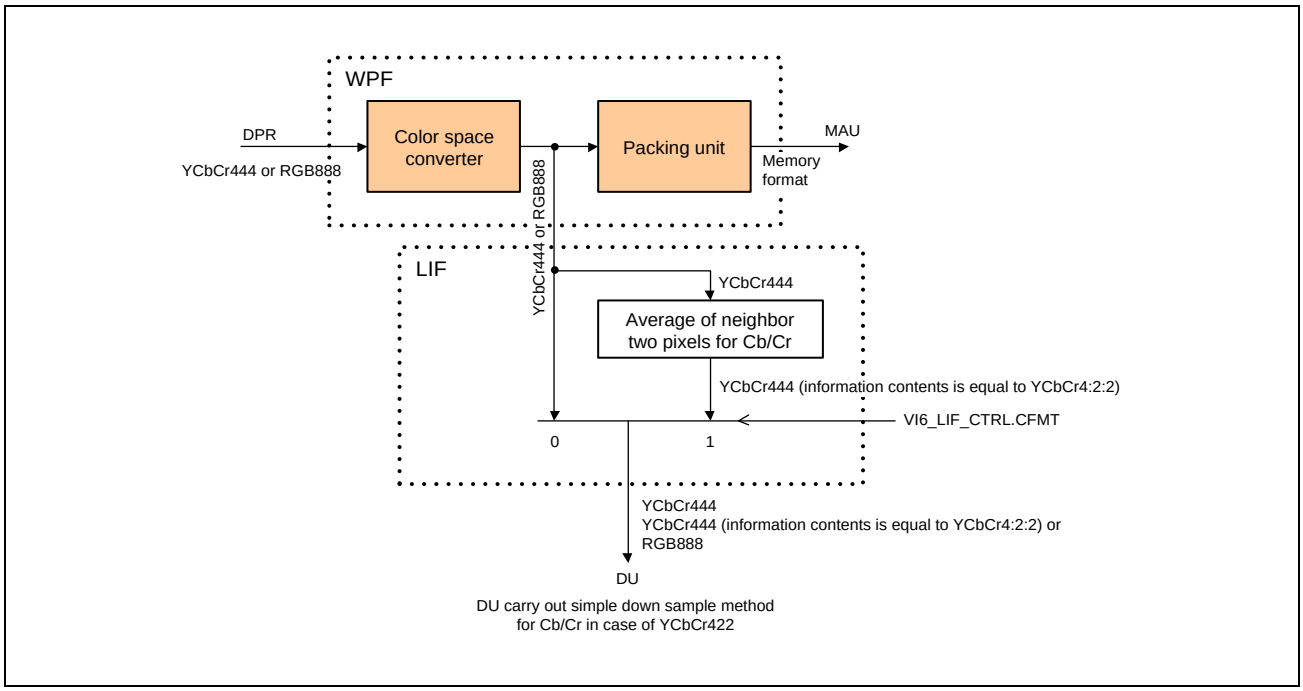


Figure 9.4-31 Data flow between LIF and DU

(2) LIF0 Clock Stop Buffer Control Register (LCDC_VI6_LIF0_CSBTH)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 3B04h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	HBTH[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	LBTH[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27 to 16	HBTH[11:0]	000h	RW	Buffer Threshold for Clock Stop in Dynamic Clock Control Set HBTH[11:0] = 0 (fixed value)
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	LBTH[11:0]	000h	RW	Buffer Threshold for Clock Start in Dynamic Clock Control Set LBTH[11:0] = 0 (fixed value)

(3) LIF0 Buffer Attribute Register (LCDC_VI6_LIF0_LBA)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 3B0Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LBA0	-	-	-	LBA1[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	LBA2[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	LBA0	0h	RW	LIF Buffer Attribute Register0 Always specify 1b, when LIF_EN is set to 1b
30 to 28	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
27 to 16	LBA1[11:0]	000h	RW	LIF Buffer Attribute Register1 Always specify 1536, when LIF_EN is set to 1b
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	LBA2[11:0]	000h	R	LIF Buffer Attribute Register2 These bits are internal status for purpose of h/w debugging.

(4) LIF0 Padding Line Cycle Register (LCDC_VI6_LIF0_PADLN_CYC)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 3B30h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	PADLN_CYC[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
5 to 0	PADLN_CYC [5:0]	00h	RW	Cycle of padding pattern. These bits specify the minimum number of cycles of the padding pattern of valid lines and dummy lines. For example, if PADLN_CYC[5:0] is 4, lower 4 bits of PADLN_PATTERN[3:0] indicates padding pattern. A value from 1 to 32 can be specified.

(5) LIF0 Padding Line Pattern Register (LCDC_VI6_LIF0_PADLN_PT)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 3B34h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PADLN_PATTERN[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PADLN_PATTERN[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PADLN_PATTERN[31:0]	0000_0000h	RW	Pattern of padding with dummy lines These bits specify the padding pattern of valid lines and dummy lines. In each bit, 0b indicates an active line, 1b indicates a dummy line. It starts from the lower bit (Bit 0). First line should be valid line (Always set 0 to PADLN_PATTERN[0]).

(6) LIF0 Padding Line Value Register (LCDC_VI6_LIF0_PADLN_VAL)

Access Size : 32 bits
Address : <LCDC_vspd_base> + 3B38h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PADA[7:0]								PADR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PADG[7:0]								PADB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	PADA[7:0]	00h	RW	Padding data of EDC code for dummy line.
23 to 16	PADR[7:0]	00h	RW	Padding data of R/Cr for dummy line.
15 to 8	PADG[7:0]	00h	RW	Padding data of G/Y for dummy line.
7 to 0	PADB[7:0]	00h	RW	Padding data of B/Cb for dummy line.

Note: Set the register to 24FF_FFFFh when the output format is YUV422.
Otherwise, set it 2EFF_FFFFh.

(7) LIF0 Padding Line Size Register (LCDC_VI6_LIF0_PADLN_SIZE)

Access Size : 32 bits

Address : <LCDC_vspd_base> + 3B3Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PADLN_VSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	PADLN_VSIZE [12:0]	0000h	RW	Vertical Size of LIF output in case of enabling Padding line When VI6_LIF0_CTRL.PADL is 1b, dummy lines are included in the output data from LIF, so set the LIF output VSIZE including dummy lines to these bits. If the number of output lines does not reach the value of PADLN_VSIZE[12:0] even if all valid lines are output according to padding pattern, dummy lines are inserted until the number of output lines reaches PADLN_VSIZE[12:0]. - Output vsize of LIF is same with input vsize to LIF when VI6_LIF0_CTRL.PADL is 0b - Output vsize of LIF is PADLN_VSIZE[12:0] when VI6_LIF0_CTRL.PADL is 1b

9.4.4 DU Registers

The DU contains the registers which can directly be read and written to by the host CPU. The host can access any long word (32 bits), word (16 bits), and byte (8 bits) in the registers. Therefore, read-modify-write is needed to change partial bits not more than 8 bits in the 32-bit register.

Never access the registers when the DU is in the module standby mode (with the operation clock placed in hold status).

Table 9.4-27 DU Register Base Address

Base Register Name	LCDC Ch.	Base Address
<LCDC0_du_base>	LCDC0	0_1646_0000h (5646_0000h*1, 4646_0000h*2)
<LCDC1_du_base>	LCDC1	0_1649_0000h (5649_0000h*1, 4649_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

9.4.4.1 List of DU Registers

The registers are listed below.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
DU Module Control Register 0	LCDC_DU_MCR0	0000_0000h	0000h	32, 16, 8
DU Module Status Register 0	LCDC_DU_MSR0	0054_0000h	0004h	32, 16, 8
DU Module Status Register 1	LCDC_DU_MSR1	0000_0000h	0008h	32, 16, 8
DU Interrupt Mask Register 0	LCDC_DU_IMR0	0000_0000h	000Ch	32, 16, 8
DU Display I/F Timing Register 0	LCDC_DU_DITR0	0003_0300h	0010h	32, 16, 8
DU Display I/F Timing Register 1	LCDC_DU_DITR1	0000_0000h	0014h	32, 16, 8
DU Display I/F Timing Register 2	LCDC_DU_DITR2	0000_0000h	0018h	32, 16, 8
DU Display I/F Timing Register 3	LCDC_DU_DITR3	0000_0000h	001Ch	32, 16, 8
DU Display I/F Timing Register 4	LCDC_DU_DITR4	0000_0000h	0020h	32, 16, 8
Reserve	-	-	0024h to 003Fh	-
DU Module Control Register 1	LCDC_DU_MCR1	0000_0000h	0040h	32, 16, 8
Reserve	-	-	0044h to 004Bh	-
DU PBUF Control Register 0	LCDC_DU_PBCR0	0000_001Fh	004Ch	32, 16, 8
DU PBUF Control Register 1	LCDC_DU_PBCR1	0000_0000h	0050h	32, 16, 8
DU PBUF Control Register 2	LCDC_DU_PBCR2	0000_0000h	0054h	32, 16, 8

9.4.4.2 DU Register Description

The prefix (LCDC_) of the register names is omitted in this and subsequent sections.

9.4.4.2.1 DU Module Control Register 0 (LCDC_DU_MCR0)

Access Size : 32, 16, 8 bits
Address : <LCDC_du_base> + 0000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PB_CLR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	DI_EN	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	PB_CLR	0h	RW	Clear PBUF pointers. This parameter must be changed while VSPD is not running and DU_MSR0.ST_DIF_BSY is '0b'. Writing '1b' to this parameter starts clearing the PBUF pointers. After clearing the PBUF pointers is completed, DU_MSR0.ST_PB_WINIT and DU_MSR0.ST_PB_RINIT are read as 1b, this parameter must be written back to '0b'.
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	DI_EN	0h	RW	Display enable. 1b: Enabled 0b: Disabled
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.

9.4.4.2.2 DU Module Status Register 0 (LCDC_DU_MSR0)

Access Size : 32, 16, 8 bits

Address : <LCDC_du_base> + 0004h

Initial Value : 0054_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	ST_PB_RINIT	ST_PB_RUF	ST_PB_REMPTY	-	ST_PB_WINIT	-	ST_PB_WFULL
Initial Value	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	ST_DI_BSY	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	ST_PB_RINIT	1h	R	PBUF FIFO read pointer cleared status. 1b: Cleared (initial value) 0b: Not cleared
21	ST_PB_RUF	0h	R	PBUF read underflow status. This parameter clears when PBUF pointers are cleared. 1b: Occur PBUF underflow 0b: Not occur PBUF underflow
20	ST_PB_REMPTY	1h	R	PBUF read empty status. 1b: Empty 0b: Not empty
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18	ST_PB_WINIT	1h	R	PBUF FIFO write pointer cleared status. 1b: Cleared (initial value) 0b: Not cleared
17	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	ST_PB_WFULL	0h	R	PBUF write full status. 1b: Full 0b: Not full
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	ST_DI_BSY	0h	R	Display I/F status. 1b: Busy 0b: Idle
7 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.4.4.2.3 DU Module Status Register 1 (LCDC_DU_MSR1)

Access Size : 32, 16, 8 bits

Address : <LCDC_du_base> + 0008h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	UF_VACT[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	UF_HACT[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	UF_VACT[12:0]	0000h	R	Vactive counter when the PBUF underflow occurs. This parameter is value latched the down counter starting from VACTIVE. This parameter is cleared when DU_MSR0.ST_PB_RUF is cleared.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	UF_HACT[12:0]	0000h	R	Hactive counter when the PBUF underflow occurs. This parameter is value latched the down counter starting from HACTIVE. This parameter is cleared when DU_MSR0.ST_PB_RUF is cleared.

9.4.4.2.4 DU Interrupt Mask Register 0 (LCDC_DU_IMR0)

Access Size : 32, 16, 8 bits

Address : <LCDC_du_base> + 000Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	IM_PB_RUF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	IM_PB_RUF	0h	RW	Mask PBUF read underflow interrupt. 1b: Mask interrupt 0b: Not mask interrupt

9.4.4.2.5 DU Display I/F Timing Register 0 (LCDC_DU_DITR0)

Access Size : 32, 16, 8 bits

Address : <LCDC_du_base> + 0010h

Initial Value : 0003_0300h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	ODDP_EN	-	-	-	-	-	-	HSPOL	VSPOL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	DEMD[1:0]		-	-	-	-	-	-	-	DPI_CLKMD
Initial Value	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	ODDP_EN	0h	RW	LVDS mode. This bit is invalid using DSI and/or Parallel I/F. 0: Single channel LVDS 1: Dual channel LVDS
23 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17	HSPOL	1h	RW	hsync polarity. 1b: High active 0b: Low active
16	VSPOL	1h	RW	vsync polarity. 1b: High active 0b: Low active
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9, 8	DEMD[1:0]	3h	RW	de output mode. 00b: Reserved 01b: Fixed to Low 10b: Reserved 11b: Data enable (High active)
7 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	DPI_CLKMD	0h	RW	Display parallel interface clock mode. 0b: Through output 1b: Inversion output

9.4.4.2.6 DU Display I/F Timing Register 1 (LCDC_DU_DITR1)

Access Size : 32, 16, 8 bits
Address : <LCDC_du_base> + 0014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	VACTIVE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	VSA[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	VACTIVE[12:0]	0000h	RW	The number of lines in the Vactive period. This parameter should be set from 1 to 1920.
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	VSA[11:0]	000h	RW	The number of lines in the Vsync period. This parameter should be set from 0 to 4095, and DU_DITR1.VSA + DU_DITR2.VBP should be set more than 0. If set to 0, vsync is not asserted to DU_DITR0.VSPOL.

Figure 9.4-32 shows the relationship between the parameters and the video output timings.

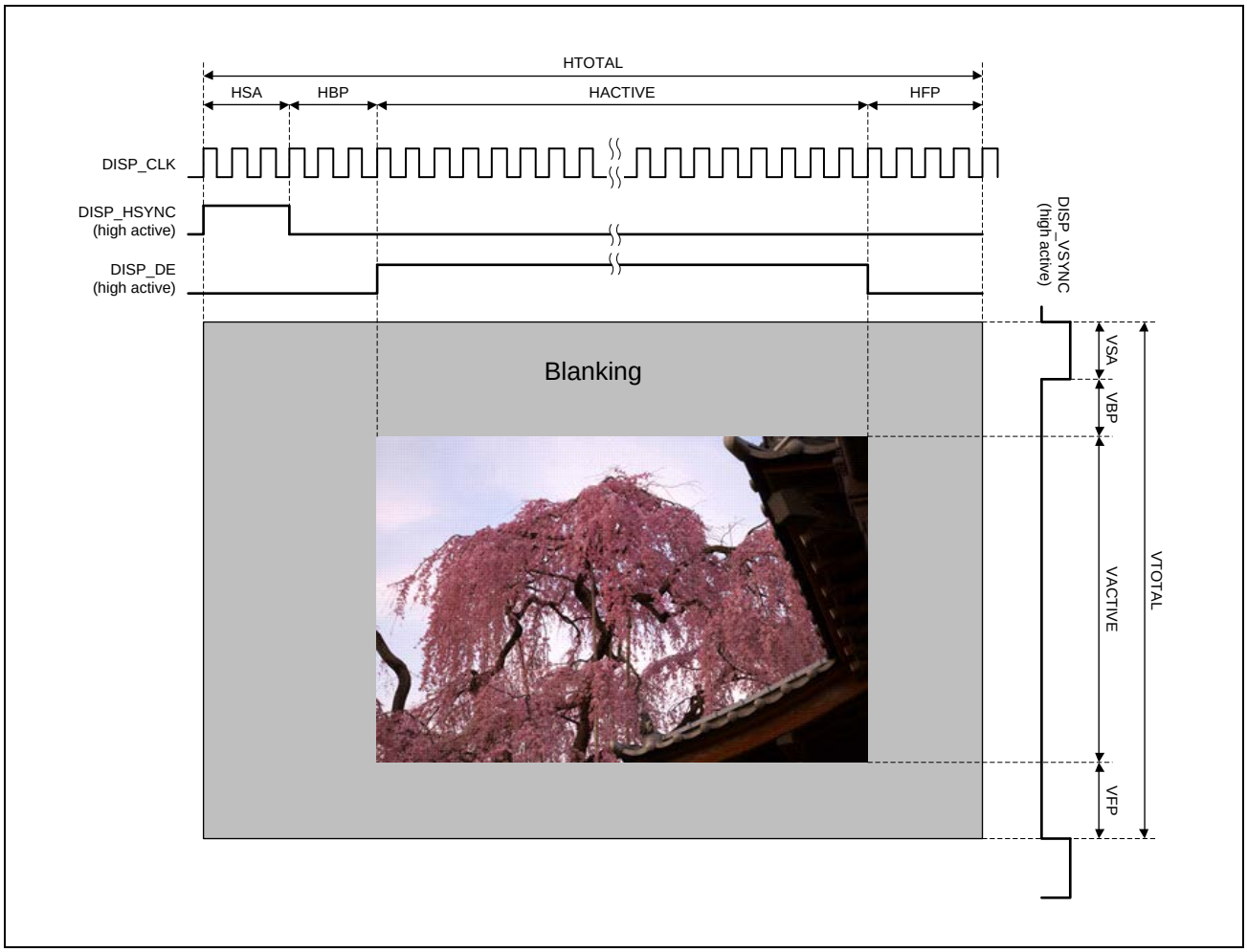


Figure 9.4-32 Video Output Timings

9.4.4.2.7 DU Display I/F Timing Register 2 (LCDC_DU_DITR2)

Access Size : 32, 16, 8 bits

Address : <LCDC_du_base> + 0018h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	VFP[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VBP[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	VFP[12:0]	0000h	RW	The number of lines in the Vfront period. This parameter should be set from 0 to 8191.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	VBP[12:0]	0000h	RW	The number of lines in the Vback period. This parameter should be set from 0 to 8191, and DU_DITR1.VSA + DU_DITR2.VBP should be set more than 0.

9.4.4.2.8 DU Display I/F Timing Register 3 (LCDC_DU_DITR3)

Access Size : 32, 16, 8 bits

Address : <LCDC_du_base> + 001Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	HACTIVE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	HSA[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	HACTIVE[12:0]	0000h	RW	The number of cycles (pixels) in the Hactive period. This parameter should be set from 1 to 1920.
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	HSA[11:0]	000h	RW	The number of cycles in the Hsync period. This parameter should be set from 0 to 4095, and DU_DITR3.HSA + DU_DITR4.HBP + DU_DITR4.HFP (=Hblank) should be set 3 or more. If set to 0, hsync is not asserted to DU_DITR0.HSPOL.

9.4.4.2.9 DU Display I/F Timing Register 4 (LCDC_DU_DITR4)

Access Size : 32, 16, 8 bits

Address : <LCDC_du_base> + 0020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	HFP[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	HBP[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	HFP[12:0]	0000h	RW	The number of cycles in the Hfront period. This parameter should be set from 1 to 8191, and DU_DITR3.HSA + DU_DITR4.HBP + DU_DITR4.HFP (=Hblank) should be set 3 or more.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	HBP[12:0]	0000h	RW	The number of cycles in the Hback period. This parameter should be set from 0 to 8191, and DU_DITR3.HSA + DU_DITR4.HBP + DU_DITR4.HFP (=Hblank) should be set 3 or more.

9.4.4.2.10 DU Module Control Register 1 (LCDC_DU_MCR1)

Access Size : 32, 16, 8 bits

Address : <LCDC_du_base> + 0040h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PB_AUTOCLR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OPMD[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	PB_AUTOCLR	0h	RW	PBUF pointers auto clear enable. Specify timing to clear the PBUF pointers. 0b: Not clear until DU_MCR0.PB_CLR is asserted 1b: Clear during blanking (according to DU_MCR1.OPMD)
15 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	-	0h	RW	Reserved Whenever it is read, 0b is read. The written value should always be 0b.
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	OPMD[1:0]	0h	RW	Operation mode. Specify behavior of LIFC and PBUF during Vfront and Hfront period. This parameter must be changed while VSPD is not running. 00b: After frame end, LIFC wait, and PBUF pointers are cleared if DU_MCR1.PB_AUTOCLR is set to '1b'. 01b: After line end, LIFC wait, and PBUF pointers are cleared if DU_MCR1.PB_AUTOCLR is set to '1'. 10b: Reserved 11b: LIFC always request to VSPD. PBUF pointers are not cleared until DU_MCR0.PB_CLR is set to '1b'.

9.4.4.2.11 DU PBUF Control Register 0 (LCDC_DU_PBCR0)

Access Size : 32, 16, 8 bits

Address : <LCDC_du_base> + 004Ch

Initial Value : 0000_001Fh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	PB_DEP[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4 to 0	PB_DEP[4:0]	1Fh	RW	Valid PBUF depth. This parameter should be set 1Fh. This parameter should be set $((2^n) - 1)$.

9.4.4.2.12 DU PBUF Control Register 1 (LCDC_DU_PBCR1)

Access Size : 32, 16, 8 bits

Address : <LCDC_du_base> + 0050h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PB_RUFOP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	PB_RUFOP	0h	RW	Specify behavior when the PBUF underflow occurs. 0b: Continue outputting invalid data 1b: Output DU_PBCR2.PB_RUFDAT

9.4.4.2.13 DU PBUF Control Register 2 (LCDC_DU_PBCR2)

Access Size : 32, 16, 8 bits

Address : <LCDC_du_base> + 0054h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	PB_RUFDAT[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PB_RUFDAT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 0	PB_RUFDAT [23:0]	00_0000h	RW	Read data when the PBUF underflow occurs.

9.4.5 Operation

9.4.5.1 Operation Control Setting

(a) To start image process

FCPVD registers do not need to be set when starting an image process. Therefore, refer **9.4.5.4 Concept of VSPD Operation Starting and Stopping** for the guidance of starting an image process.

9.4.5.2 Reset Operation

This unit can be reset by hardware reset. The hardware reset is controlled by LCDC_n_RESET_N (n = 0, 1) of CPG, and this reset initializes the whole this unit immediately.

9.4.5.3 Registers to set fixed value

Set fixed value to following registers in any case.

[1] Always set 0000 0808h to VI6_CLK_DCSWT

[2] Always set 256 to VI6_DL_CTRL.AR_WAIT [7:0] in case of using display list.

[3] Always set 2 to VI6_DL_EXT_CTRLn.POLINT [5:0] in case of using extended display list of WPFn.

[4] Always set 1 to VI6_DL_EXT_CTRLn.DLPRI in case of using extended display list of WPFn.

[5] Always set 0 to VI6_DL_EXT_CTRLn.EXPRI in case of using extended display list of WPFn.

[6] Always set 0000 0500h toVI6_DPR_WPFn_FPORCH (n = 0, 1) in case of using WPFn.

[7] Always set 1500 to VI6_LIF0_CTRL.OBTH [11:0] in case of using LIF (VI6_LIF0_CTRL.LIF_EN = 1).

[8] Always set 8600 0000h to VI6_LIF0_LBA in case of using LIF0 (VI6_LIF0_CTRL.LIF_EN = 1).

[9] Always set 1 to VI6_LIFn_CTRL.REQSEL in case of using LIFn (VI6_LIFn_CTRL.LIF_EN = 1).

[10] Always set 0 to VI6_LIFn_CSBTH in case of using LIFn (VI6_LIFn_CTRL.LIF_EN = 1).

[11] Always set 1Fh to DU_PBCR0 in case of using DU (DU_MCR0.DI_EN = 1).

9.4.5.4 Concept of VSPD Operation Starting and Stopping

The VSPD provides one channel of image processing. Each channel is started by setting the corresponding start register. Here, starting a processing channel means starting one of WPF0, which are output modules of the VSPD. Use the start registers shown in **Table 9.4-28** to start WPF modules.

After a WPF module is started and specified processing is completed, the corresponding channel stops operation and notifies the end of processing through an end interrupt. End interrupts are generated through the end interrupt source registers shown in **Table 9.4-28**; clearing a source register cancels the corresponding interrupt signal. Each of the operating status registers shown in the table indicates the busy state after the corresponding channel is started through the start register until processing is completed and operation stops. **Figure 9.4-34** shows these operation timings.

Table 9.4-28 Target Module and Corresponding Registers for Starting and Stopping Operation

Target Module	Start Register	End Interrupt Source Register	Operating Status Register
WPF0	VI6_CMD0.STRCMD	VI6_WPF0_IRQ_STA. FRE	VI6_STATUS.SYS0_ACT

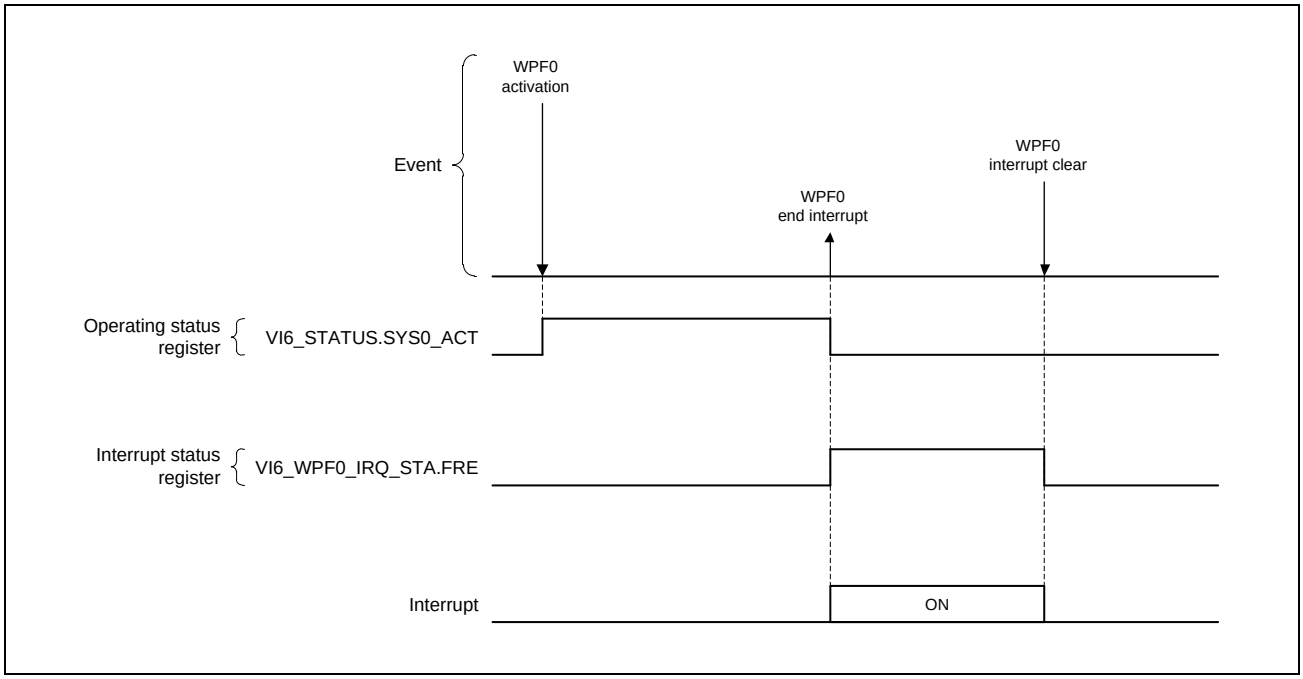


Figure 9.4-33 VSPD Startup and Status of Each Register and Interrupt

The following describes the operating states (operating or stopped) of VSPD internal modules. As described in **9.4.3.2 VSPD Register Description**, the VSPD has several image processing modules and the connections between modules are determined by the DPR. Accordingly, the operating state of a module is the same as that of the target WPF for that module. For example, when the target WPF for the BRS is WPF0, the BRS operating state is the same as that of WPF0; that is, the BRS operating state is indicated by the VI6_STATUS.SYS0_ACT as shown in **Table 9.4-28** and the status change timing is shown as VI6_STATUS. SYS0_ACT in **Figure 9.4-33**. Likewise, the operating states of all modules connected to WPF0 is indicated by VI6_STATUS. SYS0_ACT.

Connections should be changed through the DPR-related registers (described later) while all modules to be affected by any change in connections are stopped. If connections through the DPR are changed during operation, the VSPD will hang.

Figure 9.4-34 shows relation between start reservation and operating status. When start is reserved while VSPD is operating, the reservation is accepted after VSPD status is moved from “operating” into “idle” as shown in **Figure 9.4-34**.

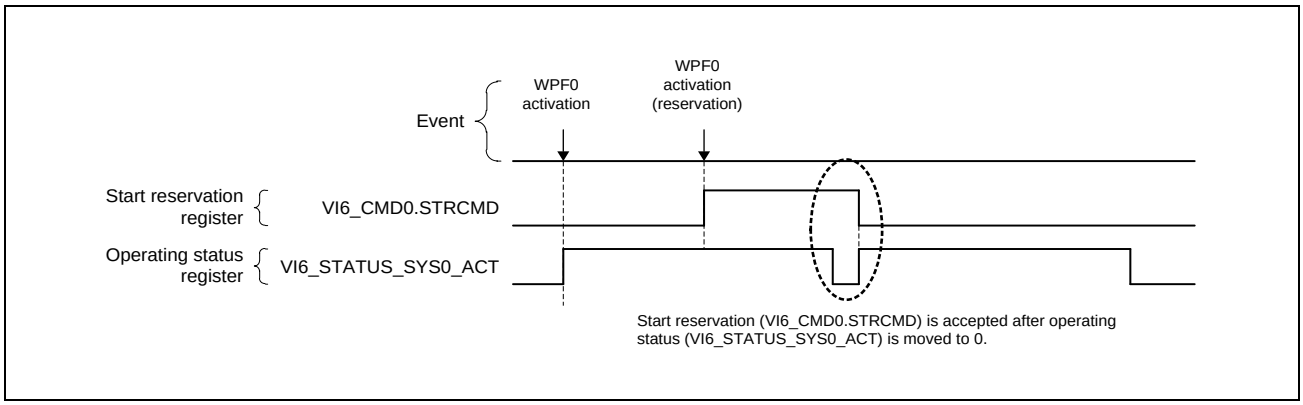


Figure 9.4-34 VSPD Start Reservation and Operating Status

9.4.5.5 Display List

9.4.5.5.1 Functional description

The VSPD provides the display list function. As a display list, the VSPD automatically downloads the register settings except for the control registers (**9.4.3.2.4 General Control Registers** and **9.4.3.2.5 Display List Control Registers**) from external memory and stores the settings in the VSPD registers. This function is advantageous in that the interrupt processing or register setting modification processing can be executed without CPU intervention during multiple-frame processing because the register settings used for VSPD processing are prepared in advance in external memory such as SDRAM.

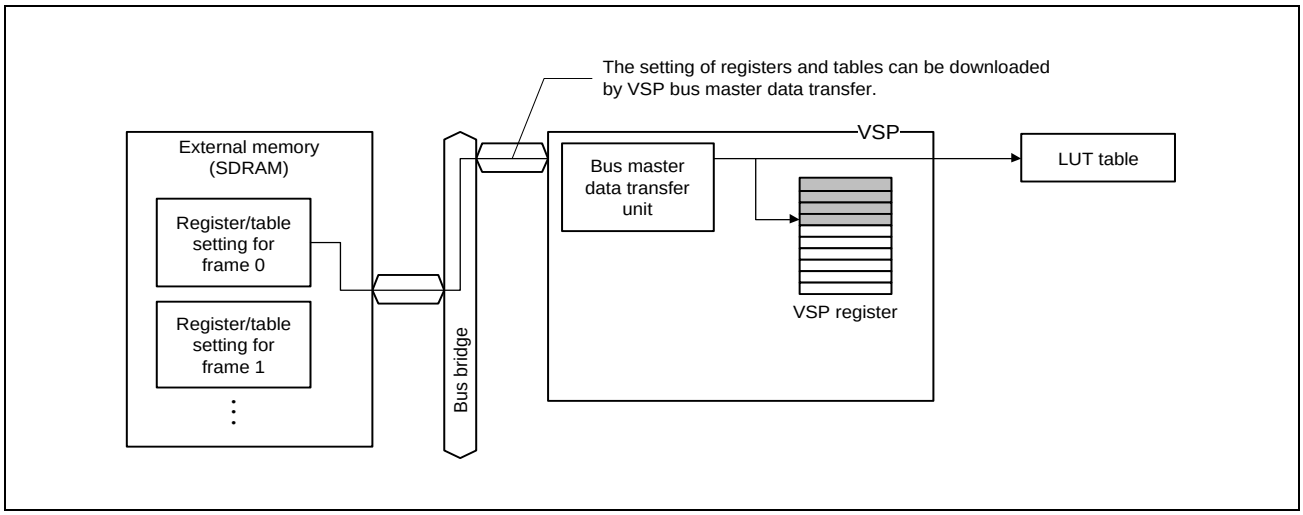


Figure 9.4-35 Concept of Display List

To use display lists, specify the external memory addresses to the display list control registers described in **9.4.3.2.5 Display List Control Registers**. The register settings or various types of information should be stored in external memory in the format described in **9.4.5.5.2 Normal display list mode**. **Figure 9.4-36** shows the difference between VSPD operation through normal register settings and through display lists.

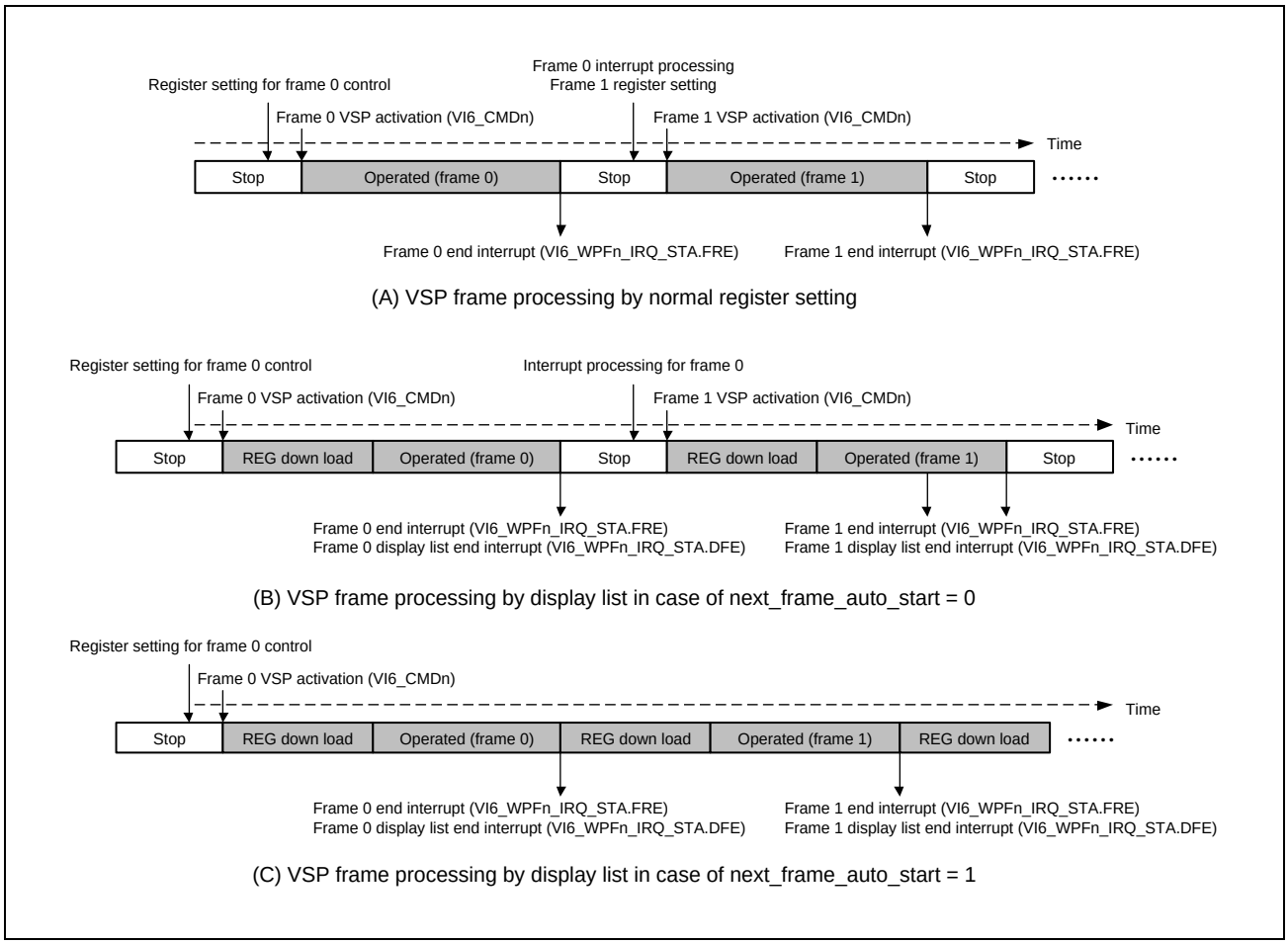


Figure 9.4-36 Comparison of VSPD Operation between Normal Register Settings and Display Lists

As shown in **Figure 9.4-36** (A), in the VSPD processing through normal register settings, all registers should be set up before the VSPD is started for each frame. After the VSPD processing is completed, the VSPD outputs a frame end interrupt (VI6_WPFn_IRQ_STA.FRE). This method requires a certain amount of time for register settings or interrupt processing by the CPU between frames. In contrast, when display lists are used, the VSPD automatically downloads register settings from external memory as shown in **Figure 9.4-36** (B) and (C), which reduces the load on the CPU between frames.

Figure 9.4-36 (B) shows the display list usage where the VSPD stops at the end of every frame; only the VSPD start processing for each frame is done by the CPU. This is suitable for the cases when the CPU controls synchronization of frame processing in frame buffer management or when the amount of register values or table data to be set in the VSPD is large. In the case shown in **Figure 9.4-36** (C), as soon as the frame processing ends, the VSPD automatically begins next frame operation and starts downloading new register settings. This is the fastest operation using display lists.

Table 9.4-29 shows the modes of the display list and the supported functions for each mode. The detail of each mode is described in the following sections.

Table 9.4-29 Display list mode and supported functions

Mode	Extended Display List	Continuous Frames
Normal Display List Mode	Supported	Controlled by “next frame auto start” in the header of the display list.
Header-less Display List Mode	Not Supported	Controlled by VI6_DL_CTRL.CFM0 bit.

9.4.5.5.2 Normal display list mode

The VSPD display lists include control information as well as simple register settings in order to control multiple-frame processing in an optimum way for each application. **Figure 9.4-37** shows the display list structure.

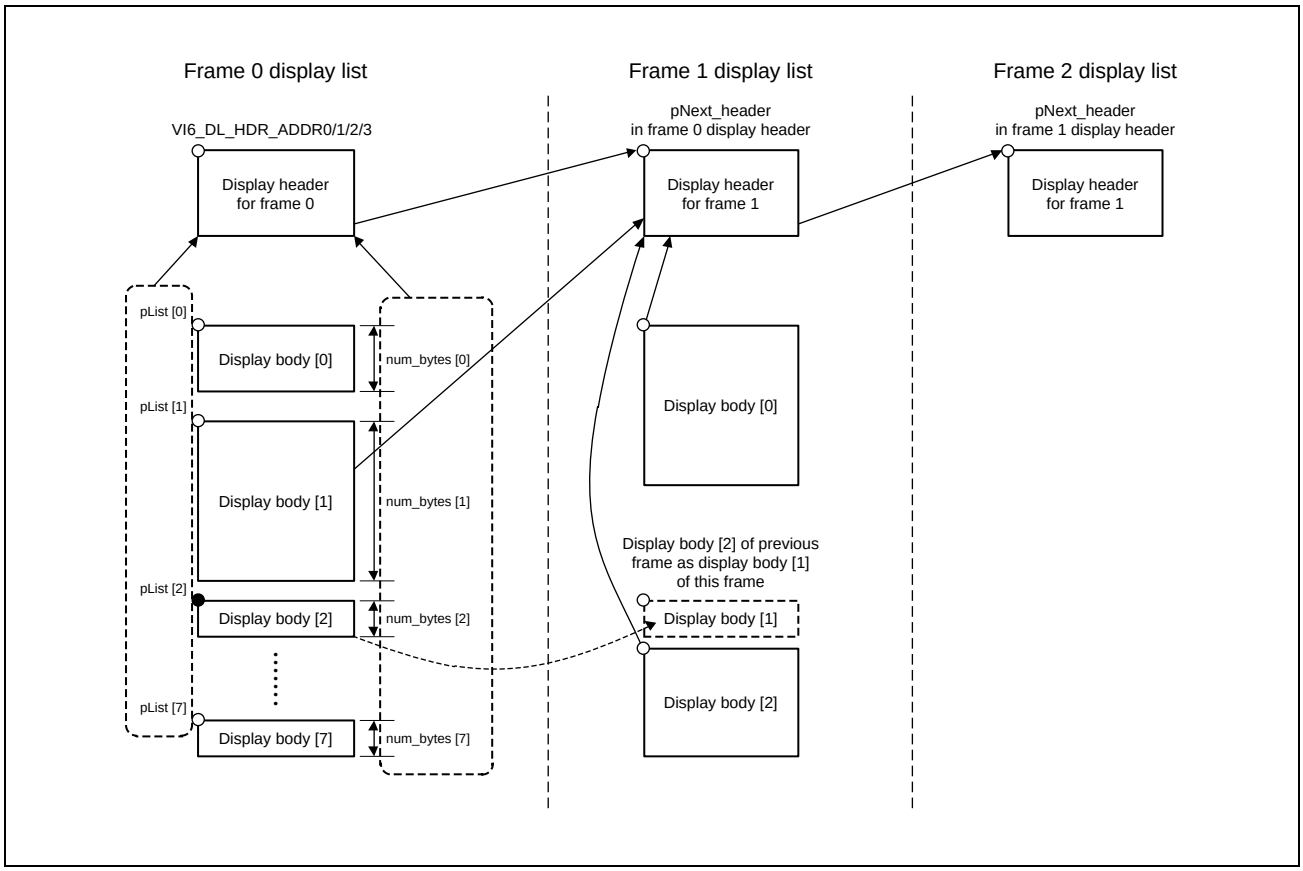


Figure 9.4-37 Structure and Concept of VSPD Display List

A VSPD display list consists of two sections; a header section for storing various information and control flags and a body section for storing register and table settings. A combination of these two sections is defined as a display list for a frame. The register and table settings can be divided and stored in up to eight separate bodies allocated in memory. Therefore, the separate bodies storing the register settings for one frame (for example, frame 0) can have non-sequential start addresses; that is, the bodies for one frame can be allocated to areas distant from each other in memory. To gather these bodies and configure the register settings for one frame, a header is used. A display header stores the number of bodies linked with the header and the start address and size of each body.

The VSPD analyzes the header to gather register and table settings stored in separate memory areas and reconfigures the complete register settings.

The addresses of display headers should be specified in the VSPD registers described in **9.4.3.2.5 Display List Control Registers**. When activated in a mode that uses display lists, the VSPD downloads display headers from the

addresses specified in VI6_DL_HDR_ADDR0 (number 0 correspond to the WPF channel index numbers), analyzes the numbers of bodies and the address and data size of each body, downloads the bodies, and completes register and table settings. After display list downloading is completed, the VSPD becomes ready for frame image processing; the VSPD then starts the actual frame processing.

After processing of a frame ends, the VSPD proceeds to the next frame processing. Here, there are two modes for starting the next frame processing as shown in **Figure 9.4-36**. In one mode, the VSPD stops operation and waits for the next activation by the CPU in the same way as when display lists are not used. In this mode, the address used for downloading the display header of the next frame is kept by the internal hardware. Therefore, if valid address information is not stored in the display header for the previous frame, a correct value should be specified in VI6_DL_HDR_ADDR0 while the VSPD is stopped. When the VSPD is started after a correct value is specified, the VSPD starts next frame processing with the same procedure for the previous frame. In contrast to this mode, which stops the VSPD after the end of one-frame processing, there is another mode for automatically starting next frame processing. In automatic start mode, the VSPD downloads the next display header as soon as the previous frame processing ends. After downloading ends, the VSPD starts image processing. The information regarding mode selection, that is, whether to automatically start next frame processing, should be stored in the display header downloaded for the previous frame.

To strictly define the display list format described above, the following shows the grammatical structure of a display list using pseudo-code. First, to simplify the description of the display list format in the following pages, **Table 9.4-30** defines a function. Function zero bits (num bits) generates a string of one-bit 0s for the number of bits specified by the parameter for the function. By using this function, **Table 9.4-31** defines the header section format of a display list and **Table 9.4-32** defines the body section format and **Table 9.4-33** defines the extended display list body section format.

Data order of Display List Header Section, Display List Body Section and Extended Display List Body Section are assumed as the data is stored in SDRAM by big endian. **Table 9.4-34** shows an example of data order. If data order is not same as the definition, data order within 8byte unit can be adjusted by setting VI6_DL_SWAP (see **(3) Display List-0 Data Swapping Register (LCDC_VSPD_VI6_DL_SWAP0)**).

Table 9.4-30 Definition of a Function for Simple Description

Syntax	Bit Count
zero_bits (num_bits)	
{	
for (i = 0; i<num_bits; i++) {	
zero_bit	1
}	
}	

Bit String	Contents
zero_bit	zero_bit indicates a 1-bit integer having a value of 0.

Table 9.4-31 Format of Display List Header Section

Syntax	Bit Count
display_header () /* Fixed length */	
{	
zero_bits (29)	
num_list_minus1	3
for (i=0; i<8; i++)	
zero_bits (15)	
num_bytes [i]	17
pList [i]	32
}	
pNext_header	32
zero_bits (23)	
wait_wup	1
zero_bits (3)	
ignore_upd_dl	1
zero_bits (2)	
current_frame_int_enable	1
next_frame_auto_start /* 76 bytes from the beginning of this header*/	1
if (VI6_DL_EXT_CTRL.EXT) {	
zero_bits (32) /* padding zero 4 bytes for alignment */	
zero_bits (6)	
pre_ext_dl_exec	1
post_ext_dl_exec	1
zero_bits (8)	
pre_ext_dl_num_cmd	16
pre_ext_dl_pList	32
zero_bits (16)	
post_ext_dl_num_cmd	16
post_ext_dl_pList /* 96 bytes from the beginning of this header*/	32
}	
}	

Bit String	Contents
num_list_minus1	Specifies the value obtained by subtracting 1 from the total number of display list bodies linked with the display header. For example, when this bit field is set to 0, this display list uses one body.
num_bytes [i]	Specifies the number of bytes in the i-th display list body (indicated by index i). Be sure to specify a multiple of eight bytes. For the bodies that are not defined in num_list_minus1 (for example, i = 5 to 7 when num_list_minus1 is set to 4), specify 0.
pList [i]	Specifies the start address of the i-th display list body (indicated by index i). Be sure to specify an address aligned with an 8-byte boundary (the lower-order three bits are 0). For the bodies that are not defined in num_list_minus1 (for example, i = 5 to 7 when num_list_minus1 is set to 4), specify 0.
pNext_header	Specifies the address of the display list header for the next frame. After display list downloading ends, the VSPD keeps its value in the internal memory and uses it in the next display list header downloading. Be sure to specify an address aligned with an 8-byte boundary (the lower-order three bits are 0).
wait_wup	When this bit is 1, VSPD starts reading image data from external memory after WUP (Wake Up) signal from any channel of VIN asserts. This bit is available for WPF0.
ignore_upd_dl	If this bit is set to 1, new display list pointed by VI6_DL_HDR_ADDR0 is not downloaded in VSPD H/W for the next frame, even if S/W sets VI6_DL_HDR_ADDR0 during the frame. New display list is downloaded in VSPD H/W at next frame of the processing which this bit is 0. This bit is available for WPF0.
current_frame_int_enable	This is a flag that indicates whether to set the display list end interrupt source (VI6_WPFn_IRQ_STA.DFE) to 1 when the current frame processing ends. If this flag is set to 0, the display list end interrupt source (VI6_WPFn_IRQ_STA.DFE) is not set to 1 when one-frame processing by this display header ends. In this state, even if the display list end interrupt is enabled (VI6_WPFn_IRQ_ENB.DFEE is set to 1), no interrupt will be generated. If this flag is set to 1, the display list end interrupt source (VI6_WPFn_IRQ_STA.DFE) is set to 1 when one-frame processing by this display header ends. In this state, if the display list end interrupt is enabled (VI6_WPFn_IRQ_ENB.DFEE is set to 1), the VSPD generates an interrupt.
Next_frame_auto_start	Enables or disables automatic start of next frame processing when one-frame processing by this display header ends. If this bit is set to 1, the VSPD starts next frame processing as soon as one-frame processing by this display header ends, and starts downloading the next frame display header from the pNext_header address specified in this display header. If this bit is set to 0, the VSPD stops operation when one-frame processing by this display header ends. In this case, start the VSPD through VI6_CMDn to process the next frame.
pre_ext_dl_exec	Enables execution of the extended display list for frame preprocessing when VI6_DL_EXT_CTRL.EXT is 1. If this bit is set to 1, the VSPD executes the extended display list for frame preprocessing. The VSPD does not execute it if this bit is set to 0. This bit setting is ignored when VI6_DL_EXT_CTRL.EXT is 0.
post_ext_dl_exec	Set 0 to this bit.
pre_ext_dl_num_cmd	Specifies the number of commands in the extended display list body section for frame preprocessing when VI6_DL_EXT_CTRL.EXT is 1. The number of commands that can be specified is 1, and a command is 16 bytes. When pre_ext_dl_exec is set to 0, the extended display list for frame preprocessing is not executed; specify 0 in this bit. This bit setting is ignored when VI6_DL_EXT_CTRL.EXT is 0.
pre_ext_dl_pList	Specifies the start address of the area where the extended display list body section for frame preprocessing is stored when VI6_DL_EXT_CTRL.EXT is 1. Be sure to specify an address aligned with a 16-byte boundary (lower-order four bits are 0). When pre_ext_dl_exec is set to 0, the extended display list for frame preprocessing is not executed; specify 0 in this bit. This bit setting is ignored when VI6_DL_EXT_CTRL.EXT is 0.
post_ext_dl_num_cmd	Set 0 to this bit.
post_ext_dl_pList	Set 0 to this bit.

Table 9.4-32 Format of Display List Body Section

Syntax	Bit Count
display_list (num_bytes) /* Variable length (num_bytes) */	
{	
for (i=0; i<num_bytes; i+=8) {	
set_address	32
set_data	32
}	
}	

Bit String	Contents
set_address	Specifies the address where the value specified by set_data is to be stored. Specify a register address. *Note*: Register space of VSPD is 32 Kbyte (0000h - 7FFCh). So, set 0 to upper 17-bit of set_address in Display List Body Section.
set_data	Specifies the value to be stored in the address specified by set_address. Specify a value to be set in a register.

Table 9.4-33 Format of Extended Display List Body Section

Syntax	Bit Count
ext_dl_display_list (num_llw) /* Variable length (pre/post_ext_dl_num_bytes) */	
{	
for (i=0; i<num_llw; i+=2) {	
ext_dl_cmd	64
ext_dl_data	64
}	
}	

Table 9.4-34 Data order of display list body

Address	Syntax
pList	set_address [0] (bit31-24)
pList + 1	set_address [0] (bit23-16)
pList + 2	set_address [0] (bit15-8)
pList + 3	set_address [0] (bit7-0)
pList + 4	set_data [0] (bit31-24)
pList + 5	set_data [0] (bit23-16)
pList + 6	set_data [0] (bit15-8)
pList + 7	set_data [0] (bit7-0)
pList + 8	set_address [1] (bit31-24)
pList + 9	set_address [1] (bit23-16)
pList + 10	set_address [1] (bit15-8)
pList + 11	set_address [1] (bit7-0)
pList + 12	set_data [1] (bit31-24)
pList + 13	set_data [1] (bit23-16)
pList + 14	set_data [1] (bit15-8)
pList + 15	set_data [1] (bit7-0)
~~	~~~

9.4.5.5.3 Header-less display list mode

The header-less display list does not have the display header listed in **Table 9.4-31**, and it has the simplest structure that has only single body. The extended display list function is not available in case of the header-less display list mode. Set the value 1 to VI6_DL_CTRL.NH0 (see **(1) Display List Control Register (LCDC_VSPD_VI6_DL_CTRL)**) to use the header-less display list. The start address downloaded by the header-less display list should be set to VI6_DL_HDR_ADDR0. And the size of the display body which is originally defined in the display header should be set to VI6_DL_BODY_SIZE0. The header-less display list is available only in WPF0.

9.4.5.5.4 Restrictions on display list usage

Access to the general control registers and display list control registers through a display list is prohibited. When using display lists, be sure to observe the following restrictions on register access by the CPU.

1. Do not execute write access to the same register (same address) from the CPU and through a display list at the same time. If such a conflict occurs, correct operation of the VSPD is not guaranteed.
2. Do not execute write access to the same LUT lookup table from the CPU and through a display list at the same time. If such a conflict occurs, correct operation of the VSPD is not guaranteed. Here, the same lookup table means the address space having the same space name shown in **Table 9.4-35**.
3. When read access by the CPU and write access through a display list to the same register (same address) occur at the same time, the read value returned to the CPU is not guaranteed.
4. When read access by the CPU and write access through a display list to the same LUT lookup table occur at the same time, the read value returned to the CPU is not guaranteed. Here, the same lookup table means the address space having the same space name shown in **Table 9.4-35**.
5. For other restrictions on values and timing of register setting through display lists, refer to the restrictions on normal register settings described in **9.4.3.2.3 Restrictions on Access to Registers and Lookup Tables**.
6. Manipulation and setting of the registers described in **9.4.3.2.4 General Control Registers** and **9.4.3.2.5 Display List Control Registers** through a display list is prohibited.
7. Do not use extended display lists.

9.4.5.6 Interrupt Processing

9.4.5.6.1 vs1o_intreq_n

When a WPF module generates an internal source that should be notified, an interrupt signal is output. As internal sources are generated in WPF independently, the VSPD has the following registers to control interrupts in WPF.

- WPFn Interrupt Enable Registers (VI6_WPFn_IRQ_ENB: n = 0) (refer to **(12) WPF0 Interrupt Enable Register (LCDC_VSPD_VI6_WPF0_IRQ_ENB)**)
- WPFn Interrupt Status Registers (VI6_WPFn_IRQ_STA: n = 0) (refer to **(13) WPF0 Interrupt Status Register (LCDC_VSPD_VI6_WPF0_IRQ_STA)**)

CAUTION

To use interrupts, enable them through WPF interrupt enable registers. If an interrupt source register has already been set to 1 for some reason, an unintended interrupt will occur as soon as the corresponding interrupt enable register is set as enabled. To avoid this, before enabling interrupts through WPF interrupt enable registers, be sure to clear all WPF interrupt sources to be enabled to 0. Be careful about this procedure when setting up registers before starting the VSPD.

9.4.5.6.2 duo_intr_n

When the PBUF will detect a FIFO underflow, an interrupt signal is output. The DU has the following registers to control the interrupt.

- DU Module Status Register 0 (DU_MSR0) (refer to **9.4.4.2.2 DU Module Status Register 0 (LCDC_DU_MSR0)**)
- DU Interrupt Mask Register 0 (DU_IMR0) (refer to **9.4.4.2.4 DU Interrupt Mask Register 0 (LCDC_DU_IMR0)**)

CAUTION

To use interrupts, enable them through DU Interrupt Mask Register. If an interrupt is asserted, the PBUF is underflowing and invalid display data is being output. When an interrupt occurs, the interrupt can only be cleared by DU_MCR0.PB_CLR register.

If DU_MCR1.PB_AUTOCLR is set a value other than 0, an interrupt is cleared according to the DU_MCR1.OPMD register.

9.4.5.7 Lookup Table Settings

9.4.5.7.1 LUT

For a single entry to the LUT space (see **Table 9.4-35**) of the VSPD, the LUT data is set by a write access in the format shown in **Figure 9.4-38**.

The entry address of each space is (start address of the space) + (entry number counting from the base point 0 × 4). For example, the address of entry 7 to the LUT space is (offset address) + 7000h + 7 × 4 = (offset address) + 701Ch.

Table 9.4-35 shows the spaces for which entries can be made. Note that if the module that references that space is operating, write accesses to the relevant space are prohibited. So, while LUT is operating, accesses to the entire space of VI6_LUT_TBL are prohibited. When a read access is made to the relevant space during operation of the referencing module, undefined values will be read out.

The operating/stopped state of each module in **Table 9.4-35** is the operating state of the WPF to which each module is connected. Determine whether the module is operating or stopped using each WPF operating status bit in the VI6_STATUS register.

Table 9.4-35 LUT Space Addresses

Space Name	Space Addresses	Entry Count	Module that References the Space in the Left Column
VI6_LUT_TBL	(offset address) + 7000h to (offset address) + 73FCh	256	LUT

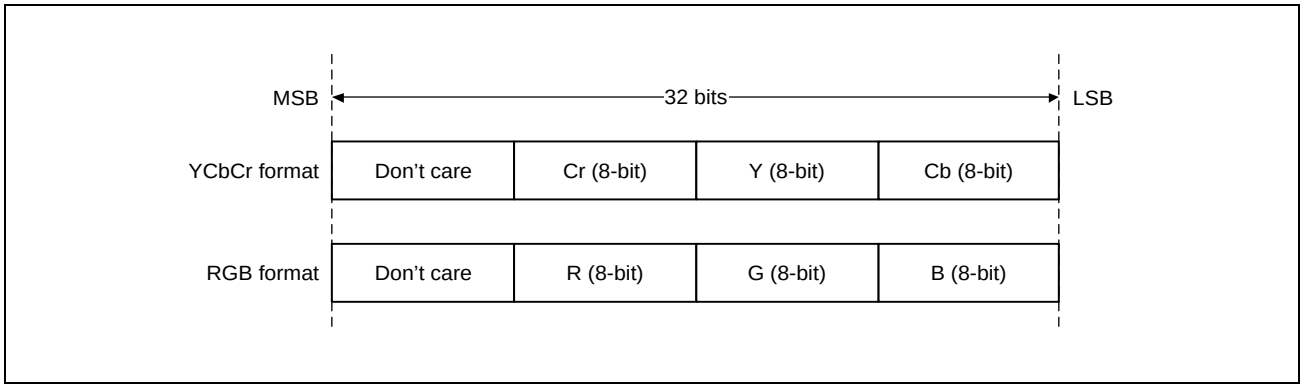


Figure 9.4-38 VI6_LUT_TBL Formats

9.4.5.8 Linked with DU

9.4.5.8.1 Operation flow of VSPD and DU

This section shows operation flow of VSPD by using Normal Display List Mode. For details of Normal Display List Mode, refer to **9.4.5.5.2 Normal display list mode**.

Figure 9.4-39 and **Figure 9.4-40** shows a procedure (register setting flow) to display image data to display panel.

Figure 9.4-39 shows setting flow to start VSPD/LIF0 linked with DU. **Figure 9.4-40** shows setting flow to update display image.

Using display list with enabling next_frame_auto_start shown in **9.4.5.5 Display List** is required to take the flow from **Figure 9.4-39** and **Figure 9.4-40**.

NOTE

If display sync operation ([1-7] step in **Figure 9.4-39**) is set before activating the VSPD, invalid data is displayed until image data read from external memory is ready to be transferred to DU. In this case, image data from external memory is displayed late by one vsync or more.

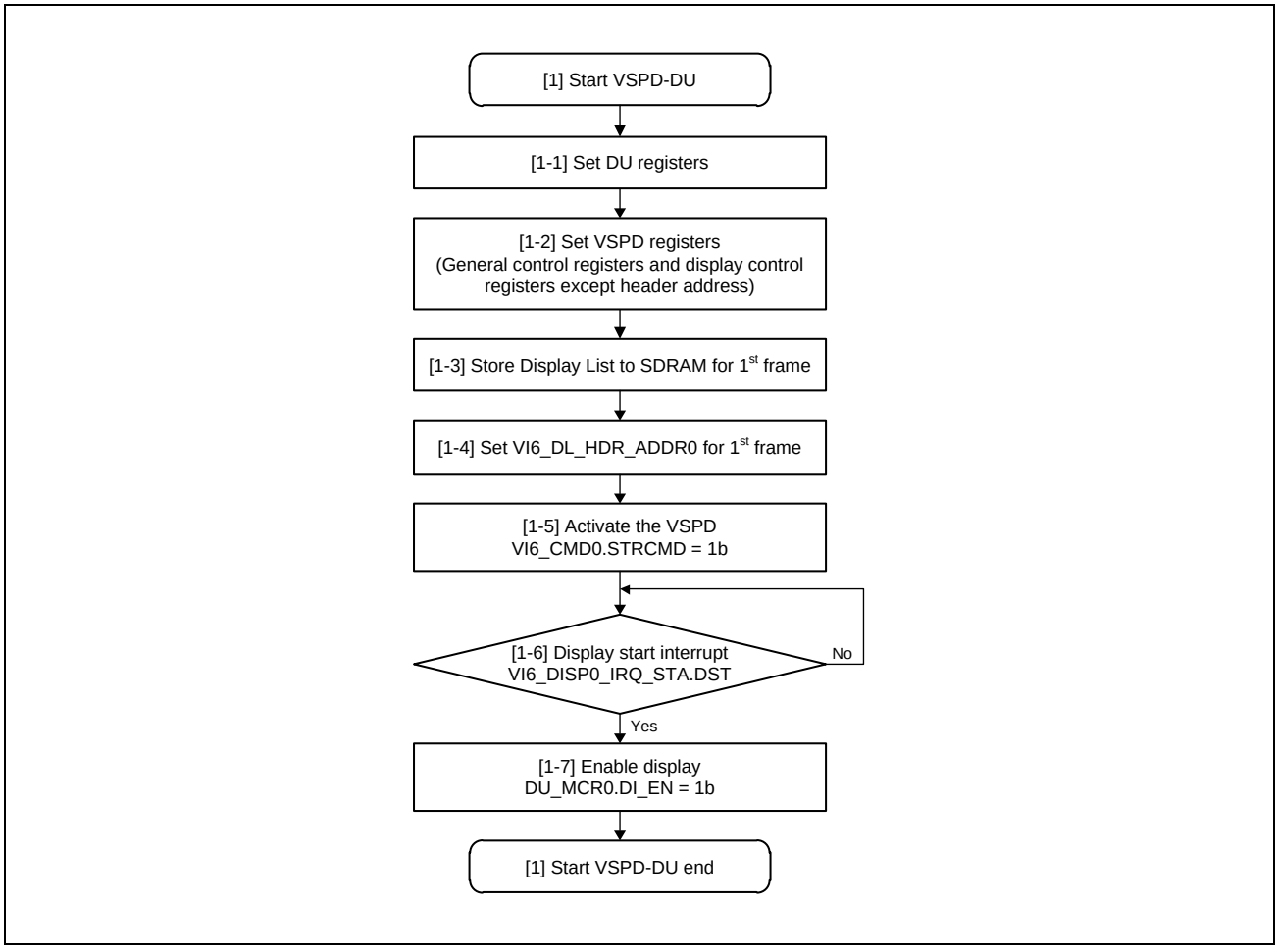


Figure 9.4-39 Setting Flow to Start VSPD Linked with DU

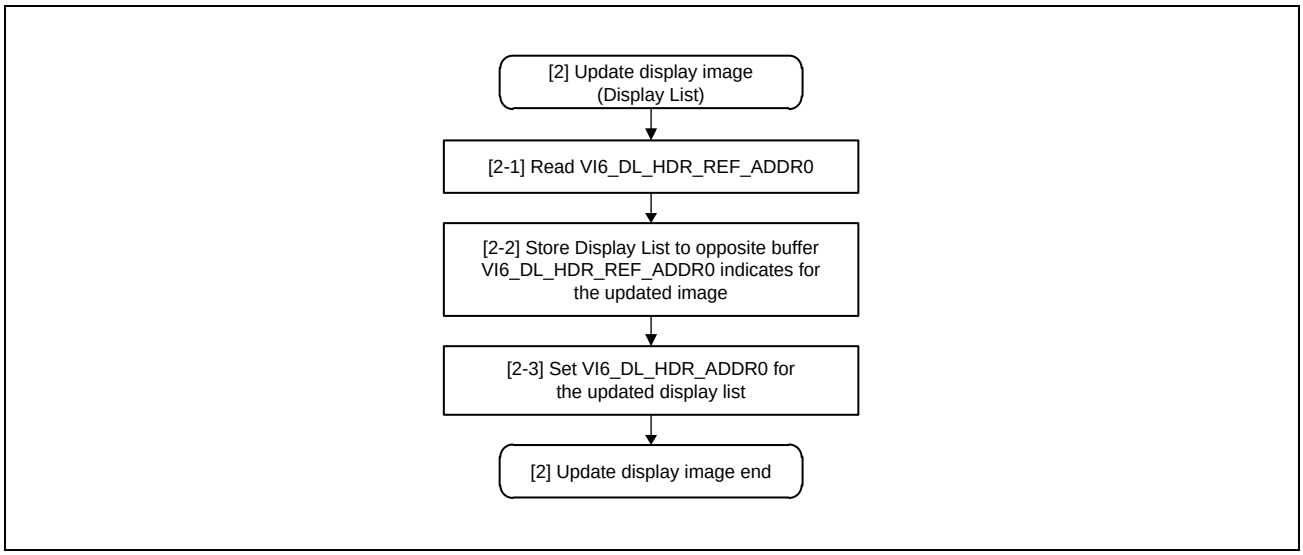


Figure 9.4-40 Setting flow to update display image

It is possible to update display list at any time (arbitrary timing) as shown in **Figure 9.4-40**. Don't overwrite display list in external memory pointed by VI6_DL_HDR_REF_ADDR. It is because VSPD is reading the display list pointed by VI6_DL_HDR_REF_ADDR or will be reading the display list soon.

Detail explanation is shown below.

As shown in **(6) Display List-0 Header Reference Address Register (LCDC_VSPD_VI6_DL_HDR_REF_ADDR0)**, when VSPD is reading display list from external memory (Period [A] in **Figure 9.4-41**), VI6_DL_HDR_REF_ADDRn indicates header address of the display list referred by VSPD. When VSPD is not reading display list from external memory (Period [B] in **Figure 9.4-41**), VI6_DL_HDR_REF_ADDRn indicates the value of VI6_DL_HDR_ADDRn, and there is possibility that the display list is being read soon by VSPD at next frame start timing. Therefore, keep display list in external memory pointed by VI6_DL_HDR_REF_ADDR. Detail software sequence to update display list and VSPD -H/W behavior are shown below.

[1] Store latest display list in external memory area different from the area pointed by VI6_DL_HDR_REF_ADDR, because the area pointed by VI6_DL_HDR_REF_ADDR is being read or is being read soon by VSPD.

[2] Set header address of new display list into VI6_DL_HDR_ADDRn.

[3] VSPD reads display list from the external memory area pointed by VI6_DL_HDR_ADDRn at frame start timing.

Refer also to **(2) Display List-0 Header Address Register (LCDC_VSPD_VI6_DL_HDR_ADDR0)** and **(6) Display List-0 Header Reference Address Register (LCDC_VSPD_VI6_DL_HDR_REF_ADDR0)**.

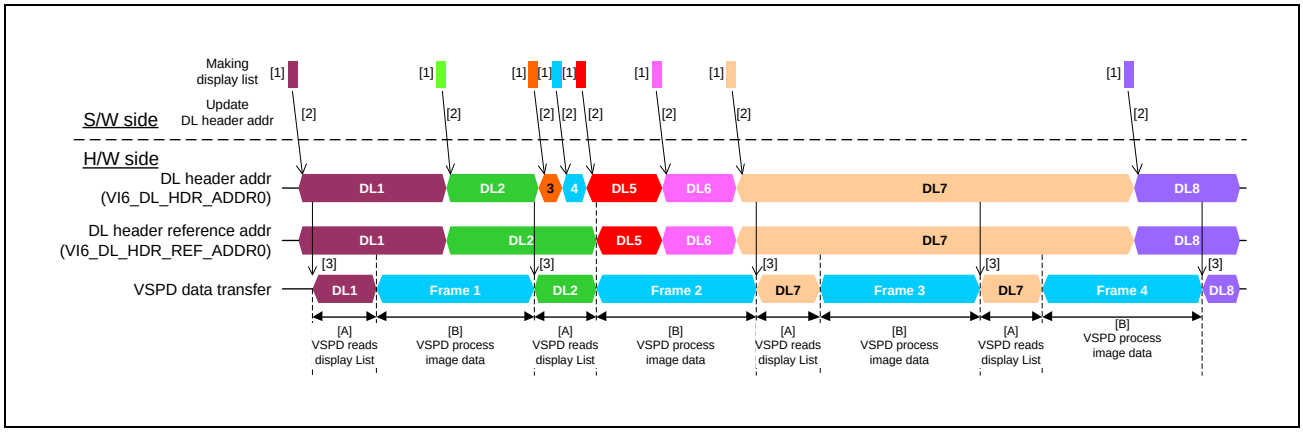
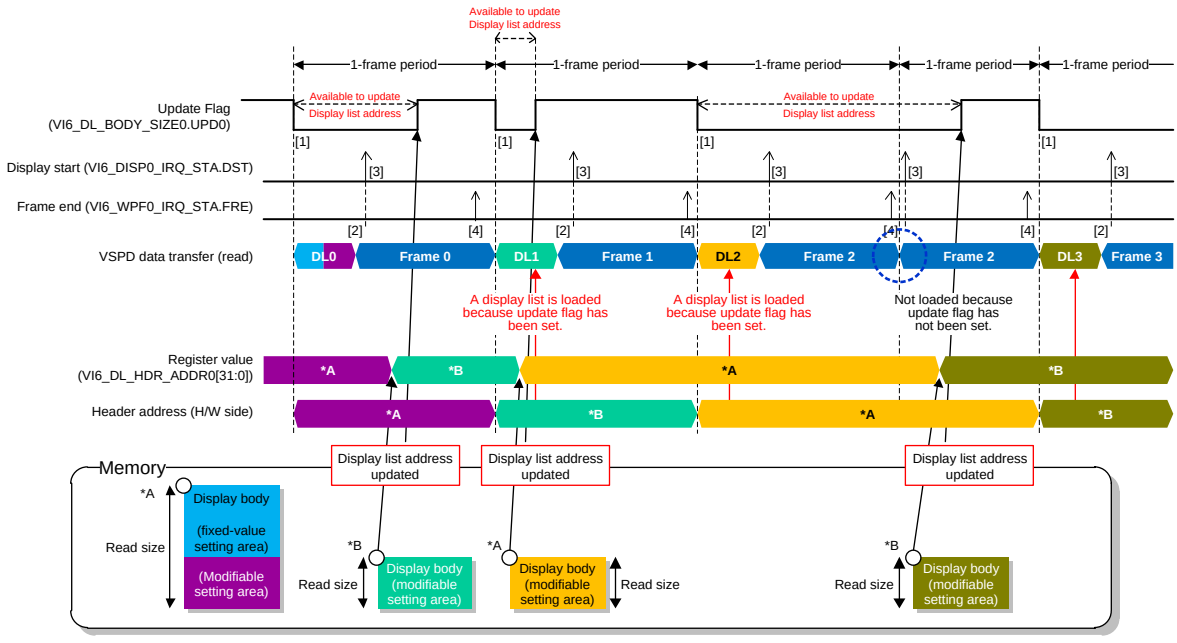


Figure 9.4-41 Updating Display List at Arbitrary Timing

9.4.5.8.2 Controlling two register planes using display lists

This section shows operation flow of VSPD by using Header-less Display List Mode. Start procedure and stop procedure are same with Normal Display List Mode in **9.4.5.8.1 Operation flow of VSPD and DU** except for the point that VI6_DL_BODY_SIZE0 should be set before step [1-5] in start procedure shown in **Figure 9.4-39**. The procedure to update display list is different with Normal Display List Mode, and is described later. **Figure 9.4-42** shows the control of two register planes using header-less display lists (see **9.4.5.5.3 Header-less display list mode**) and its timing. In the description hereafter, the use of header-less display lists is always assumed and they are simply called display lists.



Display list plane A: “Fixed-value settings” and “modifiable settings” are stored as a set in contiguous area A.
Display list plane B: Only the “modifiable settings” are stored in area B, which is separate from display list A.

To switch settings for every frame, modify two display list planes alternately. After every modification, update the display list address register (VI6_DL_HDR_ADDR0[31:0]) and writing 1 to update flag (VI6_DL_BODY_SIZE0.UPD0) in the VSPD.
Update flag (VI6_DL_BODY_SIZE0.UPD0) is negated automatically after Header Address is loaded into H/W side.

Figure 9.4-42 Controlling Two Register Planes Using a Display List

The VSPD downloads a display list immediately after activation. In the start frame, download the display list that contains all necessary settings. From the next frame on, only the necessary register or table values should be specified in a display list.

When the update flag of VI6_DL_BODY_SIZE0.UPD0 is set to 1, VSPD downloads a new display list at the start of the next frame. When the update flag of VI6_DL_BODY_SIZE0.UPD0 is set to 0, the register settings acquired from the display list previously downloaded are retained and used for the next operation without downloading a new display list.

Order of each VSPD -H/W Events mentioned as [1], [2], [3] and [4] in **Figure 9.4-42** are shown below.

[1] VI6_DL_BODY_SIZE0.BS0 and VI6_DL_HDR_ADDR0 are downloaded in H/W side. And Update Flag (VI6_DL_BODY_SIZE0.UPD0) is negated automatically.

[2] Reading DisplayList from external memory into VSPD -H/W is finished through master access.

[3] Display start interrupt status (VI6_DISP_IRQ_STA.DST) asserts.

[4] Frame end interrupt status (VI6_WPF0_IRQ_STA.FRE) asserts.

Don’t overwrite the display list in external memory for a period from “[4] VI6_WPF0_IRQ_STA.FRE” to next “[3] VI6_DISP_IRQ_STA.DST”, because VSPD is reading the display list for the period.

9.4.6 Usage Note

9.4.6.1 Assignment in Memory Space

Make sure that VSPD memory space shall be mapped to Non-Cache region.

9.4.6.2 Limitations on Software Reset and Module Standby

Stop LCDC with the software reset sequence following to the guidance in **9.4.5.1 Operation Control Setting**.

9.4.6.3 Input Image Size

Table 9.4-36 is a list of input size specifications.

Table 9.4-36 List of Input Size Specifications

Module	Min. Input Size (horizontal × vertical)	Max. Input Size (horizontal × vertical)	Restriction on Setting Unit
RPF	1 × 1 pixel	1920 × 1200 pixels 1200 × 1920 pixels	YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. YCbCr420: 2-pixel units both horizontally and vertically. Other formats: 1-pixel units both horizontally and vertically. *Note 1.* When the 1-bpp alpha plane*1 is input, the size can always be specified in 8-pixel units both horizontally and vertically regardless of the input format. *Note 2.* These restrictions including note 1 are applied to the following. VI6_RPFn_SRC_BSIZE VI6_RPFn_SRC_ESIZE
LUT	1 × 1 pixel	1920 × 1200 pixels 1200 × 1920 pixels	1-pixel units both horizontally and vertically.
BRS	1 × 1 pixel	1920 × 1200 pixels 1200 × 1920 pixels	1-pixel units both horizontally and vertically.
LIF	1 × 1 pixel	1920 × 1200 pixels 1200 × 1920 pixels	1-pixel units both horizontally and vertically.
WPF	1 × 1 pixel	1920 × 1200 pixels 1200 × 1920 pixels	YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. YCbCr420: 2-pixel units both horizontally and vertically. Other formats: 1-pixel units both horizontally and vertically. *Note 1.* This restriction on the WPF only applies to the WPF output size. The WPF input size should be specified in 1-pixel units.

Note 1. When VI6_RPFn_ALPH_SEL.ASEL is set to 011b.

The most important restriction shown in **Table 9.4-36** is the setting unit. Make appropriate register settings so that the size of the image input to each module does comply with setting units and does not exceed the limits shown in **Table 9.4-36**.

9.4.6.4 Output Image Size

The size of the output from each WPF is determined by the results of processing in the modules connected with the DPR. As shown in **Figure 9.4-1**, the data input to the VSPD is sent to the WPF output modules through the RPF and the modules connected with the DPR. When there is no processing that changes the image size through this data path, the WPF output size is the same as the RPF input size. **Table 9.4-37** is a list of the processing that changes image size.

Table 9.4-37 Image Processing that changes image size

Module	Function*1	Related Register	Size of Output from Module
WPF	Input size clipping	VI6_WPFn_HSZCLIP VI6_WPFn_VSZCLIP	When this function is disabled, the input size and the output size are the same. When this function is enabled, the output is in the following size. Horizontal output size: VI6_WPFn_HSZCLIP.HCL_SIZE × 2 setting Vertical output size: VI6_WPFn_VSZCLIP.VCL_SIZE × 2 setting
LIF	Padding line	VI6_LIFn_CTRL	When padding with dummy lines is disabled, output size is same with Input size. When padding with dummy lines is enabled: Horizontal output size: input size Vertical output size: VI6_LIF0_PADLN_SIZE setting

Note 1. For details of each function, refer to the descriptions of the related registers.

Note 2. Refer to **9.4.3.2.1 Notational Conventions for Registers and Bit Fields** for explanation of register bit field.

The input image size can be changed only with the modules and functions shown in **Table 9.4-37**. With the other modules and functions, the input size and output size are the same. Accordingly, after module connections with the DPR are determined, the VSPD output size can also be determined through the following steps.

1. The image size (VI6_RPFn_SRC_ESIZE) read from the external memory and sent to the DPR by the RPFn is the initial value.
2. When the module connected with the DPR is not a module (function) shown in **Table 9.4-37**, the output size from the module is the same as the input size; the image size does not need to be updated.
3. When a module connected with the DPR is a module (function) shown in **Table 9.4-37**, the output image size should be updated to the size shown in the table, which should be used as the input image size for the module connected behind it.
4. When the final image size at the WPFn is determined, this size is the VSPD output size for that WPFn path.

Figure 9.4-43 shows how to determine the image size in a sample DPR connection through the above steps. The related conditions that determine the image size are also shown.

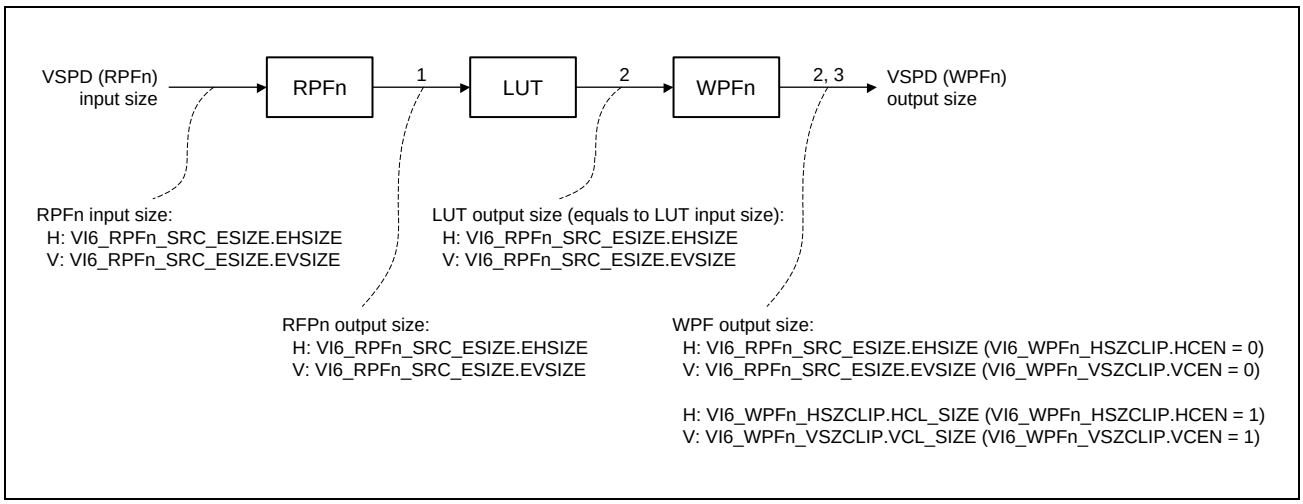


Figure 9.4-43 Input/output Size for Each Module in a Sample DPR Connection

Make appropriate register settings in each module so that the VSPD output image size determined as shown in the figure does not violate the restrictions shown in **Table 9.4-37**.

9.4.6.5 Restriction

When the data output from this unit is written back to the same memory area where the input data for this unit has been read, this unit has the following restrictions.

1. The access order and format on the frame memory are the same between the input pixels and output pixels.
2. Specifying a larger output image size than the input image size, either vertically or horizontally or both vertically and horizontally, is prohibited.
3. When the YCbCr4:2:0 format is input, operation between color components is prohibited.

These restrictions are summarized in. Refer to the descriptions of the registers related to each restriction. In the table, RPFm indicates the RPFn that inputs the master layer, and WPFwb indicates the WPFn that writes back the output image to the source image area for the master layer.

Table 9.4-38 Restrictions on Use when Output Data is Written Back to Input Data Area

No.	Restriction	Related Registers
Restriction 1	The RPFm input format and the WPFwb output format should be the same.	VI6_RPFn_INFMT.RDFMT VI6_RPFn_INFMT.VIR
	The RPFm source image storing address and the WPFwb destination address should be the same.	VI6_RPFn_SRCM_ADDR_*
	The RPFm source picture memory stride and the WPFwb destination memory stride should be the same.	VI6_RPFn_SRCM_PSTRIDE VI6_RPFn_SRCM_ASTRIDE
	The RPFm and WPFwb data swapping settings should be the same.	VI6_WPFn_DSWAP
Restriction 2	The RPFm basic read size and extended read size should be the same.	VI6_RPFn_SRC_BSIZE VI6_RPFn_SRC_ESIZE
	Color space conversion is prohibited in RPFm and WPFwb.	VI6_RPFn_INFMT.CSC VI6_WPFn_OUTFMT.CSC
Restriction 3*1	NOP should be specified for IROP operation.	VI6_RPFn_ALPH_SEL.IROP
	Color keying is prohibited.	VI6_RPFn_CKEY_CTRL.CV

Note 1. When the input format is not YCbCr4:2:0, restriction 3 in not applied.

SECTION 9 IMAGE

9.5 MIPI DSI Interface (DSI)

9.5.1 Overview

This section describes the features of the MIPI-DSI unit of this LSI.

This unit is the MIPI-DSI Tx Module; it is composed of the MIPI DSI-2 host controller (LINK) and the MIPI D-PHY Tx (D-PHY).

This unit supports the MIPI Alliance Specification for Display Serial Interface (DSI) Specification. This unit provides a solution for transmitting MIPI DSI compliant digital video and packets. The normative references are listed below.

- MIPI Alliance Specification for Display Serial Interface Version 1.3.1
- MIPI Alliance Specification for D-PHY Version 1.2

9.5.1.1 Features

The following is key features of this unit.

In this document the word “DSI” means “Display Serial Interface Version 1.3.1”.

Overview

- 1 channel
- The number of lanes: 4 lanes
- Supports up to Full HD (1920 × 1200), 60 fps (RGB888)
- Maximum bandwidth: 1.5 Gbps per lane
- Supports the output data format: RGB666 / RGB888

Protocol Interface

- PPI interface (8 bits)
- Packet generation from Video Input signal for Video Mode (Video-Input Operation)
  - Supports the RGB format (18-bit, 18-bit loosely and 24-bit)
  - Not supports the RGB format (30-bit and 36-bit)
  - Not supports the YCbCr 4:2:2 format (16-bit, 20-bit loosely and 24-bit)
  - Not supports the YCbCr 4:2:0 format (12-bit)
  - Not supports Compressed Pixel Stream
  - Supports Blanking Packet or LP-11 selection during each of HSA, HBP, and HFP blanking interval
  - Supports “Non-Burst Mode with Sync Pulse”, “Non-Burst Mode with Sync Event”, and “Burst Mode”
  - Not supports interlaced video
  - Not supports deskew pattern insertion after vertical video data timing

- LP only packet generation and LP packet reception from descriptor list (Sequence Operation channel 0)
  - Supports one sequence input channel
- HS or LP packet generation and LP packet reception from descriptor list (Sequence Operation channel 1)
  - Supports one sequence input channel

Link Layer

- Supports 1, 2, 3, and 4 lane configurations
- Supports unidirectional high-speed mode Tx
- Supports bidirectional LP mode Tx/Rx (Only Lane0)
- Supports ECC/Checksum generation for Tx packet
- Supports ECC/Checksum verification and ECC error correction for Rx packet
- Supports automated insertion of EoTp in HS mode
- Supports ULPS (Tx)
- Supports automated power change to LP mode and return to HS mode
- Supports automated clock stop and resume (non-continuous clock mode)
- Supports assignment for Virtual Channel for video input channel in Video-Input Operation
- Supports assignment for individual Virtual Channel for each packet in Sequence Operation
- Supports detection for PHY contention error and timeout error
- Supports generation of scrambled packets
- Not supports input of TE signal

PHY Layer

- Synchronous link between Master (data source) and Slave (data sink)
- All lanes support high-speed transmission in the forward direction.
- Bi-directional data transmission in Low-Power mode at the Master Data Lane 0 only
- Uses token passing to control the communication direction of the link.
- High-Speed mode for fast data traffic and Low-Power mode for controls and low-speed data transmission.
- High-Speed mode: Differential and terminated, 200 mV swing, and 80 - 1500 Mbps operation speed.
- Low-Power mode: Single-ended and non-terminated, 1.2 V swing, and maximum 10 Mbps speed.
- Number of lanes
  - Master data lane: 4 lanes
  - Master clock lane: 1 lane
- Lane module functions
  - Master data lane 0: CIL-MFAA (HS-TX, LP-TX, LP-RX, LP-CD)
  - Master data lanes 1 to 3: CIL-MFEN (HS-TX, LP-TX)
  - Master clock lane: CIL-MCNN (HS-TX, LP-TX)

- Optional functions

  [Escape mode]

  - LPDT
    - LP transmit: Supported
    - LP receive: Supported (only Master Data Lane 0)
    - ULPS: Supported
    - Reset trigger: Supported
  - Bi-direction
    - Bus turn-around: Supported
    - Contention detection: Supported
  - Calibration
    - Deskew calibration / Preamble: Not-supported
    - Pre/De-Emphasis: Not-supported
    - Equalization: Not-Supported
  - Video Input Interface
    - Selectable 2 channels

9.5.1.2 Block Diagram

Figure 9.5-1 shows a block diagram.

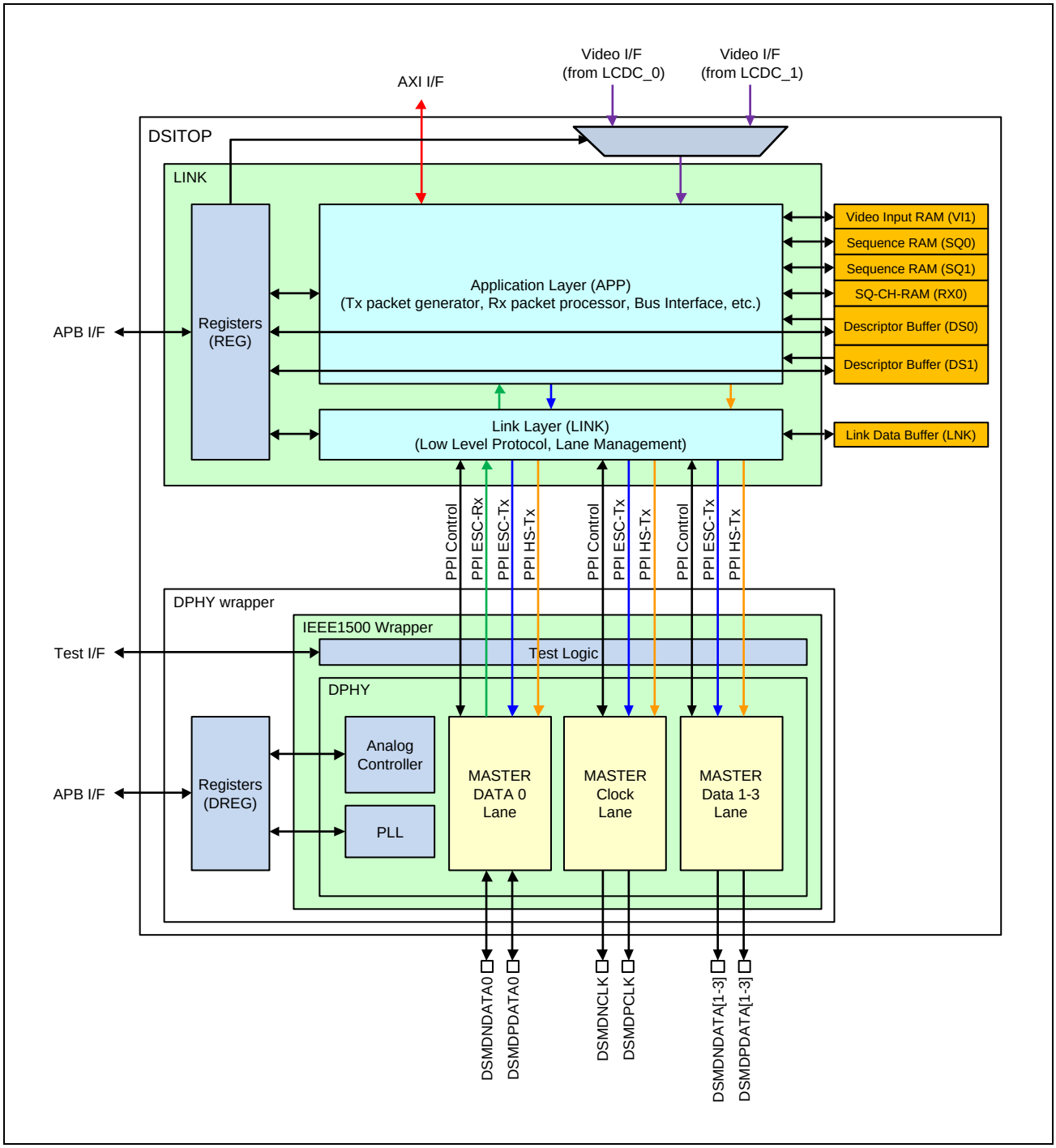


Figure 9.5-1 Block Diagram

9.5.1.3 External Pins

Table 9.5-1 shows the pin configuration.

Table 9.5-1 External Pins

Pin Name	Input/Output	Function
DSI_DPCLK	Output	MIPI DSI Tx clock lane (pos)
DSI_DNCLK	Output	MIPI DSI Tx clock lane (neg)
DSI_DPDATA0	Output	MIPI DSI Tx data 0 lane (pos)
DSI_DNDATA0	Output	MIPI DSI Tx data 0 lane (neg)
DSI_DPDATA1	Output	MIPI DSI Tx data 1 lane (pos)
DSI_DNDATA1	Output	MIPI DSI Tx data 1 lane (neg)
DSI_DPDATA2	Output	MIPI DSI Tx data 2 lane (pos)
DSI_DNDATA2	Output	MIPI DSI Tx data 2 lane (neg)
DSI_DPDATA3	Output	MIPI DSI Tx data 3 lane (pos)
DSI_DNDATA3	Output	MIPI DSI Tx data 3 lane (neg)

9.5.1.4 Clocks

The following shows a list of clocks for this unit.

- T_{LPX}: Transmitted length of any low-power state period (Min: 50 ns)
- $T_{LPX(H)}$: TLPX of the host processor (the DSI-2 TX module)
- $T_{LPX(P)}$: TLPX of the peripheral device (connected device)

Table 9.5-2 Clocks and the Frequency Range

Clock Name	CPG Clock Name	Input/ Output	Description	Frequency (MHz)
pclk	DSI_0_pclk	Input	APB clock	100
aclk	DSI_0_aclk	Input	AXI clock	400/200/100/50/13
vclk1	DSI_0_vclk1	Input	Video clock	187.5 to 5.44
lpclk	DSI_0_lpclk	Input	DSI Escape mode Transmit Clock (TxClkEsc) Set D-PHY TxClkEsc ($1/T_{LPX(H)}$) frequency. $T_{LPX(H)}/T_{LPX(P)}$ needs to set between 2/3 to 3/2.	16/8/4/2
PLLREFCLK	DSI_0_PLLREFCLK	Input	PLL reference input clock	24
hsclk	—	—	DSI High-Speed Transmit Word Clock (TxWordClkHS) Upper limit for D-PHY 1.5 Gbps Lower limit for D-PHY 80 Mbps This clock is supplied from D-PHY directly, and its frequency is 1/8 for the PLL clock frequency (F_{FOUT}).	10 to 187.5
lppclk	—	—	DSI Escape mode Receive Clock (RxClkEsc) This clock is supplied from D-PHY directly, and its frequency is ($1/T_{LPX(P)}$). This clock is generated dividing in D-PHY.	2 to 20

Video clock and DSI HS Byte clock must follow the relationship.

Video clock Frequency [Hz] × Video Pixel Bit Depth [bit]
<= DSI HS Byte clock Frequency [Hz] × 8 [bit] × Number of DSI HS Data Lane

9.5.2 DSI Registers

The base addresses for the DSI are as follows.

Table 9.5-3 Register Base Addresses

Base Register Name	Unit Name	Base Address
<DSI_LINK_base>	DSI (LINK)	0_1643_0000h (5643_0000h*1, 4643_0000h*2)
<DSI_DPHY_base>	DSI (DPHY)	0_1644_0000h (5644_0000h*1, 4644_0000h*2)

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

9.5.2.1 List of Registers

9.5.2.1.1 Registers in LINK

This section describes the registers in LINK. All registers can be accessed in 8/16/32 bits.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Interrupt Status Register	DSI_LINK_ISR	0000_0000h	0000h	8, 16, 32
Reserve	-	-	0004h to 000Fh	-
Link Status Register	DSI_LINK_LINKSR	0000_0000h	0010h	8, 16, 32
Reserve	-	-	0014h to 00BFh	-
General Purpose Output 0 Register	DSI_LINK_GPO0R	0000_0000h	00C0h	8, 16, 32
Reserve	-	-	00C4h to 00FFh	-
Tx Set Register	DSI_LINK_TXSETR	0003_0003h	0100h	8, 16, 32
HS Clock Set Register	DSI_LINK_HSCLKSETR	0000_0000h	0104h	8, 16, 32
ULPS Set Register	DSI_LINK_ULPSSETR	0000_00A0h	0108h	8, 16, 32
ULPS Control Register	DSI_LINK_ULPSCR	0000_0000h	010Ch	8, 16, 32
Reset Control Register	DSI_LINK_RSTCR	0000_0000h	0110h	8, 16, 32
Reset Status Register	DSI_LINK_RSTSR	0000_0F00h	0114h	8, 16, 32
Reserve	-	-	0118h to 011Fh	-
DSI Set Register	DSI_LINK_DSISETR	80F1_0001h	0120h	8, 16, 32
Receive Buffer Size Register	DSI_LINK_RXBUFSZR	0000_0000h	0124h	8, 16, 32
Reserve	-	-	0128h to 015Fh	-
Tx Packet Payload Data 0 Register	DSI_LINK_TXPPD0R	0000_0000h	0160h	8, 16, 32
Tx Packet Payload Data 1 Register	DSI_LINK_TXPPD1R	0000_0000h	0164h	8, 16, 32
Tx Packet Payload Data 2 Register	DSI_LINK_TXPPD2R	0000_0000h	0168h	8, 16, 32
Tx Packet Payload Data 3 Register	DSI_LINK_TXPPD3R	0000_0000h	016Ch	8, 16, 32
Reserve	-	-	0170h to 01FFh	-
Rx Status Register	DSI_LINK_RXSR	0000_0000h	0200h	8, 16, 32
Rx Status Clear Register	DSI_LINK_RXSCR	0000_0000h	0204h	8, 16, 32
Rx Interrupt Enable Register	DSI_LINK_RXIER	0000_0000h	0208h	8, 16, 32
Reserve	-	-	020Ch to 020Fh	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Peripheral Response Timeout BTA Set Register	DSI_LINK_PRESPTOBTASETR	0000_0000h	0210h	8, 16, 32
Peripheral Response Timeout LP Set Register	DSI_LINK_PRESPTOLPSETR	0000_0000h	0214h	8, 16, 32
Peripheral Response Timeout HS Set Register	DSI_LINK_PRESPTOHSSETR	0000_0000h	0218h	8, 16, 32
Reserve	-	-	021Ch to 021Fh	-
Acknowledge and Error Report Packet Parameter Latest Info Register	DSI_LINK_AKEPLATIR	0000_0000h	0220h	8, 16, 32
Acknowledge and Error Report Packet Parameter Accumulate Status Register	DSI_LINK_AKEPACMSR	0000_0000h	0224h	8, 16, 32
Acknowledge and Error Report Packet Parameter Status Clear Register	DSI_LINK_AKEPSCR	0000_0000h	0228h	8, 16, 32
Reserve	-	-	022Ch to 022Fh	-
Rx Result Saved Status Register	DSI_LINK_RXRSSR	0000_0000h	0230h	8, 16, 32
Rx Result Saved Status Clear Register	DSI_LINK_RXRSSCR	0000_0000h	0234h	8, 16, 32
Rx Result Info Overwrite Status Register	DSI_LINK_RXRINFOOWSR	0000_0000h	0238h	8, 16, 32
Rx Result Info Overwrite Status Clear Register	DSI_LINK_RXRINFOOWSCR	0000_0000h	023Ch	8, 16, 32
Rx Result Save Slot 0 Register	DSI_LINK_RXRSS0R	0000_0000h	0240h	8, 16, 32
Rx Result Save Slot 1 Register	DSI_LINK_RXRSS1R	0000_0000h	0244h	8, 16, 32
Rx Result Save Slot 2 Register	DSI_LINK_RXRSS2R	0000_0000h	0248h	8, 16, 32
Rx Result Save Slot 3 Register	DSI_LINK_RXRSS3R	0000_0000h	024Ch	8, 16, 32
Reserve	-	-	0250h to 02BFh	-
Rx Packet Payload Data 0 Register	DSI_LINK_RXPPD0R	0000_0000h	02C0h	8, 16, 32
Rx Packet Payload Data 1 Register	DSI_LINK_RXPPD1R	0000_0000h	02C4h	8, 16, 32
Rx Packet Payload Data 2 Register	DSI_LINK_RXPPD2R	0000_0000h	02C8h	8, 16, 32
Rx Packet Payload Data 3 Register	DSI_LINK_RXPPD3R	0000_0000h	02CCh	8, 16, 32
Reserve	-	-	02D0h to 02DFh	-
HSTX Timeout Set Register	DSI_LINK_HSTXTOSETR	0000_0000h	02E0h	8, 16, 32
LRX-H Timeout Set Register	DSI_LINK_LRXHTOSETR	0000_0000h	02E4h	8, 16, 32
TA Timeout Set Register	DSI_LINK_TATOSETR	0000_0000h	02E8h	8, 16, 32
Reserve	-	-	02ECh to 02FFh	-
Fatal Error Status Register	DSI_LINK_FERRSR	0000_0000h	0300h	8, 16, 32
Fatal Error Status Clear Register	DSI_LINK_FERRSCR	0000_0000h	0304h	8, 16, 32
Fatal Error Interrupt Enable Register	DSI_LINK_FERRIER	0000_0000h	0308h	8, 16, 32
Reserve	-	-	030Ch to 0313h	-
Clock Lane Stop Time Set Register	DSI_LINK_CLSTPTSETR	0000_0000h	0314h	8, 16, 32
LP Transition Time Set Register	DSI_LINK_LPTRNSTSETR	0000_0000h	0318h	8, 16, 32
Reserve	-	-	031Ch to 031Fh	-
Physical Lane Status Register	DSI_LINK_PLSR	0000_0FF3h	0320h	8, 16, 32
Physical Lane Status Clear Register	DSI_LINK_PLSCR	0000_0000h	0324h	8, 16, 32
Physical Lane Interrupt Enable Register	DSI_LINK_PLIER	0000_0000h	0328h	8, 16, 32
Reserve	-	-	032Ch to 03FFh	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Video-Input Channel 1 Set 0 Register	DSI_LINK_VICH1SET0R	0000_0000h	0400h	8, 16, 32
Video-Input Channel 1 Set 1 Register	DSI_LINK_VICH1SET1R	1093_0000h	0404h	8, 16, 32
Reserve	-	-	0408h to 040Fh	-
Video-Input Channel 1 Status Register	DSI_LINK_VICH1SR	0000_0000h	0410h	8, 16, 32
Video-Input Channel 1 Status Clear Register	DSI_LINK_VICH1SCR	0000_0000h	0414h	8, 16, 32
Video-Input Channel 1 Interrupt Enable Register	DSI_LINK_VICH1IER	0000_0000h	0418h	8, 16, 32
Reserve	-	-	041Ch to 041Fh	-
Video-Input Channel 1 Pixel Packet Set Register	DSI_LINK_VICH1PPSETR	000E_0000h	0420h	8, 16, 32
Reserve	-	-	0424h to 0427h	-
Video-Input Channel 1 Vertical Size Set Register	DSI_LINK_VICH1VSSETR	0000_0000h	0428h	8, 16, 32
Video-Input Channel 1 Vertical Porch Set Register	DSI_LINK_VICH1VPSETR	0000_0000h	042Ch	8, 16, 32
Video-Input Channel 1 Horizontal Size Set Register	DSI_LINK_VICH1HSSETR	0000_0000h	0430h	8, 16, 32
Video-Input Channel 1 Horizontal Porch Set Register	DSI_LINK_VICH1HPSETR	0000_0000h	0434h	8, 16, 32
Reserve	-	-	0438h to 05BFh	-
Sequence Channel 0 Set 0 Register	DSI_LINK_SQCH0SET0R	0080_0000h	05C0h	8, 16, 32
Sequence Channel 0 Set 1 Register	DSI_LINK_SQCH0SET1R	0800_0000h	05C4h	8, 16, 32
Reserve	-	-	05C8h to 05CFh	-
Sequence Channel 0 Status Register	DSI_LINK_SQCH0SR	0000_0000h	05D0h	8, 16, 32
Sequence Channel 0 Status Clear Register	DSI_LINK_SQCH0SCR	0000_0000h	05D4h	8, 16, 32
Sequence Channel 0 Interrupt Enable Register	DSI_LINK_SQCH0IER	0000_0000h	05D8h	8, 16, 32
Reserve	-	-	05DCh to 05FFh	-
Sequence Channel 1 Set 0 Register	DSI_LINK_SQCH1SET0R	0000_0000h	0600h	8, 16, 32
Sequence Channel 1 Set 1 Register	DSI_LINK_SQCH1SET1R	0830_0000h	0604h	8, 16, 32
Reserve	-	-	0608h to 060Fh	-
Sequence Channel 1 Status Register	DSI_LINK_SQCH1SR	0000_0000h	0610h	8, 16, 32
Sequence Channel 1 Status Clear Register	DSI_LINK_SQCH1SCR	0000_0000h	0614h	8, 16, 32
Sequence Channel 1 Interrupt Enable Register	DSI_LINK_SQCH1IER	0000_0000h	0618h	8, 16, 32
Reserve	-	-	061Ch to 077Fh	-
Sequence Channel 0 Descriptor 00-A Register	DSI_LINK_SQCH0DSC00AR	xxxx_xxxxh	0780h	32
Sequence Channel 0 Descriptor 00-B Register	DSI_LINK_SQCH0DSC00BR	xxxx_xxxxh	0784h	32
Sequence Channel 0 Descriptor 00-C Register	DSI_LINK_SQCH0DSC00CR	xxxx_xxxxh	0788h	32
Sequence Channel 0 Descriptor 00-D Register	DSI_LINK_SQCH0DSC00DR	xxxx_xxxxh	078Ch	32
Sequence Channel 0 Descriptor 01-A Register	DSI_LINK_SQCH0DSC01AR	xxxx_xxxxh	0790h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Sequence Channel 0 Descriptor 01-B Register	DSI_LINK_SQCH0DSC01BR	xxxx_xxxxh	0794h	32
Sequence Channel 0 Descriptor 01-C Register	DSI_LINK_SQCH0DSC01CR	xxxx_xxxxh	0798h	32
Sequence Channel 0 Descriptor 01-D Register	DSI_LINK_SQCH0DSC01DR	xxxx_xxxxh	079Ch	32
Sequence Channel 0 Descriptor 02-A Register	DSI_LINK_SQCH0DSC02AR	xxxx_xxxxh	07A0h	32
Sequence Channel 0 Descriptor 02-B Register	DSI_LINK_SQCH0DSC02BR	xxxx_xxxxh	07A4h	32
Sequence Channel 0 Descriptor 02-C Register	DSI_LINK_SQCH0DSC02CR	xxxx_xxxxh	07A8h	32
Sequence Channel 0 Descriptor 02-D Register	DSI_LINK_SQCH0DSC02DR	xxxx_xxxxh	07ACh	32
Sequence Channel 0 Descriptor 03-A Register	DSI_LINK_SQCH0DSC03AR	xxxx_xxxxh	07B0h	32
Sequence Channel 0 Descriptor 03-B Register	DSI_LINK_SQCH0DSC03BR	xxxx_xxxxh	07B4h	32
Sequence Channel 0 Descriptor 03-C Register	DSI_LINK_SQCH0DSC03CR	xxxx_xxxxh	07B8h	32
Sequence Channel 0 Descriptor 03-D Register	DSI_LINK_SQCH0DSC03DR	xxxx_xxxxh	07BCh	32
Sequence Channel 0 Descriptor 04-A Register	DSI_LINK_SQCH0DSC04AR	xxxx_xxxxh	07C0h	32
Sequence Channel 0 Descriptor 04-B Register	DSI_LINK_SQCH0DSC04BR	xxxx_xxxxh	07C4h	32
Sequence Channel 0 Descriptor 04-C Register	DSI_LINK_SQCH0DSC04CR	xxxx_xxxxh	07C8h	32
Sequence Channel 0 Descriptor 04-D Register	DSI_LINK_SQCH0DSC04DR	xxxx_xxxxh	07CCh	32
Sequence Channel 0 Descriptor 05-A Register	DSI_LINK_SQCH0DSC05AR	xxxx_xxxxh	07D0h	32
Sequence Channel 0 Descriptor 05-B Register	DSI_LINK_SQCH0DSC05BR	xxxx_xxxxh	07D4h	32
Sequence Channel 0 Descriptor 05-C Register	DSI_LINK_SQCH0DSC05CR	xxxx_xxxxh	07D8h	32
Sequence Channel 0 Descriptor 05-D Register	DSI_LINK_SQCH0DSC05DR	xxxx_xxxxh	07DCh	32
Sequence Channel 0 Descriptor 06-A Register	DSI_LINK_SQCH0DSC06AR	xxxx_xxxxh	07E0h	32
Sequence Channel 0 Descriptor 06-B Register	DSI_LINK_SQCH0DSC06BR	xxxx_xxxxh	07E4h	32
Sequence Channel 0 Descriptor 06-C Register	DSI_LINK_SQCH0DSC06CR	xxxx_xxxxh	07E8h	32
Sequence Channel 0 Descriptor 06-D Register	DSI_LINK_SQCH0DSC06DR	xxxx_xxxxh	07ECh	32
Sequence Channel 0 Descriptor 07-A Register	DSI_LINK_SQCH0DSC07AR	xxxx_xxxxh	07F0h	32
Sequence Channel 0 Descriptor 07-B Register	DSI_LINK_SQCH0DSC07BR	xxxx_xxxxh	07F4h	32
Sequence Channel 0 Descriptor 07-C Register	DSI_LINK_SQCH0DSC07CR	xxxx_xxxxh	07F8h	32
Sequence Channel 0 Descriptor 07-D Register	DSI_LINK_SQCH0DSC07DR	xxxx_xxxxh	07FCh	32
Sequence Channel 1 Descriptor 00-A Register	DSI_LINK_SQCH1DSC00AR	xxxx_xxxxh	0800h	32
Sequence Channel 1 Descriptor 00-B Register	DSI_LINK_SQCH1DSC00BR	xxxx_xxxxh	0804h	32
Sequence Channel 1 Descriptor 00-C Register	DSI_LINK_SQCH1DSC00CR	xxxx_xxxxh	0808h	32
Sequence Channel 1 Descriptor 00-D Register	DSI_LINK_SQCH1DSC00DR	xxxx_xxxxh	080Ch	32
Sequence Channel 1 Descriptor 01-A Register	DSI_LINK_SQCH1DSC01AR	xxxx_xxxxh	0810h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Sequence Channel 1 Descriptor 01-B Register	DSI_LINK_SQCH1DSC01BR	xxxx_xxxxh	0814h	32
Sequence Channel 1 Descriptor 01-C Register	DSI_LINK_SQCH1DSC01CR	xxxx_xxxxh	0818h	32
Sequence Channel 1 Descriptor 01-D Register	DSI_LINK_SQCH1DSC01DR	xxxx_xxxxh	081Ch	32
Sequence Channel 1 Descriptor 02-A Register	DSI_LINK_SQCH1DSC02AR	xxxx_xxxxh	0820h	32
Sequence Channel 1 Descriptor 02-B Register	DSI_LINK_SQCH1DSC02BR	xxxx_xxxxh	0824h	32
Sequence Channel 1 Descriptor 02-C Register	DSI_LINK_SQCH1DSC02CR	xxxx_xxxxh	0828h	32
Sequence Channel 1 Descriptor 02-D Register	DSI_LINK_SQCH1DSC02DR	xxxx_xxxxh	082Ch	32
Sequence Channel 1 Descriptor 03-A Register	DSI_LINK_SQCH1DSC03AR	xxxx_xxxxh	0830h	32
Sequence Channel 1 Descriptor 03-B Register	DSI_LINK_SQCH1DSC03BR	xxxx_xxxxh	0834h	32
Sequence Channel 1 Descriptor 03-C Register	DSI_LINK_SQCH1DSC03CR	xxxx_xxxxh	0838h	32
Sequence Channel 1 Descriptor 03-D Register	DSI_LINK_SQCH1DSC03DR	xxxx_xxxxh	083Ch	32
Sequence Channel 1 Descriptor 04-A Register	DSI_LINK_SQCH1DSC04AR	xxxx_xxxxh	0840h	32
Sequence Channel 1 Descriptor 04-B Register	DSI_LINK_SQCH1DSC04BR	xxxx_xxxxh	0844h	32
Sequence Channel 1 Descriptor 04-C Register	DSI_LINK_SQCH1DSC04CR	xxxx_xxxxh	0848h	32
Sequence Channel 1 Descriptor 04-D Register	DSI_LINK_SQCH1DSC04DR	xxxx_xxxxh	084Ch	32
Sequence Channel 1 Descriptor 05-A Register	DSI_LINK_SQCH1DSC05AR	xxxx_xxxxh	0850h	32
Sequence Channel 1 Descriptor 05-B Register	DSI_LINK_SQCH1DSC05BR	xxxx_xxxxh	0854h	32
Sequence Channel 1 Descriptor 05-C Register	DSI_LINK_SQCH1DSC05CR	xxxx_xxxxh	0858h	32
Sequence Channel 1 Descriptor 05-D Register	DSI_LINK_SQCH1DSC05DR	xxxx_xxxxh	085Ch	32
Sequence Channel 1 Descriptor 06-A Register	DSI_LINK_SQCH1DSC06AR	xxxx_xxxxh	0860h	32
Sequence Channel 1 Descriptor 06-B Register	DSI_LINK_SQCH1DSC06BR	xxxx_xxxxh	0864h	32
Sequence Channel 1 Descriptor 06-C Register	DSI_LINK_SQCH1DSC06CR	xxxx_xxxxh	0868h	32
Sequence Channel 1 Descriptor 06-D Register	DSI_LINK_SQCH1DSC06DR	xxxx_xxxxh	086Ch	32
Sequence Channel 1 Descriptor 07-A Register	DSI_LINK_SQCH1DSC07AR	xxxx_xxxxh	0870h	32
Sequence Channel 1 Descriptor 07-B Register	DSI_LINK_SQCH1DSC07BR	xxxx_xxxxh	0874h	32
Sequence Channel 1 Descriptor 07-C Register	DSI_LINK_SQCH1DSC07CR	xxxx_xxxxh	0878h	32
Sequence Channel 1 Descriptor 07-D Register	DSI_LINK_SQCH1DSC07DR	xxxx_xxxxh	087Ch	32

9.5.2.1.2 Registers in D-PHY

This section describes the registers in D-PHY. All registers can be accessed in 8/16/32 bits.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
DSI PLL Enable Register	DSI_DPHY_PLLENR	0000_0000h	0000h	8, 16, 32
DSI PHY Reset Register	DSI_DPHY_PHYRSTR	0000_0000h	0004h	8, 16, 32
Reserve	-	-	0008h to 000Fh	-
DSI PLL Clock Setting 0 Register	DSI_DPHY_PLLCLKSET0R	0000_0000h	0010h	8, 16, 32
DSI PLL Clock Setting 1 Register	DSI_DPHY_PLLCLKSET1R	0000_0000h	0014h	8, 16, 32
Reserve	-	-	0018h to 001Fh	-
DSI PHY TCLK Setting Register	DSI_DPHY_PHYTCLKSETR	0000_0000h	0020h	8, 16, 32
DSI PHY THS Setting Register	DSI_DPHY_PHYTHSSETR	0000_0000h	0024h	8, 16, 32
DSI PHY TLPX Setting Register	DSI_DPHY_PHYTLPXSETR	0000_0000h	0028h	8, 16, 32
Reserve	-	-	002Ch to 002Fh	-
DSI PHY Control Register	DSI_DPHY_PHYCR	0000_0000h	0030h	8, 16, 32
DSI PHY Control 1 Register	DSI_DPHY_PHYC1R	0000_0000h	0034h	8, 16, 32
DSI PHY Control 2 Register	DSI_DPHY_PHYC2R	0000_0000h	0038h	8, 16, 32
DSI PHY Control 3 Register	DSI_DPHY_PHYC3R	0000_0000h	003Ch	8, 16, 32

9.5.2.2 MIPI DSI Register Descriptions

The prefix (DSI_LINK_, DSI_DPHY_) of the register names is omitted in this and subsequent sections.

9.5.2.2.1 Interrupt Status Register (DSI_LINK_ISR)

Access Size : 8, 16, 32 bits
Address : <DSI_LINK_base> + 0000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	PPI	-	-	-	FERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	RCV	-	-	-	VIN1	-	-	-	SQ1	-	-	-	SQ0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	PPI	0h	R	PPI Interrupt When this value is 1b, a DSI D-PHY PPI related interrupt exists. Refer to the description of the PLSR register for more information.
19 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	FERR	0h	R	FERR Interrupt When this value is 1b, a DSI Fatal Error interrupt exists. Refer to the description of the FERRSR register for more information.
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12	RCV	0h	R	RCV Interrupt When this value is 1b, a DSI Packet Receive interrupt exists. Refer to the description of the RXSR register for more information.
11 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	VIN1	0h	R	VIN1 Interrupt When this value is 1b, a Video Input-Operation channel 1 interrupt exists. Refer to the description of the VICH1SR register for more information.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	SQ1	0h	R	SQ1 Interrupt When this value is 1b, a Sequence Operation channel 1 interrupt exists. Refer to the description of the SQCH1SR register for more information.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	SQ0	0h	R	SQ1 Interrupt When this value is 1b, a Sequence Operation channel 0 interrupt exists. Refer to the description of the SQCH0SR register for more information.

9.5.2.2.2 Link Status Register (DSI_LINK_LINKSR)

Access Size : 8, 16, 32 bits

Address : <DSI_LINK_base> + 0010h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	LPBUSY	HSBUSY	-	-	-	VICHRUN1	-	-	-	SQCHRUN1	-	-	-	SQCHRUN0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	LPBUSY	0h	R	LP Operation Busy The setting 1b indicates that an operation related to LP is in progress.
12	HSBUSY	0h	R	HS Operation Busy The setting 1b indicates that an operation related to HS is in progress.
11 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	VICHRUN1	0h	R	Video-Input Channel 1 Running This bit is a copy of VICH1SR.RUNNING.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	SQCHRUN1	0h	R	Sequence Channel 1 Running This bit is a copy of SQCH1SR.RUNNING.
3 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	SQCHRUN0	0h	R	Sequence Channel 0 Running This bit is a copy of SQCH0SR.RUNNING.

9.5.2.2.3 General Purpose Output 0 Register (DSI_LINK_GPO0R)

Access Size : 8, 16, 32 bits

Address : <DSI_LINK_base> + 00C0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	VICH
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	0h	R	Reserved When read, the initial value is read. When writing, be sure to write the initial value.
0	VICH	0h	RW	Select Video Input Channel. 0: Select channel 1 (vclk1, vi1_*) 1: Select channel 2 (vclk2, vi2_*)

9.5.2.2.4 Tx Set Register (DSI_LINK_TXSETR)

Access Size : 8, 16, 32 bits

Address : <DSI_LINK_base> + 0100h

Initial Value : 0003_0003h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	NUMLANE CAP[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	DLEN	CLEN	-	-	-	-	-	-	NUMLANE USE[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17, 16	NUMLANE CAP[1:0]	3h	R	Number of Lanes Capability 3h: The maximum Lane count is 4 (Lanes 0 to 3 can be used)
15 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9	DLEN	0h	RW	Data Lane Enable When this bit is 1b, Data Lanes are enabled by the PHY interface. Lanes to be enabled are determined by NUMLANEUSE.
8	CLEN	0h	RW	Clock Lane Enable When this bit is 1b, Clock Lane is enabled by the PHY interface.
7 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1, 0	NUMLANE USE[1:0]	3h	RW	Number of Lanes for Use Set the Lane count for use. Selected Data Lanes are enabled if DLEN is “1”. 0h: 1 Lane (use of Lane 0) 1h: 2 Lane (use of Lanes 0 and 1) 2h: 3 Lane (use of Lanes 0 to 2) 3h: 4 Lane (use of Lanes 0 to 3) Setting a value over NUMLANECAP is prohibited.

9.5.2.2.5 HS Clock Set Register (DSI_LINK_HSCLKSETR)

Access Size : 8, 16, 32 bits

Address : <DSI_LINK_base> + 0104h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	HSCLKMO DE	HSCLKRUN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	HSCLKMO DE	0h	RW	HS Clock running Mode 0b: Requests Clock Lane HS transmission when HS transmission is scheduled (non-continuous clock mode). 1b: Keeps Clock Lane HS transmission (continuous clock mode). If HSCLKRUN = 0, this field has no meaning. Changing the value of this bit while HSCLKRUN = 1 is prohibited.
0	HSCLKRUN	0h	RW	Start HS Clock running on Clock Lane 0b: No HS transmission (keep LP) 1b: Allows HS transmission according to HSCLKMODE Confirm stable TxWordClkHS (hsclk) before setting this field to 1. Setting this field to “1” while TXSETR.CLEN = 0 is prohibited.

9.5.2.2.6 ULPS Set Register (DSI_LINK_ULPSSETR)

Access Size : 8, 16, 32 bits

Address : <DSI_LINK_base> + 0108h

Initial Value : 0000_00A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ULPSWKUP[7:0]							
Initial Value	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	ULPSWKUP [7:0]	A0h	RW	Twakeup period of ULPS This period is specified by the count value of lpclk *128. This period should be longer than 1 ms. For example, - In the case lpclk is 4/3 MHz and lpclk*128 = 96 µs, set ULPSWKUP = 0Bh and Twakeup period = 1.06 ms (11*96 µs) - In the case lpclk is 20 MHz and lpclk*128 = 6.4 µs, set ULPSWKUP = A0h and Twakeup period = 1.02 ms (160*6.4 µs)

9.5.2.2.7 ULPS Control Register (DSI_LINK_ULPSCR)

Access Size : 8, 16, 32 bits

Address : <DSI_LINK_base> + 010Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	DLULPSEXT	DLULPSENT	-	-	CLULPSEXT	CLULPSENT	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R0W	R0W	R	R	R0W	R0W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29	DLULPSEXT	0h	R0W	The read value is always 0b. DL ULPS Exit Writing 1b to this bit issues a request for all enabled Data Lanes to exit the Ultra-Low Power State (ULPS). Writing 1b while enabled Data Lanes are not in ULPS is prohibited. Writing 0b has no effect.
28	DLULPSENT	0h	R0W	The read value is always 0b. DL ULPS Enter Writing 1b to this bit issues a request for enabled Data Lanes to enter the Ultra- Low Power State (ULPS). Writing 1b while the enabled Data Lanes are in ULPS is prohibited. Writing 0b has no effect.
27, 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25	CLULPSEXT	0h	R0W	The read value is always 0b. CL ULPS Exit Writing 1b to this bit issues a request for Clock Lane to exit the Ultra-Low Power State (ULPS). Writing 1b while Clock Lane is not in ULPS is prohibited. Writing 0b has no effect.
24	CLULPSENT	0h	R0W	The read value is always 0b. CL ULPS Enter Writing 1b to this bit issues a request for Clock Lane to enter the Ultra-Low Power State (ULPS). Before setting this field, HSCLKSETR.HSCLKRUN must be set to 0b. Writing 1b while Clock Lane is in ULPS is prohibited. Writing 0b has no effect.
23 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.5.2.2.8 Reset Control Register (DSI_LINK_RSTCR)

Access Size : 8, 16, 32 bits

Address : <DSI_LINK_base> + 0110h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	FCETXSTP
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SWRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	-	0h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	FCETXSTP	0h	RW	Force Tx Stop Mode When this bit is 1b, change to Transmit mode and Stop State is forced by ForceTxStopmode on all PPI Data Lanes. Setting this bit to 1b when SWRST = 0 is prohibited.
15 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	SWRST	0h	RW	Request for Software Reset Set this bit to 1b to request a software reset. Then, wait until you confirm the completion of the Reset procedure by RSTSR.SWRSTx=1. And set to 0b after that.

9.5.2.2.9 Reset Status Register (DSI_LINK_RSTSR)

Access Size : 8, 16, 32 bits

Address : <DSI_LINK_base> + 0114h

Initial Value : 0000_0F00h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DL0DIR	-	-	-	DLSTPST[3:0]				-	-	-	SWRSTV1	SWRSTIB	SWRSTAPB	SWRSTLP	SWRSTHS
Initial Value	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15	DL0DIR	0h	R	Direction (Tx or Rx) of PPI Data Lane 0 0b: PHY is in Tx Mode 1b: PHY is in Rx Mode This field is a copy of PLSR.DL0DIR.
14 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 8	DLSTPST[3:0]	Fh	R	Status of Stop State on Data Lane Bit 3: Stop State of Data Lane 3 Bit 2: Stop State of Data Lane 2 Bit 1: Stop State of Data Lane 1 Bit 0: Stop State of Data Lane 0 This field is a copy of PLSR.DLSTPST.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	SWRSTV1	0h	R	Status of Software Reset by RSTCR.SWRST at DSI_0_vclk1 0b: Reset procedure is not underway at DSI_0_vclk1. 1b: Reset procedure is underway DSI_0_vclk1.
3	SWRSTIB	0h	R	Status of Software Reset by RSTCR.SWRST at DSI_0_aclk 0b: Reset procedure is not underway at DSI_0_aclk. 1b: Reset procedure is underway at DSI_0_aclk.
2	SWRSTAPB	0h	R	Status of Software Reset by RSTCR.SWRST at DSI_0_pclk 0b: Reset procedure is not underway at DSI_0_pclk. 1b: Reset procedure is underway at DSI_0_pclk.
1	SWRSTLP	0h	R	Status of Software Reset by RSTCR.SWRST at DSI_0_lpclk 0b: Reset procedure is not underway at DSI_0_lpclk. 1b: Reset procedure is underway at DSI_0_lpclk.
0	SWRSTHS	0h	R	Status of Software Reset by RSTCR.SWRST at hsclk 0b: Reset procedure is not underway at hsclk. 1b: Reset procedure is underway at hsclk.

9.5.2.2.10 DSI Set Register (DSI_LINK_DSISETR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0120h

Initial Value : 80F1_0001h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EOTPEN	TEIDIR	SCREN	-	-	-	-	-	CRCEN[3:0]				-	-	-	ECCEN
Initial Value	1	0	0	0	0	0	0	0	1	1	1	1	0	0	0	1
R/W	RW	RW	RW	R	R	R	R	R	RW	RW	RW	RW	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MRPSZ[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	EOTPEN	1h	RW	HS Transfer EoTp Enable 0b: Transfer of EoTp is disabled. 1b: Transfer of EoTp is enabled. EoTp is always disabled in LP Transfer.
30	TEIDIR	0h	RW	Tearing Effect Input Direction External Tearing Effect Setting 0b: Rise edge detection. 1b: Fall edge detection.
29	SCREN	0h	RW	Scramble Enable 0b: Scrambling data is disabled. 1b: Scrambling data is enabled. This bit should not be set to “1” if the peer-device does not support a data scrambling function.
28 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 20	CRCEN[3:0]	Fh	RW	Enable Received CRC check for Virtual Channel Bit 3: Enable Received CRC check for Virtual Channel 3 (0b: Disabled, 1b: Enabled). Bit 2: Enable Received CRC check for Virtual Channel 2 (0b: Disabled, 1b: Enabled). Bit 1: Enable Received CRC check for Virtual Channel 1 (0b: Disabled, 1b: Enabled). Bit 0: Enable Received CRC check for Virtual Channel 0 (0b: Disabled, 1b: Enabled).
19 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16	ECCEN	1h	RW	Enable Received ECC Check 0b: Received ECC check is disabled. 1b: Received ECC check is enabled.
15 to 0	MRPSZ[15:0]	1h	RW	Maximum Return Packet Size If the returned Long Packet’s WC is over this value, the DSI-Tx module asserts the RXSR.MAXRPSZERR error. In that case, data will not be saved to the memory space. Setting this field to 0b is prohibited. Set this field to a value greater than or equal to the peripheral setting.

9.5.2.2.11 Receive Buffer Size Register (DSI_LINK_RXBUFSZR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0124h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	RXBUFSZ[3:0]				-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23 to 20	RXBUFSZ[3:0]	0h	R	Receive Buffer Size 0h: 128 B Buffer size = 128 * 2 RXBUFSZ
19 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.5.2.2.12 Tx Packet Payload Data 0 Register (DSI_LINK_TXPPD0R)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0160h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	0h	RW	Tx Packet Payload Data 0 When Sequence Operation Long Packet is set (SQCHxDSCyAR.FMT = 1) and the Packet Payload Data register is selected (SQCHxDSCyBR.DTSEL = 0h) for payload data access, the values in this register are used for Long Packet Payload. [7:0]: Data 0 [15:8]: Data 1 [23:16]: Data 2 [31:24]: Data 3

9.5.2.2.13 Tx Packet Payload Data 1 Register (DSI_LINK_TXPPD1R)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0164h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	0h	RW	Tx Packet Payload Data 1 When Sequence Operation Long Packet is set (SQCHxDSCyAR.FMT = 1) and the Packet Payload Data register is selected (SQCHxDSCyBR.DTSEL = 0h) for payload data access, the values in this register are used for Long Packet Payload. [7:0]: Data 4 [15:8]: Data 5 [23:16]: Data 6 [31:24]: Data 7

9.5.2.2.14 Tx Packet Payload Data 2 Register (DSI_LINK_TXPPD2R)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0168h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	0h	RW	Tx Packet Payload Data 2 When Sequence Operation Long Packet is set (SQCHxDSCyAR.FMT = 1) and the Packet Payload Data register is selected (SQCHxDSCyBR.DTSEL = 0h) for payload data access, the values in this register are used for Long Packet Payload. [7:0]: Data 8 [15:8]: Data 9 [23:16]: Data 10 [31:24]: Data 11

9.5.2.2.15 Tx Packet Payload Data 3 Register (DSI_LINK_TXPPD3R)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 016Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	0h	RW	Tx Packet Payload Data 3 When Sequence Operation Long Packet is set (SQCHxDSCyAR.FMT = 1) and the Packet Payload Data register is selected (SQCHxDSCyBR.DTSEL = 0h) for payload data access, the values in this register are used for Long Packet Payload. [7:0]: Data 12 [15:8]: Data 13 [23:16]: Data 14 [31:24]: Data 15

9.5.2.2.16 Rx Status Register (DSI_LINK_RXSR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0200h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	RXAKE	-	ECCERR1B	-	MAXRPSZERR	NORETERR	PRESPTOERR	RXOVFERR	IBERR	CRCERR	WCERR	-	UEXPKTERR	ECCERR	MLFERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EXTTEDET	RXACK	RXTE	RXRTRG	RXUK5TRG	RXEOTP	-	RXRESP	-	-	-	-	-	TATO	LRXHTO	BTAREQEND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30	RXAKE	0h	R	Rx Acknowledge and Error Report When this value is 1b, the Acknowledge and Error Report Packet is received. Write 1b to the RXSCR.RXAKE register to clear this field.
29	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	ECCERR1B	0h	R	ECC Error (1 bit) When this value is 1b, the Rx ECC Error (1 bit) is detected. Write 1b to the RXSCR.ECCERR1B register to clear this field.
27	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
26	MAXRPSZERR	0h	R	Maximum Return Packet Size Error When this value is 1b, the WC value of Rx Long Packet is larger than DSISETR.MRPSZ. Write 1b to the RXSCR.MAXRPSZERR register to clear this field.
25	NORETERR	0h	R	Nothing Return Error When this value is 1b, no trigger or packet is returned in BTA period. Write 1b to the RXSCR.NORETERR register to clear this field.
24	PRESPTOERR	0h	R	Peripheral Response Timeout Error When this value is 1b, the Rx Peripheral Response Timeout is detected after the transition to Rx mode along with Bus Turn-Around and within the time specified by PRESPTOBTASETR.PRESPTOBTA, PRESPTOLPSETR.PRESPTOLPR, PRESPTOLPSETR.PRESPTOLPW, PRESPTOLPSETR.PRESPTOHSR or PRESPTOLPSETR.PRESPTOHSW. The value is selected from kind of BTA. Write 1b to the RXSCR.PRESPTOERR register to clear this field.
23	RXOVFERR	0h	R	Receive Buffer Overflow Error When this value is 1b, the Buffer Overflow Error is detected at Rx Long Packet. Write 1b to the RXSCR.RXOVFERR register to clear this field.
22	IBERR	0h	R	Internal Bus Error When this value is 1b, the Internal Bus Write had failed to respond. Write 1b to the RXSCR.IBERR register to clear this field.
21	CRCERR	0h	R	CRC Error When this value is 1b, the Rx CRC Error is detected. Write 1b to the RXSCR.CRCERR register to clear this field.
20	WCERR	0h	R	Word Count Error When this value is 1b, the Rx packet's actual WC length is shorter than the packet header's WC. Write 1b the to RXSCR.WCERR register to clear this field.
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18	UEXPKTERR	0h	R	Unexpected Packet Error When this value is 1b, an unexpected packet is received. "Unexpected Packet" means unexpected DT for receive or an unexpected response for receive. Write 1b to the RXSCR.UEXPKTERR register to clear this field.

Bit	Bit Name	Initial Value	R/W	Description
17	ECCERR	0h	R	ECC Error (over 1 bit) When this value is 1b, the Rx ECC Error (over 1 bit) is detected. Write 1b to the RXSCR.ECCERR register to clear this field.
16	MLFERR	0h	R	Malform Error When this value is 1b, the Rx packet shorter than 4 bytes is received. Write 1b to the RXSCR.MLFERR register to clear this field.
15	EXTTEDET	0h	R	External Tearing Effect Detect When this value is 1b, External Tearing Effect is detected. Write 1b to the RXSCR.EXTTEDET register to clear this field.
14	RXACK	0h	R	Rx ACK Trigger When this value is 1b, the ACK trigger is received. Write 1b to the RXSCR.RXACK register to clear this field.
13	RXTE	0h	R	Rx Tearing Effect Trigger When this value is 1b, Tearing Effect Trigger is received. Write 1b to the RXSCR.RXTE register to clear this field.
12	RXRTRG	0h	R	Rx Reset Trigger When this value is 1b, Reset Trigger is received. Normally, the DSI-Tx module does not receive Reset Trigger. Write 1b to the RXSCR.RXRTRG register to clear this field.
11	RXUK5TRG	0h	R	Rx Unknown-5 Trigger When this value is 1b, Unknown-5 Trigger is received. Normally, the DSI-Tx module does not receive Unknown-5 Trigger. Write 1b to the RXSCR.RXUK5TRG register to clear this field.
10	RXEOTP	0h	R	Rx EoTp When this value is 1b, the EoTp is received. Write 1b to the RXSCR.RXEOTP register to clear this field.
9	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	RXRESP	0h	R	Rx Response Packet When this value is 1b, the Response Packet is received. Write 1b to the RXSCR.RXRESP register to clear this field.
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	TATO	0h	R	Turnaround Acknowledge Timeout When this value is 1b, the Turnaround Acknowledge Timeout (TA_TO) is detected. Write 1b to the RXSCR.TATO register to clear this field.
1	LRXHTO	0h	R	LP-RX Host Processor Timeout When this value is 1b, the LP-RX Host Processor Timeout (LRX-H_TO) is detected. Write 1b to RXSCR.LRXHTO register to clear this field.
0	BTAREQEN D	0h	R	BTA Request End When this value is 1b, the completion of the transmission request which includes Rx is detected. This bit is set if all BTA requests are finished including error. The BTA request status is indicated by RXSR[31:1]. Write 1b to the RXSCR.BTAREQEND register to clear this field.

9.5.2.2.17 Rx Status Clear Register (DSI_LINK_RXSCR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0204h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	RXAKE	-	ECCERR1B	-	MAXRPSZERR	NORETERR	PRESPTOERR	RXOVFERR	IBERR	CRCERR	WCERR	-	UEXPKTERR	ECCERR	MLFERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R0W	R	R0W	R	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R	R0W	R0W	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EXTTEDET	RXACK	RXTE	RXRTRG	RXUK5TRG	RXEOTP	-	RXRESP	-	-	-	-	-	TATO	LRXHTO	BTAREQEND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W	R0W	R0W	R0W	R0W	R0W	R	R0W	R	R	R	R	R	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30	RXAKE	0h	R0W	The read value is always 0b. RXSR.RXAKE Clear Set to 1b to clear RXSR.RXAKE. Writing 0b has no effect.
29	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	ECCERR1B	0h	R0W	The read value is always 0b. RXSR.ECCERR1B Clear Set to 1b to clear RXSR.ECCERR1B. Writing 0b has no effect.
27	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
26	MAXRPSZERR	0h	R0W	The read value is always 0b. RXSR.MAXRPSZERR Clear Set to 1b to clear RXSR.MAXRPSZERR. Writing 0b has no effect.
25	NORETERR	0h	R0W	The read value is always 0b. RXSR.NORETERR Clear Set to 1b to clear RXSR.NORETERR. Writing 0b has no effect.
24	PRESPTOERR	0h	R0W	The read value is always 0b. RXSR.PRESPTOERR Clear Set to 1b to clear RXSR.PRESPTOERR. Writing 0b has no effect.
23	RXOVFERR	0h	R0W	The read value is always 0b. RXSR.RXOVFERR Clear Set to 1b to clear RXSR.RXOVFERR. Writing 0b has no effect.
22	IBERR	0h	R0W	The read value is always 0b. RXSR.IBERR Clear Set to 1b to clear RXSR.IBERR. Writing 0b has no effect.
21	CRCERR	0h	R0W	The read value is always 0b. RXSR.CRCERR Clear Set to 1b to clear RXSR.CRCERR. Writing 0b has no effect.
20	WCERR	0h	R0W	The read value is always 0b. RXSR.WCERR Clear Set to 1b to clear RXSR.WCERR. Writing 0b has no effect.
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
18	UEXPKTER R	0h	R0W	The read value is always 0b. RXSR.UEXPKTERR Clear Set to 1b to clear RXSR.UEXPKTERR. Writing 0b has no effect.
17	ECCERR	0h	R0W	The read value is always 0b. RXSR.ECCERR Clear Set to 1b to clear RXSR.ECCERR. Writing 0b has no effect.
16	MLFERR	0h	R0W	The read value is always 0b. RXSR.MLFERR Clear Set to 1b to clear RXSR.MLFERR. Writing 0b has no effect.
15	EXTTEDET	0h	R0W	The read value is always 0b. RXSR.EXTTTEDET Clear Set to 1b to clear RXSR.EXTTTEDET. Writing 0b has no effect.
14	RXACK	0h	R0W	The read value is always 0b. RXSR.RXACK Clear Set to 1b to clear RXSR.RXACK. Writing 0b has no effect.
13	RXTE	0h	R0W	The read value is always 0b. RXSR.RXTE Clear Set to 1b to clear RXSR.RXTE. Writing 0b has no effect.
12	RXRTRG	0h	R0W	The read value is always 0b. RXSR.RXRTRG Clear Set to 1b to clear RXSR.RXRTRG. Writing 0b has no effect.
11	RXUK5TRG	0h	R0W	The read value is always 0b. RXSR.RXUK5TRG Clear Set to 1b to clear RXSR.RXUK5TRG. Writing 0b has no effect.
10	RXEOTP	0h	R0W	The read value is always 0b. RXSR.RXEOTP Clear Set to 1b to clear RXSR.RXEOTP. Writing 0b has no effect.
9	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	RXRESP	0h	R0W	The read value is always 0b. RXSR.RXRESP Clear Set to 1b to clear RXSR.RXRESP. Writing 0b has no effect.
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	TATO	0h	R0W	The read value is always 0b. RXSR.TATO Clear Set to 1b to clear RXSR.TATO. Writing 0b has no effect.
1	LRXHTO	0h	R0W	The read value is always 0b. RXSR.LRXHTO Clear Set to 1b to clear RXSR.LRXHTO. Writing 0b has no effect.
0	BTAREQEN D	0h	R0W	The read value is always 0b. RXSR.BTAREQEND Clear Set to 1b to clear RXSR.BTAREQEND. Writing 0b has no effect.

9.5.2.2.18 Rx Interrupt Enable Register (DSI_LINK_RXIER)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0208h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	RXAKE	-	ECCERR1B	-	MAXRPSZERR	NORETERR	PRESPTOERR	RXOVFERR	IBERR	CRCERR	WCERR	-	UEXPKTERR	ECCERR	MLFERR
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	R	RW	R	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW

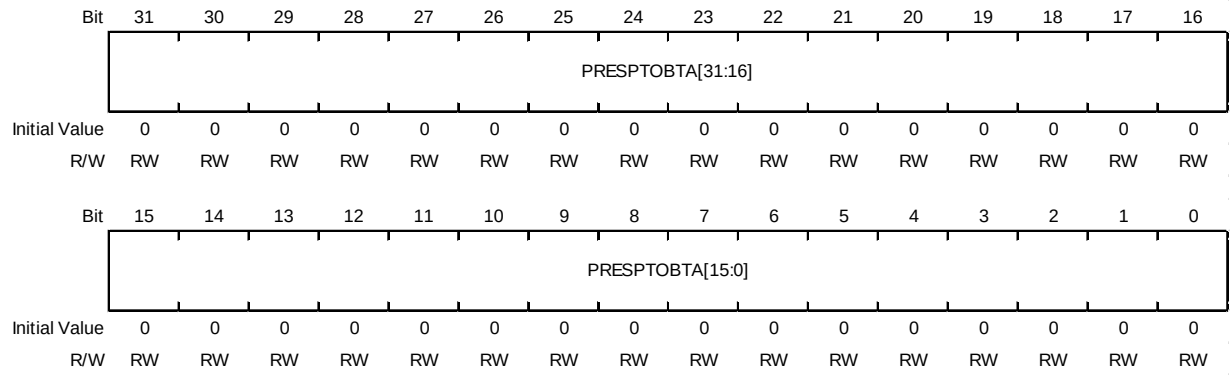
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EXTTEDET	RXACK	RXTE	RXRTRG	RXUK5TRG	RXEOTP	-	RXRESP	-	-	-	-	-	TATO	LRXHTO	BTAREQEND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	R	RW	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30	RXAKE	0h	RW	Interrupt enable for RXSR.RXAKE 0b: Disables interrupt dsi_int_rcv when RXSR.RXAKE = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.RXAKE = 1.
29	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	ECCERR1B	0h	RW	Interrupt enable for RXSR.ECCERR1B 0b: Disables interrupt dsi_int_rcv when RXSR.ECCERR1B = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.ECCERR1B = 1.
27	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
26	MAXRPSZERR	0h	RW	Interrupt enable for RXSR.MAXRPSZERR 0b: Disables interrupt dsi_int_rcv when RXSR.MAXRPSZERR = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.MAXRPSZERR = 1.
25	NORETERR	0h	RW	Interrupt enable for RXSR.NORETERR 0b: Disables interrupt dsi_int_rcv when RXSR.NORETERR = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.NORETERR = 1.
24	PRESPTOERR	0h	RW	Interrupt enable for RXSR.PRESPTOERR 0b: Disables interrupt dsi_int_rcv when RXSR.PRESPTOERR = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.PRESPTOERR = 1.
23	RXOVFERR	0h	RW	Interrupt enable for RXSR.RXOVFERR 0b: Disables interrupt dsi_int_rcv when RXSR.RXOVFERR = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.RXOVFERR = 1.
22	IBERR	0h	RW	Interrupt enable for RXSR.IBERR 0b: Disables interrupt dsi_int_rcv when RXSR.IBERR = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.IBERR = 1.
21	CRCERR	0h	RW	Interrupt enable for RXSR. CRCERR 0b: Disables interrupt dsi_int_rcv when RXSR.CRCERR = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.CRCERR = 1.
20	WCERR	0h	RW	Interrupt enable for RXSR. WCERR 0b: Disables interrupt dsi_int_rcv when RXSR.WCERR = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.WCERR = 1.
19	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
18	UEXPKTERR	0h	RW	Interrupt enable for RXSR.UEXPKTERR 0b: Disables interrupt dsi_int_rcv when RXSR.UEXPKTERR = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.UEXPKTERR = 1.
17	ECCERR	0h	RW	Interrupt enable for RXSR.ECCERR 0b: Disables interrupt dsi_int_rcv when RXSR.ECCERR = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.ECCERR = 1.
16	MLFERR	0h	RW	Interrupt enable for RXSR.MLFERR 0b: Disables interrupt dsi_int_rcv when RXSR.MLFERR = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.MLFERR = 1.

Bit	Bit Name	Initial Value	R/W	Description
15	EXTTEDET	0h	RW	Interrupt enable for RXSR.EXTTEDET 0b: Disables interrupt dsi_int_rcv when RXSR.EXTTEDET = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.EXTTEDET = 1.
14	RXACK	0h	RW	Interrupt enable for RXSR.RXACK 0b: Disables interrupt dsi_int_rcv when RXSR.RXACK = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.RXACK = 1.
13	RXTE	0h	RW	Interrupt enable for RXSR.RXTE 0b: Disables interrupt dsi_int_rcv when RXSR.RXTE = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.RXTE = 1.
12	RXRTRG	0h	RW	Interrupt enable for RXSR.RXRTRG 0b: Disables interrupt dsi_int_rcv when RXSR.RXRTRG = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.RXRTRG = 1.
11	RXUK5TRG	0h	RW	Interrupt enable for RXSR.RXUK5TRG 0b: Disables interrupt dsi_int_rcv when RXSR.RXUK5TRG = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.RXUK5TRG = 1.
10	RXEOTP	0h	RW	Interrupt enable for RXSR.RXEOTP 0b: Disables interrupt dsi_int_rcv when RXSR.RXEOTP = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.RXEOTP = 1.
9	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	RXRESP	0h	RW	Interrupt enable for RXSR.RXRESP 0b: Disables interrupt dsi_int_rcv when RXSR.RXRESP = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.RXRESP = 1.
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	TATO	0h	RW	Interrupt enable for RXSR.TATO 0b: Disables interrupt dsi_int_rcv when RXSR.TATO = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.TATO = 1.
1	LRXHTO	0h	RW	Interrupt enable for RXSR.LRXHTO 0b: Disables interrupt dsi_int_rcv when RXSR.LRXHTO = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.LRXHTO = 1.
0	BTAREQEND	0h	RW	Interrupt enable for RXSR.BTAREQEND 0b: Disables interrupt dsi_int_rcv when RXSR.BTAREQEND = 1. 1b: Enables interrupt dsi_int_rcv when RXSR.BTAREQEND = 1.

9.5.2.2.19 Peripheral Response Timeout BTA Set Register (DSI_LINK_PRESPTOBTASETR)

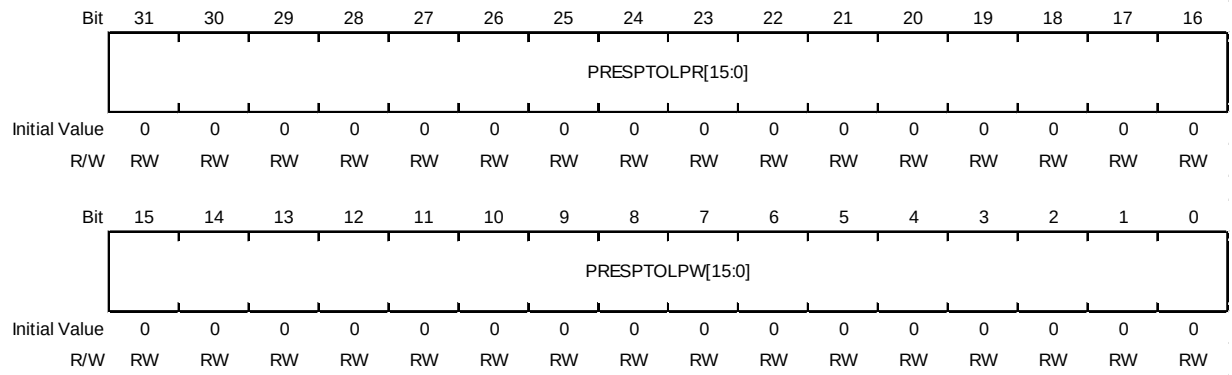
Access Size : 8, 16, 32 bits
Offset Address : <DSI_LINK_base> + 0210h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PRESPTOBTA [31:0]	0h	RW	Peripheral Response Timeout (Bus Turn Around) Count Set the timeout value to measure the interval between the start of Rx mode and the start of packet reception along with Bus Turn-Around while SQCHxDSCyAR.BTA = 3h. The limit time for timeout is calculated by the formula of PRESPTOBTA * (the period of DSI_0_lpclk). When RXSR.PRESPTOERR is 1b, a timeout is detected. Note that no timeout is detected when PRESPTOBTA = 0. Since the counting clock is asynchronous with packet reception, there is some jitter in the timeout.

9.5.2.2.20 Peripheral Response Timeout LP Set Register (DSI_LINK_PRESPTOLPSETR)

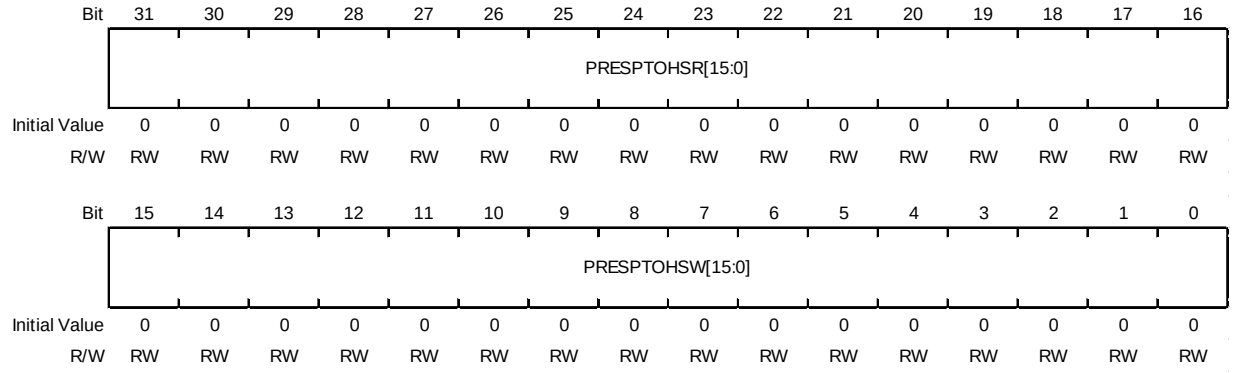
Access Size : 8, 16, 32 bits
Offset Address : <DSI_LINK_base> + 0214h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	PRESPTOLPR [15:0]	0h	RW	Peripheral Response Timeout (LPDT READ Request) Count Set the timeout value to measure the interval between the start of Rx mode and the start of packet reception along with Bus Turn-Around while SQCHxDSCyAR.BTA = 2h and SQCHxDSCyAR.SPD = 1b. The limit time for timeout is calculated by the formula of PRESPTOLPR * (the period of DSI_0_lpclk). When RXSR.PRESPTOERR is 1b, a timeout is detected. Note that no timeout is detected when PRESPTOLPR = 0. Since the counting clock is asynchronous with packet reception, there is some jitter in the timeout.
15 to 0	PRESPTOLPW [15:0]	0h	RW	Peripheral Response Timeout (LPDT WRITE Request) Count Set the timeout value to measure the interval between the start of Rx mode and the start of packet reception along with Bus Turn-Around while SQCHxDSCyAR.BTA = 01b and SQCHxDSCyAR.SPD = 1b. The limit time for timeout is calculated by the formula of PRESPTOLPW * (the period of DSI_0_lpclk). When RXSR.PRESPTOERR is 1b, a timeout is detected. Note that no timeout is detected when PRESPTOLPW = 0. Since the counting clock is asynchronous with packet reception, there is some jitter in the timeout.

9.5.2.2.21 Peripheral Response Timeout HS Set Register (DSI_LINK_PRESPTOHSSETR)

Access Size : 8, 16, 32 bits
Offset Address : <DSI_LINK_base> + 0218h
Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	PRESPTOHSR [15:0]	0h	RW	Peripheral Response Timeout (HS READ Request) Count Set the timeout value to measure the interval between the start of Rx mode and the start of packet reception along with Bus Turn-Around while SQCHxDSCyAR.BTA = 2h and SQCHxDSCyAR.SPD = 0b. The limit time for timeout is calculated by the formula of PRESPTOHSR * (the period of DSI_0_lpclk). When RXSR.PRESPTOERR is 1b, a timeout is detected. Note that no timeout is detected when PRESPTOHSR = 0. Since the counting clock is asynchronous with packet reception, there is some jitter in the timeout.
15 to 0	PRESPTOHSW [15:0]	0h	RW	Peripheral Response Timeout (HS WRITE Request) Count Set the timeout value to measure the interval between the start of Rx mode and the start of packet reception along with Bus Turn-Around while SQCHxDSCyAR.BTA = 1h and SQCHxDSCyAR.SPD = 0b. The limit time for timeout is calculated by the formula of PRESPTOHSW * (the period of DSI_0_lpclk). When RXSR.PRESPTOERR is 1b, a timeout is detected. Note that no timeout is detected when PRESPTOHSW = 0. Since the counting clock is asynchronous with packet reception, there is some jitter in the timeout.

9.5.2.2.22 Acknowledge and Error Report Packet Parameter Latest Info Register (DSI_LINK_AKEPLATIR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0220h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	VC3	VC2	VC1	VC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRRPTLAT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	VC3	0h	R	Virtual Channel Identifier 3 Latest This field indicates the Virtual Channel identifier of a received Acknowledge and Error Report Packet. 0b: Virtual Channel identifier is not 3. 1b: Virtual Channel identifier is 3. This field indicates the latest information. Whenever an Acknowledge and Error Report Packet is received, this field is updated.
18	VC2	0h	R	Virtual Channel Identifier 2 Latest This field indicates the Virtual Channel identifier of a received Acknowledge and Error Report Packet. 0b: Virtual Channel identifier is not 2. 1b: Virtual Channel identifier is 2. This field indicates the latest information. Whenever an Acknowledge and Error Report Packet is received, this field is updated.
17	VC1	0h	R	Virtual Channel Identifier 1 Latest This field indicates the Virtual Channel identifier of a received Acknowledge and Error Report Packet. 0b: Virtual Channel identifier is not 1. 1b: Virtual Channel identifier is 1. This field indicates the latest information. Whenever an Acknowledge and Error Report Packet is received, this field is updated.
16	VC0	0h	R	Virtual Channel Identifier 0 Latest This field indicates the Virtual Channel identifier of a received Acknowledge and Error Report Packet. 0b: Virtual Channel identifier is not 0. 1b: Virtual Channel identifier is 0. This field indicates the latest information. Whenever an Acknowledge and Error Report Packet is received, this field is updated.
15 to 0	ERRRPTLAT [15:0]	0h	R	Error Report Latest This field indicates the Error Report of a received Acknowledge and Error Report Packet. This field indicates the latest information. Whenever an Acknowledge and Error Report Packet is received, this field is updated.

9.5.2.2.23 Acknowledge and Error Report Packet Parameter Accumulate Status Register (DSI_LINK_AKEPACMSR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0224h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	VC3	VC2	VC1	VC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRRPTACM[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	VC3	0h	R	Virtual Channel Identifier 3 Accumulate This field is set to 1b when an Acknowledge and Error Report Packet whose Virtual Channel identifier = 3 is received. This field indicates accumulated information. This bit is cleared by AKEPSCR.VC3.
18	VC2	0h	R	Virtual Channel Identifier 2 Accumulate This field is set to 1b when an Acknowledge and Error Report Packet whose Virtual Channel identifier = 2 is received. This field indicates accumulated information. This bit is cleared by AKEPSCR.VC2.
17	VC1	0h	R	Virtual Channel Identifier 1 Accumulate This field is set to 1b when an Acknowledge and Error Report Packet whose Virtual Channel identifier = 1 is received. This field indicates accumulated information. This bit is cleared by AKEPSCR.VC1.
16	VC0	0h	R	Virtual Channel Identifier 0 Accumulate This field is set to 1b when an Acknowledge and Error Report Packet whose Virtual Channel identifier = 0 is received. This field indicates accumulated information. This bit is cleared by AKEPSCR.VC0.
15 to 0	ERRRPTACM [15:0]	0h	R	Error Report Accumulate Each bit of this field is set to 1b, when an Acknowledge and Error Report Packet whose each Error Report bit = 1 is received. This field indicates accumulated information. Each bit is cleared by the corresponding AKESCR.ERRRPTACM.

9.5.2.2.24 Acknowledge and Error Report Packet Parameter Status Clear Register (DSI_LINK_AKEPSCR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0228h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	VC3	VC2	VC1	VC0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W	R0W	R0W	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRRPTACM[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	VC3	0h	R0W	The read value is always 0b. AKEPACMSR.VC3 Clear Set to 1b to clear AKEPACMSR.VC3. Writing 0b has no effect.
18	VC2	0h	R0W	The read value is always 0b. AKEPACMSR.VC2 Clear Set to 1b to clear AKEPACMSR.VC2. Writing 0b has no effect.
17	VC1	0h	R0W	The read value is always 0b. AKEPACMSR.VC1 Clear Set to 1b to clear AKEPACMSR.VC1. Writing 0b has no effect.
16	VC0	0h	R0W	The read value is always 0b. AKEPACMSR.VC0 Clear Set to 1b to clear AKEPACMSR.VC0. Writing 0b has no effect.
15 to 0	ERRRPTACM[15:0]	0h	R0W	The read value is always 0b. Error Report Accumulate Clear Set to 1b to clear each AKEPACMSR.ERRRPTACM. Writing 0b has no effect.

9.5.2.2.25 Rx Result Saved Status Register (DSI_LINK_RXRSSR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0230h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	SLT3VLD	SLT2VLD	SLT1VLD	SLT0VLD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	SLT3VLD	0h	R	Slot 3 Valid Response packet is received, or information is saved to RXRSS3R. Write 1b to RXRSSCR.SLT3C to clear this bit. This bit and RXRSS3R are set if the next receive action is caused with SQCHxDSCyCR.ACTCODE = 03 again.
2	SLT2VLD	0h	R	Slot 2 Valid Response packet is received, or information is saved to RXRSS2R. Write 1b to RXRSSCR.SLT2C to clear this bit. This bit and RXRSS2R are set if the next receive action is caused with SQCHxDSCyCR.ACTCODE = 02 again.
1	SLT1VLD	0h	R	Slot 1 Valid Response packet is received, or information is saved to RXRSS1R. Write 1b to RXRSSCR.SLT1C to clear this bit. This bit and RXRSS1R are set if the next receive action is caused with SQCHxDSCyCR.ACTCODE = 01 again.
0	SLT0VLD	0h	R	Slot 0 Valid Response packet is received, or information is saved to RXRSS0R. Write 1b to RXRSSCR.SLT0C to clear this bit. This bit and RXRSS0R are set if the next receive action is caused with SQCHxDSCyCR.ACTCODE = 00 again.

9.5.2.2.26 Rx Result Saved Status Clear Register (DSI_LINK_RXRSSCR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0234h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	SLT3C	SLT2C	SLT1C	SLT0C
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	SLT3C	0h	R0W	The read value is always 0b. RXRSSR.SLT3VLD Clear Write 1b to clear RXRSSR.SLT3VLD. Writing 0b has no effect.
2	SLT2C	0h	R0W	The read value is always 0b. RXRSSR.SLT2VLD Clear Write 1b to clear RXRSSR.SLT2VLD. Writing 0b has no effect.
1	SLT1C	0h	R0W	The read value is always 0b. RXRSSR.SLT1VLD Clear Write 1b to clear RXRSSR.SLT1VLD. Writing 0b has no effect.
0	SLT0C	0h	R0W	The read value is always 0b. RXRSSR.SLT0VLD Clear Write 1b to clear RXRSSR.SLT0VLD. Writing 0b has no effect.

9.5.2.2.27 Rx Result Info Overwrite Status Register (DSI_LINK_RXRINFOOWSR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0238h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	SLT3INFOOW	SLT2INFOOW	SLT1INFOOW	SLT0INFOOW
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	SLT3INFOOW	0h	R	Slot 3 Information Overwrite This bit is a copy of RXRSS3R.INFOOW.
2	SLT2INFOOW	0h	R	Slot 2 Information Overwrite This bit is a copy of RXRSS2R.INFOOW.
1	SLT1INFOOW	0h	R	Slot 1 Information Overwrite This bit is a copy of RXRSS1R.INFOOW.
0	SLT0INFOOW	0h	R	Slot 0 Information Overwrite This bit is a copy of RXRSS0R.INFOOW.

9.5.2.2.28 Rx Result Info Overwrite Status Clear Register (DSI_LINK_RXRINFOOWSCR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 023Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	SLT3INFOOWC	SLT2INFOOWC	SLT1INFOOWC	SLT0INFOOWC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R0W	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	SLT3INFOOWC	0h	R0W	The read value is always 0b. RXRSS3R.INFOOW Clear Write 1b to clear RXRSS3R.INFOOW. Writing 0b has no effect. To prevent missing of RXRSS3R.INFOOW, clear RXRSS3R.INFOOW before clearing RXRSSR.SLT3VLD.
2	SLT2INFOOWC	0h	R0W	The read value is always 0b. RXRSS2R.INFOOW Clear Write 1b to clear RXRSS2R.INFOOW. Writing 0b has no effect. To prevent missing of RXRSS2R.INFOOW, clear RXRSS2R.INFOOW before clearing RXRSSR.SLT2VLD.
1	SLT1INFOOWC	0h	R0W	The read value is always 0b. RXRSS1R.INFOOW Clear Write 1b to clear RXRSS1R.INFOOW. Writing 0b has no effect. To prevent missing of RXRSS1R.INFOOW, clear RXRSS1R.INFOOW before clearing RXRSSR.SLT1VLD.
0	SLT0INFOOWC	0h	R0W	The read value is always 0b. RXRSS0R.INFOOW Clear Write 1b to clear RXRSS0R.INFOOW. Writing 0b has no effect. To prevent missing of RXRSS0R.INFOOW, clear RXRSS0R.INFOOW before clearing RXRSSR.SLT0VLD.

9.5.2.2.29 Rx Result Save Slot x Register (DSI_LINK_RXRSSxR) (x = 0, 1, 2, 3)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0240h + 0004h × (x)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INFOOW	RXAKE	RXCORERR	RXPKTDFAIL	RXFAIL	RXFATALERR	RXSUC	FMT	VC[1:0]		DT[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA1[7:0]								DATA0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	INFOOW	0h	R	Information Overwrite This bit is set if the information of RXRSSxR is updated when RXRSSR.SLTxVLD is 1b. Write 1b to RXRINFOOWSCR.SLTxINFOOWC to clear this bit.
30	RXAKE	0h	R	Receive Acknowledge and Error Report Packet RXSR.RXAKE is also reported.
29	RXCORERR	0h	R	Receive Correctable Error The received packet has a correctable error. RXSR.ECCERR1B is also reported.
28	RXPKTDFAIL	0h	R	Receive Packet Data Fail The packet header was saved correctly but data were not. One or more of the following bits are also reported. RXSR.CRCERR, RXSR.WCERR, RXSR.MAXRPSZERR, RXSR.UEXPKTERR, RXSR.RXOVFERR and RXSR.IBERR. When both of this bit and RXRESP are asserted, it means a redundant packet was received in BTA.
27	RXFAIL	0h	R	Receive Fail Expected reception did not proceed. One or more of the following bits are also reported. RXSR.PRESPTOERR, RXSR.ECCERR, RXSR.MLFERR and RXSR.NORETERR.
26	RXFATALERR	0h	R	Fatal Error A fatal timeout occurred during BTA. One or more of the following bits are also reported. FERRSR.TATO and FERRSR.LRXHTO.
25	RXSUC	0h	R	Receive Success When this value is 1b, the Response Packet or ACK trigger is received. One or more of the following bits are also reported. RXSR.RXRESP and RXSR.RXACK.
24	FMT	0h	R	Packet format of Rx Packet Header 0b: Short Packet 1b: Long Packet This field is valid when RXPKTDFAIL = 1 or (RXSUC = 1 and DT != 0).
23, 22	VC[1:0]	0h	R	Identifier of Virtual Channel of Rx Packet Header This field is valid when (RXSUC = 1 and DT != 0).
21 to 16	DT[5:0]	0h	R	Data Type of Rx Packet Header 00h: ACK trigger received 01h to 3Fh: Data Type This field is valid when RXSUC = 1.
15 to 8	DATA1[7:0]	0h	R	Data1 of Rx Packet Header When the Rx packet format is Long Packet, this value is the upper 8 bits of the word count. This field is valid when (RXSUC = 1 and DT != 0).
7 to 0	DATA0[7:0]	0h	R	Data0 of Rx Packet Header When Rx packet format is Long Packet, this value is the lower 8 bits of the word count. This field is valid when (RXSUC = 1 and DT != 0).

9.5.2.2.30 Rx Packet Payload Data 0 Register (DSI_LINK_RXPPD0R)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 02C0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	0h	R	Rx Packet Payload Data 0 When a Long Packet is received with SQCHxDSCyBR.DTSEL = 0h, the received packet payload is saved. [7: 0]: Data 0 [15: 8]: Data 1 [23: 16]: Data 2 [31: 24]: Data 3

9.5.2.2.31 Rx Packet Payload Data 1 Register (DSI_LINK_RXPPD1R)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 02C4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	0h	R	Rx Packet Payload Data 1 When a Long Packet is received with SQCHxDSCyBR.DTSEL = 0h, the received packet payload is saved. [7: 0]: Data 4 [15: 8]: Data 5 [23: 16]: Data 6 [31: 24]: Data 7

9.5.2.2.32 Rx Packet Payload Data 2 Register (DSI_LINK_RXPPD2R)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 02C8h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	0h	R	Rx Packet Payload Data 2 When a Long Packet is received with SQCHxDSCyBR.DTSEL = 0h, the received packet payload is saved. [7: 0]: Data 8 [15: 8]: Data 9 [23: 16]: Data 10 [31: 24]: Data 11

9.5.2.2.33 Rx Packet Payload Data 3 Register (DSI_LINK_RXPPD3R)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 02CCh

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	0h	R	Rx Packet Payload Data 3 When a Long Packet is received with SQCHxDSCyBR.DTSEL = 0h, the received packet payload is saved. [7: 0]: Data 12 [15: 8]: Data 13 [23: 16]: Data 14 [31: 24]: Data 15

9.5.2.2.34 HSTX Timeout Set Register (DSI_LINK_HSTXTOSETR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 02E0h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HTXTO[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HTXTO[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	HTXTO[31:0]	0h	RW	HS TX Timeout Count (HTX_TO) The limit time for timeout is calculated by the formula of HTXTO * (the period of high-speed serial UI * 32). When FERRSR.HTXTO is 1b, a timeout is detected. Note that no timeout is detected when HTXTO = 0.

9.5.2.2.35 LRX-H Timeout Set Register (DSI_LINK_LRXHTOSETR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 02E4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LRXHTO[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LRXHTO[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LRXHTO[31:0]	0h	RW	LP-RX Host Processor Timeout (LRX-H_TO) The limit time for timeout is calculated by the formula of LRXHTO * (the period of lpclk). When FERRSR.LRXHTO is 1b, a timeout is detected. Note that no timeout is detected when LRXHTO = 0.

9.5.2.2.36 TA Timeout Set Register (DSI_LINK_TATOSETR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 02E8h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TATO[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TATO[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TATO[31:0]	0h	RW	Turnaround Acknowledge Timeout (TA_TO) The limit time for timeout is calculated by the formula of TATO * (the period of lpclk). When FERRSR.TATO is 1b, a timeout is detected. Note that no timeout is detected when TATO = 0.

9.5.2.2.37 Fatal Error Status Register (DSI_LINK_FERRSR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0300h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	ERRCLP1S	ERRCLP0S	-	-	-	-	-	-	ERRCLP1	ERRCLP0	ERRCTRL	ERRSYNESC	ERRESC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	TATO	LRXHTO	HTXTO
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28	ERRCLP1S	0h	R	Status of Contention Error of LP1 (ErrContentionLP1)
27	ERRCLP0S	0h	R	Status of Contention Error of LP0 (ErrContentionLP0)
26 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	ERRCLP1	0h	R	Contention Error of LP1 (ErrContentionLP1) When this value is 1b, the Contention Error of LP1 (ErrContentionLP1) is detected on PPI Data Lane 0. Write 1b to the FERRSCR.ERRCLP1 register to clear this field.
19	ERRCLP0	0h	R	Contention Error of LP0 (ErrContentionLP0) When this value is 1b, the Contention Error of LP0 (ErrContentionLP0) is detected on PPI Data Lane 0. Write 1b to the FERRSCR.ERRCLP0 register to clear this field.
18	ERRCTRL	0h	R	Control Error (ErrControl) When this value is 1b, the Control Error (ErrControl) is detected on PPI Data Lane 0. Write 1b to the FERRSCR.ERRCTRL register to clear this field.
17	ERRSYNESC	0h	R	Sync Error of LPDT (ErrSyncEsc) When this value is 1b, the Sync Error of LPDT (ErrSyncEsc) is detected on PPI Data Lane 0. Write 1b to the FERRSCR.ERRSYNESC register to clear this field.
16	ERRESC	0h	R	Escape mode Entry Error (ErrEsc) When this value is 1b, the Escape mode Entry Error (ErrEsc) is detected on PPI Data Lane 0. Write 1b to the FERRSCR.ERRESC register to clear this field.
15 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	TATO	0h	R	Turnaround Acknowledge Timeout (TA_TO) When this value is 1b, the Turnaround Acknowledge Timeout (TA_TO) is detected. Write 1b to the FERRSCR.TATO register to clear this field.
1	LRXHTO	0h	R	LP-RX Host Processor Timeout (LRX-H_TO) When this value is 1b, the LP-RX Host Processor Timeout (LRX-H_TO) is detected. Write 1b to the FERRSCR.LRXHTO register to clear this field.
0	HTXTO	0h	R	HS TX Timeout (HTX_TO) When this value is 1b, the HS TX Timeout (HTX_TO) is detected. Write 1b to the FERRSCR.HTXTO register to clear this field.

9.5.2.2.38 Fatal Error Status Clear Register (DSI_LINK_FERRSCR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0304h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	ERRCLP1	ERRCLP0	ERRCTRL	ERRSYNESC	ERRESC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R0W	R0W	R0W	R0W	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	TATO	LRXHTO	HTXTO
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	ERRCLP1	0h	R0W	The read value is always 0b. FERRSR.ERRCLP1 Clear Set to 1b to clear FERRSR.ERRCLP1. Writing 0b has no effect.
19	ERRCLP0	0h	R0W	The read value is always 0. FERRSR.ERRCLP0 Clear Set to 1b to clear FERRSR.ERRCLP0. Writing 0b has no effect.
18	ERRCTRL	0h	R0W	The read value is always 0. FERRSR.ERRCTRL Clear Set to 1b to clear FERRSR.ERRCTRL. Writing 0b has no effect.
17	ERRSYNESC	0h	R0W	The read value is always 0. FERRSR.ERRSYNESC Clear Set to 1b to clear FERRSR.ERRSYNESC. Writing 0b has no effect.
16	ERRESC	0h	R0W	The read value is always 0. FERRSR.ERRESC Clear Set to 1b to clear FERRSR.ERRESC. Writing 0b has no effect.
15 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	TATO	0h	R0W	The read value is always 0b. FERRSR.TATO Clear Set to 1b to clear FERRSR.TATO. Writing 0b has no effect.
1	LRXHTO	0h	R0W	The read value is always 0b. FERRSR.LRXHTO Clear Set to 1b to clear FERRSR.LRXHTO. Writing 0b has no effect.
0	HTXTO	0h	R0W	The read value is always 0. FERRSR.HTXTO Clear Set to 1b to clear FERRSR.HTXTO. Writing 0b has no effect.

9.5.2.2.39 Fatal Error Interrupt Enable Register (DSI_LINK_FERRIER)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0308h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	ERRCLP1	ERRCLP0	ERRCTRL	ERRSYNESC	ERRESC
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	TATO	LRXHTO	HTXTO
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	ERRCLP1	0h	RW	Interrupt enable for the Contention Error of LP1 on PPI Data Lane 0 (ErrContentionLP1) 0b: Disables interrupt dsi_int_ferr when FERRSR.ERRCLP1 = 1 1b: Enables interrupt dsi_int_ferr when FERRSR.ERRCLP1 = 1
19	ERRCLP0	0h	RW	Interrupt enable for the Contention Error of LP0 on PPI Data Lane 0 (ErrContentionLP0) 0b: Disables interrupt dsi_int_ferr when FERRSR.ERRCLP0 = 1 1b: Enables interrupt dsi_int_ferr when FERRSR.ERRCLP0 = 1
18	ERRCTRL	0h	RW	Interrupt enable for the Control Error on PPI Data Lane 0 (ErrControl) 0b: Disables interrupt dsi_int_ferr when FERRSR.ERRCTRL = 1 1b: Enables interrupt dsi_int_ferr when FERRSR.ERRCTRL = 1
17	ERRSYNESC	0h	RW	Interrupt enable for the LPDT Sync Error on PPI Data Lane 0 (ErrSyncEsc) 0b: Disables interrupt dsi_int_ferr when FERRSR.ERRSYNESC = 1 1b: Enables interrupt dsi_int_ferr when FERRSR.ERRSYNESC = 1
16	ERRESC	0h	RW	Interrupt enable for the Escape mode Error on PPI Data Lane 0 (ErrEsc) 0b: Disables interrupt dsi_int_ferr when FERRSR.ERRESC = 1 1b: Enables interrupt dsi_int_ferr when FERRSR.ERRESC = 1
15 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	TATO	0h	RW	Interrupt enable for Turnaround Acknowledge Timeout (TA_TO) 0b: Disables interrupt dsi_int_ferr when FERRSR.TATO = 1 1b: Enables interrupt dsi_int_ferr when FERRSR.TATO = 1
1	LRXHTO	0h	RW	Interrupt enable for LP-RX Host Processor Timeout (LRX_H_TO) 0b: Disables interrupt dsi_int_ferr when FERRSR.LRXHTO = 1 1b: Enables interrupt dsi_int_ferr when FERRSR.LRXHTO = 1
0	HTXTO	0h	RW	Interrupt enable for HS TX Timeout (HTX_TO) 0b: Disables interrupt dsi_int_ferr when FERRSR.HTXTO = 1 1b: Enables interrupt dsi_int_ferr when FERRSR.HTXTO = 1

9.5.2.2.40 Clock Lane Stop Time Set Register (DSI_LINK_CLSTPTSETR)

Access Size : 8, 16, 32 bits
Offset Address : <DSI_LINK_base> + 0314h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CLKKPT[7:0]								CLKBFHT[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	CLKSTPT[9:0]										-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CLKKPT[7:0]	0h	RW	Clock Keep Time The time period over which the HS Clock Lane keeps active after HS packet transfer. The time is calculated by the formula of CLKKPT * (the period of high-speed serial UI * 32). This time corresponds to the D-PHY specification's ($T_{HS-TRAIL}$ + $T_{CLK-POST}$). The value must be greater than the one calculated. For information about each parameter, see **Table 9.5-4**. Set an appropriate value (other than 0) before HSCLKSETR.HSCLKRUN is set to "1".
23 to 16	CLKBFHT[7:0]	0h	RW	Clock Beforehand Time The time period over which Clock Lane is changed back to HS before HS packet transfer. If HS transfers are scheduled in LP, Clock Lane will start before HS transfer by this setting. The time is calculated by the formula of CLKBFHT * (the period of high-speed serial UI * 32). This time corresponds to the D-PHY specification's (T_{LPX}(For Clock Lane HS Entry) + $T_{CLK-PREPARE}$ + $T_{CLK-ZERO}$ + $T_{CLK-PRE}$). The value must be greater than the one calculated. For information about each parameter, see **Table 9.5-4**. Set an appropriate value (other than 0) before HSCLKSETR.HSCLKRUN is set to "1". The "T_{LPX}(For Clock Lane HS Entry)" means the period of LP-01 in HS entry sequence of Clock Lane. If this period is not same as normal T_{LPX} value of host device (in the range of 67% to 150% of the T_{LPX} value of peripheral device), use the actual period for this calculation.
15 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 2	CLKSTPT[9:0]	0h	RW	Clock Stop Time The time period over which the HS Clock Lane is changed to LP and back to HS. If Data Lanes exit HS and the time of CLKKPT expires, and the scheduled time till the next HS transfer is more than this setting, Clock Lane will be changed to LP. This setting has no meaning if HSCLKSETR.HSCLKMODE is 1b or if HSCLKSETR.HSCLKRUN is 0b. The time is calculated by the formula of CLKSTPT * (the period of high-speed serial UI * 32) This time corresponds to the D-PHY specification's ($T_{HS-TRAIL}$ + $T_{CLK-POST}$ + $T_{CLK-TRAIL}$ + $T_{HS-EXIT}$ + T_{LPX} * 2 + T_{LPX}(For Clock Lane HS Entry) + $T_{CLK-PREPARE}$ + $T_{CLK-ZERO}$ + $T_{CLK-PRE}$ + T_{LPX}(For Data Lane HS Entry) + $T_{HS-PREPARE}$ + $T_{HS-ZERO}$ + $T_{HS-SYNC}$). For information about each parameter, see **Figure 9.5-2** and **Table 9.5-4**. Of these parameters, it is necessary to set the time actually set in CLKKPT for ($T_{HS-TRAIL}$ + $T_{CLK-POST}$) and that actually set in CLKBFHT for (T_{LPX}(For Clock Lane HS Entry) + $T_{CLK-PREPARE}$ + $T_{CLK-ZERO}$ + $T_{CLK-PRE}$). Set the rounded-up value plus at least 3. Setting this field to "0" is prohibited. The "T_{LPX}(For Clock Lane HS Entry)" means the period of LP-01 in HS entry sequence of Clock Lane. And the "T_{LPX}(For Data Lane HS Entry)" means the period of LP-01 in HS entry sequence of Data Lane. If these periods are not same as normal T_{LPX} value of host device (in the range of 67% to 150% of the T_{LPX} value of peripheral device), use the actual periods for this calculation.
1, 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

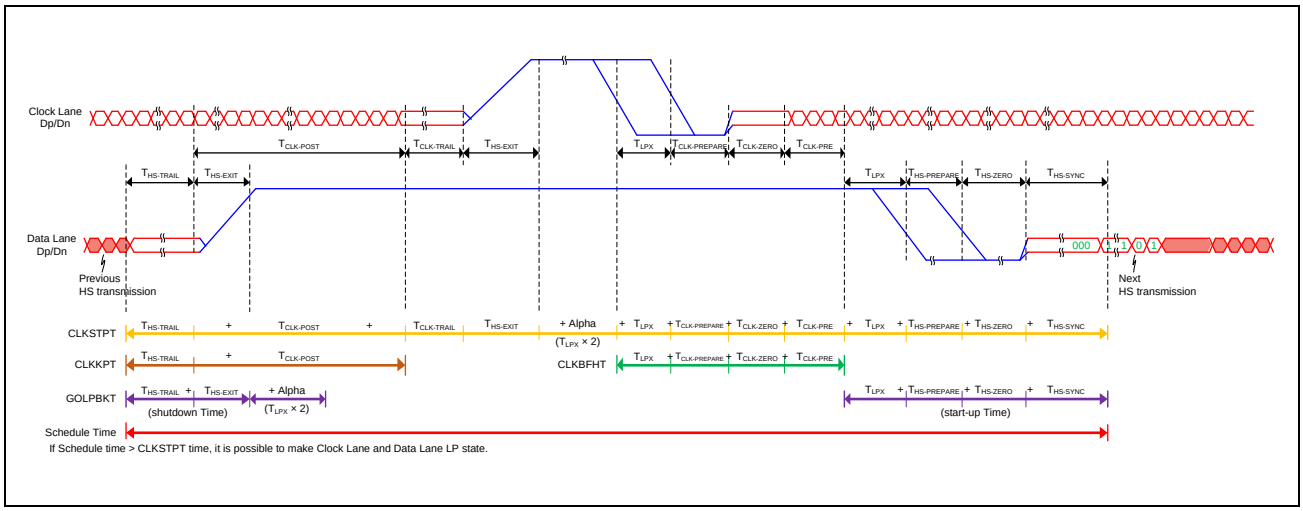


Figure 9.5-2 Switching the Clock Lane between Clock Transmission and Low-Power Mode

9.5.2.2.41 LP Transition Time Set Register (DSI_LINK_LPTRNSTSETR)

Access Size : 8, 16, 32 bits
Offset Address : <DSI_LINK_base> + 0318h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	GOLPBKT[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	GOLPBKT[9:0]	0h	RW	Go LP and Back Time The time period over which the HS Data Lanes are changed to LP and back to HS. If the scheduled time between HS transfers is more than this setting, Data Lanes will be changed to LP. The time is calculated by the formula of GOLPBKT * (the period of high-speed serial 8 * UI). This time corresponds to D-PHY specification's ($T_{HS-TRAIL}$ + $T_{HS-EXIT}$ + T_{LPX} * 2 + T_{LPX}(For Data Lane HS Entry) + $T_{HS-PREPARE}$ + $T_{HS-ZERO}$ + $T_{HS-SYNC}$). For information about each parameter, see **Figure 9.5-3** and **Table 9.5-4**. The unit of this setting is 8 UIs, so if the result of this formula contains a fraction of less than 8 UIs, round it up. Set the rounded-up value plus at least 3. The "T_{LPX}(For Data Lane HS Entry)" means the period of LP-01 in HS entry sequence of Data Lane. If this period is not same as normal T_{LPX} value of host device (in the range of 67% to 150% of the T_{LPX} value of peripheral device), use the actual period for this calculation.

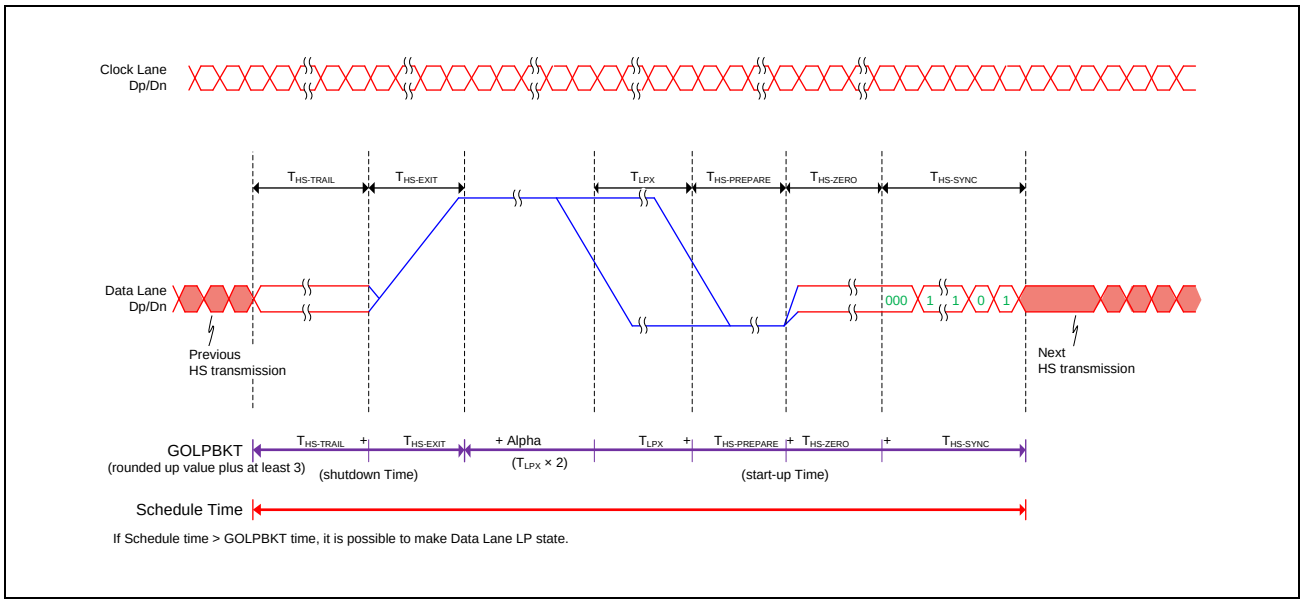


Figure 9.5-3 Switching the Data Lane between High-Speed Mode and Low-Power Mode

9.5.2.2.42 Physical Lane Status Register (DSI_LINK_PLSR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0320h

Initial Value : 0000_0FF3h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	DLFROMULPS	DLTOULPS	CLHS2LP	CLLP2HS	CLFROMULPS	CLTOULPS	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DL0DIR	-	DL0TX2RX	DL0RX2TX	DLSTPST[3:0]				DLULPSACTN[3:0]				DL0RXULPSESC	DL0RXLPDTESC	CLSTPST	CLULPSACTN
Initial Value	0	0	0	0	1	1	1	1	1	1	1	1	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29	DLFROMULPS	0h	R	From ULPS (Data Lane) When this value is 1b, all PPI Data Lanes enabled by TXSETR.NUMLANEUSE were changed from ULPS to Stop State by exiting ULPS. Write 1b to the PLSCR.DLFROMULPS register to clear this field.
28	DLTOULPS	0h	R	To ULPS (Data Lane) When this value is 1b, all PPI Data Lanes enabled by TXSETR.NUMLANEUSE were changed from Stop State to ULPS by entering ULPS. Write 1b to the PLSCR.DLTOULPS register to clear this field.
27	CLHS2LP	0h	R	HS to LP (Clock Lane) When this value is 1b, a transition of Clock Lane from HS to LP is detected. Checking this bit during non-continuous clock mode operation has no meaning. Therefore, ignore this bit when HSCLKSETR.HSCLKMODE = 0b. Write 1b to the PLSCR.CLHS2LP register to clear this field.
26	CLLP2HS	0h	R	LP to HS (Clock Lane) When this value is 1b, a transition of Clock Lane from LP to HS is detected. Checking this bit during non-continuous clock mode operation has no meaning. Therefore, ignore this bit when HSCLKSETR.HSCLKMODE = 0b. Write 1b to the PLSCR.CLLP2HS register to clear this field.
25	CLFROMULPS	0h	R	From ULPS (Clock Lane) When this value is 1b, Clock Lane exiting ULPS is detected. Write 1b to the PLSCR.CLFROMULPS register to clear this field.
24	CLTOULPS	0h	R	To ULPS (Clock Lane) When this value is 1b, Clock Lane entering ULPS is detected. Write 1b to the PLSCR.CLTOULPS register to clear this field.
23 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15	DL0DIR	0h	R	Direction (Tx or Rx) of Data Lane 0 0b: PHY is in Tx Mode 1b: PHY is in Rx Mode This field indicates Direction of PPI Data Lane 0 after synchronization to lpclk. This field does not indicate the current status while lpclk is not running.
14	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	DL0TX2RX	0h	R	Tx to Rx on Data Lane 0 When this value is 1b, a transition of Direction from 0 (Tx) to 1 (Rx) is detected on Data Lane 0. Write 1b to the PLSCR.DL0TX2RX register to clear this field.
12	DL0RX2TX	0h	R	Rx to Tx on Data Lane 0 When this value is 1b, a transition of Direction from 1 (Rx) to 0 (Tx) is detected on Data Lane 0. Write 1b to the PLSCR.DL0RX2TX register to clear this field.

Bit	Bit Name	Initial Value	R/W	Description
11 to 8	DLSTPST[3:0]	Fh	R	Stop State of Data Lane bit 3: Stop State of Data Lane 3 bit 2: Stop State of Data Lane 2 bit 1: Stop State of Data Lane 1 bit 0: Stop State of Data Lane 0 This field indicates Stop State of a PPI Data Lane after synchronization to lpclk. This field does not indicate the current status while lpclk is not running.
7 to 4	DLULPSACTN [3:0]	Fh	R	UlpsActiveNot of Data Lane bit 3: UlpsActiveNot of Data Lane 3 bit 2: UlpsActiveNot of Data Lane 2 bit 1: UlpsActiveNot of Data Lane 1 bit 0: UlpsActiveNot of Data Lane 0 This field indicates UlpsActiveNot of a PPI Data Lane after synchronization to lpclk. This field does not indicate the current status while lpclk is not running.
3	DL0RXULPSESC	0h	R	RxUlpsEsc of Data Lane 0 This field indicates RxUlpsEsc of PPI Data Lane 0 after synchronization to lpclk. This field does not indicate the current status while lpclk is not running.
2	DL0RXLPDTESC	0h	R	RxLpdtEsc of Data Lane 0 This field indicates RxLpdtEsc of PPI Data Lane 0 after synchronization to lpclk. This field does not indicate the current status while lpclk is not running.
1	CLSTPST	1h	R	Stop State of PPI Clock Lane This field indicates Stop State of the PPI Clock Lane after synchronization to lpclk. This field does not indicate the current status while lpclk is not running.
0	CLULPSACTN	1h	R	UlpsActiveNot of PPI Clock Lane This field indicates UlpsActiveNot of the PPI Clock Lane after synchronization to lpclk. This field does not indicate the current status while lpclk is not running.

9.5.2.2.43 Physical Lane Status Clear Register (DSI_LINK_PLSCR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0324h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	DLFROMULPS	DLTOULPS	CLHS2LP	CLLP2HS	CLFROMULPS	CLTOULPS	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R0W	R0W	R0W	R0W	R0W	R0W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	DL0TX2RX	DL0RX2TX	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R0W	R0W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29	DLFROMULPS	0h	R0W	The read value is always 0b. PLSR.DLFROMULPS Clear Set to 1b to clear PLSR.DLFROMULPS. Writing 0b has no effect.
28	DLTOULPS	0h	R0W	The read value is always 0b. PLSR.DLTOULPS Clear Set to 1b to clear PLSR.DLTOULPS. Writing 0b has no effect.
27	CLHS2LP	0h	R0W	The read value is always 0b. PLSR.CLHS2LP Clear Set to 1b to clear PLSR.CLHS2LP. Writing 0b has no effect.
26	CLLP2HS	0h	R0W	The read value is always 0b. PLSR.CLLP2HS Clear Set to 1b to clear PLSR.CLLP2HS. Writing 0b has no effect.
25	CLFROMULPS	0h	R0W	The read value is always 0b. PLSR.CLFROMULPS Clear Set to 1b to clear PLSR.CLFROMULPS. Writing 0b has no effect.
24	CLTOULPS	0h	R0W	The read value is always 0b. PLSR.CLTOULPS Clear Set to 1b to clear PLSR.CLTOULPS. Writing 0b has no effect.
23 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	DL0TX2RX	0h	R0W	The read value is always 0b. PLSR.DL0TX2RX Clear Set to 1b to clear PLSR.DL0TX2RX. Writing 0b has no effect.
12	DL0RX2TX	0h	R0W	The read value is always 0b. PLSR.DL0RX2TX Clear Set to 1b to clear PLSR.DL0RX2TX. Writing 0b has no effect.
11 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.5.2.2.44 Physical Lane Interrupt Enable Register (DSI_LINK_PLIER)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0328h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	DLFROMULPS	DLTOULPS	CLHS2LP	CLLP2HS	CLFROMULPS	CLTOULPS	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	DL0TX2RX	DL0RX2TX	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29	DLFROMULPS	0h	RW	Interrupt enable for all Data Lane transition from ULPS 0b: Disables interrupt dsi_int_ppi when PLSR.DLFROMULPS = 1. 1b: Enables interrupt dsi_int_ppi when PLSR.DLFROMULPS = 1.
28	DLTOULPS	0h	RW	Interrupt enable for all Data Lane transition to ULPS 0b: Disables interrupt dsi_int_ppi when PLSR.DLTOULPS = 1. 1b: Enables interrupt dsi_int_ppi when PLSR.DLTOULPS = 1.
27	CLHS2LP	0h	RW	Interrupt enable for Clock Lane transition to LP from HS 0b: Disables interrupt dsi_int_ppi when PLSR.CLHS2LP = 1. 1b: Enables interrupt dsi_int_ppi when PLSR.CLHS2LP = 1.
26	CLLP2HS	0h	RW	Interrupt enable for Clock Lane transition to HS from LP 0b: Disables interrupt dsi_int_ppi when PLSR.CLLP2HS = 1. 1b: Enables interrupt dsi_int_ppi when PLSR.CLLP2HS = 1.
25	CLFROMULPS	0h	RW	Interrupt enable for Clock Lane transition from ULPS 0b: Disables interrupt dsi_int_ppi when PLSR.CLFROMULPS = 1. 1b: Enables interrupt dsi_int_ppi when PLSR.CLFROMULPS = 1.
24	CLTOULPS	0h	RW	Interrupt enable for Clock Lane transition to ULPS 0b: Disables interrupt dsi_int_ppi when PLSR.CLTOULPS = 1. 1b: Enables interrupt dsi_int_ppi when PLSR.CLTOULPS = 1.
23 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	DL0TX2RX	0h	RW	Interrupt enable for Data Lane 0 direction change from Tx to Rx 0b: Disables interrupt dsi_int_ppi when PLSR.DL0TX2RX = 1. 1b: Enables interrupt dsi_int_ppi when PLSR.DL0TX2RX = 1.
12	DL0RX2TX	0h	RW	Interrupt enable for Data Lane 0 direction change from Rx to Tx 0b: Disables interrupt dsi_int_ppi when PLSR.DL0RX2TX = 1. 1b: Enables interrupt dsi_int_ppi when PLSR.DL0RX2TX = 1.
11 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.5.2.2.45 Video-Input Channel 1 Set 0 Register (DSI_LINK_VICH1SET0R)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0400h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VSEN	-	HFPNOLP	HBPNOLP	HSANOLP	-	-	-	-	-	-	VSTPAFT	VSTART
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R0W	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13	-	0h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
12	VSEN	0h	RW	Video Skew Calibration Enable 0b: No skew calibration is issued by this channel. 1b: Periodic skew calibration is issued in the first line of VBP. When periodic skew calibration is enabled, set HSCLKSETR.HSCLKMODE = 1.
11	-	0h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
10	HFPNOLP	0h	RW	HFP period No LP Specifies whether suppress to transition to LP during the HFP period. 0b: Does not suppress transition to LP during the period 1b: Suppresses transition to LP during the period and keeps HS. HS is kept by sending a blanking packet.
9	HBPNOLP	0h	RW	HBP period No LP Specifies whether suppress transition to LP during the HBP period. 0b: Does not suppress transition to LP during the period 1b: Suppresses transition to LP during the period and keeps HS. HS is kept by sending a blanking packet.
8	HSANOLP	0h	RW	HSA period No LP Specifies whether suppress transition to LP during the HSA period. When TXESYNC=0, this field has no meaning. 0b: Does not suppress transition to LP during the period 1b: Suppresses transition to LP during the period and keeps HS. HS is kept by sending a blanking packet.
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	-	0h	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
1	VSTPAFT	0h	R0W	The read value is always 0b. Video-Input signals stop after Video-Input Operation stop Stop Video-Input Operation. Video-Input Operation will end at the end of video-frame. After Video-Input Operation end, stop Video-Input signals. If Video-Input Operation is stopped, VICH1SR.RUNNING bit will change to 0b. Set 1b to both VSTPAFT and VSTART is prohibited. Set 1b during VICH1SR.RUNNING=0 is prohibited. Writing 0b has no effect.

Bit	Bit Name	Initial Value	R/W	Description
0	VSTART	0h	R0W	The read value is always 0b. Video-Input Operation Start Write 1b to start Video-Input Operation. Setting both VSTART and VSTPAFT to 1b is prohibited. Setting this bit to 1b while VICH1SR.RUNNING = 1 is prohibited. Writing 0b has no effect.

9.5.2.2.46 Video-Input Channel 1 Set 1 Register (DSI_LINK_VICH1SET1R)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0404h

Initial Value : 1093_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSPC	-	BPP[5:0]						CHBUFSZ[3:0]				-	-	-	-
Initial Value	0	0	0	1	0	0	0	0	1	0	0	1	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	DLY[11:0]												-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	CSPC	0h	R	Color Space of Input Video Pixel This field reflects the color space of Data Type selected by VICH1PPSETR.DT. 0h: RGB 1h: Reserved
30	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29 to 24	BPP[5:0]	10h	R	Bit Per Pixel of Input Video Pixel This field reflects bits per pixel of Data Type selected by VICH1PPSETR.DT. 10h: 16 bpp 12h: 18 bpp 18h: 24 bpp, 18 bpp (Loosely Packed) (RGB)
23 to 20	CHBUFSZ [3:0]	9h	R	Channel Buffer Size 9h: 64 kB Buffer size = 128 x 2 CHBUFSZ
19, 18	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
17, 16	-	3h	R	Reserved Whenever it is read, 1b is read. The written value will be ignored.
15, 14	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 2	DLY[11:0]	0h	RW	Delay Value The delay time is calculated by the formula of DLY * (the period of high-speed serial UI * 32). Setting to this field to 0b is prohibited. If VICH1SET0R.HFPNOLP = 1, set this field to a value greater than the HFP period. If VICH1SET0R.HBPNOLP = 1, set this field to a value greater than the HBP period. If VICH1SET0R.HSANOLP = 1, set this field to a value greater than the HSA period.
1, 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.5.2.2.47 Video-Input Channel 1 Status Register (DSI_LINK_VICH1SR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0410h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	VBUFOVF	VBUFUDF	-	TIMERR	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	VIRDY	RUNNING	STOP	START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23	VBUFOVF	0h	R	Video-Input Buffer Overflow Error A data overflow has occurred in Video-Input Buffer. Write 1b to the VICH1SCR.VBUFOVF register to clear this field.
22	VBUFUDF	0h	R	Video-Input Buffer Underflow Error A data underflow has occurred in Video-Input Buffer. Write 1b to the VICH1SCR.VBUFUDF register to clear this field.
21	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
20	TIMERR	0h	R	Timing Error A timing error has occurred during Video-Input Operation. Write 1b to the VICH1SCR.TIMERR register to clear this field.
19 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	VIRDY	0h	R	Video-Input Signal Accept Ready The value will be set to 1b when the input signal can be accepted by Video-Input Operation channel 1 ready.
2	RUNNING	0h	R	Video-Input Operation channel 1 Transfer Running The value will be set to 1b while Video-Input Operation channel 1 transfer is in progress. Changing the values of the VICH1SET0R, VICH1SET1R, VICH1PPSETR, VICH1VSSETR, VICH1VPSETR, VICH1HSSETR and VICH1HPSETR is prohibited while the transfer is in progress (VICH1SR.RUNNING = 1).
1	STOP	0h	R	Video Input Operation channel 1 Transfer Stop Write 1b to the VICH1SCR.STOP register to clear this field.
0	START	0h	R	Video Input Operation channel 1 Transfer Start. Write 1b to the VICH1SCR.START register to clear this field.

9.5.2.2.48 Video-Input Channel 1 Status Clear Register (DSI_LINK_VICH1SCR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0414h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	VBUFOVF	VBUFUDF	-	TIMERR	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R0W	R0W	R0W	R0W	R0W	R	R	R0W	R0W	R0W	R0W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	VIRDY	-	STOP	START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R0W	R	R	R	R	R	R	R	R	R	R	R0W	R	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 26	-	All 0	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
25, 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23	VBUFOVF	0h	R0W	The read value is always 0b. VICH1SR.VBUFOVF Clear Set to 1b to clear VICH1SR.VBUFOVF. Writing 0b has no effect.
22	VBUFUDF	0h	R0W	The read value is always 0b. VICH1SR.VBUFUDF Clear Set to 1b to clear VICH1SR.VBUFUDF. Writing 0b has no effect.
21	-	0h	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
20	TIMERR	0h	R0W	The read value is always 0b. VICH1SR.TIMERR Clear Set to 1b to clear VICH1SR.TIMERR. Writing 0b has no effect.
19 to 15	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14	-	0h	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
13 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	VIRDY	0h	R0W	The read value is always 0b. VICH1SR.VIRDY Clear Set to 1b to clear VICH1SR.VIRDY. Writing 0b has no effect.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	STOP	0h	R0W	The read value is always 0b. VICH1SR.STOP Clear Set to 1b to clear VICH1SR.STOP. Writing 0b has no effect.
0	START	0h	R0W	The read value is always 0b. VICH1SR.START Clear Set to 1b to clear VICH1SR.START. Writing 0b has no effect.

9.5.2.2.49 Video-Input Channel 1 Interrupt Enable Register (DSI_LINK_VICH1IER)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0418h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	VBUFOVF	VBUFUDF	-	TIMERR	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	VIRDY	-	STOP	START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	R	R	R	R	R	R	R	R	R	R	RW	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 26	-	All 0	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
25, 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23	VBUFOVF	0h	RW	Interrupt enable for the Video-Input Buffer Overflow Error 0b: Disables interrupt dsi_int_vin1 when VICH1SR.VBUFOVF = 1. 1b: Enables interrupt dsi_int_vin1 when VICH1SR.VBUFOVF = 1.
22	VBUFUDF	0h	RW	Interrupt enable for the Video-Input Buffer Underflow Error 0b: Disables interrupt dsi_int_vin1 when VICH1SR.VBUFUDF = 1. 1b: Enables interrupt dsi_int_vin1 when VICH1SR.VBUFUDF = 1.
21	-	0h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
20	TIMERR	0h	RW	Interrupt enable for the Timing Error 0b: Disables interrupt dsi_int_vin1 when VICH1SR.TIMERR = 1. 1b: Enables interrupt dsi_int_vin1 when VICH1SR.TIMERR = 1.
19 to 15	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
14	-	0h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
13 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3	VIRDY	0h	RW	Interrupt enable for the Video Input Accept Ready 0b: Disables interrupt dsi_int_vin1 when VICH1SR.VIRDY = 1. 1b: Enables interrupt dsi_int_vin1 when VICH1SR.VIRDY = 1.
2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	STOP	0h	RW	Interrupt enable for the Video Input Operation channel 1 Transfer Stop 0b: Disables interrupt dsi_int_vin1 when VICH1SR.STOP = 1. 1b: Enables interrupt dsi_int_vin1 when VICH1SR.STOP = 1.
0	START	0h	RW	Interrupt enable for the Video Input Operation channel 1 Transfer Start 0b: Disables interrupt dsi_int_vin1 when VICH1SR.START = 1. 1b: Enables interrupt dsi_int_vin1 when VICH1SR.START = 1.

9.5.2.2.50 Video-Input Channel 1 Pixel Packet Set Register (DSI_LINK_VICH1PPSETR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0420h

Initial Value : 000E_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	VC[1:0]		DT[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXESYNC	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
23, 22	VC[1:0]	0h	RW	Video-Input Operation Channel 1 Pixel Stream Packet Header Virtual Channel Identifier
21 to 16	DT[5:0]	Eh	RW	Video-Input Operation Channel 1 Pixel Stream Packet Header Data Type The following values can be set. 0Eh: Packed Pixel Stream, 16-bit RGB 1Eh: Packed Pixel Stream, 18-bit RGB 2Eh: Loosely Packed Pixel Stream, 18-bit RGB 3Eh: Packed Pixel Stream, 24-bit RGB Setting any other values is prohibited.
15	TXESYNC	0h	RW	Transmit End of Sync Pulse 0b: HSE and VSE are NOT transmitted 1b: HSE and VSE are transmitted The setting 0b is used for “Burst Mode” sequence or “Non-Burst Mode with Sync Events” sequence. The setting 1b is used for “Non-Burst Mode with Sync Pulse” sequence.
14 to 4	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
3 to 0	-	All 0	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

9.5.2.2.51 Video-Input Channel 1 Vertical Size Set Register (DSI_LINK_VICH1VSSETR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0428h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	VACTIVE[14:0]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSPOL	-	-	-	VSA[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 16	VACTIVE[14:0]	0h	RW	Sets the Line count of VACTIVE (Vertical Active lines)
15	VSPOL	0h	RW	Polarity of VSYNC Sets the polarity (active state) of the VSYNC signal. 0b: Active high 1b: Active low
14 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	VSA[11:0]	0h	RW	VSA Count Sets the Line count of VSA (Vertical Sync Active)

9.5.2.2.52 Video-Input Channel 1 Vertical Porch Set Register (DSI_LINK_VICH1VPSETR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 042Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	VFP[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VBP[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	VFP[12:0]	0h	RW	VFP Count Sets the Line count of VFP (Vertical Front Porch)
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	VBP[12:0]	0h	RW	VBP Count Sets the Line count of VBP (Vertical Back Porch)

9.5.2.2.53 Video-Input Channel 1 Horizontal Size Set Register (DSI_LINK_VICH1HSSETR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0430h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	HACTIVE[14:0]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HSPOL	-	-	-	HSA[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

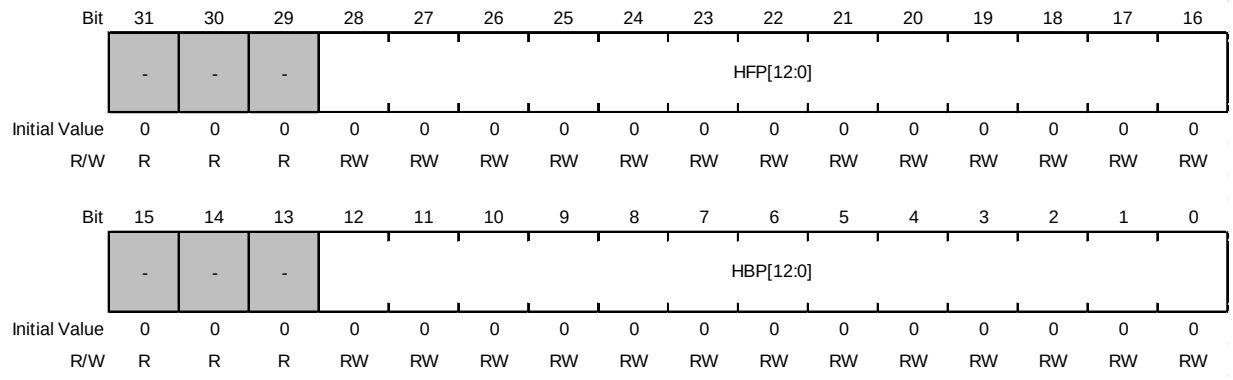
Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30 to 16	HACTIVE [14:0]	0h	RW	HACTIVE Count Sets the pixel count of HACTIVE (Horizontal Active pixels)
15	HSPOL	0h	RW	Polarity of HSYNC Sets the polarity (active state) of the HSYNC signal. 0b: Active high 1b: Active low
14 to 12	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
11 to 0	HSA[11:0]	0h	RW	HSA Count Sets the pixel count of HSA (Horizontal Sync Active)

9.5.2.2.54 Video-Input Channel 1 Horizontal Porch Set Register (DSI_LINK_VICH1HPSETR)

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 0434h

Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
28 to 16	HFP[12:0]	0h	RW	HFP Count Sets the pixel count of HFP (Horizontal Front Porch)
15 to 13	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
12 to 0	HBP[12:0]	0h	RW	HBP Count Sets the pixel count of HBP (Horizontal Back Porch)

9.5.2.2.55 Sequence Channel x Set 0 Register (DSI_LINK_SQCHxSET0R)

"x" indicates the channel number.

Note: "x" is 0 or 1.

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 05C0h (x = 0)
<DSI_LINK_base> + 0600h (x = 1)

Initial Value : 0080_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	R0W	RW	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	START
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	R	R	R0W	R0W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
23	-	1h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
22 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8 to 4	-	All 0	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
3, 2	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
1	-	0h	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
0	START	0h	R0W	The read value is always 0b. Sequence Operation Start Write 1b to start running from Descriptor #0. Setting it to 1b while RUNNING = 1 is prohibited. Writing 0b has no effect.

9.5.2.2.56 Sequence Channel x Set 1 Register (DSI_LINK_SQCHxSET1R)

"x" indicates the channel number.

Note: "x" is 0 or 1.

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 05C4h (x = 0)
<DSI_LINK_base> + 0604h (x = 1)

Initial Value : 0800_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MAXDESNUM[7:0]								CHBUFSZ[3:0]				-	-	-	-
Initial Value	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	MAXDESNUM [7:0]	8h	RW	Max Descriptor Number This field indicates the maximum number of descriptors, with the descriptor numbers starting from zero (e.g. the setting 8 means #0 to #7 can be used as descriptors). Writing a value other than the initial value is prohibited.
23 to 20	CHBUFSZ [3:0]	0h	R	Sequence Channel x Buffer Size 0h: 128 B 1h: 256 B 2h: 512 B 3h: 1 kB 4h: 2 kB 5h: 4 kB 6h: 8 kB 7h: 16 kB 8h: 32 kB 9h: 64 kB Buffer size = 128 x 2 CHBUFSZ Setting a transfer packet size larger than this field is prohibited.
19 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.5.2.2.57 Sequence Channel x Status Register (DSI_LINK_SQCHxSR)

"x" indicates the channel number.

Note: "x" is 0 or 1.

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 05D0h (x = 0)
<DSI_LINK_base> + 0610h (x = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	RXAKE	RXCORERR	RXPKTDFAIL	RXFAIL	RXFATALERR	-	TXIBERR	-	-	-	-	PKTBIGERR	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	ADESFIN	-	-	-	AACTFIN	-	RUNNING	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30	RXAKE	0h	R	Receive Acknowledge and Error Report Packet RXSR.RXAKE is also reported. Write 1b to the SQCHxSCR.RXAKE register to clear this field.
29	RXCORERR	0h	R	Receive Correctable Error The received packet has a correctable error. RXSR.ECCERR1B is also reported. Write 1b to the SQCHxSCR.RXCORERR register to clear this field.
28	RXPKTDFAIL	0h	R	Receive Packet Data Fail The packet Header was saved correctly but data were not. One or more of the following bits are also reported. RXSR.CRCERR, RXSR.WCERR, RXSR.UEXPKTERR, RXSR.MAXRPSZERR, RXSR.RXOVFERR and RXSR.IBERR. Write 1b to the SQCHxSCR.RXPKTDFAIL register to clear this field.
27	RXFAIL	0h	R	Receive Fail Expected reception did not proceed. One or more of the following bits are also reported. RXSR.PRESPTOERR, RXSR.ECCERR, RXSR.MLFERR and RXSR.NORETERR. Write 1b to the SQCHxSCR.RXFAIL register to clear this field.
26	RXFATALERR	0h	R	Receive Fatal Error A fatal timeout occurred during BTA. One or more of the following bits are also reported. FERRSR.TATO and FERRSR.LRXHTO. Write 1b to the SQCHxSCR.RXFATALERR register to clear this field.
25	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	TXIBERR	0h	R	Tx Internal Bus Error When this value is 1b, the Internal Bus Read had failed to respond. Write 1b to the SQCHxSCR.TXIBERR register to clear this field.
23 to 20	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	PKTBIGERR	0h	R	Packet is too big Error Sequence Operation channel 0: Hard wired to 0b. Sequence Operation channel 1: This bit is set when the length of a sequence packet is bigger than Video-Input channel's BLLP. Write 1b to the SQCHxSCR.PKTBIGERR register to clear this field.
18 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	ADESFIN	0h	R	All Descriptor Finished by Setting The descriptor is finished by setting SQCHxDSCyAR.NXACT = 0h for the descriptor. Write 1b to the SQCHxSCR.ADESFIN register to clear this field.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	AACTFIN	0h	R	Descriptor's All Action Finished This bit is set if this descriptor's all actions are finished with SQCHxDSCyCR.FINACT[0] =1b. Write 1b to the SQCHxSCR.AACTFIN register to clear this field.
3	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2	RUNNING	0h	R	Sequence Operation Running The value will be set to "1" during Sequence Operation. Do not change descriptor values while this bit is 1b. Do not change the values in SQCHxSET0R and SQCHxSET1R.
1, 0	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.5.2.2.58 Sequence Channel x Status Clear Register (DSI_LINK_SQCHxSCR)

"x" indicates the channel number.

Note: "x" is 0 or 1.

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 05D4h (x = 0)
<DSI_LINK_base> + 0614h (x = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	RXAKE	RXCORERR	RXPKTDFAIL	RXFAIL	RXFATALERR	-	TXIBERR	-	-	-	-	PKTBIGERR	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R0W	R0W	R0W	R0W	R0W	R	R0W	R	R0W	R	R	R0W	R	R	R0W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	ADESFIN	-	-	-	AACTFIN	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R0W	R	R	R	R	R	R	R0W	R	R	R	R0W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30	RXAKE	0h	R0W	The read value is always 0b. SQCHxSR.RXAKE Clear Set to 1b to clear SQCHxSR.RXAKE. Writing 0b has no effect.
29	RXCORERR	0h	R0W	The read value is always 0b. SQCHxSR.RXCORERR Clear Set to 1b to clear SQCHxSR.RXCORERR. Writing 0b has no effect.
28	RXPKTDFAIL	0h	R0W	The read value is always 0b. SQCHxSR.RXPKTDFAIL Clear Set to 1b to clear SQCHxSR.RXPKTDFAIL. Writing 0b has no effect.
27	RXFAIL	0h	R0W	The read value is always 0b. SQCHxSR.RXFAIL Clear Set to 1b to clear SQCHxSR.RXFAIL. Writing 0b has no effect.
26	RXFATALERR	0h	R0W	The read value is always 0b. SQCHxSR.RXFATALERR Clear Set to 1b to clear SQCHxSR.RXFATALERR. Writing 0b has no effect.
25	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	TXIBERR	0h	R0W	The read value is always 0b. SQCHxSR.TXIBERR Clear Set to 1b to clear SQCHxSR.TXIBERR. Writing 0b has no effect.
23	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	-	0h	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
21, 20		0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	PKTBIGERR	0h	R0W	The read value is always 0b. SQCHxSR.PKTBIGERR Clear Set to 1b to clear SQCHxSR.PKTBIGERR. Writing 0b has no effect.

Bit	Bit Name	Initial Value	R/W	Description
18, 17	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16, 15	-	0h	R0W	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
14 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	ADESFIN	0h	R0W	The read value is always 0b. SQCHxSR.ADESFIN Clear Set to 1b to clear SQCHxSR.ADESFIN. Writing 0b has no effect.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	AACTFIN	0h	R0W	The read value is always 0b. SQCHxSR.AACTFIN Clear Set to 1b to clear SQCHxSR.AACTFIN. Writing 0b has no effect.
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.5.2.2.59 Sequence Channel x Interrupt Enable Register (DSI_LINK_SQCHxIER)

"x" indicates the channel number.

Note: "x" is 0 or 1.

Access Size : 8, 16, 32 bits

Offset Address : <DSI_LINK_base> + 05D8h (x = 0)
<DSI_LINK_base> + 0618h (x = 1)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	RXAKE	RXCORERR	RXPKTDFAIL	RXFAIL	RXFATALERR	-	TXIBERR	-	-	-	-	PKTBIGERR	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	R	RW	R	RW	R	R	RW	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	ADESFIN	-	-	-	AACTFIN	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	R	R	R	R	RW	R	R	R	RW	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
30	RXAKE	0h	RW	Interrupt enable for the Receive Acknowledge and Error Report Packet 0b: Disables interrupt dsi_int_sqx when SQCHxSR.RXAKE = 1. 1b: Enables interrupt dsi_int_sqx when SQCHxSR.RXAKE = 1.
29	RXCORERR	0h	RW	Interrupt enable for the Receive Correctable Error 0b: Disables interrupt dsi_int_sqx when SQCHxSR.RXCORERR = 1. 1b: Enables interrupt dsi_int_sqx when SQCHxSR.RXCORERR = 1.
28	RXPKTDFAIL	0h	RW	Interrupt enable for the Receive Packet Data Fail 0b: Disables interrupt dsi_int_sqx when SQCHxSR.RXPKTDFAIL = 1. 1b: Enables interrupt dsi_int_sqx when SQCHxSR.RXPKTDFAIL = 1.
27	RXFAIL	0h	RW	Interrupt enable for the Receive Fail 0b: Disables interrupt dsi_int_sqx when SQCHxSR.RXFAIL = 1. 1b: Enables interrupt dsi_int_sqx when SQCHxSR.RXFAIL = 1.
26	RXFATALERR	0h	RW	Interrupt enable for the Receive Fatal Error 0b: Disables interrupt dsi_int_sqx when SQCHxSR.RXFATALERR = 1. 1b: Enables interrupt dsi_int_sqx when SQCHxSR.RXFATALERR = 1.
25	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
24	TXIBERR	0h	RW	Interrupt enable for the Tx Internal Bus Error 0b: Disables interrupt dsi_int_sqx when SQCHxSR.TXIBERR = 1. 1b: Enables interrupt dsi_int_sqx when SQCHxSR.TXIBERR = 1.
23	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22	-	0h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
21, 20		0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
19	PKTBIGERR	0h	RW	Interrupt enable for the Packet is too big Error Sequence Operation channel 0: This field has no meaning. Sequence Operation channel 1: 0b: Disables interrupt dsi_int_sqx when SQCHxSR.PKTBIGERR = 1. 1b: Enables interrupt dsi_int_sqx when SQCHxSR.PKTBIGERR = 1.
18, 17	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16, 15	-	0h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.

Bit	Bit Name	Initial Value	R/W	Description
14 to 9	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
8	ADESFIN	0h	RW	Interrupt enable for the Descriptor Finished by Setting 0b: Disables interrupt dsi_int_sqx when SQCHxSR.ADESFIN = 1. 1b: Enables interrupt dsi_int_sqx when SQCHxSR.ADESFIN = 1.
7 to 5	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
4	AACTFIN	0h	RW	Interrupt enable for the Descriptor's All Action Finished 0b: Disables interrupt dsi_int_sqx when SQCHxSR.AACTFIN = 1. 1b: Enables interrupt dsi_int_sqx when SQCHxSR.AACTFIN = 1.
3 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.5.2.2.60 Sequence Channel x Descriptor y-A Register (DSI_LINK_SQCHxDSCyAR)

"x" indicates the channel number.

Note: "x" is 0 or 1.

"y" indicates the descriptor number.

Note: "y" is 00 to 07.

Access Size : 32 bits

Offset Address : <DSI_LINK_base> + 0780h + 0010h × (y) (x = 0)
<DSI_LINK_base> + 0800h + 0010h × (y) (x = 1)

Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	NXACT[1:0]		BTA[1:0]		SPD	FMT	VC[1:0]		DT[5:0]					
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA1[7:0]								DATA0[7:0]							
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	xh*2	RW	Reserved When read, the last written value is read. When writing, be sure to write 0b. Operation is not guaranteed if a value other than 0b is written.
29, 28	NXACT[1:0]	xh*2	RW	Next Action 0h: Terminates operation after this descriptor is finished. 1h: Operates the next descriptor after this descriptor is finished. 2h: Reserved 3h: Reserved
27, 26	BTA[1:0]	xh*2	RW	Bus Turn Around Sets Tx Request without either Bus Turn-Around (BTA) or no-operation. Or else, sets Tx Request with BTA. 0h: Tx Request without BTA or no-operation. 1h: Tx non-Read Request with BTA. 2h: Tx Read Request with BTA. 3h: BTA only
25	SPD	xh*2	RW	Speed Sets the speed type in Tx Request. 0b: High Speed 1b: Low Power Setting this bit to "1" during Video-Input Operation is prohibited.
24	FMT	xh*2	RW	Format Packet format of Tx Packet Header. This field should be set by Data Type. 0b: Short Packet 1b: Long Packet
23, 22	VC[1:0]	xh*2	RW	Virtual Channel Identifier of Virtual Channel of Tx Packet Header.
21 to 16	DT[5:0]	xh*2	RW	Data Type Data Type of Tx Packet Header.
15 to 8	DATA1[7:0]	xh*2	RW	Data 1 Data 1 of Tx Packet Header*1. When Tx Packet format is Long Packet, this value is the upper 8 bits of the word count. The maximum Long Packet size in Command Transfer mode which uses registers is 16 bytes. (SQCHxDSCyBR.DTSEL = 0h and SQCHxDSCyAR.FMT = 1) The maximum Long Packet size in Command Transfer mode which uses the internal bus memory is the number of bytes indicated by SQCHxSET1R.CHBUFSZ. (SQCHxDSCyBR.DTSEL = 1h and SQCHxDSCyAR.FMT = 1)

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	DATA0[7:0]	xh*2	RW	Data 0 Data 0 of Tx Packet Header*1. When Tx Packet format is Long Packet, this value is the lower 8 bits of the word count. The maximum Long Packet size in Command Transfer mode which uses registers is 16 bytes. (SQCHxDSCyBR.DTSEL = 0h and SQCHxDSCyAR.FMT = 1) The maximum Long Packet size in Command Transfer mode which uses the internal bus memory is the number of bytes indicated by SQCHxSET1R.CHBUFSZ. (SQCHxDSCyBR.DTSEL = 1h and SQCHxDSCyAR.FMT = 1)

x = Undefined

Note 1. The maximum size for Command Transfer mode Long Packet by using the register is 16 bytes. (SQCHxDSCyBR.DTSEL = 0h and SQCHxDSCyAR.FMT = 1h)
The maximum size for Command Transfer mode Long Packet by using the internal bus memory is SQCHxSET1R.CHBUFSZ bytes. (SQCHxDSCyBR.DTSEL = 1h and SQCHxDSCyAR.FMT = 1h)

Note 2. The initial values are determined by the initial value of Descriptor RAM.

9.5.2.2.61 Sequence Channel x Descriptor y-B Register (DSI_LINK_SQCHxDSCyBR)

"x" indicates the channel number.

Note: "x" is 0 or 1.

"y" indicates the descriptor number.

Note: "y" is 00 to 07.

Access Size : 32 bits

Offset Address : <DSI_LINK_base> + 0784h + 0010h × (y) (x = 0)
<DSI_LINK_base> + 0804h + 0010h × (y) (x = 1)

Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	DTSEL[1:0]		-	-	-	-	-	-	-	-
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	xh*[1]	RW	Reserved When read, the last written value is read. When writing, be sure to write 0b. Operation is not guaranteed if a value other than 0b is written.
25, 24	DTSEL[1:0]	xh*[1]	RW	Data Select 0h: Packet Payload Data register 1h: Long Packet data use memory space. In the case of writing, internal read causes reading from the SQCHxDSCyDR address. In the case of reading, internal write causes writing to the SQCHxDSCyDR address. Note: Read Header is always saved to the RXRSSxR register. 2h, 3h: Reserved If a write request with Short packet, then set 0h. When DTSEL = 0h, Long packet write request data size shall be equal to or smaller than 16 bytes. If read request data to save exceeds 16 bytes, this is reported as an error.
23 to 0	-	xh*[1]	RW	Reserved When read, the last written value is read. When writing, be sure to write 0b. Operation is not guaranteed if a value other than 0b is written.

x = Undefined

Note 1. The initial values are determined by the initial value of Descriptor RAM.

9.5.2.2.62 Sequence Channel x Descriptor y-C Register (DSI_LINK_SQCHxDSCyCR)

"x" indicates the channel number.

Note: "x" is 0 or 1.

"y" indicates the descriptor number.

Note: "y" is 00 to 07.

Access Size : 32 bits

Offset Address : <DSI_LINK_base> + 0788h + 0010h × (y) (x = 0)
<DSI_LINK_base> + 0808h + 0010h × (y) (x = 1)

Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ACTCODE[7:0]								-	AUXOP	-	-	-	-	-	-
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	FINACT[2:0]		
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ACTCODE[7:0]	xh*1	RW	Action Code Indicates the rx result same slot number for BTA action. Setting a value greater than or equal to 4 is prohibited. Setting the same rx result save slot number to a different descriptor is prohibited. When AUXOP is 1. 00h: Sends Reset-Trigger. 04h: Sends initial skew calibration Prohibited to use in channel 0 (LP only). 05h: Sends periodic skew calibration Prohibited to use in channel 0 (LP only). 08h: No-operation Other: Reserved Set 00h in other cases.
23	-	xh*1	RW	Reserved When read, the last written value is read. When writing, be sure to write 0b. Operation is not guaranteed if a value other than 0b is written.
22	AUXOP	xh*1	RW	Auxiliary Operation Used for auxiliary operation execution. When this bit is set to 1b, SQCHxDSCyBR.BTA should be set to 0h. Set this bit to 1b with BTA = 0h. Setting this bit to 1b during Video-Input Operation is prohibited.
21 to 3	-	xh*1	RW	Reserved When read, the last written value is read. When writing, be sure to write 0b. Operation is not guaranteed if a value other than 0b is written.
2 to 0	FINACT[2:0]	xh*1	RW	Finish Action Bit 2: Set to 0b. Bit 1: Set to 0b. Bit 0: Sets SQCHxSR.AACTFIN if this descriptor's all actions have finished. (0b: Disabled, 1b: Enabled). Since each descriptor's operation is not the same, descriptors may finish out of order. For example, Descriptor#1 may finish before Descriptor#0.

x = Undefined

Note 1. The initial values are determined by the initial value of Descriptor RAM.

9.5.2.2.63 Sequence Channel x Descriptor y-D Register (DSI_LINK_SQCHxDSCyDR)

"x" indicates the channel number.

Note: "x" is 0 or 1.

"y" indicates the descriptor number.

Note: "y" is 00 to 07.

Access Size : 32 bits

Offset Address : <DSI_LINK_base> + 078Ch + 0010h × (y) (x = 0)
<DSI_LINK_base> + 080Ch + 0010h × (y) (x = 1)

Initial Value : xxxx_xxxxh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LADDR[31:16]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LADDR[15:0]															
Initial Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LADDR[31:0]	xh*[1]	RW	Lower Address Sets the lower 32-bit address for Long Packet Payload Data in Sequence Operation channel x Descriptor y. When Sequence Operation Data Select is "use memory space" (SQCHxDSCyBR.DTSEL =1h) and Sequence Operation Format is Long packet (SQCHxDSCyAR.FMT=1b), the DSI-Tx module accesses the address by internal bus Master Read in Write action and internal bus Write in Read action. Set the lower three bits to "000b".

x = Undefined

Note 1. The initial values are determined by the initial value of Descriptor RAM.

9.5.2.3 Registers in D-PHY

9.5.2.3.1 DSI PLL Enable Register (DSI_DPHY_PLLENR)

Access Size : 8, 16, 32 bits

Address : <DSI_DPHY_base> + 0000h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PLLEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	PLLEN	0h	RW	Initialize PLL during LOW state and enable when it is HIGH state.

9.5.2.3.2 DSI PHY Reset Register (DSI_DPHY_PHYRSTR)

Access Size : 8, 16, 32 bits

Address : <DSI_DPHY_base> + 0004h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PHYM RSTN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
0	PHYMRSTN	0h	RW	PHY Master Clock and Data Lane Reset

9.5.2.3.3 DSI PLL Clock Setting 0 Register (DSI_DPHY_PLLCLKSET0R)

Access Size : 8, 16, 32 bits
Address : <DSI_DPHY_base> + 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	PLL_M[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PLL_P[5:0]						-	-	-	-	-	PLL_S[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
25 to 16	PLL_M[9:0]	0h	RW	Value of the main-divider in PLL Can be set in the range of 64 to 1023. Values outside this range are prohibited.
15, 14	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
13 to 8	PLL_P[5:0]	0h	RW	Value of the pre-divider in PLL Can be set in the range of 1 to 4. Values outside this range are prohibited.
7 to 3	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
2 to 0	PLL_S[2:0]	0h	RW	Value of Scaler in PLL Can be set in the range of 0 to 5. Values outside this range are prohibited.

The PLL clock frequency (F_{FOUT}) is expressed by the following equation.

$$F_{FREF} = 24MHz/PLL_P, 6MHz \leq F_{FREF} \leq 24MHz$$

$$F_{FVCO} = (PLL_M + (PLL_K/65536)) \times F_{FREF}, 1050MHz \leq F_{FVCO} \leq 2100MHz$$

$$F_{FOUT} = F_{FVCO}/2^{PLL_S}, 80MHz \leq F_{FOUT} \leq 1500MHz$$

$$1 \leq PLL_P \leq 4, 64 \leq PLL_M \leq 1023, 0 \leq PLL_S \leq 5, -32768 \leq PLL_K \leq 32767$$

9.5.2.3.4 DSI PLL Clock Setting 1 Register (DSI_DPHY_PLLCLKSET1R)

Access Size : 8, 16, 32 bits

Address : <DSI_DPHY_base> + 0014h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PLL_K[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

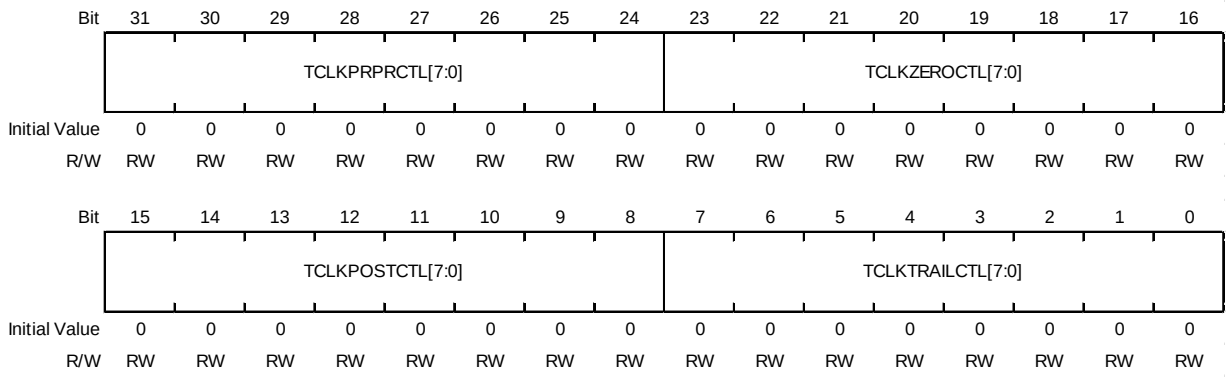
Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
15 to 0	PLL_K[15:0]	0h	RW	Value of DSM in PLL PLL_K is a two's complement, and the range is -32768 to 32767.

9.5.2.3.5 DSI PHY TCLK Setting Register (DSI_DPHY_PHYTCLKSETR)

Access Size : 8, 16, 32 bits

Address : <DSI_DPHY_base> + 0020h

Initial Value : 0000_0000h



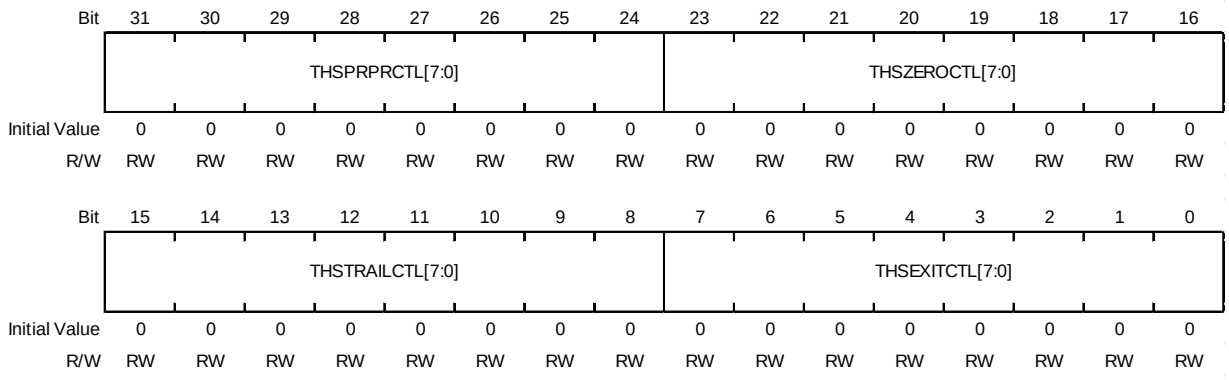
Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TCLKPRPRCTL [7:0]	0h	RW	Control TCLK-PREPARE of the Master Clock Lane.
23 to 16	TCLKZEROCTL [7:0]	0h	RW	Control TCLK-ZERO of the Master Clock Lane.
15 to 8	TCLKPOSTCTL [7:0]	0h	RW	Control TCLK-POST of the Master Clock Lane.
7 to 0	TCLKTRAILCTL [7:0]	0h	RW	Control TCLK-TRAIL of the Master Clock Lane.

9.5.2.3.6 DSI PHY THS Setting Register (DSI_DPHY_PHYTHSSETR)

Access Size : 8, 16, 32 bits

Address : <DSI_DPHY_base> + 0024h

Initial Value : 0000_0000h



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	THSPRPRCTL [7:0]	0h	RW	Control THS-PREPARE of the Master Data Lanes.
23 to 16	THSZEROCTL [7:0]	0h	RW	Control THS-ZERO of the Master Data Lanes.
15 to 8	THSTRAILCTL [7:0]	0h	RW	Control THS-TRAIL of the Master Data Lanes.
7 to 0	THSEXITCTL [7:0]	0h	RW	Control THS-EXIT of the Master Clock and Data Lanes.

9.5.2.3.7 DSI PHY TLPX Setting Register (DSI_DPHY_PHYTLPXSETR)

Access Size : 8, 16, 32 bits

Address : <DSI_DPHY_base> + 0028h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TLPXCTL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
7 to 0	TLPXCTL[7:0]	0h	RW	Control TLPX of the Master Clock and Data Lanes.

9.5.2.3.8 DSI PHY Control Register (DSI_DPHY_PHYCR)

Access Size : 8, 16, 32 bits

Address : <DSI_DPHY_base> + 0030h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	ULPSEXIT[9:0]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
9 to 0	ULPSEXIT[9:0]	0h	RW	ULPS EXIT Counter Value Control Should be set during the initialization or power-up sequence. Set 100 [µsec] x lpclk / 4.

9.5.2.3.9 DSI PHY Control 1 Register (DSI_DPHY_PHYC1R)

Access Size : 8, 16, 32 bits

Address : <DSI_DPHY_base> + 0034h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	BIASREFVCNT [1:0]		-	-	-	-	-	-	-	-	-	-	-	CLLTXSLWDWN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLLTXSLWDWN	CLLTXSLWUP	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29, 28	BIASREFVCNT [1:0]	0h	RW	Bias Reference Voltage 710 m Control 00b: 715 mV (default) 01b: 724 mV 10b: 733 mV 11b: 706 mV
27 to 17	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
16, 15	CLLTXSLWDWN[1:0]	0h	RW	Master Clock Lane's LP-TX Driver Slew Rate Down Control 00b: No change (default) 01b: Decrease the slew rate by about 15% 10b: Decrease the slew rate by about 15% 11b: Decrease the slew rate by about 30%
14	CLLTXSLWUP	0h	RW	Master Clock Lane's LP-TX Driver Slew Rate Up Control 0b: No change (default) 1b: Slew rate up
13 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.5.2.3.10 DSI PHY Control 2 Register (DSI_DPHY_PHYC2R)

Access Size : 8, 16, 32 bits

Address : <DSI_DPHY_base> + 0038h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DL3LTXSLEWDWN[1:0]		DL3LTXSLEWUP	DL2LTXSLEWDWN[1:0]		DL2LTXSLEWUP	DL1LTXSLEWDWN[1:0]		DL1LTXSLEWUP	DL0LTXSLEWDWN[1:0]		DL0LTXSLEWUP	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	DL3LTXSLEWDWN[1:0]	0h	RW	Master Data 3 Lane's LP-TX Driver Slew Rate Down Control 00b: No change (default) 01b: Decrease the slew rate by about 15% 10b: Decrease the slew rate by about 15% 11b: Decrease the slew rate by about 30%
29	DL3LTXSLEWUP	0h	RW	Master Data 3 Lane's LP-TX Driver Slew Rate Up Control 0b: No change (default) 1b: Slew rate up
28, 27	DL2LTXSLEWDWN[1:0]	0h	RW	Master Data 2 Lane's LP-TX Driver Slew Rate Down Control 00b: No change (default) 01b: Decrease the slew rate by about 15% 10b: Decrease the slew rate by about 15% 11b: Decrease the slew rate by about 30%
26	DL2LTXSLEWUP	0h	RW	Master Data 2 Lane's LP-TX Driver Slew Rate Up Control 0b: No change (default) 1b: Slew rate up
25, 24	DL1LTXSLEWDWN[1:0]	0h	RW	Master Data 1 Lane's LP-TX Driver Slew Rate Down Control 00b: No change (default) 01b: Decrease the slew rate by about 15% 10b: Decrease the slew rate by about 15% 11b: Decrease the slew rate by about 30%
23	DL1LTXSLEWUP	0h	RW	Master Data 1 Lane's LP-TX Driver Slew Rate Up Control 0b: No change (default) 1b: Slew rate up
22, 21	DL0LTXSLEWDWN[1:0]	0h	RW	Master Data 0 Lane's LP-TX Driver Slew Rate Down Control 00b: No change (default) 01b: Decrease the slew rate by about 15% 10b: Decrease the slew rate by about 15% 11b: Decrease the slew rate by about 30%
20	DL0LTXSLEWUP	0h	RW	Master Data 0 Lane's LP-TX Driver Slew Rate Up Control 0b: No change (default) 1b: Slew rate up
19 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.5.2.3.11 DSI PHY Control 3 Register (DSI_DPHY_PHYC3R)

Access Size : 8, 16, 32 bits

Address : <DSI_DPHY_base> + 003Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	HSTXRFTIM[2:0]			-	-	-	-	HSTXTMIMPD[2:0]			HSTXTMIMPU[2:0]			-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	R	R	R	R	RW	RW	RW	RW	RW	RW	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

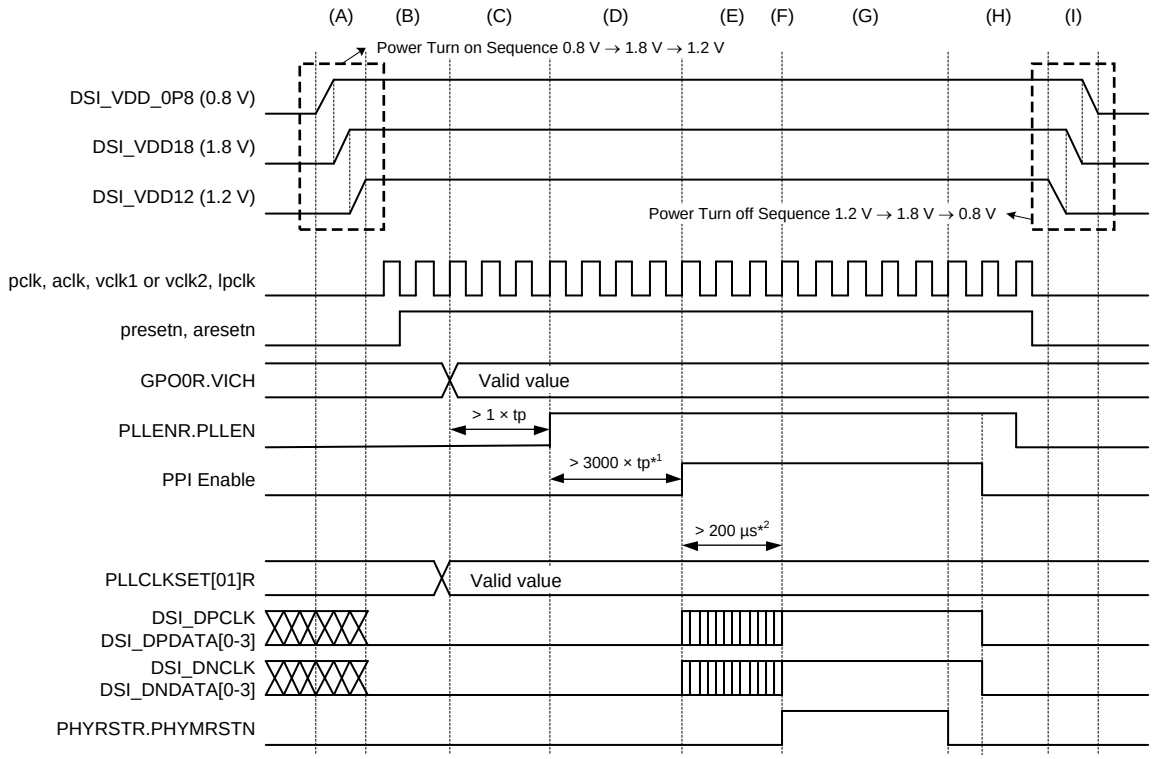
Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	0h	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
29 to 27	HSTXRFTIM [2:0]	0h	RW	HS-TX Rise & Fall Time Control 000b: 135 ps (default) 001b: 130 ps 010b: 125 ps 011b: 120 ps 100b: 230 ps 101b: 225 ps 110b: 220 ps 111b: 215 ps This is common to both Clock Lane and Data Lane.
26 to 23	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.
22 to 20	HSTXTMIMPD [2:0]	0h	RW	HS-TX Driver Termination Impedance Control (Down) 000b: 50 ohm (default) 001b: 52 ohm 010b: 54 ohm 011b: 56 ohm 100b: 44 ohm 101b: 46 ohm 110b: 47 ohm 111b: 48 ohm Note: Set the same value to HSTXMIMPU and HSTXMIMPD. Note: These bits are common to both Clock Lane and Data Lane.
19 to 17	HSTXTMIMPU [2:0]	0h	RW	HS-TX Driver Termination Impedance Control (Up) 000b: 50 ohm (default) 001b: 52 ohm 010b: 54 ohm 011b: 56 ohm 100b: 44 ohm 101b: 46 ohm 110b: 47 ohm 111b: 48 ohm Note: Set the same value to HSTXMIMPU and HSTXMIMPD. Note: These bits are common to both Clock Lane and Data Lane.
16 to 0	-	All 0	R	Reserved Whenever it is read, 0b is read. The written value will be ignored.

9.5.3 Operation

This section shows various operation procedures for using the DSI-Tx Module. As for the interrupt operation, Interrupt Enable bit of each interrupt source is enabled (set to 1b) before use, and after confirming the interrupt, the interrupt source is cleared. And more, previous action's interrupt source must be cleared before setting the interrupt enable. Flows in this section indicate only interrupt bit. The DSI-Tx Module does not care whether firmware use the bits by interrupt action or flag polling action. These descriptions are omitted in each flow in this section unless otherwise specified. Error checks are also omitted in normal flow.

9.5.3.1 Power on Reset and Initial Settings for All Operations

The power on/off sequence of the DSI-Tx Module is shown **Figure 9.5-4**.



Note: tp = 1 / (PLLREFCLK / PLLCLKSET0R.PLL_P)

Note 1. PLL Locking Time

Note 2. LP regulator initial time

Figure 9.5-4 Power On/Off Sequence

(A) Bring up DSI_VDD0P8, DSI_VDD18, and DSI_VDD12 in that order.

(B) De-assert presetn and aresetn after clocks are stable.

(C) Write appropriate values to D-PHY registers except PLLENR and PHYRSTR, then wait for more than 1 × tp.

(D) Write 1b to PLLENR.PLLEN, then wait for more than 3000 × tp.

(E) Write TXSETR, then wait for more than 200 µsec.

(F) Write 1b to PHYRSTR.PHYMRSTN and write LINK registers.

(G) The DSI-Tx Module is ready.

At power-on reset, all registers of the DSI-Tx Module are initialized.

Perform the initial settings of the register at Step (C) and (F) in the power-on sequence. It is not necessary to set the register that is used with the default value.

The value to be written to the D-PHY register in Step (C) is as follows.

- Set PLLCLKSET0R and PLLCLKSET1R so that the PLL clock frequency (F_{FOUT}) equals the data rate.
- Follow **Table 9.5-4** to set PHYTCLKSETR, PHYTHSSETR, and PHYTLPXSETR.
- Follow **Table 9.5-5** to set PHYCR.
- Follow register descriptions to set PHYC1R, PHYC2R, and PHYC3R.

Table 9.5-4 Recommend Setting for PHYTCLKSETR, PHYTHSSETR, and PHYTLPXSETR (1/4)

Data Rate [Mbps]		TCLKPRPRCTL[7:0]	TCLKZEROCTL[7:0]	TCLKPOSTCTL[7:0]	TCLKTRAILCTL[7:0]	THSPRPRCTL[7:0]	THSZEROCTL[7:0]	THSTRAILCTL[7:0]	TLPXCTL[7:0]	THSEXITCTL[7:0]
Lower (>)	Upper (<=)									
1490	1500	13	65	17	13	14	24	16	11	18
1480	1490	13	65	17	13	14	24	16	11	18
1470	1480	13	64	17	13	14	24	16	11	18
1460	1470	13	64	17	13	14	24	16	11	18
1450	1460	13	63	17	13	13	24	16	10	18
1440	1450	13	63	17	13	13	23	16	10	18
1430	1440	13	63	17	13	13	23	16	10	18
1420	1430	12	62	17	13	13	23	16	10	17
1410	1420	12	62	17	13	13	23	16	10	17
1400	1410	12	61	16	13	13	23	16	10	17
1390	1400	12	61	16	13	13	23	16	10	17
1380	1390	12	60	16	12	13	22	15	10	17
1370	1380	12	60	16	12	13	22	15	10	17
1360	1370	12	59	16	12	13	22	15	10	17
1350	1360	12	59	16	12	13	22	15	10	17
1340	1350	12	59	16	12	12	22	15	10	16
1330	1340	12	58	16	12	12	21	15	10	16
1320	1330	11	58	16	12	12	21	15	9	16
1310	1320	11	57	16	12	12	21	15	9	16
1300	1310	11	57	16	12	12	21	15	9	16
1290	1300	11	56	16	12	12	21	15	9	16
1280	1290	11	56	16	12	12	21	15	9	16
1270	1280	11	56	15	11	12	20	14	9	16
1260	1270	11	55	15	11	12	20	14	9	15

Table 9.5-4 Recommend Setting for PHYTCLKSETR, PHYTHSSETR, and PHYTLPXSETR (2/4)

Data Rate [Mbps]		TCLKPRPRCTL[7:0]	TCLKZEROCTL[7:0]	TCLKPOSTCTL[7:0]	TCLKTRAILCTL[7:0]	THSPRPRCTL[7:0]	THSZEROCTL[7:0]	THSTRAILCTL[7:0]	TLPXCTL[7:0]	THSEXITCTL[7:0]
Lower (>)	Upper (<=)									
1250	1260	11	55	15	11	12	20	14	9	15
1240	1250	11	54	15	11	11	20	14	9	15
1230	1240	11	54	15	11	11	20	14	9	15
1220	1230	11	53	15	11	11	19	14	9	15
1210	1220	10	53	15	11	11	19	14	9	15
1200	1210	10	52	15	11	11	19	14	9	15
1190	1200	10	52	15	11	11	19	14	9	15
1180	1190	10	52	15	11	11	19	14	8	14
1170	1180	10	51	15	11	11	19	13	8	14
1160	1170	10	51	15	10	11	18	13	8	14
1150	1160	10	50	15	10	11	18	13	8	14
1140	1150	10	50	15	10	11	18	13	8	14
1130	1140	10	49	14	10	10	18	13	8	14
1120	1130	10	49	14	10	10	18	13	8	14
1110	1120	10	49	14	10	10	17	13	8	14
1100	1110	9	48	14	10	10	17	13	8	13
1090	1100	9	48	14	10	10	17	13	8	13
1080	1090	9	47	14	10	10	17	13	8	13
1070	1080	9	47	14	10	10	17	13	8	13
1060	1070	9	46	14	10	10	17	12	8	13
1050	1060	9	46	14	10	10	16	12	7	13
1040	1050	9	45	14	9	10	16	12	7	13
1030	1040	9	45	14	9	10	16	12	7	13
1020	1030	9	45	14	9	9	16	12	7	12
1010	1020	9	44	14	9	9	16	12	7	12
1000	1010	8	44	13	9	9	15	12	7	12
990	1000	8	43	13	9	9	15	12	7	12
980	990	8	43	13	9	9	15	12	7	12
970	980	8	42	13	9	9	15	12	7	12
960	970	8	42	13	9	9	15	12	7	12
950	960	8	42	13	9	9	15	11	7	12
940	950	8	41	13	9	9	14	11	7	11
930	940	8	41	13	8	9	14	11	7	11
920	930	8	40	13	8	8	14	11	6	11
910	920	8	40	13	8	8	14	11	6	11
900	910	8	39	13	8	8	14	11	6	11
890	900	7	39	13	8	8	13	11	6	11
880	890	7	38	13	8	8	13	11	6	11
870	880	7	38	12	8	8	13	11	6	11
860	870	7	38	12	8	8	13	11	6	10
850	860	7	37	12	8	8	13	11	6	10

Table 9.5-4 Recommend Setting for PHYTCLKSETR, PHYTHSSETR, and PHYTLPXSETR (3/4)

Data Rate [Mbps]		TCLKPRPRCTL[7:0]	TCLKZEROCTL[7:0]	TCLKPOSTCTL[7:0]	TCLKTRAILCTL[7:0]	THSPRPRCTL[7:0]	THSZEROCTL[7:0]	THSTRAILCTL[7:0]	TLPXCTL[7:0]	THSEXITCTL[7:0]
Lower (>)	Upper (<=)									
840	850	7	37	12	8	8	13	10	6	10
830	840	7	36	12	8	8	12	10	6	10
820	830	7	36	12	8	8	12	10	6	10
810	820	7	35	12	7	7	12	10	6	10
800	810	7	35	12	7	7	12	10	6	10
790	800	7	35	12	7	7	12	10	6	10
780	790	6	34	12	7	7	11	10	5	9
770	780	6	34	12	7	7	11	10	5	9
760	770	6	33	12	7	7	11	10	5	9
750	760	6	33	12	7	7	11	10	5	9
740	750	6	32	12	7	7	11	9	5	9
730	740	6	32	11	7	7	11	9	5	9
720	730	6	31	11	7	7	10	9	5	9
710	720	6	31	11	7	6	10	9	5	9
700	710	6	31	11	6	6	10	9	5	8
690	700	6	30	11	6	6	10	9	5	8
680	690	5	30	11	6	6	10	9	5	8
670	680	5	29	11	6	6	9	9	5	8
660	670	5	29	11	6	6	9	9	5	8
650	660	5	28	11	6	6	9	9	4	8
640	650	5	28	11	6	6	9	9	4	8
630	640	5	28	11	6	6	9	8	4	8
620	630	5	27	11	6	6	9	8	4	7
610	620	5	27	11	6	6	8	8	4	7
600	610	5	26	10	6	5	8	8	4	7
590	600	5	26	10	6	5	8	8	4	7
580	590	5	25	10	5	5	8	8	4	7
570	580	4	25	10	5	5	8	8	4	7
560	570	4	24	10	5	5	7	8	4	7
550	560	4	24	10	5	5	7	8	4	7
540	550	4	24	10	5	5	7	8	4	6
530	540	4	23	10	5	5	7	8	4	6
520	530	4	23	10	5	5	7	7	3	6
510	520	4	22	10	5	5	7	7	3	6
500	510	4	22	10	5	5	6	7	3	6
490	500	4	21	10	5	4	6	7	3	6
480	490	4	21	10	5	4	6	7	3	6
470	480	4	21	9	4	4	6	7	3	6
460	470	3	20	9	4	4	6	7	3	5
450	460	3	20	9	4	4	5	7	3	5
440	450	3	19	9	4	4	5	7	3	5

Table 9.5-4 Recommend Setting for PHYTCLKSETR, PHYTHSSETR, and PHYTLPXSETR (4/4)

Data Rate [Mbps]		TCLKPRPRCTL[7:0]	TCLKZEROCTL[7:0]	TCLKPOSTCTL[7:0]	TCLKTRAILCTL[7:0]	THSPRPRCTL[7:0]	THSZEROCTL[7:0]	THSTRAILCTL[7:0]	TLPXCTL[7:0]	THSEXITCTL[7:0]
Lower (>)	Upper (<=)									
430	440	3	19	9	4	4	5	7	3	5
420	430	3	18	9	4	4	5	7	3	5
410	420	3	18	9	4	4	5	6	3	5
400	410	3	17	9	4	4	5	6	3	5
390	400	3	17	9	4	3	4	6	3	5
380	390	3	17	9	4	3	4	6	2	4
370	380	3	16	9	4	3	4	6	2	4
360	370	2	16	9	3	3	4	6	2	4
350	360	2	15	9	3	3	4	6	2	4
340	350	2	15	9	3	3	3	6	2	4
330	340	2	14	8	3	3	3	6	2	4
320	330	2	14	8	3	3	3	6	2	4
310	320	2	14	8	3	3	3	5	2	4
300	310	2	13	8	3	3	3	5	2	3
290	300	2	13	8	3	3	3	5	2	3
280	290	2	12	8	3	2	2	5	2	3
270	280	2	12	8	3	2	2	5	2	3
260	270	2	11	8	3	2	2	5	2	3
250	260	1	11	8	3	2	2	5	1	3
240	250	1	10	8	2	2	2	5	1	3
230	240	1	9	8	2	2	1	5	1	3
220	230	1	8	8	2	2	1	5	1	2
210	220	1	8	8	2	2	1	5	1	2
200	210	1	7	7	2	2	1	4	1	2
190	200	1	7	7	2	2	1	4	1	2
180	190	1	7	7	2	1	1	4	1	2
170	180	1	6	7	2	1	0	4	1	2
160	170	1	6	7	2	1	0	4	1	2
150	160	1	6	7	2	1	0	4	1	2
140	150	0	5	7	2	1	0	4	1	1
130	140	0	5	7	1	1	0	4	1	1
120	130	0	4	7	1	1	0	4	0	1
110	120	0	4	7	1	1	0	4	0	1
100	110	0	3	7	1	0	0	4	0	1
90	100	0	3	7	1	0	0	3	0	1
80	90	0	2	7	1	0	0	3	0	1
—	80	0	2	6	1	0	0	3	0	1

Table 9.5-5 Recommend Setting for PHYCR

lpclk [MHz]	ULPSEXIT[9:0]
15.625	391
7.8125	195
3.90625	98
1.953125	49

The LINK registers that need to be initialized in Step (F) are as follows.

- Common Settings
  - TXSETR
  - ULPSSETR
  - DSISETR
  - CLSTPTSETR
  - LPTRNSTSETR
- Timeout Settings
  - PRESPTOBTASETR
  - PRESPTOLPSETR
  - PRESPTOHSSETR
  - HSTXTOSETR
  - LRXHTOSETR
  - TATOSETR

These registers and D-PHY registers can only be changed during the initial setup after a power-on reset and during a software reset in **9.5.3.2.1 Software Reset**. It is prohibited to change at any other time.

9.5.3.2 Reset

9.5.3.2.1 Software Reset

Figure 9.5-5 shows the operation during software reset. Make sure that APB clock (pclk), Internal Bus clock (aclk), Video clock (vclk1), PLL Multiplied clock (PLLREFCLK), and Escape mode Transmit clock (lpclk) are running and PHYRSTR.PHYMRSTN is high before starting software reset.

Writing 1b to RSTCR.SWRST will issue a software reset. After issuing the software reset, check the start of the software reset process with RSTSR.SWRSTxx = 1b. Then write 1b to RSTCR.FCETXSTP to transition the valid data lane of the D-PHY to the Tx Stop State. By checking RSTSR.DL0DIR = 0b and RSTSR.DLSTPST [x] = 1b of the valid data lane, it is possible to confirm that the valid data lane of D-PHY has transitioned to Tx Stop State. After confirming that the valid data lane of the D-PHY has transitioned to the Tx Stop State, write 0b to RSTCR.FCETXSTP. Then write 0b to RSTCR.SWRST and check the completion of the software reset process with RSTSR.SWRSTx = 0b. Since all registers are synchronously reset except for some registers*[1], the software reset process is not completed even if a software reset is issued while the clock is stopped.

Note 1. Registers that require initial settings as described in **9.5.3.1 Power on Reset and Initial Settings for All Operations**, GPO0R register, GPO1R register, RSTCR register, RSTSR register, and SQCHxDSCyAR – SQCHxDSCyDR register are not initialized by software reset.

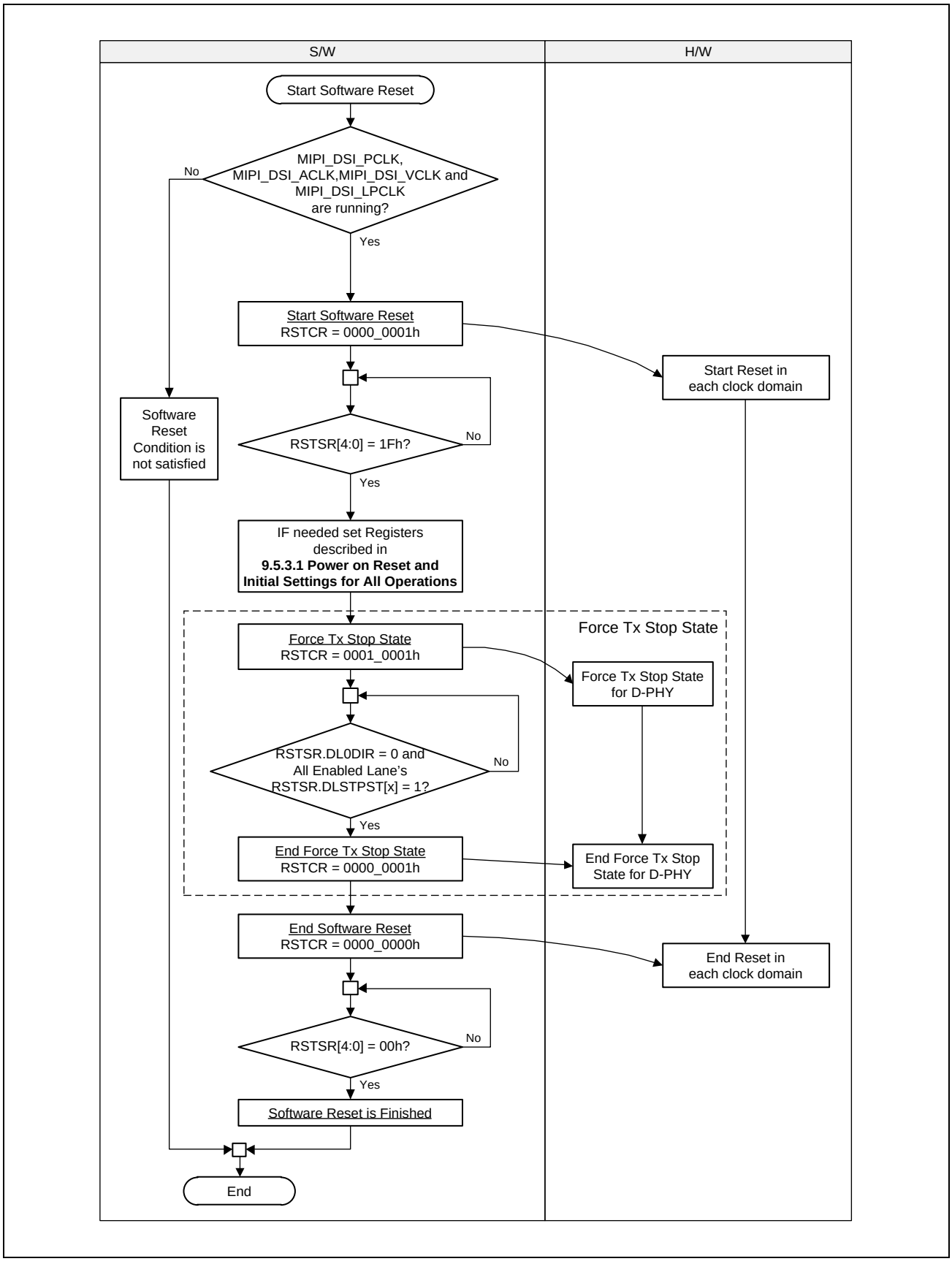


Figure 9.5-5 Software Reset

9.5.3.3 Start/Stop of HS Clock

The following describes the HS Clock Start / Stop sequence. HS Clock Start / Stop is controlled by software.

Figure 9.5-6 shows the operation at HS Clock Start.

Write 1b to TXSETR.CLEN to enable DPHY clock lanes before starting Video-Input Operation or Sequence Operation channel 1 operation.

After the clock lane is enabled and the source clock of the clock lane is stable, write 1b to HSCLKSETR.HSCLKRUN to set the clock lane to HS. It is forbidden to change HSCLKRUN while Sequence Channel is in operation.

At the same time as writing 1b to HSCLKSETR.HSCLKRUN, set continuous clock mode and non-continuous clock mode with HSCLKSETR.HSCLKMODE.

In continuous clock mode, always set the clock lane to HS. PLSR.CLLP2HS is set to 1b when the clock lane transitions from LP to HS. After checking, perform the following operations.

In non-continuous clock mode, the DSI-Tx Module automatically transitions the clock lane from LP to HS depending on the presence or absence of HS Transmission. PLSR.CLLP2HS and PLSR.CLHS2LP are set for each transition, but confirmation is not required.

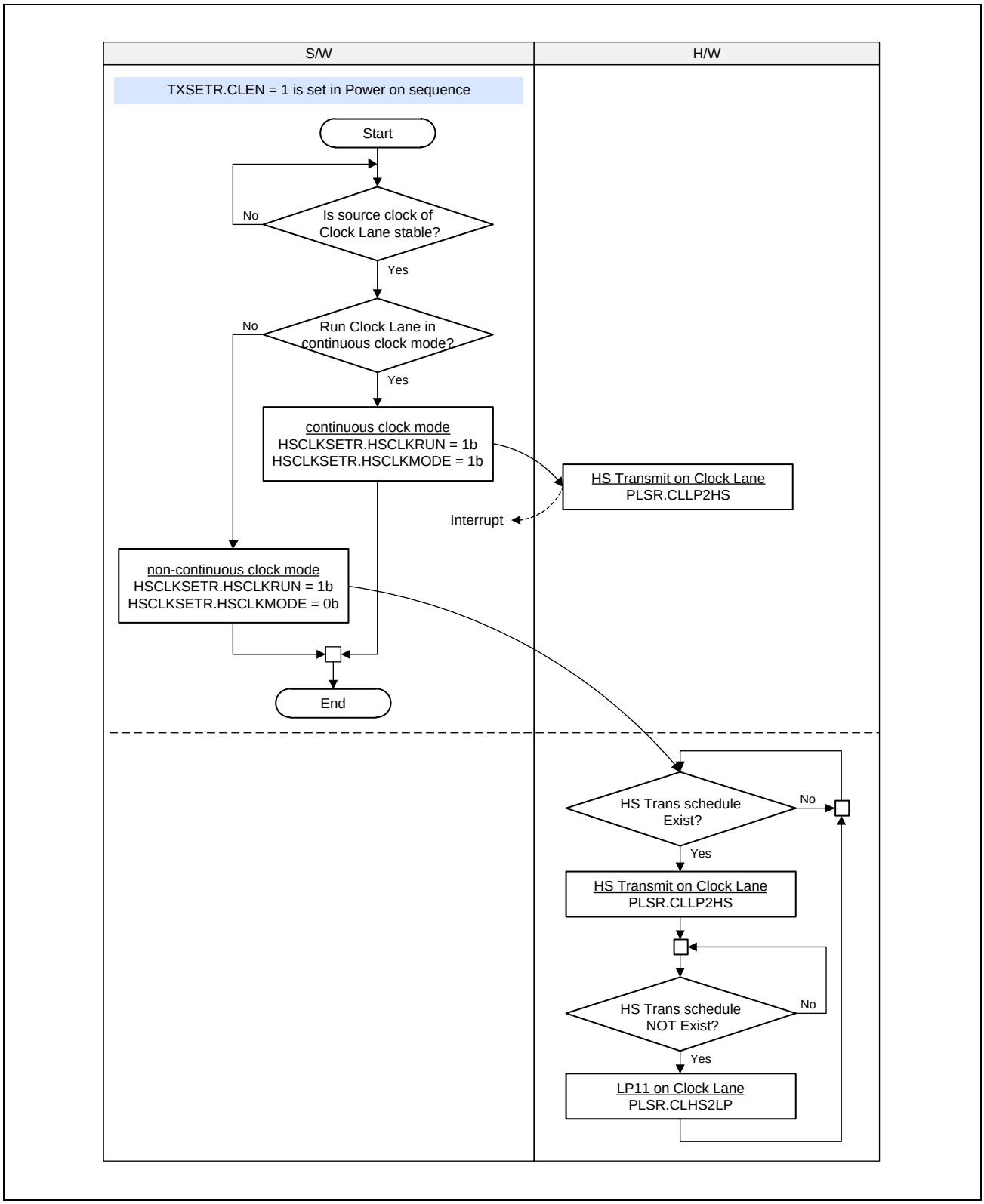


Figure 9.5-6 Start of HS Clock

Figure 9.5-7 shows the operation when HS Clock Stop. After confirming that Video-Input Operation and Sequence Operation channel 1 are not running, write 0b to HSCLKSETR.HSCLKRUN to make the clock lane LP. PLSR.CLHS2LP is set to 1b when the clock lane transitions from HS to LP. In non-continuous clock mode, you do not need to check PLSR.CLHS2LP because the clock lane is likely to be LP already when you write 0b to HSCLKSETR.HSCLKRUN. It is possible to check the clock lane status by reading PLSR.CLSTPST.

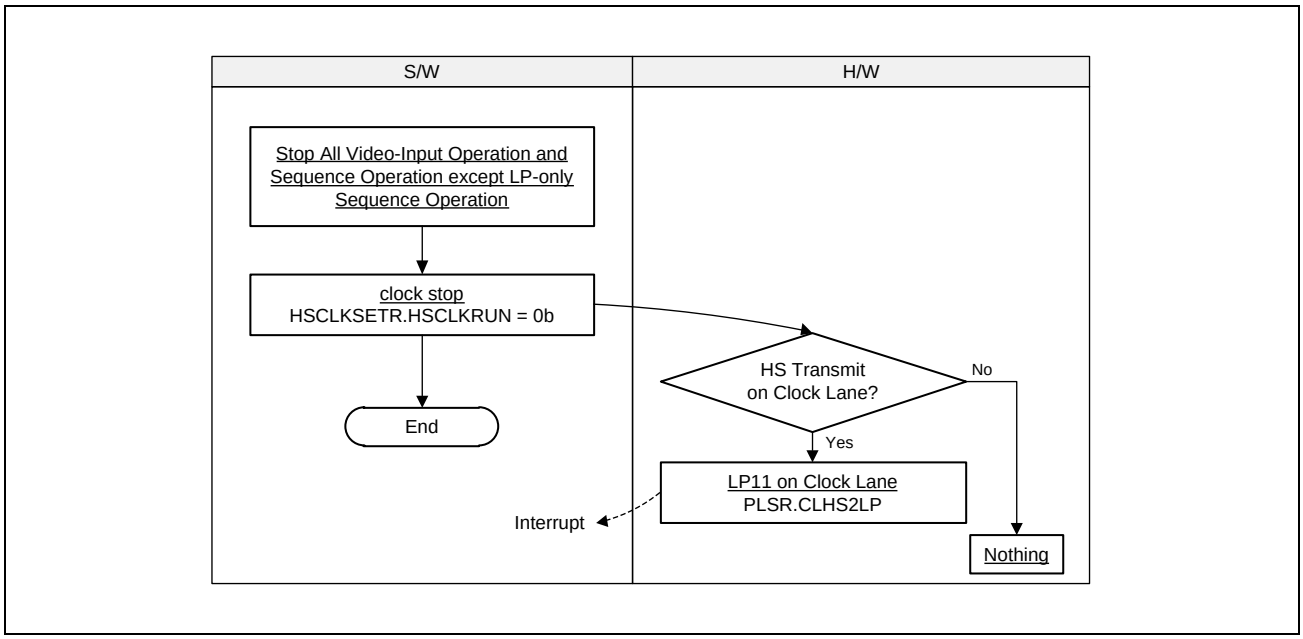


Figure 9.5-7 Stop of HS Clock

9.5.3.4 Sequence Operation

9.5.3.4.1 Basic Running Sequence

Sequence Operation requires initial settings before the operation starts.

- Channel Register Setting
  - Set SQCHxSET1R, SQCHxIER
- Prepare Write Data
  - Prepare Write Data to Memory Area
  - Prepare Write Data to TXPPDxR Registers
- Allocate Read Data Area
  - Clear RXRSSR Register if previous valid flag remains.
  - Allocate each read action’s Memory area.
- Write all Descriptors
- Confirm running HSCLK if Sequence channel 1 is targeted to run.

When “1b” is written to SQCHxSET0R.START, Sequence Operation channel x starts the processing of Descriptor, and when the processing of Descriptor of SQCHxDSCyAR.NXACT = “00b” is completed, the processing of Descriptor ends.

Since Descriptor consists of SRAM, the initial value is indefinite. When the user sets the Descriptor, set all the bits.

After writing 1b to SQCHxSET0R.START on Sequence Operation channel x and SQCHxSR.RUNNING becomes 1b, It is prohibited to change the settings of SQCHxSET1R, SQCHxDSCyAR, SQCHxDSCyBR, SQCHxDSCyCR, and SQCHxDSCyDR until SQCHxSR.RUNNING becomes 0b.

HS-related operations are prohibited on Sequence Operation channel 0.

9.5.3.4.2 Single Packet Transmission

The following describes sending non-read packets using the Sequence Operation channel.

For non-read packet transmission, set the following registers before packet transmission.

- SQCHxDSCyAR.NXACT: “00b”
- SQCHxDSCyAR.FMT: “0b” Short Packet or “1b” Long Packet
- SQCHxDSCyAR.SPD: “0b” High Speed or “1b” Low Speed
- SQCHxDSCyAR.BTA: “00b” without BTA or “01b” with BTA
- SQCHxDSCyAR.VC, DT, DATA0, DATA1
- SQCHxDSCyBR.DTSEL: Specify payload data storage location for non-read packets
  - “00b” TXPPD0R – TXPPD3R. Payloads size is limited up to 16 Bytes.
  - “01b” Memory space. Payloads size is limited up to SQCHxSET1R.CHBUFSZ Bytes.
- SQCHxDSCyCR.AUXOP “0b”

When setting a Long Packet with SQCHxDSCyAR.FMT, prepare the payload data in the location specified by SQCHxDSCyBR.DTSEL in the form of little endian.

No payload data is required when configuring Short Packets with SQCHxDSCyAR.FMT.

After the above settings, write “1b” to SQCHxSET0R.START to send non-read packets.

The user can know the completion of Single Packet Transmission by asserting SQCHxSR.ADESFIN.

If “01b” is set in SQCHxDSCyAR.BTA, BTA is performed after sending a non-read packet.

An ACK trigger or Acknowledge and Error Report Packet may be received when the bus rights are transferred to the peripheral.

The user can get an overview of the reception result by reading the RXRSSxR register with the number specified by SQCHxDSCyCR.ACTCODE. Also, by reading the RXSR register, the user can know the details of the reception result.

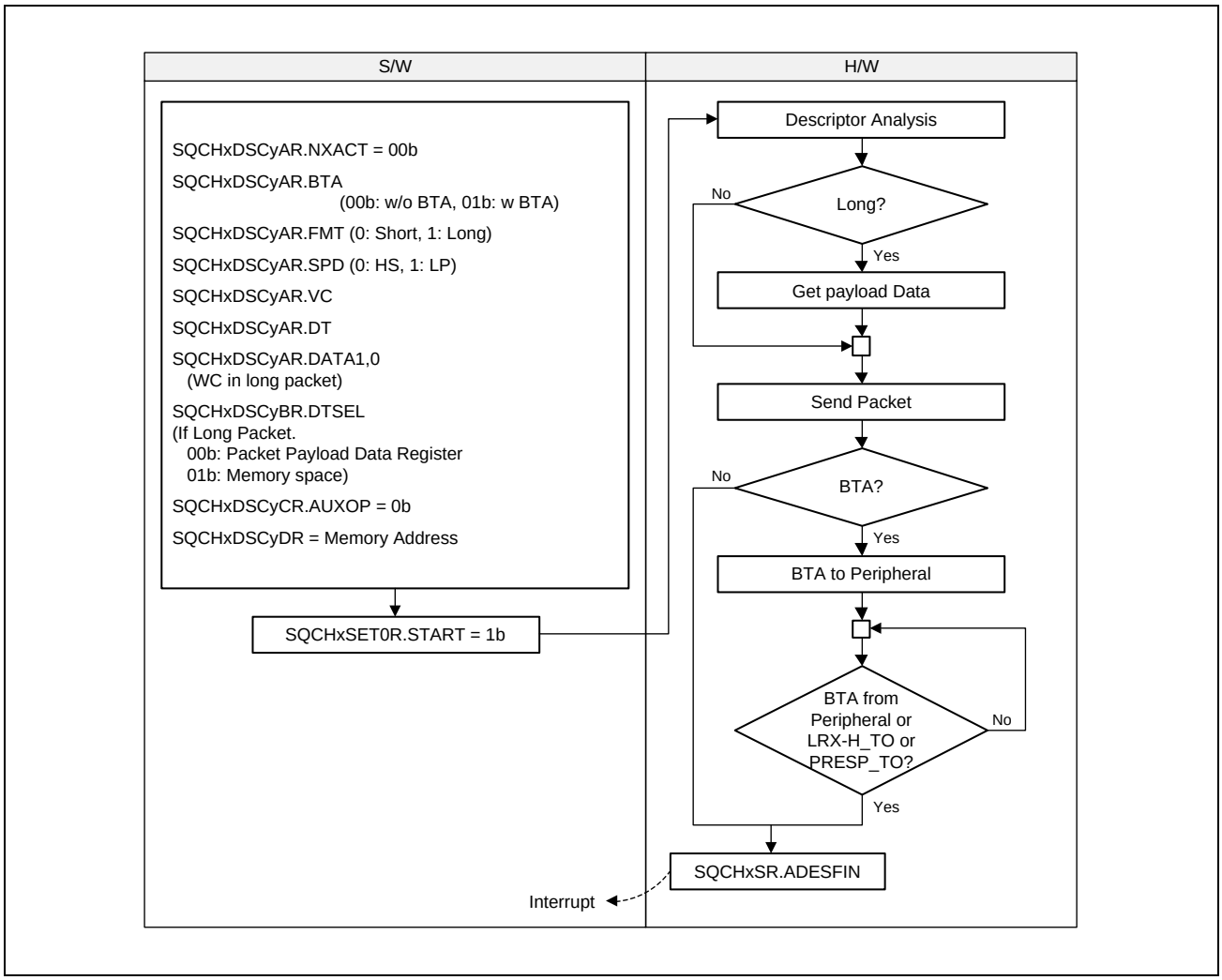


Figure 9.5-8 Single Packet Transfer

9.5.3.4.3 Single Packet Reception

This section describes sending read packets and receiving response packets using the Sequence Operation channel.

Since the read packet is a short packet, set the following registers before sending the packet.

- SQCHxDSCyAR.NXACT: “00b”
- SQCHxDSCyAR.FMT: “0b” Short Packet
- SQCHxDSCyAR.SPD: “0b” High Speed or “1b” Low Speed
- SQCHxDSCyAR.BTA: “10b” Read Request with BTA
- SQCHxDSCyAR.VC, DT, DATA0, DATA1
- SQCHxDSCyBR.DTSEL: Specify the payload data storage location of the response packet
  - “00b” RXPPD0R – RXPPD3R. Payloads size is limited up to 16 Bytes.
  - “01b” Memory space. Payloads size is limited up to RXBUFSZR.RXBUFSZ Bytes.
- SQCHxDSCyCR.AUXOP: “0b”
- SQCHxDSCyCR.ACTCODE: Rx Result Save Slot number

When SQCHxDSC00BR.DTSEL = “00b” is set, receiving payload data of 17 bytes or more is not supported. Therefore, if there is a possibility of receiving payload data of 17 bytes or more, set SQCHxDSC00BR.DTSEL = “01b”.

After the above settings, write “1b” to SQCHxSET0R.START to send the read packet.

BTA is performed after sending the read packet. When the bus right is transferred to the peripheral, the response packet is received. At this time, the Acknowledge and Error Report Packet may also be received at the same time.

The user can know the completion of Single Packet Reception by asserting SQCHxSR.ADESFIN.

The user can get an overview of the reception result by reading the RXRSSxR register with the number specified by SQCHxDSCyCR.ACTCODE, and can know the details of the previous reception result by reading the RXSR register.

If the received response packet is a Short Packet or a Long Packet with WC = 0b, the setting value of SQCHxDSC00BR.DTSEL is meaningless.

Figure 9.5-9 shows the flow of Single Packet Reception.

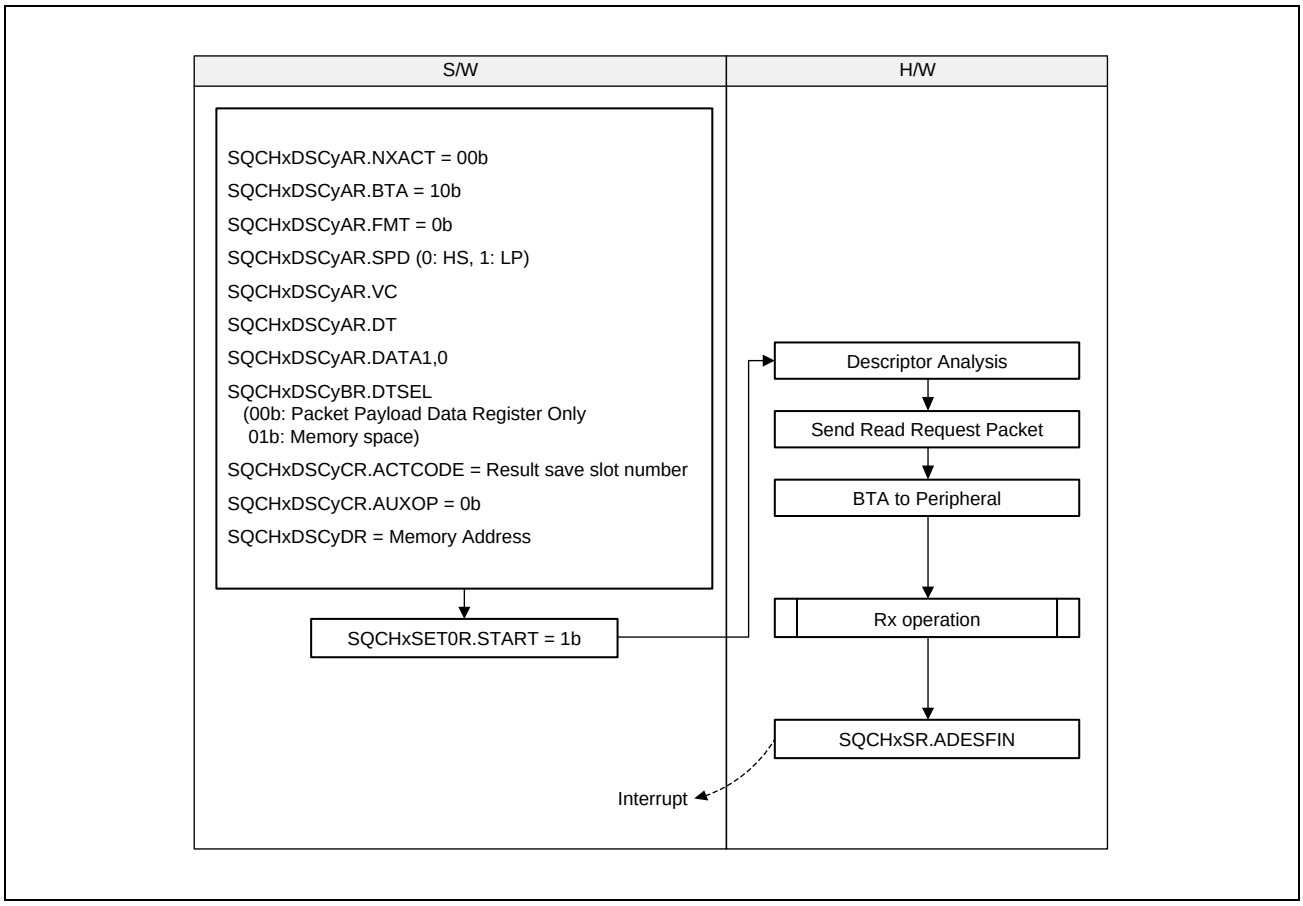


Figure 9.5-9 Single Packet Receive

9.5.3.4.4 Reset Trigger Transmission

The DSI-Tx Module can send triggers using the Sequence Operation channel.

Since the DSI-Tx Module is a host, the only trigger that can be sent is a reset trigger.

After setting SQCHxDSCyAR.NXACT = “00b”, SQCHxDSCyCR.AUXOP = “1b”, and SQCHxDSCyCR.ACTCODE = “00h”, write “1b” to SQCHxSET0R.START to send a reset trigger.

The user can know the completion of reset trigger transmission by asserting SQCHxSR.ADESFIN.

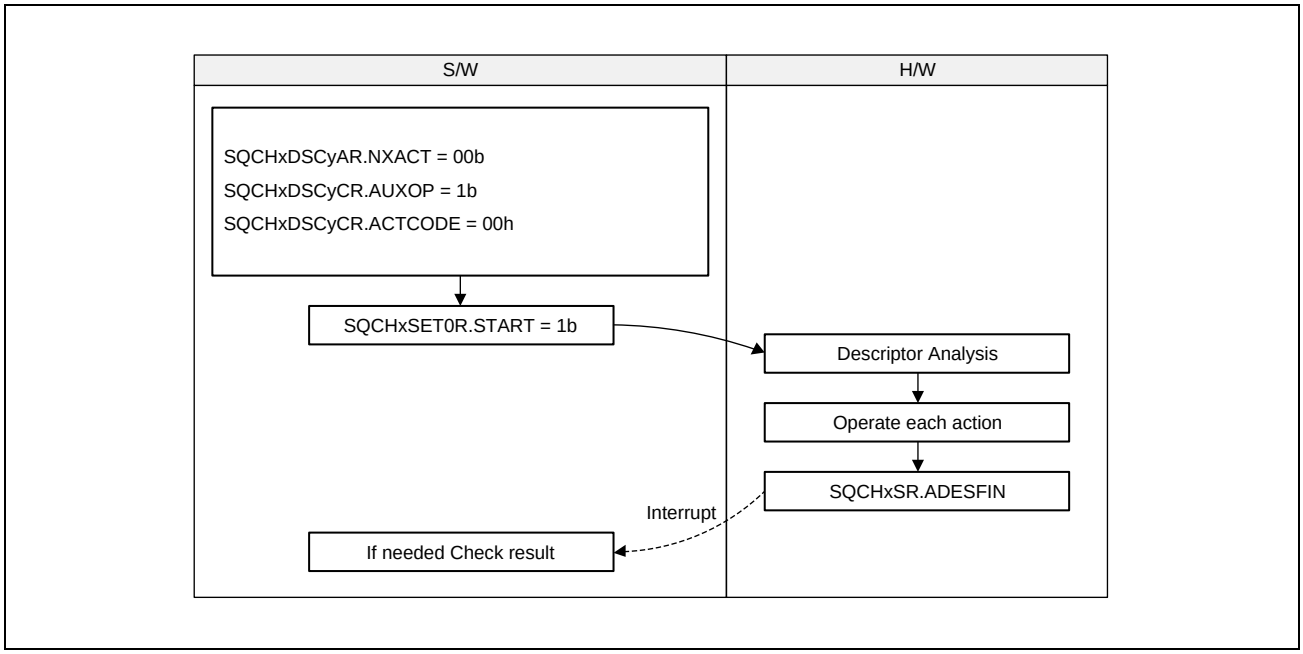


Figure 9.5-10 Trigger Transmission

9.5.3.4.5 Execute Operation in Sequence

The following describes the descriptor in Sequence Operation.

- By writing “1b” to SQCHxSET0R.START, Sequence Operation channel x will be enabled and processing will be performed from descriptor # 0.
- When the processing of the corresponding descriptor is completed, the next processing follows SQCHxDSCyAR.NXACT.
  - SQCHxDSCyAR.NXACT= “00b”:
    Sequence Operation channel x stop after the processing of the corresponding descriptor is completed
  - SQCHxDSCyAR.NXACT= “01b”:
    After the processing of the corresponding descriptor is completed, the next descriptor is executed. The next descriptor after descriptor # N is descriptor # (N + 1).

9.5.3.4.6 Receiving EoTp

The DSI-Tx Module can receive End of Transmission Packets (EoTp).

The DSI-Tx Module notifies the reception of EoTp by setting RXSR.RXEOTP to “1b”.

9.5.3.4.7 Acknowledge and Error Reporting Mechanism

The user can know the reliability of the serial bus by using “Acknowledge and Error Reporting Mechanism”.

If Peripheral detects an error on the serial bus during the period from the previous Peripheral to Host communication to this Peripheral to Host communication, Peripheral will send an “Acknowledge and Error Report” packet.

When the DSI-Tx Module receives the “Acknowledge and Error Report” packet, it sets “1b” to RXRSSxR.RXAKE, RXSR.RXAKE, and SQCHxSR.RXAKE. The Virtual Channel Identifier and Error Report of the latest “Acknowledge and Error Report” packet received are stored in the AKEPLATIR register, and the integrated value is displayed in the AKEPACMSR register.

9.5.3.5 Video-Input Operation

9.5.3.5.1 Video-Input Timing

Video-Input Timing and Video-Input Parameters are described in **Figure 9.5-11** and **Table 9.5-6**.

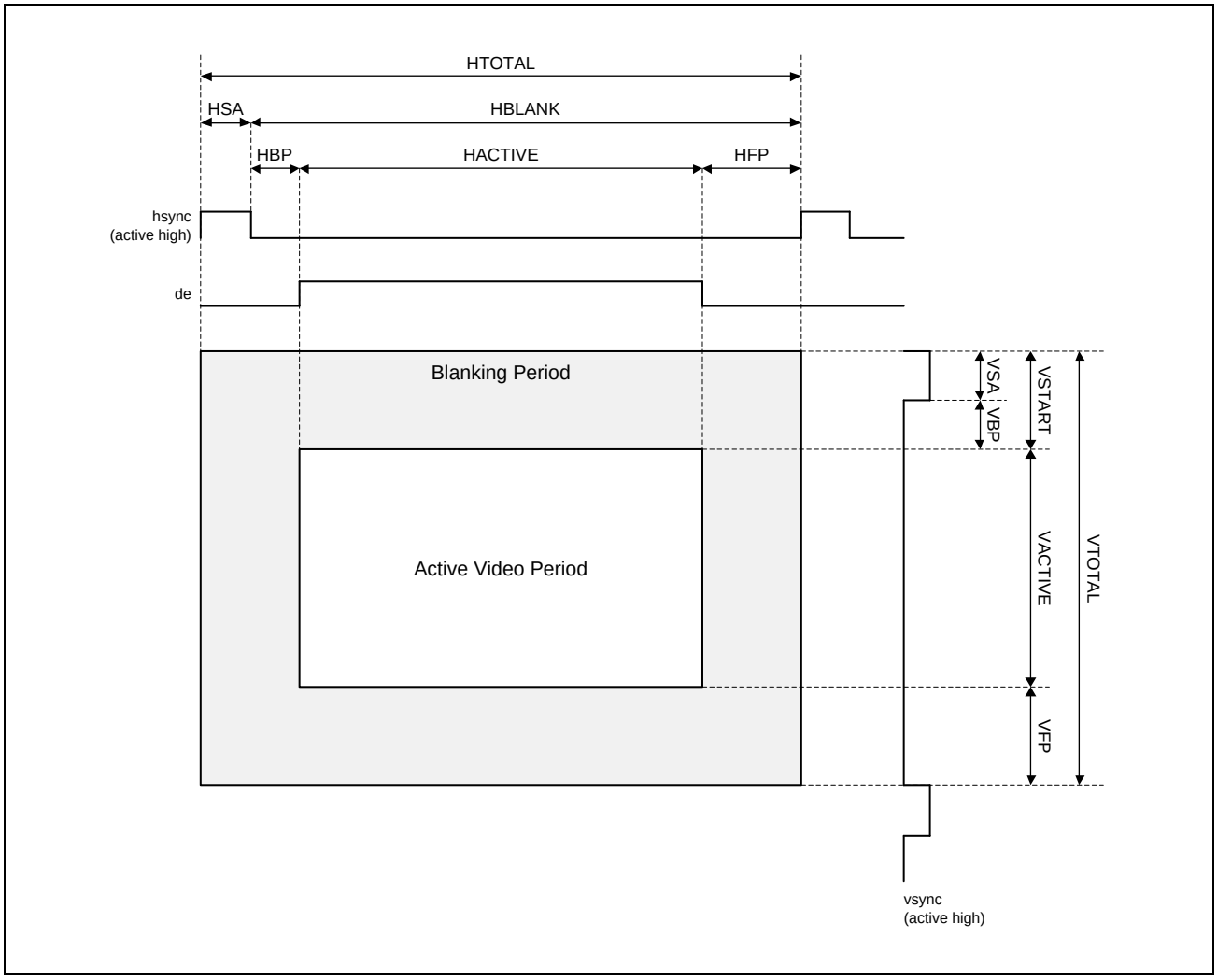


Figure 9.5-11 Video Input Timing

Table 9.5-6 Configurations for Video Input Interface (1/2)

- Common Rules for all mode

Parameter	Description
All	All parameters related to Input Video should be set to the value of not less than 1b. Example for “All parameters”: HSA, HBP, HACTIVE, HFP, VSA, VBP, VACTIVE, VFP
HACTIVE	

VICH1PPSETR.DT	VICH1HSSETR.HACTIVE
1Eh: Packed Pixel Stream, 18-bit RGB	This value should be the multiple of 4.
2Eh: Loosely Packed Pixel Stream, 18-bit RGB 3Eh: Packed Pixel Stream, 24-bit RGB	No limitation

Parameter	Description
HFP	Round down (HFP × BPP/ 8, 0) >= 12, (12 = means Pixel Packet Header & footer 6 + Blanking Packet Header & footer 6)
HTOTAL	

VICH1PPSETR.DT	VICH1HSSETR.HSA + VICH1HPSETR.HBP + VICH1HSSETR.HACTIVE + VICH1HPSETR.HFP
1Eh: Packed Pixel Stream, 18-bit RGB	This value should be the multiple of 4.
2Eh: Loosely Packed Pixel Stream, 18-bit RGB 3Eh: Packed Pixel Stream, 24-bit RGB	No limitation

- Additional Rules for non-burst Sync Pulse mode

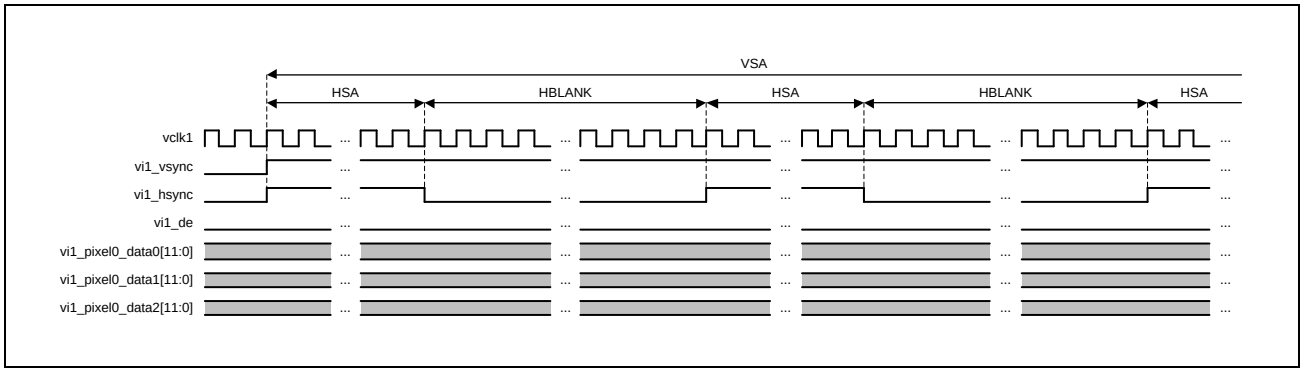
Parameter	Description
Sequence Operation Packet	When a packet by sequence operation is transmitted during Video Operation (VICH1SR.RUNNING=1b), the period of (HBP + HACTIVE + HFP) should be set the value longer than sum of below periods. • Period for the Tx Sync Event packet (4 bytes) • Period for the packet by sequence operation • Period for the Header and CRC of a Blanking Packet (6 bytes)
HBP+HACTIVE+HFP	Transition of HS-LP-HS is caused within the period between current Sync Event Packet and next Sync Event Packet, the period of (HBP+HACTIVE+HFP) should be set to the value within the range described below. (HBP + HACTIVE + HFP) > (period for the Tx Sync Event Packet) + (period for transition time as HS-LP-HS)
HSA	Transmission of Blanking Packet is caused within the period between current Sync Event Packet and next Sync Event Packet, the period of HSA should be set to the value within the range described below. (HSA × BPP)/8 >= 10b Notice, HSA should be set to the value not less than minimum value of (period for the Tx Sync Event Packet) + (period for the Tx Blanking Packet).
HBP	Transmission of Blanking Packet is caused within the period between current Sync Event Packet and next Sync Event Packet, the period of HPB should be set to the value within the range described below. (HBP × BPP)/8 >= 10b Notice, HPB should be set to the value not less than minimum value of (period for the Tx Sync Event Packet) + (period for the Tx Blanking Packet).

Table 9.5-6 Configurations for Video Input Interface (2/2)

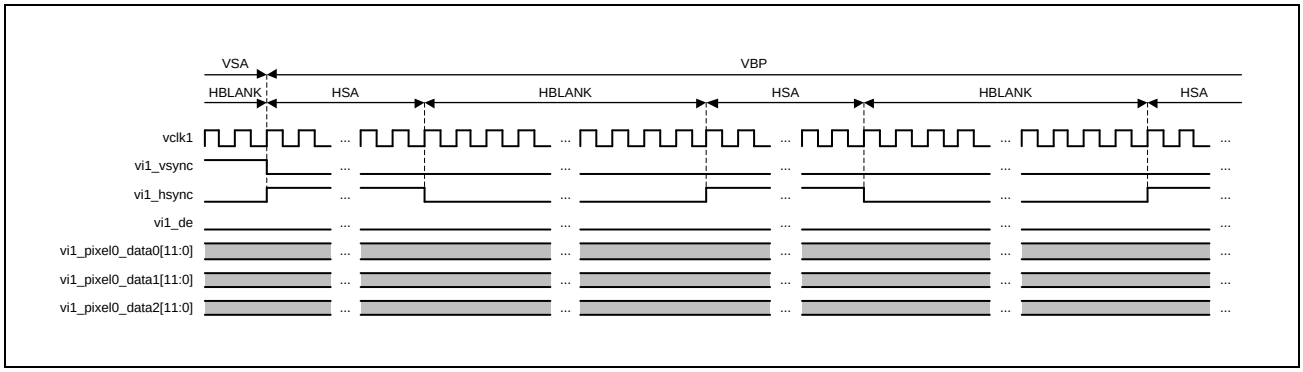
- Additional Rules for non-burst Sync-Event mode and Burst mode

Parameter	Description
Sequence Operation Packet	When a packet by sequence operation is transmitted during Video Operation (VICH1SR.RUNNING=1), the period of HTOTAL should be set the value longer than sum of below periods. • Period for the Tx Sync Event packet (4 bytes) • Period for the packet by sequence operation • Period for the Header and CRC of a Blanking Packet (6 bytes)
HTOTAL	Transition of HS-LP-HS is caused within the period between current Sync Event Packet and next Sync Event Packet, the period of HTOTAL should be set to the value within the range described below. HTOTAL > (period for the Tx Sync Event Packet) +(period for transition time as HS-LPHS)
HSA+HBP	Transmission of Blanking Packet is caused within the period between current Sync Event Packet and next Sync Event Packet, the period of (HSA+HBP) should be set to the value within the range described below. ((HAS + HBP) × BPP)/8 >= 10b Notice, HAS + HBP should be set to the value not less than minimum value of (period for the Tx Sync Event Packet) + (period for the Tx Blanking Packet).

(1) VSA Start (and VFP End)

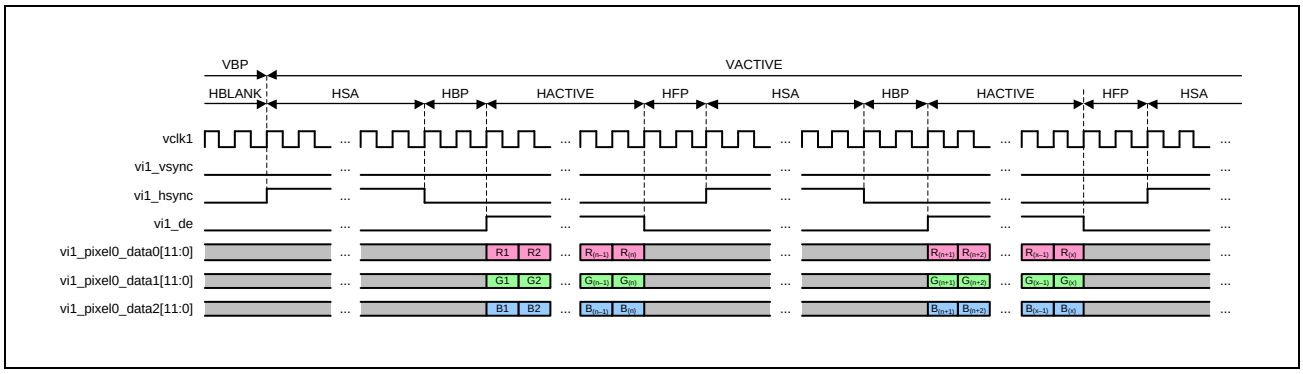


(2) VSA End and VBP Start

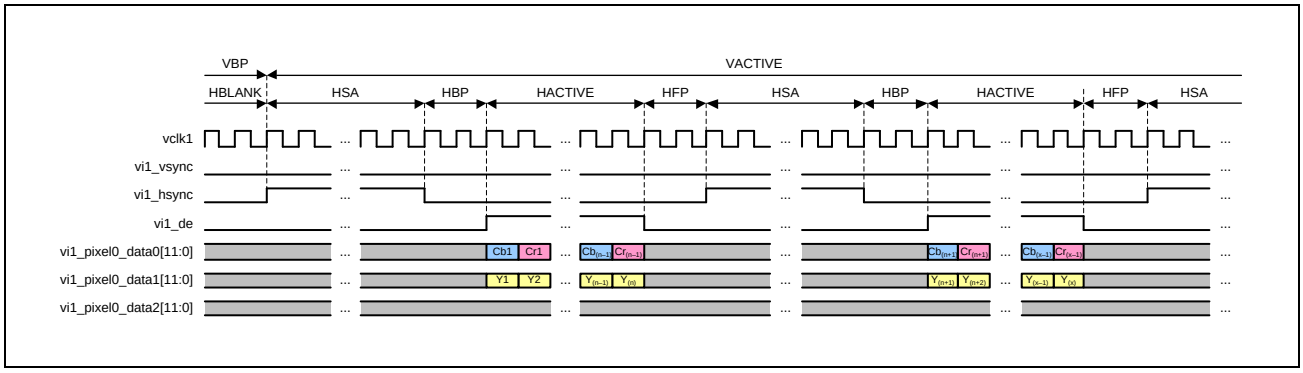


(3) VBP End and VACTIVE Start

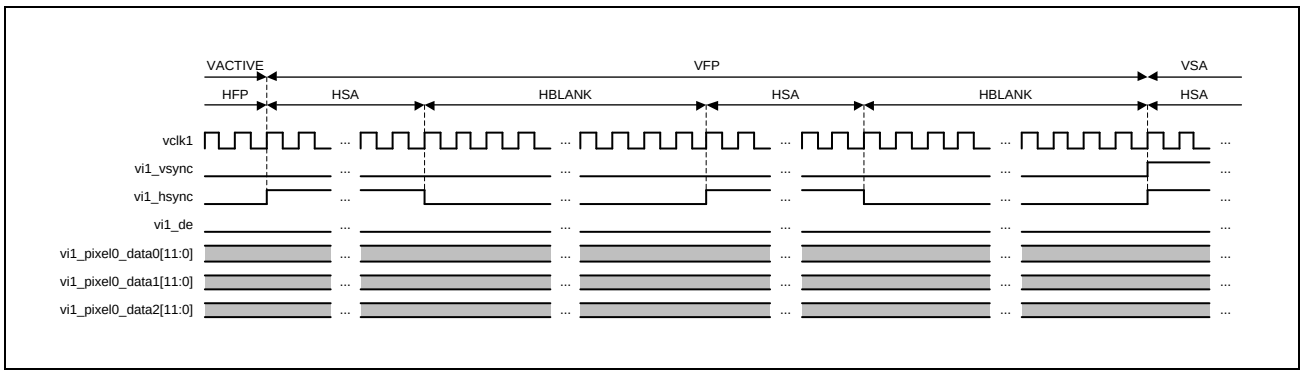
(a) RGB



(b) YCbCr 4: 2: 2



(4) VACTIVE End and VFP Start



9.5.3.5.2 Video-Input Start, End, Change Setting

1) Video Input Start

Video input must be started after Video-Input Operation Start sequence. It must be start from V-Sync start edge.

2) Video Input End

End of Video Input must follow sequence of **9.5.3.5.4 End of Video-Input Operation**.

3) Change of Video Input Setting

Changing of Video-Input setting must be done during Video-Input operation is not running.

9.5.3.5.3 Start of Video-Input Operation

The procedure for Start of Video-Input Operation is shown in **Figure 9.5-12**.

1. Initialize common settings. See **9.5.3.1 Power on Reset and Initial Settings for All Operations**.
2. Some peripheral setting may be done by Sequence operation. Since it is differed from peripheral devices, it is out of this document.
3. Confirm Video-Input clock and hsclk stable and set HS clock. See **9.5.3.3 Start/Stop of HS Clock**.
4. Set Video-Input Operation channel parameters (VICH1PPSETR, VICH1VSSETR, VICH1VPSETR,VICH1HSSETR, VICH1HPSETR)
5. Calculate VICH1SET1R.DLY value and set. See **9.5.3.5.9 Delay about Video-Input Operation**.
6. Set “1b” to VICH1SET0R.VSTART with other VICH1SET0R setting values.
7. Wait until VICH1SR.VIRDY = 1b.
8. Start Video signal input

VICH1SET0R, VICH1SET1R, VICH1PPSETR, VICH1VSSETR, VICH1VPSETR, VICH1HSSETR and VICH1HPSETR are prohibited to change value between VICH1SET0R.VSTART = 1b and VICH1SR.RUNNING=0b after VICH1SET0R.STOP = 1b.

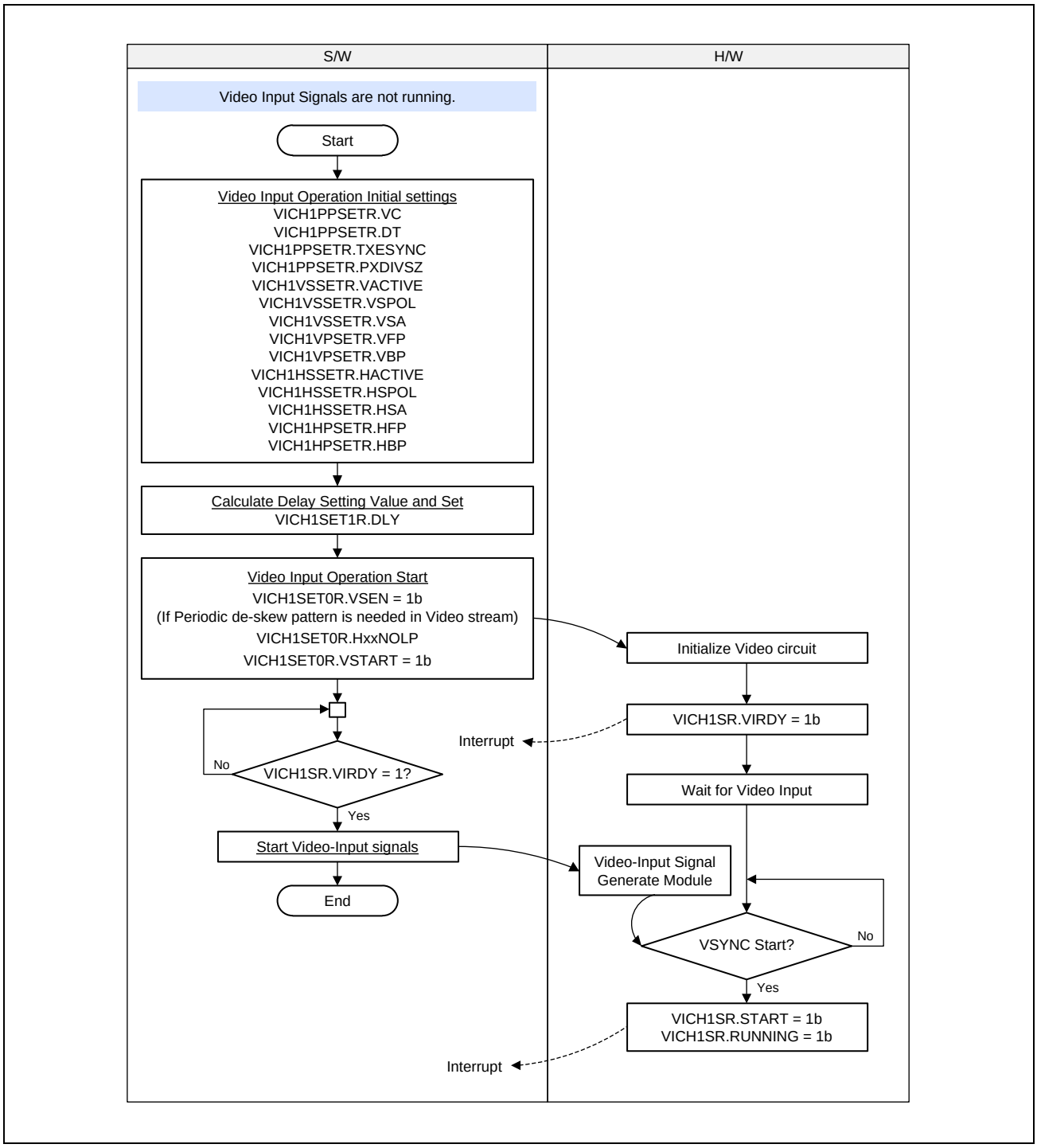


Figure 9.5-12 Video-Input Channel Start

9.5.3.5.4 End of Video-Input Operation

The video-input operation stop flow is shown in **Figure 9.5-13**. Execute the flow when VICH1SR.RUNNING = 1b.

1) VICH1SET0R.VSTPAFT (Figure 9.5-13)

Write 1b to VICH1SET0R.VSTPAFT to request the Video-Input Operation to stop. When the DSI-Tx Module detects the start of one frame (assertion of VSYNC), it stops sending video mode packets and sets VICH1SR.STOP to 1b.
After checking VICH1SR.STOP, stop the video input. The video input can be stopped at any time after VICH1SR.STOP becomes 1b.
After stopping the video input, wait for LINKSR.HSBUSY to become 0b.

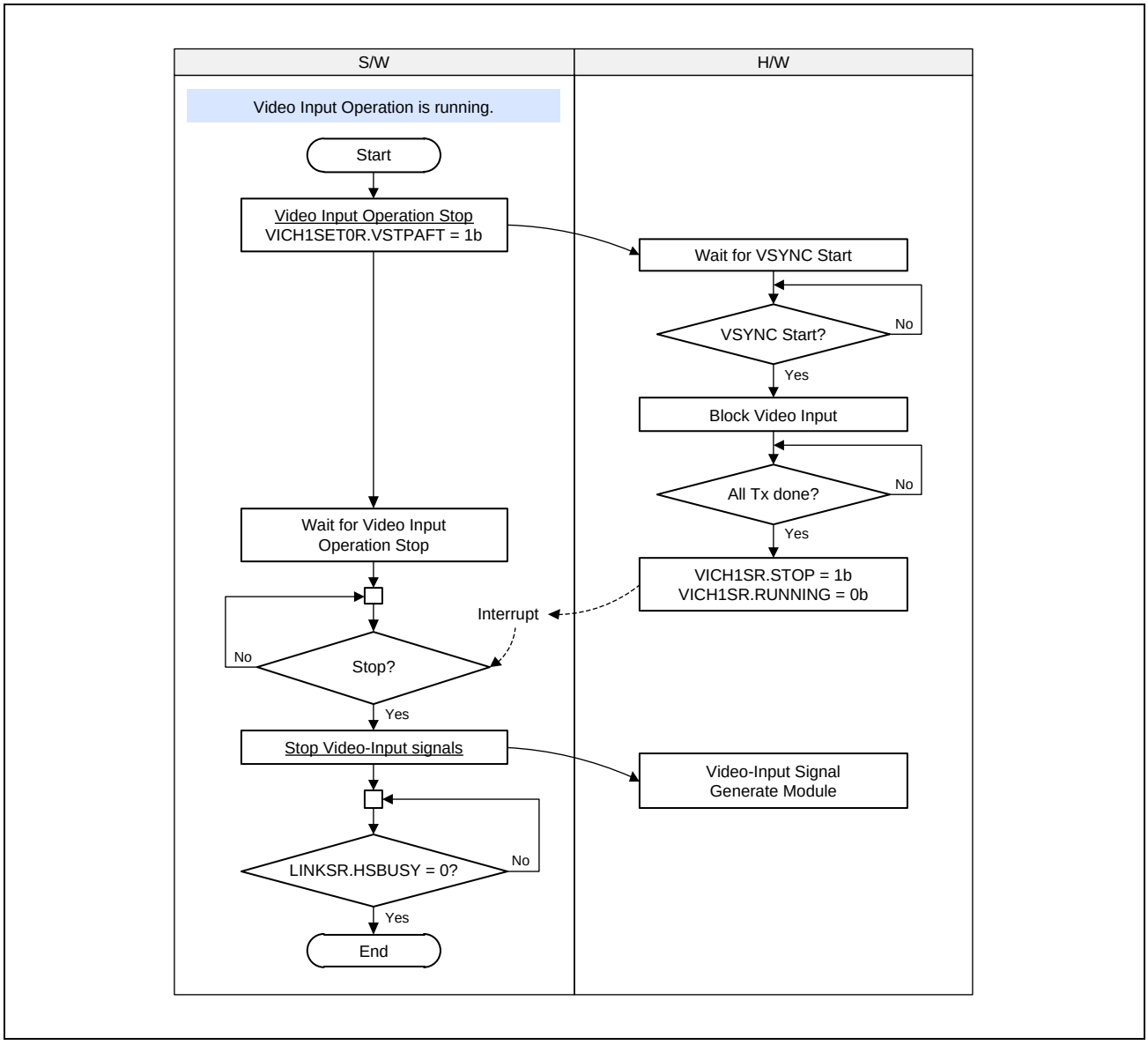


Figure 9.5-13 End of Video-Input Operation (VICH1SET0R.VSTPAFT)

9.5.3.5.5 Burst Mode

The DSI-Tx Module supports Burst Mode.

Burst Mode is a mode that time-compresses pixel data.

Make the D-PHY bandwidth wider than the Video-Input bandwidth for time compression of pixel data. Video transmissions with a 1-line pixel data size (VICH1HSSETR.HACTIVE × VICH1SET1R.BPP / 8) greater than 65,535 Bytes are not supported.

Video clock Frequency [Hz] × Video Pixel Bit Depth (BPP) [bit]
<= DSI HS Byte clock Frequency[Hz] × 8[bit] × Number of DSI HS Data Lane

In the case of Video-Input bandwidth equals DSI output bandwidth, it will be indistinguishable from non-burst sync-event mode.

9.5.3.5.6 Non-Burst Mode

The DSI-Tx Module supports Non-Burst Mode.

Make the D-PHY bandwidth the same as the Video-Input bandwidth. Video transmissions with a 1-line pixel data size (VICH1HSSETR.HACTIVE × VICH1SET1R.BPP / 8) greater than 65,535 bytes are not supported.

Video clock Frequency [Hz] × Video Pixel Bit Depth (BPP) [bit]
= DSI HS Byte clock Frequency[Hz] × 8[bit] × Number of DSI HS Data Lane

9.5.3.5.7 LP Transition and Blanking

The DSI-Tx Module automatically transitions the Data Lane from HS to LP if both of the following two conditions are met:

If it does not transition from HS to LP, it sends a Blanking Packet with the same VC as the VC in the previous packet to maintain the HS.

- There is time to make an HS => LP => HS transition before the next HS transfer starts.
- The period is not prohibited by VICH1SET0R.xxxNOLP to transition to LP

9.5.3.5.8 Non video packet action during Video-Input Operation Running

1) Packet Transmission/Reception by Sequence operation

The DSI-Tx Module can send and receive packets (see **9.5.3.4.2 Single Packet Transmission** and **9.5.3.4.3 Single Packet Reception**) using Sequence Operation while Video-Input Operation is operating*[1].

Make sure that all Sequence Operations are finished before starting the Video-Input Operation.

While Video-Input Operation is operating*[1], packet transmission / reception using Sequence Operation is performed only immediately after the HSE packet (VICH1PPSETR.TXESYNC = 1b) or VSS packet (VICH1PPSETR.TXESYNC = 0b) on the first horizontal line. Also, sending packets using Sequence Operation is possible only with HS (SQCHxDSCyAR.SPD = 0b). Sending with LP (SQCHxDSCyAR.SPD = 1b) is not supported. Send and receive packets that are completed within the BLLP period of the first horizontal line.

Execution of Sequence Operation processing is prohibited except for sending and receiving HS packets.

Note 1. “While Video-Input Operation is operating” here is different from VICH1SR.RUNNING. It means the period from setting VICH1SET0R.VSTART to 0b in the flow of **Figure 9.5-12** until VICH1SR.RUNNING becomes 0 in the flow of **Figure 9.5-13**. Also, Note that there is some latency from executing VICH1SET0R.VSTART to being transmitted to the inside of the module for clock transfer.

9.5.3.5.9 Delay about Video-Input Operation

Video-Input Operation run in some latency which call in the DSI-Tx Module “Delay”. “Delay” value must set to VICH1SET1R.DLY before operation.

“Delay” value is defined by calculation using below parameters.

- Clock frequencies
  - Video channel 1 clock (MIPI_DSI_VCLK) frequency
  - DSI High-Speed Transmit Word Clock (TxWordClkHS, hsclk) frequency
- Video parameters
  - HSA period, set by VICH1HSSETR.HSA
  - HACTIVE period, set by VICH1HSSETR.HACTIVE
  - HBP period, set by VICH1HPSETR.HBP
  - HFP period, set by VICH1HPSETR.HFP
  - BPP value, shown in VICH1SET1R.BPP and derived from VICH1PPSETR.DT
- Video transmit settings
  - The number of active lanes, set by TXSETR.NUMLANEUSE
  - Transmit mode, set by VICH1PPSETR.TXESYNC
  - No LP, set by VICH1SET0R.HSANOLP, HBPNOLP, HFPNOLP
  - Clock mode, set by HSCLKSETR.HSCLKMODE
  - Clock stop time, set by CLSTPTSETR.CLKSTPT
  - Go LP and back time, set by LPTRNSTSETR.GOLPBKT
- Other parameters
  - Video-Input channel buffer size, shown in VICH1SET1R.CHBUFSZ

9.5.3.6 Enter/Exit of ULPS

The following describes the ULPS transition / return sequence.

The clock lane and data lane (lanes 0 to 3) can be independently transitioned / restored to ULPS.

The clock lane can make ULPS transitions when TXSETR.CLEN = 1b. It is forbidden to change TXSETR.CLEN to 0b while the clock lane is ULPS.

The data lane can make ULPS transitions when TXSETR.DLEN = 1b. It is forbidden to change TXSETR.DLEN to 0b while the data lane is ULPS.

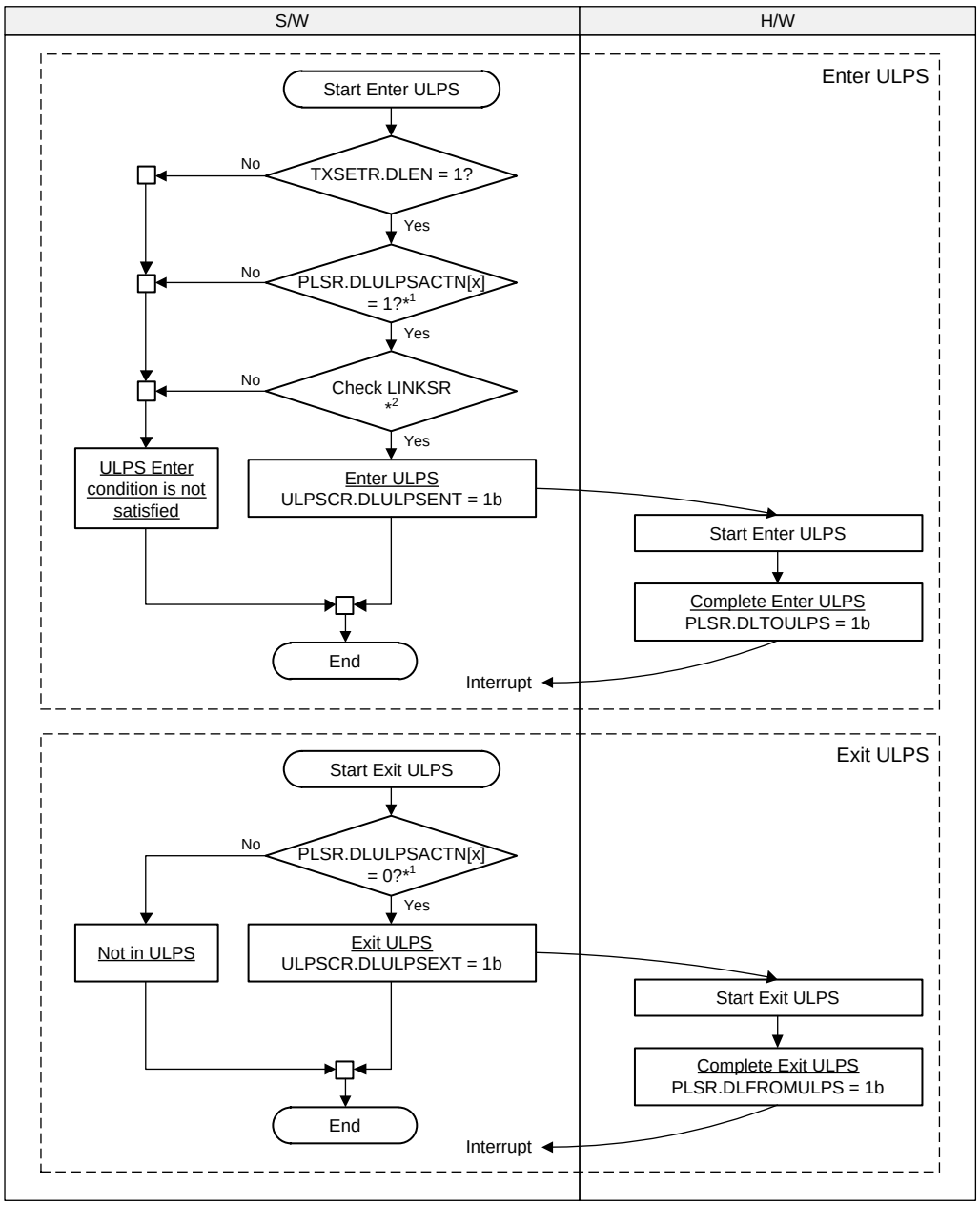
Data lanes enabled in TXSETR.NUMLANEUSE can be ULPS transitioned / restored. ULPS transition / return of data lanes is done simultaneously for all valid lanes. Only certain data lanes cannot be ULPS transitioned / restored.

When the data lane transitions to ULPS, another operation*[1] using the data lane cannot be performed until ULPS is restored.

When the clock lane transitions to ULPS, another operation*[2] using the clock lane cannot be performed until ULPS is restored.

Note 1. Sequence Operation, Video-Input Operation.

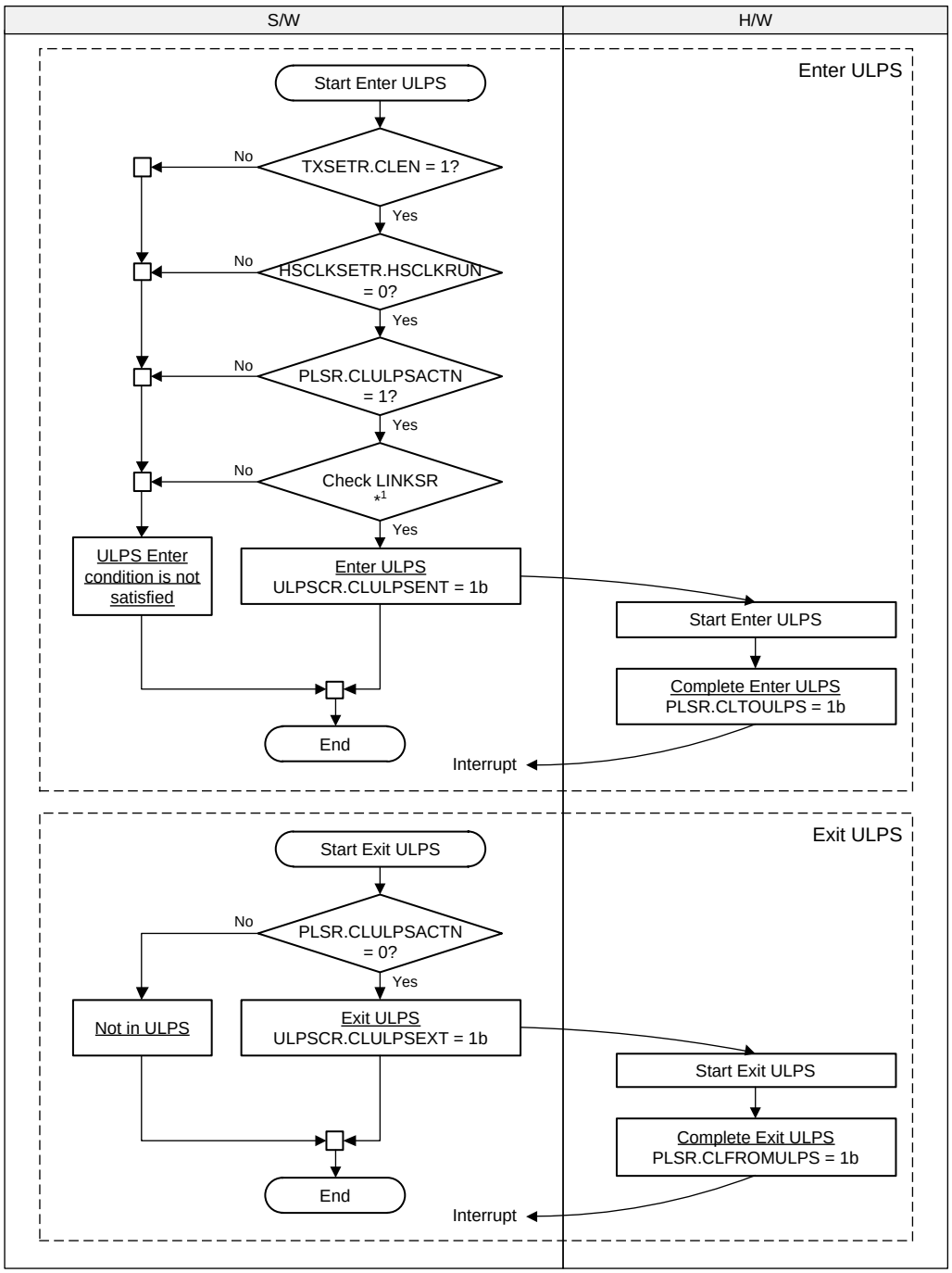
Note 2. HS Sequence Operation, Video-Input Operation.



Note 1. Check all enabled data lanes selected by TXSETR.NUMLANEUSE

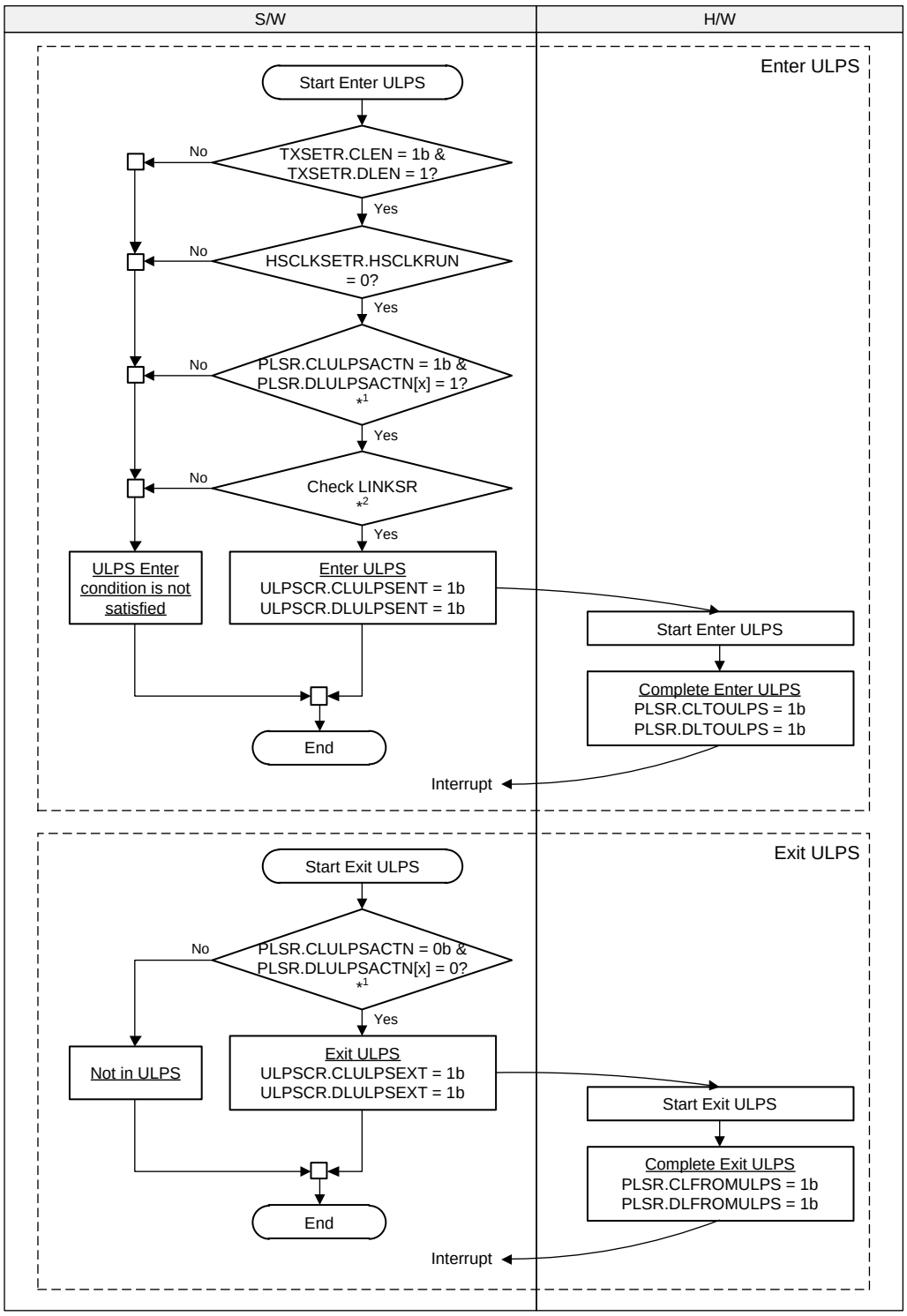
Note 2. (SQCHRUN0 = 0b and SQCHRUN1 = 0b and VICHRUN1 = 0b and HSBUSY = 0b and LPBUSY = 0b)?

Figure 9.5-14 Enter/Exit of ULPS (Data Lane)



Note 1. (SQCHRUN1 = 0b and VICHRUN1 = 0b and HSBUSY = 0b)?

Figure 9.5-15 Enter/Exit of ULPS (Clock Lane)



Note 1. Check all enabled data lanes selected by TXSETR.NUMLANEUSE

Note 2. (SQCHRUN0 = 0b and SQCHRUN1 = 0b and VICHRUN1 = 0b and HSBUSY = 0b and LPBUSY = 0b)?

Figure 9.5-16 Enter/Exit of ULPS (Clock and Data Lane)

9.5.3.7 Error Handling

9.5.3.7.1 Error Detection by Peripheral

Peripheral will notify you of the detected error by “Acknowledge and Error Report” packet.

When the DSI-Tx Module receives the “Acknowledge and Error Report” packet, it displays the latest result in the AKEPLATIR register and the accumulated result in the AKEPACMSR register.

When the user receives the “Acknowledge and Error Report” packet, review and change various settings as necessary.

9.5.3.7.2 Error Detection in D-PHY

If the D-PHY detects an error, the DSI-Tx Module sets the corresponding register flag.

The errors supported by the DSI-Tx Module are as follows. For details on the error, refer to Annex A Logical PHY-Protocol Interface Description of MIPI Alliance Specification for D-PHY Version 1.2.

- ErrEsc
- ErrSyncEsc
- ErrControl
- ErrContentionLP0
- ErrContentionLP1

If the user detects this error, perform Software Reset (**9.5.3.2.1 Software Reset**) and Reset Trigger Transmission (**9.5.3.4.4 Reset Trigger Transmission**) if necessary.

When an ErrContentionLP0 / ErrContentionLP1 error occurs, be sure to execute Software Reset (**9.5.3.2.1 Software Reset**). At that time, write 1b to RSTCR.FCETXSTP, wait for the time when Contention is resolved (equivalent to LRX-H_TO), and then write 0b to RSTCR.FCETXSTP. Also, if necessary, perform Reset Trigger Transmission (**9.5.3.4.4 Reset Trigger Transmission**).

For more information on returning from Contention, see Section 7.2, Contention Detection and Recovery in the MIPI Alliance Specification for Display Serial Interface 1.3.1.

9.5.3.7.3 Timeout Error

The DSI-Tx Module has the timers defined in Section 7.2.2, Contention Recovery Using Timers of the DSI standard and the DSI-Tx Module original timers.

- Timers defined in DSI Specification
  - HS TX Timeout Error (HTX_TO)
  - LP-RX Host Processor Timeout Error (LRX-H_TO)
  - Turnaround Acknowledge Timeout Error (TA_TO)
- Original Timers
  - Rx Response Timeout Error

(1) HS TX Timeout (HTX_TO) Error

If this error occurs, perform Software Reset (**9.5.3.2.1 Software Reset**). In addition, perform Reset Trigger Transmission (**9.5.3.4.4 Reset Trigger Transmission**) if necessary.

If this error is detected, the setting value of HSTXTOSETR.HTXTO may be too small. When multiple channels are operating at the same time, multiple HS packets may be concatenated and transmitted continuously by HS. Set HSTXTOSETR.HTXTO to a value larger than the HS transmission period.

(2) LP-RX Host Processor Timeout (LRX-H_TO) Error

If this error occurs, perform Software Reset (**9.5.3.2.1 Software Reset**). In addition, perform Reset Trigger Transmission (**9.5.3.4.4 Reset Trigger Transmission**) if necessary.

If this error is detected, the setting value of LRXHTOSETR. LRXHTO may be too small.

There is a possibility of receiving multiple packets such as response packet and “Acknowledge and Error Report” packet in one reception.

Measure the time from the completion of the Bus Turnaround sequence from Host to Peripheral (LP-11 drive detection by Peripheral) to the completion of Bus Turnaround sequence from Peripheral to Host (LP-11 drive start by Host).

Check the Peripheral specifications, and set the value of DSI_LPCLK cycle count (rounded up) + X or more corresponding to the above time in the LRXHTOSETR.LRXHTO register.

(3) Turnaround Acknowledge Timeout (TA_TO) Error

If this error occurs, perform Software Reset (**9.5.3.2.1 Software Reset**) and Reset Trigger Transmission (**9.5.3.4.4 Reset Trigger Transmission**) if necessary.

If this error is detected, the setting value of TATOSETR.TATO may be too small.

Measure the time from the start of the Peripheral Bus Turnaround sequence (LP-10 drive by the Host) to the completion (LP-11 drive detection by the Peripheral) from the Host.

Check the Peripheral specifications and set the value of DSI_LPCLK cycle count (rounded up) + X or more corresponding to the above time in the TATOSETR.TATO register. This error also occurs if the Peripheral does not support bidirectional communication.

(4) Rx Response Timeout Error

Refer to the Peripheral Response Timeout Error (**(a) Peripheral Response Timeout Error**) for more information.

9.5.3.7.4 Rx Related Error in Sequence Operation

(1) Rx Packet Related Error in Sequence Operation

The following describes the received packet error detected by the DSI-Tx Module.

Detects up to one error per packet. (However, ECC 1bit Error may occur at the same time as other errors.)

Errors are determined in the following order:

Peripheral Response Timeout Error, Malformed Packet Error, ECC Error, Unexpected Packet Error, WC Error, CRC Error, Nothing Return Error

(a) Peripheral Response Timeout Error

RXSR.PRESPTOERR is set to 1b if the BTA does not start receiving packets or triggers within the time set in the register after the bus right transitions from Host to Peripheral.

If this error is detected, packets and triggers will not be received until the bus right transitions from Peripheral to Host.

[Timeout time setting register]

- Peripheral Response Timeout value for BTA-only is set by PRESPTOBTASETR.
- Peripheral Response Timeout value for LP Read with BTA is set by PRESPTOLPSETR.PRESPTOLPR.
- Peripheral Response Timeout value for LP Write with BTA is set by PRESPTOLPSETR.PRESPTOLPW.
- Peripheral Response Timeout value for HS Read with BTA is set by PRESPTOHSSETR.PRESPTOHSR.
- Peripheral Response Timeout value for HS Write with BTA is set by PRESPTOHSSETR.PRESPTOHSW.

If this error is detected, the register setting value may be too small, Measure the time from the completion of the Bus Turnaround sequence of Peripheral from the Host (LP-11 drive detection by Peripheral) to the completion of reception of the first byte of the packet or the completion of trigger reception. After checking the Peripheral specifications, set the value of DSI_LPCLK cycle number (rounded up) + X or more corresponding to the above time.

If this error occurs, perform the necessary processing such as resending the previous command.

(b) Malformed Packet Error

If a packet less than 4 bytes is received, RXSR.MLFERR is set to 1b. Since there is an error in the received data, perform necessary processing such as resending the previous command.

(c) ECC Error

ECC generation is required for DSI v1.3.1 standard compliant Peripherals. However, since ECC generation is an option for “earlier revision of DSI Peripherals”, the DSI-Tx Module can select whether to enable or disable ECC checks using DSISETR.ECCEN.

When connecting to a Peripheral that does not support ECC generation, disable ECC checking by setting DSISETR.ECCEN to 0b.

(d) ECC 1-bit Error

If an ECC 1bit error is detected, RXSR.ECCERR1B is set to 1b and 1bit error correction is performed.

The DSI-Tx Module automatically corrects the error and restores the normal packet header, so the subsequent packet reception processing continues.

(e) ECC Multi-bit Error

If an ECC multi-bit error is detected, RXSR.ECCERR is set to 1b and the received packet is discarded. If this error occurs, perform the necessary processing such as resending the previous command.

(f) Unexpected Packet Error

If an Unexpected Packet is received, RXSR.UEXPKTERR is set to 1b.

When this error occurs, the packet header is stored in the RXRSSxR register, and the subsequent reception processing is stopped until the bus right transitions from Peripheral to Host.

The DSI-Tx Module determines the following packets as Unexpected Packets.

- Received packets in which Data Type field of the packet header is “Reserved” as described in Table 23 Data Types for Peripheral-Sourced Packets of the DSI standard
- Received packets other than “Acknowledge and Error Report” packet and “EoTp” in BTA after sending non-read packet
- Received packets other than “Acknowledge and Error Report” packet and “EoTp” when the BTA is BTA only
- BTA after sending read packet receives response packet and then receives response packet again

(g) WC Error

RXSR.WCERR is set to 1b if a long packet with payload data shorter than the WC value is received.

When this error occurs, there is an error in the payload data length, but the payload data is not discarded due to this error, and the packet header is stored in the RXRSSxR register.

Since there is an error in the received data, perform necessary processing such as resending the previous command.

(h) CRC Error

To connect to Peripherals that do not support CRC generation, the DSI-Tx Module can enable or disable CRC checking for each Virtual Channel using DSISETR.CRCEN[3:0]. When connecting to a Peripheral that does not support CRC generation, disable the CRC check by setting the corresponding bit in DSISETR.CRCEN to 0b.

When this error occurs, there is an error in the payload data, but the payload data is not discarded due to this error, and the packet header is stored in the RXRSSxR register.

If this error occurs, perform the necessary processing such as resending the previous command.

(i) Nothing Return Error

RXSR.NORETERR is set to 1b if the BTA transitions the bus right from Host to Peripheral and then does not receive packets or triggers and the bus right transitions from Peripheral to Host.

If this error occurs, perform the necessary processing such as resending the command. If the error recurs, perform Software Reset (**9.5.3.2.1 Software Reset**) and Reset Trigger Transmission (**9.5.3.4.4 Reset Trigger Transmission**) as necessary.

(2) Rx Long Packet Data Payload Related Error in Sequence Operation

(a) Maximum Return Packet Size Error

RXSR.MAXRPSZERR is set to 1b if the WC value of the received long packet is greater than the DSISETR.MRPSZ setting.

When this error occurs, the packet header is stored in the RXRSSxR register and the payload data is discarded.

Although the payload data is discarded, errors related to payload data reception (WC Error, CRC Error) are detected.

If this error occurs, set DSISETR.MRPSZ to a value greater than or equal to the value set by the Set Max Return Packet Size command.

If you still get this error despite the correct settings, consider setting DSISETR.MRPSZ to a larger value because Peripheral is returning a large packet in violation of the DSI standard.

(b) Internal Bus Error

If Internal Bus Write detects an error, RXSR.IBERR is set to 1b.

Review the settings of SQCHxDSCyDR.LADDR and DSISETR.MRPSZ.

(c) Rx Buffer Overflow Error

If the following Rx Buffer Overflow Error is detected, RXSR.RXOVFERR is set to 1b.

In the case of SQCHxDSCyBR.DTSEL = 00b Packet Payload Data register Mode:

- The Rx Payload Data is over 16 Byte length

In the case of SQCHxDSCyBR.DTSEL = 01b Long Packet data use memory space Mode:

- RAM Overflow for the LP Rx is detected

If this error occurs, execute the following.

In the case of Data register Mode:

- Change to use memory space Mode
- Change peripheral device’s Maximum Return Packet Size as not over 16 bytes

In the case of memory space Mode:

- Change peripheral device’s Maximum Return Packet Size smaller
- Make Internal Bus faster

9.5.3.7.5 Tx Packet Related Error in Sequence Operation

(1) Internal Bus Error

If Internal Bus Read detects an error, SQCHxSR.TXIBERR is set to 1b. Review the settings of SQCHxDSCyAR.DATA1 / 0b and SQCHxDSCyDR.LADDR.

(2) Packet too Big Error

SQCH1SR.PKTBIGERR is set to 1b if the Sequence Operation packet is of a size that cannot be sent within the BLLP period of the first horizontal line of the Video-Input Operation.

If this error occurs, the Sequence Operation packet will not be sent until the Video-Input Operation is completed.

9.5.3.7.6 Error in Video-Input Operation

(1) Video-Input Buffer Overflow Error

VICH1SR.VBUFOVF is set to 1b if Overflow of Video-Input Buffer is detected.

If this error occurs, perform Software Reset (**9.5.3.2.1 Software Reset**). Also, take necessary measures such as increasing the frequency of TxWordClkHS (hsclk) and decreasing the VICH1SET1R.DLY value. In addition, perform Reset Trigger Transmission (**9.5.3.4.4 Reset Trigger Transmission**) if necessary.

(2) Video-Input Buffer Underflow Error

If the Video-Input Buffer Underflow is detected, VICH1SR.VBUFUDF is set to 1b.

If this error occurs, perform Software Reset (**9.5.3.2.1 Software Reset**). Also, take necessary measures such as lowering the frequency of TxWordClkHS (hsclk) and increasing the VICH1SET1R.DLY value. In addition, perform Reset Trigger Transmission (**9.5.3.4.4 Reset Trigger Transmission**) if necessary.

(3) Timing Error

VICH1SR.TIMERR is set to 1b if no video packets (including Sync Event packets) were sent during the Video-Input Operation at the intended timing.

If this error occurs, perform Software Reset (**9.5.3.2.1 Software Reset**). Also, take necessary measures such as reviewing the setting value of LPTRNSTSETR.GOLPBKT. In addition, perform Reset Trigger Transmission (**9.5.3.4.4 Reset Trigger Transmission**) if necessary.

9.5.4 Interrupts

9.5.4.1 Interrupt List

Interrupts can be divided into the following categories:

Table 9.5-7 Interrupt Signals (1/2)

Interrupt Signal	Category	Interrupt Condition	Interrupt Source Register
dsi_int_sq0	Sequence operation channel 0 interrupt	Receive Acknowledge and Error Report Packet in Sequence Operation BTA	SQCH0SR.RXAKE
		Receive Correctable Error in Sequence Operation BTA	SQCH0SR.RXCORERR
		Receive Packet Data Error in Sequence Operation BTA	SQCH0SR.RXPKTDFAIL
		Receive Fail Error in Sequence Operation BTA	SQCH0SR.RXFAIL
		Receive Fatal Error in Sequence Operation BTA	SQCH0SR.RXFATALERR
		Transmit Internal Bus Error in Sequence Operation	SQCH0SR.TXIBERR
		Descriptor Finished by Setting in Sequence Operation	SQCH0SR.ADESFIN
		Descriptor's All Action Finished in Sequence Operation	SQCH0SR.AACTFIN
dsi_int_sq1	Sequence operation channel 1 interrupt	Receive Acknowledge and Error Report Packet in Sequence Operation BTA	SQCH1SR.RXAKE
		Receive Correctable Error in Sequence Operation BTA	SQCH1SR.RXCORERR
		Receive Packet Data Error in Sequence Operation BTA	SQCH1SR.RXPKTDFAIL
		Receive Fail Error in Sequence Operation BTA	SQCH1SR.RXFAIL
		Receive Fatal Error in Sequence Operation BTA	SQCH1SR.RXFATALERR
		Transmit Internal Bus Error in Sequence Operation	SQCH1SR.TXIBERR
		Transmit Packet is too big Error in Sequence Operation	SQCH1SR.PKTBIGERR
		Descriptor Finished by Setting in Sequence Operation	SQCH1SR.ADESFIN
		Descriptor's All Action Finished in Sequence Operation	SQCH1SR.AACTFIN
dsi_int_vin1	Video-Input operation channel 1 interrupt	Transmit Buffer Overflow in Video-Input Operation	VICH1SR.VBUFOVF
		Transmit Buffer Underflow in Video-Input Operation	VICH1SR.VBUFUDF
		Transmit Timing Error in Video-Input Operation	VICH1SR.TIMERR
		Video-Input Signal Accept Ready	VICH1SR.VIRDY
		Video-Input Operation Stopped	VICH1SR.STOP
		Video-Input Operation Started	VICH1SR.START

Table 9.5-7 Interrupt Signals (2/2)

Interrupt Signal	Category	Interrupt Condition	Interrupt Source Register
dsi_int_rcv	DSI Packet Receive interrupt	Receive Acknowledge and Error Report Packet	RXSR.RXAKE
		Single-bit ECC Error	RXSR.ECCERR1B
		Maximum Return Packet Size Error	RXSR.MAXRPSZERR
		No Return Error	RXSR.NORETERR
		Peripheral Response timeout	RXSR.PRESPTOERR
		Receive Buffer overflow error	RXSR.RXOVFERR
		Internal Bus Error detect on data save	RXSR.IBERR
		CRC Error	RXSR.CRCERR
		Word Count Error	RXSR.WCERR
		Unexpected Packet Error	RXSR.UEXPKTERR
		Multi-bit ECC Error	RXSR.ECCERR
		Malformed Error	RXSR.MLFERR
		External Tearing Effect	RXSR.EXTTEDET
		Receive ACK trigger	RXSR.RXACK
		Receive Tearing Effect trigger	RXSR.RXTE
		Receive Reset trigger	RXSR.RXRTRG
		Receive Unknown-5 trigger	RXSR.RXUK5TRG
		Receive EoTp	RXSR.RXEOTP
		Receive Response packet	RXSR.RXRESP
		Turnaround Acknowledge Timeout	RXSR.TATO
		LP-RX Host Processor Timeout	RXSR.LRXHTO
		BTA request End	RXSR.BTAREQEND
dsi_int_ferr	DSI Fatal Error interrupt	Contention Error LP1	FERRSR.ERRCLP1
		Contention Error LP0	FERRSR.ERRCLP0
		Control Error	FERRSR.ERRCTRL
		Sync Escape Error	FERRSR.ERRSYNESC
		Error Escape	FERRSR.ERRESC
		Turnaround Acknowledge Timeout	FERRSR.TATO
		LP-RX Host Processor Timeout	FERRSR.LRXHTO
		HS TX Timeout	FERRSR.HTXTO
dsi_int_ppi	DSI D-PHY PPI interrupt	Data Lanes return from ULPS	PLSR.DLFROMULPS
		Data Lanes go to ULPS	PLSR.DLTOULPS
		Clock Lane change HS to LP	PLSR.CLHS2LP
		Clock Lane change LP to HS	PLSR.CLLP2HS
		Clock Lanes return from ULPS	PLSR.CLFROMULPS
		Clock Lanes go to ULPS	PLSR.CLTOULPS
		Data Lane 0 change TX to RX	PLSR.DL0TX2RX
		Data Lane 0 change RX to TX	PLSR.DL0RX2TX

The interrupt signal consists of combination logic. The interrupt signal from a Status Register is enabled by the signal related Interrupt Enable Register. These interrupt signals are bound by OR logic.

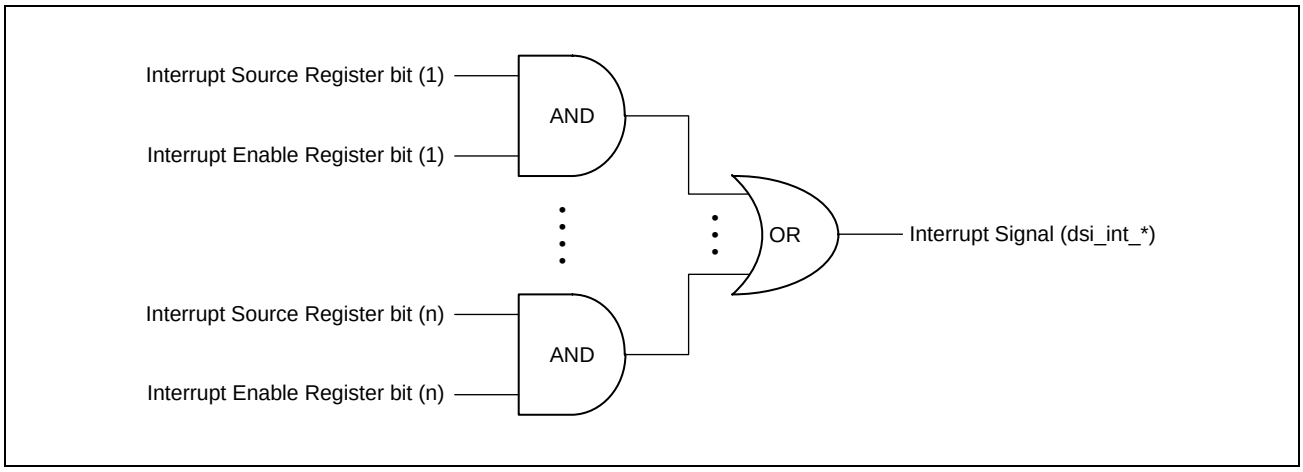


Figure 9.5-17 Interrupt Signal Output

9.5.5 Usage Notes

9.5.5.1 Restriction and Notes

In this section, the restriction and notes on the DSI-Tx Module are described.

Table 9.5-8 Note and Restriction List

Items	Restriction and Notes
TxWordClkHS	Each Lanes TxWordClkHS must be synchronized. The DSI-Tx Module uses Data Lane0 TxWordClkHS as hsclk.
Unused input signals	All unused input signals must be input 0, except following signals. Following signals must be input 1, when they are not used. • ppi_dl1_ulpsactivenot • ppi_dl2_ulpsactivenot • ppi_dl3_ulpsactivenot
TxReadyHS assert	D-PHY must be asserted TxReadyHS within ($2^{RELDSI2TX_LNK_RAM_WORD_DEPTH}$ – 2) hsclk cycles after TxRequestHS asserted.

SECTION 9 IMAGE

9.6 H.265/H.264 Multi Codec (VCD)

This section describes the functions of the H.265/H.264 multi codec (VCD).

9.6.1 Functional Overview

The VCD is a video codec that supports H.264 and H.265. It has the features shown below.

- H.265/HEVC MP (main profile) supported
  - H.265/HEVC main profile at level 5
- H.264/AVC constrained baseline/main/high profile supported
  - High profile at level 4.2,
  - Main profile at level 4.2,
  - Baseline profile at level 4.2
- H.265 encoding and decoding performance
  - 3840 × 2160 encoding, 3840 × 2160 decoding
- H.264 encoding and decoding performance
  - 1920 × 1080 p encoding, 1920 × 1080 p decoding
- Equipped with lossless compression and reference image buffer to reduce memory bandwidth

9.6.2 Connection Configuration

Figure 9.6-1 shows the connection configuration for the H.265/H.264 multi codec (VCD).

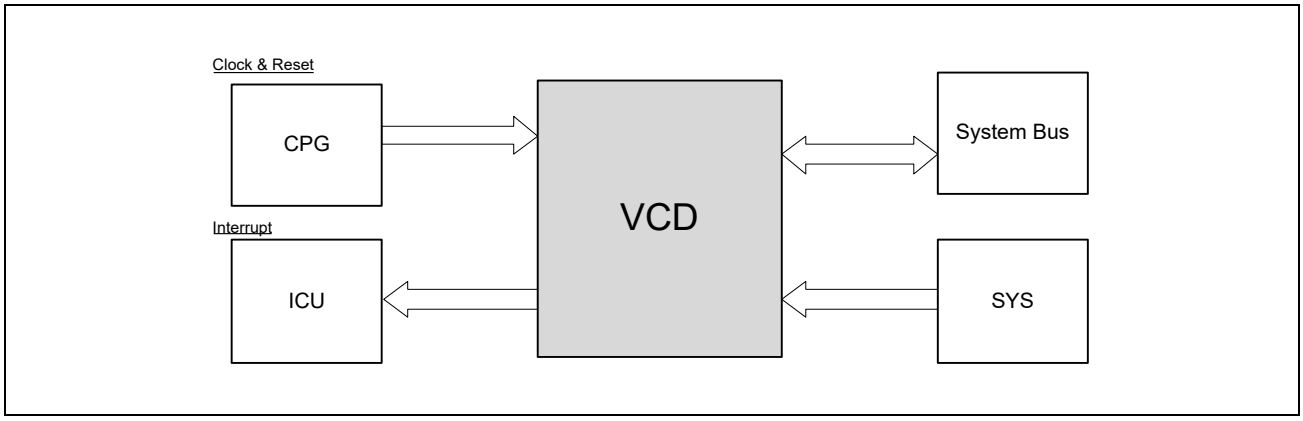


Figure 9.6-1 Configuration Diagram of VCD Connection

9.6.3 Operation Procedure

Use the driver provided by Renesas to operate this unit.

SECTION 9 IMAGE

9.7 3D Graphics Engine (GE3D)

9.7.1 Overview

This LSI includes the Arm Mali-G52 GPU as a 3D Graphics Engine (GE3D). The Arm Mali-G52 GPU is a graphics acceleration platform that is based on open standards. It supports 2D graphics, 3D graphics, and General Purpose computing on GPU (GPGPU).

9.7.1.1 Features

The GE3D has the following features.

- One double-pixel shader core and two execution engine.
- 64 Kbytes L2 cache
- A programmable architecture.
- An API feature set with support for shader-based and fixed-function graphics APIs.
- Anti-aliasing capabilities.
- An effective core for General Purpose computing on GPU (GPGPU) applications.
- High memory bandwidth and low power consumption for 3D graphics content.
- Latency tolerance.
- Compressed texture formats.
- Coherency aware interconnects for system memory and resource sharing.

The GE3D and software support compute and graphics API standards.

— The GE3D supports these compute API standards:

OpenCL 2.0 Full Profile.

— The GE3D supports these graphics API standards:

OpenGL ES 1.1, 2.0, 3.0, 3.1 and 3.2.

9.7.1.2 Block Diagram

The GE3D contains the shader processor core and associated blocks, including the tiler, the memory-management unit, and the level 2 cache memory subsystem.

Figure 9.7-1 shows a block diagram.

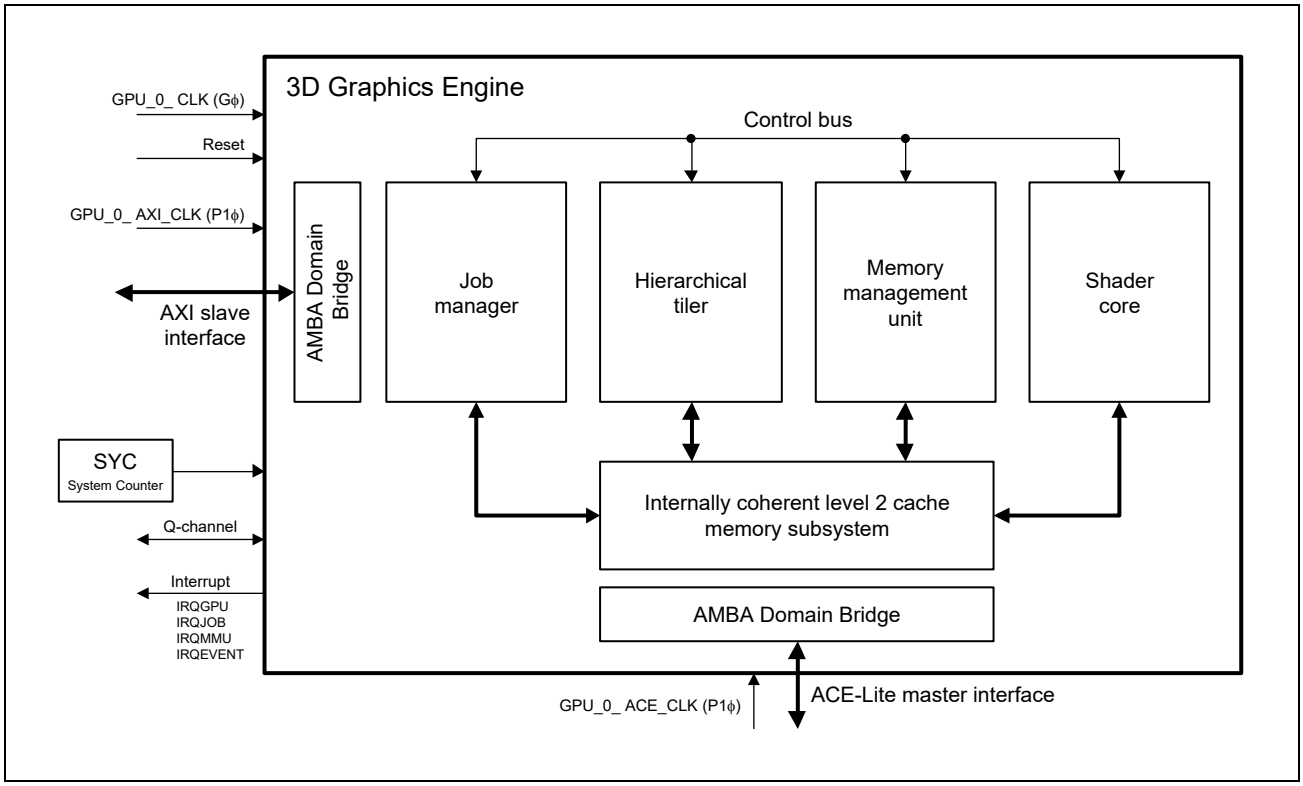


Figure 9.7-1 Block Diagram

The GE3D uses a centralized hardware resource manager, called the job manager, that controls the internal GPU functions.

The application software runs on the application processor. When it requires graphics to be displayed on screen, it uses the graphics device driver to send a graphics job to the GE3D. In the GE3D, the job manager receives the job, interprets it, and then sends a series of graphics tasks to the internal functional units.

The job manager issues and sends commands to specific functional blocks in the GE3D.

- The shader core is the main processing engines of the GE3D. The shader core carries out all the rendering and computation operations.
- The tiler creates lists of all the objects in a scene, so that the shader core processes the objects efficiently.
- The memory management unit performs virtual address to physical address translation of external bus interface accesses.
- The Level 2 cache memory subsystem provides caching for all internal master blocks. It also supports cache coherency for caches in the shader cores. The L2 cache sits between the internal master blocks and the external interconnect. All transactions go through the cache. If the cache cannot service an internal transaction, it performs an external transaction to read or write data.

9.7.1.3 External Pins

In 3D Graphics Engine, there are no external pins.

9.7.1.4 Register Configuration

Base address of registers is allocated to 0_1485_0000h. The address range is 64 KB (0_1485_0000h - 0_1485_FFFFh).

9.7.1.5 Usage Note

Renesas does not provide technical support for users to program GE3D registers directly. Renesas provides users with drivers to program GE3D. GE3D supports OpenGL ES as a graphics API. When drawing graphics, control according to this API.

9.7.1.6 Starting SYC

When using GE3D, it is necessary to supply the clock to the SYC (system counter) and release the reset to keep it running.

SECTION 9 IMAGE

9.9 Video Signal Processor (VSP)

VSP consist of VSPI and FCPVI modules. Refer the detail for **9.9.1 VSPI** and **9.9.2 FCPVI**.

The VSPI supports image processing by memory to memory such as super resolution, sharpness, up/down scaling, look-up tables, histogram and rotation/flipping.

The Clock Divider divides clock from CPG and generates clk_m, clk_a, and clk_p clocks which are used by VSPI and FCPVI.

The ADB400 and APB Bridges are used for bridge for clock domain crossing between CPG generated clock and Clock Divider generated clocks.

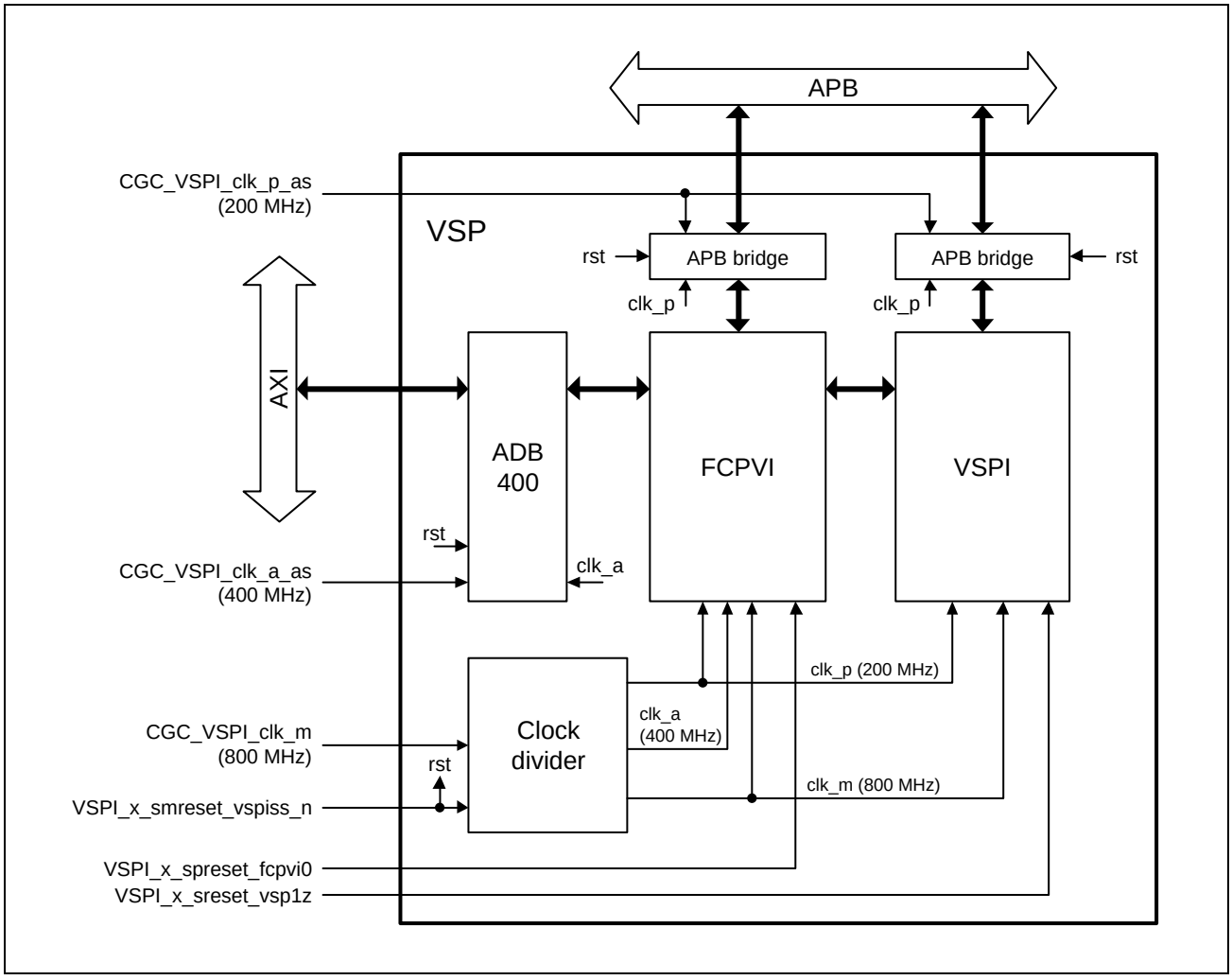


Figure 9.9-1 VSP Modules Configuration

9.9.1 VSPI

The VSPI supports image processing by memory to memory such as super resolution, sharpness, up/down scaling, look-up tables, histogram and rotation/flipping.

9.9.1.1 Features

- Supports various data formats and conversion
  - Supports YCbCr444/422/420, RGB, α RGB, α plane.
  - Color space conversion and changes to the number of colors by dithering
  - Color keying
  - Supports combination between pixel alpha and global alpha.
  - Supports generating pre multiplied alpha.
- Video processing
  - Supporting 3840 pixels × 2160 lines
  - Up and down scaling with arbitrary scaling ratio
  - Super resolution processing
  - Sharpness filter
  - 1D look up table
  - 3D look up table
  - Supports Hue, brightness, and saturation format for look up table or/and histogram.
  - 1D histogram
  - 2D histogram
  - Image rotation with unit of 90-degree, vertical and horizontal flipping
- Performance of VSPI*1
  - 500 Mpix/s process rate (processing rate*2) × 1 set

Note 1. This performance cannot be realized in some condition. Detail performance is shown in **9.9.1.5.5 (1) Performance of VSPI**.

Note 2. When scale-down is applied to input image, processing rate is input rate. When scale-up is applied to input image, processing rate is output rate. When no-scaling is applied, processing rate is input rate.

9.9.1.1.1 Block diagram of VSPI

Figure 9.9-2 shows the configuration of VSPI module.

All the registers and functions which doesn't appears in the each VSPI unit's block diagram should be treated as un-implemented functions in each VSPI unit. Do not set any registers related to the un-implemented function without any statement.

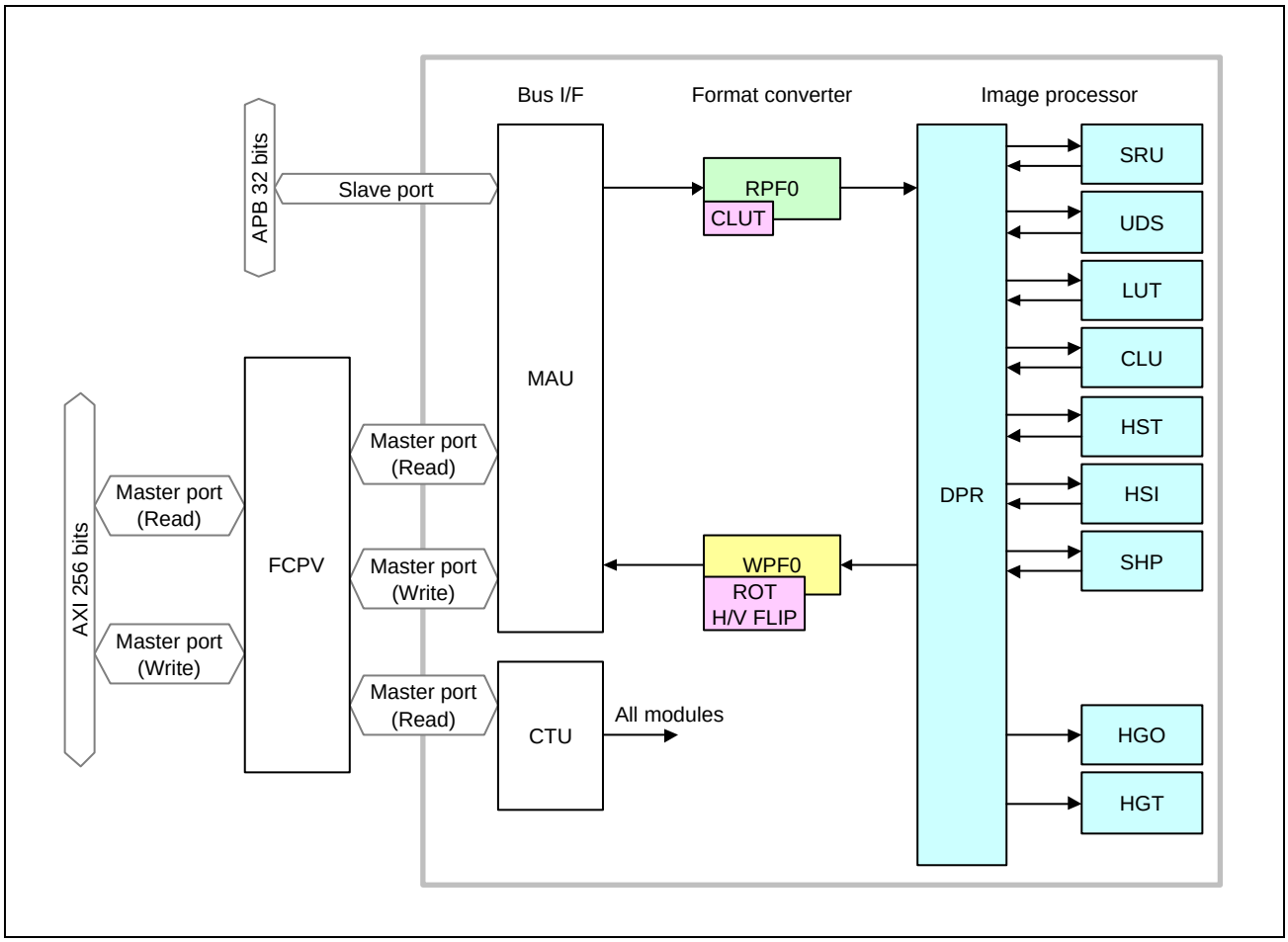


Figure 9.9-2 VSPI Module Configuration

The following descriptions give an overview of the block function shown in **Figure 9.9-2**.

The operation of these functions is determined by the register setting explained in **9.9.1.3 Register Description.** Refer to **9.9.1.3 Register Description** for the relationship between the register setting and the functional behavior.

9.9.1.1.2 Memory Access Unit (MAU)

The VSPI applies processing to the image data stored in the external memory and writes the resultant data back to the external memory. The data transfer between the external memory and VSPI necessary for this operation is done by the MAU, which works as the bus master, according to the register settings. The MAU executes this data transfer between the external memory and VSPI.

9.9.1.1.3 Command Transfer Unit (CTU)

The VSPI can directly read register parameters for image processing by display lists stored in external memory. The CTU module is a bus interface and controls display lists when the CTU reads display lists as a bus master.

9.9.1.1.4 Read Pixel Formatter (RPF)

The RPF reads image data from the external memory through the MAU, unpacks data according to the specified format, converts the color space, converts the number of colors, executes color keying, ROP operation, Multiply-alpha and OSD processing, and outputs the resultant data to the DPR. The RPF has an input format unpacking unit, an index color expanding unit (OSD-CLUT), a 1-bit mask generator, a raster operation unit (ROP unit),a color keying unit, a color space converter and multiply-alpha. **Figure 9.9-3** shows the processing flow in the RPF.

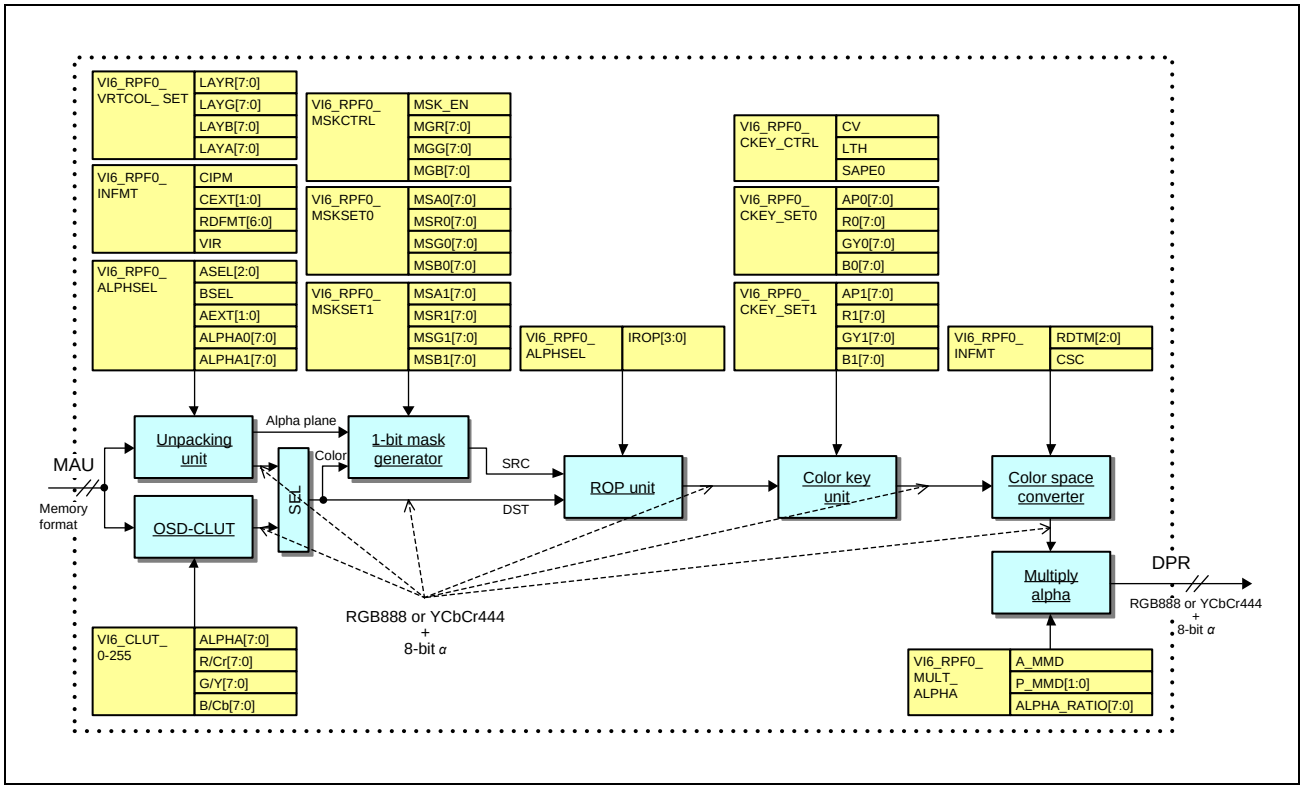


Figure 9.9-3 RPF Processing Flow

The input format unpacking unit and index color expanding unit (OSD-CLUT) expand the image data input from the MAU into the image format for internal processing (YCbCr444 or RGB888), and the 1-bit mask generator generates a 1-bit image mask from the image data expanded through the unpacking unit and OSD-CLUT. Alternately, a 1-bit image mask can be generated from the alpha plane that is different from the picture plane read through the MAU.

The raster operation unit (ROP unit) executes raster operation between the data from the 1-bit mask generator and the image data expanded from the input format, and the color keying unit applies color replacement and specifies the transparent color for the image data input from the ROP unit. The color space converter converts the color space (RGB-YCbCr) of the image data input from the color keying unit as necessary. The multiply-alpha unit multiplies pixel-alpha by specified alpha or/and multiplies image data by pixel-alpha or/and specified alpha.

The implementation of OSD-CLUT depends on the configuration of this LSI.

9.9.1.1.5 Data Path Router (DPR)

The DPR controls the data paths among RPFs, function modules, and WPFs. The DPR selects one of the images input from RPF, outputs it to a function module (SRU, UDS, LUT, CLU, HST, HSI, HGO, HGT, or SHP), and WPF as the destination where the image data processed in the function module will be output. Before output to the WPF, the output from each function module can be input to another function module, which enables multiple image processing functions to be executed continuously without involving the external memory.

9.9.1.1.6 Super Resolution Unit (SRU)

The SRU is a module connected to the DPR, which executes the super resolution processing.

9.9.1.1.7 Up Down Scalar (UDS)

The UDS is a module connected to the DPR, which upscales or downscales the image size. It can also upscale or downscale the alpha value.

9.9.1.1.8 Look Up Table (LUT)

This is a 1D-LUT that converts each of three color components by using a lookup table. The LUT is connected to the DPR and can be used for gamma correction, negative-positive conversion, posterization, and binarization through desired tone curve settings.

9.9.1.1.9 Write Pixel Formatter (WPF)

The WPF is an output module that receives 32 bits image data (YCbCr444 or RGB888 + 8bit-α) from the DPR, converts the color space, number of colors, and format of the data, and outputs the results of VSPI image processing to external memory through the MAU. The WPF is mainly configured from a color space converter and an output format converter (the packing unit).

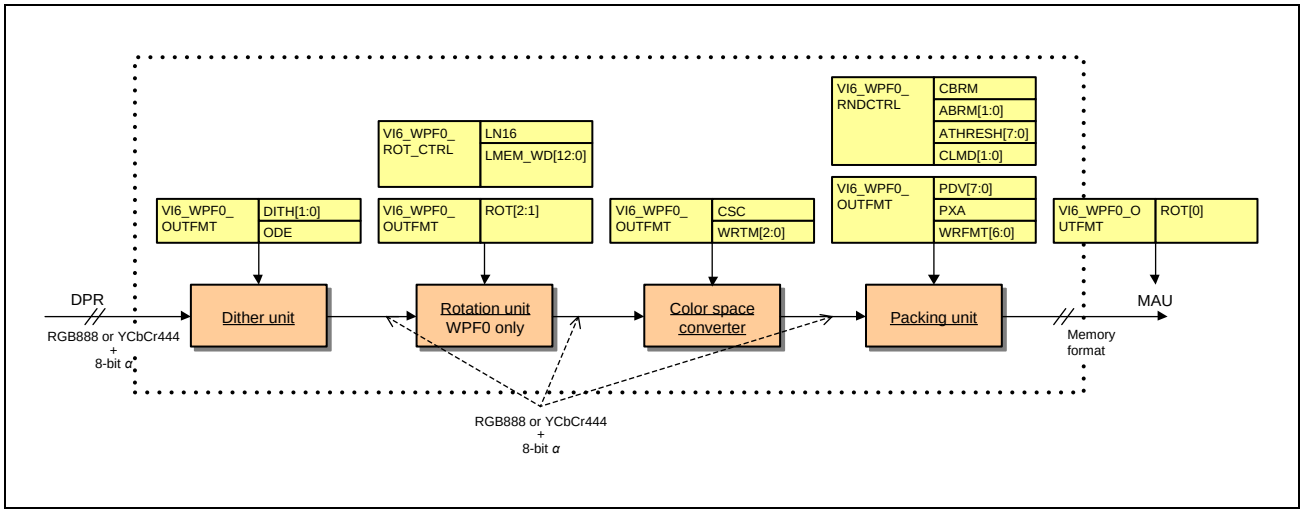


Figure 9.9-4 WPF Processing Flow

The color space converter converts the color space between RGB and YCbCr, and the packing unit converts the format into the picture plane storing format.

9.9.1.1.10 Cubic Look Up table (CLU)

This is a three-dimensional LUT (3D-LUT) that converts the input three-color-component data into desired three color components by using a lookup table. The 1D-LUT can only control each of three color components separately, but the 3D-LUT can convert a specified color into a different color. The CLU is connected to the DPR and can be used for specific color adjustment, such as correction to memorable color.

9.9.1.1.11 Hue Saturation value Transform (HST)

The HST is a module connected to the DPR, which converts the RGB color space into the HSV color space. The HSV color space can represent a color with the hue, saturation, and value (brightness), and it makes color control through the LUT easier than in the RGB color space.

9.9.1.1.12 Hue Saturation value Inverse transform (HSI)

The HSI is a module connected to the DPR, which converts the HSV color space into the RGB color space. It executes conversion in the direction opposite to that of the HST.

9.9.1.1.13 Histogram Generator-One dimension (HGO)

The HGO is a module connected to the DPR and generates the one-dimensional histogram for the dynamic gamma correction. There is no output port of the image data.

9.9.1.1.14 Histogram Generator-Two dimension (HGT)

The HGT is a module connected to the DPR and generates the two-dimensional histogram for the dynamic color correction. There is no output port of the image data.

9.9.1.1.15 SHarPness (SHP)

The SHP is a module connected to the DPR, which sharpen or un-sharpen the image.

9.9.1.1.16 Detail Function

Table 9.9-1 Detail Function of VSPI (1/4)

• Image Data Transfer Function			
Bus interface	Protocol		AXI 256 bits through FCPVI
	Data alignment	Conversion method	Byte, Word, LW, or LLW data swapping
		Channel	Data alignment can be specified separately for each input/output channel
Image memory	Input	Address setting	1-byte units
	Output	Address setting	1-byte units
		Memory area	Images can be written to the same memory area where the master layer is stored. Note the restrictions shown in **Table 9.9-2**.
YCbCr memory storage format			Interleaved, planar, or semi-planar
• Display list/Extended Display List Transfer Function			
Bus interface	Protocol		AXI 256 bits through FCPVI
	Data alignment	Conversion method	Byte, Word, LW data swapping
		Channel	Data alignment can be specified separately for display list and each entry of extended display list
Data memory	Load	Address setting	8-byte units
	Store	—	Not available

Table 9.9-1 Detail Function of VSPI (2/4)

• Read Pixel Formatter (RPF)			
Number of channels			One channel (RPF0)
Image format	Input	RGB	RGB888, RGB565, RGB666, αRGB8888, αRGB4444, αRGB1555, α plane (8 bpp, 1 bpp)
		YCbCr	YCbCr4:4:4/YCbCr4:2:2/YCbCr4:2:0 α plane (8 bpp, 1 bpp)
		Maximum size	8190 × 8190 pixels The internal data path modules have separate restrictions on the maximum image size. For details, refer to **9.9.1.5.2 Input Image Size**.
		Minimum size	1 × 1 pixel The internal data path modules have separate restrictions on the minimum image size. For details, refer to **9.9.1.5.2 Input Image Size**.
		Size setting unit	• YCbCr420: 2-pixel units both horizontally and vertically. • YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. In other formats, the size can be set in 1-pixel units.
Color keying	Color replacement	Compared data	RGB888 or Y
		Replaced data	αRGB8888 or αYCbCr
		Comparison Mode	Matched color mode
	Transparent color	Compared data	RGB888 or Y
		Replaced data	α
		Comparison Mode	Matched color mode, Luma threshold mode
OSD/CLUT	Format	Memory storage format	8 bpp
		Image format	αRGB32 or αYCbCr32
Raster operation	ROP2 (within input channels)	Operator	16 types (OpenGL2.0 is supported)
		Sources of operation	ROP2 operation between the 1-bpp α plane and RGB/YCbCr data. Note that 1-bpp α is converted to αRGB888 or αYCbCr4:4:4.
Color space conversion	RGB to YCbCr	Conversion expression	• RGB (0, 255) to BT.601 (16, 235/240) • RGB (0, 255) to BT.709 (16, 235/240) • RGB (0, 255) to BT.601 (0, 255) • RGB (16, 235) to BT.709 (16, 235/240)
	YCbCr to RGB		• BT.601 (16, 235/240) to RGB (0, 255) • BT.709 (16, 235/240) to RGB (0, 255) • BT.601 (0, 255) to RGB (0, 255) • BT.709 (16, 235/240) to RGB (16, 235)
Changing number of colors	Reducing RGB color depth	Target format	RGB666, RGB565, RGB555, RGB444, or RGB332
	Increasing RGB color depth		Padded with 0. Copied from the most significant bits.
	YCbCr444 generation		CbCr copying or CbCr vertical copying and horizontal interpolation.
Multiply-alpha function	Fade-alpha		Available for straight pixel and pre-multiplied pixel.
	Generate pre-multiplied alpha		
Virtual display	Color format		αRGB8888 or αYCbCr4:4:4 single-color
	Display size		Same as the size of the input channel
α bit count conversion	Input	Bit increase	Padded with 0. Copied from the most significant bits.

Table 9.9-1 Detail Function of VSPI (3/4)

• Write Pixel Formatter (WPF)			
Number of channels			One channel (WPF0)
Image format	Output	RGB	RGB332, RGB444, RGB565, RGB666, RGB888, αRGB8666, αRGB8888, αRGB4444, αRGB1555
		YCbCr	YCbCr4:4:4, YCbCr4:2:2, YCbCr4:2:0
		Maximum size	8190 × 8190 pixels The internal data path modules have separate restrictions on the maximum image size. For details, refer to **9.9.1.5.3 Output Image Size**.
		Minimum size	1 × 1 pixel The internal data path modules have separate restrictions on the minimum image size. For details, refer to **9.9.1.5.3 Output Image Size**.
		Size setting unit	• YCbCr420: 2-pixel units both horizontally and vertically. • YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. In other formats, the size can be set in 1-pixel units.
Color space conversion	RGB to YCbCr	Conversion expression	• RGB (0, 255) to BT.601 (16, 235/240) • RGB (0, 255) to BT.709 (16, 235/240) • RGB (0, 255) to BT.601 (0, 255) • RGB (16, 235) to BT.709 (16, 235/240)
	YCbCr to RGB		• BT.601 (16, 235/240) to RGB (0, 255) • BT.709 (16, 235/240) to RGB (0, 255) • BT.601 (0, 255) to RGB (0, 255) • BT.709 (16, 235/240) to RGB (16, 235)
Changing number of colors	Output	Reducing RGB color depth	Dithering, lower-order bit truncation, or rounding
		YCbCr422/420	CbCr skipping or CbCr vertical skipping and horizontal skipping
α bit count conversion	Output	Bit reduction	Truncation, rounding, or comparison with threshold (for 1 bpp)
• Video Processing Functions			
Super resolution	Processing method		Processing within one frame
	Scaling ratio		×1 (same size) or ×2
	Arbitrary scaling ratio		Super resolution processing can be combined with the arbitrary ratio scaling function.
	α processing		Fixed α output
Scalar with Arbitrary scaling ratio	Scaling factor		1/16 < scaling factor ≤ 16
	Scaling ratio setting	Expression	1 / [Mant.Frac]
		Precision	Mant = 4 bits, Frac = 12 bits
	α processing		Supported (for 1/4 < scaling factor ≤16)
Sharpness processing	Method		Sharpness or blur is applied in horizontal direction
Rotation/flipping	Rotation function		90, 180 and 270 degree
	Flipping function		Vertical flipping, horizontal flipping
	Combination of rotation and flipping		Supported
	α processing		Supported
Simultaneous processing			All video processing functions and color adjustment functions can be processed simultaneously (the order of function execution is limited).

Table 9.9-1 Detail Function of VSPI (4/4)

• Color Adjustment Function		
1D-LUT	LUT configuration	Independent R/Y, G/Cb, and B/Cr. 256 entries each
3D-LUT	Number of grid points	17 × 17 × 17
HSV color space conversion	RGB to HSV	8 bits each for H/S/V
	HSV to RGB	8 bits each for H/S/V
1D Histogram	Number of bins	• 64 for each color component • 256 for Y component or Max-RGB
2D Histogram	Number of hue areas	6
	Number of bins	32 for each hue area

9.9.1.1.17 Restrictions

When the data output from the VSPI is written back to the same memory area where the input data for the VSPI has been read, the VSPI has the following restrictions.

1. The access order and format on the frame memory are the same between the input pixels and output pixels.
2. Specifying a larger output image size than the input image size, either vertically or horizontally or both vertically and horizontally, is prohibited.
3. When the YCbCr4:2:0 format is input, operation between color components is prohibited.

These restrictions are summarized in **Table 9.9-2**. Refer to the descriptions of the registers related to each restriction.

Table 9.9-2 Restrictions on Use when Output Data is Written Back to Input Data Area

No.	Restriction	Related Registers
Restriction 1	The RPF0 input format and the WPF0 output format should be the same.	VI6_RPF0_INFMT.RDFMT VI6_RPF0_INFMT.VIR
	The RPF0 source image storing address and the WPF0 destination address should be the same.	VI6_RPF0_SRCM_ADDR_*
	The RPF0 source picture memory stride and the WPF0 destination memory stride should be the same.	VI6_RPF0_SRCM_PSTRIDE VI6_RPF0_SRCM_ASTRIDE
	The RPF0 and WPF0 data swapping settings should be the same.	VI6_WPF0_DSWAP
	Super resolution processing with double scale-up by the SRU is prohibited (super resolution processing of the same size is allowed).	VI6_SRU_CTRL0
Restriction 2	Scale-up and color filling by UDS is prohibited.	VI6_UDS_SCALE VI6_UDS_CLIP_SIZE
	The value of scaling filter horizontal and vertical phase registers should be zero in case of using UDS.	VI6_UDS_HPHASE VI6_UDS_VPHASE
	The RPF0 basic read size and extended read size should be the same.	VI6_RPF0_SRC_BSIZE VI6_RPF0_SRC_ESIZE
	Color space conversion is prohibited in RPF0 and WPF0.	VI6_RPF0_INFMT.CSC VI6_WPF0_OUTFMT.CSC
Restriction 3*1	Use of the CLU is prohibited. Make sure that the CLU is not used in DPR routing.	VI6_DPR_CLU_ROUTE.RT
	NOP should be specified for IROP operation.	VI6_RPF0_ALPH_SEL.IROP
	Color keying is prohibited.	VI6_RPF0_CKEY_CTRL.CV

Note 1. When the input format is not YCbCr4:2:0, restriction 3 in not applied.

9.9.1.2 Register Configuration

9.9.1.2.1 Base Address

Below are the base addresses of VSPI has 32 Kbytes address space.

<VSPI0_base>: 00_1605_0000h (5605_0000h*1, 4605_0000h*2)

Note 1. CM33 address space (non-secure)
Note 2. CM33 address space (secure)

9.9.1.2.2 Register/Table Address

Table 9.9-3 shows the VSPI register configuration (represents in relative address).

Table 9.9-3 VSPI Register Configuration

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
VSPI Start Registers 0	VSPI0_VI6_CMD0	0000_0000h	0000h	32
Reserve	-	-	0004h to 000Fh	-
Clock Control Register 0	VSPI0_VI6_CLK_CTRL0	0000_0000h	0010h	32
Clock Control Register 1	VSPI0_VI6_CLK_CTRL1	0000_0000h	0014h	32
Dynamic Clock Stop Control Register	VSPI0_VI6_CLK_DCSWT	0000_0000h	0018h	32
Dynamic Clock Stop Disable Register 0	VSPI0_VI6_CLK_DCSM0	0000_0000h	001Ch	32
Dynamic Clock Stop Disable Register 1	VSPI0_VI6_CLK_DCSM1	0000_0000h	0020h	32
Reserve	-	-	0024h to 0027h	-
Software Reset Register	VSPI0_VI6_SRESET	0000_0000h	0028h	32
Module Reset Enable Register 0	VSPI0_VI6_MRESET_ENB0	0000_0000h	002Ch	32
Module Reset Enable Register 1	VSPI0_VI6_MRESET_ENB1	0000_0000h	0030h	32
Module Reset Issuing Register	VSPI0_VI6_MRESET	0000_0000h	0034h	32
Operating Status Register	VSPI0_VI6_STATUS	0000_0000h	0038h	32
Reserve	-	-	003Ch to 0047h	-
WPF0 Interrupt Enable Registers	VSPI0_VI6_WPF0_IRQ_ENB	0000_0000h	0048h	32
WPF0 Interrupt Status Registers	VSPI0_VI6_WPF0_IRQ_STA	0000_0000h	004Ch	32
Reserve	-	-	0050h to 00FFh	-
Display List Control Register	VSPI0_VI6_DL_CTRL	0000_0000h	0100h	32
Display List-0 Header Address Register	VSPI0_VI6_DL_HDR_ADDR0	0000_0000h	0104h	32
Reserve	-	-	0108h to 0113h	-
Display List-0 Data Swapping Register	VSPI0_VI6_DL_SWAP0	0000_0000h	0114h	32
Reserve	-	-	0118h to 011Bh	-
Extended Display List-0 Control Register	VSPI0_VI6_DL_EXT_CTRL0	0000_0000h	011Ch	32
Display List Body Size Register-0	VSPI0_VI6_DL_BODY_SIZE0	0000_0000h	0120h	32
Reserve	-	-	0124h to 012Fh	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Display List-0 Header Reference Address Register	VSPI0_VI6_DL_HDR_REF_ADDR 0	0000_0000h	0130h	32
Reserve	-	-	0134h to 02FFh	-
RPF0 Basic Read Size Registers	VSPI0_VI6_RPF0_SRC_BSIZE	0000_0000h	0300h	32
RPF0 Extended Read Size Registers	VSPI0_VI6_RPF0_SRC_ESIZE	0000_0000h	0304h	32
RPF0 Input Format Registers	VSPI0_VI6_RPF0_INFMT	0000_0000h	0308h	32
RPF0 Data Swapping Registers	VSPI0_VI6_RPF0_DSWAP	0000_0000h	030Ch	32
RPF0 Display Location Registers	VSPI0_VI6_RPF0_LOC	0000_0000h	0310h	32
RPF0 a Plane Selection Control Registers	VSPI0_VI6_RPF0_ALPH_SEL	0000_0000h	0314h	32
RPF0 Virtual Plane Color Information Registers	VSPI0_VI6_RPF0_VRTCOL_SET	0000_0000h	0318h	32
RPF0 Mask Control Registers	VSPI0_VI6_RPF0_MSKCTRL	0000_0000h	031Ch	32
RPF0 IROP-SRC Input Value Registers 0	VSPI0_VI6_RPF0_MSKSET0	0000_0000h	0320h	32
RPF0 IROP-SRC Input Value Registers 1	VSPI0_VI6_RPF0_MSKSET1	0000_0000h	0324h	32
RPF0 Color Keying Control Registers	VSPI0_VI6_RPF0_CKEY_CTRL	0000_0000h	0328h	32
RPF0 Color Keying Color Setting Registers-0	VSPI0_VI6_RPF0_CKEY_SET0	0000_0000h	032Ch	32
RPF0 Color Keying Color Setting Registers-1	VSPI0_VI6_RPF0_CKEY_SET1	0000_0000h	0330h	32
RPF0 Source Picture Memory Stride Setting Registers	VSPI0_VI6_RPF0_SRCM_PSTRI DE	0000_0000h	0334h	32
RPF0 Source α Memory Stride Setting Registers	VSPI0_VI6_RPF0_SRCM_ASTRI DE	0000_0000h	0338h	32
RPF0 Source Y/RGB Address Registers	VSPI0_VI6_RPF0_SRCM_ADDR_ Y	0000_0000h	033Ch	32
RPF0 Source Chroma Address Registers 0	VSPI0_VI6_RPF0_SRCM_ADDR_ C0	0000_0000h	0340h	32
RPF0 Source Chroma Address Registers 1	VSPI0_VI6_RPF0_SRCM_ADDR_ C1	0000_0000h	0344h	32
RPF0 Source α Address Registers	VSPI0_VI6_RPF0_SRCM_ADDR_ AI	0000_0000h	0348h	32
RPF0 Bus Access Control Registers	VSPI0_VI6_RPF0_BAC	0000_0000h	0350h	32
Reserve	-	-	0354h to 036Bh	-
RPF0 Multiple Alpha Control	VSPI0_VI6_RPF0_MULT_ALPH	0000_0000h	036Ch	32
Reserve	-	-	0370h to 0FFFh	-
WPF0-Source-RPF Registers	VSPI0_VI6_WPF0_SRCRPF	0000_0000h	1000h	32
WPF0 Horizontal Input Size Clipping Registers	VSPI0_VI6_WPF0_HSZCLIP	0000_0000h	1004h	32
WPF0 Vertical Input Size Clipping Registers	VSPI0_VI6_WPF0_VSZCLIP	0000_0000h	1008h	32
WPF0 Output Format Registers	VSPI0_VI6_WPF0_OUTFMT	0000_0000h	100Ch	32
WPF0 Data Swapping Registers	VSPI0_VI6_WPF0_DSWAP	0000_0000h	1010h	32
WPF0 Rounding Control Registers	VSPI0_VI6_WPF0_RNDCTRL	0000_0000h	1014h	32
WPF0 Rotation Control Registers	VSPI0_VI6_WPF0_ROT_CTRL	0000_0000h	1018h	32
WPF0 Destination Y Plane Memory Stride Registers	VSPI0_VI6_WPF0_DSTM_STRID E_Y	0000_0000h	101Ch	32
WPF0 Destination C Plane Memory Stride Registers	VSPI0_VI6_WPF0_DSTM_STRID E_C	0000_0000h	1020h	32
WPF0 Destination Y/RGB Address Registers	VSPI0_VI6_WPF0_DSTM_ADDR_ Y	0000_0000h	1024h	32
WPF0 Destination Chroma Address Registers 0	VSPI0_VI6_WPF0_DSTM_ADDR_ C0	0000_0000h	1028h	32
WPF0 Destination Chroma Address Registers 1	VSPI0_VI6_WPF0_DSTM_ADDR_ C1	0000_0000h	102Ch	32
Reserve	-	-	1130h to 1FFFh	-
RPF0 Routing Register	VSPI0_VI6_DPR_RPF0_ROUTE	0000_0000h	2000h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	2004h to 2010h	-
WPF0 Timing Control Register	VSPI0_VI6_DPR_WPF0_FPORCH	0000_0000h	2014h	32
Reserve	-	-	2018h to 2023h	-
SRU Routing Register	VSPI0_VI6_DPR_SRU_ROUTE	0000_0000h	2024h	32
UDS Routing Register	VSPI0_VI6_DPR_UDS_ROUTE	0000_0000h	2028h	32
Reserve	-	-	202Ch to 203Bh	-
LUT Routing Register	VSPI0_VI6_DPR_LUT_ROUTE	0000_0000h	203Ch	32
CLU Routing Register	VSPI0_VI6_DPR_CLU_ROUTE	0000_0000h	2040h	32
HST Routing Register	VSPI0_VI6_DPR_HST_ROUTE	0000_0000h	2044h	32
HSI Routing Register	VSPI0_VI6_DPR_HSI_ROUTE	0000_0000h	2048h	32
HGO Sampling Point Register	VSPI0_VI6_DPR_HGO_SMPPT	0000_0000h	2054h	32
HGT Sampling Point Register	VSPI0_VI6_DPR_HGT_SMPPT	0000_0000h	2058h	32
Reserve	-	-	205Ch to 205Fh	-
SHP Routing Register	VSPI0_VI6_DPR_SHP_ROUTE	0000_0000h	2060h	32
Reserve	-	-	2064h to 21FFh	-
Super Resolution Control Register 0	VSPI0_VI6_SRU_CTRL0	0000_0000h	2200h	32
Super Resolution Control Register 1	VSPI0_VI6_SRU_CTRL1	0000_0000h	2204h	32
Super Resolution Control Register 2	VSPI0_VI6_SRU_CTRL2	0000_0000h	2208h	32
Reserve	-	-	220Ch to 22FFh	-
Scaling Control Registers	VSPI0_VI6_UDS_CTRL	0000_0000h	2300h	32
Scaling Factor Registers	VSPI0_VI6_UDS_SCALE	0000_0000h	2304h	32
α Data Threshold Setting Registers	VSPI0_VI6_UDS_ALPTH	0000_0000h	2308h	32
α Data Replacing Value Setting Registers	VSPI0_VI6_UDS_ALPVAL	0000_0000h	230Ch	32
Passband Registers	VSPI0_VI6_UDS_PASS_BWIDTH	0000_0000h	2310h	32
Scaling Filter Horizontal Phase Registers	VSPI0_VI6_UDS_HPHASE	0000_0000h	2314h	32
2D IPC Setting Register	VSPI0_VI6_UDS_IPC	0000_0000h	2318h	32
UDS Horizontal Input Clipping Registers	VSPI0_VI6_UDS_HSZCLIP	0000_0000h	231Ch	32
Reserve	-	-	2320h to 2323h	-
UDS Output Size Clipping Registers	VSPI0_VI6_UDS_CLIP_SIZE	0000_0000h	2324h	32
Color Fill Register	VSPI0_VI6_UDS_FILL_COLOR	0000_0000h	2328h	32
Reserve	-	-	232Ch to 27FFh	-
LUT Control Register	VSPI0_VI6_LUT_CTRL	0000_0000h	2800h	32
Reserve	-	-	2804h to 28FFh	-
CLU Control Register	VSPI0_VI6_CLU_CTRL	0000_0000h	2900h	32
Reserve	-	-	2904h to 29FFh	-
HST Control Register	VSPI0_VI6_HST_CTRL	0000_0000h	2A00h	32
Reserve	-	-	2A04h to 2AFFh	-
HSI Control Register	VSPI0_VI6_HSI_CTRL	0000_0000h	2B00h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	2B04h to 2FFFh	-
HGO Detection Window Offset Register	VSPI0_VI6_HGO_OFFSET	0000_0000h	3000h	32
HGO Detection Window Size Register	VSPI0_VI6_HGO_SIZE	0000_0000h	3004h	32
HGO Mode Register	VSPI0_VI6_HGO_MODE	0000_0000h	3008h	32
HGO LB Detection Threshold Register	VSPI0_VI6_HGO_LB_TH	0000_0000h	300Ch	32
HGO Horizontal Position Register for LB Detection Zone-0	VSPI0_VI6_HGO_LB0_H	0000_0000h	3010h	32
HGO Vertical Position Register for LB Detection Zone-0	VSPI0_VI6_HGO_LB0_V	0000_0000h	3014h	32
HGO Horizontal Position Register for LB Detection Zone-1	VSPI0_VI6_HGO_LB1_H	0000_0000h	3018h	32
HGO Vertical Position Register for LB Detection Zone-1	VSPI0_VI6_HGO_LB1_V	0000_0000h	301Ch	32
HGO Horizontal Position Register for LB Detection Zone-2	VSPI0_VI6_HGO_LB2_H	0000_0000h	3020h	32
HGO Vertical Position Register for LB Detection Zone-2	VSPI0_VI6_HGO_LB2_V	0000_0000h	3024h	32
HGO Horizontal Position Register for LB Detection Zone-3	VSPI0_VI6_HGO_LB3_H	0000_0000h	3028h	32
HGO Vertical Position Register for LB Detection Zone-3	VSPI0_VI6_HGO_LB3_V	0000_0000h	302Ch	32
HGO Component-R Histogram n Register (n = 0 to 63)	VSPI0_VI6_HGO_R_HISTO_n	0000_0000h	3030 + 0004 × n (n = 0 to 63)h	32
Reserve	-	-	3034h to 312Fh	-
HGO Component-R Min/Max Value Register	VSPI0_VI6_HGO_R_MAXMIN	0000_0000h	3130h	32
HGO Component-R Sum Register	VSPI0_VI6_HGO_R_SUM	0000_0000h	3134h	32
HGO Component-R LB Detection Result Register	VSPI0_VI6_HGO_R_LB_DET	0000_0000h	3138h	32
Reserve	-	-	313Ch to 313Fh	-
HGO Component-G Histogram n Register (n = 0 to 63)	VSPI0_VI6_HGO_G_HISTO_n	0000_0000h	3140 + 0004 × n (n = 0 to 63)h	32
HGO Component-G Min/Max Value Register	VSPI0_VI6_HGO_G_MAXMIN	0000_0000h	3240h	32
HGO Component-G Sum Register	VSPI0_VI6_HGO_G_SUM	0000_0000h	3244h	32
HGO Component-G LB Detection Result Register	VSPI0_VI6_HGO_G_LB_DET	0000_0000h	3248h	32
Reserve	-	-	324Ch to 324Fh	-
HGO Component-B Histogram n Register (n = 0 to 63)	VSPI0_VI6_HGO_B_HISTO_n	0000_0000h	3250 + 0004 × n (n = 0 to 63)h	32
HGO Component-B Min/Max Value Register	VSPI0_VI6_HGO_B_MAXMIN	0000_0000h	3350h	32
HGO Component-B Sum Register	VSPI0_VI6_HGO_B_SUM	0000_0000h	3354h	32
HGO Component-B LB Detection Result Register	VSPI0_VI6_HGO_B_LB_DET	0000_0000h	3358h	32
HGO Y/MaxRGB Extended Histogram Address Register	VSPI0_VI6_HGO_EXT_HIST_ADDR	0000_0000h	335Ch	32
HGO Y/MaxRGB Extended Histogram Data Register	VSPI0_VI6_HGO_EXT_HIST_DATA	0000_0000h	3360h	32
HGO Histogram Storing Address Register	VSPI0_VI6_HGO_HISTADD	0000_0000h	336Ch	32
HGO Histogram Swapping Register	VSPI0_VI6_HGO_HSWAP	0000_0000h	3370h	32
Reserve	-	-	3374h to 33FBh	-

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
HGO Parameter Register Reset	VSPI0_VI6_HGO_REGRST	0000_0000h	33FCh	32
HGT Detection Window Offset Register	VSPI0_VI6_HGT_OFFSET	0000_0000h	3400h	32
HGT Detection Window Size Register	VSPI0_VI6_HGT_SIZE	0000_0000h	3404h	32
HGT Mode Register	VSPI0_VI6_HGT_MODE	0000_0000h	3408h	32
HGT Hue Area 0 Register	VSPI0_VI6_HGT_HUE_AREA_0	0000_0000h	340Ch	32
HGT Hue Area 1 Register	VSPI0_VI6_HGT_HUE_AREA_1	0000_0000h	3410h	32
HGT Hue Area 2 Register	VSPI0_VI6_HGT_HUE_AREA_2	0000_0000h	3414h	32
HGT Hue Area 3 Register	VSPI0_VI6_HGT_HUE_AREA_3	0000_0000h	3418h	32
HGT Hue Area 4 Register	VSPI0_VI6_HGT_HUE_AREA_4	0000_0000h	341Ch	32
HGT Hue Area 5 Register	VSPI0_VI6_HGT_HUE_AREA_5	0000_0000h	3420h	32
HGT LB Detection Threshold Register	VSPI0_VI6_HGT_LB_TH	0000_0000h	3424h	32
HGT Horizontal Position Register for LB Detection Zone-0	VSPI0_VI6_HGT_LB0_H	0000_0000h	3428h	32
HGT Vertical Position Register for LB Detection Zone-0	VSPI0_VI6_HGT_LB0_V	0000_0000h	342Ch	32
HGT Horizontal Position Register for LB Detection Zone-1	VSPI0_VI6_HGT_LB1_H	0000_0000h	3430h	32
HGT Vertical Position Register for LB Detection Zone-1	VSPI0_VI6_HGT_LB1_V	0000_0000h	3434h	32
HGT Horizontal Position Register for LB Detection Zone-2	VSPI0_VI6_HGT_LB2_H	0000_0000h	3438h	32
HGT Vertical Position Register for LB Detection Zone-2	VSPI0_VI6_HGT_LB2_V	0000_0000h	343Ch	32
HGT Horizontal Position Register for LB Detection Zone-3	VSPI0_VI6_HGT_LB3_H	0000_0000h	3440h	32
HGT Vertical Position Register for LB Detection Zone-3	VSPI0_VI6_HGT_LB3_V	0000_0000h	3444h	32
Reserve	-	-	3448h to 344Fh	-
HGT Histogram Register	VSPI0_VI6_HGT_HISTO_m_n	0000_0000h	3450 + 0128 × m + 0004 × n (m = 0 to 5, n = 0 to 31)h	32
HGT Max/Min Value Register	VSPI0_VI6_HGT_MAXMIN	0000_0000h	3750h	32
HGT Sum Register	VSPI0_VI6_HGT_SUM	0000_0000h	3754h	32
HGT LB Detection Result Register	VSPI0_VI6_HGT_LB_DET	0000_0000h	3758h	32
Reserve	-	-	375Ch to 3768h	-
HGT Histogram Storing Address Register	VSPI0_VI6_HGT_HISTADD	0000_0000h	376Ch	32
HGT Histogram Swapping Register	VSPI0_VI6_HGT_HSWAP	0000_0000h	3770h	32
Reserve	-	-	3774h to 37FBh	-
HGT Parameter Register Reset	VSPI0_VI6_HGT_REGRST	0000_0000h	37FCh	32
Reserve	-	-	3800h to 3DFFh	-
SHP Control Register0	VSPI0_VI6_SHP_CTRL0	0000_0000h	3E00h	32
SHP Control Register1	VSPI0_VI6_SHP_CTRL1	0000_0000h	3E04h	32
SHP Control Register2	VSPI0_VI6_SHP_CTRL2	0000_0000h	3E08h	32
Reserve	-	-	3E0Ch to 3EFFh	-
IP Version Register	VSPI0_VI6_IP_VERSION	0101_1404h	3F00h	32
Reserve	-	-	3F04h to 3FFFh	-
RPF0 OSD-CLUT Table	VSPI0_VI6_CLUTn_TBLp	0000_0000h	4000 + 0004 × p + 0400 × n (n = 0 to 2, p = 0 to 255)h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Reserve	-	-	4700h to 6FFFh	-
LUT table	VSPI0_VI6_LUT_TBL_p	0000_0000h	7000 + 0004 × p (p = 0 to 255)h	32
CLU table address register	VSPI0_VI6_CLU_ADDR	0000_0000h	7400h	32
CLU table data register	VSPI0_VI6_CLU_DATA	0000_0000h	7404h	32

Note: Read only bits or write only bits or read/write bits are mixed in the registers.

Note: Initial value of IP version registers depends on product and Modules. See **9.9.1.3.18 (1) IP Version Register (VI6_IP_VERSION)** for more detail.

9.9.1.3 Register Description

9.9.1.3.1 Notational Conventions for Registers and Bit Fields

This document uses the following notational conventions for the VSPI registers and bit fields.

1. The names of registers and bits are written in uppercase.
2. A bit or bit field in a register is indicated as [register name.bit name]. For example, the STRCMD bit in the VI6_CMD0 register is indicated as VI6_CMD0.STRCMD.
3. In each subsection for register description in **9.9.1.3 Register Description**, when only a bit name is written without showing its register name, the bit is in the register described in that subsection.
4. A wildcard (*) indicates any characters in a name and represents all registers or bits that match the specified first part of a name. For example, when there are two registers VI6_RPF_SRC_BSIZE and VI6_RPF_SRC_ESIZE, VI6_RPF_SRC_* indicates both registers.

9.9.1.3.2 Register Classification

The VSPI registers are arranged in the following order; the general control registers to display list control registers control operation of the entire VSPI, and the other registers control each image processing and specify parameters for the processing. The functions of the registers are described in this order starting from **9.9.1.3.4 General Control Registers**.

1. General control registers
2. Display list control registers
3. RPF control registers (RPF0)
4. WPF control registers (WPF0)
5. DPR control registers
6. SRU control registers
7. UDS control registers
8. LUT control register
9. CLU control register
10. HST control register
11. HSI control register
12. HGO control registers
13. HGT control registers
14. SHP control registers
15. VSPI internal registers

9.9.1.3.3 Restrictions on Access to Registers and Lookup Tables

The VSPI has control registers and lookup tables. All VSPI registers are writable and readable by only 32 bits unit. To write partial bits in each register, read-modify-write is needed. When accessing the addresses where these registers and lookup tables are allocated, the following restrictions should be satisfied. If any restriction is violated, the VSPI will not operate correctly.

1. For the read-only bits and reserved bits in all VSPI registers, writing 1 is prohibited unless otherwise specified.
2. Addresses undefined in **9.9.1.2.2 Register/Table Address** are reserved areas and write access is prohibited in these areas.
3. For all registers and lookup tables, except VI6_CMD0, VI6_SRESET, VI6_*IRQ* and two plane registers such as VI6_DL_HDR_ADDR0, and VI6_DL_BODY_SIZE0, modifying register values during operation of the module is prohibited. Modify registers while the corresponding module is stopped. For the operating status of the target module, refer to **9.9.1.4.2 Concept of VSPI Operation Starting and Stopping**.
4. There are three types about General control registers (**9.9.1.3.4 General Control Registers**) and Display List Control Registers (**9.9.1.3.5 Display List Control Registers**) as below.
   - Controlling WPF0 (ex. VI6_CMD0, VI6_SRESET.SRST0)
   - Common setting of WPF0 (ex. VI6_DL_CTRL.AR_WAIT [15:0])

Table 9.9-4 Correspondence between Modules and Register Names

Module Name	Register Name
RPF0	VI6_RPF0_*
WPF0	VI6_WPF0_*
DPR	VI6_DPR_*
SRU	VI6_SRU_*
SHP	VI6_SHP_*
UDS	VI6_UDS_*
LUT	VI6_LUT_CTRL
CLU	VI6_CLU_CTRL
HST	VI6_HST_CTRL
HSI	VI6_HSI_CTRL
HGO	VI6_HGO_*
HGT	VI6_HGT_*

9.9.1.3.4 General Control Registers

(1) VSPI Start Registers 0 (VI6_CMD0)

Access Size : 32 bits
Address : <VSPI0_base> + 0000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	UPDHDR	-	-	-	STRCMD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
4	UPDHDR	0h	RW	Reserved state of updating DLH address. UPDHDR can be negated by software reset (VI6_SRESET.SRSTn). [Write]0: NOP 1: Don't set 1 to this bit [Read]0: Not reserved to update DLHeader address and VI6_HGO/HGT_WBUFS. 1: Reserved to update DLHeader address and VI6_HGO/HGT_WBUFS. Set 0 to this bit when STRCMD bit is
3 to 1	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
0	STRCMD	0h	RW	Start reservation of each WPF VSPI has two output channels (WPFs). Two WPFs can be started at the same time. VI6_CMDn.STRCMD controls WPFn. Writing 1 to this bit starts WPFn in VSPI. Set this bit for activation only after all register settings in each output channel have been completed. If WPFn is idle, WPFn starts right after this bit is set to 1. If WPFn is active, writing 1 to this bit reserves starting WPFn operation. Wait to set 1 to VI6_CMDn.STRCMD until this bit is read as 0. VI6_CMDn.STRCMD can be negated by software reset (VI6_SRESET.SRSTn). [Write]0: Start reservation of WPFn is canceled. 1: Start reservation of WPFn is set. [Read]0: Starting VSP is not reserved. 1: Starting VSP is reserved.

The basic concept of image processing operation started by activating the VSPI is shown in **Figure 9.9-5**. The actual data input/output is executed by the MAU, which is the bus interface module, as described in **9.9.1.1.1 Block diagram of VSPI**, but conceptually the RPF works as the data entry point to the VSPI and the WPF works as the data exit point. To process images through the VSPI, the RPF (entrance) and WPF (exit) should be connected and a data path from the RPF to the WPF should be formed.

To connect RPF0 to WPF0, specify RPF0 as the source RPF for WPF0 in VI6_WPF0_SRCRPF, which is a register for WPF0 (refer to **9.9.1.3.7 (1) WPF0-Source-RPF Registers (VI6_WPF0_SRCRPF)**). This setting determines that RPF0 will be started when WPF0 is started through VI6_CMD0.

A data path to execute desired image processing should then be formed between the RPF (entrance) and WPF (exit). To form a data path, connect the necessary function modules in the VSPI between RPF and WPF. This function is provided

by the DPR; specify the information for each module connection in data path routing registers VI6_DPR_*_ROUTE (refer to **9.9.1.3.8 DPR Control Registers**).

After a data path is formed (RPF0 → WPF0) as described above, starting output module WPF0 in the VSPI through VI6_CMDn starts all function modules connected to WPF0 and the desired image processing is executed. According to this design concept, starting a WPF module means starting the VSPI.

Figure 9.9-5 shows an example of a configuration where modules with single input and single output are implemented through the DPR.

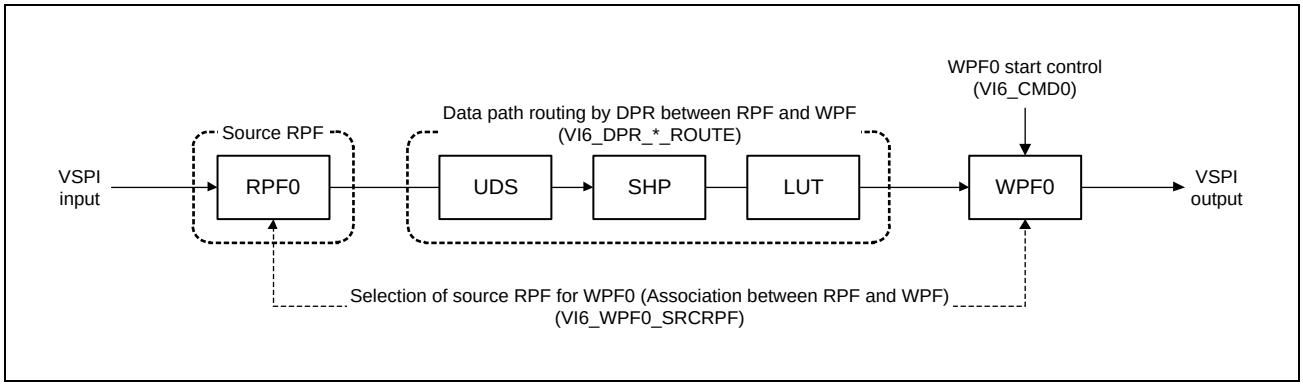


Figure 9.9-5 Basic Concept of VSPI Startup

(2) Clock Control Register 0 (VI6_CLK_CTRL0)

Access Size : 32 bits
Address : <VSPI0_base> + 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	GCS0	-	-	-	-	-	-	-	-	-	-	-	GCS1
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	GCS2[3:0]				-	-	-	GCS3[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28	GCS0	0h	RW	Clock Control Setting 0
27 to 17	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
16	GCS1	0h	RW	Clock Control Setting 1
15 to 12	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
11 to 8	GCS2[3:0]	0h	RW	Clock Control Setting 2
7 to 5	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
4 to 0	GCS3[4:0]	0h	RW	Clock Control Setting 3

VSPI can stop its operating clock for reducing power consumption.

To enable clock stop function, set following registers:

- VI6_CLK_CTRL0 = 0000_0000h
- VI6_CLK_CTRL1 = 0000_0000h
- VI6_CLK_DCSWT = 0000_0808h
- VI6_CLK_DCSM0 = 0000_0000h
- VI6_CLK_DCSM1 = 0000_0000h

To disable clock stop function, set following registers:

- VI6_CLK_CTRL0 = 1001_0F1Fh
- VI6_CLK_CTRL1 = FF10_FFFFh
- VI6_CLK_DCSWT = 0033_0808h
- VI6_CLK_DCSM0 = 1FFF_0F1Fh
- VI6_CLK_DCSM1 = FF10_FFFFh

(3) Clock Control Register 1 (VI6_CLK_CTRL1)

Access Size : 32 bits
Address : <VSPI0_base> + 0014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GCS4[7:0]								-	-	-	GCS6	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	RW	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GCS5[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GCS4[7:0]	0h	RW	Clock Control Setting 4
23 to 21	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
20	GCS6	0h	RW	Clock Control Setting 6
19 to 16	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
15 to 0	GCS5[15:0]	0h	RW	Clock Control Setting 5

See **9.9.1.3.4 (2) Clock Control Register 0 (VI6_CLK_CTRL0)** for detail.

(4) Dynamic Clock Stop Control Register (VI6_CLK_DCSWT)

Access Size : 32 bits
Address : <VSPI0_base> + 0018h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	DCC0[1:0]		-	-	DCC1	DCC2
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	R	R	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSTPW[7:0]								CSTRW[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
21, 20	DCC0[1:0]	0h	RW	Clock Control Setting 0
19 to 18	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
17	DCC1	0h	RW	Clock Control Setting 1
16	DCC2	0h	RW	Clock Control Setting 2
15 to 8	CSTPW[7:0]	0h	RW	Dynamic Clock Stop Control 1 Always specify 8.
7 to 0	CSTRW[7:0]	0h	RW	Dynamic Clock Stop Control 2 Always specify 8.

See **9.9.1.3.4 (2) Clock Control Register 0 (VI6_CLK_CTRL0)** for detail.

(5) Dynamic Clock Stop Disable Register 0 (VI6_CLK_DCSM0)

Access Size : 32 bits
Address : <VSPI0_base> + 001Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	DCD0[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	DCD1[3:0]				-	-	-	DCD2[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28 to 16	DCD0[12:0]	0h	RW	Dynamic Clock Stop Disable Setting 0
15 to 12	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
11 to 8	DCD1[3:0]	0h	RW	Dynamic Clock Stop Disable Setting 1
7 to 5	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
4 to 0	DCD2[4:0]	0h	RW	Dynamic Clock Stop Disable Setting 2

See **9.9.1.3.4 (2) Clock Control Register 0 (VI6_CLK_CTRL0)** for detail.

(6) Dynamic Clock Stop Disable Register 1 (VI6_CLK_DCSM1)

Access Size : 32 bits
Address : <VSPI0_base> + 0020h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCD3[7:0]								-	-	-	DCD5	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	RW	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCD4[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DCD3[7:0]	0h	RW	Dynamic Clock Stop Disable Setting 3
23 to 21	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
20	DCD5	0h	RW	Dynamic Clock Stop Disable Setting 5
19 to 16	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
15 to 0	DCD4[15:0]	0h	RW	Dynamic Clock Stop Disable Setting 4

See **9.9.1.3.4 (2) Clock Control Register 0 (VI6_CLK_CTRL0)** for detail.

(7) Software Reset Register (VI6_SRESET)

Access Size : 32 bits
Address : <VSPI0_base> + 0028h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SRST0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
1	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
0	SRST0	0h	RW	WPF0 Software Reset (SRST0) Writing 1 to this bit aborts the current processing in WPF0 (the partially-completed image undergoing processing is output). The period until this software reset processing is completed depends on the bus state. When this reset processing is completed, the VI6_WPF0_IRQ_STA.FRE interrupt source bit is set to 1; when the FRE interrupt is enabled, the FRE end interrupt is output to notify the end of the reset processing. This bit is always read as 0. 0: NOP 1: WPF0 software reset (*)

Note 1. Applying a software reset to each WPF has the following restrictions.

1) A software reset can be applied to only one of WPF0 through single write access to VI6_SRESET.
2) After a software reset is issued, no more software reset can be issued to another WPF until the issued software reset processing is completed.
3) The end of software reset processing is notified through the FRE bit in VI6_WPF0_IRQ_STA, but the software reset issued while WPF is stopped is ignored as NOP. As it takes a while until the reset is actually issued after the reset bit is set, the VSPI may complete operation before the reset is actually issued. In this case, no interrupt is output for the software reset that is issued after the VSPI completes operation.
4) If a software reset is issued during downloading of a display list, the downloading processing is not aborted. After the end of downloading that is in progress when a software reset is issued, a frame end interrupt is output.
5) When software reset is issued to VSPI, see **9.9.2.6.2 (1) Software Reset**.

(8) Module Reset Enable Register 0 (VI6_MRESET_ENB0)

This register is for purpose of h/w debugging.

Access Size : 32 bits
Address : <VSPI0_base> + 002Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	MRSTE0[1:0]		-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	MRSTE1[3:0]				-	-	-	MRSTE2[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
29, 28	MRSTE0[1:0]	0h	RW	Module Reset Enable 0
27 to 12	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
11 to 8	MRSTE1[3:0]	0h	RW	Module Reset Enable 1
7 to 5	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
4 to 0	MRSTE2[4:0]	0h	RW	Module Reset Enable 2

(9) Module Reset Enable Register 1 (VI6_MRESET_ENB1)

This register is for purpose of h/w debugging.

Access Size : 32 bits
Address : <VSPI0_base> + 0030h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MRSTE3[7:0]								-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MRSTE4[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	MRSTE3[7:0]	0h	RW	Module Reset Enable 3
23 to 16	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
15 to 0	MRSTE4[15:0]	0h	RW	Module Reset Enable 4

(10) Module Reset Issuing Register (VI6_MRESET)

Access Size : 32 bits

Address : <VSPI0_base> + 0034h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
0	MRST	0h	RW	Module Reset Assertion This register is for purpose of h/w debugging.

(11) Operating Status Register (VI6_STATUS)

Access Size : 32 bits

Address : <VSPI0_base> + 0038h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	FLDST0	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	SYS0_ACT	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28	FLDST0	0h	R	Field status of previous frame of WPF0. 0: Previous finished frame is TOP field. 1: Previous finished frame is BOT field. This bit can be referred in case of AUTO-FLD or AUTO-DISP. This bit is changed at the timing of VI6_WPF0_IRQ_STA.FRE.
27 to 9	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
8	SYS0_ACT	0h	R	WPF0 Operating Status (SYS0_ACT) Each bit indicates the operating or stopped state of control channel 0 (WPF0). 0: WPF0 is stopped. 1: WPF0 is operating.
7 to 0	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.

(12) WPF0 Interrupt Enable Registers (VI6_WPF0_IRQ_ENB)

Each bit controls the interrupt enable of the corresponding interrupt source.

0: Interrupt Disabled

1: Interrupt Enabled

Access Size : 32 bits
Address : <VSPI0_base> + 0048h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	UNDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DFEE	FREE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
16	UNDE	0h	RW	Interrupt Enable for WPF0 Underrun in case of DU connection.
15 to 2	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
1	DFEE	0h	RW	Interrupt Enable for WPF0 Display List Frame End
0	FREE	0h	RW	Interrupt Enable for WPF0 Frame End

Each bit in VI6_WPF0_IRQ_STA is set to 1 when the corresponding interrupt source is generated. VI6_WPF0_IRQ_ENB specifies whether to output an interrupt signal for the generated source. When an interrupt is disabled in this register, no interrupt signal is generated even when the corresponding bit in VI6_WPF0_IRQ_STA is set to 1. When an interrupt is enabled in this register, an interrupt signal is output when the corresponding bit in VI6_WPF0_IRQ_STA is set to 1.

(13) WPF0 Interrupt Status Registers (VI6_WPF0_IRQ_STA)

The read value from each bit is the status of the interrupt source, and the write access to each bit controls the interrupt status.

[Read Access] Interrupt Status

0: No interrupt

1: Interrupt activated

[Write Access] Interrupt Clear

0: The interrupt status is cleared to 0

1: Hold the interrupt status value

Access Size : 32 bits
Address : <VSPI0_base> + 004Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	UND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DFE	FRE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
16	UND	0h	RW	Interrupt Status and Clear for WPF0 Underrun in case of DU connection. This interrupt source bit is set to 1 when data underrun occurs in case of DU connection. Timing that this bit is set to 1 is at end of 1 frame.
15 to 2	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
1	DFE	0h	RW	Interrupt Status and Clear for WPF0 Display List Frame End This interrupt source bit is set to 1 when VSPI completes one-frame processing while the current frame in enable value stored in the display list header is 1 (refer to **9.9.1.4.3 (2)**). When display lists are not used, this bit is not used. In this case, clear VI6_WPFn_IRQ_ENB.DFEE to 0 to mask the interrupt generation by this interrupt source. This bit can hold the most recent of two times of interrupt status at the maximum. When writing 0 to this bit, the oldest interrupt status is cleared. And the status bit becomes 0 after all interrupt statuses are cleared.
0	FRE	0h	RW	Interrupt Status and Clear for WPF0 Frame End This interrupt source bit is set to 1 when VSPI completes one-frame processing. This bit is also set to 1 when one-frame processing using a display list is completed. The interrupt status is set to 1 by any of the following conditions. a) Processing one frame is finished normally b) Software reset is issued during VSP is processing. c) One frame's data from DU is displayed while VSP couldn't transfer one frame data to DU. (in case of Linked with DU) This bit can hold the most recent of two times of interrupt status at the maximum. When writing 0 to this bit, the oldest interrupt status is cleared. And the status bit becomes 0 after all interrupt statuses are cleared.

VI6_WPF0_IRQ_STA indicates the state of the interrupt sources generated in the VSPI. Whether to output a VSPI interrupt when an interrupt source is generated and the corresponding bit is set to 1 is determined by the corresponding bit setting in VI6_WPF0_IRQ_ENB. While an interrupt is disabled in VI6_WPF0_IRQ_ENB, the VSPI does not output an interrupt signal even when an interrupt source is generated, but the source flag in this register is set to 1.

Note that the interrupt source bits in this register cannot be cleared by write access using a display list.

9.9.1.3.5 Display List Control Registers

(1) Display List Control Register (VI6_DL_CTRL)

Access Size : 32 bits
Address : <VSPI0_base> + 0100h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AR_WAIT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	DC2	-	-	-	DC1	-	-	-	-	RLM0	CFM0	NH0	DLE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	AR_WAIT[15:0]	0h	RW	Display List Control Setting Always specify 256.
15 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12	DC2	0h	RW	This bit doesn't affect anything to VSP (NOP)
11 to 9	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
8	DC1	0h	RW	This bit doesn't affect anything to VSP (NOP)
7 to 5	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
4	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
3	RLM0	0h	RW	Loading two plane Registers Mode for WPF0. 0: Reserved status (VI6_CMD0.UPDHDR) is accepted by next frame auto start. Two plane registers are downloaded by next_frame_start also. 1: Reserved status (VI6_CMD0.UPDHDR) is not accepted by next frame auto start. pNextHeader in previous frame's DLH is used for loading DisplayList. Two plane registers are not downloaded by next_frame_start also.
2	CFM0	0h	RW	Continuous Frame Mode for Header-less Display List for WPF0 This bit determines whether the next frame is automatically started or not. When the updated flag of the display list, VI6_DL_BODY_SIZE0.UPD0, is not updated, the display list of the next frame is not transferred and the same register values are used for the next frame. When the value of VI6_DL_BODY_SIZE0.UPD0 is updated, the new display list is transferred. 0: Stopped at the end of every frame 1: The next frame is automatically started
1	NH0	0h	RW	Header-less Display List Mode This bit is used for specifying the header-less display list mode. In case of header-less mode, the number of the display lists is 1. The address of the display body is set in VI6_DL_HDR_ADDR0 register, and the body size is set in VI6_DL_BODY_SIZE0 register. When this bit is changed, make sure that VSPI is stopped. And also make sure the following before starting VSP. - Header Address (VI6_DL_HDR_ADDR0) - Body Size (VI6_DL_BODY_SIZE0) in case of header-less mode 0: Use Display List Header (Normal DL Mode) 1: Don't use Display List Header (Header-less Mode) Note: When DLE0 bit is 0, set 0 to NH0 bit.

Bit	Bit Name	Initial Value	R/W	Description
0	DLE0	0h	RW	Display List Enable/Disable for WPF0 Enables or disables the WPF0 display list function. When the display list function is enabled through this bit, all WPF processing channels work in display list mode. When using display lists, note the restrictions in **9.9.1.3.3**. 0: The display list function is disabled 1: The display list function is enabled

(2) Display List-0 Header Address Register (VI6_DL_HDR_ADDR0)

Access Size : 32 bits
Address : <VSPI0_base> + 0104h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DL_HDR_PTR-0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DL_HDR_PTR-0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DL_HDR_PTR-0[31:0]	0h	RW	Display List-0 Header Address These bits specify the address of the display list header to be read for display list-0 in 16-byte units (the lower-order four bits are read-only). When WPF0 is first started in display list mode, the display list header is loaded from the address specified in this register. After loading of the header is completed, the register value of the display list address is updated to the next header address stored in the loaded header to prepare for loading of the next display list header. After that, this header address updating is repeated. When VI6_DL_CTRL.NH0 is 0, writing header address to this register reserves updating two plane registers for the next frame. Two plane register is VI6_DL_HDR_ADDR0. A value from 0000_0000h to FFFF_FFF0h can be specified.

(3) Display List-0 Data Swapping Register (VI6_DL_SWAP0)

Access Size : 32 bits
Address : <VSPI0_base> + 0114h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	LWS	WDS	BTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

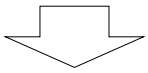
Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved These bits are read as 0b. The write value should always be 0.
2	LWS	0h	RW	Display List Data Swapping in long word Units The effect of this bit setting is defined in **Table 9.9-5**. 0: Data swapping in long word (32-bit) units is disabled 1: Data swapping in long word (32-bit) units is enabled
1	WDS	0h	RW	Display List Data Swapping in Word Units The effect of this bit setting is defined in **Table 9.9-5**. 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled
0	BTS	0h	RW	Display List Data Swapping in Byte Units The effect of this bit setting is defined in **Table 9.9-5**. 0: Data swapping in byte (8-bit) units is disabled 1: Data swapping in byte (8-bit) units is enabled

Table 9.9-5 shows the data order before and after swapping according to the long word, word, and byte swapping settings.

When data order in memory for each format is the same as **Table 9.9-41**, set 111b to {LWS, WDS, BTS}. If data order is not the same as the definition, change data order within 8byte unit by these bits as shown in **Table 9.9-5**.

Table 9.9-5 Changing Data Order According to Display List Swap Register

Data Order in Memory											
Byte Address	8n+0	8n+1	8n+2	8n+3	8n+4	8n+5	8n+6	8n+7	*_LWS	*_WDS	*_BTS
Data	0	1	2	3	4	5	6	7	1	1	1
	1	0	3	2	5	4	7	6	1	1	0
	2	3	0	1	6	7	4	5	1	0	1
	3	2	1	0	7	6	5	4	1	0	0
	4	5	6	7	0	1	2	3	0	1	1
	5	4	7	6	1	0	3	2	0	1	0
	6	7	4	5	2	3	0	1	0	0	1
	7	6	5	4	3	2	1	0	0	0	0



Data Order Defined in **Table 9.9-41**								
Byte Address	8n+0	8n+1	8n+2	8n+3	8n+4	8n+5	8n+6	8n+7
Data	0	1	2	3	4	5	6	7

(4) Extended Display List-0 Control Register (VI6_DL_EXT_CTRL0)

Access Size : 32 bits
Address : <VSPI0_base> + 011Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	NEW
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	POLINT[5:0]						-	-	DLPRI	EXPRI	-	-	-	EXT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
16	NEW	0h	RW	No Wait for Polling When this bit is set to 1, the polling condition for extended display lists is always assumed to be true.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13 to 8	POLINT[5:0]	0h	RW	Extended Display List Command Control Always specify 2.
7, 6	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
5	DLPRI	0h	RW	Display List Control 0 Always specify 1.
4	EXPRI	0h	RW	Display List Control 1 Always specify 0.
3 to 1	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
0	EXT	0h	RW	Extended Display List for WPF0 Enables or disables the extended display list function. When extended display lists are used, the display list header size is 96 bytes; when they are not used, the header size is 80 bytes. 0: No extended display lists are used 1: Extended display lists are used [Note] When using extended display lists, be sure to also use normal display list mode (VI6_DL_CTRL.DLEn); executing only extended display lists is not possible. When the header-less display list mode is activated, this bit should be set to 0. The extended display list cannot be used with the header-less display list mode.

(5) Display List Body Size Register-0 (VI6_DL_BODY_SIZE0)

Access Size : 32 bits
Address : <VSPI0_base> + 0120h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	UPD0	-	-	-	-	-	-	-	BS0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BS0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
24	UPD0	0h	RW	Update Flag This bit controls the download of the display list at the next downloading timing in case that the header-less display list mode is used. Set 1 to this bit when display list need to be downloaded for next frame. This bit is automatically cleared to 0 after VI6_DL_HDR_ADDRn and VI6_DL_BODY_SIZEn.BS0 (n = 0) is downloaded in H/W side. When this bit is set to 1, the value of VI6_DL_HDR_ADDRn and VI6_DL_BODY_SIZEn.BS0 (n = 0) should not be changed. 0: Updating display list for the next frame is not reserved 1: Updating display list for the next frame is reserved
23 to 17	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
16 to 0	BS0[16:0]	0h	RW	Header-less Display List Body Size (WPFn, n = 0) These bits are used for specifying the body size of the display list in case of header-less display list mode. The unit of the size is byte. The value should be set in multiples of 8.

(6) Display List-0 Header Reference Address Register (VI6_DL_HDR_REF_ADDR0)

Access Size : 32 bits

Address : <VSPI0_base> + 0130h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DL_HDR_REF_PTR-0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DL_HDR_REF_PTR-0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DL_HDR_REF_ PTR-0[31:0]	0h	R	Display List-0 reference Header Address Following value is read out from the VI6_DL_HDR_REF_ADDR0 for each period. (1) When H/W is reading display list from external memory, header address of the display list referred by VSPI-H/W is read out. (2) When H/W is not reading display list from external memory, the value of VI6_DL_HDR_ADDR0 is read out.

9.9.1.3.6 RPF Control Registers

(1) RPF0 Basic Read Size Registers (VI6_RPF0_SRC_BSIZE)

Access Size : 32 bits
Address : <VSPI0_base> + 0300h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	BHSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	BVSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28 to 16	BHSIZE[12:0]	0h	RW	Horizontal Size of RPF Basic Read Area These bits specify the horizontal size of the basic area to be read from the external RAM by the RPF0. When the input format is YCbCr4:2:2 or YCbCr4:2:0, specify the size in 2-pixel units. A value from 1 to 8190 can be specified. Specify a value equal to or smaller than the extended read size (VI6_RPF0_SRC_ESIZE.EHSIZE).
15 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12 to 0	BVSIZE[12:0]	0h	RW	Vertical Size of RPF Basic Read Area These bits specify the vertical size of the basic area to be read from the external RAM by the RPF0. When the input format is YCbCr4:2:0, specify the size in 2-pixel units. A value from 1 to 8190 can be specified. Specify a value equal to or smaller than the extended read size (VI6_RPF0_SRC_ESIZE.EVSIZE).

Figure 9.9-6 shows the relationship between the basic read size and extended read size. The RPF reads data from the source memory area specified by the basic read size. The RPF repeats reading the basic read area in the horizontal and vertical directions up to the extended read size and sends the read data to the processing modules in the VSPI.

For basic read size reading, the reading start address, called the RPF0 source image storing address, should be specified in VI6_RPF0_SRCM_ADDR_*. In the memory area where the basic read area image is stored, the distance (number of bytes) between addresses for lines n and n + 1 of two-dimensional image data, called the memory stride, should be specified in VI6_RPF0_SRCM_PSTRIDE.

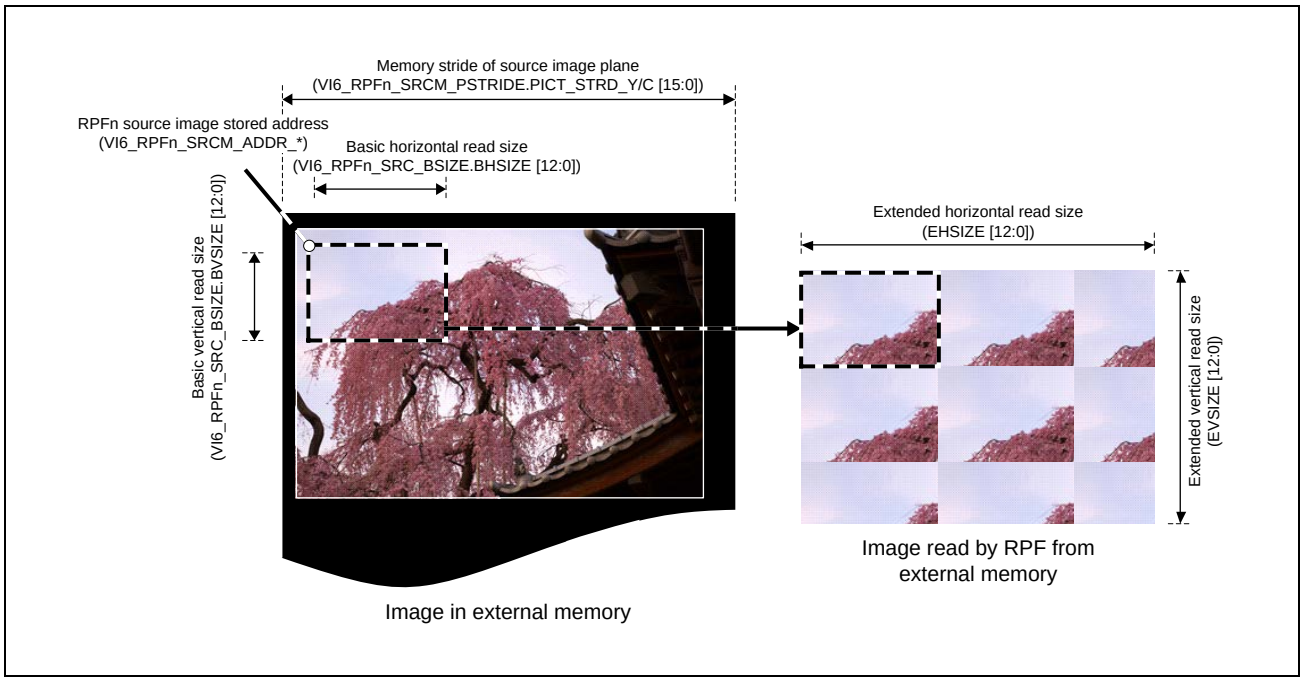


Figure 9.9-6 Relationship between Basic Read Size and Extended Read Size

Refer also to the following sections.

- **9.9.1.3.6 (2) RPF0 Extended Read Size Registers (VI6_RPF0_SRC_ESIZE)**
- **9.9.1.3.6 (13) RPF0 Source Picture Memory Stride Setting Registers (VI6_RPF0_SRCM_PSTRIDE)**
- **9.9.1.3.6 (15) RPF0 Source Y/RGB Address Registers (VI6_RPF0_SRCM_ADDR_Y)**
- **9.9.1.3.6 (16) RPF0 Source Chroma Address Registers 0 (VI6_RPF0_SRCM_ADDR_C0)**
- **9.9.1.3.6 (17) RPF0 Source Chroma Address Registers 1 (VI6_RPF0_SRCM_ADDR_C1)**

(2) RPF0 Extended Read Size Registers (VI6_RPF0_SRC_ESIZE)

Access Size : 32 bits
Address : <VSPI0_base> + 0304h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	EHSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	EVSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28 to 16	EHSIZE[12:0]	0h	RW	RPF Extended Horizontal Read Size These bits specify the horizontal size of the extended read area to which the RPF0 reads data from the external RAM. As shown in **Figure 9.9-6**, the basic read area image is repeatedly placed in the extended read area; in the EHSIZE bits, specify a value not smaller than the horizontal size of the basic read area. When the input format is YCbCr4:2:2 or YCbCr4:2:0, specify the size in 2-pixel units (an even value). A value from 1 to 8190 can be specified.
15 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12 to 0	EVSIZE[12:0]	0h	RW	RPF Extended Vertical Read Size These bits specify the vertical size of the extended read area to which the RPF0 reads data from the external RAM. As shown in **Figure 9.9-6**, the basic read area image is repeatedly placed in the extended read area; in the EVSIZE bits, specify a value not smaller than the vertical size of the basic read area. When the input format is YCbCr4:2:0, specify the size in 2-pixel units (an even value). A value from 1 to 8190 can be specified.

VI6_RPF0_SRC_ESIZE specifies the extended size for RPF0. The extended horizontal and vertical sizes should be equal to or greater than the basic sizes specified in VI6_RPF0_SRC_BSIZE. The RPF internal data processing described later and image processing described in **9.9.1.3.8 DPR Control Registers** and later sections are all applied to the image in the extended read size shown on the right side in **Figure 9.9-6**.

(3) RPF0 Input Format Registers (VI6_RPF0_INFMT)

Access Size : 32 bits
Address : <VSPI0_base> + 0308h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	VIR	-	-	-	-	-	-	-	-	-	-	-	CIPM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPYCS	SPUVS	CEXT[1:0]		RDTM[2:0]			CSC	-	RDFMT[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28	VIR	0h	RW	Virtual Input Enable Enables or disables the virtual input function of the RPF0. The image to be processed by the RPF0 is usually read from the external memory by the MAU. Instead of this input, the virtual input function generates a single-color image within the RPF0 and sends it to the modules in VSP. When the virtual input function is enabled, the fixed value specified in VI6_RPF0_VRTCOL_SET is used as the input to the RPF0. While the virtual input function is enabled, data is not read from the external memory; that is, the α plane is not read and the IROP calculation thus cannot be executed. In this case, set VI6_RPF0_ALPH_SEL.ASEL to 4. Neither the color space conversion through CSC nor the color keying described in **9.9.1.3.6 (11)** can be used. 0: RPF0 uses general input. 1: RPF0 uses virtual input.
27 to 17	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
16	CIPM	0h	RW	Horizontal Chrominance Interpolation Method Setting Image data is processed in the YCbCr444 format inside VSP in case of YCbCr color space. When the chrominance format of the input image is YCbCr422 or YCbCr420, data is up-sampled as shown in **Figure 9.9-7** for internal processing. This bit specifies the method of up-sampling for this purpose. 0: The nearest-neighbor method is used for horizontal chrominance interpolation. 1: The bilinear method is used for horizontal chrominance interpolation.
15	SPYCS	0h	RW	RPF Input Mode Setting 1 When the input format is YUY2, set this bit to 1 and set the RDFMT bits to 71 (47h). When the input format is YVYU, set this bit and the SPUVS bit to 1 and set the RDFMT bits to 71 (47h). In other cases, set this bit to 0.
14	SPUVS	0h	RW	RPF Input Mode Setting 2 When the input format is NV61, set this bit to 1 and set the RDFMT bits to 65 (41h). When the input format is NV21, set this bit to 1 and set the RDFMT bits to 66 (42h). When the input format is YVYU, set this bit and the SPYCS bit to 1 and set the RDFMT bits to 71 (47h). In other cases, set this bit to 0.
13, 12	CEXT[1:0]	0h	RW	Lower-Bit Color Data Extension Method Setting When an RGB input format where each color component is expressed in less than 8 bits are selected from **Table 9.9-7** through the RDFMT bits, VSP internally extends each color component to 8 bits before using the data. These bits select this extension method. 00b: Lower-order bits of color data are extended with 0. 01b: Upper-order bits of color data are copied to the lower -order bits. 10b: Lower-order bits of color data are extended with 0. The maximum value is limited to FFh. 11b: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
11 to 9	RDTM[2:0]	0h	RW	CSC Conversion Expression Setting These bits select the expression used for color space conversion. The conversion direction is RGB → YCbCr when RGB is selected through the RDFMT bits; the direction is YCbCr → RGB when YCbCr is selected. 000b: BT.601 YCbCr [16,235/240] ↔ RGB [0,255] 001b: BT.601 YCbCr [0,255] ↔ RGB [0,255] 010b: BT.709 YCbCr [16,235/240] ↔ RGB [0,255] 011b: BT.709 YCbCr [16,235/240] ↔ RGB [16,235] 100b to 111b: Setting prohibited
8	CSC	0h	RW	Color Space Conversion Enable Enables or disables color space conversion between YCbCr and RGB to be executed in RPF0. The characteristics of color space conversion are determined by the RDTM bit setting. (*1) When using the virtual input (VIR = 1), specify 0. 0: Color space conversion is disabled. 1: Color space conversion is enabled.
7	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.
6 to 0	RDFMT[6:0]	0h	RW	RPF Input Image Format Setting These bits select the format of the image input from the external RAM to the RPF0. Select a value corresponding to the desired format from those shown in **Table 9.9-7** and **Table 9.9-8**. When the virtual input function is used (VIR = 1), the color information for the virtual input should be specified in VI6_RPF0_VRTCOL_SET. If this color information is in the RGB format, set the RDFMT bits to 19. If the color information is in the YCbCr format, set these bits to 64. [Note 1] Number of input pixels When YCbCr4:2:2 is selected through the RDFMT bits, the horizontal size of the input image should be specified in 2-pixel units. When YCbCr4:2:0 is selected, the vertical and horizontal sizes should be specified in 2-pixel units. Observe these restrictions when specifying the image size in VI6_RPF0_SRC_BSIZE and VI6_RPF0_SRC_ESIZE. [Note 2] OSD-CLUT Setting When the RDFMT bits are set to 3Fh, RGB color data should be stored in the OSD-CLUT. When these bits are set to 7Fh, YCbCr color data should be stored in the OSD-CLUT. Note that while the target WPF is operating, the RPF is also operating and the OSD-CLUT cannot be read or written to.

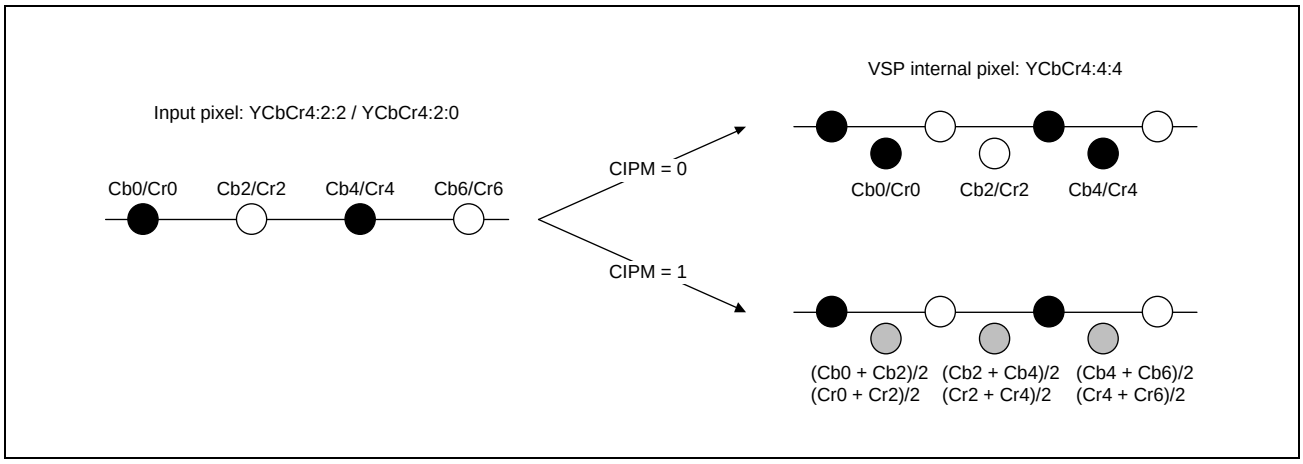


Figure 9.9-7 Chrominance Interpolation Methods Selectable through CIPM Setting

Note on color space settings

The color space for the image output from the RPF to VSPI internal modules is determined by the combination of the color space for the image input to the RPF, which is selected through the VI6_RPF0_INFMT.RDFMT setting, and the enabled or disabled state of the color space conversion function, which is selected through the VI6_RPF0_INFMT.CSC setting (**Table 9.9-6**). For example, when the image input to the RPF is in the YCbCr format, the RPF outputs data to VSPI internal modules in the YCbCr format if color space conversion is disabled through the CSC bit, and the RPF outputs data in the RGB format if color space conversion is enabled. When the image input to the RPF is in the RGB format, the relationship between the output format and the color space conversion setting is the opposite of the YCbCr case. For some VSPI internal modules, the YCbCr format is recommended for image processing because of the

characteristics of the processing, or the same color space needs to be specified between multiple RPF outputs. In these cases, set VI6_RPF0_INFMT.RDFMT and VI6_RPF0_INFMT.CSC appropriately so that the RPF can output the required color space according to the color space conditions described above.

Table 9.9-6 RPF0 Input Color Space and Output Color Space

RPF0 Input Color Space (VI6_RPF0_INFMT. RDFMT)		Color Space Conversion Setting (VI6_RPF0_INFMT.CSC)		RPF0 Output Color Space
RGB	(00h to 3Fh)*1	Disabled	(0)*1	RGB
		Enabled	(1)*1	YCbCr
YCbCr	(40h to 7Fh)*1	Disabled	(0)*1	YCbCr
		Enabled	(1)*1	RGB

Note 1. Value specified in the register

A color space conversion function equivalent to that in the RPF0 is also provided by the WPF. As shown in **Table 9.9-6**, the color space (YCbCr or RGB) output from the RPF becomes the input format for the WPF. Here, the color space of the output image obtained by the color space conversion function of the WPF must match the color space of the format specified through VI6_WPF0_OUTFMT.WRFMT.

Figure 9.9-8 shows the relationship between the input/output format and color space. The input color space for the RPF is determined when the input image format for the RPF is specified through the RDFMT bits. The color space for the image output from the RPF to subsequent VSPI internal modules depends on the combination of the RPF input format and CSC (color space conversion function) enabled or disabled state in the RPF as shown in **Table 9.9-6** and **Figure 9.9-8**. The user should first determine whether the image processing in the VSPI is done in YCbCr or RGB, and then specify the RPF input format and CSC enabled or disabled state to obtain the desired color space. The color space of the RPF output image is also that of the WPF input image; the color space of the data output from the WPF to the outside of VSPI depends on the combination of the WPF input color space and the enabled or disabled state of the CSC implemented in the WPF as shown in **Figure 9.9-8**. The color space of the WPF output image must match that of the WPF output format (determined by VI6_WPF0_OUTFMT.WRFMT). For example, in the flow shown in **Figure 9.9-8**, YCbCr should not be specified as the WPF output format regardless of the fact that the color space of the WPF output image is in RGB format.

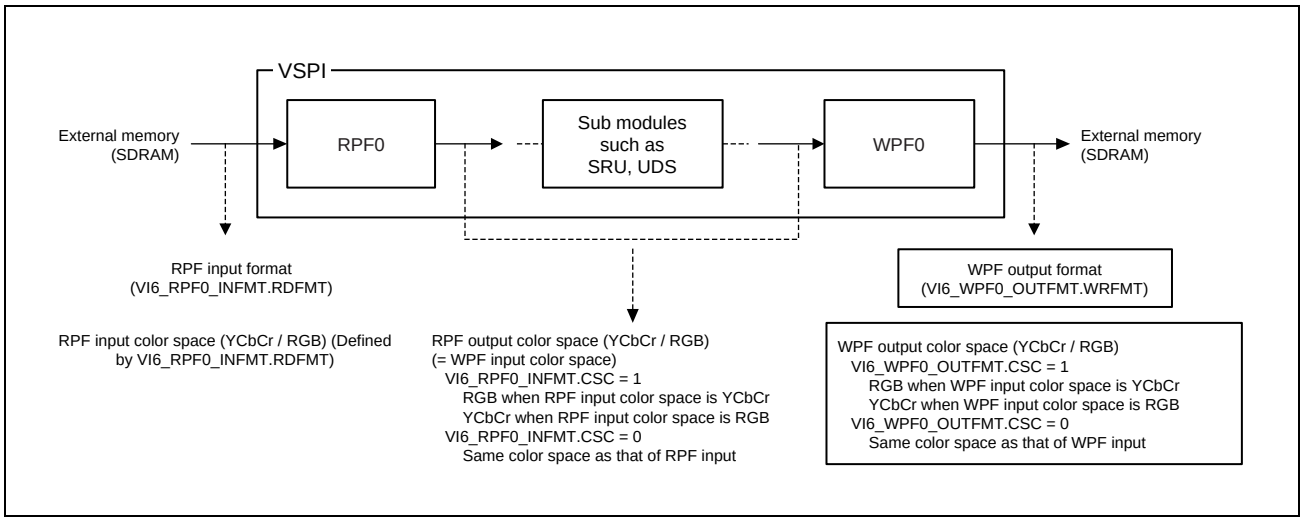


Figure 9.9-8 Relationship between Input/Output Format and Color Space

Table 9.9-7 Packed Formats for RPF Input (1/2)

RDFMT[6:0]	Bit per pixel	Phase	upper row - address / bottom row - bit field																													
			n								n+1								n+2								n+3							
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
00h	8	—	R0	R0	R0	G0	G0	G0	B0	B0	R1	R1	R1	G1	G1	G1	B1	B1	R2	R2	R2	G2	G2	G2	B2	B2	R3	R3	R3	G3	G3	G3	B3	B3
01h	12	—					R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0					R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
02h			R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0					R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1				
03h	—	—	Reserved								Reserved								Reserved								Reserved							
04h	15	—		R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0		R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
05h			R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0		R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	
06h	16	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
07h	18	—	A	A	A	A	A	A	A	A							R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
08h			R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0							A	A	A	A	A	A	A	A
09h									R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	A	A	A	A	A	A	A	A
0Ah			A	A	A	A	A	A	A	A	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0						
0Bh			A	A	A	A	A	A	A	A			R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0
0Ch					R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0	A	A	A	A	A	A	A	A
0Dh			A	A	A	A	A	A	A	A	R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0		
0Eh			R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0			A	A	A	A	A	A	A	A
0Fh		0							R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0							R1	R1
		1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1							R2	R2	R2	R2	R2	R2	G2	G2	G2	G2
		2	G2	G2	B2	B2	B2	B2	B2	B2							R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3
10h		0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0							R1	R1	R1	R1	R1	R1	G1	G1
		1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1							R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	B2	B2	B2	B2
		2	B2	B2							R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3						
11h		0			R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0			R1	R1	R1	R1	R1	R1
		1			G1	G1	G1	G1	G1	G1			B1	B1	B1	B1	B1	B1			R2	R2	R2	R2	R2	R2			G2	G2	G2	G2	G2	G2
		2			B2	B2	B2	B2	B2	B2			R3	R3	R3	R3	R3	R3			G3	G3	G3	G3	G3	G3			B3	B3	B3	B3	B3	B3
12h		0	R0	R0	R0	R0	R0	R0			G0	G0	G0	G0	G0	G0			B0	B0	B0	B0	B0	B0			R1	R1	R1	R1	R1	R1		
		1	G1	G1	G1	G1	G1	G1			B1	B1	B1	B1	B1	B1			R2	R2	R2	R2	R2	R2			G2	G2	G2	G2	G2	G2		
		2	B2	B2	B2	B2	B2	B2			R3	R3	R3	R3	R3	R3			G3	G3	G3	G3	G3	G3			B3	B3	B3	B3	B3	B3		
13h	24	—	A	A	A	A	A	A	A	A	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
14h			R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	A	A	A	A	A	A	A	A
15h		0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
		1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
		2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
16h	18	—								R0	R0	R0	R0	R0	R0	G0	G0	G0								G0	G0	G0	B0	B0	B0	B0	B0	B0
17h																	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
18h	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1
		1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2
		2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3
19h	12	—	A	A	A	A	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	A	A	A	A	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
1Ah		—	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	A	A	A	A	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	A	A	A	A
1Bh	15	—	A	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	A	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
1Ch		—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	A	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	A
1Dh	12	—	A	A	A	A	B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	A	A	A	A	B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1
1Eh		—	B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	A	A	A	A	B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1	A	A	A	A
1Fh	15	—	A	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	A	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1
20h		—	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	A	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	A
21h	18	0			B0	B0	B0	B0	B0	B0			G0	G0	G0	G0	G0	G0			R0	R0	R0	R0	R0	R0			B1	B1	B1	B1	B1	B1
		1			G1	G1	G1	G1	G1	G1			R1	R1	R1	R1	R1	R1			B2	B2	B2	B2	B2	B2			G2	G2	G2	G2	G2	G2
		2			R2	R2	R2	R2	R2	R2			B3	B3	B3	B3	B3	B3			G3	G3	G3	G3	G3	G3			R3	R3	R3	R3	R3	R3
22h	24	—	A	A	A	A	A	A	A	A	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0
23h	16	—																	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0

Table 9.9-7 Packed Formats for RPF Input (2/2)

RDFMT[6:0]	Bit per pixel	Phase	upper row - address / bottom row - bit field																									
			n								n+1								n+2								n+3							
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
24h to 3Eh	—	—	Reserved								Reserved								Reserved								Reserved							
3Fh	—	—	RGB_CLUT_DATA0								RGB_CLUT_DATA1								RGB_CLUT_DATA2								RGB_CLUT_DATA3							

When the RDFMT [6:0] bits are set to 3Fh, it is necessary to set up the OSD-CLUT in the RGB color space.

Table 9.9-8 Packed YCbCr Formats for RPF Input

RDFMT[6:0]	Packed YCbCr Input Format	Reference
40h	YCbCr4:4:4 semi-planar	**Figure 9.9-9***6
41h	YCbCr4:2:2 semi-planar (NV16, NV61*1)	
42h	YCbCr4:2:0 semi-planar (NV12, NV21*1)*7	
43h to 45h	Reserved	—
46h	YCbCr4:4:4 interleaved	**Figure 9.9-10***6
47h	YCbCr4:2:2 interleaved type 0 (UYVY, YUY2*2, YVYU*3)	
48h	YCbCr4:2:2 interleaved type 1	
49h	YCbCr4:2:0 interleaved*8	
4Ah	YCbCr4:4:4 planar	**Figure 9.9-11***6
4Bh	YCbCr4:2:2 planar (YV16)	
4Ch	YCbCr4:2:0 planar (YV12, I420)*7	
4Dh to 7Eh	Reserved	—
7Fh	YCBCR_CLUT_DATA*4,*5	—

Note 1. When the input format is NV61 or NV21, set the SPUVS bit to 1.

Note 2. When the input format is YUY2, set the SPYCS bit to 1.

Note 3. When the input format is YVYU, set the SPUVS bit to 1 and SPYCS bit to 1.

Note 4. When the RDFMT[6:0] bits are set to 7Fh, it is necessary to set up the OSD-CLUT in the YCbCr color space.

Note 5. The Monochrome format can be implemented through the OSD-CLUT.

Note 6. **Figure 9.9-12** shows the definition of memory address for each pixel in **Figure 9.9-9** to **Figure 9.9-11**.

Note 7. Each line of C plane is read twice, so byte/pixel of YCbCr420 is 2 byte/pixel (same as YCbCr422).

Note 8. Each line of plane is read twice, so byte/pixel of YCbCr420ITL is 3 byte/pixel (same as YCbCr444ITL).

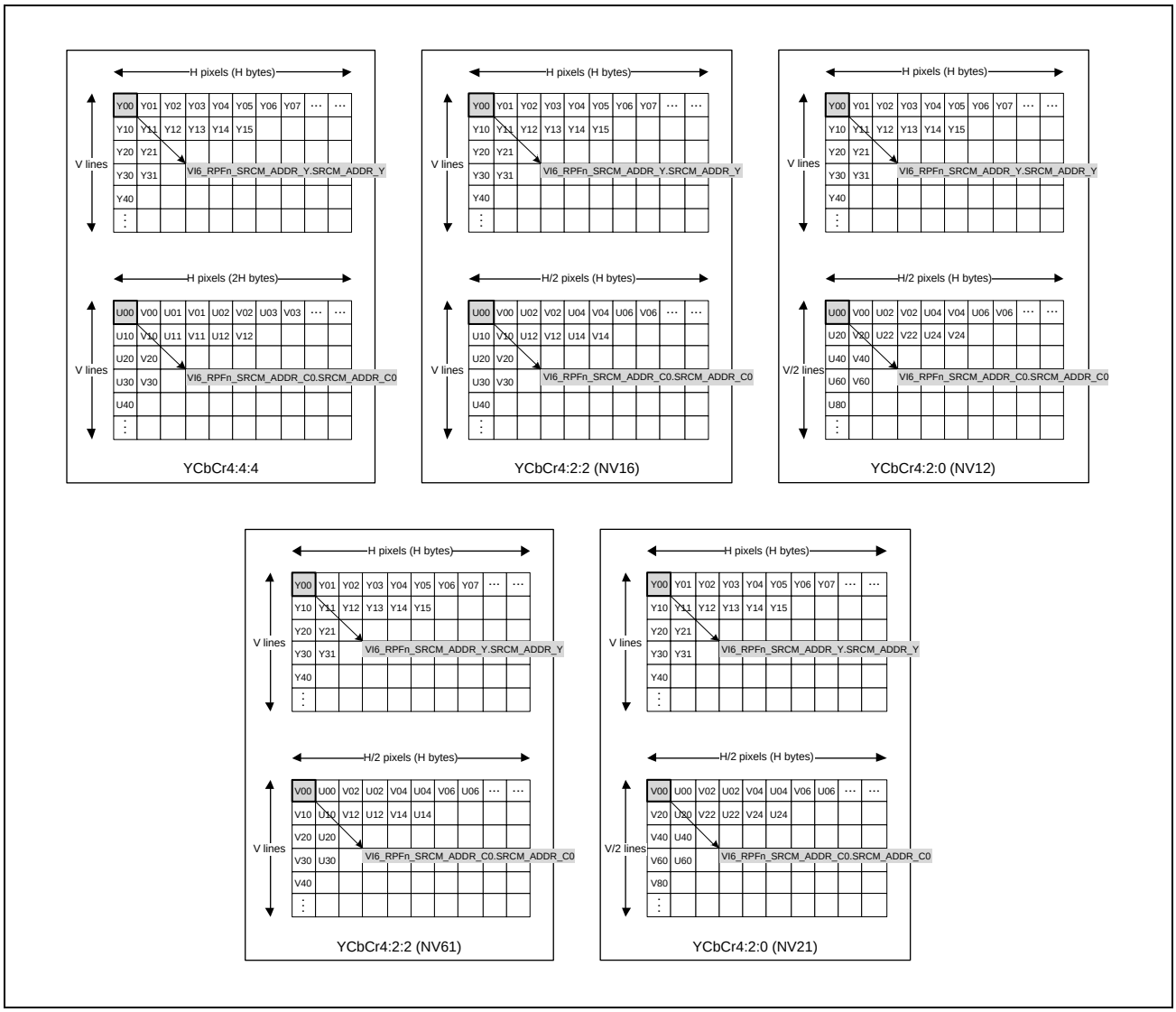


Figure 9.9-9 YCbCr Semi-Planar Formats

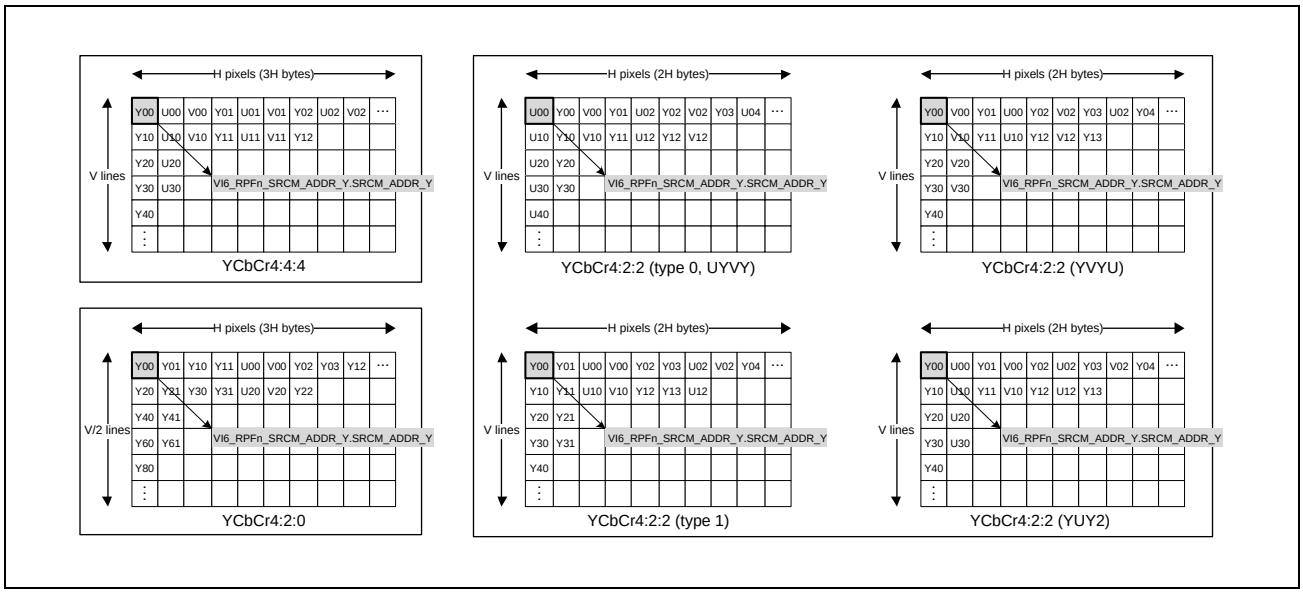


Figure 9.9-10 YCbCr Interleaved Formats

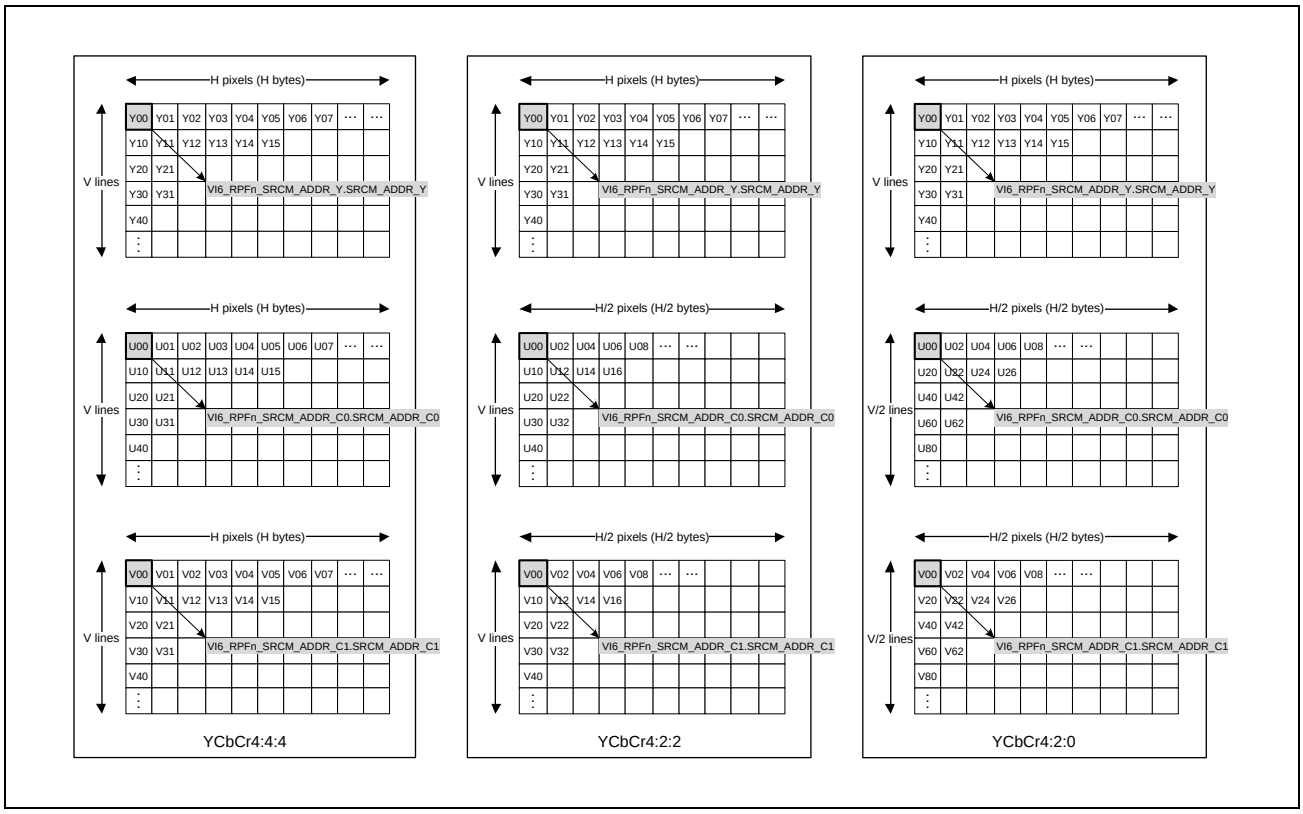


Figure 9.9-11 YCbCr Planar Formats

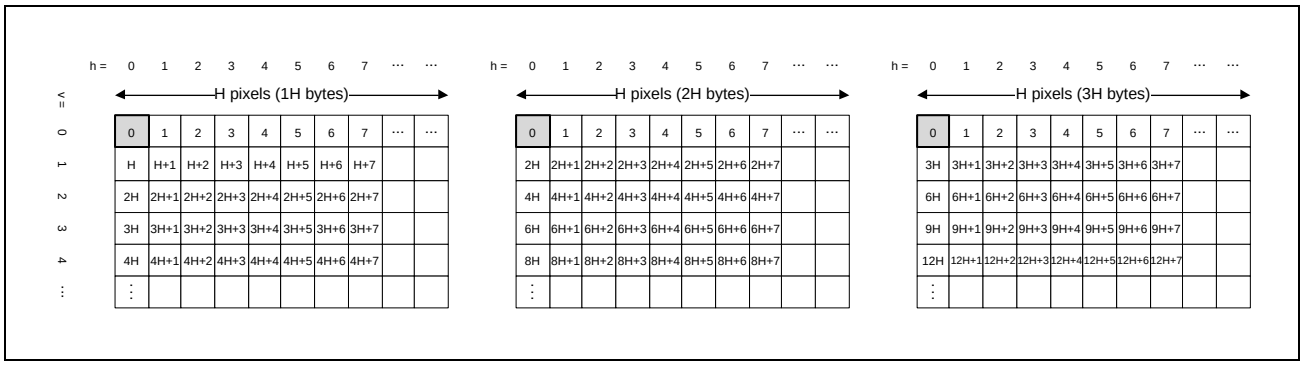


Figure 9.9-12 Memory Address for the Corresponding Pixel's Position

Legend of **Figure 9.9-12**

- H is horizontal image size in pixel unit.
- 1H, 2H and 3H is corresponding to the case 1 pixel has 1 byte, 2 bytes, and 3 bytes of data respectively.

[OSD-CLUT Setting]

To set up the OSD-CLUT, store the values to replace each of R (Cr), G (Y), and B (Cb) components in the OSD-CLUT allocated as shown in **Table 9.9-9**. The OSD-CLUT space has 256 32-bit entries, which should be accessed in 32 bits in the same way as for register access. The address for each entry is obtained from the start address of each space (**Table 9.9-9**) + entry number × 4. For example, the OSD-CLUT address of entry 7 for RPF0 is (offset address) + 401Ch. **Figure 9.9-13** shows the OSD-CLUT format.

Table 9.9-9 OSD-CLUT Space for Each RPF

RPF	Address Area for CLUT Space
RPF0	(offset address) + 4000h to (offset address) + 43FFh

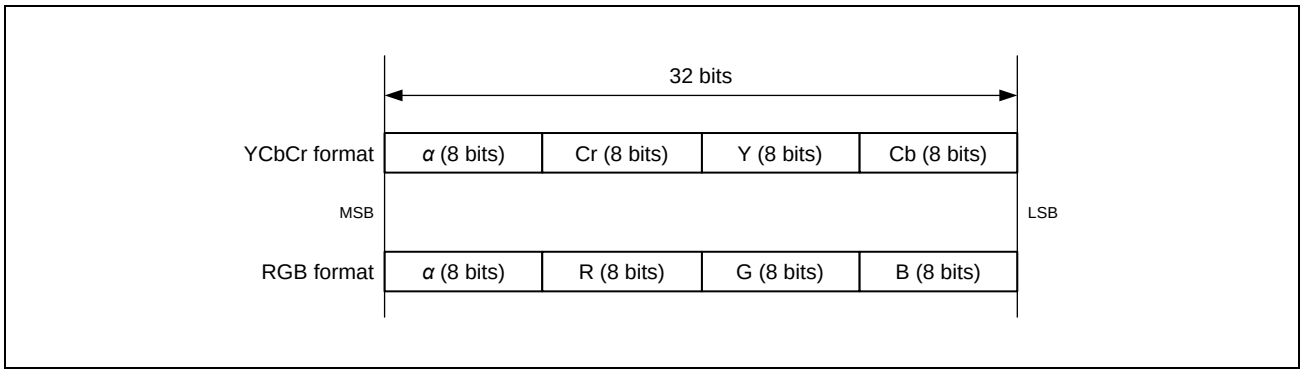


Figure 9.9-13 OSD-CLUT Format

According to the specified 256-entry replacement table (OSD-CLUT), input data is replaced as shown in **Figure 9.9-14**. Input data is 8 bits but output data is 32 bits including α and RGB or YCbCr components.

When writing to an entry in the OSD-CLUT, the host CPU writes all components at the same time. Therefore, specify replacement data for all components in the OSD-CLUT at the same time as shown in **Figure 9.9-13**.

Note that write access to the OSD-CLUT space is prohibited while the RPF is operating.

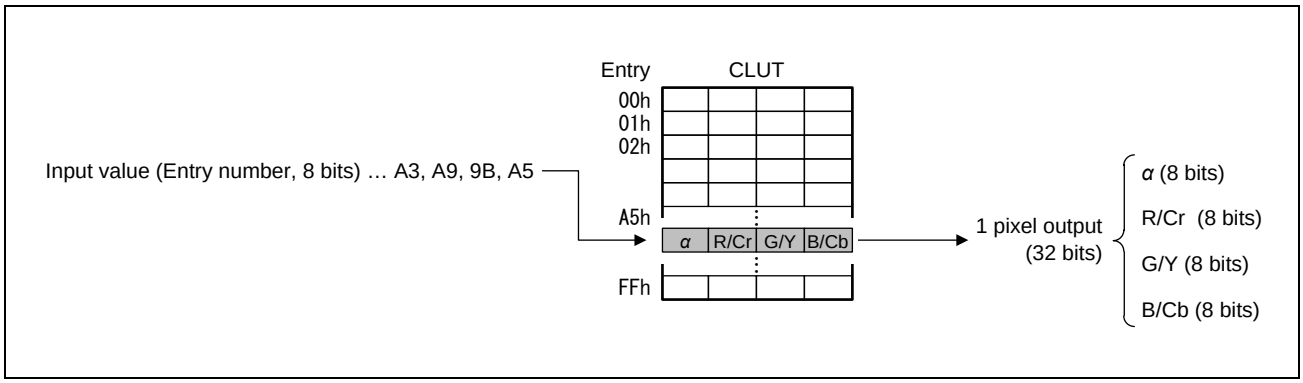


Figure 9.9-14 Replacement of Input Data with Pixel Values Specified in OSD-CLUT

(4) RPF0 Data Swapping Registers (VI6_RPF0_DSWAP)

Access Size : 32 bits
Address : <VSPI0_base> + 030Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	A_LLS	A_LWS	A_WDS	A_BTS	-	-	-	-	P_LLS	P_LWS	P_WDS	P_BTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
11	A_LLS	0h	RW	α Plane Data Swapping in LONG LWORD Units The effect of this bit setting is defined in **Table 9.9-10**. 0: Data swapping in LONG LWORD (64-bit) units is disabled 1: Data swapping in LONG LWORD (64-bit) units is enabled
10	A_LWS	0h	RW	α Plane Data Swapping in long word Units The effect of this bit setting is defined in **Table 9.9-10**. 0: Data swapping in long word (32-bit) units is disabled 1: Data swapping in long word (32-bit) units is enabled
9	A_WDS	0h	RW	α Plane Data Swapping in Word Units The effect of this bit setting is defined in **Table 9.9-10**. 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled
8	A_BTS	0h	RW	α Plane Data Swapping in Byte Units The effect of this bit setting is defined in **Table 9.9-10**. 0: Data swapping in byte (8-bit) units is disabled 1: Data swapping in byte (8-bit) units is enabled
7 to 4	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
3	P_LLS	0h	RW	Picture Plane Data Swapping in LONG LWORD Units The effect of this bit setting is defined in **Table 9.9-10**. 0: Data swapping in LONG LWORD (64-bit) units is disabled 1: Data swapping in LONG LWORD (64-bit) units is enabled
2	P_LWS	0h	RW	Picture Plane Data Swapping in long word Units The effect of this bit setting is defined in **Table 9.9-10**. 0: Data swapping in long word (32-bit) units is disabled 1: Data swapping in long word (32-bit) units is enabled
1	P_WDS	0h	RW	Picture Plane Data Swapping in Word Units The effect of this bit setting is defined in **Table 9.9-10**. 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled
0	P_BTS	0h	RW	Picture Plane Data Swapping in Word Units The effect of this bit setting is defined in **Table 9.9-10**. 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled

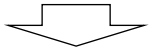
When the virtual input function of the RPF0 is used (VI6_RPF0_INFMT.VIR = 1), this register setting is ignored. Swapping of RPF input data can be specified separately for the α plane and picture plane.

Table 9.9-10 shows the data order before and after swapping according to the long word, word, half word, and byte swapping settings.

When data order in memory for each format is the same as **Table 9.9-7** for RGB format and **Figure 9.9-9** to **Figure 9.9-11** for YCbCr format, set 1111b to {*_LLS, *_LWS, *_WDS, *_BTS}. If data order is not the same as the definition, change data order within 16-byte unit by these bits as shown in **Table 9.9-10**.

Table 9.9-10 Changing Data Order According to Swap Register

Data Order in Memory																	*_LLS	*_LWS	*_WDS	*_BTS
Byte Address	16n+0	16n+1	16n+2	16n+3	16n+4	16n+5	16n+6	16n+7	16n+8	16n+9	16n+10	16n+11	16n+12	16n+13	16n+14	16n+15				
Data	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	1	1	1	1
	1	0	3	2	5	4	7	6	9	8	11	10	13	12	15	14	1	1	1	0
	2	3	0	1	6	7	4	5	10	11	8	9	14	15	12	13	1	1	0	1
	3	2	1	0	7	6	5	4	11	10	9	8	15	14	13	12	1	1	0	0
	4	5	6	7	0	1	2	3	12	13	14	15	8	9	10	11	1	0	1	1
	5	4	7	6	1	0	3	2	13	12	15	14	9	8	11	10	1	0	1	0
	6	7	4	5	2	3	0	1	14	15	12	13	10	11	8	9	1	0	0	1
	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	1	0	0	0
	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	0	1	1	1
	9	8	11	10	13	12	15	14	1	0	3	2	5	4	7	6	0	1	1	0
	10	11	8	9	14	15	12	13	2	3	0	1	6	7	4	5	0	1	0	1
	11	10	9	8	15	14	13	12	3	2	1	0	7	6	5	4	0	1	0	0
	12	13	14	15	8	9	10	11	4	5	6	7	0	1	2	3	0	0	1	1
	13	12	15	14	9	8	11	10	5	4	7	6	1	0	3	2	0	0	1	0
	14	15	12	13	10	11	8	9	6	7	4	5	2	3	0	1	0	0	0	1
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	0	0	0	0



Data Order Defined in **Table 9.9-7** for RGB Format and **Figure 9.9-9** to **Figure 9.9-11** for YCbCr Format																
Byte Address	16n+0	16n+1	16n+2	16n+3	16n+4	16n+5	16n+6	16n+7	16n+8	16n+9	16n+10	16n+11	16n+12	16n+13	16n+14	16n+15
Data	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

(5) RPF0 Display Location Registers (VI6_RPF0_LOC)

Access Size : 32 bits

Address : <VSPI0_base> + 0310h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	HCOORD[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VCOORD[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28 to 16	HCOORD[12:0]	0h	RW	Horizontal Coordinate of Sublayer Display Location on Master Layer These bits specify the left-end location of the sublayer displayed by the RPF0 and the subsequent module connected through the DPR. Specify the horizontal coordinate of the location in pixel units with the left-end pixel of the master layer set at coordinate 0. When the RPF0 is the master layer, set these bits to 0. If the sublayer extends beyond the master layer according to the HCOORD setting, the extended section is cut off at the right end of the master layer. Even in this case, however, a bus transfer that is unnecessary for output image generation is executed since the whole sublayer data is read from the external memory. Appropriate coordinate setting is required so that the sublayer does not extend beyond the right end of the master layer. A value from 0 to 8189 can be specified.
15 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12 to 0	VCOORD[12:0]	0h	RW	Vertical Coordinate of Sublayer Display Location on Master Layer These bits specify the top-end location of the sublayer displayed by the RPF0 and the subsequent module connected through the DPR. Specify the vertical coordinate of the location in pixel units with the top-end pixel of the master layer set at coordinate 0. When the RPF0 is the master layer, set these bits to 0. If the sublayer extends beyond the master layer according to the VCOORD setting, the extended section is cut off at the bottom end of the master layer. Even in this case, however, a bus transfer that is unnecessary for output image generation is executed since the whole sublayer data is read from the external memory. Appropriate coordinate setting is required so that the sublayer does not extend beyond the bottom end of the master layer. A value from 0 to 8189 can be specified.

(6) RPF0 α Plane Selection Control Registers (VI6_RPF0_ALPH_SEL)

Access Size : 32 bits
Address : <VSPI0_base> + 0314h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	ASEL[2:0]			IROP[3:0]				BSEL	-	-	-	AEXT[1:0]		-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	RW	RW	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALPHA1[7:0]								ALPHA0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.
30 to 28	ASEL[2:0]	0h	RW	α Format and Processing Method Select These bits select how to handle the α value to be used. The RPF handles two types of α value; 8-bit and 1-bit values. When a 1-bit α value is used, VSPI assumes that the 1-bpp α value for each pixel is stored in the order from MSB to LSB in each byte (big endian). The α value is used as either transparency information or mask information. Transparency information is included in the α plane read from external memory when the ASEL bits are set to 1 or 3 and in the α value stored in the packed RGB bit field when these bits are set to 0 or 2. The α value as transparency information is sent as the destination value to the IROP as shown in **Figure 9.9-15** and then output to the subsequent modules. The α value as mask information is used for IROP operation in the RPF. The mask information is included in the α plane read from external RAM when the ASEL bits are set to 0 or 2 and the source value is used in IROP operation (IROP setting other than 0, 5, 10, or 15). This α value is sent as the source value to the IROP as shown in **Figure 9.9-15**. Note that the α value selected through the ASEL bits has a lower priority than the VI6_RPF0_CKEY_SET*.AP* value replaced through the color keying function. When the color keying function is used, the α value may be replaced with the VI6_RPF0_CKEY_SET*.AP* value regardless of the ASEL bit setting. When using the virtual input function (VI6_RPF0_INFMT.VIR = 1), specify 4. 000b: 1, 4, or 8-bit packed α + plane α (IROP ! = 0, 5, 10, 15) The α bit field in 1, 4, or 8-bit packed α is handled as transparency information. Be sure to specify the packed format that includes α through VI6_RPF0_INFMT.RDFMT. When VI6_RPF0_MSKCTRL.MSK_EN is 0 and the IROP bit value is not 0, 5, 10, or 15, the α plane should be read as mask information. Specify the number of α data bits (BSEL) stored in the α plane and the α plane read start address (VI6_RPF0_SRCM_ADDR_AI). When the IROP bits are set to 0, 5, 10, or 15, the α plane is not read. 001b: 8-bit plane α The 8-bit α plane is read from external RAM as transparency information. When the packed RGB format has a bit field for α, the information in the α bit field is discarded. The α plane read start address (VI6_RPF0_SRCM_ADDR_AI) should be specified. The α value goes through the 8-bit transparent α generator shown in Figure 9.9-15 without change. When VI6_RPF0_MSKCTRL.MSK_EN is 0, IROP operation cannot be executed; set the IROP bits to 0 in this case. When VI6_RPF0_MSKCTRL.MSK_EN is 1. IROP operation can be executed.

Bit	Bit Name	Initial Value	R/W	Description
30 to 28	ASEL[2:0]	0h	RW	010b: 1-bit packed α + plane α (IROP ! = 0, 5, 10, 15) The 1-bit packed α input is converted by the 8-bit transparent α generator shown in **Figure 9.9-15** according to the ALPHA0/1 setting into the 8-bit α value as transparency information. Select the packed input format that includes a 1-bit α field. When VI6_RPF0_MSKCTRL.MSK_EN is 0 and the IROP value is not 0, 5, 10, or 15, the α plane should be read as mask information. Specify the number of α data bits (BSEL) stored in the α plane and the α plane read start address (VI6_RPF0_SRCM_ADDR_AI). When the IROP bits are set to 0, 5, 10, or 15, the α plane is not read. 011b: 1-bit plane α The 1-bit α plane is read from external RAM and converted by the 8-bit transparent α generator shown in **Figure 9.9-15** according to the ALPHA0/1 setting into the 8-bit α value as transparency information. When the packed RGB format has a bit field for α, the information in the α bit field is discarded. The α plane read start address (VI6_RPF0_SRCM_ADDR_AI) should be specified. When VI6_RPF0_MSKCTRL.MSK_EN is 0, IROP operation cannot be executed; set the IROP bits to 0 in this case. When VI6_RPF0_MSKCTRL.MSK_EN is 1. IROP operation can be executed. 100b: Fixed α The fixed α value (VI6_RPF0_VRTCOL_SET.LAYA value) is output from the RPF. IROP operation cannot be executed; set the IROP bits to 0 in this case. 101b to 111b: Setting prohibited.
27 to 24	IROP[3:0]	0h	RW	IROP Operation Setting These bits specify the operator to be executed in the IROP operation unit shown in **Figure 9.9-15**. The source (S) for the IROP operation is the pixel data and α data specified in the VI6_RPF0_MSKSET0 or VI6_RPF0_MSKSET1 IROP input value register, which is selected according to the value (0 or 1) generated by the 1-bit mask generator. The destination (D) is the image data (RGB/YCbCr) and 8-bit α data output from the unpack/OSD processor. IROP operation is applied both for the image data and α data between the source and destination data. If these bits are set to the operation that involves the source (S) (IROP setting other than 0, 5, 10, or 15) while VI6_RPF0_MSKCTRL.MSK_EN is 0, the α plane is read from the external RAM to be used for the α value for IROP operation; specify the α plane read start address (VI6_RPF0_SRCM_ADDR_AI). When the virtual input function is used (VI6_RPF0_INFMT.VIR = 1), IROP operation is not available; set these bits to 0000b. 0000b: NOP(D) 0001b: AND(S & D) 0010b: AND_REVERSE(S & ~D) 0011b: COPY(S) 0100b: AND_INVERTED(~S & D) 0101b: CLEAR(0) 0110b: XOR(S ^ D) 0111b: OR(S | D) 1000b: NOR(~(S | D)) 1001b: EQUIV(~(S ^ D)) 1010b: INVERT(~D) 1011b: OR_REVERSE(S | ~D) 1100b: COPY_INVERTED(~S) 1101b: OR_INVERTED(~S | D) 1110b: NAND(~(S & D)) 1111b: SET(all 1)
23	BSEL	0h	RW	α Bit Count Conversion Selection for 1-Bit Mask Generator Specifies the number of bits in the α plane to be read as mask information from the external RAM. When α plane data is 8 bits, it is converted to one bit through the 1-bit mask generator shown in **Figure 9.9-15**. Note that this bit setting is valid when the ASEL bits are set to 0 or 2 and VI6_RPF0_MSKCTRL.MSK_EN is set to 0. In other cases, this bit setting has no effect. 0: 8-bit α is converted to 1-bit α through the 1-bit mask generator. When the 8-bit α value input to the RPF is not 0, it is converted to 1b; when the value is 0, it is converted to 0b. 1: α value goes through the 1-bit mask generator. The 1-bit α value input to the RPF is output through the 1-bit mask generator without change.

Bit	Bit Name	Initial Value	R/W	Description
22 to 20	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
19, 18	AEXT[1:0]	0h	RW	Lower-Bit α Value Extension Method Set These bits specify the method for extending the input α data to 8 bits through the unpack processing. 00b: The lower-order bits of α value are extended with 0. 01b: The upper-order bits of α value are copied to the lower-order bits. 10b: The lower-order bits of α value are extended with 0. The maximum value is limited to FFh. 11b: Setting prohibited
17, 16	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
15 to 8	ALPHA1[7:0]	0h	RW	8-Bit α Value Output when 1-Bit α Value is 1 These bits specify the 8-bit α value to be output when 1-bit α data is input and the α value input to the 8-bit transparent α generator shown in **Figure 9.9-15** is 1b. This setting is valid when the ASEL bits are set to 010b or 011b. A value from 0 to 255 can be specified.
7 to 0	ALPHA0[7:0]	0h	RW	8-Bit α Value Output when 1-Bit α Value is 0 These bits specify the 8-bit α value to be output when 1-bit α data is input and the α value input to the 8-bit transparent α generator shown in **Figure 9.9-15** is 0b. This setting is valid when the ASEL bits are set to 010b or 011b. A value from 0 to 255 can be specified.

Figure 9.9-15 shows the relationship between the α selector, IROP operation unit, color keying unit, and related registers. The IROP operation unit receives two inputs, source and destination. The image data input from the external memory is processed through the unpack processor and 8-bit transparent α generator and then input to the IROP operation unit as destination data. The α plane data input from the external memory is sent to the 8-bit transparent α generator when the ASEL bits are set to 1 or 3, or sent to the 1-bit mask α generator when the ASEL bits are set to 0 or 2. For the pixel data and 8-bit α value on the source side of the IROP operation unit, either the VI6_RPF0_MSKSET0 value or VI6_RPF0_MSKSET1 values will be selected according to the 1-bit α value output by the 1-bit mask α generator.

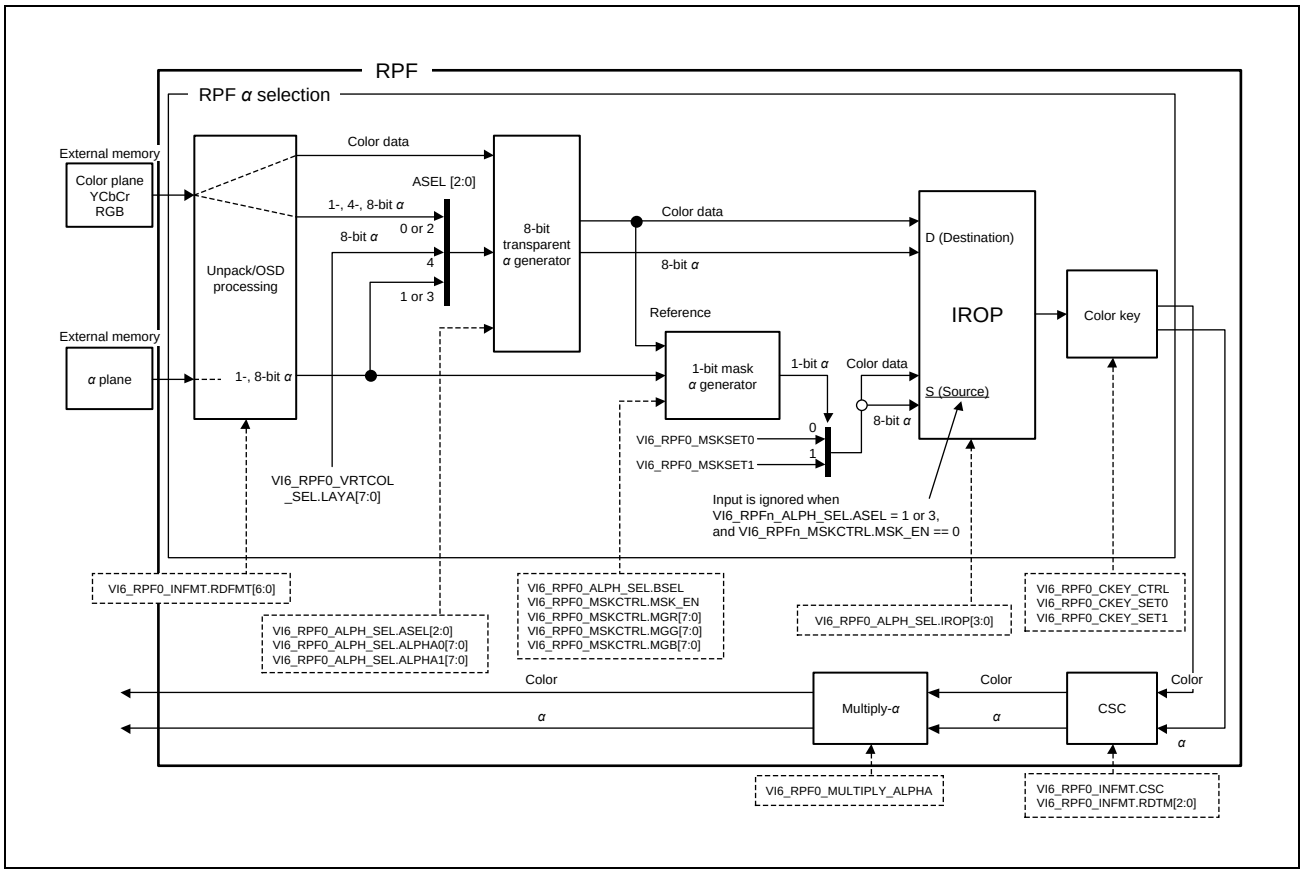


Figure 9.9-15 Configuration of Alpha Selector and IROP Operation Unit in RPF

The following describes the function of each block shown in **Figure 9.9-15**. Read the following while referring to the figure as necessary.

Unpack/OSD processor:

Unpacks each component and α value of the image data according to the packed format specified in VI6_RPF0_INFMT.RDFMT.

8-bit transparent α generator:

Converts the input α value into 8-bit α when the input α is four bits or one bit.

When VI6_RPF0_ALPH_SEL.ASEL are set to 0 (8-, 4-, or 1-bit packed α), this generator outputs the input α value without change if the α bit field in the packed α data is 8 bits; if the α bit field is less than 8 bits, it is converted to an 8-bit α value by extending the LSB side according to the VI6_RPF0_ALPH_SEL.AEXT setting.

When VI6_RPF0_ALPH_SEL.ASEL are set to 1 (8-bit plane α) or 4 (fixed α), this generator outputs the input 8-bit plane α without change. If a packed α value is included in RGB data, it is discarded.

When VI6_RPF0_ALPH_SEL.ASEL are set to 2 (1-bit packed α) or 3 (8-bit α generated from 1-bit plane α), an 8-bit α value is generated by using the VI6_RPF0_ALPH_SEL.ALPHA0 [7:0] value when the input 1-bit α value is 0 or by using the VI6_RPF0_ALPH_SEL.ALPHA1 [7:0] value when the input 1-bit α value is 1. When VI6_RPF0_ALPH_SEL.ASEL is set to 3, a packed α value that is included in RGB data is discarded.

1-bit mask α generator:

Generates 1-bit α data from the input 8-bit α data or pixel data. When the input α data is one bit, this generator outputs it without change.

When VI6_RPF0_ALPH_SEL.ASEL are set to 0 (8-, 4-, or 1-bit packed α) or 2 (plane α) and VI6_RPF0_MSKCTRL.MSK_EN is set to 0, the α plane read from the external memory to be used in IROP is converted to 1-bit α data when necessary. When the α plane data read from the external RAM is 8 bits (BSEL = 0), if the value is 0, a 1-bit α value of 0b is generated; if the value is not 0, a 1-bit α value of 1b is generated. When the α plane data is one bit (BSEL = 1), this generator outputs it without change.

When the value of the 1-bit α generated by the 1-bit mask α generator is 0b, the 8-bit α and pixel data specified in VI6_RPF0_MSKSET0 are output as the source. When the generated 1-bit α value is 1b, the 8-bit α and pixel data specified in VI6_RPF0_MSKSET1 are output as the source.

As shown in **Figure 9.9-15**, when VI6_RPF0_ALPH_SEL.ASEL are set to 1 (8-bit plane α) or 3 (1-bit plane α), the α plane read from the external RAM is sent to the 8-bit transparent α generator as transparency information. When VI6_RPF0_MSKCTRL.MSK_EN is set to 0, the 1-bit α for masking is generated according to the input α plane (refer to **9.9.1.3.6 (8) RPF0 Mask Control Registers (VI6_RPF0_MSKCTRL)**), but the 1-bit mask α generator does not refer to the input α plane because it is input to the 8-bit transparent α generator as transparency information. Accordingly, the 1-bit mask α generator does not generate a 1-bit α for masking and the data on the source side becomes invalid; that is, IROP operation cannot be executed. Set the IROP bits to 0 in this case. In contrast, when VI6_RPF0_MSKCTRL.MSK_EN is set to 1, the 1-bit mask α generator creates α data for masking according to the pixel data instead of the input α plane data, and IROP operation can be executed in this case.

IROP operation unit:

Executes ROP operation according to the opcode specified in VI6_RPF0_ALPH_SEL.IROP. For ROP operation (other than NOP), valid values should be input both for the source and destination. As described in the above (description of the 1-bit mask α generator), when VI6_RPF0_ALPH_SEL.ASEL are set to 1 or 3 and VI6_RPF0_MSKCTRL.MSK_EN is set to 0, the source data for the IROP operation unit is treated as invalid; set VI6_RPF0_ALPH_SEL.IROP to 0 (NOP). When VI6_RPF0_ALPH_SEL.ASEL are set to 4, a fixed α value is output from the RPF and IROP operation is not available. In the same way as the above case, set VI6_RPF0_ALPH_SEL.IROP to 0 (NOP).

To specify a valid source value for the IROP operation unit and execute IROP operation (specify an opcode other than NOP in the IROP bits), specify register values as shown in **Table 9.9-11**. Where the source input state is indicated as “Valid” in the table, IROP operation can be executed. In the cases where IROP operation is not available, set the IROP bits to 0 (NOP).

Table 9.9-11 Source Input State in IROP Operation Unit

VI6_RPF0_ALPH_SEL.ASEL [2:0]		VI6_RPF0_MSKCTRL.MSK_EN		
		0 (Source data is generated according to input α plane)		1 (Source data is generated according to the destination-side pixel data)
000b	(1-, 4-, or 8-bit packed α + plane α)	Valid	(α plane input)	Valid
001b	(8-bit α plane)	Invalid	(IROP operation is not available; α plane is output to the subsequent modules behind RPF)	Valid
010b	(8-bit α generated from 1-bit packed α + plane α)	Valid	(α plane input)	Valid
011b	(8-bit α generated from 1-bit plane α)	Invalid	(IROP operation is not available; α plane is output to the subsequent modules behind RPF)	Valid
100b	(Fixed α)	Invalid (IROP operation is not available; fixed α is output to the subsequent modules behind RPF)		

For the handling of the α values shown in **Figure 9.9-15** and **Table 9.9-11**, the relationship between the RPF input format and RPF output α value is shown in **Table 9.9-12**. Where only bit names are shown in the table, the bits are in VI6_RPF0_ALPH_SEL described in this section.

Table 9.9-12 α Value Selected and Output According to ASEL Bits in Each Input Format

ASEL Setting		α Value Output for Each Input Format		
		RGB	YCbCr	OSD-CLUT
000b	(8-, 4-, or 1-bit packed α is input)	1-, 4-, or 8-bit pixel α	FFh*1	α value in CLUT
001b	(8-bit plane α is input)	8-bit α plane	8-bit α plane	8-bit α plane
010b	(8-bit α is generated from the 1-bit packed α input)	ALPHA0 or ALPHA1 setting	FFh*1	FFh
011b	(8-bit α is generated from the 1-bit plane α input)	ALPHA0 or ALPHA1 setting	ALPHA0 or ALPHA1 setting	ALPHA0 or ALPHA1 setting
100b	(Fixed α is output)	VI6_RPF0_VRTCOL_SET.LAYA setting		

Note 1. Fixed value FFh is output because packed α is not included in YCbCr.

(7) RPF0 Virtual Plane Color Information Registers (VI6_RPF0_VRTCOL_SET)

Access Size : 32 bits

Address : <VSPI0_base> + 0318h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LAYA[7:0]								LAYR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LAYG[7:0]								LAYB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	LAYA[7:0]	0h	RW	Virtual-Input Fixed α Value These bits specify the fixed α value generated in the RPF0 when the virtual input function is enabled through the VI6_RPF0_INFMT.VIR setting. When the virtual input function is disabled (VI6_RPF0_INFMT.VIR = 0), these bits are used to specify the fixed α value to be output from the RPF while VI6_RPF0_ALPH_SEL.ASEL are set to 4. A value from 0 to 255 can be specified.
23 to 16	LAYR[7:0]	0h	RW	Virtual-Input Fixed R/Cr Component Value These bits specify the fixed R or Cr value generated in the RPF0 when the virtual input function is enabled through the VI6_RPF0_INFMT.VIR setting. When VI6_RPF0_INFMT.RDFMT is set to 19, the value specified in these bits is used as the R value. When VI6_RPF0_INFMT.RDFMT is set to 64, the value specified in these bits is used as the Cr value. A value from 0 to 255 can be specified.
15 to 8	LAYG[7:0]	0h	RW	Virtual-Input Fixed G/Y Component Value These bits specify the fixed G or Y value generated in the RPF0 when the virtual input function is enabled through the VI6_RPF0_INFMT.VIR setting. When VI6_RPF0_INFMT.RDFM T is set to 19, the value specified in these bits is used as the G value. When VI6_RPF0_INFMT.RDFMT is set to 64, the value specified in these bits is used as the Y value. A value from 0 to 255 can be specified.
7 to 0	LAYB[7:0]	0h	RW	Virtual-Input Fixed B/Cb Component Value These bits specify the fixed B or Cb value generated in the RPF0 when the virtual input function is enabled through the VI6_RPF0_INFMT.VIR setting. When VI6_RPF0_INFMT.RDFMT is set to 19, the value specified in these bits is used as the B value. When VI6_RPF0_INFMT.RDFMT is set to 64, the value specified in these bits is used as the Cb value. A value from 0 to 255 can be specified.

(8) RPF0 Mask Control Registers (VI6_RPF0_MSKCTRL)

Access Size : 32 bits
Address : <VSPI0_base> + 031Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	MSK_EN	MGR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MGG[7:0]								MGB[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
24	MSK_EN	0h	RW	Mask Generation Specification Specifies the method of α value generation in the 1-bit mask α generator shown in **Figure 9.9-15**. 0b: A 1-bit mask value is generated according to the input α plane value. When the input α is in the 1-bit format (VI6_RPF0_ALPH_SEL.BSEL = 1), the 1-bit mask value is output without change. When the input α is in the 8-bit format (VI6_RPF0_ALPH_SEL.BSEL = 0), the 1-bit mask value is 0 if the α value is 00h; otherwise, the 1-bit mask value is 1. 1b: The R/Cr, G/Y, and B/Cb components of the image input to the destination side of the IROP operation unit are compared with the values specified in the MGR, MGG, and MGB bits, respectively. When all values match, 1 is output as the 1-bit mask value, and in other cases, 0 is output. When the generated 1-bit mask data is not used, set VI6_RPF0_ALPH_SEL.IROP to 0.
23 to 16	MGR[7:0]	0h	RW	R/Cr Comparison Value for 1-Bit α Generation These bits specify the R/Cr value to be compared for 1-bit α generation by using the pixel data on the destination side. When RGB is specified as the RPF0 input format through VI6_RPF0_INFMT.RDFMT, specify an R value for comparison. When YCbCr is specified, specify a Cr value for comparison. This setting is ignored when the MSK_EN bit is set 0. A value from 0 to 255 can be specified.
15 to 8	MGG[7:0]	0h	RW	G/Y Comparison Value for 1-Bit α Generation These bits specify the G/Y value to be compared for 1-bit α generation by using the pixel data on the destination side. When RGB is specified as the RPF0 input format through VI6_RPF0_INFMT.RDFMT, specify a G value for comparison. When YCbCr is specified, specify a Y value for comparison. This setting is ignored when MSK_EN is set to 0. A value from 0 to 255 can be specified.
7 to 0	MGB[7:0]	0h	RW	B/Cb Comparison Value for 1-Bit α Generation These bits specify the B/Cb value to be compared for 1-bit α generation by using the pixel data on the destination side. When RGB is specified as the RPF0 input format through VI6_RPF0_INFMT.RDFMT, specify a B value for comparison. When YCbCr is specified, specify a Cb value for comparison. This setting is ignored when MSK_EN is set to 0. A value from 0 to 255 can be specified.

(9) RPF0 IROP-SRC Input Value Registers 0 (VI6_RPF0_MSKSET0)

Access Size : 32 bits
Address : <VSPI0_base> + 0320h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSA0[7:0]								MSR0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSG0[7:0]								MSB0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	MSA0[7:0]	0h	RW	IROP-Source Input α Value when 1-Bit α is 0 These bits specify the 8-bit α value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (**Figure 9.9-15**). A value from 0 to 255 can be specified.
23 to 16	MSR0[7:0]	0h	RW	IROP-Source Input R/Cr Value when 1-Bit α is 0 These bits specify the R/Cr value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (**Figure 9.9-15**). When RGB is specified as the RPF0 input format through VI6_RPF0_INFMT.RDFMT, specify an R component value. When YCbCr is specified, specify a Cr component value. A value from 0 to 255 can be specified.
15 to 8	MSG0[7:0]	0h	RW	IROP-Source Input G/Y Value when 1-Bit α is 0 These bits specify the G/Y value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (**Figure 9.9-15**). When RGB is specified as the RPF0 input format through VI6_RPF0_INFMT.RDFMT, specify a G component value. When YCbCr is specified, specify a Y component value. A value from 0 to 255 can be specified.
7 to 0	MSB0[7:0]	0h	RW	IROP-Source Input B/Cb Value when 1-Bit α is 0 These bits specify the B/Cb value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (**Figure 9.9-15**). When RGB is specified as the RPF0 input format through VI6_RPF0_INFMT.RDFMT, specify a B component value. When YCbCr is specified, specify a Cb component value. A value from 0 to 255 can be specified.

(10) RPF0 IROP-SRC Input Value Registers 1 (VI6_RPF0_MSKSET1)

Access Size : 32 bits
Address : <VSPI0_base> + 0324h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSA1[7:0]								MSR1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSG1[7:0]								MSB1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	MSA1[7:0]	0h	RW	IROP-Source Input α Value when 1-Bit α is 1 These bits specify the 8-bit α value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1. A value from 0 to 255 can be specified (**Figure 9.9-15**).
23 to 16	MSR1[7:0]	0h	RW	IROP-Source Input R/Cr Value when 1-Bit α is 1 These bits specify the R/Cr value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1 (**Figure 9.9-15**). When RGB is specified as the RPF0 input format through VI6_RPF0_INFMT.RDFMT, specify an R component value. When YCbCr is specified, specify a Cr component value. A value from 0 to 255 can be specified.
15 to 8	MSG1[7:0]	0h	RW	IROP-Source Input G/Y Value when 1-Bit α is 1 These bits specify the G/Y value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1 (**Figure 9.9-15**). When RGB is specified as the RPF0 input format through VI6_RPF0_INFMT.RDFMT, specify a G component value. When YCbCr is specified, specify a Y component value. A value from 0 to 255 can be specified.
7 to 0	MSB1[7:0]	0h	RW	IROP-Source Input B/Cb Value when 1-Bit α is 1 These bits specify the B/Cb value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1 (**Figure 9.9-15**). When RGB is specified as the RPF0 input format through VI6_RPF0_INFMT.RDFMT, specify a B component value. When YCbCr is specified, specify a Cb component value. A value from 0 to 255 can be specified.

(11) RPF0 Color Keying Control Registers (VI6_RPF0_CKEY_CTRL)

Access Size : 32 bits
Address : <VSPI0_base> + 0328h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	LTH	-	-	-	CV	-	-	SAPE1	SAPE0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
8	LTH	0h	RW	Transparent color-luma Threshold Mode Enable/Disable This bit enables or disables transparent color-luma threshold mode for the color keying module. In transparent color-luma threshold mode, color information 0 (GY0) specified in VI6_RPF0_CKEY_SET0 register is compared with the input luma values. When the input data is in YCbCr format, and if input Y value is equal to or smaller than specified color information (VI6_RPF0_CKEY_SET0.GY0), the input α value is replaced with the value specified in VI6_RPF0_CKEY_SET0.AP0. This bit is available only when the input data is in YCbCr format. When the input data is in an RGB format, set this bit to 0. This bit setting is valid only when the CV bit is set to 0; it is ignored when the CV bit is set to 1. When using the virtual input function (VI6_RPF0_INFMT.VIR = 1), set this bit to 0. 0b: Luma threshold mode is disabled (Matched color mode) 1b: Luma threshold mode is enabled
7 to 5	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
4	CV	0h	RW	Color Replacement Control This bit controls the color replacement function in the color keying module shown in **Figure 9.9-3**. When an RGB format is specified as the color space of the RPF0 input data through VI6_RPF0_INFMT.RDFMT, and if all components of an input pixel match the color components specified in VI6_RPF0_CKEY_SET0, the color replacement function replaces the values of the input α and all RGB components with the α and color components specified in VI6_RPF0_CKEY_SET1. When a YCbCr format is specified as the color space of the RPF0 input data through VI6_RPF0_INFMT.RDFMT, only the Y data is compared; if the luminance component of an input pixel matches the value specified in VI6_RPF0_CKEY_SET0.GY0, the color replacement function replaces the values of the input α and all YCbCr components with the α and color components specified in VI6_RPF0_CKEY_SET1. When the CV bit is set to 1, the color replacement function is enabled. When using the virtual input function (VI6_RPF0_INFMT.VIR = 1), set this bit to 0. 0b: Color replacement function is disabled (transparent color mode). 1b: Color replacement function is enabled.
3, 2	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
1	SAPE1	0h	RW	Comparison Color Data Setting 1 Enable/Disable This bit enables or disables comparison color data setting 1 in the transparent color mode for the color keying module. This bit setting is valid when the CV bit and the LTH bit are set to 0; it is ignored when the CV bit or the LTH bit is set to 1. In transparent color mode, color information 1 (VI6_RPF0_CKEY_SET1.R1/GY1/B1) specified in VI6_RPF0_CKEY_SET1 is compared with the input component values. When the input data is in an RGB format, and if all input values match the specified color information, the input α value is replaced with the value specified in VI6_RPF0_CKEY_SET1.AP. When the input data is in YCbCr format, only the Y data is compared. When using the virtual input function (VI6_RPF0_INFMT.VIR = 1), set this bit to 0. 0b: Comparison color data setting 1 is disabled. 1b: Comparison color data setting 1 is enabled.
0	SAPE0	0h	RW	Comparison Color Data Setting 0 Enable/Disable This bit enables or disables comparison color data setting 0 in the transparent color mode for the color keying module. This bit setting is valid when the CV bit and the LTH bit are set to 0; it is ignored when the CV bit or the LTH bit is set to 1. In transparent color mode, color information 0 (VI6_RPF0_CKEY_SET0.R0/GY0/B0) specified in VI6_RPF0_CKEY_SET0 is compared with the input component values. When the input data is in an RGB format, and if all input values match the specified color information, the input α value is replaced with the value specified in VI6_RPF0_CKEY_SET0.AP. When the input data is in YCbCr format, only the Y data is compared. When using the virtual input function (VI6_RPF0_INFMT.VIR = 1), clear this bit to 0. 0b: Comparison color data setting 0 is disabled. 1b: Comparison color data setting 0 is enabled.

(12) RPF0 Color Keying Color Setting Registers-m (VI6_RPF0_CKEY_SETm) (m = 0, 1)

[m = 0]

Access Size : 32 bits
Address : <VSPI0_base> + 032Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Apm[7:0]								Rm[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Gym[7:0]								Bm[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Apm[7:0]	0h	RW	α Data in Color Keying Color Information-m •In transparent color-matched color mode for color keying When the input data matches color setting-m (Rm, GYm, and Bm) for transparent color comparison in the color keying module, the input α value is replaced with the value specified in these bits. Specify the α value to replace the input value. •In transparent color-luma threshold mode for color keying When the input data is in YCbCr format, and if input Y value is equal to or smaller than GY0, the input α value is replaced with the value specified in this bit field (AP0). Specify the α value in this bit field (AP0) to replace the input value. AP1 is not used in this mode. Set AP1 to 0 value. •In color replacement mode for color keying These bits are not used in this mode. Clear them to 0. α value replacement through these bits in transparent color mode for color keying takes priority over the α value selected through the VI6_RPF0_ALPH_SEL.ASEL setting. A value from 0 to 255 can be specified.
23 to 16	Rm[7:0]	0h	RW	R*/Cr Component Data in Color Keying Color Information-m •In transparent color-matched color mode for color keying Specify the R component value for comparison enabled through the VI6_RPF0_CKEY_CTRL.SAPEm setting. When the RPF0 input is in YCbCr format, the color keying module does not compare the Cr component, and the setting of these bits is ignored. •In transparent color-luma threshold mode for color keying The color keying module does not refer this bit field, and the setting of these bits is ignored. •In color replacement mode for color keying Specify the R component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified.
15 to 8	Gym[7:0]	0h	RW	G*/Y Component Data in Color Keying Color Information-m •In transparent color-matched color mode for color keying Specify the G/Y component value for comparison enabled through the VI6_RPF0_CKEY_CTRL.SAPEm setting. •In transparent color-luma threshold mode for color keying Specify the Y component value in the GY0 to be compared. •In color replacement mode for color keying Specify the G/Y component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	Bm[7:0]	0h	RW	B*/Cb Component Data in Color Keying Color Information-m •In transparent color-matched color mode for color keying Specify the B component value for comparison enabled through the VI6_RPF0_CKEY_CTRL.SAPEm setting. When the RPF0 input is in YCbCr format, the color keying module does not compare the Cb component, and the setting of these bits is ignored. •In transparent color-luma threshold mode for color keying The color keying module does not refer this bit field, and the setting of these bits is ignored. •In color replacement mode for color keying Specify the B component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified.

Note 1. When comparison data is specified in an RGB format, if a packed format is selected for RGB input and each of the RGB components is not 8 bits, the lower-order bits of input data are extended as specified through VI6_RPF0_INFMT.CEXT before comparison. The RGB components to be compared with the input should also be extended in the same way and the extended values should be specified in this register.

[m = 1]

Access Size : 32 bits

Address : <VSPI0_base> + 0330h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Apm[7:0]								Rm[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Gym[7:0]								Bm[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Apm[7:0]	0h	RW	α Data in Color Keying Color Information-m •In transparent color-matched color mode for color keying When the input data matches color setting-m (Rm, GYm, and Bm) for transparent color comparison in the color keying module, the input α value is replaced with the value specified in these bits. Specify the α value to replace the input value. •In transparent color-luma threshold mode for color keying When the input data is in YCbCr format, and if input Y value is equal to or smaller than GY0, the input α value is replaced with the value specified in this bit field (AP0). Specify the α value in this bit field (AP0) to replace the input value. AP1 is not used in this mode. Set AP1 to 0 value. •In color replacement mode for color keying These bits are not used in this mode. Clear them to 0. α value replacement through these bits in transparent color mode for color keying takes priority over the α value selected through the VI6_RPF0_ALPH_SEL.ASEL setting. A value from 0 to 255 can be specified.
23 to 16	Rm[7:0]	0h	RW	R*/Cr Component Data in Color Keying Color Information-m •In transparent color-matched color mode for color keying Specify the R component value for comparison enabled through the VI6_RPF0_CKEY_CTRL.SAPEm setting. When the RPF0 input is in YCbCr format, the color keying module does not compare the Cr component, and the setting of these bits is ignored. •In transparent color-luma threshold mode for color keying The color keying module does not refer this bit field, and the setting of these bits is ignored. •In color replacement mode for color keying Specify the R component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified.
15 to 8	Gym[7:0]	0h	RW	G*/Y Component Data in Color Keying Color Information-m •In transparent color-matched color mode for color keying Specify the G/Y component value for comparison enabled through the VI6_RPF0_CKEY_CTRL.SAPEm setting. •In transparent color-luma threshold mode for color keying Specify the Y component value in the GY0 to be compared. •In color replacement mode for color keying Specify the G/Y component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	Bm[7:0]	0h	RW	B*/Cb Component Data in Color Keying Color Information-m •In transparent color-matched color mode for color keying Specify the B component value for comparison enabled through the VI6_RPF0_CKEY_CTRL.SAPEm setting. When the RPF0 input is in YCbCr format, the color keying module does not compare the Cb component, and the setting of these bits is ignored. •In transparent color-luma threshold mode for color keying The color keying module does not refer this bit field, and the setting of these bits is ignored. •In color replacement mode for color keying Specify the B component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified.

Note 1. When comparison data is specified in an RGB format, if a packed format is selected for RGB input and each of the RGB components is not 8 bits, the lower-order bits of input data are extended as specified through VI6_RPF0_INFMT.CEXT before comparison. The RGB components to be compared with the input should also be extended in the same way and the extended values should be specified in this register.

(13) RPF0 Source Picture Memory Stride Setting Registers (VI6_RPF0_SRCM_PSTRIDE)

Access Size : 32 bits
Address : <VSPI0_base> + 0334h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PICT_STRD_Y[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PICT_STRD_C[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	PICT_STRD_Y [15:0]	0h	RW	Memory Stride of Source Picture Y/RGB Plane These bits specify in 1-byte units the memory stride of the source picture Y/RGB plane read by the RPF0. A value from 0 to 65535 can be specified. Refer to **Figure 9.9-16** for settings.
15 to 0	PICT_STRD_C [15:0]	0h	RW	Memory Stride of Source Picture C Plane These bits specify in 1-byte units the memory stride of the source picture C plane read by the RPF0. When an RGB-format picture is read, these bits do not need to be set. A value from 0 to 65535 can be specified. Refer to **Figure 9.9-16** for settings. In the YCbCr planar format, this setting is used as the memory stride of the Cb and Cr planes.

This register specifies the memory stride of the picture planes in the source area as shown in **Figure 9.9-16**. The memory stride of the α plane should be specified through VI6_RPF0_SRCM_ASTRIDE.ALPH_STRD. When the RPF input is in an RGB format, only the RGB plane is read; when the input is in YCbCr format, the Y and C planes are read as shown in **Figure 9.9-16**. When the α plane is used, the α plane is also read. According to the image format and the necessity of the α plane, specify the necessary addresses where the source image is stored (VI6_RPF0_SRCM_ADDR_Y, VI6_RPF0_SRCM_ADDR_C*, and VI6_RPF0_SRCM_ADDR_AI).

Whether the α plane needs to be read is determined according to the α plane selection method and IROP operation type. For details, refer to **9.9.1.3.6 (6) RPF0 α Plane Selection Control Registers (VI6_RPF0_ALPH_SEL)**.

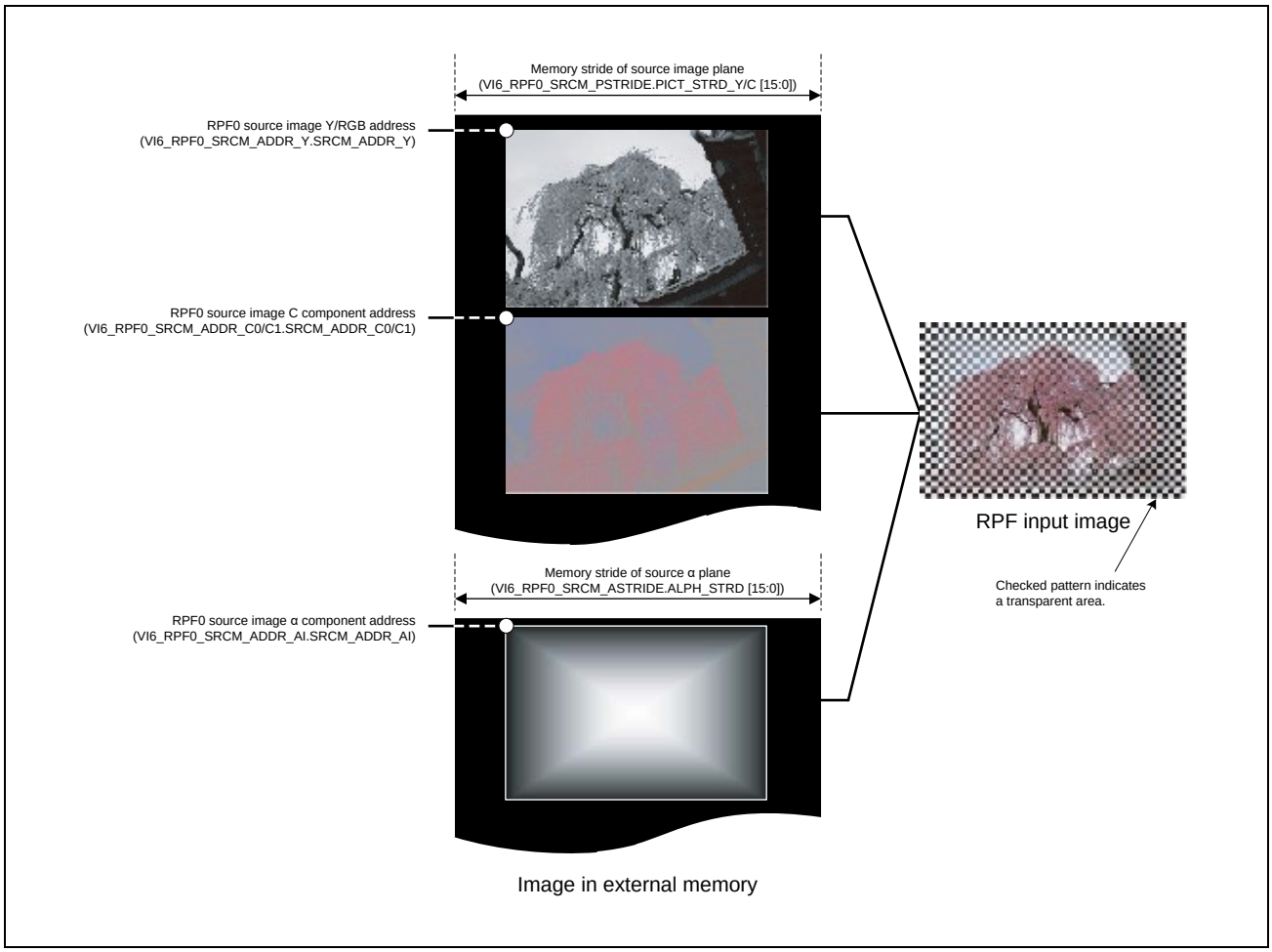


Figure 9.9-16 Reading an Image from RPF0 Source Area

(14) RPF0 Source α Memory Stride Setting Registers (VI6_RPF0_SRCM_ASTRIDE)

Access Size : 32 bits
Address : <VSPI0_base> + 0338h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALPH_STRD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
15 to 0	ALPH_STRD [15:0]	0h	RW	Memory Stride of Source α Plane These bits specify in 1-byte units the memory stride of the source α plane read by the RPF0. A value from 0 to 65535 can be specified. Refer to **Figure 9.9-16** in **9.9.1.3.6 (13)** for settings .

(15) RPF0 Source Y/RGB Address Registers (VI6_RPF0_SRCM_ADDR_Y)

Access Size : 32 bits
Address : <VSPI0_base> + 033Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCM_ADDR_Y[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCM_ADDR_Y[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRCM_ADDR_Y[31:0]	0h	RW	Source Image Y/RGB Plane Storing Address These bits specify in 1-byte units the start address of the source image Y plane and packed RGB plane read by the RPF0. A value from 0000_0000h to FFFF_FFFFh can be specified. Refer to **Figure 9.9-16** in **9.9.1.3.6 (13)** for settings.

(16) RPF0 Source Chroma Address Registers 0 (VI6_RPF0_SRCM_ADDR_C0)

Access Size : 32 bits
Address : <VSPI0_base> + 0340h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCM_ADDR_C0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCM_ADDR_C0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRCM_ADDR_C0[31:0]	0h	RW	Source Image C Plane Storing Address 0 These bits specify in 1-byte units the start address of the source image C plane read by the RPF0. Here, the C plane indicates the combined CbCr plane when a semi-planar format is selected from the packed YCbCr formats shown in **Table 9.9-8** or the Cb plane when a planar format is selected. When an interleaved format is selected or the RPF input is in an RGB format, this setting is not used. A value from 0000_0000h to FFFF_FFFFh can be specified. Refer to **Figure 9.9-16** in **9.9.1.3.6 (13)** for settings.

(17) RPF0 Source Chroma Address Registers 1 (VI6_RPF0_SRCM_ADDR_C1)

Access Size : 32 bits
Address : <VSPI0_base> + 0344h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCM_ADDR_C1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCM_ADDR_C1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRCM_ADDR_C1[31:0]	0h	RW	Source Image C Plane Storing Address 1 These bits specify in 1-byte units the start address of the Cr plane when a planar YCbCr format shown in **Table 9.9-8** is read by the RPF0. This setting is not used when the RPF input is in YCbCr format that is not a planar format or in an RGB format. A value from 0000_0000h to FFFF_FFFFh can be specified. Refer to **Figure 9.9-16** in **9.9.1.3.6 (13)** for settings.

(18) RPF0 Source α Address Registers (VI6_RPF0_SRCM_ADDR_AI)

Access Size : 32 bits
Address : <VSPI0_base> + 0348h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCM_ADDR_AI[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCM_ADDR_AI[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SRCM_ADDR_AI[31:0]	0h	RW	Source Image α Plane Storing Address These bits specify in 1-byte units the start address of the α plane of the source image read by the RPF0. Specify in the same way as the start address of the picture plane. When the α plane is not read from the source area, these bits do not need to be set. A value from 0000_0000h to FFFF_FFFFh can be specified. Refer to **Figure 9.9-16** in **9.9.1.3.6 (13)** for settings.

(19) RPF0 Bus Access Control Registers (VI6_RPF0_BAC)

Access Size : 32 bits
Address : <VSPI0_base> + 0350h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	B512
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
16	B512	0h	RW	Burst access in 512 pixels' enable 0: burst access in 256 pixels 1: burst access in 512 pixels
15 to 0	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.

(20) RPF0 Multiple Alpha Control (VI6_RPF0_MULT_ALPH)

Access Size : 32 bits
Address : <VSPI0_base> + 036Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	A_MMD	-	-	P_MMD[1:0]		APLPHA_RATIO[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12	A_MMD	0h	RW	0b: Alpha data go through multiply-alpha unit. 1b: Multiply-alpha unit multiplies alpha data by specified alpha (ALPHA_RATIO [7:0]). When output format from CSC unit is YCbCr, set 0 to this bit.
11, 10	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
9, 8	P_MMD[1:0]	0h	RW	00b: Image data go through multiply-alpha unit. 01b: Multiply-alpha unit multiplies image data by specified alpha (ALPHA_RATIO [7:0]). 10b: Multiply-alpha unit multiplies image data by alpha data. 11b: Multiply-alpha unit multiplies image data by alpha data and specified alpha (ALPHA_RATIO [7:0]). When output format from CSC unit is YCbCr, set 0 to this bit.
7 to 0	APLPHA_RATIO[7:0]	0h	RW	Multiply alpha value

- ALPin: Input Alpha data to Multiply-alpha unit
- PIXin(R): Input R data to Multiply-alpha unit
- PIXin(G): Input G data to Multiply-alpha unit
- PIXin(B): Input B data to Multiply-alpha unit
- ALPout: Output Alpha data from Multiply-alpha unit
- PIXout(R): Output R data from Multiply-alpha unit
- PIXout(G): Output G data from Multiply-alpha unit
- PIXout(B): Output B data from Multiply-alpha unit

Table 9.9-13 Expression of Output Alpha Data from Multiply-Alpha Unit

A_MMD	ALPHA_RATIO[7:0]	Expression
0	Don't care	ALPout = ALPin
1	Not 255	ALPout = ALPin × ALPHA_RATIO / 256
	255	ALPout = ALPin

Table 9.9-14 Expression of Output Pixel Data from Multiply-Alpha Unit

P_MMD[1:0]	ALPHA_RATIO[7:0]	ALPin	Expression
0	Don't care	Don't care	PIXout(R) = PIXin(R) PIXout(G) = PIXin(G) PIXout(B) = PIXin(B)
1	Not 255	Don't care	PIXout(R) = PIXin(R) × ALPHA_RATIO / 256 PIXout(G) = PIXin(G) × ALPHA_RATIO / 256 PIXout(B) = PIXin(B) × ALPHA_RATIO / 256
	255	Don't care	PIXout(R) = PIXin(R) PIXout(G) = PIXin(G) PIXout(B) = PIXin(B)
2	Don't care	Not 255	PIXout(R) = PIXin(R) × ALPin / 256 PIXout(G) = PIXin(G) × ALPin / 256 PIXout(B) = PIXin(B) × ALPin / 256
	Don't care	255	PIXout(R) = PIXin(R) PIXout(G) = PIXin(G) PIXout(B) = PIXin(B)
3	Not 255	Not 255	PIXout(R) = PIXin(R) × ALPin × ALPHA_RATIO / 256 / 256 PIXout(G) = PIXin(G) × ALPin × ALPHA_RATIO / 256 / 256 PIXout(B) = PIXin(B) × ALPin × ALPHA_RATIO / 256 / 256
	255	Not 255	PIXout(R) = PIXin(R) × ALPin / 256 PIXout(G) = PIXin(G) × ALPin / 256 PIXout(B) = PIXin(B) × ALPin / 256
	Not 255	255	PIXout(R) = PIXin(R) × ALPHA_RATIO / 256 PIXout(G) = PIXin(G) × ALPHA_RATIO / 256 PIXout(B) = PIXin(B) × ALPHA_RATIO / 256
	255	255	PIXout(R) = PIXin(R) PIXout(G) = PIXin(G) PIXout(B) = PIXin(B)

9.9.1.3.7 WPF Control Registers

(1) WPF0-Source-RPF Registers (VI6_WPF0_SRCRPF)

Access Size : 32 bits
Address : <VSPI0_base> + 1000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	RW	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RPF0_ACT[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
29, 28	-	0h	RW	Reserved These bits are always read as 0. The write value should always be 0.
27, 26	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
25, 24	-	0h	RW	Reserved These bits are always read as 0. The write value should always be 0.
23 to 10	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
9 to 2	-	0h	RW	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	RPF0_ACT[1:0]	0h	RW	RPF0 Start Enable (RPF0_ACT) These bits enable start of RPF0 as the source RPF for the WPF0 when the WPF0 is started. When RPF0 is not started by WPF0, set the VI6_DPR_RPF0_ROUTE.RT_RPF0 bits to D'63. 00b: RPF0 is not started. 01b: RPF0 is started as the master-source RPF for the WPF0. 10b: RPF0 is started as the master-layer source RPF for the WPF0. 11b: Setting prohibited

(2) WPF0 Horizontal Input Size Clipping Registers (VI6_WPF0_HSZCLIP)

Access Size : 32 bits
Address : <VSPI0_base> + 1004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	HCEN	-	-	-	-	HCL_OFST[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	HCL_SIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28	HCEN	0h	RW	Horizontal Size Clipping Enable/Disable Enables or disables clipping of the horizontal size of the WPF0 input image. 0b: Horizontal size clipping is disabled 1b: Horizontal size clipping is enabled
27 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 16	HCL_OFST[7:0]	0h	RW	Horizontal Size Clipping Offset Value Setting These bits specify the offset size (pixels) from the left end of the image in horizontal size clipping when the HCEN bit is 1 (**Figure 9.9-17**). The left side of the image input to the WPF is cut off for the size specified in these bits. When the HCEN bit is 0, this setting is ignored. For WPF0, specify the left-side offset size (pixels) for the image before rotation. A value from 0 to 255 can be specified. (HCL_OFST + HCL_SIZE) should not exceed the horizontal size of the WPF input. If the setting shown in the bottom example in **Figure 9.9-17** is made, VSPI does not operate correctly.
15 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12 to 0	HCL_SIZE[12:0]	0h	RW	Horizontal Clipping Size Setting When the HCEN bit is 1, these bits specify the clipping size for horizontal clipping processing. Through this processing, the area of the horizontal size specified through the HCL_SIZE bits starting from the offset position specified through the HCL_OFST bits is determined as the valid image area. Accordingly, the right-side pixels beyond the (HCL_OFST + HCL_SIZE) size in the WPF0 input image are discarded. When the HCEN bit is 0, this setting is ignored. For WPF0, specify the horizontal clipping size for the image before rotation. A value from 1 to 8190 can be specified. (HCL_OFST + HCL_SIZE) should not exceed the horizontal size of the WPF input. If the setting shown in the bottom example in **Figure 9.9-17** is made, VSPI does not operate correctly. (Note) When the WPF0 does not execute 90° or 270° rotation (bit 2 in VI6_WPF0_OUTFMT.ROT is 0) and the WPF0 output format is YCbCr4:2:2 or YCbCr4:2:0, specify an even value in these bits. When the WPF0 executes 90° or 270° rotation (bit 2 in VI6_WPF0_OUTFMT.ROT is 1) and the WPF0 output format is YCbCr4:2:0, specify an even value in these bits.

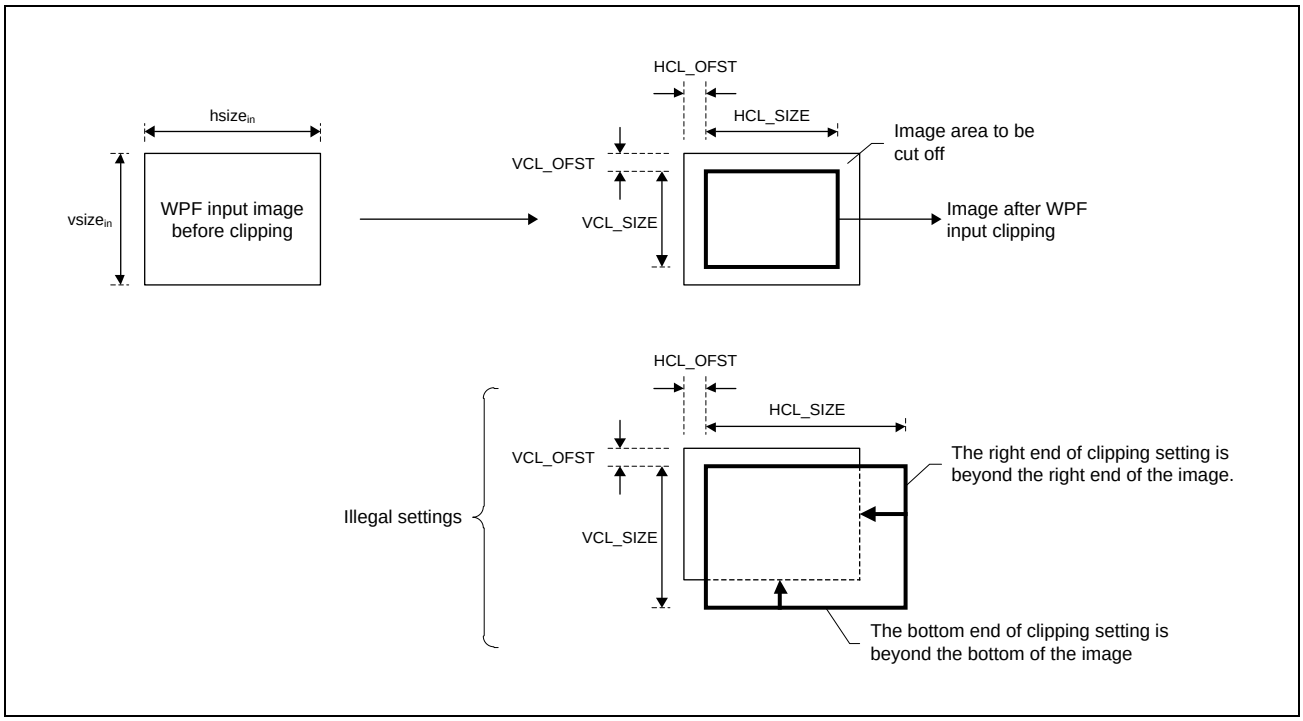


Figure 9.9-17 Image Clipping in WPF Input Section

(3) WPF0 Vertical Input Size Clipping Registers (VI6_WPF0_VSZCLIP)

Access Size : 32 bits
Address : <VSPI0_base> + 1008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	VCEN	-	-	-	-	VCL_OFST[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VCL_SIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28	VCEN	0h	RW	Vertical Size Clipping Enable/Disable Enables or disables clipping of the vertical size of the WPF0 input image. 0b: Vertical size clipping is disabled 1b: Vertical size clipping is enabled
27 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 16	VCL_OFST[7:0]	0h	RW	Vertical Size Clipping Offset Value Setting These bits specify the offset size (pixels) from the top end of the image in vertical size clipping when the VCEN bit is 1b (**Figure 9.9-17**). The top of the image input to the WPF is cut off for the size specified in these bits. When the VCEN bit is 0, this setting is ignored. For WPF0, specify the top offset size (pixels) for the image before rotation. A value from 0 to 255 can be specified. (VCL_OFST + VCL_SIZE) should not exceed the vertical size of the WPF input. If the setting shown in the bottom example in **Figure 9.9-17** is made, VSPI does not operate correctly.
15 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12 to 0	VCL_SIZE[12:0]	0h	RW	Vertical Clipping Size Setting When the VCEN bit is 1, these bits specify the clipping size for vertical clipping processing. Through this processing, the area of the vertical size specified through the VCL_SIZE bits starting from the offset position specified through the VCL_OFST bits is determined as the valid image area. Accordingly, the bottom pixels beyond the (VCL_OFST + VCL_SIZE) size in the WPF0 input image are discarded. When the VCEN bit is 0, this setting is ignored. For WPF0, specify the vertical clipping size for the image before rotation. A value from 1 to 8190 can be specified. (VCL_OFST + VCL_SIZE) should not exceed the vertical size of the WPF input. If the setting shown in the bottom example in **Figure 9.9-17** is made, VSPI does not operate correctly. (Note) When the WPF0 does not execute 90° or 270° rotation (bit 2 in VI6_WPF0_OUTFMT.ROT is 0) and the WPF0 output format is YCbCr4:2:0, specify an even value in these bits. When the WPF0 executes 90°or 270° rotation (bit 2 in VI6_WPF0_OUTFMT.ROT is 1) and the WPF0 output format is YCbCr4:2:2 or YCbCr4:2:0, specify an even value in these bits.

(4) WPF0 Output Format Registers (VI6_WPF0_OUTFMT)

Access Size : 32 bits
Address : <VSPI0_base> + 100Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PDV[7:0]								PXA	ODE	-	-	-	ROT[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	R	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPYCS	SPUVS	DITH[1:0]		WRTM[2:0]			CSC	-	WRFMT[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	PDV[7:0]	0h	RW	PAD Value in Output Packed Data These bits specify the value to be stored in the bit field indicated as PAD or P in the output formats shown in **Table 9.9-15**. To store this value in PAD, specify 0 in the PXA bit. A value from 0 to 255 can be specified.
23	PXA	0h	RW	PAD Data Select Selects the value to be stored in the bit field indicated as PAD or P in the packed RGB output formats shown in **Table 9.9-15**. Both the value specified in the PDV bits and the α data input from the DPR to WPF are 8 bits, but some of the PAD and P bit fields shown in **Table 9.9-15** are four bits or one bit. When the target bit field is not 8 bits, the number of bits in the PDV value and the α data input from the DPR to WPF is reduced according to the VI6_WPF0_RNDCTRL.ABRM setting. For bit count reduction, refer to **Figure 9.9-18** and the description of VI6_WPF0_RNDCTRL.ABRM. 0b: The value specified in the PDV bits is stored in the PAD shown in **Table 9.9-15**. 1b: The α value output from DPR in pixel units is stored in the PAD shown in **Table 9.9-15**.
22	ODE	0h	RW	Ordered Dither (mode A) Enable/Disable 0b: Ordered dither (mode A) is disabled. 1b: Ordered dither (mode A) is enabled. When the output format specified through the WRFMT bits is YCbCr, specify 0 in this bit. And when VI6_WPF0_OUTFMT.CSC is set to 1, specify 0 in this bit even in the case that the output format specified through the WRFMT bits is RGB. Ordered dither is available only for 18bpp. So, when ODE bit is equal to 1, set WRFMT at 18bpp format. When ODE bit is equal to 0, WPF dither method is specified by DITH[1:0] in the register Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.
21	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.
20	-	0h	RW	Reserved These bits are read as 0b. The write value should always be 0b.
19	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
18 to 16	ROT[2:0]	0h	RW	Rotation Processing Select These bits select the rotation/flipping processing to be applied to the WPF0 output image. 90° rotation and 270° rotation are done clockwise. **Figure 9.9-19** and **Figure 9.9-20** show the correspondence between the original image and the rotation/flipping result according to each setting. 000b: No rotation or flipping 001b: Vertical flipping 010b: Horizontal flipping 011b: 180° rotation 100b: 90° rotation 101b: 90° rotation + vertical flipping (equal to 270° rotation + horizontal flipping) 110b: 90° rotation + horizontal flipping (equal to 270° rotation + vertical flipping) 111b: 270° rotation Rotation/flipping can be specified only in VSPI - WPF0. In other channels, refer to **9.9.1.5.4** for restriction of these bits. Note that the destination address setting should be changed according to the setting of these bits. For details, refer to **9.9.1.3.7 (10)**. When the LIF module is used (VI6_LIF_CTRL.LIF_EN = 1), set 0 to ROT [2:0].
15	SPYCS	0h	RW	WPF Output Mode Setting 1 When the output format is YUY2, set this bit to 1 and set the WRFMT bits to 71 (47h). When the output format is YVYU, set this bit and the SPUVS bit to 1 and set the WRFMT bits to 71 (47h). In other cases, set this bit to 0.
14	SPUVS	0h	RW	WPF Output Mode Setting 2 When the output format is NV61, set this bit to 1 and set the WRFMT bits to 65 (41h). When the output format is NV21, set this bit to 1 and set the WRFMT bits to 66 (42h). When the output format is YVYU, set this bit and the SPYCS bit to 1 and set the WRFMT bits to 71 (47h). In other cases, set this bit to 0.
13, 12	DITH[1:0]	0h	RW	Ordered Dither (mode B) Enable/Disable When the output format specified through the WRFMT bits is RGB with 18 bpp (260000 colors) or less, the color reduction processing is applied to match the number of colors. The color reduction processing may generate the artifacts of pseudo gradation, which can be suppressed through dithering. The DITH bits enable or disable dithering during color reduction. When the output format specified through the WRFMT bits is YCbCr, specify 0 in these bits. And when VI6_WPF0_OUTFMT.CSC is set to 1, specify 0 in these bits even in the case that the output format specified through the WRFMT bits is RGB. 00b: Dithering (mode B) is disabled 11b: Dithering (mode B) is enabled 01b, 10b: Setting prohibited When ODE bit in the register is 1, set this bit to 0.
11 to 9	WRTM[2:0]	0h	RW	CSC Conversion Expression Setting These bits select the expression for color space conversion. The conversion direction is RGB to YCbCr when the format specified in the WRFMT bits is YCbCr, or YCbCr to RGB when the format specified in the WRFMT bits is RGB. 000b: BT.601 YCbCr [16,235/240] ↔ RGB [0,255] 001b: BT.601 YCbCr [0,255] ↔ RGB [0,255] 010b: BT.709 YCbCr [16,235/240] ↔ RGB [0,255] 011b: BT.709 YCbCr [16,235/240] ↔ RGB [16,235] 100b to 111b: Setting prohibited
8	CSC	0h	RW	Color Space Conversion Setting Enables or disables YCbCr ↔ RGB color space conversion to be executed in the WPF0. The characteristics of color space conversion are determined by the WRTM setting. There are some points to be noted about the relationship between the CSC setting and output format (WRFMT). For details refer to (*1) in **9.9.1.3.6 (3)**, RPF0 Input Format Registers (VI6_RPF0_INFMT). 0b: Color space is not converted. 1b: Color space is converted.
7	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
6 to 0	WRFMT[6:0]	0h	RW	WPF Output Image Format Setting These bits select the format of the image output from the WPF0 to the external memory from among those listed in **Table 9.9-15** and **Table 9.9-16**. [Note 1] Number of output pixels When YCbCr4:2:2 is specified through WRFMT, the horizontal size of the output image should be a multiple of 2 pixels. When YCbCr4:2:0 is specified, the vertical and horizontal sizes of the output image should be multiples of 2 pixels. Specify an appropriate data flow of the source RPF → DPR → target WPF so that the size of the image input to the target WPF satisfies the above restrictions. In particular, when the data flow includes a module or a function that modifies (up-scales, down-scales, or clips) the image size, take special care about the module or function settings. In addition, note the rotation function of the WPF shown in **Figure 9.9-4** When 90° or 270° rotation is used (bit 2 in the ROT bits has a value of 1), the horizontal size of the WPF output image is equal to the vertical size of the WPF input image, and the vertical size of the output image is equal to the horizontal size of the input image; that is, the horizontal and vertical sizes are switched between the input and output. For example, because of the above restriction on YCbCr4:2:2 that the horizontal output size should be a multiple of 2 pixels, the vertical size of the WPF input should be a multiple of 2 pixels. [Note 2] Output lines in YCbCr4:2:0 In the YCbCr4:2:0 output format, the number of chrominance lines in the vertical direction is one-half the number of luminance lines. For this reason, the WPF outputs only even-numbered chrominance lines (lines 0, 2, 4, 6, ...) (conversion from (A) to (B) in. When vertical flipping is also specified through the ROT bits, the flipping processing is executed last and the chrominance line locations are inverted (lines 1, 3, 5, 7, …) in the output image ((C) in **Figure 9.9-21**). [Note 3] Down sampling of CbCr in horizontal direction in YCbCr4:2:0 or YCbCr4:2:2 In the YCbCr4:2:2 or YCbCr4:2:0 output format, method of down sample of Cb/Cr in horizontal direction is average of neighbor two pixels.

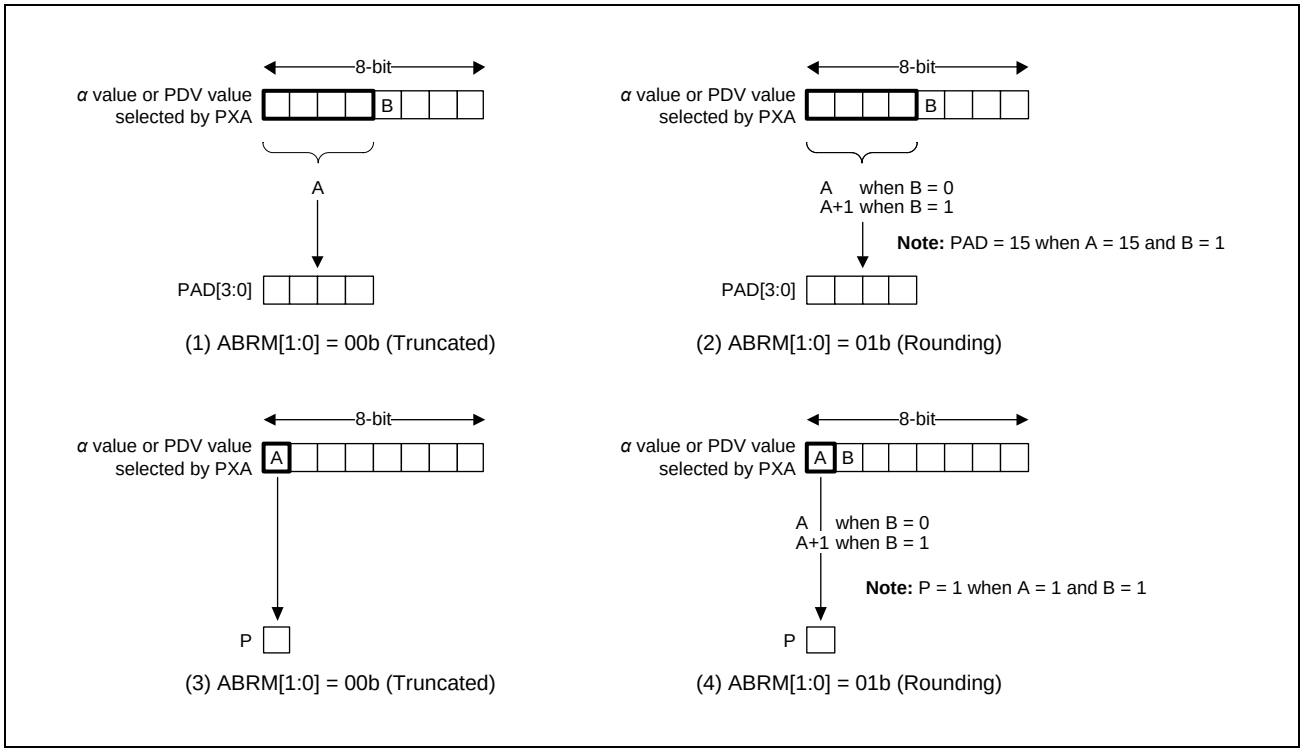


Figure 9.9-18 Selection of PAD Value and Reduction of Bit Count through PXA Setting

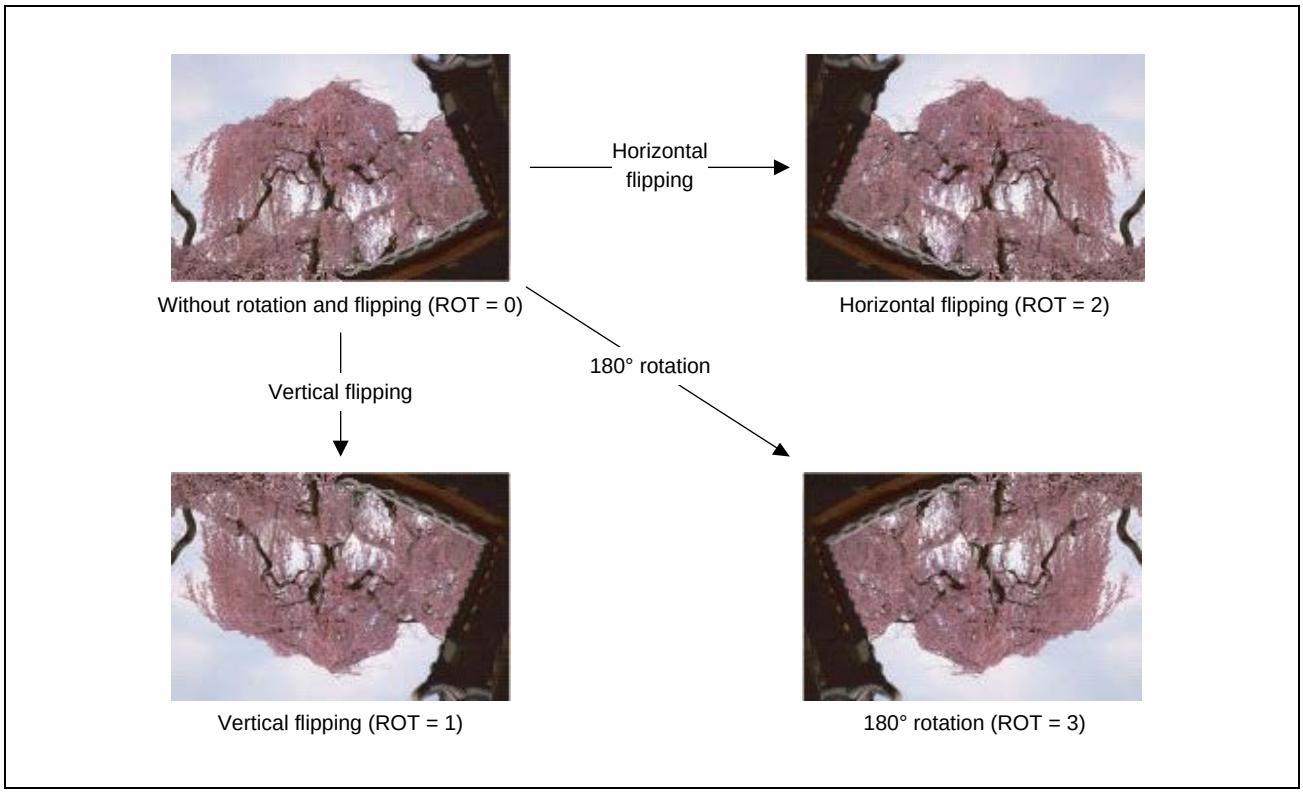


Figure 9.9-19 Correspondence between Original Image and Rotation/Flipping Result according to ROT Setting (1)

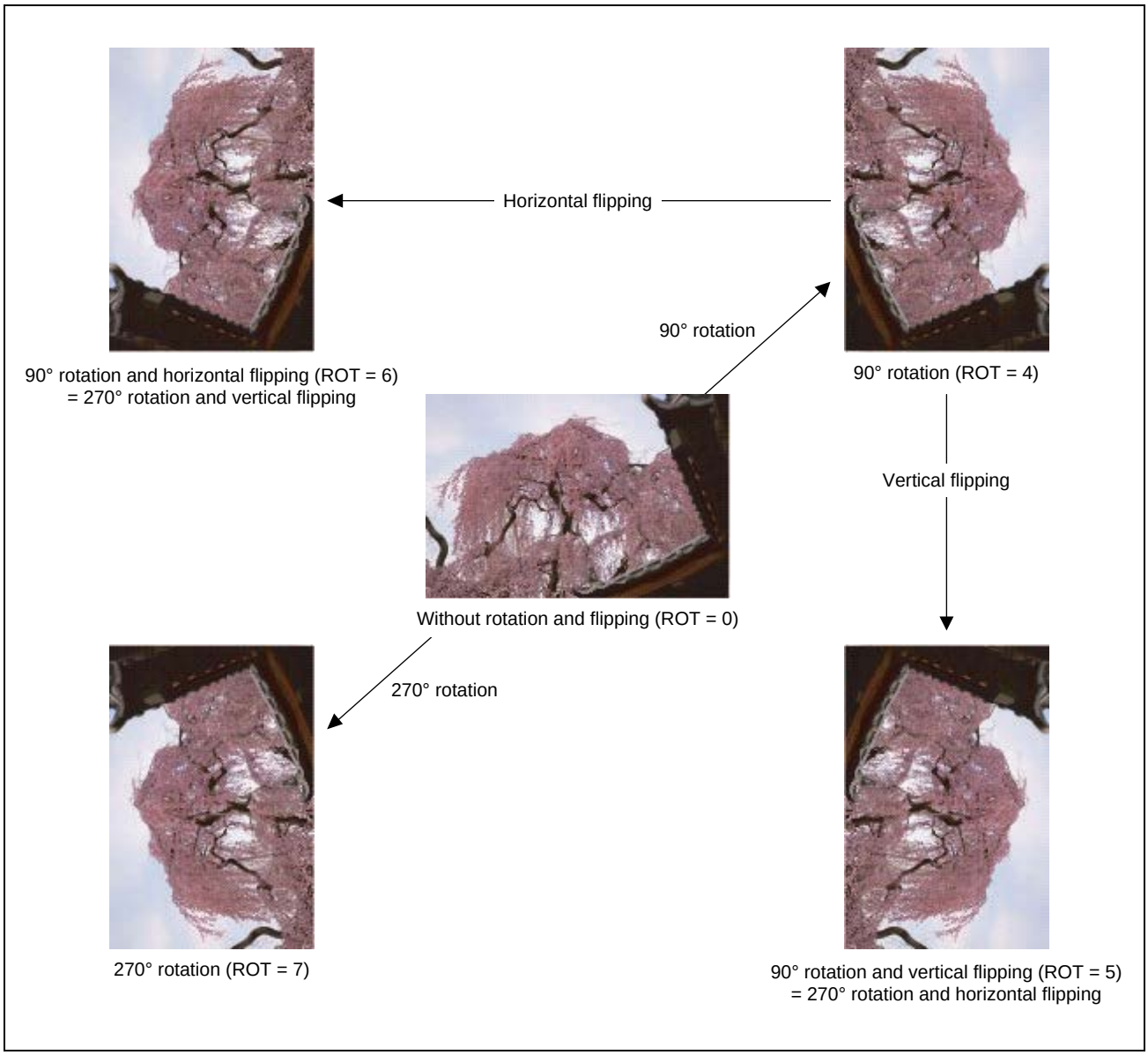


Figure 9.9-20 Correspondence between Original Image and Rotation/Flipping Result according to ROT Setting (2)

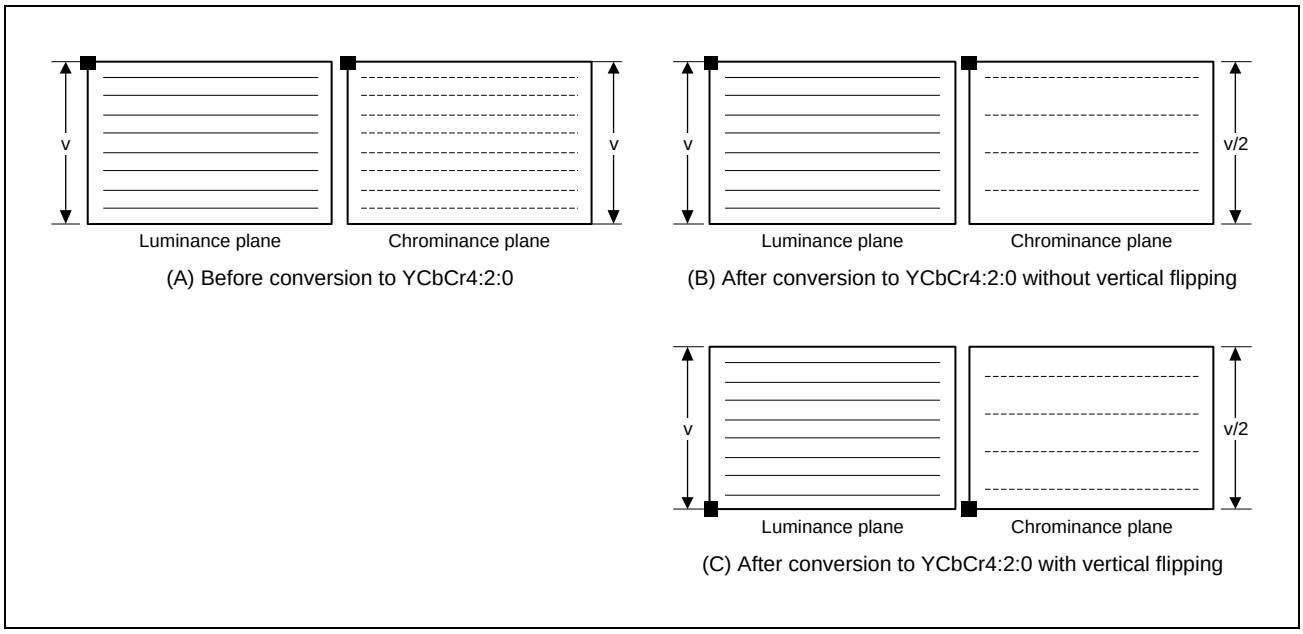


Figure 9.9-21 Chrominance Output Lines in YCbCr4:2:0 and Vertical Flipping Result

Table 9.9-15 Packed RGB Formats for WPF Output

WRFMT[6:0]	Bit per Pixel	Phase	Upper Row - Address / Bottom Row - Bit Field																													
			n								n+1								n+2								n+3							
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
00h	8	—	R0	R0	R0	G0	G0	G0	B0	B0	R1	R1	R1	G1	G1	G1	B1	B1	R2	R2	R2	G2	G2	G2	B2	B2	R3	R3	R3	G3	G3	G3	B3	B3
01h	12	—	0	0	0	0	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
02h			R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	0	0	0	0
03h	—	—	Reserved								Reserved								Reserved								Reserved							
04h	15	—	0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
05h			R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	0
06h	16	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
07h	18	—	PAD								0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
08h			R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	PAD							
09h			0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	PAD							
0Ah			PAD								R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0
0Bh			PAD								0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0
0Ch			0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	PAD							
0Dh			PAD								R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0
0Eh			R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	PAD							
0Fh		0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	R1	R1
		1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	0	0	0	0	0	0	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2
		2	G2	G2	B2	B2	B2	B2	B2	B2	0	0	0	0	0	0	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3
10h		0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	R1	R1	R1	R1	R1	R1	G1	G1
		1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	0	0	0	0	0	0	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	B2	B2	B2	B2
		2	B2	B2	0	0	0	0	0	0	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	0	0	0	0	0	0
11h		0	0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1
		1	0	0	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2
		2	0	0	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3
12h		0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1	0	0
		1	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2	0	0
		2	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3	0	0
13h	24	—	PAD								R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
14h			R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	PAD							
15h		0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
		1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
		2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
16h	18	—	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	0	0	0	0	0	0	0	G0	G0	G0	B0	B0	B0	B0	B0	B0
17h			0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
18h	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1
		1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2
		2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3
19h	12	—	PAD				R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	PAD				R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
1Ah		—	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	PAD				R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	PAD			
1Bh	15	—	P	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	P	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
1Ch		—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	P	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	P
1Dh	12	—	PAD				B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	PAD				B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1
1Eh		—	B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	PAD				B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1	PAD			
1Fh	15	—	P	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	P	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1
20h		—	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	P	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	P
21h	18	0	0	0	B0	B0	B0	B0	B0	B0	0	0	G0	G0	G0	G0	G0	G0	0	0	R0	R0	R0	R0	R0	R0	0	0	B1	B1	B1	B1	B1	B1
		1	0	0	G1	G1	G1	G1	G1	G1	0	0	R1	R1	R1	R1	R1	R1	0	0	B2	B2	B2	B2	B2	B2	0	0	G2	G2	G2	G2	G2	G2
		2	0	0	R2	R2	R2	R2	R2	R2	0	0	B3	B3	B3	B3	B3	B3	0	0	G3	G3	G3	G3	G3	G3	0	0	R3	R3	R3	R3	R3	R3
22h	24	—	PAD								B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0
23h	16	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0
24h to 3Fh	—	—	Reserved								Reserved								Reserved								Reserved							

Table 9.9-16 Packed YCbCr Formats for WPF Output

WRFMT[6:0]	Packed YCbCr Output Format	Reference
40h	YCbCr4:4:4 semi-planar	**Figure 9.9-9***4
41h	YCbCr4:2:2 semi-planar (NV16, NV61*1)	
42h	YCbCr4:2:0 semi-planar (NV12, NV21*1)	
43h to 45h	Reserved	—
46h	YCbCr4:4:4 interleaved	**Figure 9.9-10***4
47h	YCbCr4:2:2 interleaved type 0 (UYVY, YUY2*2, YVYU*3)	
48h	YCbCr4:2:2 interleaved type 1	
49h	YCbCr4:2:0 interleaved*5	
4Ah	YCbCr4:4:4 planar	**Figure 9.9-11***4
4Bh	YCbCr4:2:2 planar (YV16)	
4Ch	YCbCr4:2:0 planar (YV12, I420)	
4Dh to 7Fh	Reserved	—

Note 1. When the output format is NV61 or NV21, set SPUVS (bit 14) to 1.

Note 2. When the output format is YUY2, set SPYCS (bit 15) to 1.

Note 3. When the output format is YVYU, set SPUVS (bit 14) to 1 and SPYCS (bit 15) to 1.

Note 4. **Figure 9.9-12** shows the definition of memory address for each pixel in **Figure 9.9-9** to **Figure 9.9-11**.

Note 5. Each line of plane is written twice, so byte/pixel of YCbCr420ITL is 3 byte/pixel (same as YCbCr444ITL).

For details of each YCbCr format, refer to **Figure 9.9-9**, **Figure 9.9-10**, **Figure 9.9-11**, and **Figure 9.9-12**. In these figures, registers for the RPF are indicated; read them as registers for the WPF as follows.

(RPF registers in the figures)	→	(Corresponding WPF registers)
VI6_RPF0_SRCM_ADDR_Y.SRCM_ADDR_Y	→	VI6_WPF0_DSTM_ADDR_Y.DSTM_ADDR_Y
VI6_RPF0_SRCM_ADDR_C0.SRCM_ADDR_C0	→	VI6_WPF0_DSTM_ADDR_C0.DSTM_ADDR_C0
VI6_RPF0_SRCM_ADDR_C1.SRCM_ADDR_C1	→	VI6_WPF0_DSTM_ADDR_C1.DSTM_ADDR_C1

(5) WPF0 Data Swapping Registers (VI6_WPF0_DSWAP)

Access Size : 32 bits
Address : <VSPI0_base> + 1010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	P_LLS	P_LWS P_LWS	P_WDS	P_BTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
3	P_LLS	0h	RW	WPF Output Data Swapping in LONG LWORD Units The effect of this bit setting is the same as data swapping in the RPF; refer to **Table 9.9-10**. 0: Data swapping in LONG LWORD (64-bit) units is disabled 1: Data swapping in LONG LWORD (64-bit) units is enabled
2	P_LWSP_LWS	0h	RW	WPF Output Data Swapping in long word Units The effect of this bit setting is the same as data swapping in the RPF; refer to **Table 9.9-10**. 0: Data swapping in long word (32-bit) units is disabled 1: Data swapping in long word (32-bit) units is enabled
1	P_WDS	0h	RW	WPF Output Data Swapping in Word Units The effect of this bit setting is the same as data swapping in the RPF; refer to **Table 9.9-10**. 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled
0	P_BTS	0h	RW	WPF Output Data Swapping in Byte Units The effect of this bit setting is the same as data swapping in the RPF; refer to **Table 9.9-10**. 0: Data swapping in byte (8-bit) units is disabled 1: Data swapping in byte (8-bit) units is enabled

Table 9.9-10 shows the data order before and after swapping according to the long word, word, half word, and byte swapping settings.

When data order in memory for each format is the same as in **Table 9.9-15** for RGB format and **Figure 9.9-9** to **Figure 9.9-11** for YCbCr format, set 1111b to {*_LLS, *_LWS, *_WDS, *_BTS}. If data order is not the same as the definition, change data order within 16byte unit by these bits as shown in **Table 9.9-10**.

(6) WPF0 Rounding Control Registers (VI6_WPF0_RNDCTRL)

Access Size : 32 bits
Address : <VSPI0_base> + 1014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	CBRM	-	-	ABRM[1:0]		ATHRESH[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	CLMD[1:0]		-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28	CBRM	0h	RW	Bit Count Reduction Method Selection for Data Storage in Packed RGB This bit specifies the method for reducing the number of bits when data is stored in the bit fields indicated as R, G, and B in **Table 9.9-15** and the target bit fields are not 8 bits. 0: Bit count conversion: The lower-order bits are truncated 1: Bit count conversion: Rounding (rounding off)
27, 26	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
25, 24	ABRM[1:0]	0h	RW	Bit Count Reduction Method Selection for Data Storage in PAD These bits specify the method for reducing the number of bits when the data selected through VI6_WPF0_OUTFMT.PXA is stored in the bit fields indicated as PAD or P in **Table 9.9-15** and the target bit field is four bits or one bit. A value of 10b can be specified only when the packed RGB format specified through VI6_WPF0_OUTFMT.WRFMT includes a 1-bit P field. In this case, when the data selected through VI6_WPF0_OUTFMT.PXA is greater than the ATHRESH value, 1 is stored in the P field; when the selected data is not greater than the ATHRESH value, 0 is stored. 00b: Bit count conversion: The lower-order bits are truncated 01b: Bit count conversion: Rounding (rounding off) 10b: Bit count conversion: Comparison with the threshold value (this setting is allowed only when the storage field is one bit) 11b: Setting prohibited
23 to 16	ATHRESH[7:0]	0h	RW	Threshold for Conversion to 1-Bit α Data These bits specify the threshold value used for conversion from 8-bit α data to one bit when the ABRM bits are set to 10b. When the 8-bit α value before bit count reduction is equal to or smaller than the ATHRESH value, 0 is stored as the reduced 1-bit α data. In other cases, 1 is stored as the 1-bit α data. A value from 0 to 255 can be specified.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13, 12	CLMD[1:0]	0h	RW	Color Data Clipping These bits specify the method for clipping the YCbCr color data output from the WPF. When RGB color data is output from the WPF, specify 0 in these bits. 00b: Output value is not clipped (0-255) 01b: Output value is clipped: YCbCr mode 1 (16-235 (Y), 16-240 (Cb/Cr)) 10b: Output value is clipped: YCbCr mode 2 (Y/Cb/Cr = 1-254) 11b: Setting prohibited
11 to 0	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.

(7) WPF0 Rotation Control Registers (VI6_WPF0_ROT_CTRL)

Access Size : 32 bits

Address : <VSPI0_base> + 1018h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LN16	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	LMEM_WD[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
17	LN16	0h	RW	16-Line Rotation Mode Selects the mode for rotation processing. The 16-line rotation mode can improve the bus transfer efficiency in rotation processing, but the maximum horizontal image size that can be rotated is one half of that in 8-line rotation mode. 0b: 8-line rotation mode 1b: 16-line rotation mode Set 1 to this bit for transfer efficiency.
16 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12 to 0	LMEM_WD [12:0]	0h	RW	Rotation Memory Word Count Always specify 256 here.

(8) WPF0 Destination Y Plane Memory Stride Registers (VI6_WPF0_DSTM_STRIDE_Y)

Access Size : 32 bits
Address : <VSPI0_base> + 101Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PICT_STRD_Y[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
15 to 0	PICT_STRD_Y [15:0]	0h	RW	Memory Stride of Destination Picture Y/RGB Plane These bits specify in 1-byte units the memory stride of the destination picture in the external memory to be written to by the WPFn as shown in **Figure 9.9-22**. A value from 0000h to FFFFh can be specified.

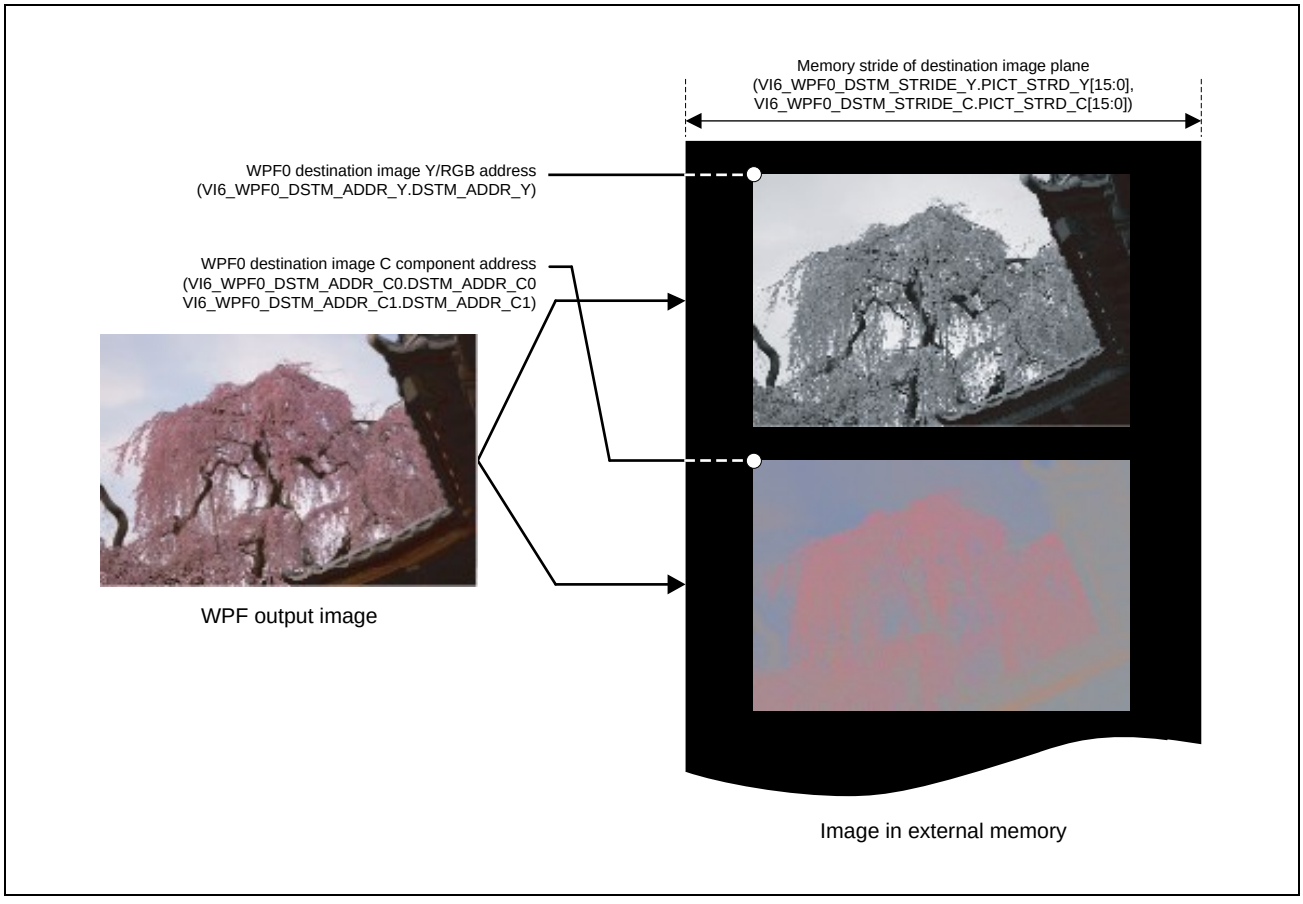


Figure 9.9-22 Writing Image Data to Destination Area in WPF0

(9) WPF0 Destination C Plane Memory Stride Registers (VI6_WPF0_DSTM_STRIDE_C)

Access Size : 32 bits

Address : <VSPI0_base> + 1020h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PICT_STRD_C[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
15 to 0	PICT_STRD_C [15:0]	0h	RW	Memory Stride of Destination Picture C Plane These bits specify in 1-byte units the memory stride for the C plane of the destination picture in the external memory to be written to by the WPF0 as shown in **Figure 9.9-22**. When the WPF0 outputs images in an RGB format, this setting is not used. When the WPF0 outputs images in YCbCr planar format, this setting is applied to both the Cb and Cr planes. A value from 0000h to FFFFh can be specified.

(10) WPF0 Destination Y/RGB Address Registers (VI6_WPF0_DSTM_ADDR_Y)

Access Size : 32 bits
Address : <VSPI0_base> + 1024h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSTM_ADDR_Y[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSTM_ADDR_Y[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSTM_ADDR_Y [31:0]	0h	RW	Destination Image Y/RGB Plane Storing Address These bits specify in 1-byte units the address for storing the destination-image Y plane or packed RGB plane to be written to by the WPF0 in the method described later. A value from 0000_0000h to FFFF_FFFFh can be specified.

[Destination Address Specification Method]

When rotation/flipping is not used, the start addresses of a frame (address FHA shown in **Figure 9.9-23**) should be specified as the destination address. When flipping/rotation is used, the destination address is not the frame start address (FHA); one of addresses A0 to A3 shown in **Figure 9.9-23** should be selected according to the combination of desired rotation and flipping (VI6_WPF0_OUTFMT.ROT setting).

To strictly define locations A0 to A3, let the horizontal size of the output image after rotation be H, the vertical size of the output image after rotation be V, and the memory stride (VI6_WPF0_DSTM_STRIDE_Y/C setting) be S as shown in **Figure 9.9-23**. Calculate the destination address (one of A0 to A3) using the formula shown in **Table 9.9-17** and specify it in the destination address storing register.

The values of variables N, P, and L in **Table 9.9-17** depend on the other register settings and luminance and chrominance components. These values should be obtained by referring to **Table 9.9-18** to **Table 9.9-20** when calculating the address to be specified in VI6_WPF0_DSTM_ADDR_Y, or **Table 9.9-18**, **Table 9.9-21**, and **Table 9.9-22** when calculating the address to be specified in VI6_WPF0_DSTM_ADDR_C0 or VI6_WPF0_DSTM_ADDR_C1.

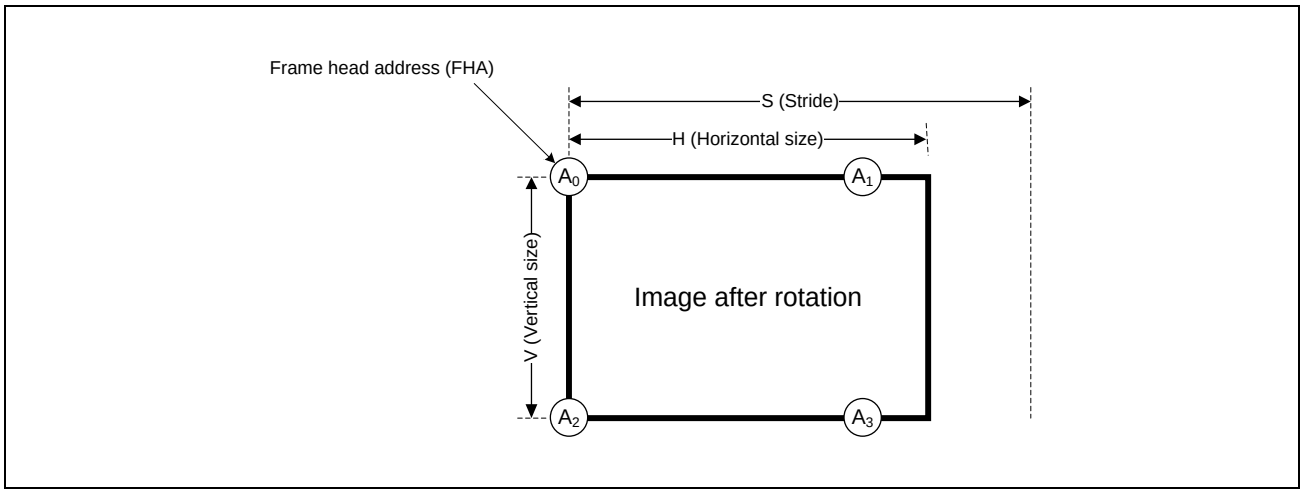


Figure 9.9-23 Location of Destination Address to be Specified for Rotation/Flipping

Table 9.9-17 Destination Address A_0, A_1, A_2, and A_3 Calculation Formulas

VI6_WPF0_OUTFMT.ROT Setting	Formula for Calculating Address to be Set in VI6_WPF0_DSTM_ADDR_Y, VI6_WPF0_DSTM_ADDR_C0, and VI6_WPF0_DSTM_ADDR_C1
0, 2, or 6	A_0 = FHA
4	A_1 = FHA + clip0 ((H − N) × P)
1, 3, or 7	A_2 = FHA + (V×L – 1) × S
5	A_3 = FHA + (V×L – 1) × S + clip0 ((H – N) × P)

Table 9.9-18 Value of N according to VI6_WPF0_ROT_CTRL.LN16 Setting (Used in Common for RGB, Y, and C)

VI6_WPF0_ROT_CTRL.LN16	N
0	8
1	16

Table 9.9-19 Value of P according to VI6_WPF0_OUTFMT.WRFMT Setting (for RGB and Luminance Y Address Calculation)

VI6_WPF0_OUTFMT.WRFMT	P
0, 64 to 66, or 74 to 76	1
1 to 2, 4 to 6, 25 to 32, or 71 to 72,	2
15 to 18, 21, 24, 33, 70, or 73	3
7 to 14, 19 to 20, 22 to 23, or 34 to 35	4

Table 9.9-20 Value of L according to VI6_WPF0_OUTFMT.WRFMT Setting (for RGB and Luminance Y Address Calculation)

VI6_WPF0_OUTFMT.WRFMT	L
73	0.5
0 to 2, 4 to 35, 64 to 66, 70 to 72, or 74 to 76	1

Table 9.9-21 Value of P according to VI6_WPF0_OUTFMT.WRFMT Setting (for Chrominance C0 and C1 Address Calculation)

VI6_WPF0_OUTFMT.WRFMT	P
70 to 73	Not defined
75 to 76	0.5
65 to 66, 74	1
64	2

Table 9.9-22 Value of L according to VI6_WPF0_OUTFMT.WRFMT Setting (for Chrominance C0 and C1 Address Calculation)

VI6_WPF0_OUTFMT.WRFMT	L
70 to 73	Not defined
66, 76	0.5
64 to 65, 74 to 75	1

(11) WPF0 Destination Chroma Address Registers 0 (VI6_WPF0_DSTM_ADDR_C0)

Access Size : 32 bits
Address : <VSPI0_base> + 1028h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSTM_ADDR_C0[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSTM_ADDR_C0[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSTM_ADDR_C0[31:0]	0h	RW	Destination Image C Plane Storing Address 0 These bits specify in 1-byte units the address for storing the destination-image C plane to be written to by the WPF0. Refer to the description of VI6_WPF0_DSTM_ADDR_Y for settings. Here, the C plane indicates the combined CbCr plane when a semi-planar format is selected from the packed YCbCr formats shown in **Table 9.9-16** or the Cb plane when a planar format is selected. When an interleaved format is selected or the output is in an RGB format, this setting is not used. A value from 0000_0000h to FFFF_FFFFh can be specified. Refer to **Figure 9.9-23** in **9.9.1.3.7** for settings.

(12) WPF0 Destination Chroma Address Registers 1 (VI6_WPF0_DSTM_ADDR_C1)

Access Size : 32 bits
Address : <VSPI0_base> + 102Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSTM_ADDR_C1[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSTM_ADDR_C1[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSTM_ADDR_C1[31:0]	0h	RW	Destination Image C Plane Storing Address 1 These bits specify in 1-byte units the address for storing the Cr plane when the WPF0 outputs images to the external memory in YCbCr planar format shown in **Table 9.9-16**. Refer to the description of VI6_WPF0_DSTM_ADDR_Y for settings. This setting is not used when the WPF outputs in YCbCr format that is not a planar format or in an RGB format. A value from 0000_0000h to FFFF_FFFFh can be specified. Refer to **Figure 9.9-23** in **9.9.1.3.7** for settings.

9.9.1.3.8 DPR Control Registers

(1) Concept of DPR Settings

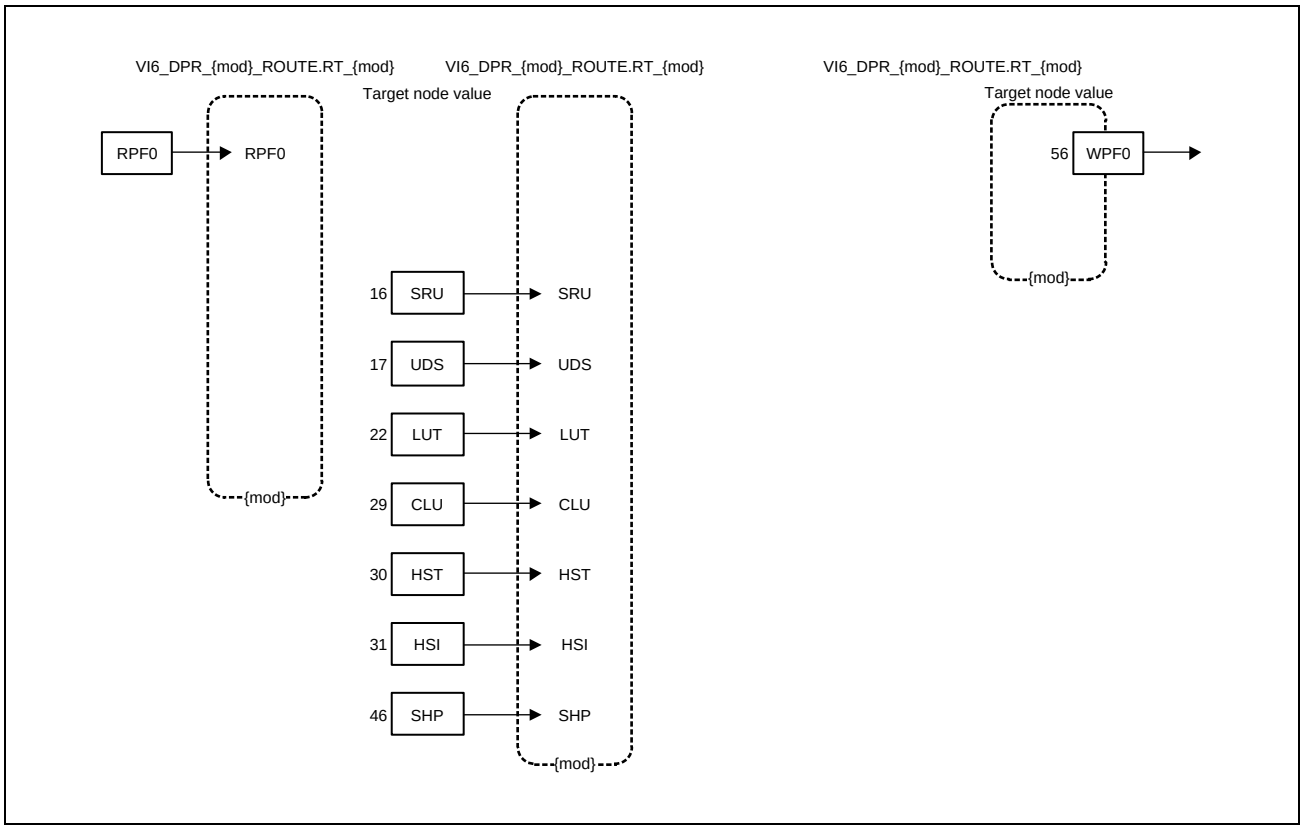


Figure 9.9-24 Node Register Names and Target Node Values on Data Path Router

Remark **Figure 9.9-24** shows node value of all sub modules implemented.

In the VSPI internal data path, the order of processes can be specified as desired. The module for performing each process has a unique node value. Set each bit field in VI6_DPR_*_ROUTE to an appropriate node value shown in **Figure 9.9-24**.

For DPR settings, all of the following restrictions should be observed. If any of them is violated, even the WPF paths operating correctly at that time will be affected as well as the WPF paths connected through the DPR, and correct operation will not be guaranteed (for example, if a restriction is violated for the DPR setting related to WPF0 that are operating correctly will be affected).

1. Specify 63 for the output node values of all RPFs and processing modules that are not used in the DPR. Here, make sure that no module is connected to a module for which 63 is specified as the node value.
2. When specifying a value other than 63 for an output node value in the DPR, make sure that valid inputs (RPF0) and a target WPF are determined.
3. Only one module can be connected to each module; specifying the same target node value for two or more modules is prohibited.
4. Each node can be used only once throughout all paths from RPF0 to WPF0. When a module shown in **Figure 9.9-24** is assigned in one RPF to WPF path, it cannot be used in another RPF to WPF path.

5. While a WPF is operating, modifying the DPR connection settings in VI6_DPR_*_ROUTE is prohibited for modules used by the WPF but allowed for modules not used by the WPF. Be careful not to accidentally modify the settings of the modules included in the WPF path that is operating.

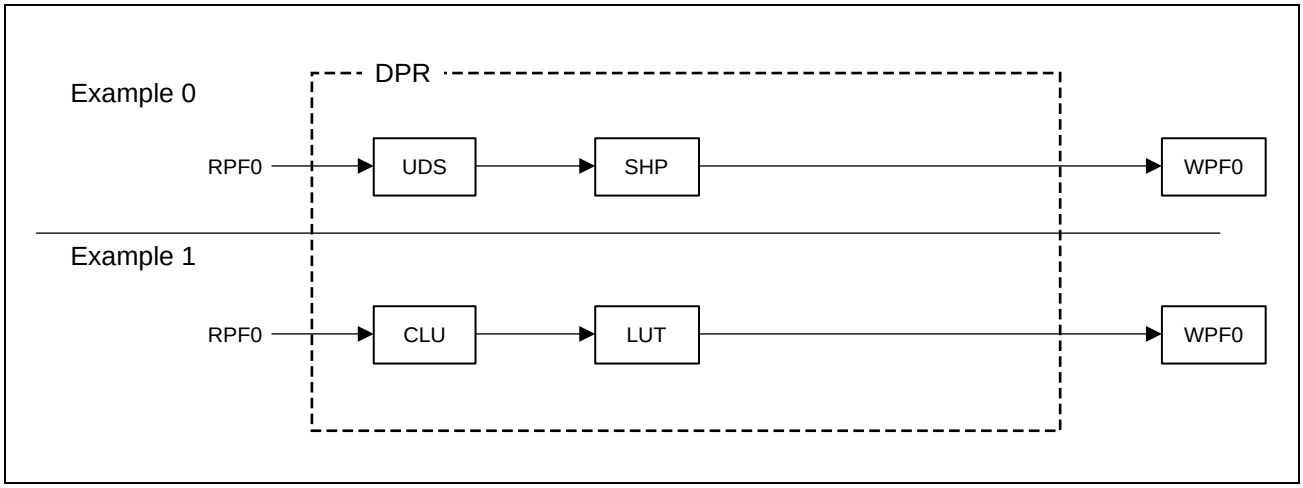


Figure 9.9-25 Examples of Internal Data Path Routing

Figure 9.9-25 shows examples of internal data path routing. Example 0 is WPF0 processing (RPF0 is the source RPF), and example 1 is also WPF0 (RPF0). Each example has the configuration shown in **Figure 9.9-25**. Example 0 performs UDS (up/down scaling) and SHP (sharpness) processing. Example 1 performs CLU and LUT processing (e.g. γ correction), for input 0 (RPF0). The VI6_ DPR_*_ROUTE settings for these examples are shown in **Table 9.9-23**. The bit fields for the modules that are not used in the examples should be set to 63.

Table 9.9-23 VI6_DPR_*_ROUTE Register Settings in Connection Examples Shown in **Figure 9.9-25**

	Register Name	Setting	
Example 0	VI6_DPR_RPF0_ROUTE.RT_RPF0	17	(To UDS)
	VI6_DPR_SRU_ROUTE.RT	63	(UNUSED)
	VI6_DPR_UDS_ROUTE.RT	46	(To SHP)
	VI6_DPR_SHP_ROUTE.RT	56	(To WPF0)
	VI6_DPR_LUT_ROUTE.RT	63	(UNUSED)
	VI6_DPR_CLU_ROUTE.RT	63	(UNUSED)
	VI6_DPR_HST_ROUTE.RT	63	(UNUSED)
	VI6_DPR_HSI_ROUTE.RT	63	(UNUSED)
Example 1	VI6_DPR_RPF0_ROUTE.RT_RPF0	29	(To CLU)
	VI6_DPR_SRU_ROUTE.RT	63	(UNUSED)
	VI6_DPR_UDS_ROUTE.RT	63	(UNUSED)
	VI6_DPR_SHP_ROUTE.RT	63	(UNUSED)
	VI6_DPR_CLU_ROUTE.RT	22	(To LUT)
	VI6_DPR_HST_ROUTE.RT	63	(UNUSED)
	VI6_DPR_HSI_ROUTE.RT	63	(UNUSED)

Although both examples shown in **Figure 9.9-25** include operation in image processing modules, connect the RPF to the WPF directly when only image format conversion or packed format conversion is required.

(2) RPF0 Routing Register (VI6_DPR_RPF0_ROUTE)

Access Size : 32 bits
Address : <VSPI0_base> + 2000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	RT_RPF0[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
5 to 0	RT_RPF0[5:0]	0h	RW	RPF0 Target Node Value These bits specify the target node value for RPF0. When using RPF0, refer to **Figure 9.9-24** for settings. When RPF0 is not started through the VI6_WPF0_SRCRPF setting, specify 63.

(3) WPF0 Timing Control Register (VI6_DPR_WPF0_FPORCH)

Access Size : 32 bits
Address : <VSPI0_base> + 2014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	FP_WPF0[5:0]						-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13 to 8	FP_WPF0[5:0]	0h	RW	WPF0 Internal Operation Timing Setting Specify 5.
7 to 0	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.

(4) {mod} Routing Register (VI6_DPR_{mod}_ROUTE) ({mod} = SRU, UDS, LUT, CLU, HST, HSI, SHP)

Access Size : 32 bits

Address : <VSPI0_base> + 2024h/ 2028h/ 203Ch/ 2040h/ 2044h/ 2048h/ 2060h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	FXA[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	FP[5:0]						-	-	RT[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28	-	0h	RW	Reserved These bits are read as undefined value. The write value should always be 0b.
27 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 16	FXA[7:0]	0h	RW	Fixed α Output Value for {mod} The {mod} does not support input/output of the α value. The α value input to the {SHP, SRU, LUT, CLU, HST and HSI} is discarded, and the fixed α value specified in these bits is always output from the {SHP, SRU, LUT, CLU, HST and HSI}. A value from 0 to 255 can be specified. These bits are valid for SHP, SRU, LUT, CLU, HST and HSI modules. For UDS, modules, these bits are reserved.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13 to 8	FP[5:0]	0h	RW	{mod} Internal Operation Timing Setting Specify 0.
7, 6	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
5 to 0	RT[5:0]	0h	RW	{mod}Target Node Value These bits specify the target node value for the {mod}. When using the {mod}, refer to **Figure 9.9-24** for settings. When not using the {mod}, specify 63.

(5) {mod} Sampling Point Register (VI6_DPR_{mod}_SMPPT) ({mod} = HGO, HGT)

Access Size : 32 bits
Address : <VSPI0_base> + 2054h/ 2058h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	TGW[2:0]			-	-	PT[5:0]					
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
10 to 8	TGW[2:0]	0h	RW	Target WPF Index for HGO These bits are used for specifying the target WPF index of HGO. If HGO is not used, set 7 to these bits.
7, 6	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
5 to 0	PT[5:0]	0h	RW	Target Node Index for HGO Histogram Sampling HGO generates the histogram for the target node specified by these bits. For example, if these bits are set to 0, HGO module generates the histogram for the output data of RPF0. Refer to below table for the target node index which can be used for HGO. If HGO is not used, set 63 to these bits.

Table 9.9-24 Target Node Index for {mod} Module

Node Index	Module for Histogram Generation
0	RPF0
16	SRU
17	UDS
22	LUT
29	CLU
30	HST
31	HSI
46	SHP

9.9.1.3.9 SRU Control Registers

(1) Super Resolution Control Register 0 (VI6_SRU_CTRL0)

Access Size : 32 bits
Address : <VSPI0_base> + 2200h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	SRU_PARAM0[8:0]								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SRU_PARAM1[4:0]					-	SRU_MODE[2:0]			SRUPARAM2	SRUPARAM3	SRUPARAM4	SRU_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
24 to 16	SRU_PARAM0 [8:0]	0h	RW	Super Resolution Parameter 0 Specify an appropriate value shown in **Table 9.9-25**; do not specify any other value.
15 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12 to 8	SRU_PARAM1 [4:0]	0h	RW	Super Resolution Parameter 1 Specify an appropriate value shown in **Table 9.9-25**; do not specify any other value.
7	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.
6 to 4	SRU_MODE [2:0]	0h	RW	Super Resolution Mode Setting These bits specify a super resolution mode. In super resolution without scaling, the output size is the same as the input size. In super resolution with double scale-up, the output size is twice the input size. 000b: Super resolution without scaling 100b: Super resolution with double scale-up Other settings are prohibited.
3	SRUPARAM2	0h	RW	Super Resolution Parameter 2 This parameter setting depends on the color space of the image input to the SRU. Specify an appropriate value shown in **Table 9.9-26**; do not specify any other value.
2	SRUPARAM3	0h	RW	Super Resolution Parameter 3 Specify an appropriate value shown in **Table 9.9-26**; do not specify any other value.
1	SRUPARAM4	0h	RW	Super Resolution Parameter 4 This parameter setting depends on the color space of the image input to the SRU. Specify an appropriate value shown in **Table 9.9-26**; do not specify any other value.
0	SRU_EN	0h	RW	Super Resolution Processing Enable/Disable Enables or disables super resolution processing. This setting has the highest priority over all other SRU registers. 0b: Super resolution processing is disabled (input goes through the SRU without change) 1b: Super resolution processing is enabled

YCbCr format is recommended for the super resolution processing. If the super resolution processing is applied to an RGB-format image, adverse effects such as color blur may be generated; when using an RGB format, evaluate the image quality carefully before applying the processing to practical use.

To execute the super resolution processing in YCbCr format, set up the RPF and DPR appropriately so that YCbCr-format image data is input to the SRU. Specifically, make the RPF0 output YCbCr-format image data and send the output data to the SRU through the DPR. For details of RPF0 input/output settings, refer to **9.9.1.3.6 (3) RPF0 Input Format Registers (VI6_RPF0_INFMT)**. For DPR settings, refer to **9.9.1.3.8 (1) Concept of DPR Settings** to

9.9.1.3.8 (4) {mod} Routing Register (VI6_DPR_{mod}_ROUTE) ({mod} = SRU, UDS, LUT, CLU, HST, HSI, SHP).

Table 9.9-25 Super Resolution Parameter Setting 1

Intensity		VI6_SRU_CTRL0			VI6_SRU_CTRL1	VI6_SRU_CTRL2		
		SRU_MODE	SRU_PARAM0	SRU_PARAM1	SRU_PARAM5	SRU_PARAM6	SRU_PARAM7	SRU_PARAM8
Weak	1	0 or 4	256	4	7FFh	24	40	255
	2	0 or 4	256	4	7FFh	8	16	255
	3	0 or 4	384	5	7FFh	36	60	255
	4	0 or 4	384	5	7FFh	12	27	255
	5	0 or 4	511	6	7FFh	48	80	255
Strong	6	0 or 4	511	6	7FFh	16	36	255

Table 9.9-26 Super Resolution Parameter Setting 2

Color Space of SRU Input Image	SRU_PARAM2	SRU_PARAM3	SRU_PARAM4
YCbCr*1	0	1	0
RGB	1	1	1

Note 1. YCbCr format is recommended for the image input to the SRU.

(2) Super Resolution Control Register 1 (VI6_SRU_CTRL1)

Access Size : 32 bits
Address : <VSPI0_base> + 2204h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	SRU_PARAM5[10:0]										
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
10 to 0	SRU_PARAM5 [10:0]	0h	RW	Super Resolution Parameter 5 Specify an appropriate value shown in **Table 9.9-25**; do not specify any other value.

(3) Super Resolution Control Register 2 (VI6_SRU_CTRL2)

Access Size : 32 bits
Address : <VSPI0_base> + 2208h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	SRU_PARAM6[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRU_PARAM7[7:0]								SRU_PARAM8[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 16	SRU_PARAM6 [7:0]	0h	RW	Super Resolution Parameter 6 Specify an appropriate value shown in **Table 9.9-25**; do not specify any other value.
15 to 8	SRU_PARAM7 [7:0]	0h	RW	Super Resolution Parameter 7 Specify an appropriate value shown in **Table 9.9-25**; do not specify any other value.
7 to 0	SRU_PARAM8 [7:0]	0h	RW	Super Resolution Parameter 8 Specify an appropriate value shown in **Table 9.9-25**; do not specify any other value.

9.9.1.3.10 UDS Control Registers

(1) Scaling Control Registers (VI6_UDS_CTRL)

Access Size : 32 bits
Address : <VSPI0_base> + 2300h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	AMD	FMD	BLADV	-	-	AON	ATHON	-	-	-	BC	NE_A	NE_RCR	NE_GY	NE_BCB
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	R	R	RW	RW	R	R	R	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	AMDSLH	EXT (TDIPC)	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.
30	AMD	0h	RW	Pixel Count at Scale-Up Specifies the number of pixels generated through scale-up in the UDS. This bit setting is ignored for scale-down. 0b: Pixel count after scale-up is 1 + < (n - 1) × scale-up factor > 1b: Pixel count after scale-up is 1 + < n × scale-up factor > *n: Number of pixels input to UDS
29	FMD	0h	RW	Padding for Insufficient Clipping Size When the scaling filter outputs an image that is smaller than the clipping size (VI6_UDS_CLIP_SIZE), pixels are interpolated to match the clipping size. This bit specifies the pixel filling method. 0b: Pixels are filled by copying pixels at the right edge and the bottom edge 1b: Pixels are filled with the color specified by VI6_UDS_FILL_COLOR
28	BLADV	0h	RW	Bilinear or Nearest Neighbor Interpolation Characteristic Control Controls the characteristics of bilinear or nearest neighbor interpolation. Setting this bit to 1 improves the aliasing characteristics at ×1/2 to 1/8 scaling (VI6_UDS_SCALE.VMANT/HMANT = 2 to 8). Note that the apparent resolution around edges may deteriorate. In multi-tap mode, this control is not available; to use this control, set the BC bit to 0.
27, 26	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
25	AON	0h	RW	Scale-Up/Down of α Plane These bits specify whether to enable or disable scale-up/-down of the α plane when scaling up/down in the RGB format. When the AON bit is set to 0, the UDS outputs the value of the VI6_UDS_ALPVAL. ALPH_VAL0 bits as a fixed α value. 0b: α scale-up/-down is not performed 1b: α scale-up/-down is performed
24	ATHON	0h	RW	α Output Data Threshold Comparison Enable/Disable Enables or disables comparison with the α output data threshold. When this bit is 1, the output α value is replaced according to the VI6_UDS_ALPTH and VI6_UDS_ALPVAL values. When the AON bit is 0 (scale-up/-down of the α plane is disabled), this bit setting has no effect. 0b: α output data threshold comparison is disabled 1b: α output data threshold comparison is enabled
23 to 21	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
20	BC	0h	RW	Pixel Component Interpolation Method at Scale-Up/Down Specifies the method for interpolating pixel components at scale-up/-down. 0b: Bilinear or nearest neighbor interpolation method is used 1b: Interpolation method equivalent to 4 to 17 taps in accordance with the scaling factor is used (multi-tap mode) Setting 1 to BC bit (Multi tap mode) is recommended. Note that the α component is interpolated by the method specified in the NE_A bit instead of the multi-tap method. When the BC and AON bits are both set to 1, the scaling shall be scale-up, no scaling, or scale-down with scaling factor 1/1 to 1/2.
19	NE_A	0h	RW	α Interpolation Method Specifies the interpolation method of the α plane. When the AON bit is 0 (scale-up/-down of the α plane is disabled), this bit setting has no effect. 0b: Bilinear method 1b: Nearest neighbor method*1
18	NE_RCR	0h	RW	α Interpolation Method Specifies the interpolation method of the α plane. When the AON bit is 0 (scale-up/-down of the α plane is disabled), this bit setting has no effect. 0b: Bilinear method 1b: Nearest neighbor method*1
17	NE_GY	0h	RW	G/Y Interpolation Method When Bilinear/Nearest Neighbor Interpolation is Selected Specifies the interpolation method of the G/Y component when bilinear/nearest neighbor interpolation is selected (BC = 0). 0b: Bilinear method 1b: Nearest neighbor method*1
16	NE_BCB	0h	RW	B/Cb Interpolation Method When Bilinear/Nearest Neighbor Interpolation is Selected Specifies the interpolation method of the B/Cb component when bilinear/nearest neighbor interpolation is selected (BC = 0). 0b: Bilinear method 1b: Nearest neighbor method*1
15 to 3	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
2	AMDSLH	0h	RW	Horizontal Filter Phase 0b: The horizontal start and end phase is automatically determined by the setting of AMD. In this case, the value of VI6_UDS_HPHASE.HSTP/HEDP is not valid. 1b: The horizontal start and end phase is determined by the value of VI_UDS_HPHASE.HSTP/HEDP.
1	EXT (TDIPC)	0h	RW	Extended UDS function Enable/Disable Select 0b: Extended UDS function Disable 1b: 2D IPC function or low pass filter Enable UDS execute 2D IPC function based on this bit and the setting of VI6_UDS_IPC register. UDS execute low pass filter based on this bit and the setting of VI6_UDS_IPC register. See **9.9.1.3.10 (7)**, 2D IPC Setting Register (VI6_UDS_IPC) for more details.
0	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.

Note 1. This method can be used only when the scale-up/-down factor is 1/1 to 1/4.

(2) Scaling Factor Registers (VI6_UDS_SCALE)

Access Size : 32 bits
Address : <VSPI0_base> + 2304h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HMANT[3:0]				HFRAC[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VMANT[3:0]				VFRAC[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	HMANT[3:0]	0h	RW	Multiplier (Integral Part) of Horizontal Scaling Factor These bits specify the integral part of the horizontal scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size. A value from 0h to Fh can be specified. Select a value within the range shown in **Table 9.9-27**.
27 to 16	HFRAC[11:0]	0h	RW	Multiplier (Fractional Part) of Horizontal Scaling Factor These bits specify the fractional part of the horizontal scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size. A value from 100h to FFFh can be specified when an image is up-scaled (the HMANT value is 0) in the horizontal direction. A value from 000h to FFFh can be specified when an image is downscaled (the HMANT value is not 0) in the horizontal direction. Select a value within the range shown in **Table 9.9-27**.
15 to 12	VMANT[3:0]	0h	RW	Multiplier (Integral Part) of Vertical Scaling Factor These bits specify the integral part of the vertical scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size. A value from 0h to Fh can be specified. Select a value within the range shown in **Table 9.9-27**.
11 to 0	VFRAC[11:0]	0h	RW	Multiplier (Fractional Part) of Vertical Scaling Factor These bits specify the fractional part of the vertical scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size. A value from 100h to FFFh can be specified when an image is up-scaled (the VMANT value is 0) in the vertical direction. A value from 000h to FFFh can be specified when an image is downscaled (the VMANT value is not 0) in the vertical direction. Select a value within the range shown in **Table 9.9-27**.

The HMANT and HFRAC bits set the scale-up/-down factor for an image in the horizontal direction, and the VMANT and VFRAC bits set the scale-up/-down factor for an image in the vertical direction. The UDS operation switches between horizontal scale-up and horizontal scale-down according to the HMANT and HFRAC bit settings, as shown in **Table 9.9-27**. (Setting a value outside the ranges of **Table 9.9-27** in UDS operation is prohibited). **Table 9.9-27** is for the horizontal direction, but it is similar for the vertical direction. In this case, replace HMANT with VMANT and HFRAC with VFRAC when reading.

Note that settings for scaling in the horizontal direction and settings for scaling in the vertical direction can be made independently. Therefore, a setting for scale-up in the horizontal direction and scale-down in the vertical direction is possible. In such a case, because the formula (described later) for obtaining the image size after scale-up/-down is different between scale-up and scale-down, a formula matching the scale-up or scale-down operation should be selected independently for the horizontal direction and vertical direction.

Table 9.9-27 Switching of Horizontal Scale-Up/Down Operation According to HMANT and HFRAC Bit Settings

HMANT	HFRAC	UDS Operation
0h	100h to FFFh	Scale-up
1h	000h	Same size (no scale-up/-down)
	001h to FFFh	Scale-down
2h to Fh	000h to FFFh	

Described here is the method for calculating the size of the up-scaled/downscaled image that was obtained based on this register setting. First, define the variables necessary for calculating the horizontal size of the up-scaled/downscaled image, as shown below.

$$hscale = \frac{4096}{4096 \times m_h \times f_h}$$

m_h is the value of VI6_UDS_SCALE.HMANT, and f_h is the value of VI6_UDS_SCALE.HFRAC. This formula expresses the estimate of the scale-up/-down factor processed by the UDS. If the horizontal size of the image before scale-up/-down is set as $hsize_{org}$, the horizontal size of the image after scale-up/-down can be roughly obtained through $hsize_{org} \times hscale$.

Similarly, define the variables necessary for calculating the vertical size of the up-scaled/down-scaled image.

$$vscale = \frac{4096}{4096 \times m_v \times f_v}$$

When setting mv as the value of VI6_UDS_SCALE.VMANT, f_v as the value of VI6_UDS_SCALE.VFRAC, and $vsize_{org}$ as the vertical size of the image before scale-up/-down, the vertical size of the image after scale-up/-down can be roughly obtained through $vsize_{org} \times vscale$.

When the UDS performs scale-down with the settings of **Table 9.9-27**, using the variables defined so far, the horizontal size of the downscaled image $hsize_{down_scaled}$ and the vertical size of the downscaled image $vsize_{down_scaled}$ become as follows:

$$hsize_{down_scaled} = \left\langle 1 + \left(\left(\left\langle 1 + \frac{\langle hsize_{org} - 1 \rangle}{m_h{}'} \right\rangle - 1 \right) \times m_h{}' \times 4096 + hstp + hedp \right) / (4096 \times m_h + f_h) \right\rangle$$

$$vsize_{down_scaled} = \left\langle 1 + \left(\left(\left\langle 1 + \frac{\langle vsize_{org} - 1 \rangle}{m_v{}'} \right\rangle - 1 \right) \times m_v{}' \times 4096 \right) / (4096 \times m_v + f_v) \right\rangle$$

$hstp$ is the value of VI6_UDS_HPHASE.HSTP, $hedp$ is the value of VI6_UDS_HPHASE.HEDP.

The value of $m_h{}'$ or $m_v{}'$, which is shown in **Table 9.9-28**, changes according to the setting of VI6_UDS_SCALE.HMANT or VI6_UDS_SCALE.VMANT and VI6_UDS_CTRL.BLADV.

Table 9.9-28 m_h' or m_v' Setting

VI6_UDS_SCALE.HMANT Setting (VI6_UDS_SCALE.VMANT Setting)	VI6_UDS_CTRL.BLADV	m_h' (m_v')
0 to 3	0	1
4 to 7	0	2
8 to 15	0	4
2 to 3	1	2
4 to 7	1	4
0, 1, 8 to 15	1	Forbidden

When the UDS performs scale-up with the settings of **Table 9.9-27** and VI6_UDS_CTRL.AMD is 0, the horizontal size of the up-scaled image $hsize_{up_scaled}$ and the vertical size of the up-scaled image $vsize_{up_scaled}$ become as follows:

$$hsize_{up_scaled} = \left\langle 1 + \frac{(hsize_{org} - 1) \times 4096 + hstp + hedp}{4096 \times m_h + f_h} \right\rangle$$

$$vsize_{up_scaled} = \left\langle 1 + \frac{(vsize_{org} - 1) \times 4096}{4096 \times m_v + f_v} \right\rangle$$

When VI6_UDS_CTRL.AMD is 1, the horizontal size of the up-scaled image $hsize_{up_scaled}$ and the vertical size of the up-scaled image $vsize_{up_scaled}$ become as follows:

$$hsize_{up_scaled} = \langle hsize_{org} \times hscale \rangle$$

$$vsize_{up_scaled} = \langle vsize_{org} \times vscale \rangle$$

After considering the division of hscale and vscale, the formulas become as follows:

$$hsize_{up_scaled} = \left\langle \frac{hsize_{org} \times 4096}{4096 \times m_h + f_h} \right\rangle$$

$$vsize_{up_scaled} = \left\langle \frac{vsize_{org} \times 4096}{4096 \times m_v + f_v} \right\rangle$$

(3) α Data Threshold Setting Registers (VI6_UDS_ALPTH)

Access Size : 32 bits

Address : <VSPI0_base> + 2308h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALPH_TH1[7:0]								ALPH_TH0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
15 to 8	ALPH_TH1[7:0]	0h	RW	α Data Threshold Setting 1 When the α value is equal to or greater than the value of the ALPH_TH1 bits, the α value is replaced with that of VI6_UDS_ALPVAL.ALPH_VAL2. When VI6_UDS_CTRL.AON is 0 (scale-up/-down of the α plane is disabled), the setting of these bits has no effect. A value from 00h to FFh can be specified.
7 to 0	ALPH_TH0[7:0]	0h	RW	α Data Threshold Setting 0 When the α value is equal to or smaller than the value of the ALPH_TH0 bits, the α value is replaced with that of VI6_UDS_ALPVAL.ALPH_VAL0. When VI6_UDS_CTRL.AON is 0 (scale-up/-down of the α plane is disabled), the setting of these bits has no effect. A value from 00h to FFh can be specified.

(4) α Data Replacing Value Setting Registers (VI6_UDS_ALPVAL)

Access Size : 32 bits
Address : <VSPI0_base> + 230Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	ALPH_VAL2[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALPH_VAL1[7:0]								ALPH_VAL0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 16	ALPH_VAL2 [7:0]	0h	RW	Replacing α Value Setting 2 These bits set a value that replaces the α value when it is equal to or greater than the value of VI6_UDS_ALPTH.ALPH_TH1. When VI6_UDS_CTRL.AON is 0 (scale-up/-down of the α plane is disabled), the setting of these bits has no effect. A value from 00h to FFh can be specified.
15 to 8	ALPH_VAL1 [7:0]	0h	RW	Replacing α Value Setting 1 These bits set a value that replaces the α value when it is greater than the value of VI6_UDS_ALPTH.ALPH_TH0 and also smaller than that of VI6_UDS_ALPTH.ALPH_TH1. When VI6_UDS_CTRL.AON is 0 (scale-up/-down of the α plane is disabled), the setting of these bits has no effect. A value from 00h to FFh can be specified.
7 to 0	ALPH_VAL0 [7:0]	0h	RW	Replacing α Value Setting 0 These bits set a value that replaces the α value when it is equal to or smaller than the value of VI6_UDS_ALPTH.ALPH_TH0. When VI6_UDS_CTRL.AON is 0 (scale-up/-down of the α plane is disabled), α output value for the UDS is the fixed value specified in these bits. A value from 00h to FFh can be specified.

(5) Passband Registers (VI6_UDS_PASS_BWIDTH)

Access Size : 32 bits
Address : <VSPI0_base> + 2310h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	BWIDTH_H[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	BWIDTH_V[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
22 to 16	BWIDTH_H [6:0]	0h	RW	Horizontal Signal Passband at Image Scale-Up/Down Set these bits following the method described later.
15 to 7	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
6 to 0	BWIDTH_V [6:0]	0h	RW	Vertical Signal Passband at Image Scale-Up/Down Set these bits following the method described later.

The method for setting the passband in the horizontal direction of an image is described. When the VI6_UDS_SCALE.HMANT bits for horizontal scale-up/-down factor setting are not 0, set the BWIDTH_H bits according to the following formula. When the VI6_UDS_SCALE.HMANT bits are 0, set 64 in the BWIDTH_H bits.

$$\mathrm{BWIDTH_H} \left\langle 64 \times \frac{4096 \times {m_h}'}{4096 \times m_h + f_h} \right\rangle \quad (\mathrm{VI6_UDS_SCALE.HMANT} \neq 0)$$

$$\mathrm{BWIDTH_H} = 64 \quad (\mathrm{VI6_UDS_SCALE.HMANT} = 0)$$

m_h is the value of VI6_UDS_SCALE.HMANT, and f_h is the value of VI6_UDS_SCALE.HFRAC. For the ${m_h}'$ value, see **Table 9.9-28**. The method for setting the passband in the vertical direction of an image is similar to that for the horizontal direction described earlier. Since only the correspondence relationship of the registers is changed as shown below, replace the variables in the previous explanation as shown below when reading.

BWIDTH_H → BWIDTH_V

${m_h}' \rightarrow {m_v}'$

$m_h \rightarrow m_v$

$f_h \rightarrow f_v$

VI6_UDS_SCALE.HMANT → VI6_UDS_SCALE.VMANT

VI6_UDS_SCALE.HFRAC → VI6_UDS_SCALE.VFRAC

(6) Scaling Filter Horizontal Phase Registers (VI6_UDS_HPHASE)

Access Size : 32 bits
Address : <VSPI0_base> + 2314h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	HSTP[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	HEDP[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
27 to 16	HSTP [11:0]	0h	RW	Horizontal Starting Phase
15 to 12	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
11 to 0	HEDP [11:0]	0h	RW	Horizontal Ending Phase

This register can be used for the image division mode. When this register is not used, set VI6_UDS_CTRL.AMDSLH to 0, and set HSTP and HEDP to 0. When the value of HSTP or HEDP is not equal to 0, the size is different from the value shown in **9.9.1.3.10 (2) Scaling Factor Registers (VI6_UDS_SCALE)**.

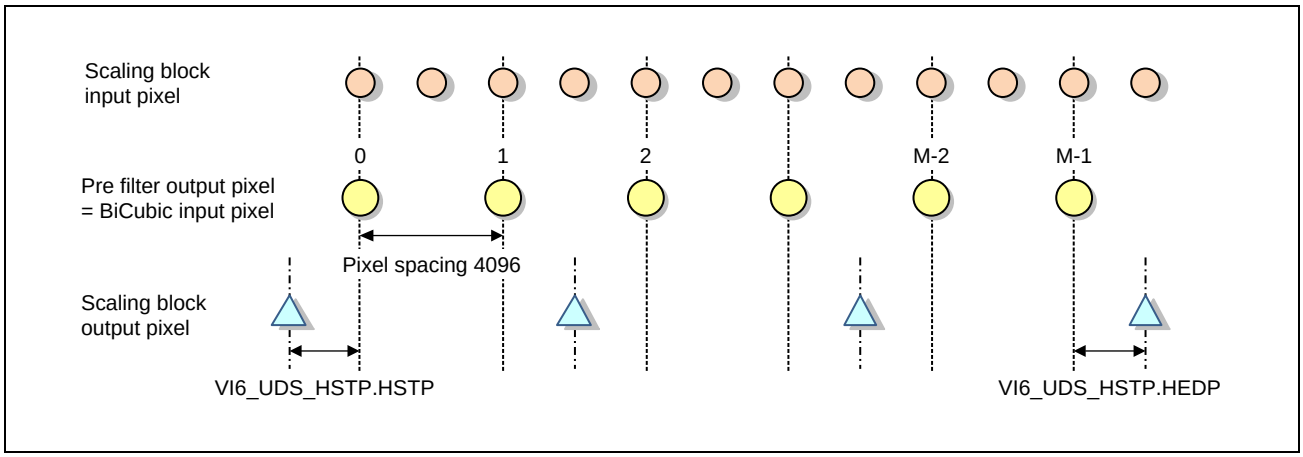


Figure 9.9-26 Horizontal Starting and Ending Phase

(7) 2D IPC Setting Register (VI6_UDS_IPC)

Access Size : 32 bits
Address : <VSPI0_base> + 2318h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	FIELD	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LPFE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
27	FIELD	0h	RW	Top/Bottom Field Select Select Top/Bottom Field of image data when 2D IPC function is enabled (VI6_UDSn_CTRL.EXT = 1). 0b: Top Field 1b: Bottom Field
26 to 1	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
0	LPFE	0h	RW	Low Pass Filter Enable 0b: low pass filter disable 1b: low pass filter enable

IPC setting is shown below.

1. Set 1 to VI6_UDSn_CTRL.EXT.
2. Set UDS register as double up-scale in vertical direction and no scaling in horizontal direction.
3. Set appropriate setting of processing image.

   Top Field: 0000_0000h
   Bottom Field: 0800_0000h
4. Set 0 to VI6_UDS_CTRL.FMD
5. Set 1 to VI6_UDS_CTRL.BC

Low pass filter setting is show in below.

1. Set 1 to VI6_UDS0_CTRL.EXT and set 1 to VI6_UDS0_CTRL.BC
2. Set equal setting (×1) to VI6_UDS_SCALE
3. Set 0 - 64 to VI6_UDS_PASS_BWIDTH. BWIDTH_H for low pass filter in horizontal direction.
4. Set 0 - 64 to VI6_UDS_PASS_BWIDTH. BWIDTH_V for low pass filter in vertical direction.

Strength of low pass filter can be set by VI6_UDS_PASS_BWIDTH.

When UDS is used for IPC processing, low pass filter cannot be enabled.

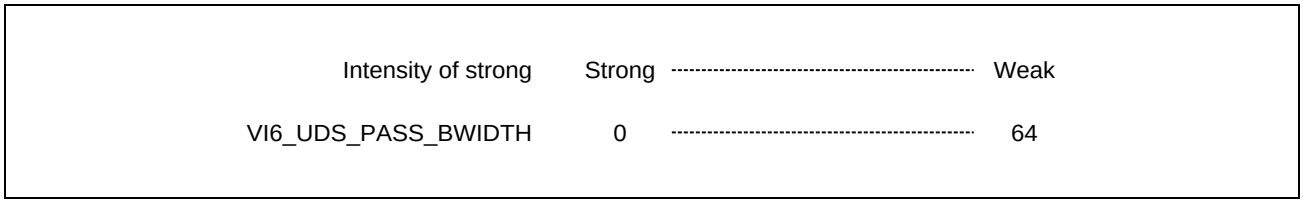


Figure 9.9-27 Strength of Low Pass Filter and Setting of Pass Band Width

(8) UDS Horizontal Input Clipping Registers (VI6_UDS_HSZCLIP)

Access Size : 32 bits
Address : <VSPI0_base> + 231Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	HCEN	-	-	-	-	FCL_OFST[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	HCL_SIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28	HCEN	0h	RW	Horizontal Input Size Clipping Enable This bit is used for setting the horizontal size clipping of UDS. 0b: Disable Horizontal Size Clipping 1b: Enable Horizontal Size Clipping
27 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 16	FCL_OFST [7:0]	0h	RW	Horizontal Offset Value for Input Size Clipping
15 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12 to 0	HCL_SIZE [12:0]	0h	RW	Horizontal Clipping Size

This register can be used for the image division mode. If the image partition mode is used, contact to RENESAS for detail.

(9) UDS Output Size Clipping Registers (VI6_UDS_CLIP_SIZE)

Access Size : 32 bits
Address : <VSPI0_base> + 2324h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	CL_HSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	CL_VSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28 to 16	CL_HSIZE [12:0]	0h	RW	Clipping Size of Horizontal Pixel Count after Scale-Up/-Down The horizontal width of an image output from the scaling filter is adjusted (clipped or padded) to match the pixel count set in the CL_HSIZE bits. The setting range is 4 to 2048 in a scale-down operation (see **Table 9.9-27**) and 4 to 8190 in a scale-up operation. These bits always have to be set when using the UDS, regardless of the scale-up, scale-down, or no-scaling setting by the VI6_UDS_SCALE register.
15 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12 to 0	CL_VSIZE [12:0]	0h	RW	Clipping Size of Vertical Pixel Count after Scale-Up/-Down The vertical width of an image output from the scaling filter is adjusted (clipped or padded) to match the pixel count set in the CL_VSIZE bits. The setting range is 4 to 2048 in a scale-down operation (see **Table 9.9-27**) and 4 to 8190 in a scale-up operation. These bits always have to be set when using the UDS, regardless of the scale-up, scale-down, or no-scaling setting by the VI6_UDS_SCALE register.

Figure 9.9-28 shows the configuration of the UDS. The UDS consists of a scaling filter and clipping circuit, such as the configuration shown in **Figure 9.9-28**. The scaling filter and clipping circuit are independent of each other.

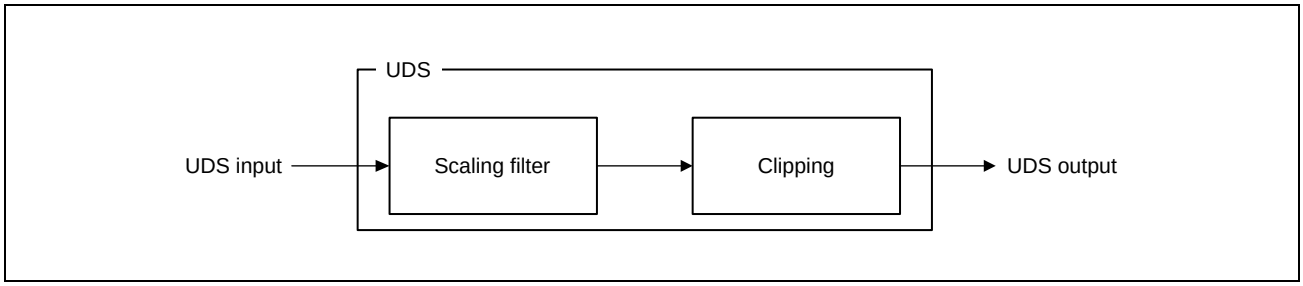


Figure 9.9-28 UDS Configuration

The size of the image actually output by the scaling filter (refer to **9.9.1.3.10 (2) Scaling Factor Registers (VI6_UDS_SCALE)** for calculation) is determined from the size of the image input to the scaling filter and the VI6_UDS_SCALE setting; that is, three types of image are output by the UDS according to the relationship between the size of the image actually output from the scaling filter ($hsize_{scaled}$ and $vsize_{scaled}$) and this register setting as shown in **Figure 9.9-29**.

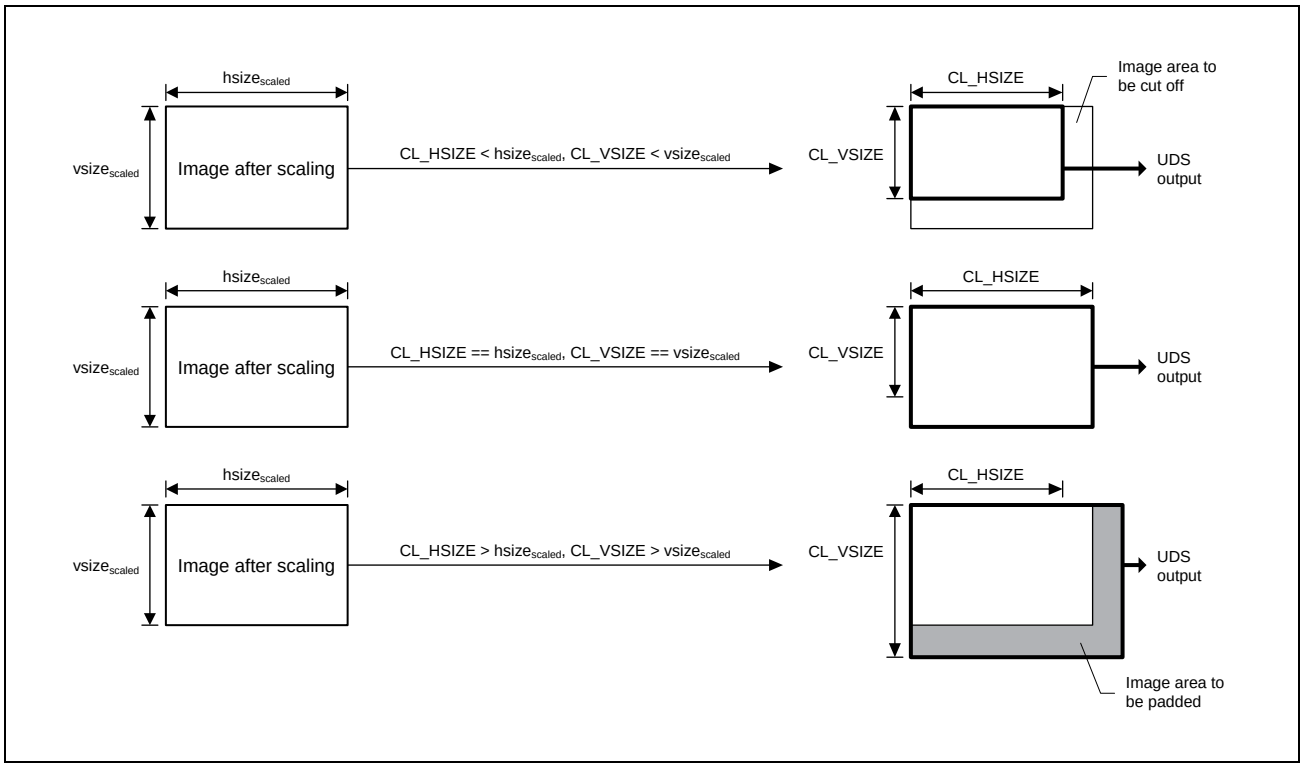


Figure 9.9-29 UDS Output Image for Each CL_HSIZE/VSIZE Setting

When the settings of the CL_HSIZE and CL_VSIZE bits are smaller than the horizontal and vertical pixel counts ($hsize_{scaled}$ and $vsize_{scaled}$) actually output by the scaling filter, the up-scaled/down-scaled image is clipped to become the UDS output image.

When the settings of the CL_HSIZE and CL_VSIZE bits are equal to the horizontal and vertical pixel counts ($hsize_{scaled}$ and $vsize_{scaled}$) actually output by the scaling filter, the image actually output by the scaling filter becomes the UDS output image without change.

When the settings of the CL_HSIZE and CL_VSIZE bits are greater than the horizontal and vertical pixel counts ($hsize_{scaled}$ and $vsize_{scaled}$) actually output by the scaling filter, in order to obtain the UDS output image, the image is padded in the mode set by VI6_UDS_CTRL.FMD for the area in which the settings of the CL_HSIZE and CL_VSIZE bits have exceeded the number of pixels actually output by the scaling filter.

(10) Color Fill Register (VI6_UDS_FILL_COLOR)

Access Size : 32 bits
Address : <VSPI0_base> + 2328h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	RFILC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GFILC[7:0]								BFILC[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 16	RFILC [7:0]	0h	RW	R/Cr Component of Fill Color A value from 00h to FFh can be specified.
15 to 8	GFILC [7:0]	0h	RW	G/Y Component of Fill Color A value from 00h to FFh can be specified.
7 to 0	BFILC [7:0]	0h	RW	B/Cb Component of Fill Color A value from 00h to FFh can be specified.

The scaling filter creates an up-scaled/down-scaled image based on the scaling factor settings of the VI6_UDS_SCALE register. If the size of the up-scaled/down-scaled image created by the scaling filter smaller than the clipping size set in the VI6_UDS_CLIP_SIZE register (lowest case in **Figure 9.9-29**) while VI6_UDS_CTRL.FMD is set to 1, the color specified by this register is used to fill the space until the clipping size is reached.

The α value of the area to be padded by this register is dependent on VI6_UDS_CTRL.FMD setting. When VI6_UDS_CTRL.FMD is 0 (repetition), the pixel value at the right edge (bottom edge) of the image is repeated as the α value. When VI6_UDS_CTRL.FMD is 1 (color information of this register is used), the α value is 0.

9.9.1.3.11 LUT Control Register

(1) LUT Control Register (VI6_LUT_CTRL)

Access Size : 32 bits
Address : <VSPI0_base> + 2800h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LUT_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
0	LUT_EN	0h	RW	1D-LUT Enable/Disable Enables or disables the 1D-LUT function by the LUT. When the 1D-LUT is used, the color component curve information needs to be set separately in the LUT table. For the LUT table settings, refer to **9.9.1.4.5 (1)**. 0b: 1D-LUT function is disabled 1b: 1D-LUT function is enabled

In the LUT, various image processing, such as curves with high operation load (e.g., γ correction), negative-positive conversion, and gain adjustment of images, can be achieved by the data replacement processing by the 1D-LUT. As shown in **Figure 9.9-30**, the LUT replaces each component of the input pixel data using the set replacement table of 256 entries. For example, if the LUT is set as in **Figure 9.9-31**, when there is an input of 150, the data stored in address 150 of the 1D-LUT is read and output as the LUT output. **Figure 9.9-31** shows a case in which the input and output become equal for convenience in explaining.

In the LUT settings shown in **Figure 9.9-32** the input bits are reversed. This has the effect of negative-positive flipping. In **Figure 9.9-33**, γ correction (γ = 1.8 is shown as an example) is possible. As described above, information to be set in the LUT indicates LUT processing characteristics. If the same value is set for each component in the LUT, an equal effect can be obtained for each component of the input image when it is processed. If the LUT is set with different characteristics for each component, the processing characteristics can be changed for each component.

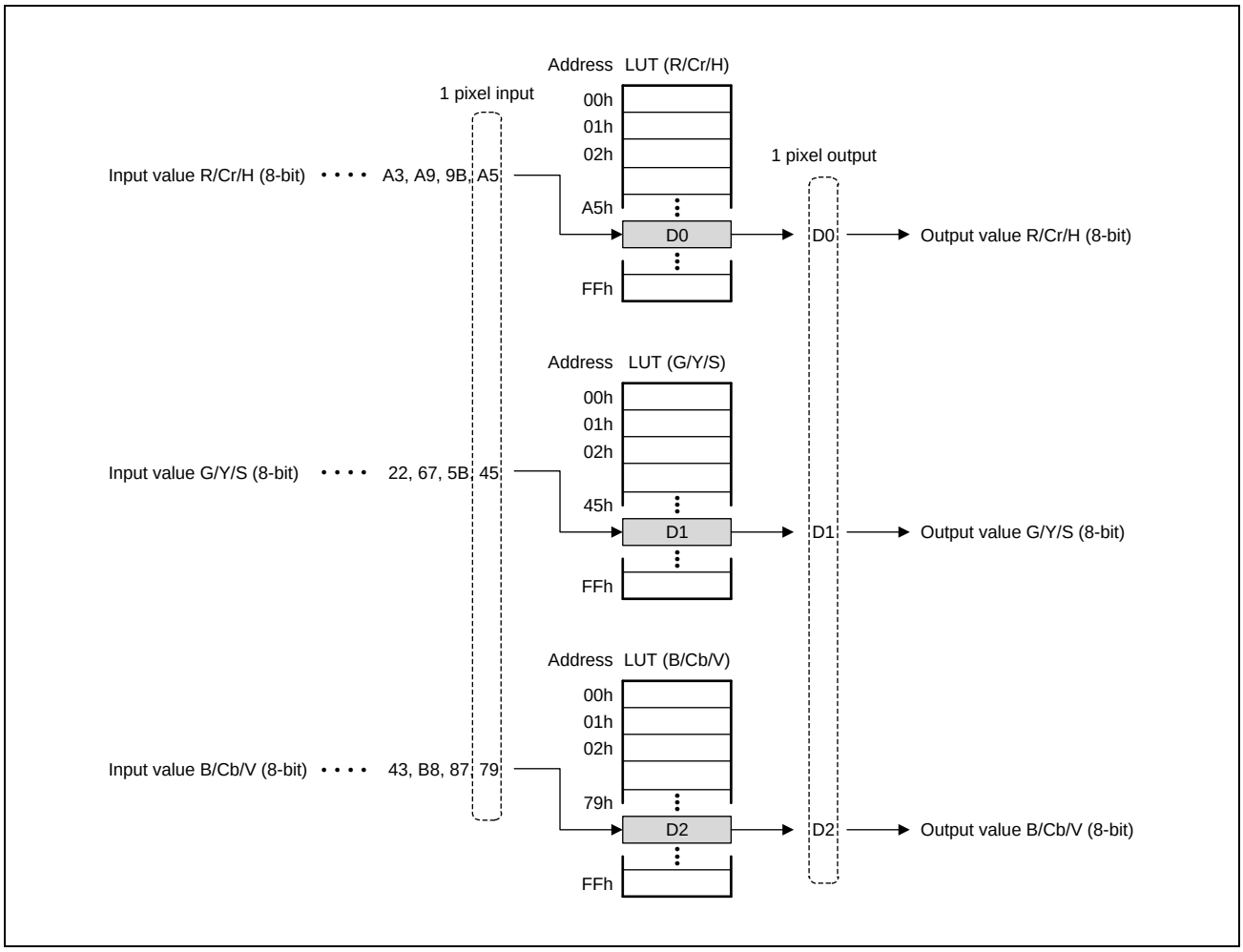


Figure 9.9-30 Relationship between Input and Output for 1D-LUT Table

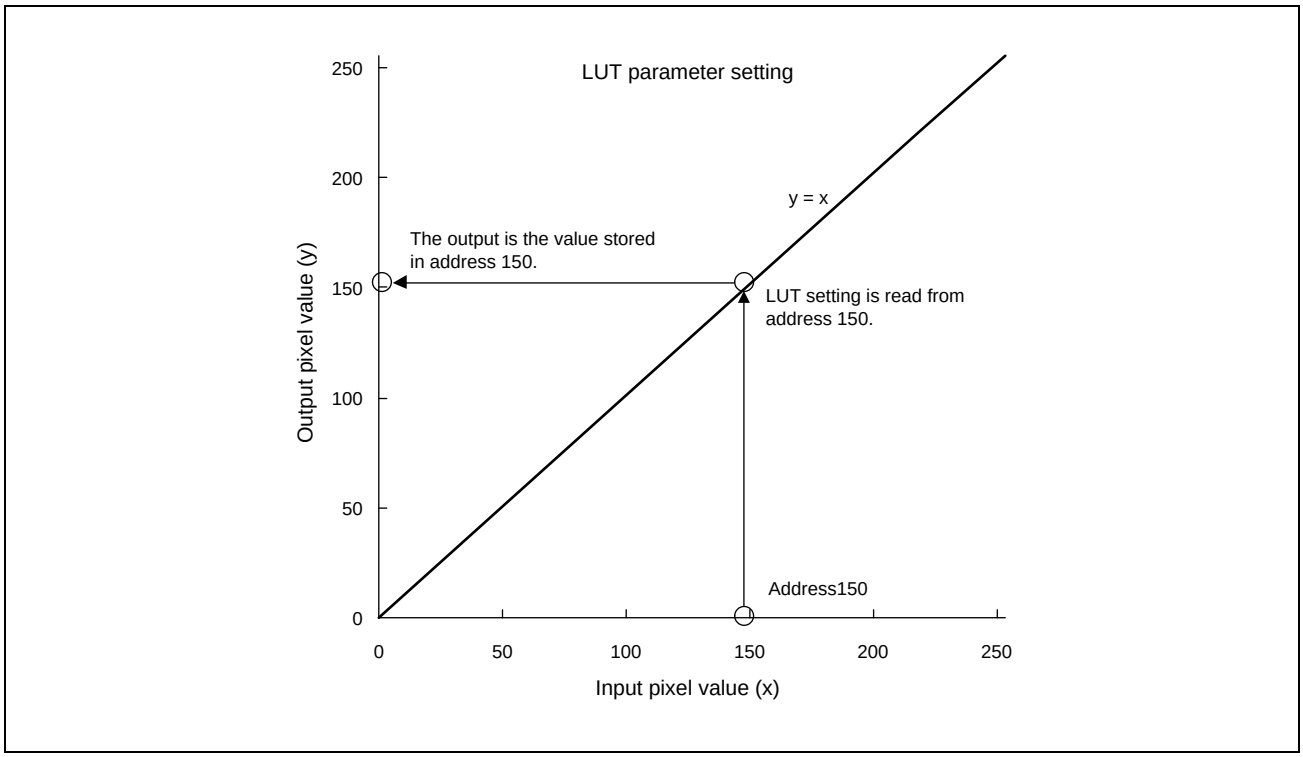


Figure 9.9-31 Setting Example in Which Output Becomes Equal to Input

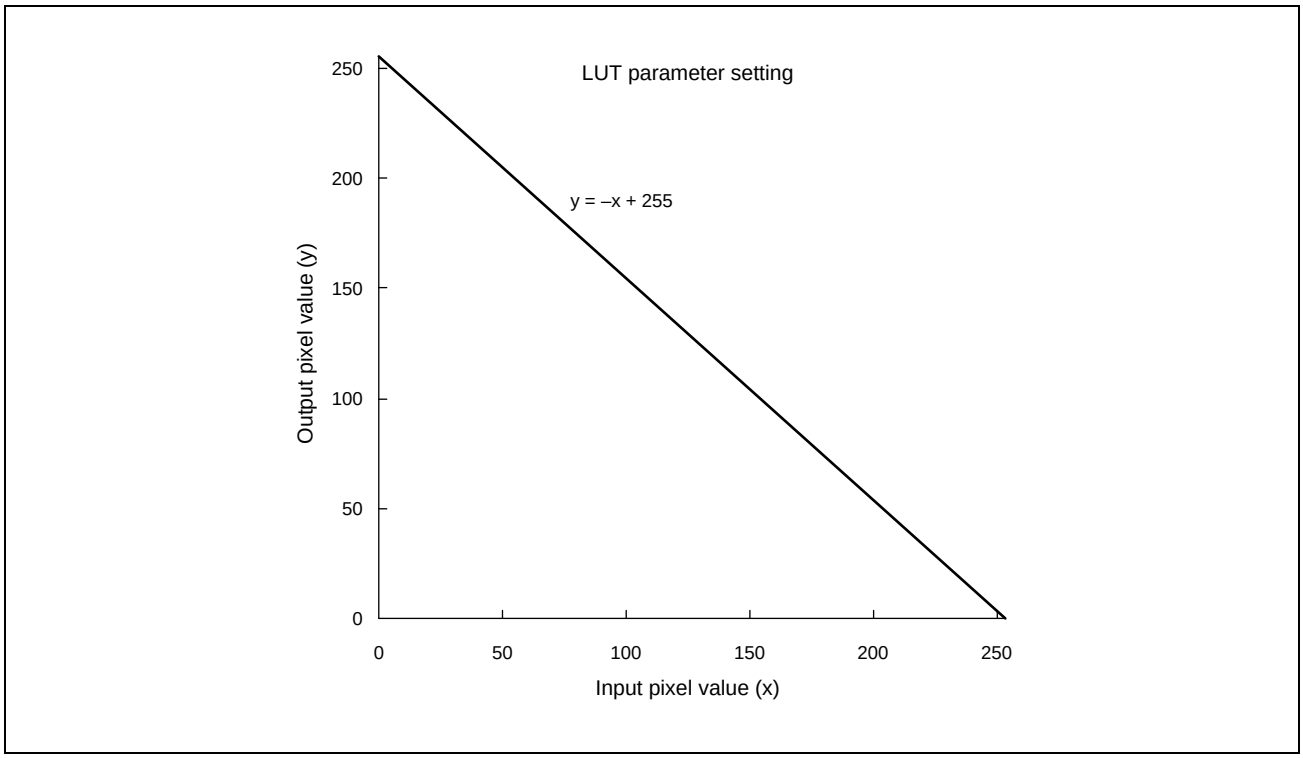


Figure 9.9-32 Setting Example of Negative-Positive Conversion

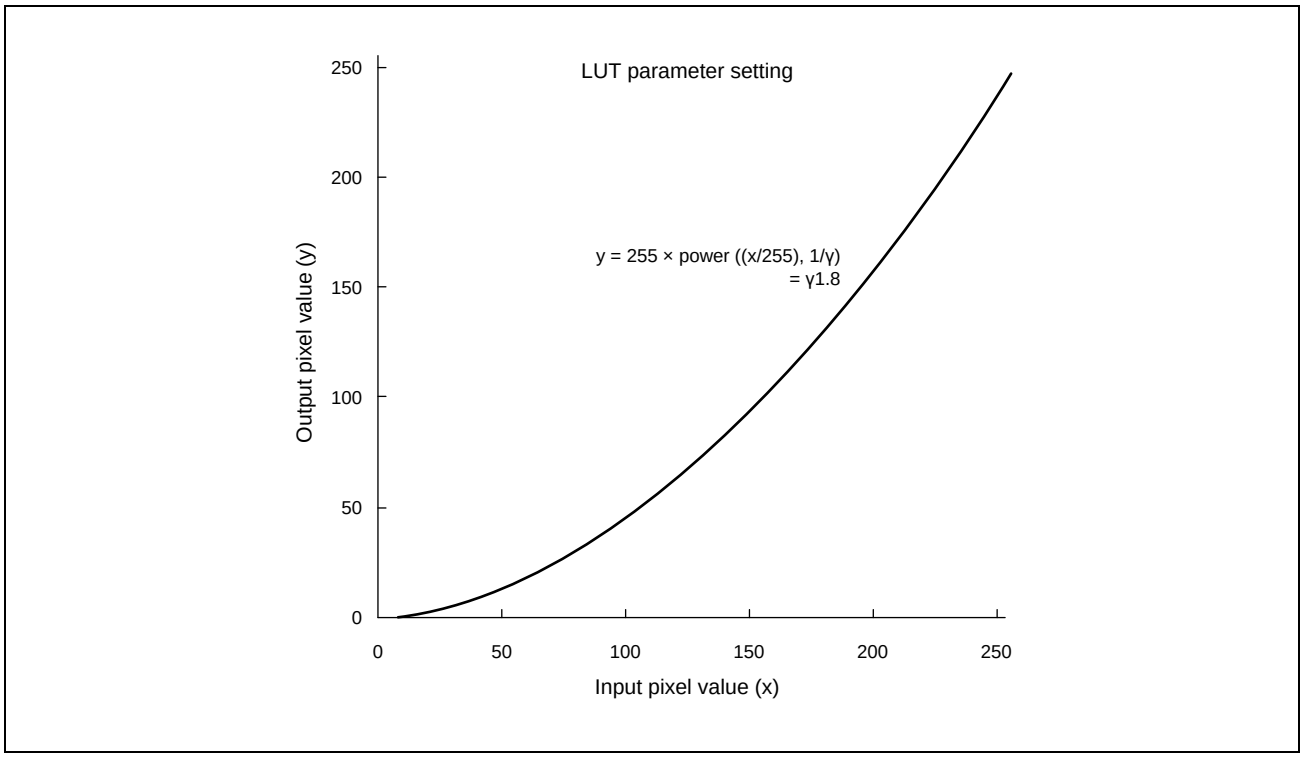


Figure 9.9-33 Setting Example of γ Correction

9.9.1.3.12 CLU Control Register

(1) CLU Control Register (VI6_CLU_CTRL)

Access Size : 32 bits
Address : <VSPI0_base> + 2900h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	AAI	-	-	-	MVS	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	RW	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AX1I[1:0]		AX2I[1:0]		-	-	OS0[1:0]		OS1[1:0]		OS2[1:0]		-	-	MD2	CLU_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	R	R	RW	RW	RW	RW	RW	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28	AAI	0h	RW	Automatic Table Address Increment Selects whether to specify the CLU table address in VI6_CLU_ADDR (refer to **9.9.1.4.5 (2)**) every time or update the address automatically. For details, refer to **9.9.1.4.5 (2)**. 0b: CLU table address should be specified every time 1b: CLU table address is automatically incremented
27 to 25	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
24	MVS	0h	RW	Max Value Stretch Select calculation method in max value region. 0b: Method 0 (Lower compatible) 1b: Method 1 (Improved characteristics) Setting 1 to MVS bit is recommended.
23 to 16	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
15, 14	AX1I[1:0]	0h	RW	Input Control 0 in 2D Mode When setting the M2D bit to 0, be sure to set these bits to 0. When setting the M2D bit to 1, be sure to set these bits to 3.
13, 12	AX2I[1:0]	0h	RW	Input Control 1 in 2D Mode When setting the M2D bit to 0, be sure to set these bits to 0. When setting the M2D bit to 1, be sure to set these bits to 1.
11, 10	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
9, 8	OS0[1:0]	0h	RW	Output Control 0 in 2D Mode When setting the M2D bit to 0, be sure to set these bits to 0. When setting the M2D bit to 1, be sure to set these bits to 3.
7, 6	OS1[1:0]	0h	RW	Output Control 1 in 2D Mode When setting the M2D bit to 0, be sure to set these bits to 0. When setting the M2D bit to 1, be sure to set these bits to 1.
5, 4	OS2[1:0]	0h	RW	Output Control 2 in 2D Mode When setting the M2D bit to 0, be sure to set these bits to 0. When setting the M2D bit to 1, be sure to set these bits to 3.

Bit	Bit Name	Initial Value	R/W	Description
3, 2	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
1	MD2	0h	RW	LUT Dimension Number Specifies the number of LUT dimensions. The details of each mode will be described later. 2D mode can be used only when the CLU input color space is YCbCr. 0b: Operates in 3D mode 1b: Operates in 2D mode
0	CLU_EN	0h	RW	CLU Processing Enable/Disable Enables or disables the 3D/2D color correction function by the CLU. When the CLU is used, the color component information needs to be set in the CLU table. For how to set data in the CLU table, see **9.9.1.4.5 (2)**. 0b: CLU color correction is disabled 1b: CLU color correction is enabled

The CLU handles three-dimensional or two-dimensional LUTs. A three-dimensional LUT is called a 3D-LUT and a two-dimensional LUT is called a 2D-LUT. The 3D or 2D operating mode of the CLU is selected by the M2D bit.

In 3D mode, a three-dimensional space such as that shown in **Figure 9.9-34** (*a*) is considered. The total number of coordinate points that can exist in this three-dimensional space is the total number of combinations of input data. For the CLU, this becomes the cube of each 8-bit component (= 16777216 points). Since it is impractical to have table values (= table memory) for the same number of these coordinate points, the CLU divides the three-dimensional space of **Figure 9.9-34** (*a*) into grids as shown in **Figure 9.9-34** (*b*) in 3D mode. The apex in each divided grid is defined as the coordinate point in the three-dimensional space, and a table value with three component values is set for all of these coordinate points. The total number of coordinate points in **Figure 9.9-34** (*b*) is $17^3 = 4913$. For the method of setting the table value for these 4913 points, refer to **9.9.1.4.5 (2) CLU Table**.

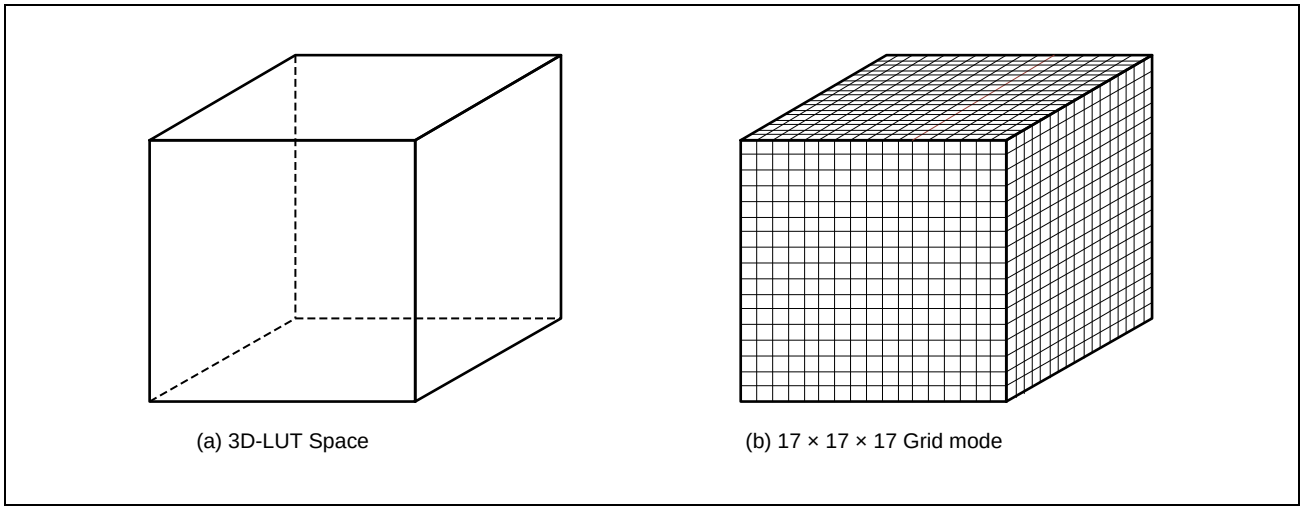


Figure 9.9-34 Concept and Division Count of a 3-Dimensional LUT

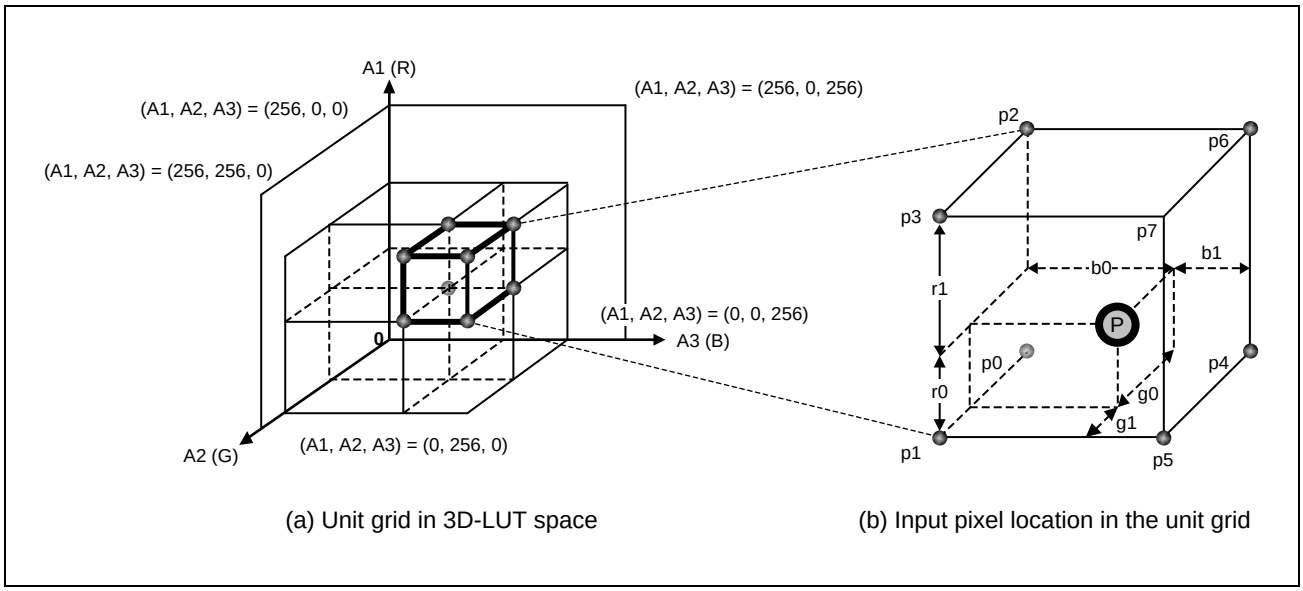


Figure 9.9-35 Conceptual Diagram of LUT Space when CLU is in 3D Mode

To express a three-dimensional space such as that shown in **Figure 9.9-34**, consider the three-dimensional coordinates and axes shown in **Figure 9.9-35** and define the minimum value and maximum value of each axis as 0 and 256, respectively. In 3D-LUT operation, the CLU forcibly assigns the four components of input data (α, R/Cr/H, G/Y/S, and B/Cb/V) to axes: R, Cr, and H components to the first axis (A1), G, Y, and S components to the second axis (A2), and B, Cb, and V components to the third axis (A3). (The case of 2D mode is described later.) Among multiple components assigned to each axis (e.g., R, Cr, and H components for the first axis), one component is selected according to the format of the data input to the CLU. For example, when the color space of the data input to the CLU is RGB, the first axis (A1) shows the R component, the second axis (A2) shows the G component, and the third axis (A3) shows the B component. In a YCbCr color space, the first axis (A1) shows the Cr component, the second axis (A2) shows the Y component, and the third axis (A3) shows the Cb component.

Each component value of the CLU input data becomes a value on each axis. Since the input data ranges from 0 to 255, the maximum value of 256 is handled as a virtual point in a grid space like that shown in **Figure 9.9-34**. In a case where the coordinates obtained by plotting the values of the input three components on axes are exactly on a grid point (e.g., p0 in **Figure 9.9-35** (*b*)), the 3-component value of the table value (p0) set to that location becomes the CLU output. If the input value of the three components result in an intermediate location between grid points and not on a grid point, such as position P in **Figure 9.9-35** (*b*), value P of the three components is interpolated from the table values (p0 to p7 in **Figure 9.9-35** (*b*)) for the eight surrounding grid points and this is used as the CLU output.

By setting arbitrary table value for a three-dimensional LUT space (three-dimensional color space) such as that shown in **Figure 9.9-34** using the processing described above, not only color conversion or color correction for the entire color space, but color conversion or color correction (memory color correction, skin smoothing, etc.) for a particular color area can be achieved.

Next, a 2D-LUT that is used when the CLU is operating in 2D mode is described (M2D bit should be set to 1 for operation in 2D mode). 2D-LUT mode is similar to 3D-LUT mode. However, since the axes for selecting the LUT become two-dimensional, the concept of a 3D-LUT space illustrated in **Figure 9.9-35** changes to a 2D-LUT space like that shown in **Figure 9.9-36**.

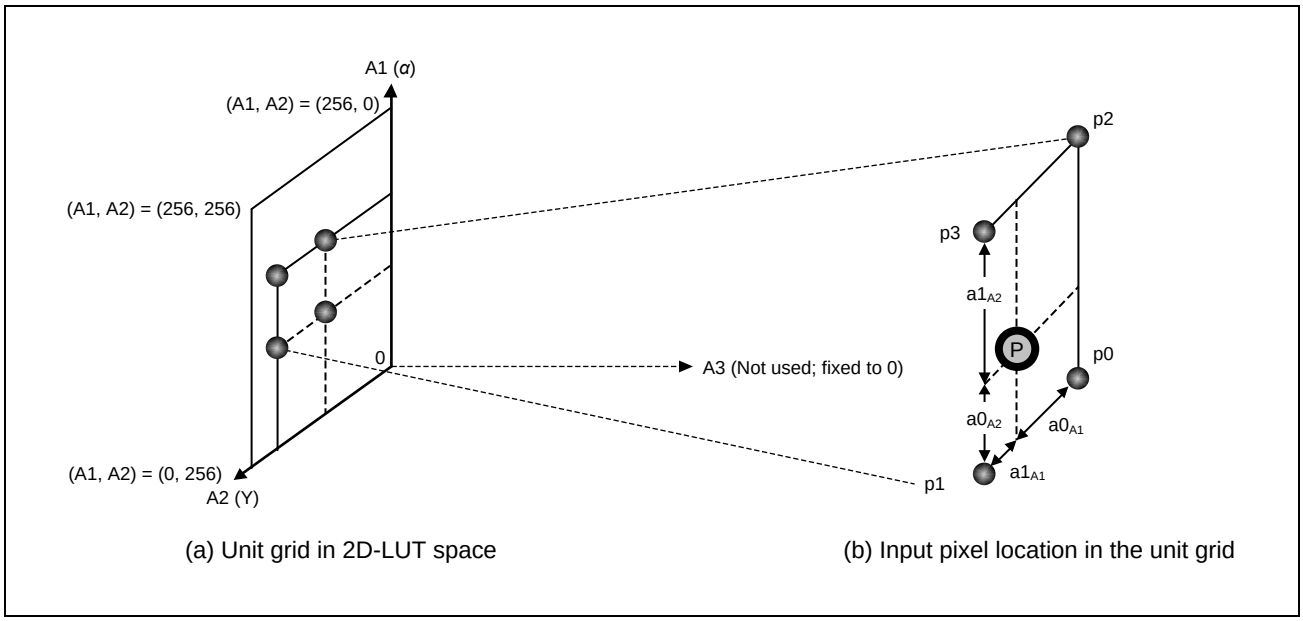


Figure 9.9-36 Conceptual Diagram of LUT Space when CLU is in 2D Mode

As shown in **Figure 9.9-36** (*a*), the first axis (A1) and second axis (A2) are used, and the A1-A2 plane becomes the LUT space in 2D mode. Because the third axis (A3) is not used, the CLU always handles the value on the A3 axis as 0.

2D mode can be used only when the CLU input is in the YCbCr color space. The input value of axis A1 is the input α value of the CLU, and the input value of axis A2 is the input luminance Y of the CLU. If the coordinates in the 2D-LUT space which were determined by the input values are exactly on a grid point (e.g., p2 in **Figure 9.9-36** (*b*)), the table value (p2) set to that location becomes the output value of the 2D-LUT. If the input 2-component value exists at an intermediate location between grid points and not on a grid point, such as position P in **Figure 9.9-36** (*b*), value P is interpolated from the table values (p0 to p3 in **Figure 9.9-36** (*b*)) for the four surrounding grid points and this is used as the 2D-LUT output. The interpolated component of the second axis of the 2D-LUT output acquired in this manner becomes the Y component output of the CLU. In 2D mode, the Cb/Cr component is a pass-through output.

For the CLU table values in **Figure 9.9-36** in 2D mode, set a value for only the component value of the second axis, and set 0 for the component values of the first and third axes. For details on the setting method, see **9.9.1.4.5 (2) CLU Table**.

9.9.1.3.13 HST Control Register

(1) HST Control Register (VI6_HST_CTRL)

Access Size : 32 bits
Address : <VSPI0_base> + 2A00h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	HST_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
0	HST_EN	0h	RW	HSV Conversion (RGB → HSV) Enable/Disable Enables or disables RGB → HSV conversion. H indicates hue, S indicates saturation, and V indicates a value (brightness). 0b: RGB → HSV conversion is disabled 1b: RGB → HSV conversion is enabled

Various color correction processing can be applied to HSV-format images by combining the 1D-LUT and 3D-LUT. For details on the 1D-LUT and 3D-LUT, see **9.9.1.3.11 (1) LUT Control Register (VI6_LUT_CTRL)** and **9.9.1.3.12 (1) CLU Control Register (VI6_CLU_CTRL)**.

To return HSV-converted data to RGB data, HSV → RGB conversion is used (see **9.9.1.3.14 (1) HSI Control Register (VI6_HSI_CTRL)**). RGB ↔ HSV conversion causes an operation error. Therefore, even when only just performing conversion of RGB → HSV → RGB, the original RGB value and converted RGB value will not completely match.

9.9.1.3.14 HSI Control Register

(1) HSI Control Register (VI6_HSI_CTRL)

Access Size : 32 bits
Address : <VSPI0_base> + 2B00h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	HIS_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
0	HIS_EN	0h	RW	Reversed HSV Conversion (HSV → RGB) Enable/Disable Enables or disables HSV → RGB conversion to return HSV data that was converted by HSV conversion (see **9.9.1.3.13 (1)**) back to RGB data. 0b: HSV → RGB conversion is disabled 1b: HSV → RGB conversion is enabled

RGB ↔ HSV conversion causes an operation error. Therefore, even when only just performing conversion of RGB → HSV → RGB, the original RGB value and converted RGB value will not completely match.

9.9.1.3.15 HGO Control Registers

(1) HGO Detection Window Offset Register (VI6_HGO_OFFSET)

Access Size : 32 bits
Address : <VSPI0_base> + 3000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	HOFFSET[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	VOFFSET[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
29 to 16	HOFFSET[13:0]	0h	RW	Horizontal Offset of Histogram Detection Window The HGO creates a histogram for the detection window in the input image (**Figure 9.9-37**). In these bits, specify the value of horizontal offset (hoffset shown in **Figure 9.9-37**) in pixel units. A value from 0 to 8189 can be specified. The value of HOFFSET shall be smaller than that of the input image size of HGO.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13 to 0	VOFFSET[13:0]	0h	RW	Vertical Offset of Histogram Detection Window The HGO creates a histogram for the detection window in the input image (**Figure 9.9-37**). In these bits, specify the value of vertical offset (voffset shown in **Figure 9.9-37**) in pixel units. A value from 0 to 8189 can be specified. The value of VOFFSET shall be smaller than that of the input image size of HGO.

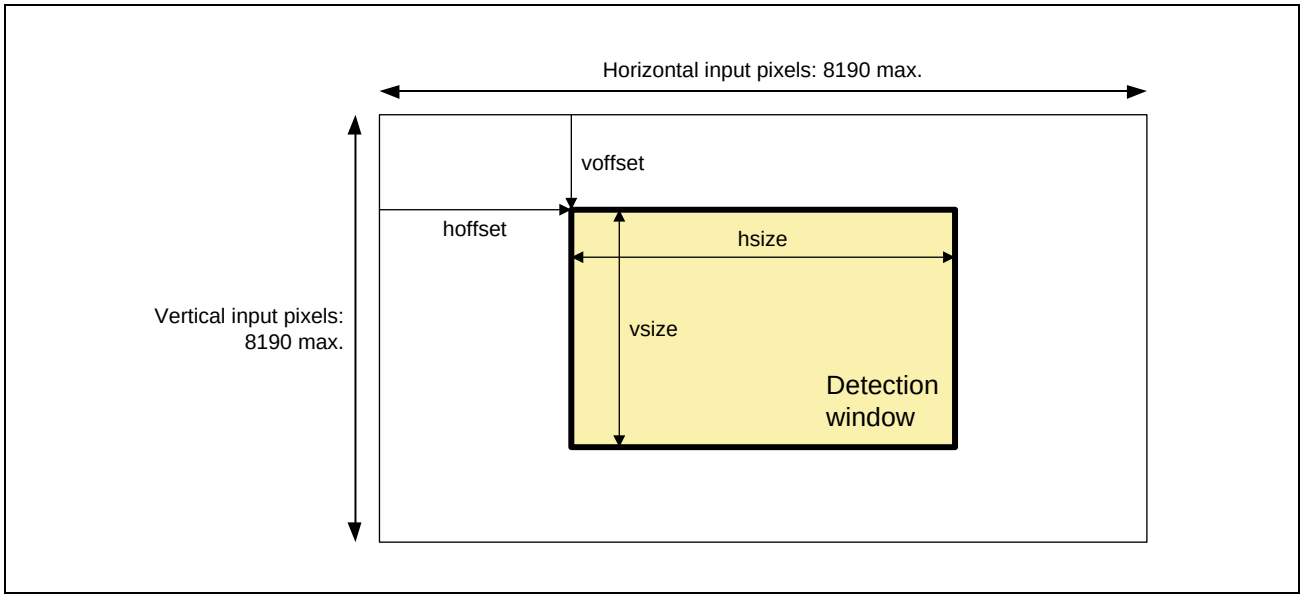


Figure 9.9-37 Histogram Detection Window of HGO

(2) HGO Detection Window Size Register (VI6_HGO_SIZE)

Access Size : 32 bits
Address : <VSPI0_base> + 3004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	HSIZE[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	VSIZE[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
29 to 16	HSIZE[13:0]	0h	RW	Horizontal Size of Histogram Detection Window The HGO creates a histogram for the detection window in the input image (**Figure 9.9-37**). In these bits, specify the value of horizontal size (hsize shown in **Figure 9.9-37**) in pixel units. A value from 1 to 8190 can be specified.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13 to 0	VSIZE[13:0]	0h	RW	Vertical Size of Histogram Detection Window The HGO creates a histogram for the detection window in the input image (**Figure 9.9-37**). In these bits, specify the value of vertical size (vsize shown in **Figure 9.9-37** in pixel units. A value from 1 to 8190 can be specified.

(3) HGO Mode Register (VI6_HGO_MODE)

Access Size : 32 bits
Address : <VSPI0_base> + 3008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	AUTOW	-	STEP	-	-	MAXRGB	OFSB_R	OFSB_G	OFSB_B	HRATIO[1:0]		VRATIO[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	RW	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12	AUTOW	0h	RW	HGO automatically histogram write. 0: Histogram is not dumped to external memory automatically. S/W needs to dump histogram from histogram registers to external memory. 1: Histogram is dumped to SDRAM automatically. Dump format is shown in **Figure 9.9-38**. HGO histogram swapping register in **9.9.1.3.15 (14)** specifies data order in 4byte unit as **Table 9.9-29**.
11	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.
10	STEP	0h	RW	Histogram step of Y/MaxRGB. 0: 64 step mode. 1: 256 step mode for Y/MaxRGB histogram. Restriction of color format is shown in **Table 9.9-30** about 64 step mode and **Table 9.9-31** about 256 step mode.
9, 8	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
7	MAXRGB	0h	RW	Histogram Source Component Setting 0: The histogram is generated from 3 color components independently. 1: The histogram is generated from the maximum value of input R, G and B data.
6	OFSB_R	0h	RW	Offset Binary Mode for R/Cr/H Component 0: Straight binary 1: Offset binary In offset binary mode, values are converted to absolute values before they are used to detect the maximum value, minimum value, sum, and black band. Note that values without conversion are always used for histogram creation regardless of this mode setting.
5	OFSB_G	0h	RW	Offset Binary Mode for G/Y/S Component 0: Straight binary 1: Offset binary In offset binary mode, values are converted to absolute values before they are used to detect the maximum value, minimum value, sum, and black band. Note that values without conversion are always used for histogram creation regardless of this mode setting.
4	OFSB_B	0h	RW	Offset Binary Mode for B/Cb/V Component 0: Straight binary 1: Offset binary In offset binary mode, values are converted to absolute values before they are used to detect the maximum value, minimum value, sum, and black band. Note that values without conversion are always used for histogram creation regardless of this mode setting.

Bit	Bit Name	Initial Value	R/W	Description
3, 2	HRATIO[1:0]	0h	RW	Horizontal Pixel Skipping Mode for Histogram Detection 00b: No skipping for horizontal pixels. 01b: Horizontal 1/2 skipping. One pixel is discarded from every two pixels before a histogram is created. 10b: Horizontal 1/4 skipping. Three pixels are discarded from every four pixels before a histogram is created. 11b: Setting prohibited. The first pixel in the horizontal direction is not discarded but subsequent pixels are discarded (regardless of the histogram detection window). Among the pixels that have not been discarded, only the pixels within the detection window are used to create a histogram.
1, 0	VRATIO[1:0]	0h	RW	Vertical Pixel Skipping Mode for Histogram Detection 00b: No skipping for vertical pixels. 01b: Vertical 1/2 skipping. One pixel is discarded from every two pixels before a histogram is created. 10b: Vertical 1/4 skipping. Three pixels are discarded from every four pixels before a histogram is created. 11b: Setting prohibited. The first pixel in the vertical direction is not discarded but subsequent pixels are discarded (regardless of the histogram detection window). Among the pixels that have not been discarded, only the pixels within the detection window are used to create a histogram.

Mode	64 Step Mode & maxRGB Disable		64 Step Mode & maxRGB Enable		256 Step Mode	
Address (Offset)	Component	Bit[31:0]	Component	Bit[31:0]	Component	Bit[31:0]
4 × 0	R/Cr/H	VI6_HGO_R_HISTO_0	N.A	N.A	Y or max (R, G, B)	VI6_HGO_EXT_HIST_DATA (0)
4 × 1		VI6_HGO_R_HISTO_1				VI6_HGO_EXT_HIST_DATA (1)
~		~				~
4 × 62		VI6_HGO_R_HISTO_62				VI6_HGO_EXT_HIST_DATA (62)
4 × 63		VI6_HGO_R_HISTO_63				VI6_HGO_EXT_HIST_DATA (63)
4 × 64	G/Y/S	VI6_HGO_G_HISTO_0	max (R, G, B)	VI6_HGO_G_HISTO_0		VI6_HGO_EXT_HIST_DATA (64)
4 × 65		VI6_HGO_G_HISTO_1		VI6_HGO_G_HISTO_1		VI6_HGO_EXT_HIST_DATA (65)
~		~		~		~
4 × 126		VI6_HGO_G_HISTO_62		VI6_HGO_G_HISTO_62		VI6_HGO_EXT_HIST_DATA (126)
4 × 127		VI6_HGO_G_HISTO_63		VI6_HGO_G_HISTO_63		VI6_HGO_EXT_HIST_DATA (127)
4 × 128	B/Cb/V	VI6_HGO_B_HISTO_0	N.A	N.A		VI6_HGO_EXT_HIST_DATA (128)
4 × 129		VI6_HGO_B_HISTO_1				VI6_HGO_EXT_HIST_DATA (129)
~		~				~
4 × 190		VI6_HGO_B_HISTO_62				VI6_HGO_EXT_HIST_DATA (190)
4 × 191		VI6_HGO_B_HISTO_63				VI6_HGO_EXT_HIST_DATA (191)
4 × 192	N.A	N.A	N.A	N.A		VI6_HGO_EXT_HIST_DATA (192)
4 × 193						VI6_HGO_EXT_HIST_DATA (193)
~						~
4 × 254						VI6_HGO_EXT_HIST_DATA (254)
4 × 255						VI6_HGO_EXT_HIST_DATA (255)
4 × 256	R/Cr/H	VI6_HGO_R_MAXMIN	N.A	N.A	N.A	N.A
4 × 257		VI6_HGO_R_SUM				
4 × 258		VI6_HGO_R_LB_DET				
4 × 259	N.A	N.A				
4 × 260	G/Y/S	VI6_HGO_G_MAXMIN	max (R, G, B)	VI6_HGO_G_MAXMIN	max (R, G, B)	VI6_HGO_G_MAXMIN
4 × 261		VI6_HGO_G_SUM		VI6_HGO_G_SUM		VI6_HGO_G_SUM
4 × 262		VI6_HGO_G_LB_DET		VI6_HGO_G_LB_DET		VI6_HGO_G_LB_DET
4 × 263	N.A	N.A	N.A	N.A	N.A	N.A
4 × 264	B/Cr/B	VI6_HGO_B_MAXMIN	N.A	N.A	N.A	N.A
4 × 265		VI6_HGO_B_SUM				
4 × 266		VI6_HGO_B_LB_DET				
4 × 267	N.A	N.A				
4 × 268	N.A	N.A	N.A	N.A	N.A	N.A
4 × 269						
4 × 270						
4 × 271						

Figure 9.9-38 HGO Histogram Dump Format in Case of Automatically Histogram Write

Table 9.9-29 Data Order of HGO Histogram in 4 Bytes

Memory Address	Data Order in 4 Bytes	
	H_LWS = 1 H_WDS = 1 H_BTS = 1	H_LWS = 1 H_WDS = 0 H_BTS = 0
4n + 0	Bit[31:24]	Bit[7:0]
4n + 1	Bit[23:15]	Bit[15:8]
4n + 2	Bit[15:8]	Bit[23:15]
4n + 3	Bit[7:0]	Bit[31:24]

(4) HGO LB Detection Threshold Register (VI6_HGO_LB_TH)

Access Size : 32 bits
Address : <VSPI0_base> + 300Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	BLACK_TH[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
7 to 0	BLACK_TH[7:0]	0h	RW	Threshold for Black Level Determination in Letter Box Detection In letter box detection, when all input pixels have the value equal to or smaller than the value specified in these bits, the image data is determined as black. Note that the letter box detection is not affected by the histogram detection window illustrated in **Figure 9.9-37**. A value from 0 to 255 can be specified.

(5) HGO Horizontal Position Register for LB Detection Zone-n (VI6_HGO_LBn_H) (n = 0, 1, 2, 3)

Access Size : 32 bits
Address : <VSPI0_base> + 3010h /3018h / 3020h / 3028h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	HPOS0[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	HPOS1[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
29 to 16	HPOS0[13:0]	0h	RW	Horizontal Start Position for Letter Box Detection Zone-n When all pixels on a line (shown in **Figure 9.9-39**) in the input image have the value equal to or smaller than the threshold value, the HGO determines that there is a letter box. In these bits, specify the value of hpos0 of the detection zone-n (**Figure 9.9-39**) in pixel units. A value from 0 to 8189 can be specified. hpos0 ≤ hpos1 should be satisfied. These bits are valid for Zone-0 and 1. For Zone-2 and 3, these bits are reserved.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13 to 0	HPOS1[13:0]	0h	RW	Horizontal End Position for Letter Box Detection Zone-n When all pixels on a line (shown in **Figure 9.9-39**) in the input image have the value equal to or smaller than the threshold value, the HGO determines that there is a letter box. In these bits, specify the value of hpos1 of the detection zone-n (**Figure 9.9-39**) in pixel units. A value from 0 to 8189 can be specified. hpos0 ≤ hpos1 should be satisfied.

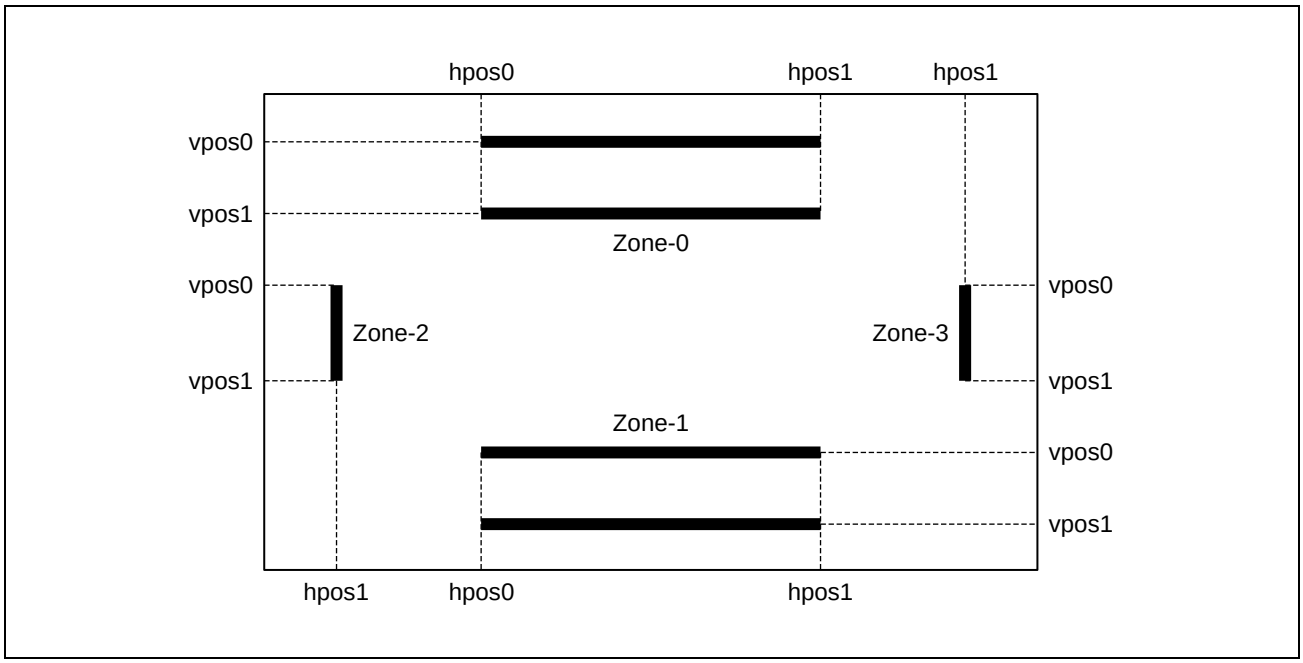


Figure 9.9-39 Letter Box Detection Position Settings for HGO

(6) HGO Vertical Position Register for LB Detection Zone-n (VI6_HGO_LBn_V) (n = 0, 1, 2, 3)

Access Size : 32 bits

Address : <VSPI0_base> + 3014h / 301Ch / 3024h / 302Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	VPOS0[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	VPOS1[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
29 to 16	VPOS0[13:0]	0h	RW	Vertical Start Position for Letter Box Detection Zone-n When all pixels on a line (shown in **Figure 9.9-39**) in the input image have the value equal to or smaller than the threshold value, the HGO determines that there is a letter box In these bits, specify the value of vpos0 of the detection zone-n (**Figure 9.9-39**) in pixel units. A value from 0 to 8189 can be specified. vpos0 ≤ vpos1 should be satisfied.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13 to 0	VPOS1[13:0]	0h	RW	Vertical End Position for Letter Box Detection Zone-n When all pixels on a line (shown in **Figure 9.9-39**) in the input image have the value equal to or smaller than the threshold value, the HGO determines that there is a letter box. In these bits, specify the value of vpos1 of the detection zone-n (**Figure 9.9-39**) in pixel units. A value from 0 to 8189 can be specified. vpos0 ≤ vpos1 should be satisfied.

(7) HGO Component-m Histogram Register (VI6_HGO_m_HISTO_n) (n = 0 to 63, m = R, G, B)

Access Size : 32 bits

Address : <VSPI0_base> + 3030 + 0004 × n (n = 0 to 63)h (R)
<VSPI0_base> + 3140 + 0004 × n (n = 0 to 63)h (G)
<VSPI0_base> + 3250 + 0004 × n (n = 0 to 63)h (B)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	HISTOGRAM_n[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HISTOGRAM_n[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 0	HISTOGRAM_n [23:0]	0h	R	Frequency of Component-m in the value range-n From these bits, the count of the pixels whose component-m value (val) satisfies the following condition within the histogram detection window **Figure 9.9-37**) is read (after pixel skipping is applied when skipping mode is selected). $4*n \leq val < 4*(n+1)$ Counting starts after the HGO is activated. To read the histogram, read these bits after the HGO has completed processing of one screen of data and before the HGO is activated again. These bits are available when VI6_HGO_MODE.step is 0 (64 step mode). When VI6_HGO_MODE.step is 1 (256 step mode), read Y/MaxRGB histogram from the VI6_HGO_EXT_HIST_DATA after setting histogram address in the VI6_HGO_EXT_HIST_ADDR.

The component-m in a color space is determined by the following table in case of VI6_HGO_MODE.step = 0 (64 step mode)

Table 9.9-30 The Component of Index-m for Each Color Space in 64 Step Mode

	Color Space			
	RGB		YCbCr	HSV
Index-m	VI6_HGO_MODE.MAXRGB = 0	VI6_HGO_MODE.MAXRGB = 1	VI6_HGO_MODE MAXRGB = 0*1	VI6_HGO_MODE.MAXRGB = 0*1
R	R	N/A*2	Cr	H
G	G	max(R, G, B)*3	Y	S
B	B	N/A*2	Cb	V

Note 1. When color space of input data is in YCbCr or HSV, set VI6_HGO_MODE.MAXRGB = 0.

Note 2. When VI6_HGO_MODE.MAXRGB = 1, the histogram of index-R / index-B are not ensured.

Note 3. max(R, G, B) indicates maximum value of input R, G, and B data.

(8) HGO Component-m Min/Max Value Register (VI6_HGO_m_MAXMIN) (m = R, G, B)

Access Size : 32 bits

Address : <VSPI0_base> + 3130h (R)
<VSPI0_base> + 3240h (G)
<VSPI0_base> + 3350h (B)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	MAXVAL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	MINVAL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 16	MAXVAL[7:0]	0h	R	Maximum Value of Component-m From these bits, the maximum value of the component-m of the pixels within the histogram detection window (**Figure 9.9-37**) is read (after pixel skipping is applied when skipping mode is selected). Maximum value detection starts after the HGO is activated. To read the maximum value, read these bits after the HGO has completed processing of one screen of data and before the HGO is activated again.
15 to 8	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
7 to 0	MINVAL[7:0]	0h	R	Minimum Value of Component-m From these bits, the minimum value of the component-m of the pixels within the histogram detection window (**Figure 9.9-37**) is read (after pixel skipping is applied when skipping mode is selected). Minimum value detection starts after the HGO is activated. To read the minimum value, read these bits after the HGO has completed processing of one screen of data and before the HGO is activated again.

(9) HGO Component-m Sum Register (VI6_HGO_m_SUM) (m = R, G, B)

Access Size : 32 bits

Address : <VSPI0_base> + 3134h (R)
<VSPI0_base> + 3244h (G)
<VSPI0_base> + 3354h (B)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SUMVAL[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SUMVAL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SUMVAL[31:0]	0h	R	Sum of Component-m From these bits, the sum of the component-m of the pixels within the histogram detection window (**Figure 9.9-37**) is read (after pixel skipping is applied when skipping mode is selected). Accumulation starts after the HGO is activated. To read the sum, read these bits after the HGO has completed processing of one screen of data and before the HGO is activated again.

(10) HGO Component-m LB Detection Result Register (VI6_HGO_m_LB_DET) (m = R, G, B)

Access Size : 32 bits

Address : <VSPI0_base> + 3138h (R)
<VSPI0_base> + 3248h (G)
<VSPI0_base> + 3358h (B)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	LTRBOX1	LTRBOX2	SIDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
2	LTRBOX1	0h	R	Letter Box Detection Result #1 of Zone-0/1 for Component-m This bit is set to 1 when none of the component-m of the pixels on the lines of zone-0/vpos0 and zone-1/vpos1 (two lines in total) in **Figure 9.9-39** exceeds the threshold (BLACK_TH). If any pixel is greater than the threshold, this bit is set to 0. Read this bit after the HGO has completed processing of one screen of data and before the HGO is activated again. The value read during processing is not guaranteed to be correct.
1	LTRBOX2	0h	R	Letter Box Detection Result #2 of Zone-0/1 for Component-m This bit is set to 1 when none of the component-m of the pixels on the four lines of zone-0 and 1 in **Figure 9.9-39** exceeds the threshold (BLACK_TH). If any pixel is greater than the threshold, this bit is set to 0. Read this bit after the HGO has completed processing of one screen of data and before the HGO is activated again. The value read during processing is not guaranteed to be correct.
0	SIDE	0h	R	Letter Box Detection Result of Zone-2/3 for Component-m This bit is set to 1 when none of the component-m of the pixels on the two lines of zone-2 and 3 in **Figure 9.9-39** exceeds the threshold (BLACK_TH). If any pixel is greater than the threshold, this bit is set to 0. Read this bit after the HGO has completed processing of one screen of data and before the HGO is activated again. The value read during processing is not guaranteed to be correct.

(11) HGO Y/MaxRGB Extended Histogram Address Register (VI6_HGO_EXT_HIST_ADDR)

Access Size : 32 bits
Address : <VSPI0_base> + 335Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	EXT_HIST_ADDR[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
7 to 0	EXT_HIST_ADDR[7:0]	0h	RW	Histogram address in case of 256 step histogram mode. Set value-m when user wants to read frequency of Y or MaxRGB pixel value-m. After setting VI6_HGO_EXT_HIST_ADDR, read VI6_HGO_EXT_HIST_DATA to read frequency of Y or MaxRGB pixel value-m. This is in-direct addressing way. This in-direct addressing way is available in case of VI6_HGO_MODE.STEP = 1 (256 step histogram mode).

Table 9.9-31 Restriction in Case of 256 Step Mode (VI6_HGO_MODE.STEP = 1)

Color Space			
RGB		YCbCr	HSV
VI6_HGO_MODE.MAXRGB = 0	VI6_HGO_MODE.MAXRGB = 1	VI6_HGO_MODE MAXRGB = 0*1	VI6_HGO_MODE.MAXRGB = 0*1
N/A	max(R, G, B)*2	Y	N/A

Note 1. When color space of input data is in YCbCr or HSV, set VI6_HGO_MODE.MAXRGB = 0.

Note 2. max(R, G, B) indicates maximum value of input R, G, and B data.

(12) HGO Y/MaxRGB Extended Histogram Data Register (VI6_HGO_EXT_HIST_DATA)

Access Size : 32 bits

Address : <VSPI0_base> + 3360h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	EXT_HIST_DATA[23:16]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EXT_HIST_DATA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 0	EXT_HIST_DATA[23:0]	0h	RW	Histogram data in case of 256 step histogram mode. See VI6_HGO_EXT_HIST_ADDR register's explanation. This in-direct addressing way is available in case of VI6_HGO_MODE.STEP = 1 (256 step histogram mode).

(13) HGO Histogram Storing Address Register (VI6_HGO_HISTADD)

Access Size : 32 bits
Address : <VSPI0_base> + 336Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HISTADD[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HISTADD[15:8]								-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	HISTADD[31:8]	0h	RW	HGO histogram storing address These bits specify the start address of storing the HGO histogram plane to be written in case of enabling histogram auto write (VI6_HGO_MODE.AUTOW = 1). Start address of HGO histogram can be set in 256-byte units. And HISTADD [31:8] × 256 is recognized as start address to write HGO histogram.
7 to 0	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.

(14) HGO Histogram Swapping Register (VI6_HGO_HSWAP)

Access Size : 32 bits
Address : <VSPI0_base> + 3370h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	H_LWS	H_WDS	H_BTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
2	H_LWS	0h	RW	HGO histogram Swapping in Longword Units 0: Data swapping in longword (32-bit) units is disabled 1: Data swapping in longword (32-bit) units is enabled
1	H_WDS	0h	RW	HGO histogram Swapping in Word Units 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled
0	H_BTS	0h	RW	HGO histogram Swapping in Byte Units 0: Data swapping in byte (8-bit) units is disabled 1: Data swapping in byte (8-bit) units is enabled

See **Figure 9.9-38** and **Table 9.9-29** for setting HGO histogram swap.

(15) HGO Parameter Register Reset (VI6_HGO_REGRST)

Access Size : 32 bits
Address : <VSPI0_base> + 33FCh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	RCPART	-	-	-	RCLEA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	W	R	R	R	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
4	RCPART	0h	W	Status Reset Writing 1 to this bit resets counter bits in HGO.
3 to 1	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
0	RCLEA	0h	W	Register Reset Writing 1 to this bit resets all read-only registers (**9.9.1.3.15 (7)** to **9.9.1.3.15 (10)**) to their initial values. This register is write-only and is always read as 0. Note that R/W registers (**9.9.1.3.15 (1)** to **9.9.1.3.15 (6)**) are not affected by write access to this register.

When image partition is used, refer to **9.9.1.4.6 (7) Histogram of HGO/HGT**.

9.9.1.3.16 HGT Control Registers

(1) HGT Detection Window Offset Register (VI6_HGT_OFFSET)

Access Size : 32 bits
Address : <VSPI0_base> + 3400h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	HOFFSET[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	VOFFSET[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
29 to 16	HOFFSET[13:0]	0h	RW	Horizontal Offset of Histogram Detection Window The HGT creates a histogram for the detection window in the input image (**Figure 9.9-40**). In these bits, specify the value of horizontal offset (hoffset shown **Figure 9.9-40** in pixel units. A value from 0 to 8189 can be specified. The value of HOFFSET shall be smaller than that of the input image size of HGT.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13 to 0	VOFFSET[13:0]	0h	RW	Vertical Offset of Histogram Detection Window The HGT creates a histogram for the detection window in the input image (**Figure 9.9-40**). In these bits, specify the value of vertical offset (voffset shown in **Figure 9.9-40**) in pixel units. A value from 0 to 8189 can be specified. The value of VOFFSET shall be smaller than that of the input image size of HGT.

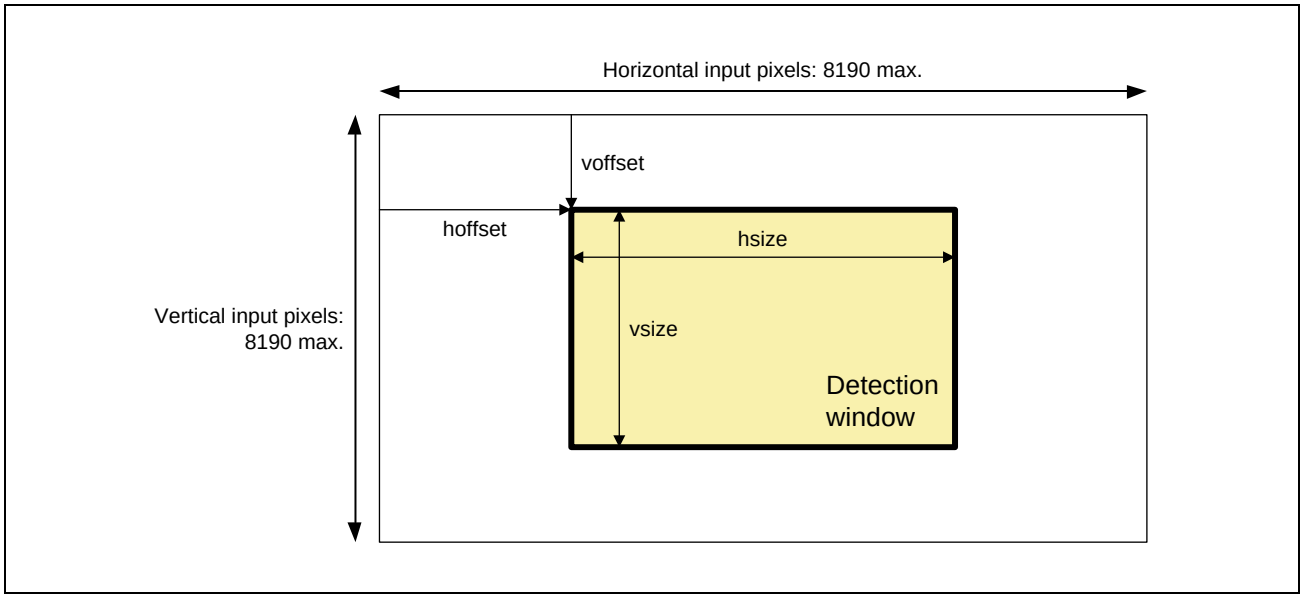


Figure 9.9-40 Histogram Detection Window of HGT

(2) HGT Detection Window Size Register (VI6_HGT_SIZE)

Access Size : 32 bits
Address : <VSPI0_base> + 3404h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	HSIZE[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	VSIZE[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
29 to 16	HSIZE[13:0]	0h	RW	Horizontal Size of Histogram Detection Window The HGT creates a histogram for the detection window in the input image (**Figure 9.9-40**). In these bits, specify the value of horizontal offset (hsize shown in **Figure 9.9-40**) in pixel units. A value from 1 to 8190 can be specified.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13 to 0	VSIZE[13:0]	0h	RW	Vertical Size of Histogram Detection Window The HGT creates a histogram for the detection window in the input image **Figure 9.9-40**). In these bits, specify the value of vertical offset (vsize shown in **Figure 9.9-40**) in pixel units. A value from 1 to 8190 can be specified.

(3) HGT Mode Register (VI6_HGT_MODE)

Access Size : 32 bits
Address : <VSPI0_base> + 3408h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	AUTOW	-	-	-	-	-	-	-	-	HRATIO[1:0]		VRATIO[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12	AUTOW	0h	RW	HGT automatically histogram write. 0b: Histogram is not dumped to external memory automatically. S/W needs to dump histogram from histogram registers to external memory. 1b: Histogram is dumped to SDRAM automatically. Dump format is shown in **Figure 9.9-41**. HGT Histogram Swapping Register in **9.9.1.3.16 (16)** specifies data order in 4byte unit as **Table 9.9-32**. Horizontal Pixel Skipping Mode for Histogram Detection
11 to 4	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
3, 2	HRATIO[1:0]	0h	RW	Horizontal Pixel Skipping Mode for Histogram Detection 00b: No skipping for horizontal pixels. 01b: Horizontal 1/2 skipping. One pixel is discarded from every two pixels before a histogram is created. 10b: Horizontal 1/4 skipping. Three pixels are discarded from every four pixels before a histogram is created. 11b: Setting prohibited. The first pixel in the horizontal direction is not discarded but subsequent pixels are discarded (regardless of the histogram detection window). Among the pixels that have not been discarded, only the pixels within the detection window are used to create a histogram.
1, 0	VRATIO[1:0]	0h	RW	Vertical Pixel Skipping Mode for Histogram Detection 00b: No skipping for vertical pixels. 01b: Vertical 1/2 skipping. One pixel is discarded from every two pixels before a histogram is created. 10b: Vertical 1/4 skipping. Three pixels are discarded from every four pixels before a histogram is created. 11b: Setting prohibited. The first pixel in the vertical direction is not discarded but subsequent pixels are discarded (regardless of the histogram detection window). Among the pixels that have not been discarded, only the pixels within the detection window are used to create a histogram.

Address (Offset)	Hue Area	Bit[31:0]
4 × 0	HueArea 0	VI6_HGT_HISTO_0_0
4 × 1		VI6_HGT_HISTO_0_1
~		~
4 × 30		VI6_HGT_HISTO_0_30
4 × 31		VI6_HGT_HISTO_0_31
4 × 32	HueArea 1	VI6_HGT_HISTO_1_0
4 × 33		VI6_HGT_HISTO_1_1
~		~
4 × 62		VI6_HGT_HISTO_1_30
4 × 63		VI6_HGT_HISTO_1_31
4 × 64	HueArea 2	VI6_HGT_HISTO_2_0
4 × 65		VI6_HGT_HISTO_2_1
~		~
4 × 94		VI6_HGT_HISTO_2_30
4 × 95		VI6_HGT_HISTO_2_31
4 × 96	HueArea 3	VI6_HGT_HISTO_3_0
4 × 97		VI6_HGT_HISTO_3_1
~		~
4 × 126		VI6_HGT_HISTO_3_30
4 × 127		VI6_HGT_HISTO_3_31
4 × 128	HueArea 4	VI6_HGT_HISTO_4_0
4 × 129		VI6_HGT_HISTO_4_1
~		~
4 × 158		VI6_HGT_HISTO_4_30
4 × 159		VI6_HGT_HISTO_4_31
4 × 160	HueArea 5	VI6_HGT_HISTO_5_0
4 × 161		VI6_HGT_HISTO_5_1
~		~
4 × 190		VI6_HGT_HISTO_5_30
4 × 191		VI6_HGT_HISTO_5_31
4 × 192	S	VI6_HGT_MAXMIN
4 × 193	S	VI6_HGT_SUM
4 × 194	V	VI6_HGT_LB_DET
4 × 195	N.A	N.A
~		
4 × 199		

Figure 9.9-41 HGT Histogram Dump Format in Case of Automatically Histogram Write

Table 9.9-32 Data Order of HGT Histogram in 4 Bytes

Memory Address	Data Order in 4 Bytes	
	H_LWS = 1 H_WDS = 1 H_BTS = 1	H_LWS = 1 H_WDS = 0 H_BTS = 0
4n + 0	Bit[31:24]	Bit[7:0]
4n + 1	Bit[23:15]	Bit[15:8]
4n + 2	Bit[15:8]	Bit[23:15]
4n + 3	Bit[7:0]	Bit[31:24]

(4) HGT Hue Area Register (VI6_HGT_HUE_AREA_n) (n = 0 to 5)

Access Size : 32 bits

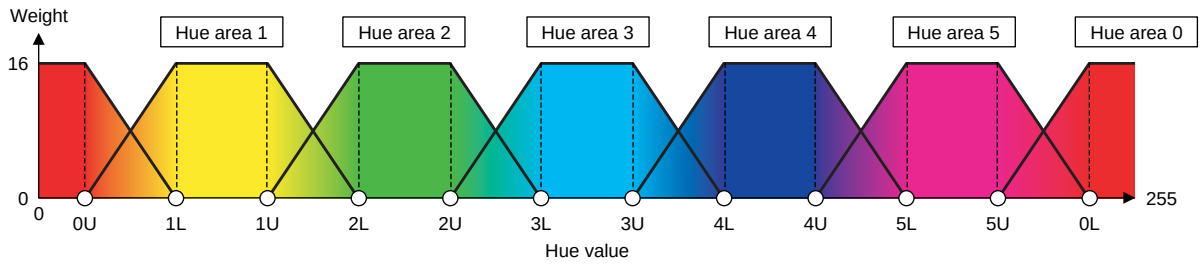
Address : <VSPI0_base> + 340Ch/ 3410h/ 3414h/ 3418h/ 341Ch/ 3420h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	HUE_LOWER_0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	HUE_UPPER_0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 16	HUE_LOWER_0 [7:0]	0h	RW	Lower Boundary Value for Hue Area - n The HGT creates a two-dimensional histogram of H (hue) and S (saturation) components. Division of hue areas and overlapping of adjacent hue areas can be specified through these bits. In these bits, specify the value of n L (hue_lower n) shown in **Figure 9.9-42**. A value from 0 to 255 can be specified. The specified value should satisfy the restrictions shown in **Figure 9.9-42**.
15 to 8	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
7 to 0	HUE_UPPER_0 [7:0]	0h	RW	Upper Boundary Value for Hue Area - n The HGT creates a two-dimensional histogram of H (hue) and S (saturation) components. Division of hue areas and overlapping of adjacent hue areas can be specified through these bits. In these bits, specify the value of n U (hue_upper n) shown in **Figure 9.9-42**. A value from 0 to 255 can be specified. The specified value should satisfy the restrictions shown in **Figure 9.9-42**.



Restrictions: 0U ≤ 1L ≤ 1U ≤ 2L ≤ 2U ≤ 3L ≤ 3U ≤ 4L ≤ 4U ≤ 5L ≤ 5U, 0L ≤ 0U, or 5U ≤ 0L

- Values are rounded down in the diagonal-line sections.
- When overlapped areas are not specified, the boundary value is included in the lower-value area.

0L = HUE_LOWER_0 register setting 0U = HUE_UPPER_0 register setting
1L = HUE_LOWER_1 register setting 1U = HUE_UPPER_1 register setting
2L = HUE_LOWER_2 register setting 2U = HUE_UPPER_2 register setting
3L = HUE_LOWER_3 register setting 3U = HUE_UPPER_3 register setting
4L = HUE_LOWER_4 register setting 4U = HUE_UPPER_4 register setting
5L = HUE_LOWER_5 register setting 5U = HUE_UPPER_5 register setting

Figure 9.9-42 Weighting Histogram Using Hue

(5) HGT LB Detection Threshold Register (VI6_HGT_LB_TH)

Access Size : 32 bits
Address : <VSPI0_base> + 3424h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	BLACK_TH[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
7 to 0	BLACK_TH[7:0]	0h	RW	Threshold for Black Level Determination in Letter Box Detection In letter box detection, when all pixels of the input image data have the value equal to or smaller than the value specified in these bits, the data is determined as black. Note that the letter box detection is not affected by the histogram detection window illustrated in **Figure 9.9-40**. A value from 0 to 255 can be specified.

(6) HGT Horizontal Position Register for LB Detection Zone-n (VI6_HGT_LBn_H) (n = 0, 1, 2, 3)

Access Size : 32 bits

Address : <VSPI0_base> + 3428h/ 3430h/ 3438h/ 3440h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	HPOS0[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	HPOS1[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
29 to 16	HPOS0[13:0]	0h	RW	Horizontal Start Position for Letter Box Detection Zone-n When all pixels on a line (shown in **Figure 9.9-43**) in the input image have the value equal to or smaller than the threshold value, the HGT determines that there is a letter box. In these bits, specify the value of hpos0 of the detection zone-n (**Figure 9.9-43**) in pixel units. A value from 0 to 8189 can be specified. hpos0 ≤ hpos1 should be satisfied. These bits are valid for Zone-0 and 1. For Zone-2 and 3, these bits are reserved.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13 to 0	HPOS1[13:0]	0h	RW	Horizontal End Position for Letter Box Detection Zone-n When all pixels on a line (shown in **Figure 9.9-43**) in the input image have the value equal to or smaller than the threshold value, the HGT determines that there is a letter box. In these bits, specify the value of hpos1 of the detection zone-n (**Figure 9.9-43**) in pixel units. A value from 0 to 8189 can be specified. hpos0 ≤ hpos1 should be satisfied.

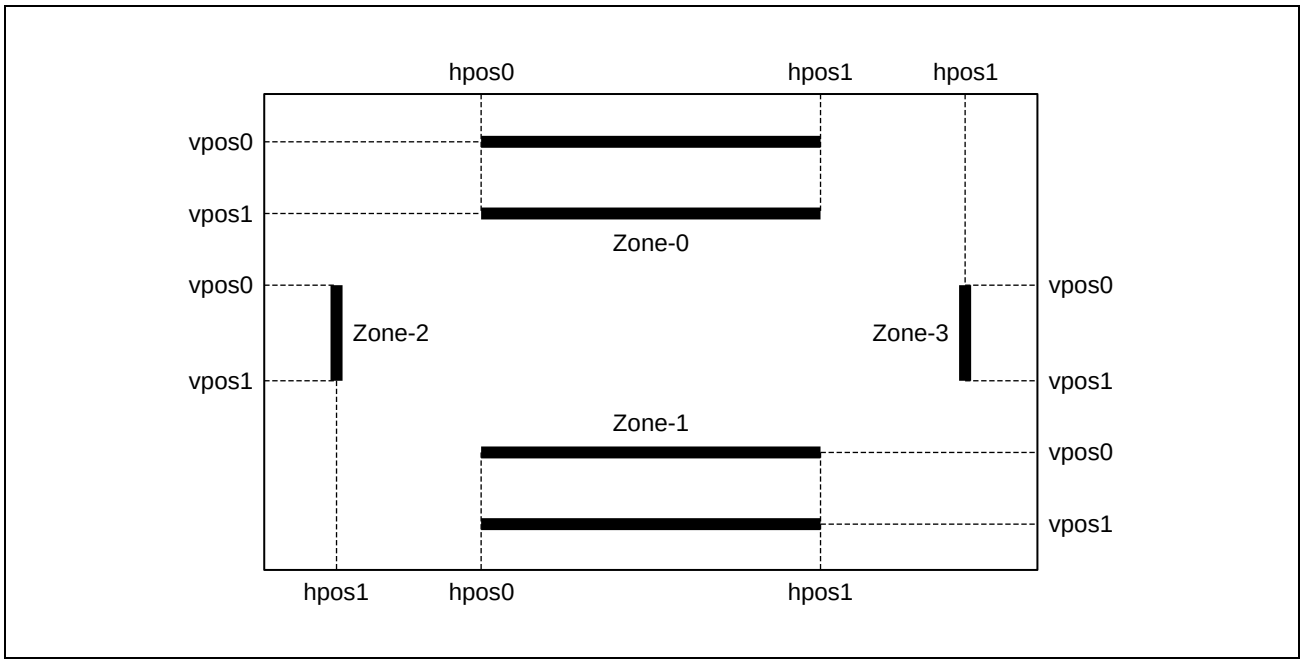


Figure 9.9-43 Letter Box Detection Position Settings for HGT

(7) HGT Vertical Position Register for LB Detection Zone-n (VI6_HGT_LBn_V) (n = 0, 1, 2, 3)

Access Size : 32 bits

Address : <VSPI0_base> + 342Ch/ 3434h/ 343Ch/ 3444h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	VPOS0[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	VPOS1[13:0]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
29 to 16	VPOS0[13:0]	0h	RW	Vertical Start Position for Letter Box Detection Zone-n When all pixels on a line (shown in **Figure 9.9-43**) in the input image have the value equal to or smaller than the threshold value, the HGT determines that there is a letter box. In these bits, specify the value of vpos0 of the detection zone-n (**Figure 9.9-43**) in pixel units. A value from 0 to 8189 can be specified. vpos0 ≤ vpos1 should be satisfied.
15, 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13 to 0	VPOS1[13:0]	0h	RW	Vertical Position for Letter Box Detection Zone-n When all pixels on a line (shown in **Figure 9.9-43**) in the input image have the value equal to or smaller than the threshold value, the HGT determines that there is a letter box. In these bits, specify the value of vpos1 of the detection zone-n (**Figure 9.9-43**) in pixel units. A value from 0 to 8189 can be specified. vpos0 ≤ vpos1 should be satisfied.

(8) HGT Histogram Register (VI6_HGT_HISTO_m_n) (m = 0 to 5, n = 0 to 31)

Access Size : 32 bits

Address : <VSPI0_base> + 3450h + 0128h × m + 0004h × n (m = 0 to 5, n = 0 to 31)

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	HISTOGRAM_m_n[25:16]									
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HISTOGRAM_m_n[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
25 to 0	HISTOGRAM_m_n[25:0]	0h	R	Weighted Frequency of Hue Area-m and Saturation Area-n From these bits, the count of weighting for the pixels whose H component value is in hue area m (**Figure 9.9-42**) and whose S component value (val) satisfies the following condition within the histogram detection window (**Figure 9.9-40**) is read (after pixel skipping is applied when skipping mode is selected). $8 \times n \leq val < 8 \times (n + 1)$ The weight is determined by the H component value as shown in **Figure 9.9-42** and the maximum weight is 16. Counting starts after the HGT is activated. To read the histogram, read these bits after the HGT has completed processing of one screen of data and before the HGT is activated again.

(9) HGT Max/Min Value Register (VI6_HGT_MAXMIN)

Access Size : 32 bits
Address : <VSPI0_base> + 3750h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	MAXVAL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	MINVAL[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
23 to 16	MAXVAL[7:0]	0h	R	Maximum Value of S Components From these bits, the maximum value of the S components of the pixels within the histogram detection window (**Figure 9.9-40**) is read (after pixel skipping is applied when skipping mode is selected). Maximum value detection starts after the HGT is activated. To read the maximum value, read these bits after the HGT has completed processing of one screen of data and before the HGT is activated again.
15 to 8	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
7 to 0	MINVAL[7:0]	0h	R	Minimum Value of S Components From these bits, the minimum value of the S components of the pixels within the histogram detection window (**Figure 9.9-40**) is read (after pixel skipping is applied when skipping mode is selected). Minimum value detection starts after the HGT is activated. To read the minimum value, read these bits after the HGT has completed processing of one screen of data and before the HGT is activated again.

(10) HGT Sum Register (VI6_HGT_SUM)

Access Size : 32 bits
Address : <VSPI0_base> + 3754h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	SUMVAL[29:16]													
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SUMVAL[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
29 to 0	SUMVAL[29:0]	0h	R	Sum of S Components From these bits, the sum of the S components of the pixels within the histogram detection window (**Figure 9.9-40**) is read (after pixel skipping is applied when skipping mode is selected). Accumulation starts after the HGT is activated. To read the sum, read these bits after the HGT has completed processing of one screen of data and before the HGT is activated again.

(11) HGT LB Detection Result Register (VI6_HGT_LB_DET)

Access Size : 32 bits

Address : <VSPI0_base> + 3758h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	LTRBOX1	LTRBOX2	SIDE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
2	LTRBOX1	0h	R	Letter Box Detection Result #1 of Zone-0/1 for V Component This bit is set to 1 when none of the V components of the pixels on the lines of zone-0/vpos0 and zone-1/vpos1 (two lines in total) in **Figure 9.9-43** exceeds the threshold (BLACK_TH). If any pixel is greater than the threshold, this bit is set to 0. Read this bit after the HGT has completed processing of one screen of data and before the HGT is activated again. The value read during processing is not guaranteed to be correct.
1	LTRBOX2	0h	R	Letter Box Detection Result #2 of Zone-0/1 for V Component This bit is set to 1 when none of the V components of the pixels on the four lines of zone-0 and 1 in **Figure 9.9-43** exceeds the threshold (BLACK_TH). If any pixel is greater than the threshold, this bit is set to 0. Read this bit after the HGT has completed processing of one screen of data and before the HGT is activated again. The value read during processing is not guaranteed to be correct.
0	SIDE	0h	R	Letter Box Detection Result of Zone-2/3 for V Component This bit is set to 1 when none of the V components of the pixels on the two lines of zone-2 and 3 in **Figure 9.9-43** exceeds the threshold (BLACK_TH). If any pixel is greater than the threshold, this bit is set to 0. Read this bit after the HGT has completed processing of one screen of data and before the HGT is activated again. The value read during processing is not guaranteed to be correct.

(12) HGT Parameter Register Reset (VI6_HGT_REGRST)

Access Size : 32 bits
Address : <VSPI0_base> + 37FCh
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	RCPART	-	-	-	RCLEA
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	W	R	R	R	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
4	RCPART	0h	W	Status Reset Writing 1 to this bit resets counter bits in HGT.
3 to 1	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
0	RCLEA	0h	W	Register Reset Writing 1 to this bit resets all read-only registers **(9.9.1.3.16 (8)** to **9.9.1.3.16 (11)**) to their initial values. This register is write-only and is always read as 0. Note that R/W registers (**9.9.1.3.16 (1)** to **9.9.1.3.16 (7)**) are not affected by write access to this register.

When image partition is used, refer to **9.9.1.4.6 (7) Histogram of HGO/HGT**.

(13) HGT Histogram Storing Address Register (VI6_HGT_HISTADD)

Access Size : 32 bits
Address : <VSPI0_base> + 376Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HISTADD[23:8]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HISTADD[7:0]								-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	HISTADD[23:0]	0h	RW	HGT histogram storing address These bits specify the start address of storing the HGT histogram plane to be written in case of enabling histogram auto write (VI6_HGT_MODE.AUTOW = 1). Start address of HGT histogram can be set in 256-byte units. And HISTADD [31:8] × 256 is recognized as start address to write HGT histogram.
7 to 0	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.

(14) HGT Histogram Swapping Register (VI6_HGT_HSWAP)

Access Size : 32 bits

Address : <VSPI0_base> + 3770h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	H_LWS	H_WDS	H_BTS
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
2	H_LWS	0h	RW	HGT histogram Swapping in Longword Units 0b: Data swapping in longword (32-bit) units is disabled 1b: Data swapping in longword (32-bit) units is enabled
1	H_WDS	0h	RW	HGT histogram Swapping in Word Units 0b: Data swapping in word (16-bit) units is disabled 1b: Data swapping in word (16-bit) units is enabled
0	H_BTS	0h	RW	HGT histogram Swapping in Byte Units 0b: Data swapping in byte (8-bit) units is disabled 1b: Data swapping in byte (8-bit) units is enabled

See **Figure 9.9-41** and **Table 9.9-32** for setting HGT histogram swap.

9.9.1.3.17 SHP Control Registers

(1) SHP Control Register0 (VI6_SHP_CTRL0)

Access Size : 32 bits
Address : <VSPI0_base> + 3E00h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Limit0[7:0]								Gain0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SIGN	SHP_EN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Limit0[7:0]	0h	RW	Sharpness Parameter Limit0 There are 24 types of setting value to apply sharpness as shown in **Table 9.9-33** and 24 types of setting value to apply blurring as shown in **Table 9.9-34**. Specify an appropriate value from among its combination.
23 to 16	Gain0[7:0]	0h	RW	Sharpness Parameter Gain0 There are 24 types of setting value to apply sharpness as shown in **Table 9.9-33** and 24 types of setting value to apply blurring as shown in **Table 9.9-34**. Specify an appropriate value from among its combination.
15 to 2	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
1	SIGN	0h	RW	Sharpness Parameter SIGN There are 24 types of setting value to apply sharpness as shown in **Table 9.9-33** and 24 types of setting value to apply blurring as shown in **Table 9.9-34**. Specify an appropriate value from among its combination.
0	SHP_EN	0h	RW	Sharpness Processing Enable/Disable 0: Sharpness processing is disabled (input goes through the SHP without change) 1: Sharpness processing is enabled When this bit is set to 1, input format to the SHP module should be YCbCr format. Effect of sharpness is applied to Y component, not applied to Cb, Cr component. Effect of sharpness is applied to horizontal direction, not applied to vertical direction.

To execute sharpness processing in YCbCr format, set up the RPF and DPR appropriately so that YCbCr-format image data is input to the SHP. Refer to **9.9.1.3.6 (3) RPF0 Input Format Registers (VI6_RPF0_INFMT)**. For DPR settings, refer to **9.9.1.3.8 (1) Concept of DPR Settings** to **9.9.1.3.8 (4) {mod} Routing Register (VI6_DPR_{mod}_ROUTE) ({mod} = SRU, UDS, LUT, CLU, HST, HSI, SHP)**. Refer to **Table 9.9-33** to apply sharpness. The SHP also has a function of blurring. Refer to **Table 9.9-34** to apply blurring.

Table 9.9-33 Sharpness Parameter Setting (Sharpness)

Intensity		VI6_SHP_CTRL0				VI6_SHP_CTRL1				VI6_SHP_CTRL2			
		shp_en	sign	Gain0	Limit0	Gain10	Limit10	Gain11	Limit11	Gain20	Limit20	Gain21	Limit21
Enhance weak	1	1	0	4	64	0	64	0	0	0	64	0	0
	2	1	0	8	64	0	64	0	0	0	64	0	0
	3	1	0	12	64	0	64	0	0	0	64	0	0
	4	1	0	16	64	0	64	0	0	0	64	0	0
	5	1	0	20	64	0	64	0	0	0	64	0	0
	6	1	0	24	64	0	64	0	0	0	64	0	0
	7	1	0	24	64	16	64	224	255	0	64	0	0
	8	1	0	24	64	32	64	224	255	0	64	0	0
	9	1	0	24	64	48	64	224	255	0	64	0	0
	10	1	0	24	64	64	64	224	255	0	64	0	0
	11	1	0	24	64	80	64	224	255	0	64	0	0
	12	1	0	24	64	96	64	224	255	0	64	0	0
	13	1	0	24	64	120	64	224	255	0	64	0	0
	14	1	0	24	64	144	64	224	255	0	64	0	0
	15	1	0	24	64	168	64	224	255	0	64	0	0
	16	1	0	24	64	192	64	224	255	0	64	0	0
	17	1	0	24	64	192	64	224	255	24	64	224	255
	18	1	0	24	64	192	64	224	255	48	64	224	255
	19	1	0	24	64	192	64	224	255	72	64	224	255
	20	1	0	24	64	192	64	224	255	96	64	224	255
	21	1	0	26	64	208	64	224	255	104	64	224	255
	22	1	0	28	64	224	64	224	255	112	64	224	255
	23	1	0	30	64	240	64	224	255	120	64	224	255
Enhance strong	24	1	0	32	64	255	64	224	255	128	64	224	255

Table 9.9-34 Sharpness Parameter Setting (Blurring)

Intensity		VI6_SHP_CTRL0				VI6_SHP_CTRL1				VI6_SHP_CTRL2			
		shp_en	sign	Gain0	Limit0	Gain10	Limit10	Gain11	Limit11	Gain20	Limit20	Gain21	Limit21
Blur weak	1	1	1	3	64	0	64	0	0	0	64	0	0
	2	1	1	6	64	0	64	0	0	0	64	0	0
	3	1	1	8	64	0	64	0	0	0	64	0	0
	4	1	1	11	64	0	64	0	0	0	64	0	0
	5	1	1	14	64	0	64	0	0	0	64	0	0
	6	1	1	16	64	0	64	0	0	0	64	0	0
	7	1	1	16	64	3	64	0	0	0	64	0	0
	8	1	1	16	64	6	64	0	0	0	64	0	0
	9	1	1	16	64	8	64	0	0	0	64	0	0
	10	1	1	16	64	11	64	0	0	0	64	0	0
	11	1	1	16	64	14	64	0	0	0	64	0	0
	12	1	1	16	64	16	64	0	0	0	64	0	0
	13	1	1	16	64	19	64	0	0	0	64	0	0
	14	1	1	16	64	22	64	0	0	0	64	0	0
	15	1	1	16	64	24	64	0	0	0	64	0	0
	16	1	1	16	64	27	64	0	0	0	64	0	0
	17	1	1	16	64	30	64	0	0	0	64	0	0
	18	1	1	16	64	32	64	0	0	0	64	0	0
	19	1	1	16	64	32	64	0	0	3	64	0	0
	20	1	1	16	64	32	64	0	0	6	64	0	0
	21	1	1	16	64	32	64	0	0	8	64	0	0
	22	1	1	16	64	32	64	0	0	11	64	0	0
	23	1	1	16	64	32	64	0	0	14	64	0	0
Blur Strong	24	1	1	16	64	32	64	0	0	16	64	0	0

(2) SHP Control Register1 (VI6_SHP_CTRL1)

Access Size : 32 bits
Address : <VSPI0_base> + 3E04h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Limit11[7:0]								Gain11[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Limit10[7:0]								Gain10[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Limit11[7:0]	0h	RW	Sharpness Parameter Limit11 There are 24 types of setting value to apply sharpness as shown in **Table 9.9-33** and 24 types of setting value to apply blurring as shown in **Table 9.9-34**. Specify an appropriate value from among its combination.
23 to 16	Gain11[7:0]	0h	RW	Sharpness Parameter Gain11 There are 24 types of setting value to apply sharpness as shown in **Table 9.9-33** and 24 types of setting value to apply blurring as shown in **Table 9.9-34**. Specify an appropriate value from among its combination.
15 to 8	Limit10[7:0]	0h	RW	Sharpness Parameter Limit10 There are 24 types of setting value to apply sharpness as shown in **Table 9.9-33** and 24 types of setting value to apply blurring as shown in **Table 9.9-34**. Specify an appropriate value from among its combination.
7 to 0	Gain10[7:0]	0h	RW	Sharpness Parameter Gain10 There are 24 types of setting value to apply sharpness as shown in **Table 9.9-33** and 24 types of setting value to apply blurring as shown in **Table 9.9-34**. Specify an appropriate value from among its combination.

(3) SHP Control Register2 (VI6_SHP_CTRL2)

Access Size : 32 bits
Address : <VSPI0_base> + 3E08h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Limit21[7:0]								Gain21[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Limit20[7:0]								Gain20[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Limit21[7:0]	0h	RW	Sharpness Parameter Limit21 There are 24 types of setting value to apply sharpness as shown in **Table 9.9-33** and 24 types of setting value to apply blurring as shown in **Table 9.9-34**. Specify an appropriate value from among its combination.
23 to 16	Gain21[7:0]	0h	RW	Sharpness Parameter Gain21 There are 24 types of setting value to apply sharpness as shown in **Table 9.9-33** and 24 types of setting value to apply blurring as shown in **Table 9.9-34**. Specify an appropriate value from among its combination.
15 to 8	Limit20[7:0]	0h	RW	Sharpness Parameter Limit20 There are 24 types of setting value to apply sharpness as shown in **Table 9.9-33** and 24 types of setting value to apply blurring as shown in **Table 9.9-34**. Specify an appropriate value from among its combination.
7 to 0	Gain20[7:0]	0h	RW	Sharpness Parameter Gain20 There are 24 types of setting value to apply sharpness as shown in **Table 9.9-33** and 24 types of setting value to apply blurring as shown in **Table 9.9-34**. Specify an appropriate value from among its combination.

9.9.1.3.18 VSPI Internal Registers

(1) IP Version Register (VI6_IP_VERSION)

Access Size : 32 bits
Address : <VSPI0_base> + 3F00h
Initial Value : 0101_1404h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IP_Version[31:16]															
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP_Version[15:0]															
Initial Value	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IP_Version [31:0]	1011404h	R	IP version register to indicate module kind and IP version. VSPI: 0101_1404h

9.9.1.3.19 RPF0 OSD-CLUT table

Refer to **9.9.1.4.5 (1) OSD-CLUT or LUT**.

9.9.1.3.20 LUT table

Refer to **9.9.1.4.5 (1) OSD-CLUT or LUT**.

9.9.1.3.21 CLU table address register and table data register

Refer to **9.9.1.4.5 (2) CLU Table**.

9.9.1.4 Operation

9.9.1.4.1 Registers to set fixed value

Set fixed value to following registers in any case.

[1] Always set 0000_0808h to VI6_CLK_DCSWT

[2] Always set D’256 to VI6_DL_CTRL.AR_WAIT [15:0] in case of using display list.

[3] Always set D’2 to VI6_DL_EXT_CTRLn.POLINT [5:0] in case of using extended display list of WPF0.

[4] Always set D’1 to VI6_DL_EXT_CTRLn.DLPRI in case of using extended display list of WPF0.

[5] Always set D’0 to VI6_DL_EXT_CTRLn.EXPRI in case of using extended display list of WPF0.

[6] Always set D’256 to VI6_WPF0_ROT_CTRL.LMEM_WD [12:0] in case of using rotation.

[7] Always set 0000_0500h toVI6_DPR_WPF0_FPORCH in case of using WPF0.

9.9.1.4.2 Concept of VSPI Operation Starting and Stopping

The VSPI is started by setting the corresponding start register. It means starting of WPF, which is output module of the VSPI. Use the start registers shown in **Table 9.9-35** to start WPF modules.

After a WPF module is started and specified processing is completed, the corresponding channel VSPI stops operation and notifies the end of processing through an end interrupt. End interrupts are generated through the end interrupt source registers shown in **Table 9.9-35**; clearing a source register cancels the corresponding interrupt signal (for details of interrupt processing, refer to **9.9.1.4.4 Interrupt Processing**). Each of the operating status registers shown in the table indicates the busy state after the processing is started through the start register until processing is completed and operation stops. **Figure 9.9-44** shows these operation timings.

Table 9.9-35 Target Module and Corresponding Registers for Starting and Stopping Operation

Target Module	Start Register	End Interrupt Source Register	Operating Status Register
WPF0	VI6_CMD0.STRCMD	VI6_WPF0_IRQ_STA.FRE	VI6_STATUS.SYS0_ACT

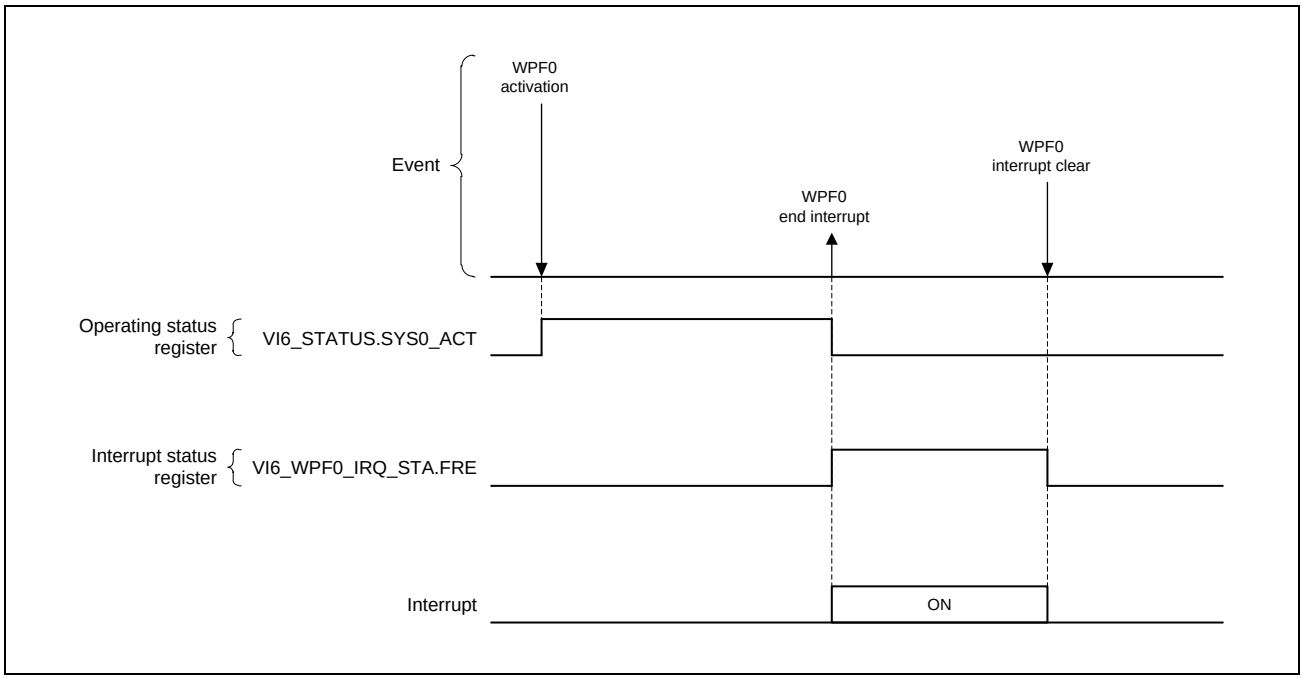


Figure 9.9-44 VSPI Startup and Status of Each Register and Interrupt

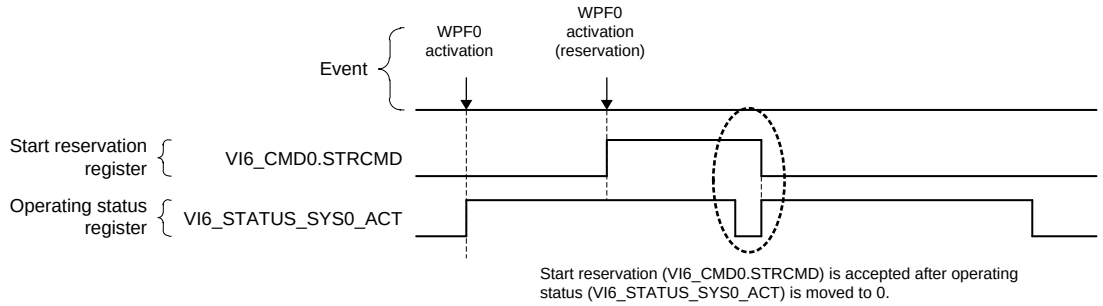
The following describes the operating states (operating or stopped) of VSPI internal modules. As described in **9.9.1.1.1 Block diagram of VSPI**, the VSPI has several image processing modules and the connections between modules are determined by the DPR. Accordingly, the operating state of a module is the same as that of the target WPF for that module.

After connections through the DPR are changed and the target WPF for a module is changed, refer to the correct register for the new target WPF to check the operating state of that module.

Connections should be changed through the DPR-related registers (described later) while all modules to be affected by any change in connections are stopped. If connections through the DPR are changed during operation, the VSPI will hang.

Figure 9.9-45 shows relation between start reservation and operating status. When start is reserved while VSPI is operating, the reservation is accepted after VSPI status is moved from “operating” into “idle” as shown in **Figure 9.9-45**. Therefore, when start reservation is used for VSPI as shown in **9.9.1.4.6 (3) Start Reservation for VSPI** take following procedure to confirm VSPI isn’t operating.

1. Check VI6_CMD0.STRCMD is zero (Start is not reserved)
2. Check VI6_STATUS.SYS0_ACT is zero (VSPI is IDLE status)



Note: WPF0 cannot be executed at same time for VSPI.

Figure 9.9-45 VSPI Start Reservation and Operating Status

9.9.1.4.3 Display List

(1) Functional Description

The VSPI provides the display list function. As a display list, the VSPI automatically downloads the register settings except for the control registers (**9.9.1.3.4 General Control Registers** and **9.9.1.3.5 Display List Control Registers** from external memory and stores the settings in the VSPI registers. This function is advantageous in that the interrupt processing or register setting modification processing can be executed without CPU intervention during multiple-frame processing because the register settings used for VSPI processing are prepared in advance in external memory such as SDRAM.

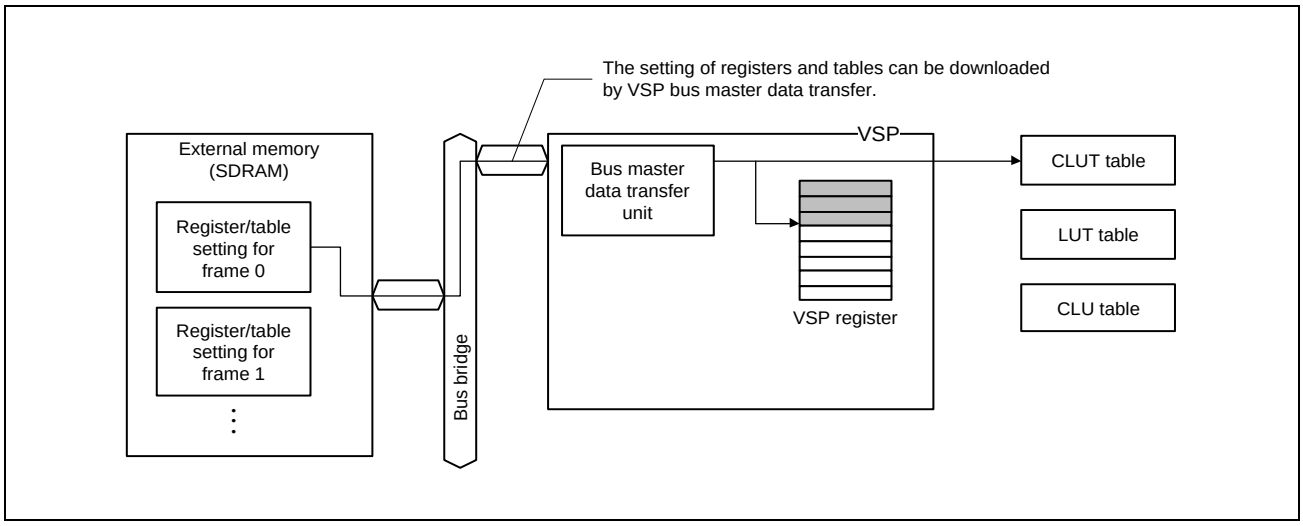


Figure 9.9-46 Concept of Display List

To use display lists, specify the external memory addresses to the display list control registers described in **9.9.1.3.5 Display List Control Registers**. The register settings or various types of information should be stored in external memory in the format described in **9.9.1.4.3 (2) Normal Display List Mode**. **Figure 9.9-47** shows the difference between VSPI operation through normal register settings and through display lists.

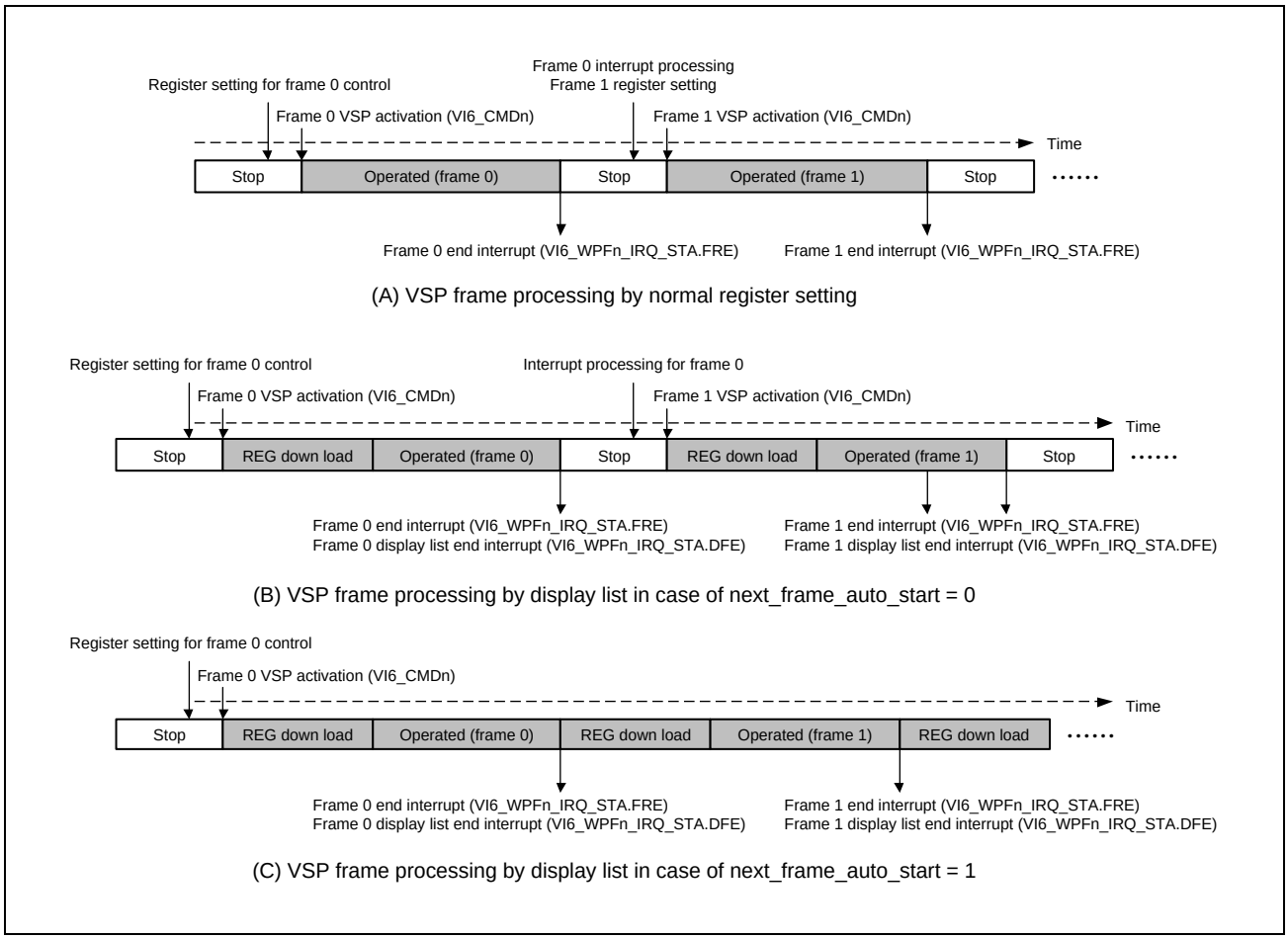


Figure 9.9-47 Comparison of VSPI Operation between Normal Register Settings and Display Lists

As shown in **Figure 9.9-47** (*A*), in the VSPI processing through normal register settings, all registers should be set up before the VSPI is started for each frame. After the VSPI processing is completed, the VSPI outputs a frame end interrupt (VI6_WPF0_IRQ_STA.FRE). This method requires a certain amount of time for register settings or interrupt processing by the CPU between frames. In contrast, when display lists are used, the VSPI automatically downloads register settings from external memory as shown in **Figure 9.9-47** (*B*) and (*C*), which reduces the load on the CPU between frames.

Figure 9.9-47 (*B*) shows the display list usage where the VSPI stops at the end of every frame; only the VSPI start processing for each frame is done by the CPU. This is suitable for the cases when the CPU controls synchronization of frame processing in frame buffer management or when the amount of register values or table data to be set in the VSPI is large. In the case shown in **Figure 9.9-47** (*C*), as soon as the frame processing ends, the VSPI automatically begins next frame operation and starts downloading new register settings. This is the fastest operation using display lists.

Table 9.9-36 shows the modes of the display list and the supported functions for each mode. The detail of each mode is described in the following sections.

Table 9.9-36 Display List Mode and Supported Functions

Mode	Extended Display List	Continuous Frames
Normal Display List Mode	Supported	Controlled by “next frame auto start” in the header of the display list.
Header-less Display List Mode	Not Supported	Controlled by VI6_DL_CTRL.CFM0 bit.

(2) Normal Display List Mode

The VSPI display lists include control information as well as simple register settings in order to control multiple-frame processing in an optimum way for each application. **Figure 9.9-48** shows the display list structure.

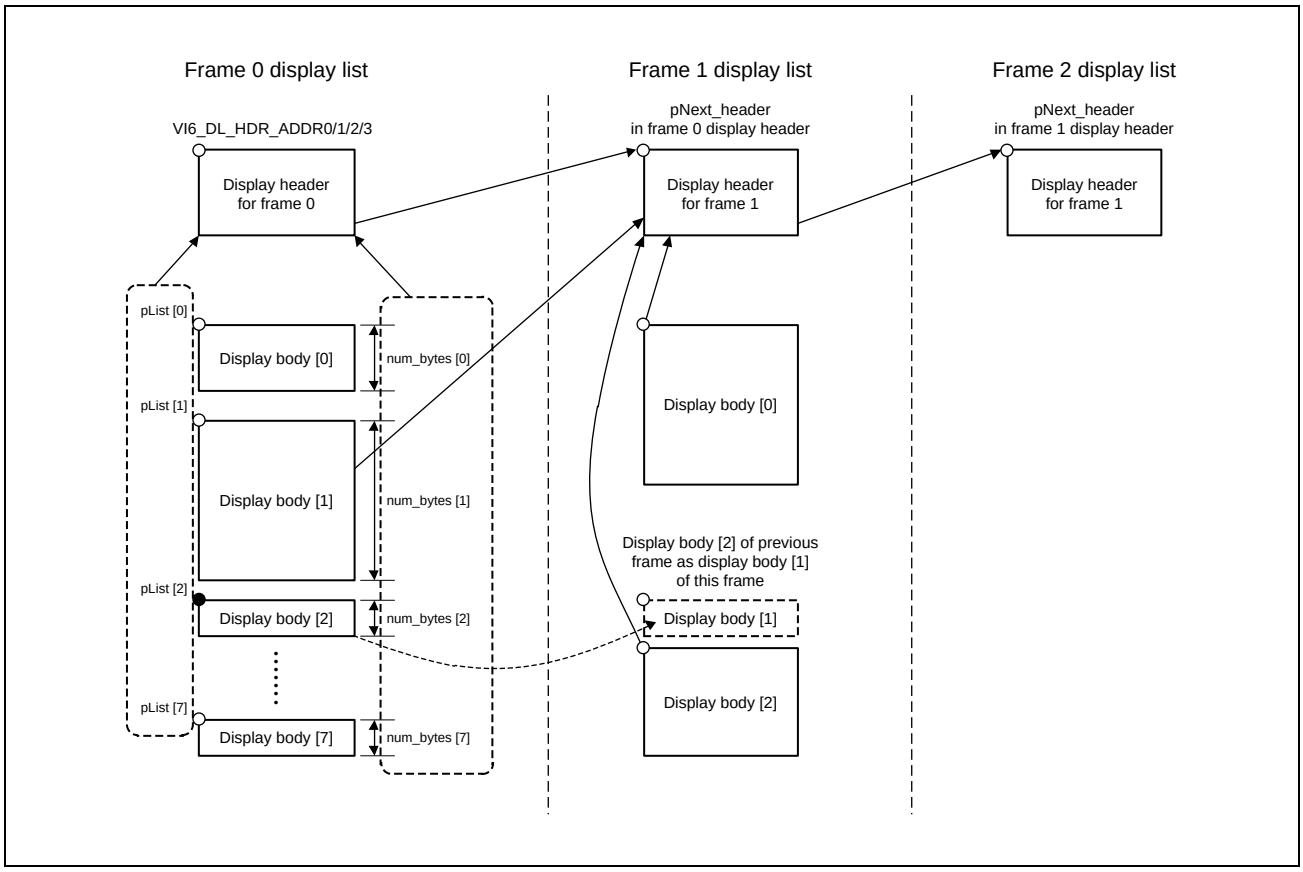


Figure 9.9-48 Structure and Concept of VSPI Display List

A VSPI display list consists of two sections; a header section for storing various information and control flags and a body section for storing register and table settings. A combination of these two sections is defined as a display list for a frame. The register and table settings can be divided and stored in up to eight separate bodies allocated in memory. Therefore, the separate bodies storing the register settings for one frame (for example, frame 0) can have non-sequential start addresses; that is, the bodies for one frame can be allocated to areas distant from each other in memory. To gather these bodies and configure the register settings for one frame, a header is used. A display header stores the number of bodies linked with the header and the start address and size of each body.

The VSPI analyzes the header to gather register and table settings stored in separate memory areas and reconfigures the complete register settings.

The addresses of display headers should be specified in the VSPI registers described in **9.9.1.3.5 Display List Control Registers**. When activated in a mode that uses display lists, the VSPI downloads display headers from the addresses specified in VI6_DL_HDR_ADDR0 (number 0 corresponds to the WPF channel index number), analyzes the numbers of bodies and the address and data size of each body, downloads the bodies, and completes register and table settings. After display list downloading is completed, the VSPI becomes ready for frame image processing; the VSPI then starts the actual frame processing.

After processing of a frame ends, the VSPI proceeds to the next frame processing. Here, there are two modes for starting the next frame processing as shown in **Figure 9.9-47**. In one mode, the VSPI stops operation and waits for the next activation by the CPU in the same way as when display lists are not used. In this mode, the address used for downloading the display header of the next frame is kept by the internal hardware. Therefore, if valid address information is not stored in the display header for the previous frame, a correct value should be specified in VI6_DL_HDR_ADDR0 while the VSPI is stopped. When the VSPI is started after a correct value is specified, the VSPI starts next frame processing with the same procedure for the previous frame. In contrast to this mode, which stops the VSPI after the end of one-frame processing, there is another mode for automatically starting next frame processing. In automatic start mode, the VSPI downloads the next display header as soon as the previous frame processing ends. After downloading ends, the VSPI starts image processing. The information regarding mode selection, that is, whether to automatically start next frame processing, should be stored in the display header downloaded for the previous frame.

In the automatic start mode, the VSPI continues processing until the display header for a frame specifies that the next frame should not be started automatically. To stop processing during automatic execution, use a software reset (VI6_SRESET).

To strictly define the display list format described above, the following shows the grammatical structure of a display list using pseudo-code. First, to simplify the description of the display list format in the following pages, **Table 9.9-37** defines a function. Function zero bits (num bits) generates a string of one-bit 0s for the number of bits specified by the parameter for the function. By using this function, **Table 9.9-38** defines the header section format of a display list and **Table 9.9-39** defines the body section format and **Table 9.9-40** defines the extended display list body section format.

Data order of Display List Header Section, Display List Body Section and Extended Display List Body Section are assumed as the data is stored in SDRAM by big endian. **Table 9.9-41** shows an example of data order. If data order is not same as the definition, data order within 8byte unit can be adjusted by setting VI6_DL_SWAP (**9.9.1.3.5 (3) Display List-0 Data Swapping Register (VI6_DL_SWAP0)**).

Table 9.9-37 Definition of a Function for Simple Description

Syntax	Bit Count
zero_bits (num_bits)	
{	
for (i = 0; i<num_bits; i++) {	
zero_bit	1
}	
}	

Bit String	Contents
zero_bit	zero_bit indicates a 1-bit integer having a value of 0.

Table 9.9-38 Format of Display List Header Section

Syntax	Bit Count
display_header () /* Fixed length */	
{	
zero_bits (29)	
num_list_minus1	3
for (i=0; i<8; i++)	
zero_bits (15)	
num_bytes [i]	17
pList [i]	32
}	
pNext_header	32
zero_bits (30)	
current_frame_int_enable	1
next_frame_auto_start /* 76 bytes from the beginning of this header*/	1
if (VI6_DL_EXT_CTRL.EXT) {	
zero_bits (32) /* padding zero 4 bytes for alignment */	
zero_bits (6)	
pre_ext_dl_exec	1
post_ext_dl_exec	1
zero_bits (8)	
pre_ext_dl_num_cmd	16
pre_ext_dl_pList	32
zero_bits (16)	
post_ext_dl_num_cmd	16
post_ext_dl_pList /* 96 bytes from the beginning of this header*/	32
}	
}	

Bit String	Contents
num_list_minus1	Specifies the value obtained by subtracting 1 from the total number of display list bodies linked with the display header. For example, when this bit field is set to 0, this display list uses one body.
num_bytes [i]	Specifies the number of bytes in the i-th display list body (indicated by index i). Be sure to specify a multiple of eight bytes. For the bodies that are not defined in num_list_minus1 (for example, i = 5 to 7 when num_list_minus1 is set to 4), specify 0.
pList [i]	Specifies the start address of the i-th display list body (indicated by index i). Be sure to specify an address aligned with an 8-byte boundary (the lower-order three bits are 0). For the bodies that are not defined in num_list_minus1 (for example, i = 5 to 7 when num_list_minus1 is set to 4), specify 0.
pNext_header	Specifies the address of the display list header for the next frame. After display list downloading ends, the VSPI keeps its value in the internal memory and uses it in the next display list header downloading. Be sure to specify an address aligned with an 8-byte boundary (the lower-order three bits are 0).
current_frame_int_enable	This is a flag that indicates whether to set the display list end interrupt source (VI6_WPF0_IRQ_STA.DFE) to 1 when the current frame processing ends. If this flag is set to 0, the display list end interrupt source (VI6_WPF0_IRQ_STA.DFE) is not set to 1 when one-frame processing by this display header ends. In this state, even if the display list end interrupt is enabled (VI6_WPF0_IRQ_ENB.DFEE is set to 1), no interrupt will be generated. If this flag is set to 1, the display list end interrupt source (VI6_WPF0_IRQ_STA.DFE) is set to 1 when one-frame processing by this display header ends. In this state, if the display list end interrupt is enabled (VI6_WPF0_IRQ_ENB.DFEE is set to 1), the VSPI generates an interrupt.
Next_frame_auto_start	Enables or disables automatic start of next frame processing when one-frame processing by this display header ends. If this bit is set to 1, the VSPI starts next frame processing as soon as one-frame processing by this display header ends, and starts downloading the next frame display header from the pNext_header address specified in this display header. If this bit is set to 0, the VSPI stops operation when one-frame processing by this display header ends. In this case, start the VSPI through VI6_CMDn to process the next frame.
pre_ext_dl_exec	Enables execution of the extended display list for frame preprocessing when VI6_DL_EXT_CTRL.EXT is 1. If this bit is set to 1, the VSPI executes the extended display list for frame preprocessing. The VSPI does not execute it if this bit is set to 0. This bit setting is ignored when VI6_DL_EXT_CTRL.EXT is 0.
post_ext_dl_exec	Set 0 to this bit.
pre_ext_dl_num_cmd	Specifies the number of commands in the extended display list body section for frame preprocessing when VI6_DL_EXT_CTRL.EXT is 1. The number of commands that can be specified is 1, and a command is 16 bytes. When pre_ext_dl_exec is set to 0, the extended display list for frame preprocessing is not executed; specify 0 in this bit. This bit setting is ignored when VI6_DL_EXT_CTRL.EXT is 0.
pre_ext_dl_pList	Specifies the start address of the area where the extended display list body section for frame preprocessing is stored when VI6_DL_EXT_CTRL.EXT is 1. Be sure to specify an address aligned with a 16-byte boundary (lower-order four bits are 0). When pre_ext_dl_exec is set to 0, the extended display list for frame preprocessing is not executed; specify 0 in this bit. This bit setting is ignored when VI6_DL_EXT_CTRL.EXT is 0.
post_ext_dl_num_cmd	Set 0 to this bit.
post_ext_dl_pList	Set 0 to this bit.

Table 9.9-39 Format of Display List Body Section

Syntax	Bit Count
display_list (num_bytes) /* Variable length (num_bytes) */	
{	
for (i=0; i<num_bytes; i+=8) {	
set_address	32
set_data	32
}	
}	

Bit String	Contents
set_address	Specifies the address where the value specified by set_data is to be stored. Specify a register address. *Note*: Register space of VSPI is 32 Kbyte (0000h - 7FFCh). So, set 0 to upper 17-bit of set_address in Display List Body Section.
set_data	Specifies the value to be stored in the address specified by set_address. Specify a value to be set in a register.

Table 9.9-40 Format of Extended Display List Body Section

Syntax	Bit Count
ext_dl_display_list (num_llw) /* Variable length (pre/post_ext_dl_num_bytes) */	
{	
for (i=0; i<num_llw; i+=2) {	
ext_dl_cmd	64
ext_dl_data	64
}	
}	

For setting extended display list body section.

Table 9.9-41 Data Order of Display List Body

Address	Syntax
pList	set_address [0] (bit31-24)
pList + 1	set_address [0] (bit23-16)
pList + 2	set_address [0] (bit15-8)
pList + 3	set_address [0] (bit7-0)
pList + 4	set_data [0] (bit31-24)
pList + 5	set_data [0] (bit23-16)
pList + 6	set_data [0] (bit15-8)
pList + 7	set_data [0] (bit7-0)
pList + 8	set_address [1] (bit31-24)
pList + 9	set_address [1] (bit23-16)
pList + 10	set_address [1] (bit15-8)
pList + 11	set_address [1] (bit7-0)
pList + 12	set_data [1] (bit31-24)
pList + 13	set_data [1] (bit23-16)
pList + 14	set_data [1] (bit15-8)
pList + 15	set_data [1] (bit7-0)
~~	~~~

(3) Header-less Display List Mode

The header-less display list does not have the display header listed in **Table 9.9-38**, and it has the simplest structure that has only single body. The extended display list function is not available in case of the header-less display list mode. Set the value 1 to VI6_DL_CTRL.NH0 (see the **9.9.1.3.5 (1) Display List Control Register (VI6_DL_CTRL)**) to use the header-less display list. The start address downloaded by the header-less display list should be set to VI6_DL_HDR_ADDR0. And the size of the display body which is originally defined in the display header should be set to VI6_DL_BODY_SIZE0. The header-less display list is available only in WPF0.

(4) Restrictions on Display List Usage

Access to the general control registers and display list control registers through a display list is prohibited. When using display lists, be sure to observe the following restrictions on register access by the CPU.

1. Do not execute write access to the same register (same address) from the CPU and through a display list at the same time. If such a conflict occurs, correct operation of the VSPI is not guaranteed.
2. Do not execute write access to the same OSD-CLUT, CLU, or LUT lookup table from the CPU and through a display list at the same time. If such a conflict occurs, correct operation of the VSPI is not guaranteed. Here, the same lookup table means the address space having the same space name shown in **Table 9.9-42** or **Table 9.9-43**.
3. When read access by the CPU and write access through a display list to the same register (same address) occur at the same time, the read value returned to the CPU is not guaranteed.
4. When read access by the CPU and write access through a display list to the same OSD-CLUT, CLU, or LUT lookup table occur at the same time, the read value returned to the CPU is not guaranteed. Here, the same lookup table means the address space having the same space name shown in **Table 9.9-42** or **Table 9.9-43**.
5. For other restrictions on values and timing of register setting through display lists, refer to the restrictions on normal register settings described in **9.9.1.3.3 Restrictions on Access to Registers and Lookup Tables**.

6. Manipulation and setting of the registers described in **9.9.1.3.4 General Control Registers** and **9.9.1.3.5 Display List Control Registers** through a display list is prohibited.
7. The extended display list function enabled by VI6_DL_EXT_CTRL.EXT is to control the registers of the ICB connected with the VSPI. Do not use extended display lists for any other purpose.

9.9.1.4.4 Interrupt Processing

The VSPI provides multiple image processing channels (WPF0) and they can operate simultaneously in parallel. When a WPF module generates an internal source that should be notified, an interrupt signal is output. As internal sources are generated in each WPF independently, the VSPI has the following registers to control interrupts in each WPF.

- WPF0 Interrupt Enable Registers (VI6_WPF0_IRQ_ENB) (refer to **9.9.1.3.4 (12) WPF0 Interrupt Enable Registers (VI6_WPF0_IRQ_ENB)**)
- WPF0 Interrupt Status Registers (VI6_WPF0_IRQ_STA) (refer to **9.9.1.3.4 (13) WPF0 Interrupt Status Registers (VI6_WPF0_IRQ_STA)**)

While multiple WPF modules operate in parallel, they may output interrupt requests at the same time (this is called a multiple interrupt state). Read the WPF0 Interrupt Status Registers (VI6_WPF0_IRQ_STA) in sequence and execute the interrupt response processing for each interrupt source.

CAUTION

To use interrupts, enable them through WPF interrupt enable registers. If an interrupt source register has already been set to 1 for some reason, an unintended interrupt will occur as soon as the corresponding interrupt enable register is set as enabled. To avoid this, before enabling interrupts through WPF interrupt enable registers, be sure to clear all WPF interrupt sources to be enabled to 0. Be careful about this procedure when setting up registers before starting the VSPI.

9.9.1.4.5 Lookup Table Settings

(1) OSD-CLUT or LUT

For a single entry to the OSD-CLUT space (see **Table 9.9-42**) or LUT space (see **Table 9.9-42**) of the VSPI, the OSD-CLUT or LUT data is set by a write access in the format shown in **Figure 9.9-49**.

The entry address of each space is (start address of the space) + (entry number counting from the base point 0 × 4). For example, the address of entry 7 to the LUT space is (offset address) + 7000h + 7 × 4 = (offset address) + 701Ch.

Table 9.9-42 shows the spaces for which entries can be made. Note that if the module that references that space is operating, write accesses to the relevant space are prohibited. For example, while RPF0 is operating, accesses to the entire space of VI6_CLUT2_TBL are prohibited. When a read access is made to the relevant space during operation of the referencing module, undefined values will be read out.

The operating/stopped state of each module in **Table 9.9-42** is the operating state of the WPF to which each module is connected. Determine whether the module is operating or stopped using each WPF operating status bit in the VI6_STATUS register.

Table 9.9-42 OSD-CLUT Space and LUT Space Addresses

Space Name	Space Addresses	Entry Count	Module that References the Space in the Left Column
VI6_CLUT0_TBL	(offset address) + 4000h to (offset address) + 43FCh	256	RPF0
VI6_LUT_TBL	(offset address) + 7000h to (offset address) + 73FCh	256	LUT

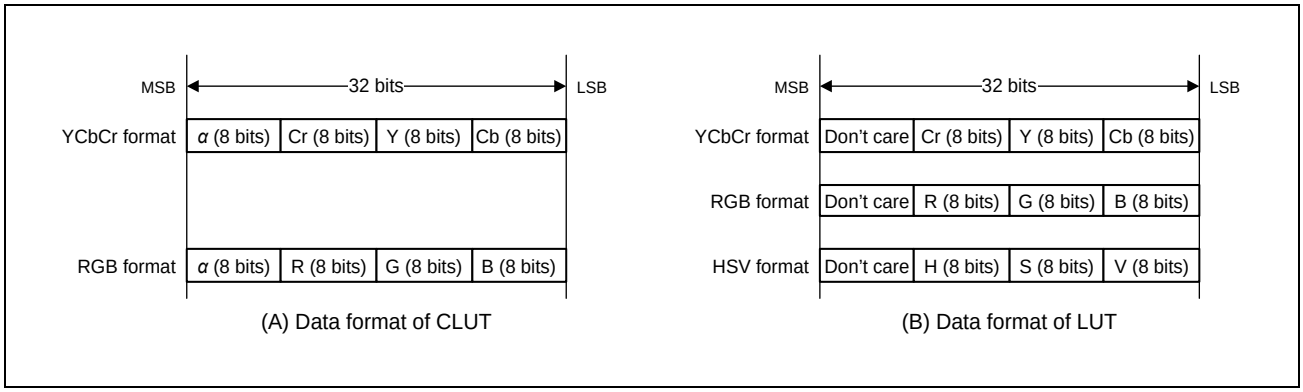


Figure 9.9-49 VI6_CLUT_TBL and VI6_LUT_TBL Formats

(2) CLU Table

Data is set in the CLU table in the space shown in Table 9.9-43 through indirect addressing.

Table 9.9-43 CLU Space Addresses

Space Name	Space Addresses	Module that References the Space in the Left Column
VI6_CLU_TBL	(offset address) + 7400h to (offset address) + 7407h	CLU

The addresses used in indirect addressing (VI6_CLU_ADDR, VI6_CLU_DATA) are shown in **Table 9.9-44**. The CLU table can be accessed by setting the VI6_CLU_ADDR register and then reading from or writing to the VI6_CLU_DATA register. To write-access the CLU table continuously, such as when setting data to the CLU table, perform the following.

1. Set the first CLU table coordinates in VI6_CLU_ADDR.
2. Write the first data to be set in VI6_CLU_DATA.
3. Set the next CLU table coordinates in VI6_CLU_ADDR.
4. Write the next data to be set in VI6_CLU_DATA.
   Execute these processes in a similar manner.

Accesses to VI6_CLU_ADDR and VI6_CLU_DATA registers are repeated in this way. The location where to access the CLU table to write the VI6_CLU_DATA contents immediately takes effect after a value is set in the VI6_CLU_ADDR register.

To read-access the CLU table continuously, change the write-access operation to the read-access operation in the sequence of processes of step 1 to step 4 above.

The following describes how to specify VI6_CLU_ADDR.

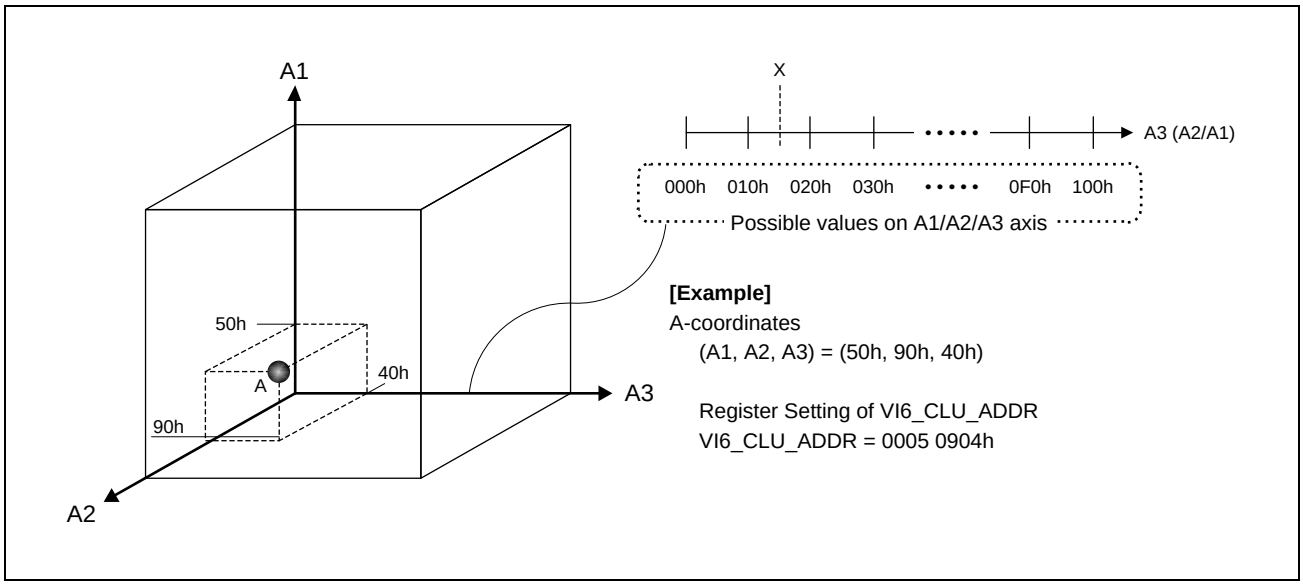


Figure 9.9-50 VI6_CLU_ADDR Setting

In the CLU table, a table value should be specified for every increase by 10h along each of the axes and for every intersection between two axes. For a certain axis, as shown in the top right of **Figure 9.9-50**, a value to be set in the table corresponds to a scale mark at an interval of 10h. The location of a point between two scale marks (for example, point X in **Figure 9.9-50**) is calculated through interpolation by the CLU (refer to **9.9.1.3.12 (1) CLU Control Register (VI6_CLU_CTRL)**). Accordingly, the lower four bits of table settings for each axis has no meaning; only the upper-order five bits of the desired coordinate for each component should be specified in VI6_CLU_ADDR (which should be shifted to the right by four bits) as shown in **Figure 9.9-50**.

Table 9.9-44 Address Spaces Used in Indirect Addressing to CLU Table

Address (Name)	Bit	Initial Value	R/W	Description
(offset address) + 7400h (VI6_CLU_ADDR)	31 to 24	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
	23 to 16	All 0	R/W	Coordinate Value of First Axis These bits set the coordinate value of the first axis on the three-dimensional space coordinates of the CLU table. A value from 0 to 16 can be specified.
	15 to 8	All 0	R/W	Coordinate Value of Second Axis These bits set the coordinate value of the second axis on the three-dimensional space coordinates of the CLU table. A value from 0 to 16 can be specified.
	7 to 0	All 0	R/W	Coordinate Value of Third Axis These bits set the coordinate value of the third axis on the three-dimensional space coordinates of the CLU table. A value from 0 to 16 can be specified when the CLU is in 3D mode. In 2D mode, these bits must be set to 0.
(offset address) + 7404h (VI6_CLU_DATA)	31 to 24	Undefined	R	Reserved These bits are always read as 0. The write value should always be 0.
	23 to 16	Undefined	R/W	Component Value of First Axis These bits set the component value of the first axis on the three-dimensional space coordinates of the CLU table, which is defined by the VI6_CLU_ADDR register. A value from 0 to 255 can be specified when the CLU is in 3D mode. In 2D mode, these bits must be set to 0.
	15 to 8	Undefined	R/W	Component Value of Second Axis These bits set the component value of the second axis on the three-dimensional space coordinates of the CLU table, which is defined by the VI6_CLU_ADDR register. A value from 0 to 255 can be specified.
	7 to 0	Undefined	R/W	Component Value of Third Axis These bits set the component value of the third axis on the three-dimensional space coordinates of the CLU table, which is defined by the VI6_CLU_ADDR register. A value from 0 to 255 can be specified when the CLU is in 3D mode. In 2D mode, these bits must be set to 0.

Accessing way of “Automatic Table Address Increment (VI6_CLU_CTRL.AAI = 1)” is shown below.

(1) Set 1 in VI6_CLU_CTRL.AAI

(2) Set 0000_0000h in VI6_CLU_ADDR (initial coordinate)

(3) Write the data in VI6_CLU_DATA by following order (pseudo-code).

```
for (ax3 = 0; ax3 < 17; ax3 ++) {
   for (ax2 = 0; ax2 < 17; ax2 ++) {
      for (ax1 = 0; ax1 < 17; ax1 ++) {
         Write the data in VI6_CLU_DATA
         for the coordinate of “VI6_CLU_ADDR = (ax1 << 16) + (ax2 << 8) + ax3”.
      }
   }
}
```

9.9.1.4.6 Image partition for VSPI processing

(1) Necessity of image partition

Because SRU, UDS and WPF/Rotation (VI6_WPF0_OUTFMT.ROT > 1) need to satisfy horizontal size restriction as shown in **Table 9.9-45**. Several executions of VSPI by partitioning one frame's image is needed to realize larger size in horizontal direction. In the chapter, the method is called as image partition. This section shows how to realize image partition for VSPI.

When either restriction shown in **Table 9.9-45** applies, image partition method is needed.

Table 9.9-45 Condition of Necessity to Take Image Partition

Location of Restriction	Max Horizontal Size	Condition of Necessity to Take the Restriction.
Input to UDS after clipping horizontal size by VI6_UDS_HSZCLIP register	304 pixels	UDS is used by up-scaling in horizontal direction.
Output from UDS after clipping horizontal size by VI6_UDS_CLIP_SIZE	304 Pixels	UDS is used by down-scaling or no scaling in horizontal direction.
Input to SRU	288 pixels	SRU is used.
Input to WPF after clipping horizontal size by VI6_WPF0_HSZCLIP	256 pixels	WPF/rotation (VI6_WPF0_OUTFMT.ROT > 1) is used.

To realize image partition, see seven items below.

a) Auto execution of VSPI by using automatic start of next frame processing (next_frame_auto_start) of DisplayList is recommended. See **9.9.1.4.6 (2) Auto execution by using display list** for more detail.

b) Start reservation is available to reduce overhead of interrupt response. See **9.9.1.4.6 (3)Start Reservation for VSPI** to use start reservation.

c) Discontinuous line is generated by VSPI end processing at partition boundary. So, reading overlapped pixels with neighbor partition and clipping discontinuous line at partition boundary is needed. **Figure 9.9-51** shows example of overlap pixel and clipping in case of 3 partition. See **9.9.1.4.6 (4) Reading overlapped pixels and clipping discontinuous line** for more detail.

d) The number of taps for each function's filter in horizontal direction is shown in **9.9.1.4.6 (5) The number of taps for each function's filter in horizontal direction for user to realize how many overlappe**d pixels to be read and how many pixels to be clipped.

e) Horizontal size clipping registers are explained in **9.9.1.4.6 (6) Horizontal size clipping registers for image partition**.

f) How to set HGO and HGT registers related to image partition is shown in **9.9.1.4.6 (7) Histogram of HGO/HGT**.

g) There are some restrictions when VSPI image partition is used. See **9.9.1.4.6 (8) Restriction when VSPI image partition is used**.

src_d_0: Input partition position 0
src_d_1: Input partition position 1

P_0 horizontal input size = $src_h_0 + m_0$
P_1 horizontal input size = $src_h_1 + m_1 + m_2$
P_2 horizontal input size = $src_h_2 + m_3$

$src_h_0 = src_d_0 + 1$
$src_h_1 = src_d_1 - src_d_0$
$src_h_2 = src_h - src_d_1 - 1$

Gray hatching is discontinuous region.

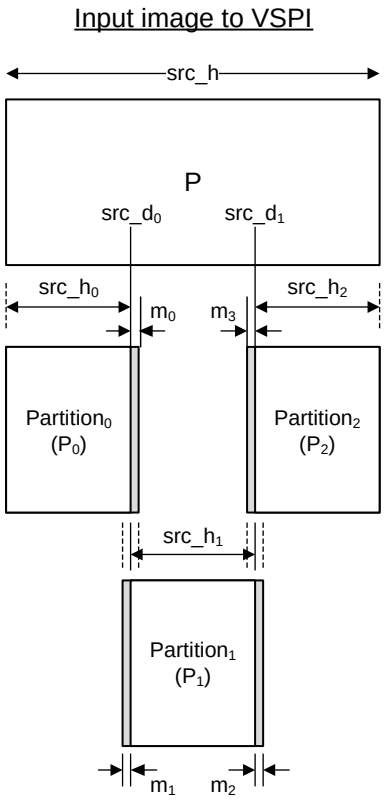


Figure 9.9-51 Example of Overlap Pixel and Clipping in Case of 3 Partitions

(2) Auto execution by using display list

Because partitioning one frame image into several partitioned images, several activations of VSPI for one frame processing are needed. So, using display list and auto execution by next frame auto start is recommended. Detail way is shown below.

a) Store all partitions' display list in SDRAM as shown in **Figure 9.9-52**.

b) In order to get better performance, don't set registers which are not updated per partition in display body from 2nd partition to last partition as shown in **Figure 9.9-52**. Registers which need to be updated per partition is listed in **Table 9.9-46**.

c) Set 1 to next_frame_auto_start in display list header from 1st partition to 2nd partition from the last. Then S/W has to activate VSPI only once for 1st partition.

d) Set 1 to current_frame_int_enable in display list header for only last partition. Set 0000_0002h to VI6_WPF0_IRQ_ENB to enable Display List Frame End (DFE) interrupt and to disable Frame End (FRE) interrupt. Then DFE interrupt occurs after all partitions' processing is finished by VSPI-H/W.

e) Set below to VI6_DL_CTRL

- VI6_DL_CTRL.DLE0 = 1 (Use display list)
- VI6_DL_CTRL.NH0 = 0 (Use Header Mode)
- VI6_DL_CTRL.RLM0 = 1 (Registers with two planes are not downloaded into H/W side at partition boundary)

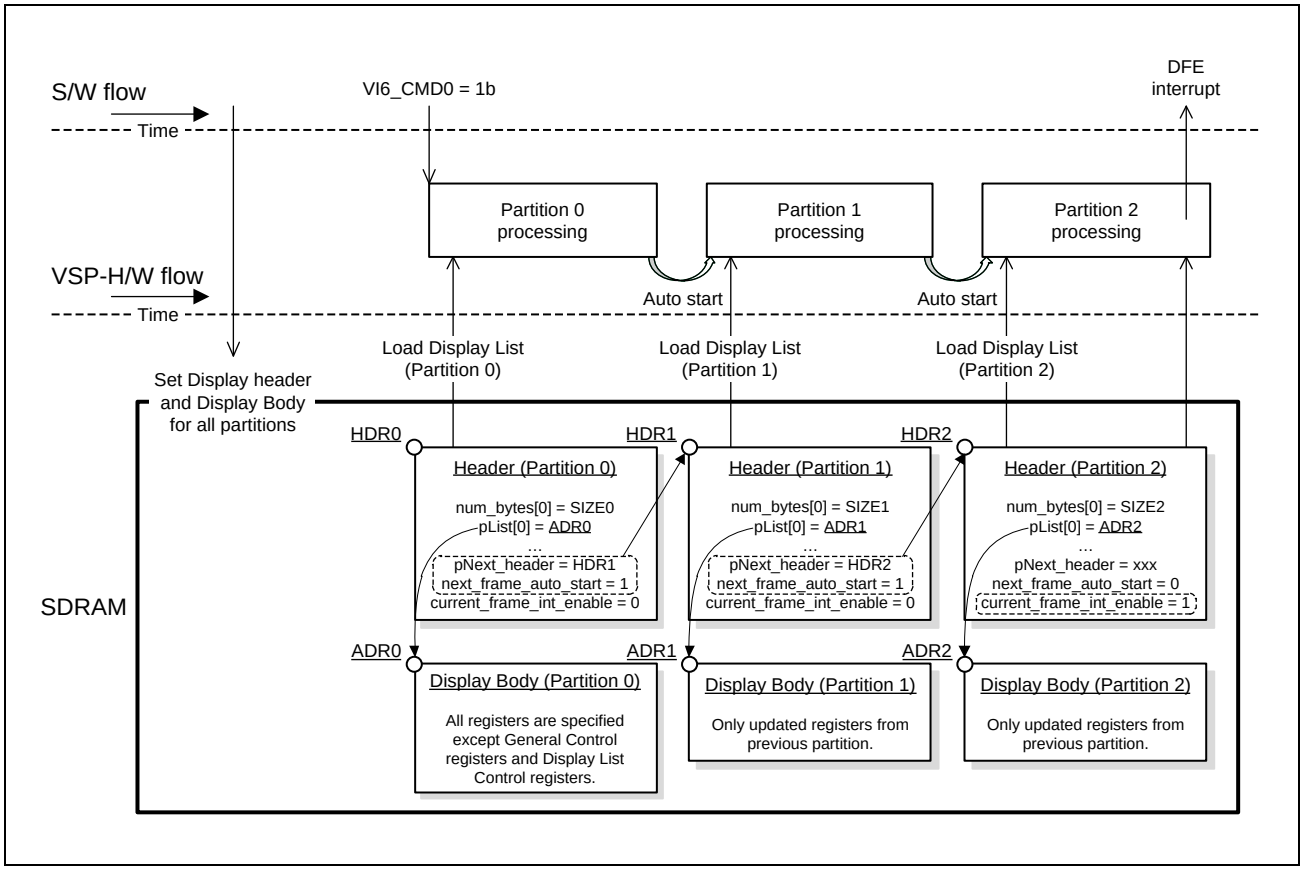


Figure 9.9-52 Flow of Image Partition Process (Example of 3 Partitions Case)

Table 9.9-46 Registers which Need to be Updated per Partition

1	VI6_RPF0_SRC_BSIZE
2	VI6_RPF0_SRC_ESIZE
3	VI6_RPF0_SRCM_ADDR_Y
4	VI6_RPF0_SRCM_ADDR_C0
5	VI6_RPF0_SRCM_ADDR_C1
6	VI6_RPF0_SRCM_ADDR_AI
7	VI6_WPF0_HSZCLIP
8	VI6_WPF0_DSTM_ADDR_Y
9	VI6_WPF0_DSTM_ADDR_C0
10	VI6_WPF0_DSTM_ADDR_C1
11	VI6_UDS_HPHASE
12	VI6_UDS_HSZCLIP
13	VI6_UDS_CLIP_SIZE
14	VI6_HGO_MODE
15	VI6_HGO_OFFSET
16	VI6_HGO_SIZE
17	VI6_HGO_REGRST
18	VI6_HGT_MODE
19	VI6_HGT_OFFSET
20	VI6_HGT_SIZE
21	VI6_HGT_REGRST
22	VI6_DPR_HGO_SMPPT
23	VI6_DPR_HGT_SMPPT

(3) Start Reservation for VSPI

This section shows sequence of software when start reservation is used. Software can reserve starting VSPI before VSPI processing of one frame is completed. Start reservation can remove interrupt response time and software overhead by reserving execution of VSPI before completion of previous frame.

a) Below shows software sequence of start reservation.

b) Below shows software sequence to manage the number of interrupt generation.

Remark Follow **9.9.1.4.6 (2) Auto execution by using display list** about the setting of display list and VI6_WPF0_IRQ_ENB.

(a) Software sequence of start reservation

[1] to [9] shown in **Figure 9.9-53** and **Figure 9.9-54** shows S/W flow to set registers for start reservation of VSPI.

[1] Wait until 0 is read from VI6_CMD0.STRCMD to confirm that starting WPF0 is not reserved.

[2] Wait until 0 is read from VI6_STATUS.SYS0_ACT to confirm WPF0 is IDLE.

[3] Set registers with two planes for frame 0 (1st frame) except DLH (Display List Header) address. Vi6_DL_HDR_ADDR0 has two planes.

[4] Set header address of display list for frame 0 to VI6_DL_HDR_ADDR0. Then VI6_CMD0.UPDHDR is asserted automatically by VSPI to indicate downloading registers with two planes is reserved.

[5] Activate WPF0 for frame 0 (1st frame) by writing 0000_0001h to VI6_CMD0. If next frame's execution of VSPI is needed, go to step [6].

[6] Wait until start reservation is accepted and registers with two planes are downloaded into H/W side. If 0 is read from VI6_CMD0, it means that starting WPF0 is accepted and register with two planes are downloaded into internal registers H/W side refers.

[7] Set registers with two planes for next frame (fr{m}) except DLH (Display List Header) address.

[8] Set header address of display list for next frame (fr{m}) to VI6_DL_HDR_ADDR0. Then VI6_CMD0.UPDHDR is asserted automatically by VSPI to indicate downloading registers with two planes is reserved.

[9] Reserve activating WPF0 for next frame (fr{m}) by writing 0000_0001h to VI6_CMD0.
If next frame's execution of VSPI is needed, go to step [6].

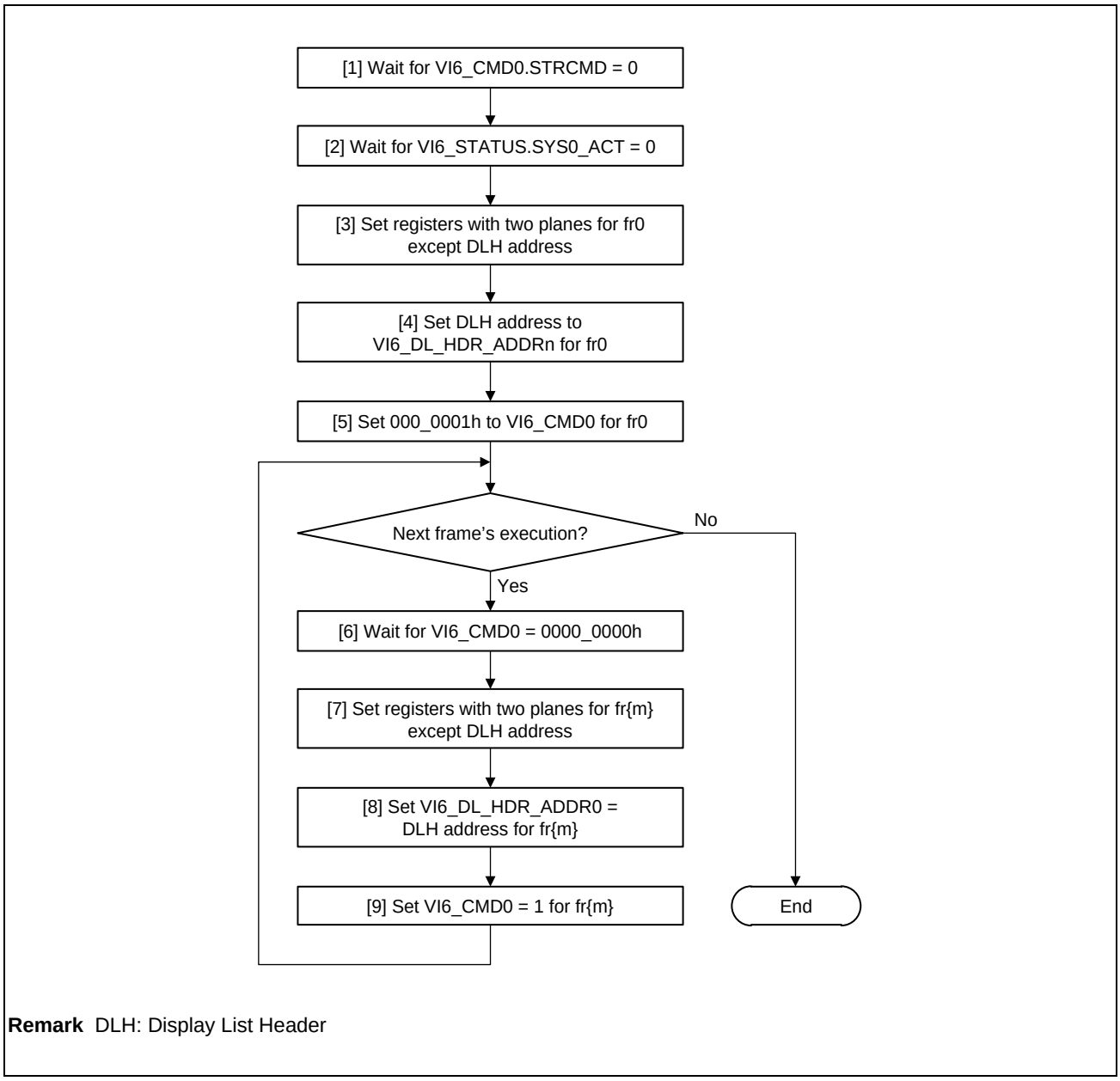


Figure 9.9-53 Flow Chart of Setting Registers by S/W in Case of VSPI Start Reservation

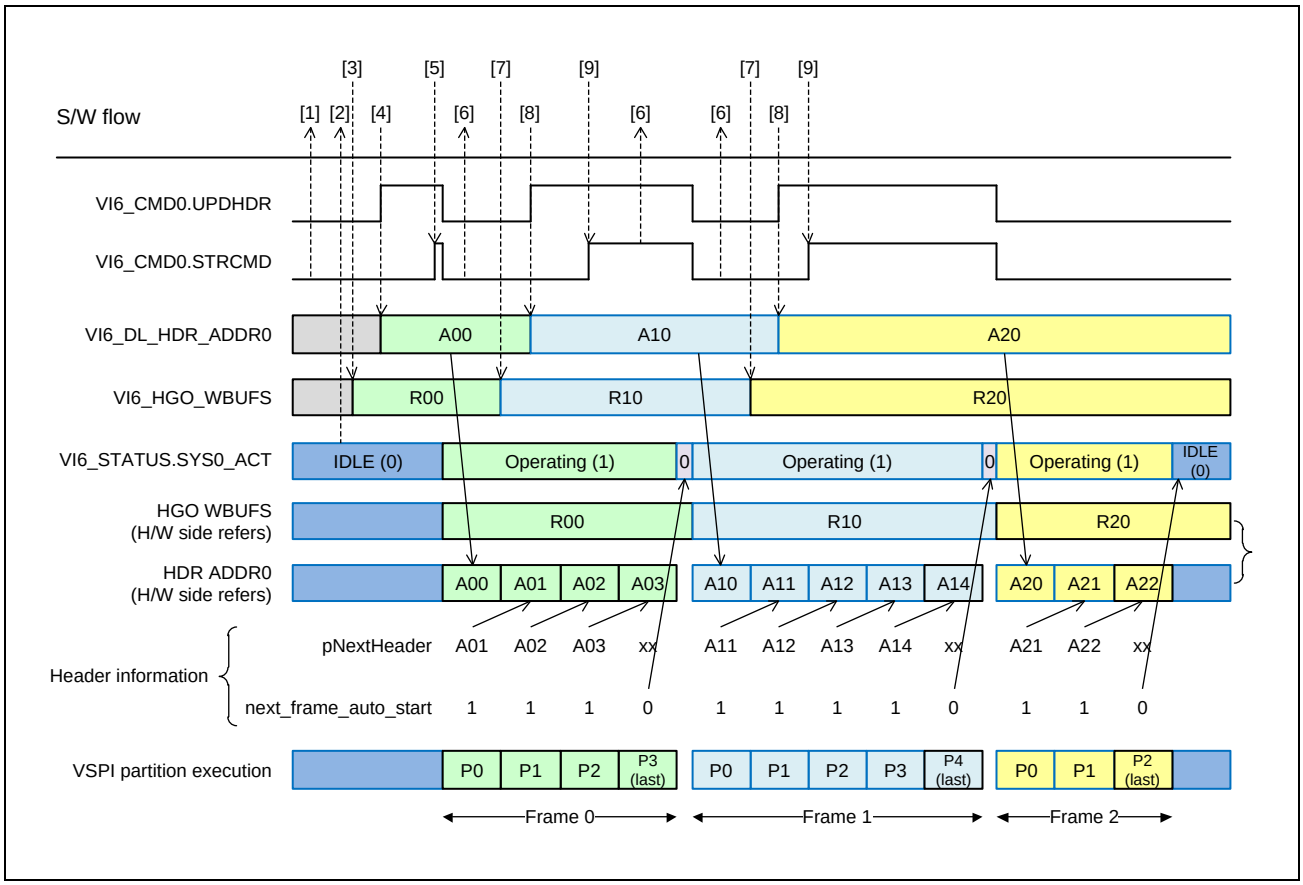


Figure 9.9-54 Timing of Setting Registers by S/W in Case of VSPI Start Reservation

(b) Software sequence to manage the number of interrupt generation.

The way of using interrupt is shown below.

[1] Set 1 to the VI6_WPF0_IRQ_ENB.DFEE to use frame end interrupt (VI6_WPF0_IRQ_STA.DFE)

[2] Write 0 to VI6_WPF0_IRQ_STA.DFE to clear status after interrupt processing as shown in **Figure 9.9-55**.

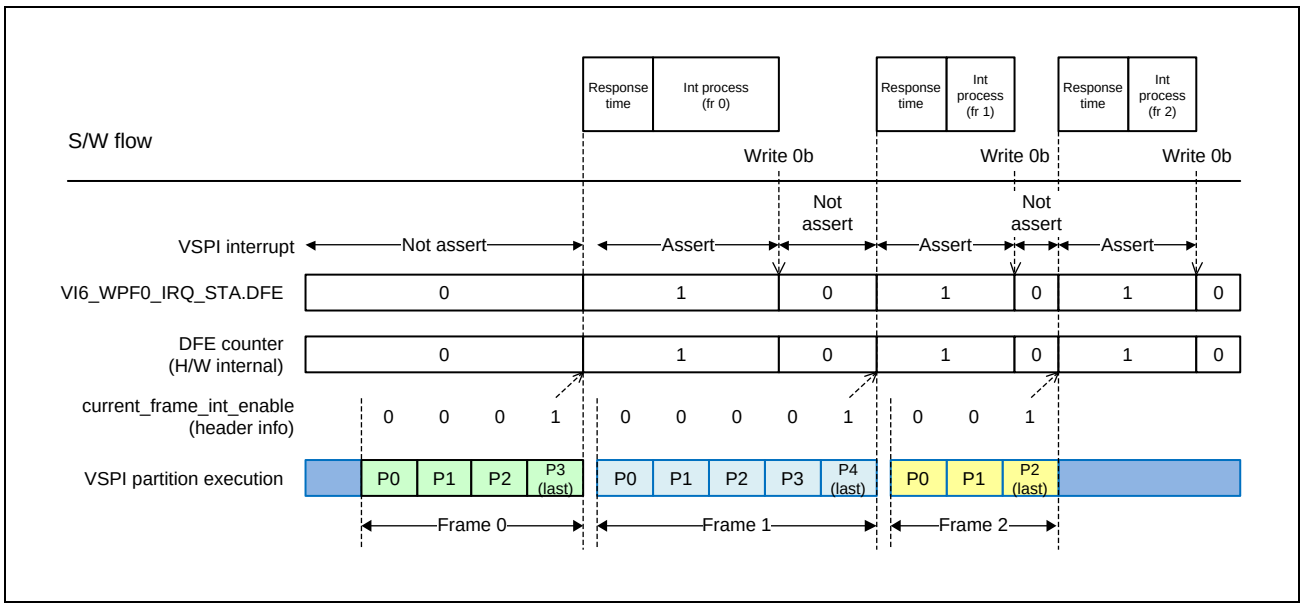


Figure 9.9-55 Timing of Interrupt Processing by S/W in Case of VSPI Start Reservation (Case1)

Be careful following points as shown in **Figure 9.9-56**.

[1] Set 0 to VI6_WPF0_IRQ_STA.DFE for the number of interrupt generation to clear VI6_WPF0_IRQ_STA.DFE.

[2] The VSPI H/W can count at most two of interrupt generation. If three times of interrupts occur before clearing the interrupt factor, interrupt counter overflows. Therefore, wait to reserve starting VSPI of frame {n+2} until interrupt factor of the frame {n} is cleared.

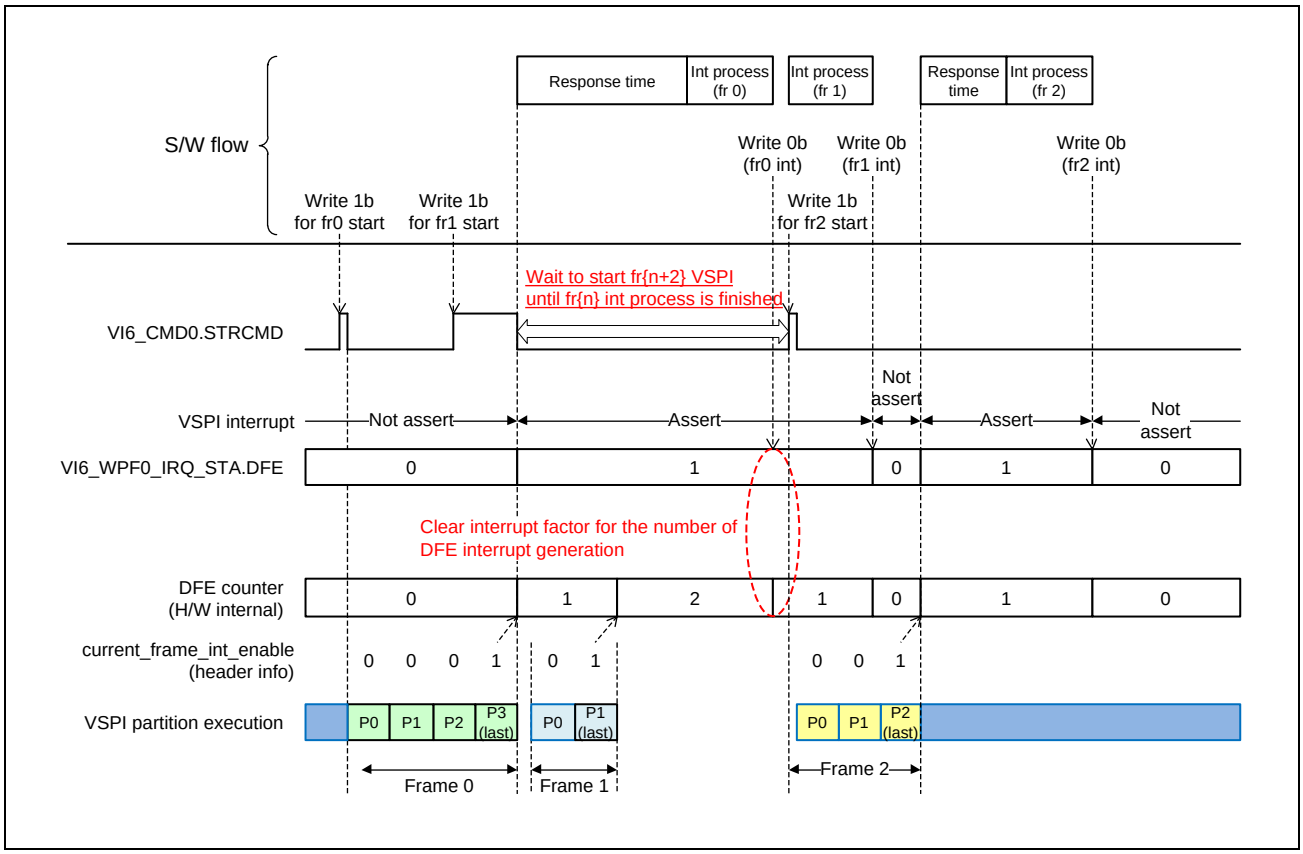


Figure 9.9-56 Timing of Interrupt Processing by S/W in Case of VSPI Start Reservation (Case2)

(4) Reading overlapped pixels and clipping discontinuous line

The VSPI processes partition boundary as end processing, and discontinuous line is generated at partition boundary. **Figure 9.9-57** shows why reading overlapped pixels and clipping discontinuous line are needed. As example that 3 tap filter is used inside VSPI, right neighbor pixel and left neighbor pixel are referred to generate output pixel. (*a*), (*b*) and (*c*) pixels are referred to generate (*b'*) in left partition. But if the VSPI doesn't read (*c*), right end pixel (*b*) is copied inside VSPI. And (*b*) is referred instead of (*c*) to generate (*b'*). And (*b'*) is recognized as discontinuous line. So reading overlapped pixels and clipping discontinuous line are needed.

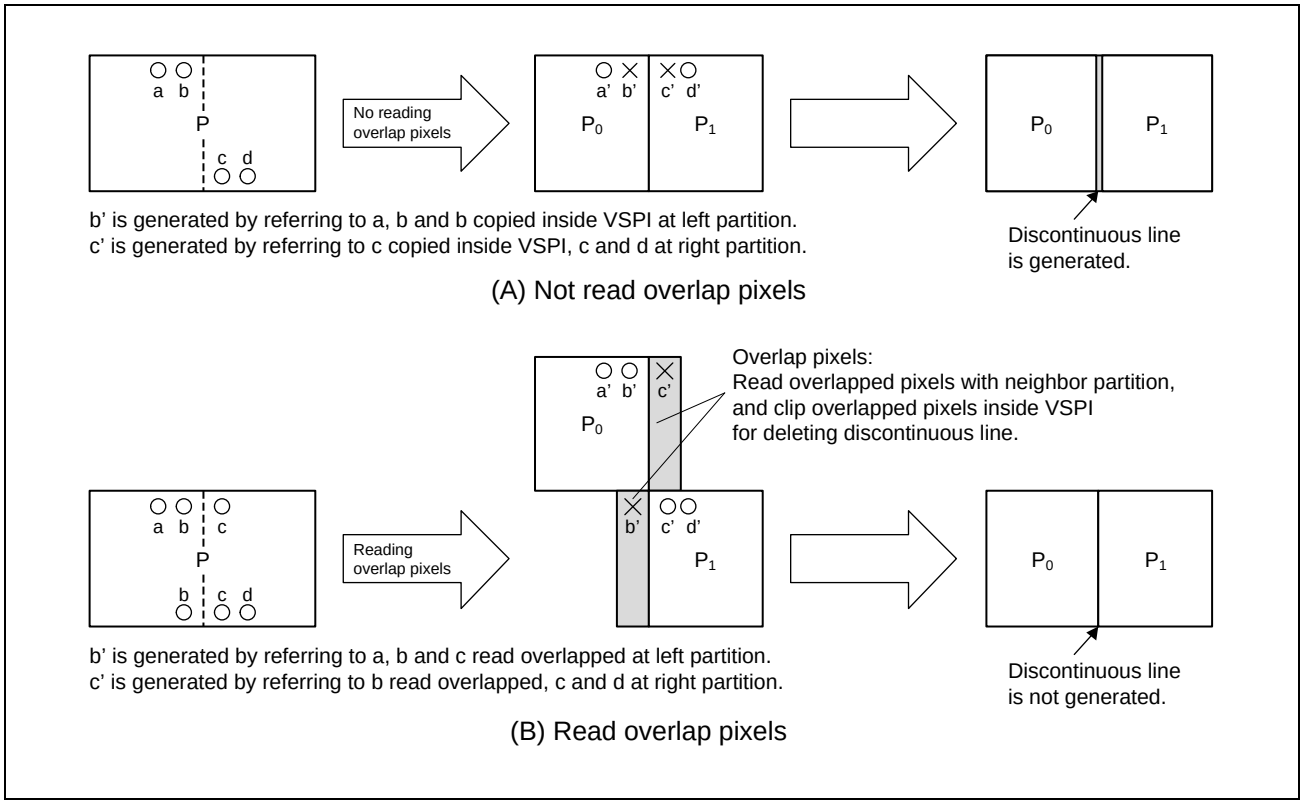


Figure 9.9-57 Discontinuous Line at Partition Boundary (The Example is 3 Tap Filter)

(5) The number of taps for each function's filter in horizontal direction

SRU, SHP, UDS and RPF have filter function which refer to not only current position's pixel but also several neighbor pixels to generate current position's output pixel. The number of taps for each function's filter are shown in this section to realize how many overlapped pixels to be read and how many pixels to be clipped.

(a) RPF (YCbCr42x -> YCbCr444 up sampling for Chroma)

Treatment of Cb and Cr is same as up sampling method. So C is used as Cb or Cr in the explanation.

(a-1) In case of VI6_RPF0_INFMT.CIPM = 0 for YCbCr42x input format

- Duplicate C component twice in horizontal direction.
- Horizontal size setting is restricted as 2-pixels units as shown in **Table 9.9-50**.

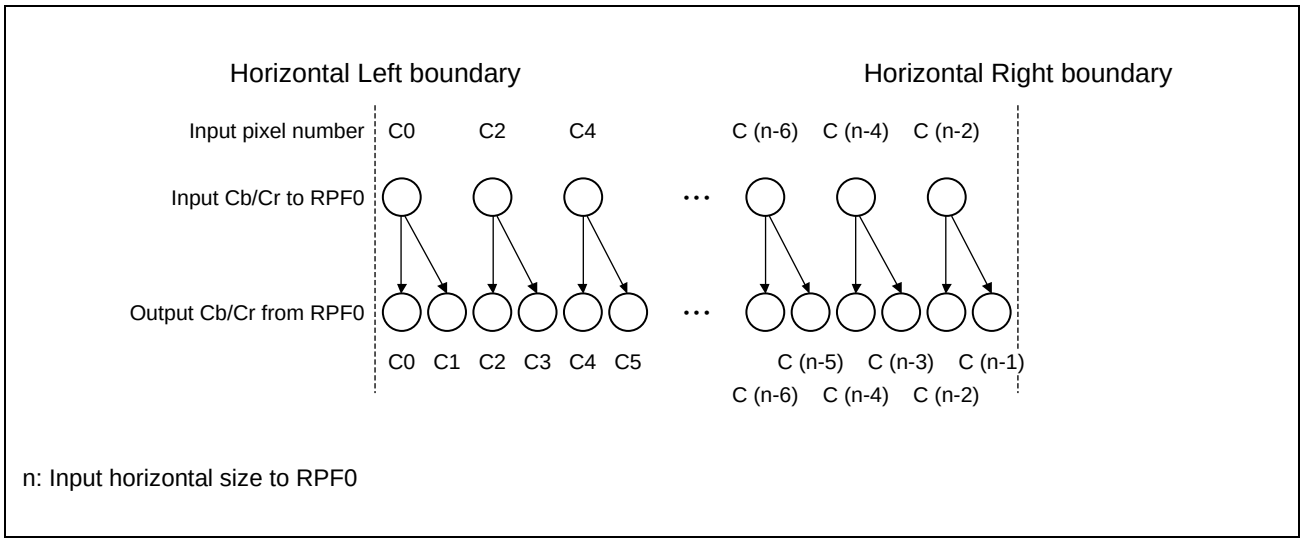


Figure 9.9-58 Chroma (Cb, Cr) Interpolation Way in RPF0 (VI6_RPF0_INFMT.CIPM = 0)

(a-2) In case of VI6_RPF0_INFMT.CIPM = 1 for YCbCr42x input format

- Even (0, 2, 4 …) output pixel of C is generated as copy of input pixel.
- Odd (1, 3, 5 …) output pixel of C is generated as 2 taps filter.
- Output pixel at right boundary is influenced by copied pixel C (n-2) of input pixel as shown in **Figure 9.9-59**. So it needs to clip C (n-1) output pixel for deleting discontinuous line when horizontal partition's right boundary is not frame boundary.
- Horizontal size setting is restricted as 2-pixels units as shown in **Table 9.9-50**.

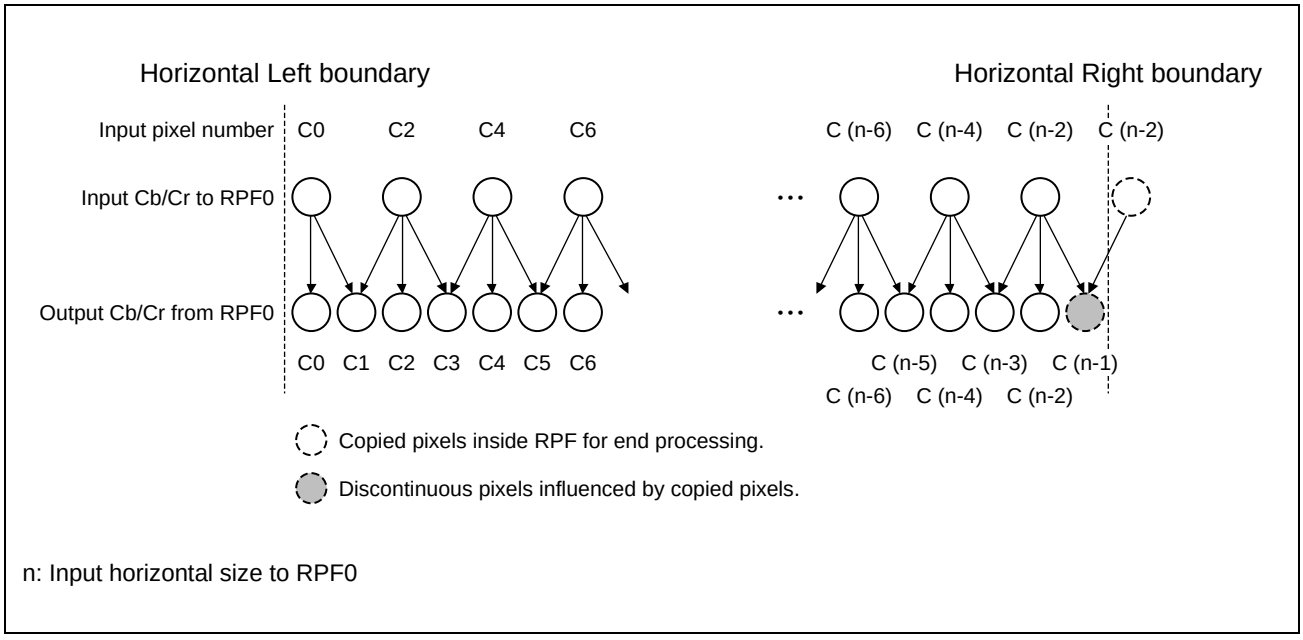


Figure 9.9-59 Chroma (Cb, Cr) Interpolation Way in RPF0 (VI6_RPF0_INFMT.CIPM = 1)

(b) SHP

- 9 taps filter
- From n-4 to n-1 output pixels are influenced by copied pixel of n-1 input pixel as shown in **Figure 9.9-60**. So it needs to clip from n-4 to n-1 output pixel for deleting discontinuous line when horizontal partition's right boundary is not frame boundary.
- From 0 to 3 output pixels are influenced by copied pixel of 0 input pixel as shown in **Figure 9.9-60**. So it needs to clip from 0 to 3 output pixels for deleting discontinuous line when horizontal partition's left boundary is not frame boundary.

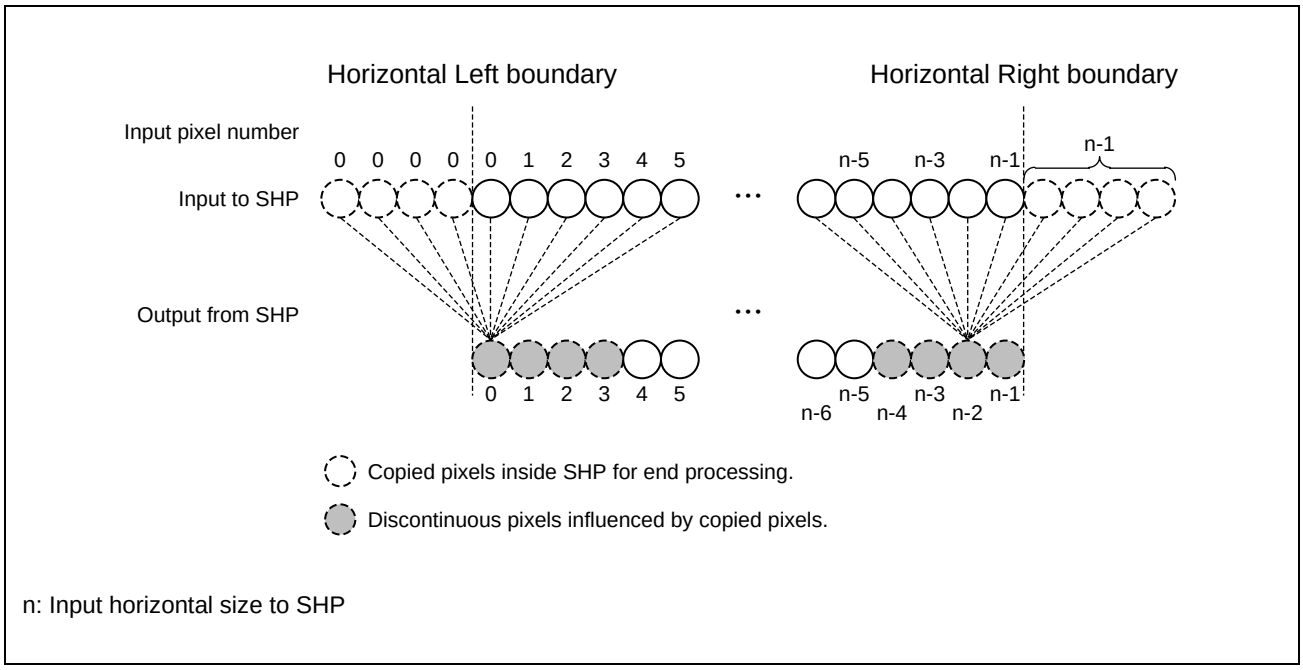


Figure 9.9-60 The Way of Applying Tap in SHP Filter

(c) SRU

SRU filter processing is composed of pre filter and post filter as shown in **Figure 9.9-61**.

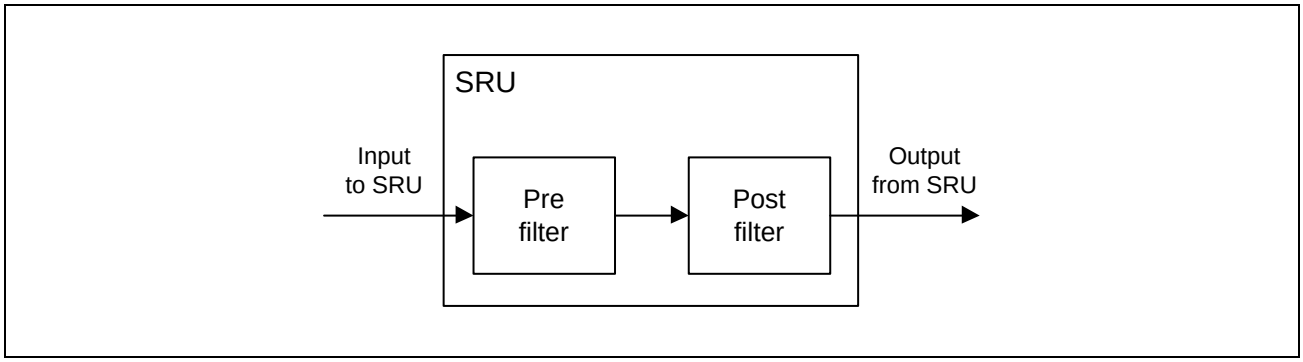


Figure 9.9-61 Structure of SRU Filter Processing

(c-1) In case of super resolution without scaling (VI6_SRU_CTRL0.SRU_MODE= 000b)

- Pre filter is 3 taps filter and go through post filter
- N-1 output pixel is influenced by copied pixel of n-1 input pixel as shown in **Figure 9.9-62**. So it needs to clip from n-1 output pixel for deleting discontinuous line when horizontal partition's right boundary is not frame boundary.
- 0 output pixel is influenced by copied pixel of 0 input pixel as shown in **Figure 9.9-62**. So it needs to clip 0 output pixel for deleting discontinuous line when horizontal partition's left boundary is not frame boundary.

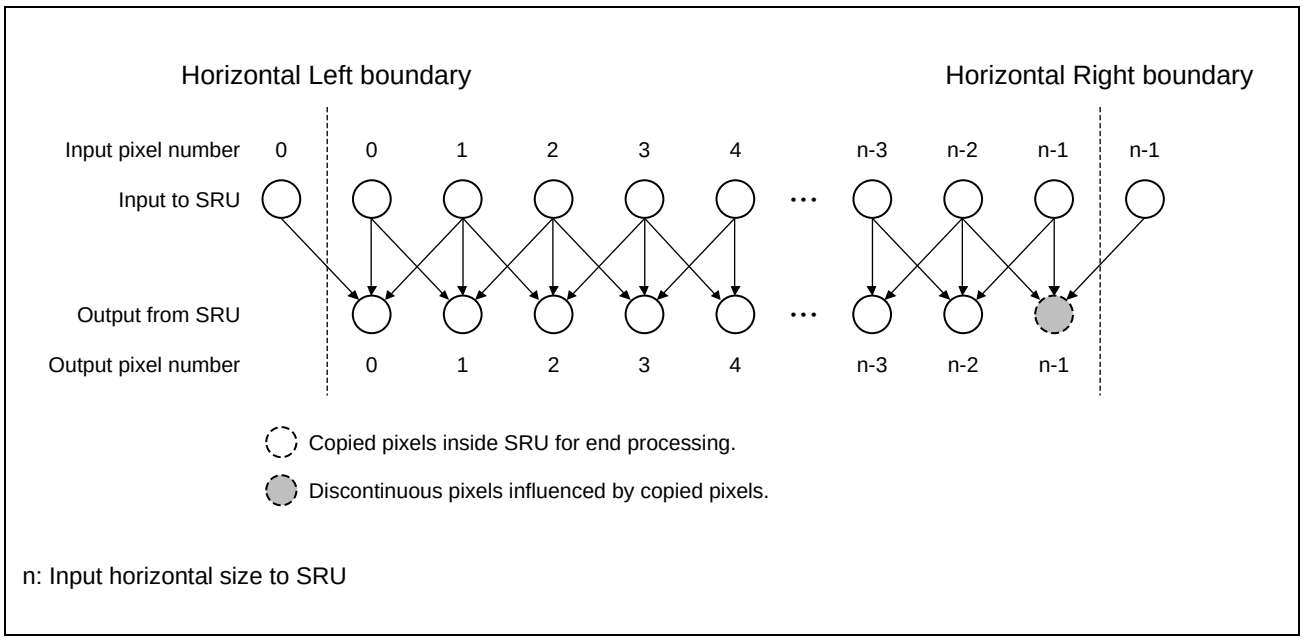


Figure 9.9-62 The Way of Applying Tap in SRU Filter (VI6_SRU_CTRL0.SRU_MODE = 000b)

(c-2) In case of super resolution with double scale-up (VI6_SRU_CTRL0.SRU_MODE = 100b)

- Pre filter is 3 taps filter and post filter is 2 taps filter.
- Even (0, 2, 4 …) output pixel is generated by 3 pixels (3 taps).
- Odd (1, 3, 5 …) output pixel is generated by 4 pixels (4 taps).
- From m-3 to m-1 output pixels are influenced by copied pixel of n-1 input pixel as shown in **Figure 9.9-63**. So it needs to clip from m-3 to m-1 output pixels for deleting discontinuous line when horizontal partition's right boundary is not frame boundary.
- From 0 to 1 output pixels are influenced by copied pixel of 0 input pixel as shown in **Figure 9.9-63**. So it needs to clip from 0 to 1 output pixels for deleting discontinuous line when horizontal partition's left boundary is not frame boundary.

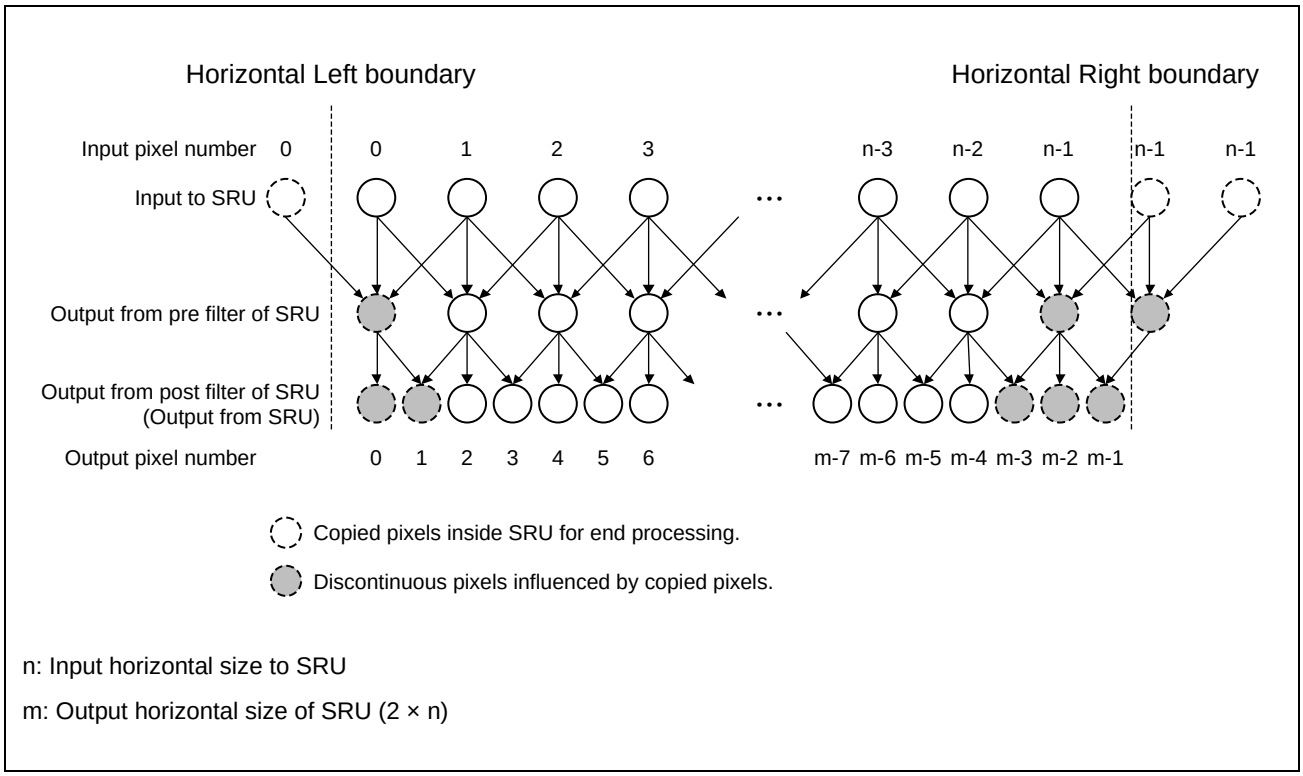


Figure 9.9-63 The Way of Applying Tap in SRU Filter (VI6_SRU_CTRL0.SRU_MODE = 100b)

(d) UDS

This section shows the way to calculate necessary input pixels of UDS (from src_pos0 position to src_pos1 position shown in **Figure 9.9-64**), discontinuous pixels generated by UDS (left position of discontinuous pixels is dst_pos0_pb as shown in **Figure 9.9-64**), necessary setting of horizontal start phase (hstp) and necessary setting of horizontal end phase (hedp) for UDS in order to generate valid output pixels from dst_pos0 position to dst_pos1 position as shown in **Figure 9.9-64**. Each parameter's definition is shown in (d-1) and (d-2) below. The way to calculate each parameter is shown in (d-3) below.

(d-1) Input parameters decided by user

- dst_pos0: Left position of desired valid pixels output from UDS.
- dst_pos1: Right position of desired valid pixels output from UDS.

(d-2) Parameters calculated in this section

- dst_pos0_pb: Left position of discontinuous pixels output from UDS.
- src_pos0: Left position of input pixels to UDS to generate valid pixels from dst_pos0 position.
- src_pos1: Right position of input pixels to UDS to generate valid pixels to dst_pos1 position.
- hstp: Horizontal start phase corresponding to dst_pos0_pb position.
- hedp: Horizontal end phase.

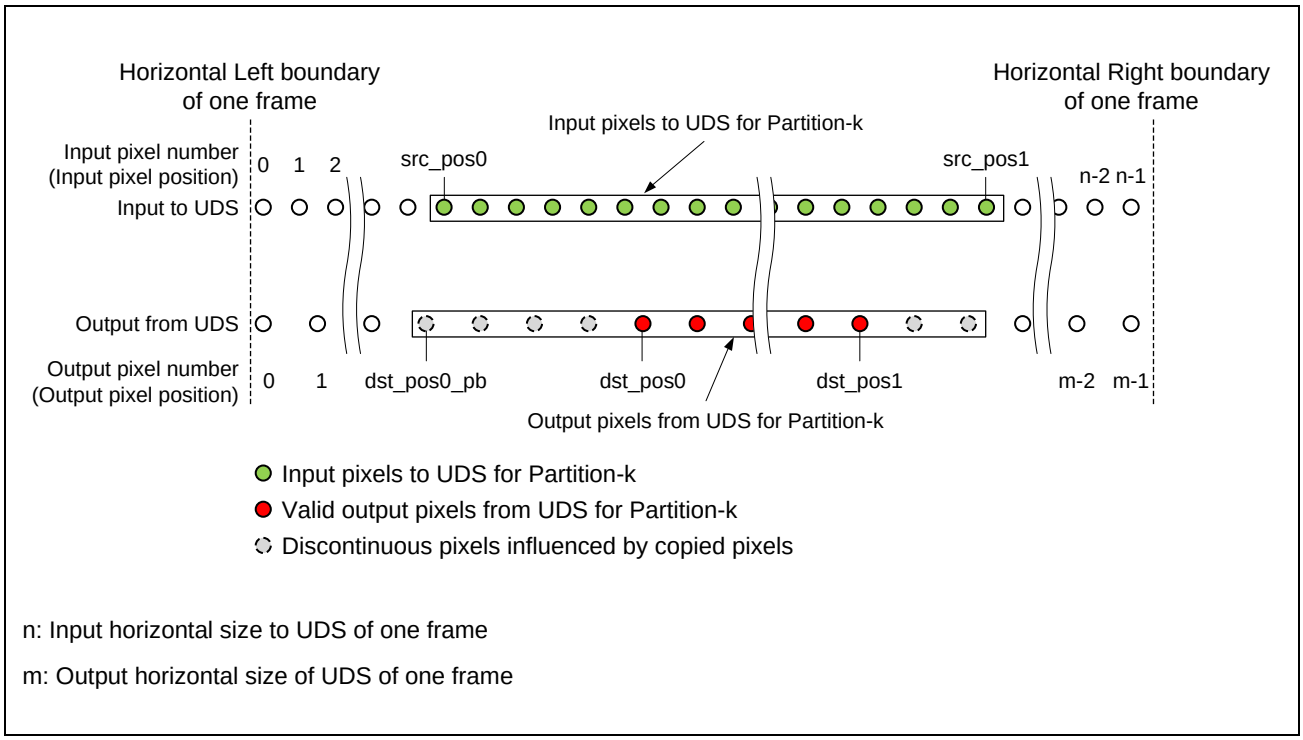


Figure 9.9-64 Relation between Valid Output Pixels' Position and Necessary Input Pixels' Position of UDS

(d-3) The way to calculate each parameter

[PROCEDURE 1] Calculate temporary src_pos0 position

For 1st partition (left end partition of source image)

$$src_pos0\ (temporal) = 0$$

For from 2nd partition to last partition (except left end partition of source image)

$$src_pos0\ (temporal) = \left(rdown \frac{dst_pos0 \times alpha - phase_edge \times {m_h}'}{4096 \times {m_h}'} - sub_src \right) \times {m_h}'$$

mh is the value of VI6_UDS_SCALE.HMANT, and f_h is the value of VI6_UDS_SCALE.HFRAC. mh' is defined in **Table 9.9-28**. phase_edge is defined in **Table 9.9-47**. rdown is defined in **9.9.1.5.6 Legend of Terms**. sub_src is 1 in case of mh' = 1, sub_src is 2 in case of mh' = 2 or 4. alpha can be expressed as below.

$$alpha = 4096 \times {m_h}' + f_h$$

Table 9.9-47 Value of phase_edge referenced in above calculation

VI6_UDS_SCALE.HMANT	VI6_UDS_CTRL.AMD	phase_edge
0	0	0
	1	rdown ((4096-VI6_UDS_SCALE.HFRAC)/2)
1 to 15	0 to 1	0

[PROCEDURE 2] Calculate temporary dst_pos0_pb position

For 1st partition (left end partition of source image)

$$dst_pos0_pb\ (temporal) = 0$$

For from 2nd partition to last partition (except left end partition of source image)

$$dst_pos0_pb\ (temporal) = rdown\frac{src_pos0\ (temporal) \times 4096 + \ phase_edge \times m_h'}{alpha}$$

[PROCEDURE 3] Decide dst_pos0_pb position from temporal dst_pos0_pb

There is a restriction about the left position which UDS outputs including discontinuous pixels. So it's necessary to shift the position of dst_pos0_pb (temporal) to the left so that the restrictions may be satisfied. dst_pos0_pb can be expressed as below.

$$dst_pos0_pb = dst_pos0_pb\ (temporal) - dst_pos0_pb_shift$$

Here, dst_pos0_pb_shift means the number of pixels by which dst_pos0_pb (temporal) is shifted to the left. Decide dst_pos0_pb_shift as satisfying restriction of [$(dst_pos0_pb \times alpha)$ should be multiple of mh']. The restriction can be expressed as below in other word.

There is no restriction in case of $mh' = 1$

- $(dst_pos0_pb \times alpha)$ must be multiple of 2 in case of $mh' = 2$
- $(dst_pos0_pb \times alpha)$ must be multiple of 4 in case of $mh' = 4$

Because clip register of UDS output (VI6_UDS_CLIP_SIZE) doesn't have clip offset setting, set the number of pixels of sum of valid output pixels and left discontinuous pixels to VI6_UDS_CLIP_SIZE.CL_HSIZE as shown in **Figure 9.9-65**. The number of sum of valid output pixels and left discontinuous pixels is $(dst_pos1 - dst_pos0_pb + 1)$ in the case. Left discontinuous pixels output from UDS should be clipped by offset register of WPF input clip (VI6_WPF0_HSZCLIP.HCL_OFST) as shown in **Figure 9.9-65**. The number of left discontinuous pixels is $(dst_pos0 - dst_pos0_pb)$ in the case. When modules which output discontinuous pixels such as SRU and/or SHP are connected after UDS as shown in **Figure 9.9-66**, discontinuous pixels are generated by also those modules. Add the number of pixels to the offset setting in the case.

The number of valid pixels from UDS should be set in size setting of WPF input clip register (VI6_WPF0_HSZCLIP.HCL_SIZE) as shown in **Figure 9.9-65**. The number of valid pixels is $(dst_pos1 - dst_pos0 + 1)$ in the case. When modules which output discontinuous pixels such as SRU and/or SHP is connected after UDS as shown in **Figure 9.9-66**, subtract the number of the discontinuous pixels from the size setting.

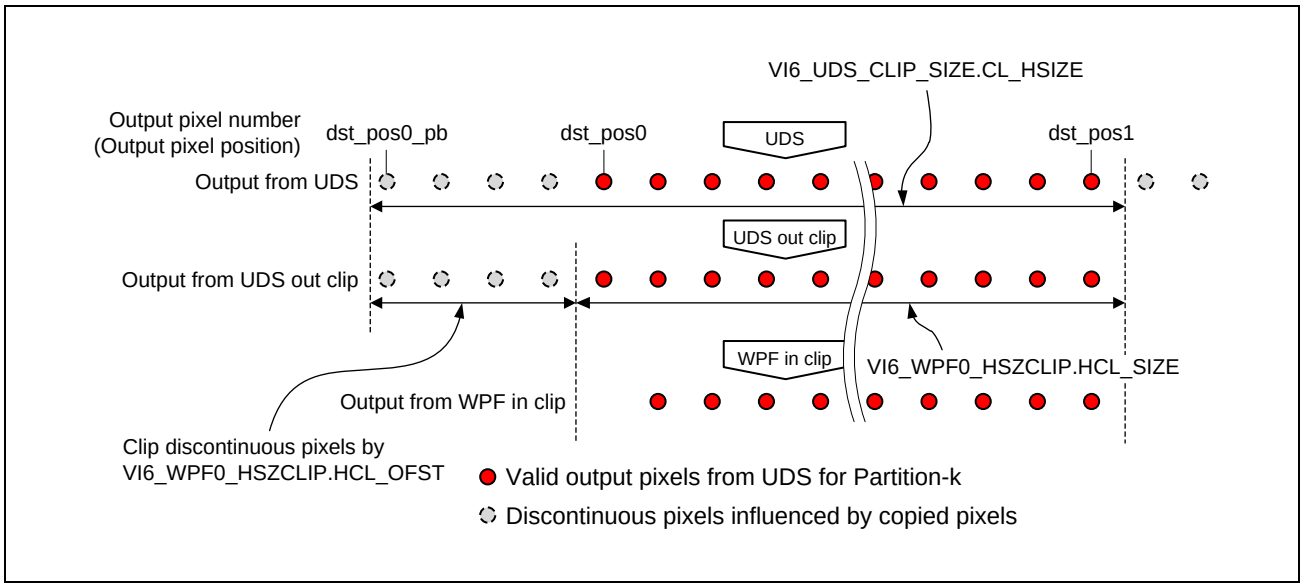


Figure 9.9-65 UDS Output Pixels and Clip Register's Setting (Case 1)

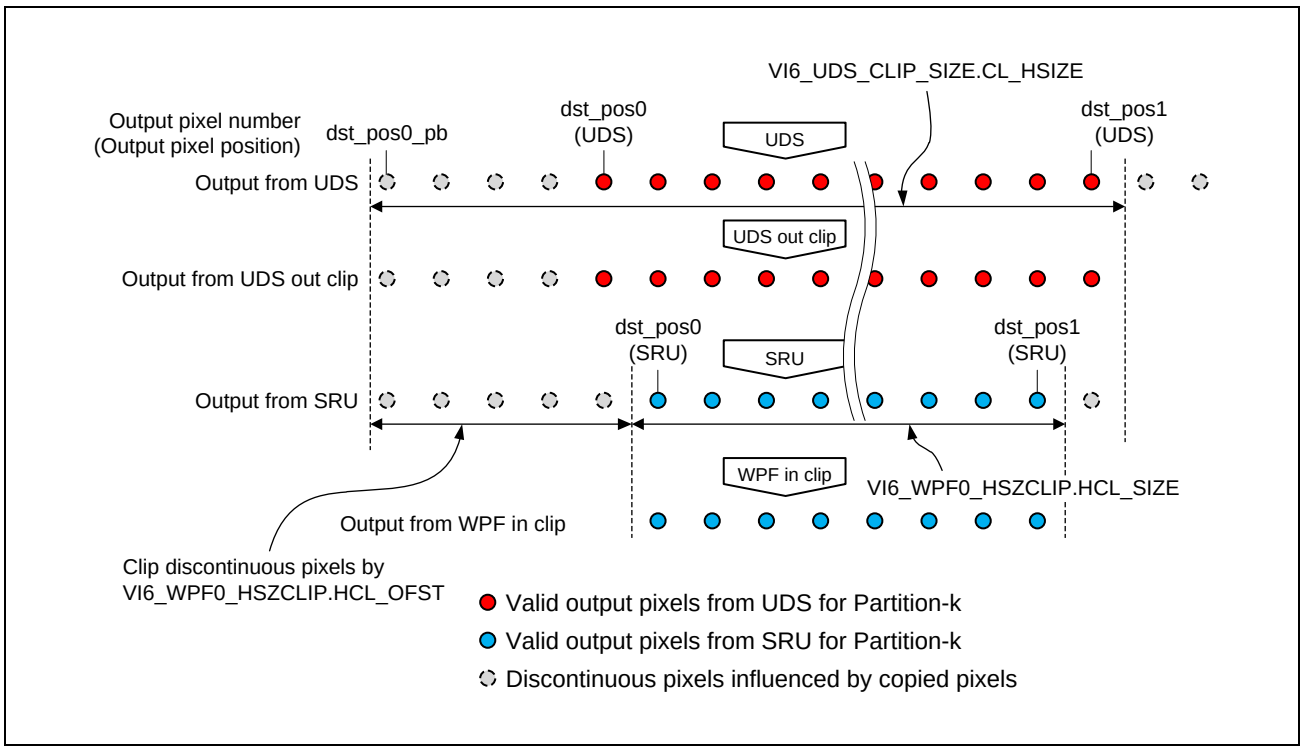


Figure 9.9-66 UDS Output Pixels and Clip Register'S Setting (Case 2)

[PROCEDURE 4] Calculate src_pos0 position from dst_pos0_pb

For 1st partition (left end partition of source image)

$$src_pos0 = 0$$

For from 2nd partition to last partition (except left end partition of source image)

$$src_pos0 = rdown\left(\frac{dst_pos0_pb \times alpha - phase_edge \times {m_h}'}{4096 \times {m_h}'} + add_src\right) \times {m_h}'$$

add_src is 0 in case that phase_residual expressed below is 0, add_src is 1 in case that phase_residual expressed below is not 0. phase_residual can be expressed as below.

$$phase_residual = rdown\left(\frac{alpha \times -dst_pos0_pb - phase_edge \times {m_h}'}{4096 \times {m_h}'}\right)$$

[PROCEDURE 5] Calculate src_pos1 position from dst_pos1

For from 1st partition to 2nd last partition (except right end partition of source image)

$$src_pos1 = rdown\left(\frac{dst_pos1 \times alpha - phase_edge \times {m_h}'}{4096 \times {m_h}'} + 2\right) \times {m_h}' + rdown\left(\frac{{m_h}'}{2}\right)$$

For last partition (right end partition of source image)

$$src_pos1 = n - 1$$

n is horizontal input size of UDS of one frame as shown in **Figure 9.9-64**.

[PROCEDURE 6] Calculate hstp

For 1st partition (left end partition of source image)

$$hstp = phase_edge$$

For from 2nd partition to last partition (except left end partition of source image)

$hstp = 4096 - rdown(phase_residual/mh')$ in case that phase_residual is not 0

$hstp = 0$ in case that phase_residual is 0

Set the value of hstp to VI6_UDS_HPHASE.HSTP. To enable VI6_UDS_HPHASE, set 1 to VI6_UDS_CTRL.AMDSLH.

[PROCEDURE 7] Calculate hedp

For from 1st partition to 2nd last partition (except right end partition of source image)

$$hedp = 0$$

For last partition (right end partition of source image)

$$hedp = phase_edge$$

Set the value of hedp to VI6_UDS_HPHASE.HEDP. To enable VI6_UDS_HPHASE, set 1 to VI6_UDS_CTRL.AMDSLH.

(6) Horizontal size clipping registers for image partition

As explained in **9.9.1.4.6 (1) Necessity of image partition** and **9.9.1.4.6 (4) Reading overlapped pixels and clipping discontinuous line**, discontinuous line is generated at partition boundary. So clipping discontinuous line is needed at boundary of image partition.

Horizontal input size clipping can be realized at input point of UDS module, input point of WPF0 module, input point of HGO module, input point of HGT module and output point of UDS module as shown in **Figure 9.9-67**. **Table 9.9-48** shows horizontal size clipping registers of each modules.

Other than an output point of UDS, arbitrary horizontal size can be specified from arbitrary horizontal offset position.

For an output point of UDS, arbitrary horizontal size can be specified from the left edge.

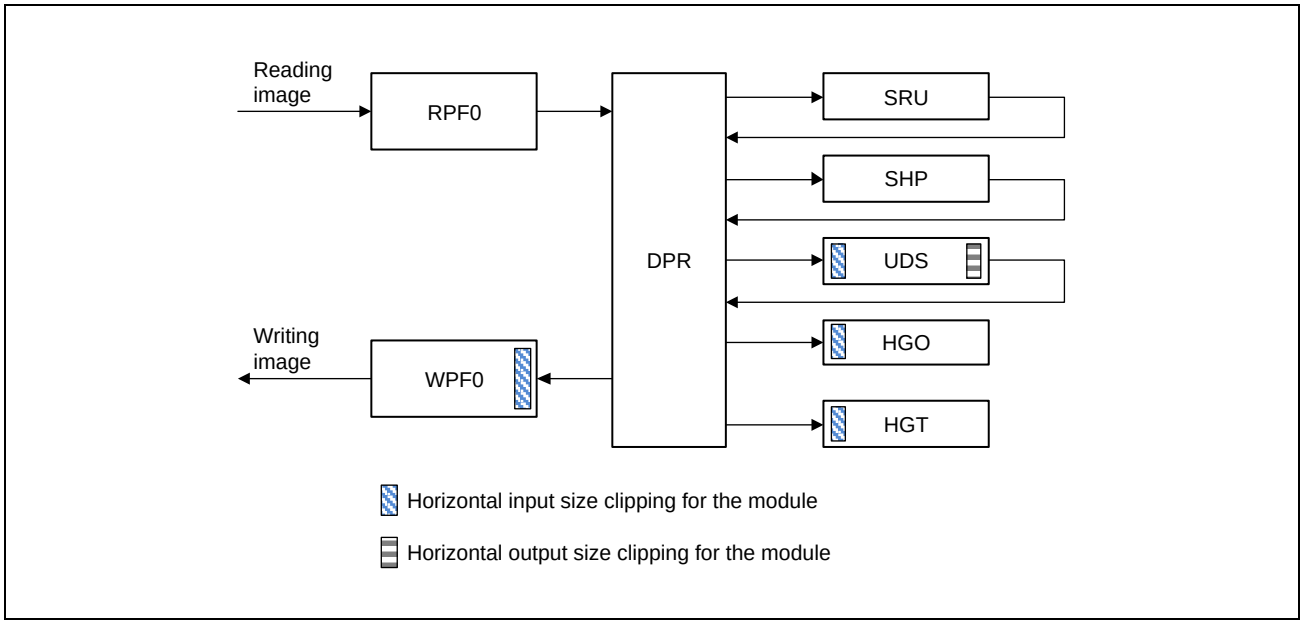


Figure 9.9-67 Horizontal Input Size Clipping in VSPI

Table 9.9-48 Horizontal Size Clipping Registers of Each Modules

Point	Size or Offset	Registers
UDS input*1	Horizontal offset	VI6_UDS_HSZCLIP.HCL_OFST [7:0]
	Horizontal size	VI6_UDS_HSZCLIP.HCL_SIZE [12:0]
UDS output*2	Horizontal size	VI6_UDS_CLIP_SIZE.CL_HSIZE [12:0]
WPF input*3	Horizontal offset	VI6_WPF0_HSZCLIP.HCL_OFST [7:0]
	Horizontal size	VI6_WPF0_HSZCLIP.HCL_SIZE [12:0]
HGO input	Horizontal offset	VI6_HGO_OFFSET.HOFFSET [13:0]
	Horizontal size	VI6_HGO_SIZE.HSIZE [13:0]
HGT input	Horizontal offset	VI6_HGT_OFFSET.HOFFSET [13:0]
	Horizontal size	VI6_HGT_SIZE.HSIZE [13:0]

Note 1. Set 1 to VI6_UDS_HSZCLIP.HCEN to enable clipping horizontal size at input to UDS.

Note 2. Horizontal offset at the UDS output point cannot be specified.

Note 3. Set 1 to VI6_WPF0_HSZCLIP.HCEN to enable clipping horizontal size at input to WPF0.

(7) Histogram of HGO/HGT

This section shows the setting value of registers about histogram detection of HGO and HGT. **Figure 9.9-68** shows the example of getting HGO histogram for UDS output. Those registers shown in **Figure 9.9-68** can be set by display list.

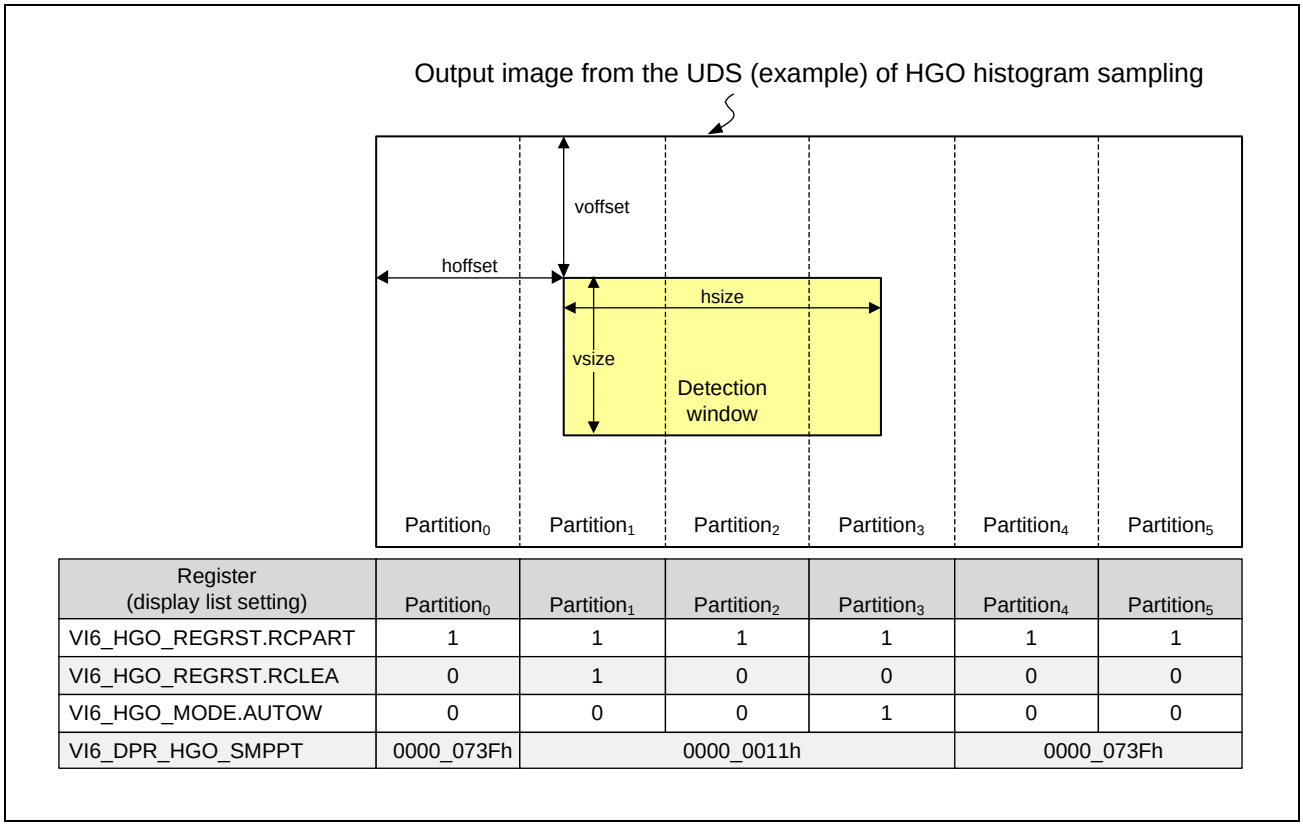
If user wants to get histogram for partial window (like **Figure 9.9-68**), set 0000_073Fh to VI6_DPR_HGO_SMPPT/VI6_DPR_HGT_SMPPT for partitions without detection window of histogram.

Set 1 to VI6_HGO_REGRST.RCLEA (VI6_HGT_REGRST.RCLEA) to initialize histogram counter for 1st partition including detection window of histogram (Partition1 in case of **Figure 9.9-68**). Then histogram counter is initialized before starting processing of the partition. To accumulate counting histogram for all partitions to get one frame's histogram, don't set 1 to VI6_HGO_REGRST.RCLEA (VI6_HGT_REGRST.RCLEA) after the partition.

Set 1 to VI6_HGO_REGRST.RCPART (VI6_HGT_REGRST.RCPART) for all partitions to initialize HGO (HGT) except histogram counter.

If user wants to dump histogram to external memory automatically set 1 to VI6_HGO_MODE.AUTOW (VI6_HGT_MODE.AUTOW) for last partition with detection window (partition3 in case of **Figure 9.9-68**). Then histogram is dumped to external memory after processing the partition with VI6_HGO_MODE.AUTOW =1 (VI6_HGT_MODE.AUTOW =1).

hsize and hoffset in **Figure 9.9-68** means horizontal size and horizontal offset as one frame. Set hsize and hoffset as each partition to VI6_HGO_SIZE.HSIZE, VI6_HGT_SIZE.HSIZE, VI6_HGO_OFFSET.HOFFSET and VI6_HGT_OFFSET.HOFFSET. If target node of histogram sampling generates discontinuous pixels, consider the number of discontinuous pixels and set offset register (VI6_HGO_OFFSET.HOFFSET, VI6_HGT_OFFSET.HOFFSET) as clipping also discontinuous pixels.



Register (display list setting)	$Partition_0$	$Partition_1$	$Partition_2$	$Partition_3$	$Partition_4$	$Partition_5$
VI6_HGO_REGRST.RCPART	1	1	1	1	1	1
VI6_HGO_REGRST.RCLEA	0	1	0	0	0	0
VI6_HGO_MODE.AUTOW	0	0	0	1	0	0
VI6_DPR_HGO_SMPPT	0000_073Fh	0000_0011h			0000_073Fh	

Figure 9.9-68 Example of Histogram Detection for Image Partition

(8) Restriction when VSPI image partition is used

When VSPI image partition is used, there are some restrictions below.

Remark SRU (×1) means SRU without scaling. SRU (×2) means SRU with double scale-up.

RPF

- Tiling (VI6_RPF0_SRC_BSIZE.BHSIZE != VI6_RPF0_SRC_ESIZE.EHSIZE) is prohibited.
- Reading 1bit/pixel alpha plane is prohibited.

SRU and UDS

- There are restriction when SRU and UDS are used in a VSPI execution. The restriction depends on SRU resolution mode, UDS horizontal scaling factor, the order of UDS and SRU and usage of SHP. The restriction is shown in **Table 9.9-49**.

  Example 1: RPF0 → UDS → LUT → SRU (×2) → WPF0 is prohibited.

  Example 2: RPF0 → SRU (×2) → LUT → UDS (hscale = 2) → WPF0 can be used.

  Example 3: RPF0 → SRU (×1) → LUT → UDS (hscale = 4) → WPF0 can be used.

  Example 4: RPF0 → SRU (×1) → UDS (hscale = 1/2) → WPF0 is prohibited.

Table 9.9-49 Restriction in Case of Connecting SRU and UDS

Order	SRU	UDS*1	VSPI supports	Note
UDS → SRU (×1)	×1	1/16 < hscale ≤ 16	✓	—
SRU (×1) → UDS	×1	hscale < 1	N/A	—
		1 ≤ hscale	✓	—
UDS → SRU (×2)	×2	1/16 < hscale ≤ 16	N/A	—
SRU (×2) → UDS	×2	hscale ≤ 1/4	N/A	—
		1/4 < hscale	✓	SHP cannot be used in this case

HGO/HGT

- HGO/HGT sampling points should not be specified at modules before UDS or SRU (×2).

  Example: In case of RPF0 → SHP → UDS → SRU (×1) → CLU → LUT → WPF0 connection
  UDS, SRU, CLU or LUT can be specified as HGO/HGT sampling points
  RPF0 or SHP cannot be specified as HGO/HGT sampling points.

  Example: In case of RPF0 → SHP → SRU (×2) → CLU → LUT → WPF0 connection
  SRU, CLU or LUT can be specified as HGO/HGT sampling points
  RPF0 or SHP cannot be specified as HGO/HGT sampling points.

- Letter box detection (VI6_HGO_m_LB_DET: m = R, G, B and VI6_HGT_LB_DET) cannot be available.
- Horizontal Pixel Skipping Mode cannot be used. Fix 0 to VI6_HGO_MODE.HRATIO [1:0], Fix 0 to VI6_HGT_MODE.HRATIO [1:0].

WPF

- Mode B dither cannot be used.

9.9.1.4.7 The step to read HGO histogram registers of 256 step mode by indirect addressing

When HGO is used as 256 step mode (VI6_HGO_MODE.STEP = 1) and HGO histogram is not automatically dumped to external memory (VI6_HGO_MODE.AUTOW = 0), read HGO histogram registers by indirect addressing. The step is shown below.

1. Set value-m in VI6_HGO_EXT_HIST_ADDR (**9.9.1.3.15 (11) HGO Y/MaxRGB Extended Histogram Address Register (VI6_HGO_EXT_HIST_ADDR)**). m is frequency of Y or MaxRGB pixel with value-m.
2. Read VI6_HGO_EXT_HIST_DATA (**9.9.1.3.15 (12) HGO Y/MaxRGB Extended Histogram Data Register (VI6_HGO_EXT_HIST_DATA)**) to get frequency of Y or MaxRGB pixel with value-m.

Repeat step 1 and step 2 256 times from m = 0 to m = 255.

9.9.1.5 Usage Notes

9.9.1.5.1 Assignment in Memory Space

Make sure that VSPI register space shall be mapped to Non-Cache region.

9.9.1.5.2 Input Image Size

Table 9.9-50 is a list of input size specifications.

Table 9.9-50 List of Input Size Specifications

Module	Min. Input Size (horizontal × vertical)	Max. Input Size (horizontal × vertical)	Restrictions on Setting Unit
RPF	1 × 1 pixel	8190 × 8190 pixels	• YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. • YCbCr420: 2-pixel units both horizontally and vertically. • Other formats: 1-pixel units both horizontally and vertically. *Note*: When the 1-bpp alpha plane*1 is input, the size can always be specified in 8-pixel units both horizontally and vertically regardless of the input format. *Note*: These restrictions including note 1 are applied to the following. VI6_RPF0_SRC_BSIZE VI6_RPF0_SRC_ESIZE
SRU*2	4 × 4 pixels	• Without scaling (same size): 288 × 8190 pixels • With double scale-up: 288 × 4095 pixels	1-pixel units both horizontally and vertically.
UDS*2	4 × 4 pixels	304*3 × 8190 pixels	1-pixel units both horizontally and vertically.
SHP	4 × 4 pixels	8190 × 8190 pixels	1-pixel units both horizontally and vertically.
LUT	1 × 1 pixel	8190 × 8190 pixels	1-pixel units both horizontally and vertically.
CLU	1 × 1 pixel	8190 × 8190 pixels	1-pixel units both horizontally and vertically.
HST	1 × 1 pixel	8190 × 8190 pixels	1-pixel units both horizontally and vertically.
HSI	1 × 1 pixel	8190 × 8190 pixels	1-pixel units both horizontally and vertically.
HGO	1 × 1 pixel	8190 × 8190 pixels	1-pixel units both horizontally and vertically.
HGT	1 × 1 pixel	8190 × 8190 pixels	1-pixel units both horizontally and vertically.
WPF*2	1 × 1 pixel	• When VI6_WPF0_OUTFMT.ROT is smaller than 2, 8190 × 8190 pixels • When VI6_WPF0_OUTFMT.ROT is bigger than 1, 256 × 8190 pixels	• YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. • YCbCr420: 2-pixel units both horizontally and vertically. • Other formats: 1-pixel units both horizontally and vertically. *Note*: This restriction on the WPF only applies to the WPF output size. The WPF input size should be specified in 1-pixel units. *Note*: In case of image rotation and flipping, other restrictions are applied. Refer to **9.9.1.3.7 (7) WPF0 Rotation Control Registers (VI6_WPF0_ROT_CTRL)**, which explains LMEM_WD.

Note 1. When VI6_RPF0_ALPH_SEL.ASEL is set to 011b.

Note 2. To support horizontal input size is larger than 256 pixels about SRU, UDS and/or WPF/ROT (VI6_WPF0_OUTFMT.ROT>1), the way of image partition is needed. Refer to **9.9.1.4.6 Image partition for VSPI processing** for realizing the way of image partition.

Note 3. The restriction of horizontal size is applied to input size of UDS in case of up-scaling in horizontal direction. The restriction of horizontal size is applied to output size of UDS in case of down-scaling or no-scaling in horizontal direction.

The most important restriction shown in **Table 9.9-50** is the setting unit. Make appropriate register settings so that the size of the image input to each module does comply with setting units and does not exceed the limits shown in **Table 9.9-50**. Especially for the register settings of SRU and UDS, where the size can be changed within the modules, be careful not to exceed the input size limit for the module connected behind each of them.

9.9.1.5.3 Output Image Size

The size of the output from each WPF is determined by the results of processing in the modules connected with the DPR. As shown in **Figure 9.9-2**, the data input to the VSPI is sent to the WPF output modules through the RPF and the modules connected with the DPR. When there is no processing that changes the image size through this data path, the WPF output size is the same as the RPF input size. **Table 9.9-51** is a list of the processing that changes image size.

Table 9.9-51 Image Processing that Changes Image Size

Module	Function*1	Related Register	Size of Output from Module
SRU	Super resolution with double scale-up	VI6_SRU_CTRL0	• Super resolution without scaling (same size): Input size • Super resolution with double scale-up: Input size × 2
UDS	Output size clipping	VI6_UDS_CLIP_SIZE	• Horizontal output size: VI6_UDS_CLIP_SIZE.CL_HSIZE setting • Vertical output size: VI6_UDS_CLIP_SIZE.CL_VSIZE setting
WPF	Input size clipping	VI6_WPF0_HSZCLIP VI6_WPF0_VSZCLIP	When this function is disabled, the input size and the output size are the same. When this function is enabled, the output is in the following size. • Horizontal output size: VI6_WPF0_HSZCLIP.HCL_SIZE*2 setting • Vertical output size: VI6_WPF0_VSZCLIP.VCL_SIZE*2 setting

Note 1. For details of each function, refer to the descriptions of the related registers.

Note 2. Refer to **9.9.1.3.1 Notational Conventions for Registers and Bit Fields** for explanation of register bit field.

The input image size can be changed only with the modules and functions shown in **Table 9.9-51**. With the other modules and functions, the input size and output size are the same. Accordingly, after module connections with the DPR are determined, the VSPI output size can also be determined through the following steps.

1. The image size (VI6_RPF0_SRC_ESIZE) read from the external memory and sent to the DPR by the RPF0 is the initial value.
2. When the module connected with the DPR is not a module (function) shown in **Table 9.9-51**, the output size from the module is the same as the input size; the image size does not need to be updated.
3. When a module connected with the DPR is a module (function) shown in **Table 9.9-51**, the output image size should be updated to the size shown in the table, which should be used as the input image size for the module connected behind it.
4. When the final image size at the WPF0 is determined, this size is the VSPI output size for that WPF0 path.

Figure 9.9-69 shows how to determine the image size in a sample DPR connection through the above steps. The related conditions that determine the image size are also shown.

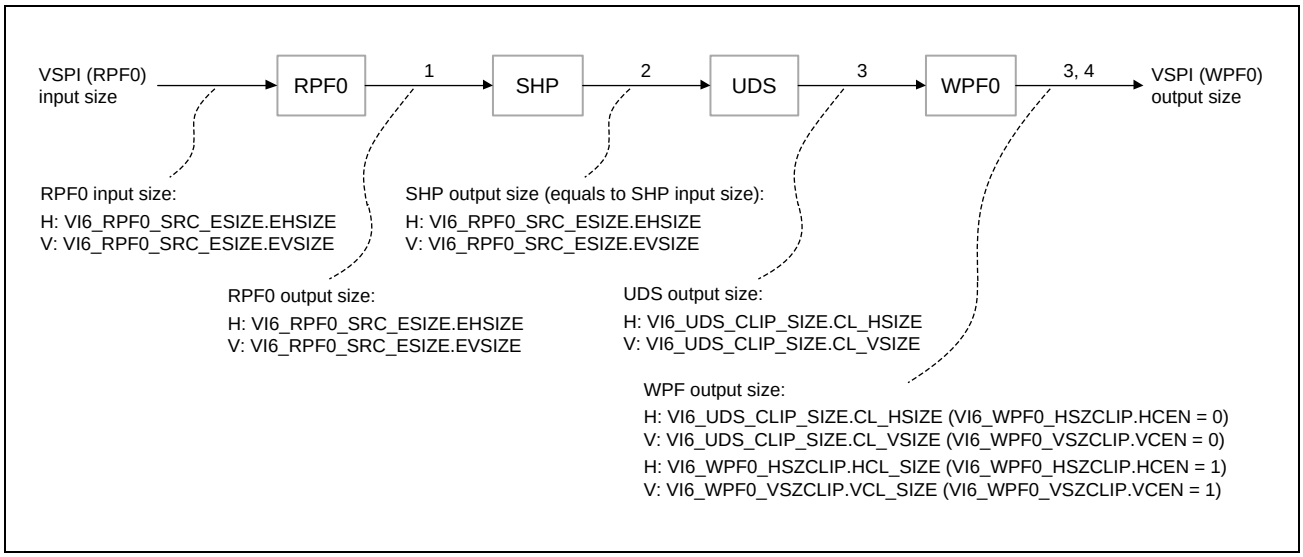


Figure 9.9-69 Input/Output Size for Each Module in a Sample DPR Connection

Make appropriate register settings in each module so that the VSPI output image size determined as shown in the figure does not violate the restrictions shown in **Table 9.9-51**.

9.9.1.5.4 Configuration of VSPI

Table 9.9-52 show the configuration of each type of VSPI unit on this LSI.

All the registers and functions listed in [Module] category of **Table 9.9-52** should be treated as un-implemented functions in each VSPI unit. Do not set any registers related to the un-implemented function without any statement.

Table 9.9-52 VSPI Configuration

Category	Configuration	Restricted contents
Restriction	WPF/format	See **9.9.2 FCPVI** for more detail. Related registers: VI6_WPF0_OUTFMT.ROT, VI6_WPF0_OUTFMT.WRFMT
	WPF/Addr, Stride	See **9.9.2 FCPVI** for more detail. Related registers: VI6_WPF0_STRIDE_Y. VI6_WPF0_STRIDE_C VI6_WPF0_DSTM_ADDR_Y VI6_WPF0_DSTM_ADDR_C0 VI6_WPF0_DSTM_ADDR_C1

9.9.1.5.5 Performance of VSPI

VSPI performance is shown in this section. VSPI performance shown in below is based on evaluation under the condition that only one IP is operated.

(1) Performance of VSPI

500 Mpix/s process rate per 1 set of VSPI except following cases. When one of following conditions is true, performance of VSPI is lower than 500 Mpix/s process rate. The execution of the VSPI is up to at most 480 frames per second.

(1) 90 degree or 270 degree rotation (VI6_WPF0_OUTFMT.ROT > 3) is enabled

(2) Horizontal scaling ratio is larger than ×4 times and SRU (×1) is used at same time.

(3) At least one RPF has all following conditions.

- RGB32bpp or YCbCr planar or YCbCr444 semi-planar
- Reading alpha plane

9.9.1.5.6 Legend of Terms

This section describes the configurations of VSPI on this LSI. Descriptions in this section show the full-function specifications of the VSPI IP. Some of the functions described in this section are not available on VSPI with restrictions written here.

(1) Terminology

This section explains the terminology for VSPI.

- Alpha
  The VSPI can execute both color data and its transparent data. The alpha is 8-bit data and expresses the transparency of pixel. The value FFh represents opaque pixel, and the value 00h represents transparent pixel.
- Plane
  One layer consists of color components such as R/G/B or Y/Cb/Cr, and alpha data representing its transparency of the pixel. The frame which is one of color components and alpha data is called as a plane.
- Picture Plane
  The picture plane represents color data in one layer.
- Alpha Plane
  The alpha plane represents transparent data in one layer.

(2) Definition of operators and functions

The following operators, notations are defined for explaining the functions of VSPI.

- $\langle x \rangle$
  Discard decimal places of value x
- $rdown\ (x)$
  Discard decimal places of value x: Ex, $rdown(3.81) = 3$
- $clip0\ (x)$
  Clip to 0 value if x is less than 0. $[x = (x < 0)?\ 0 : x]$

- $clip3\ (min, max, x)$
  The value x shall be clipped so that x shall be min < x < max. $\left[x = (x < min)?\ min: \left((x > max)?\ max: x\right)\right]$

(3) Unit

The unit is defined here for explaining VSPI's functions.

- [bpp]: bits per pixel

VSPI can handle a number of image formats. Each image format has different number of data bits. The unit [bpp] shows a number of data bits for one pixel. For example, RGB 8bpp means that its format is RGB and the pixel consists of 8 data bits.

9.9.2 FCPVI

The FCPVI conducts the bus accesses with 256-bit AXI bus for VSPI. This module is under control of software for the VSPI with the control function of the FCPVI executed on the host CPU.

9.9.2.1 Features

The FCPVI has the following features.

- Support out-of-order for the whole outstanding transactions

Table 9.9-53 Main Function of the FCPVI

Item	Description
Picture size supported	• Minimum: 1 pixel × 1 line • Maximum: 8190 pixels × 8190 lines The picture sizes must follow the restriction specified in **9.9 Video Signal Processor (VSP)** (sections “Usage Notes”, “Input Image Size” and “Output Image Size”).
Color format supported	Support all formats specified in **9.9 Video Signal Processor (VSP)** (sections “Register Descriptions”, “RPF0 Input Format Registers (VI6_RPF0_INFMT)” and “WPF0 Output Format Registers (VI6_WPF0_OUTFMT)”).

AXI transactions	• Read access:
	  • Access at 256 bytes boundary
	  • Support out-of-order for the whole outstanding transactions
	  • Support data interleaving
	• Write access:
	  • Access at 256 bytes boundary
	  • Support out-of-order for the whole outstanding transactions

9.9.2.2 Block Diagram

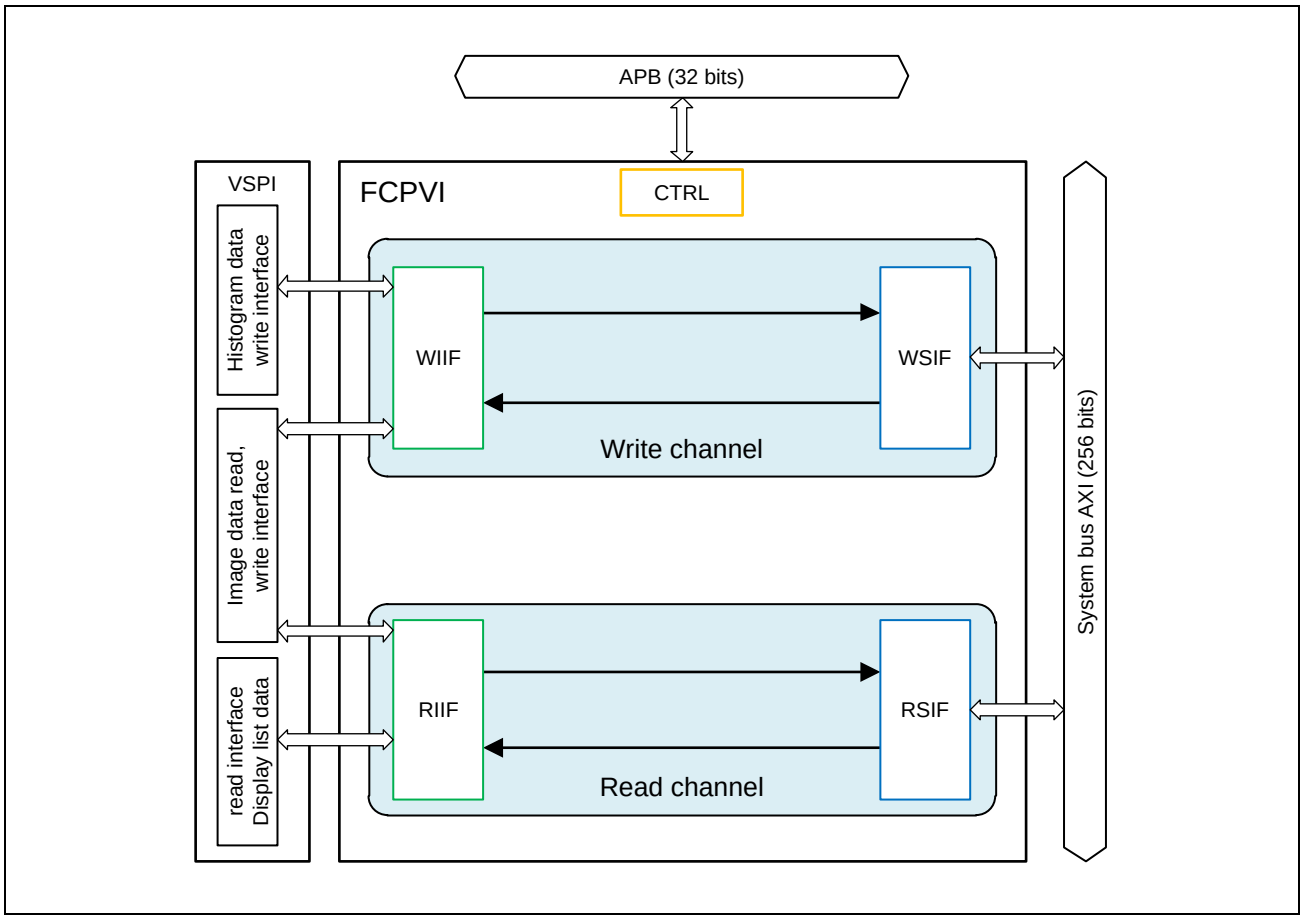


Figure 9.9-70 Overall Configuration of the FCPVI

Table 9.9-54 Sub-Module Description

Abbr.	Description
CTRL	FCPVI controller
WIIF	VSPI write channel interface
WSIF	System AXI write channel interface

Abbr.	Description
RIIF	VSPI read channel interface
RSIF	System AXI read channel interface

9.9.2.3 IP Set and Bus Access Functions

FCPVI does not work alone but in companion with other IPs (VSPI) in an IP set. For each IP set the bus access function conducted by FCPVI is different.

Table 9.9-55 Bus Access Function for Each IP Set

IP Set	Bus Access
VSPI + FCPVI	Following functions are conducted by FCPVI • Read linear addressing image data • Read display list data • Write image data • Write histogram data*1

Note 1. This function is active when FCPVI is connected to VSPI.

9.9.2.4 Register Configuration

The FCPVI contains the registers which can directly be read and written to by the host CPU. The host can access any long word (32 bits) in the registers. Therefore, read-modify-write is needed to change partial bits in 32bits register.

The grayed portions represent reserved areas for debugging or future functional expansion. Do not access the reserved areas. Never access the registers when the FCPVI is in the module standby mode (with the operation clock placed in hold status).

The following are the base addresses of the FCPVI channels.

<FCPVI0_base>: 00_1606_0000h (5606_0000h*1, 4606_0000h*2)

Note 1. CM33 address space (non-secure)
Note 2. CM33 address space (secure)

Table 9.9-56 List of FCPVI Register

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
FCP version control register	FCPV0_FCP_VCR	0000_0104h	0000h	32
FCPV configuration register 0	FCPV0_FCP_CFG0	0000_0000h	0004h	32
Reserve	-	-	0008h to 000Fh	-
FCP reset register	FCPV0_FCP_RST	0000_0000h	0010h	32
Reserve	-	-	0014h to 0017h	-
FCP status register	FCPV0_FCP_STA	0000_0000h	0018h	32

9.9.2.5 Register Description

(1) FCPVI Version Control Register (FCP_VCR)

FCP_VCR is the version control register of the FCPVI. The value read is fixed.

Access Size : 32 bits
Address : <FCPV0_base> + 0000h
Initial Value : 0000_0104h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATEGORY[7:0]								REVISION[7:0]							
Initial Value	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
15 to 8	CATEGORY [7:0]	1h	R	FCPVI use case category. 01h: This LSI
7 to 0	REVISION[7:0]	4h	R	Version of LSI product 04h: FCPVI revision is this LSI

(2) FCPVI Configuration Register 0 (FCP_CFG0)

FCP_CFG0 specifies the configuration of the FCPVI.

Access Size : 32 bits
Address : <FCPV0_base> + 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	FCPSEL	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
1	FCPSEL	0h	RW	This bit must be set to 1b in the FCPVI initialization
0	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.

(3) FCP Reset Register (FCP_RST)

FCP_RST controls the FCPVI reset operation. The reset operation of the FCPVI highly depends on the reset sequence of the connected VSPI. For detailed information, refer to **9.9.2.6 Operation**.

Access Size : 32 bits
Address : <FCPV0_base> + 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	MODRST	-	-	-	SOFTRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
4	MODRST	0h	RW	FCPV MODule ReSeT This reset stops the FCPVI module immediately regardless of the bus transaction state. To operate the FCPVI after this reset, reconfigure all FCPVI registers. After the active period of MODRST is finished, it is switched back to 0 automatically. 0: NOP 1: Forcibly terminate the FCPVI operation. Note: Do not set both MODRST and SOFTRST bit to 1.
3 to 1	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
0	SOFTRST	0h	RW	FCPV SOFT ReSeT This reset stops the FCPVI module after finishing active bus transaction. To operate the FCPVI after this reset, reconfigure all FCPVI registers. After the active period of SOFTRST is finished, it is switched back to 0 automatically. 0: NOP 1: Terminates the FCPVI operation. Note: Do not set both MODRST and SOFTRST bit to 1.

(4) FCP Status Register (FCP_STA)

FCP_STA indicates the FCPVI status.

Access Size : 32 bits
Address : <FCPV0_base> + 0018h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ACT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
0	ACT	0h	R	FCPVI Active Flag 0: Indicates the FCPVI is not active. 1: Indicates the FCPVI is active.

9.9.2.6 Operation

FCPVI is not invoked directly by register setting. It works if there are bus accesses from the connected IP VSPI. This section explains how to control operation for FCPVI connected with VSPI.

9.9.2.6.1 Operation Control Setting IP set VSPI + FCPVI

(1) FCPVI initialization

FCPVI must be initialized with following settings. This initialization must be conducted before starting image process. After initialization, those settings are kept no change until power-off or hardware reset, therefore, they don't need to be set for every image process.

1. Set 1 to FCP_CFG0.FCPVSEL to enable configuration of FCPVI.

(2) To start image process of VSPI + FCPVI

FCPVI registers do not need to be set when starting an image process. Therefore, refer to **9.9.1 VSPI** for the guidance of starting an image process.

(3) To stop image process of VSPI+FCPVI immediately

Do following steps to stop immediately an image process of VSPI + FCPVI

1. Set value 1 to VSPI register VI6_WPF0_IRQ_ENB.FREE so that frame end interruption can be issued.
2. Set value 1 to VSPI register VI6_SRESET.SRST0. VSPI will invoke termination process immediately.
3. Wait frame end interruption from VSPI or until the register bit VI6_WPF0_IRQ_STA.FRE is set to 1. After it occurs, VSPI has finished its processing.
4. Set value 1 to FCPVI register FCP_RST.SOFTRST. FCPVI will invoke termination process immediately.
5. Wait until ACT bit of the FCP_STA register and become 0. After it occurs, FCPVI has finished its processing.

With this procedure, VSPI can stop its process quickly, but the output frame of the last frame is corrupted.

9.9.2.6.2 Reset Operation

FCPVI reset can be conducted by software reset and power on reset. The software reset which is controlled by software, is used to stop the FCPVI operation. The software reset operation is not executed immediately; it is executed on completion of the bus transaction. The other is power on reset which initialize the whole FCPVI logic immediately.

(1) Software Reset

1. Activate software reset sequence:
   Set FCP_RST.SOFTRST to the value 1.
2. Confirm software reset sequence finished:
   FCP_STA.ACT becomes 0. All remain transactions with the bus system finished.

The procedure to stop operation of FCPVI and connected IP is descripted at **9.9.2.6.1 Operation Control Setting IP set VSPI + FCPVI**.

(2) Power-on Reset, Hardware reset

The power-on reset is controlled by an externally input signal called hardware reset. This reset initializes the whole internal logic of FCPVI.

9.9.2.6.3 Register Settings and Restrictions on Connected VSPI

The connected IP VSPI have to follow the setting and restrictions in following tables.

Table 9.9-57 Setting Restriction about Output Color Formats

	Format setting (VI6_WPF0_OUTFMT)	Restrictions
Register	ROT[2:0]	Register Value (Explanation)
VI6_WPF0_ROT_CTRL.LN16	4-7	1 (always output in 16 lines mode)

9.9.2.7 Usage Notes

9.9.2.7.1 Manipulating FCPVI, VSPI with CPG module stop, software reset

The following notes 1. and 2. are to prevent FCPVI and VSPI from the unstable status which may cause hang or wrong operation.

NOTES

1. In the case of stopping FCPVI and VSPI using the corresponding register, do following steps. Note that it is needed to stop both FCPVI and VSPI.
   [Step 1] Stop FCPVI and VSPI with the software reset sequence following to the guidance in **9.9.2.6.2 Reset Operation**.
   [Step 2] Stop VSPI and FCPVI using the corresponding register with following **9.9.3.2 Procedure for Stopping VSP**.
2. In the case of re-starting FCPVI and VSPI after stopping those modules, be sure to run start process as defined in **9.9.3.1 Procedure for Starting VSP** and **9.9 Video Signal Processor (VSP)**.

9.9.3 Procedure for Starting and Stopping VSP

Follow the steps below.

For the clocks used in the VSP are provided by the CPG. For activating and stopping the clocks to VSP, access to the CPG module. Besides that, VPS have a clock divider to generate clocks used in the VSP. This clock divider is controlled by the reset signal and the module stop (mstp) signal for the VSP.

For controlling and observing the signals in the following steps, access to the SYS module or CPG modules.

Clock On/Off control register, Reset control register and Low power control registers are used for VSP (VSPI, FCPVI). Detail descriptions refer to **4.3 System Controller (SYS)** and **4.4 Clock Pulse Generator (CPG)**.

9.9.3.1 Procedure for Starting VSP

Table 9.9-58 is the Procedure for starting for VSP.

Table 9.9-58 Procedure for Starting VSP

Step	Description	Register Operation
1	Activate clocks for VSP from CPG. *Note*: This step can be omitted if clocks are already provided from CPG.	• Set 1b to CPG_CLKON_26.CLK5_ON (bit-5: CGC_VSPI_clk_m). • Set 1b to CPG_CLKON_26.CLK6_ON (bit-6: CGC_VSPI_clk_a_as). • Set 1b to CPG_CLKON_26.CLK7_ON (bit-7: CGC_VSPI_clk_p_as).
2	De-assert the reset signal for clock divider from CPG.	Set 1b to CPG_RST_17.RSTB11 (bit-11: VSPI_x_smreset_vspiss_n).
3	De-assert module stop signal for clock divider from SYS.	Set 1b to SYS_VSPI_CLK.x_mstp_vsp_clk_n (bit-1).
4	Confirm that clock divider is activated.	Read SYS_VSPI_CLK. • Confirm o_vspi_clk_mon (bit-0) is 1b.
5	De-assert reset signals for VSPI and FCPVI.	• Set 0b to CPG_RST_17.RSTB13 (bit-13: VSPI_x_spreset_fcpvi0). • Set 0b to CPG_RST_17.RSTB12 (bit-12: VSPI_x_sreset_vsp1z).
6	Release the low-power state of ADB400.	• Set 1b to CPG_LP_VSPI_CTL1.fvi_axi_clkqreqnm (bit-4). • Set 1b to CPG_LP_VSPI_CTL1.fvi_axi_clkqreqns (bit-0).
7	Confirm that ADB400 have leaved from the low-power state.	Read CPG_LP_VSPI_CTL1. • Confirm fvo_axi_clkqacceptnm (bit-5) is 1b. • Confirm fvo_axi_clkqacceptns (bit-1) is 1b.
8	Release the low-power state of the APB-IF	• Set 1b to CPG_LP_VSPI_CTL1.fvi_apb_pwr_qreq_n (bit-16). • Set 1b to CPG_LP_VSPI_CTL2.vsi_apb_pwr_qreq_n (bit-0).
9	Confirm that APB-IF have leaved from the low-power state.	Read CPG_LP_VSPI_CTL1. • Confirm fvo_apb_pwr_qaccept_n (bit-17) is 1b. Read CPG_LP_VSPI_CTL2. • Confirm vso_apb_pwr_qaccept_n (bit-1) is 1b

9.9.3.2 Procedure for Stopping VSP

Table 9.9-59 is the Procedure for stopping for VSP.

Table 9.9-59 Procedure for Stopping VSP

Step	Description	Register Operation
1	Make the APB-IF to the low-power state.	• Set 0b to CPG_LP_VSPI_CTL1.fvi_apb_pwr_qreq_n (bit-16). • Set 0b to CPG_LP_VSPI_CTL2.vsi_apb_pwr_qreq_n (bit-0).
2	Confirm that APB-IF have been in the low-power state.	Read CPG_LP_VSPI_CTL1. • Confirm fvo_apb_pwr_qaccept_n (bit-17) is 0b. Read CPG_LP_VSPI_CTL2. • Confirm vso_apb_pwr_qaccept_n (bit-1) is 0b
3	Make the ADB400 to the low-power state.	• Set 0b to CPG_LP_VSPI_CTL1.fvi_axi_clkqreqnm (bit-4). • Set 0b to CPG_LP_VSPI_CTL1.fvi_axi_clkqreqns (bit-0).
4	Confirm that ADB400 have been in the low-power state.	Read CPG_LP_VSPI_CTL1. • Confirm fvo_axi_clkqacceptnm (bit-5) is 0b. • Confirm fvo_axi_clkqacceptns (bit-1) is 0b.
5	Assert reset signals for VSPI and FCPVI.	• Set 1b to CPG_RST_17.RSTB13 (bit-13: VSPI_x_spreset_fcpvi0). • Set 1b to CPG_RST_17.RSTB12 (bit-12: VSPI_x_sreset_vsp1z).
6	Assert module stop signal for clock divider from SYS.	Set 0b to SYS_VSPI_CLK_x_mstp_vsp_clk_n (bit 1).
7	Confirm that clock divider have stopped.	Read SYS_VSPI_CLK. • Confirm o_vspi_clk_mon (bit-0) is 0b.
8	Assert the reset signal for clock divider from CPG.	Set 0b to CPG_RST_17.RSTB11 (bit-11: VSPI_x_smreset_vspiss_n).
9	Stop clocks for VSP from CPG. *Note*: This step can be omitted if clock stopping is not required.	• Set 0b to CPG_CLKON_26.CLK5_ON (bit-5: CGC_VSPI_clk_m). • Set 0b to CPG_CLKON_26.CLK6_ON (bit-6: CGC_VSPI_clk_a_as). • Set 0b to CPG_CLKON_26.CLK7_ON (bit-7: CGC_VSPI_clk_p_as).

SECTION 9 IMAGE

9.10 Frame Data Processor (FDP)

FDP consist of FDP1B and FCPF modules. Refer the detail for **9.10.1 FDP1B** and **9.10.2 FCPF**.

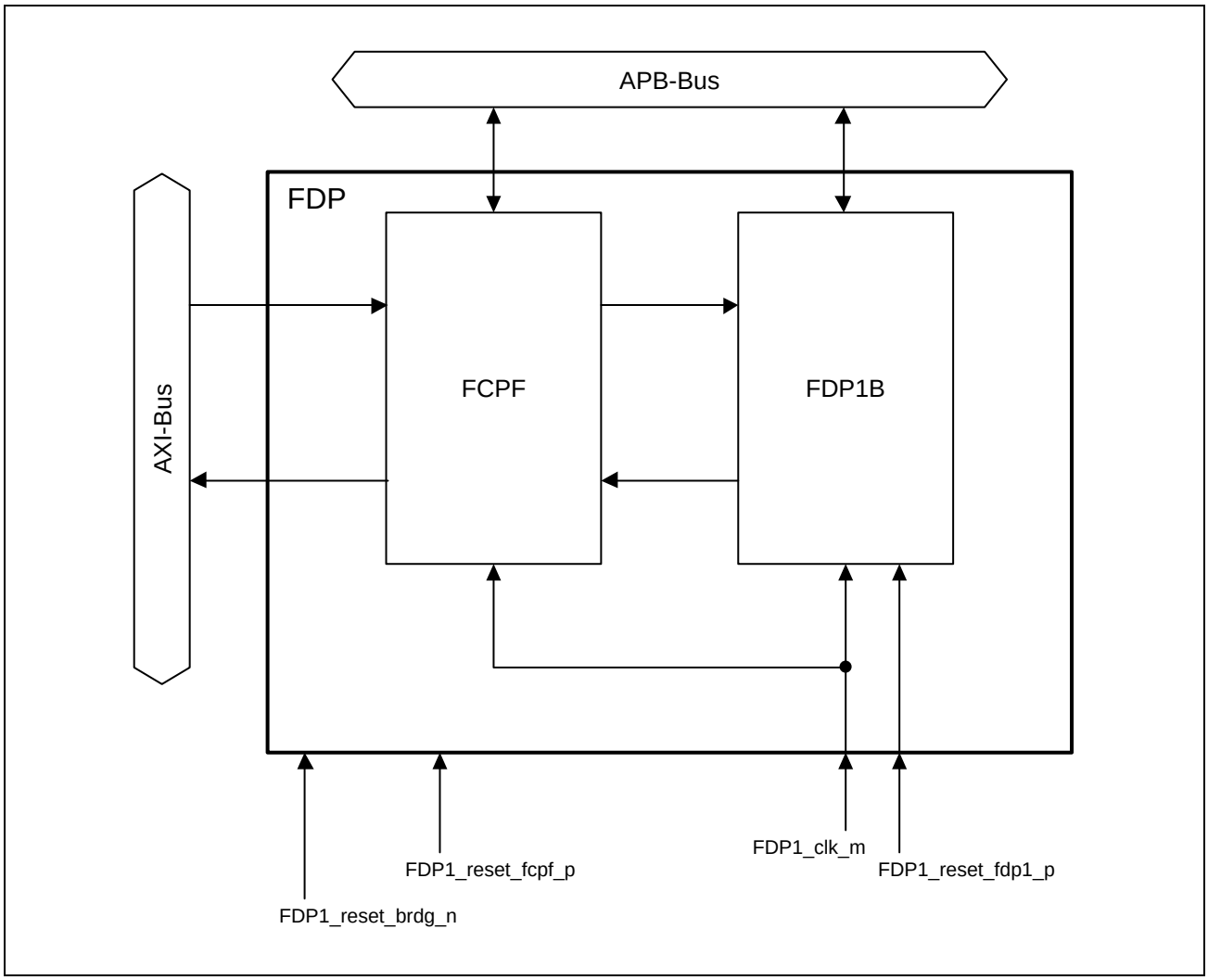


Figure 9.10-1 FDP Block Diagram

9.10.1 FDP1B

The FDP1B is the module which converts the tile video data to linear video. Require software or the library to RENESAS for operating this module.

This LSI provides one channel.

9.10.1.1 Features

This module has the following features.

- Support 8190 × 8190 resolution
- Performance of FDP1B is as following:
  - Output performance is 500 Mpixs/s. Input performance (progressive mode) is equal to output performance.

Table 9.10-1 Main Function of the FDP1B

• Read Pixel Formatter (RPF)			
Number of Channels			One Channel (RPF1)
Image Format	Input	YCbCr	• YCbCr444-PL, SP, ILV*1 • YCbCr422-PL, SP, ILV*1 • YCbCr420-PL, SP*1
		Maximum size	H8190 × V8190 pixels
		Minimum size	H32 × V32 pixels
		Size setting unit	• YCbCr420: 2-pixel units both horizontally and vertically. • YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. In other formats, the size can be set in 1-pixel units.
• Write Pixel Formatter (WPF)			
Number of Channels			One Channel (WPF)
Image format	Output	RGB	RGB332, RGB444, RGB565, RGB666, RGB888, αRGB8666, αRGB8888, αRGB4444, αRGB1555 α value is fixed.
		YCbCr	• YCbCr444-PL, SP, ILV*1 • YCbCr422-PL, SP, ILV*1 • YCbCr420-PL, SP*1
		Maximum size	H8190 × V8190 pixels
		Minimum size	H32 × V32 pixel
		Size setting unit	• YCbCr420: 2-pixel units both horizontally and vertically. • YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. In other formats, the size can be set in 1-pixel units.
Color space conversion	YCbCr to RGB	Conversion expression	• BT.601 (16, 235/240) to RGB (0, 255) • BT.709 (16, 235/240) to RGB (0, 255) • BT.601 (0, 255) to RGB (0, 255) • BT.709 (16, 235/240) to RGB (16, 235)
Changing number of colors	Output	Reducing RGB color depth	Dithering, lower-order bit truncation, or rounding
		YCbCr422/420	CbCr skipping or CbCr vertical skipping and horizontal skipping

Note 1. PL: Planar, SP: Semi-planar, ILV: Interleaved

9.10.1.2 Block Diagram

Figure 9.10-2 shows the block diagram of FDP1B.

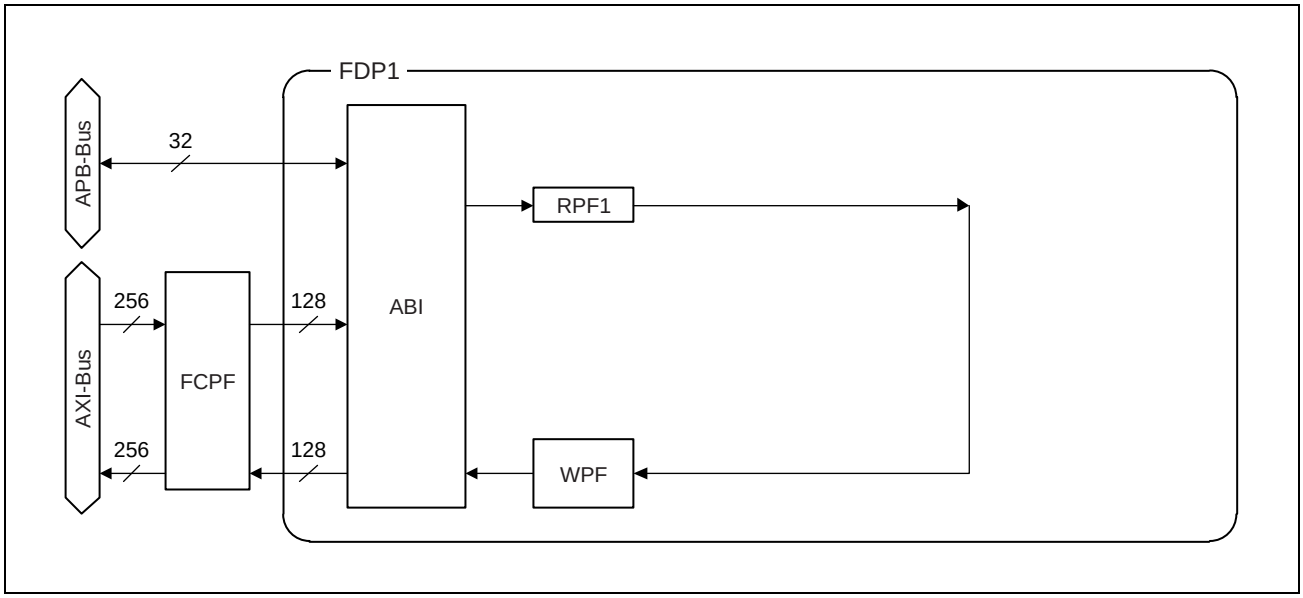


Figure 9.10-2 FDP1B Block Diagram

The following gives an overview of each block function shown in **Figure 9.10-2**.

The operation of these functions is determined by the register setting explained in **9.10.1.4 Register Description**. See **9.10.1.4 Register Description** for the relationship between the register setting and the functional behavior.

9.10.1.2.1 ABI (AXI Bus Interface)

The FDP1B applies processing to the image data stored in the external memory and writes the resultant data back to the external memory. The data transfer between the external memory and FDP1B necessary for this operation is done by the ABI, which works as the bus master, according to the register settings. The ABI executes this data transfer between the external memory and FDP1B through FCPF.

9.10.1.2.2 RPF (Read Pixel Formatter)

The RPF reads image data from the external memory through the ABI, unpacks data according to the specified format, and outputs the resultant data to the WPF. The input format unpacking unit expands the image data input from the ABI into the image format for internal processing.

The FDP1B provides RPF1 module.

9.10.1.2.3 WPF (Write Pixel Formatter)

The WPF is an output module that receives image data from the RPF1, converts the color space, number of colors, and format of the data, and outputs the results of FDP1B image processing to external memory through the ABI. The WPF is mainly configured from a color space converter and an output format converter (the packing unit).

The color space converter converts the color space between RGB and YCbCr, and the packing unit converts the format into the picture plane storing format. The FDP1B provides one WPF.

9.10.1.2.4 Input/Output Data Format

The FDP1B input/output data should be stored in a format where the address is incremented in the direction from the MSB to the LSB as in big endian. For example, the FDP1B reads and writes data to and from the external memory through the dedicated bus interface with a 128-bit width, and the data that should come first should be stored in the MSB side regardless of the unit data size as shown in **Figure 9.10-3**.

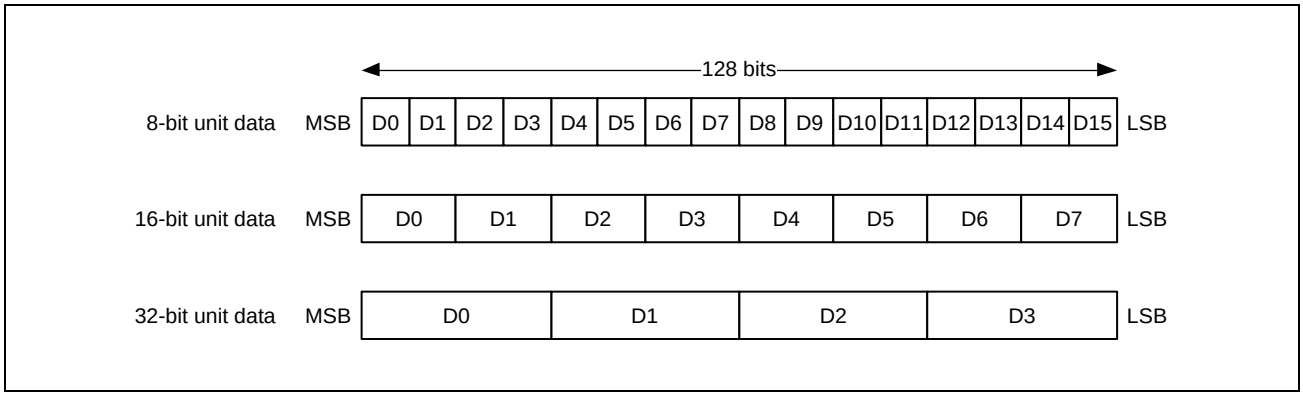


Figure 9.10-3 FDP1B Input/output Data Format

However, data may be arranged in a way unexpected by the FDP1B depending on the address handling method (such as little endian) of the FDP1B access destination in some cases as in the external memory format shown in **Figure 9.10-4**. In such a case, use the data swapping function (Refer to **9.10.1.4.7 (5) RPF Data Swap Register (FD1_RPF1_SWAP)** and **9.10.1.4.8 (4) Destination Component-m Address Register (FD1_WPF_ADDR_m) (m = Y, C0, C1)**) in the FDP1B to convert the data format to be suitable for the FDP1B.

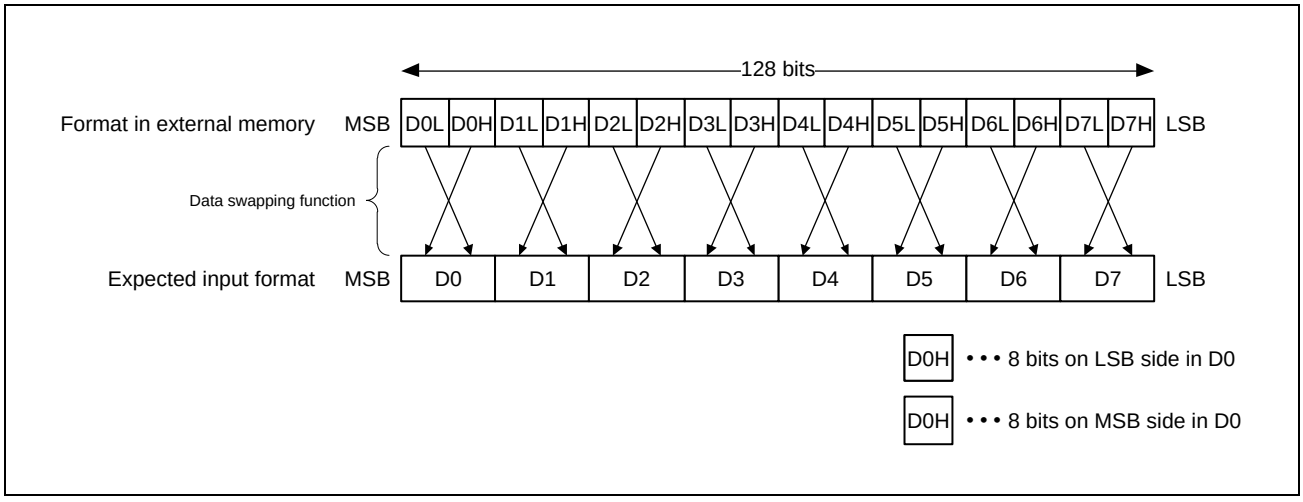


Figure 9.10-4 Input Data Alignment by Data Swapping Function

9.10.1.3 Register Configuration

Table 9.10-2 shows the FDP1B memory map. Immediate registers are specified as “Imm”, the V-update registers are indicated as “Vupdt” and the V-update status registers are indicated as “VupdtSt” in register type column in **Table 9.10-2**.

Base address of FDP1B:

Ch0: 0_1630_0000h (5630_0000h*[1], 4630_0000h*[2])

Note 1. CM33 address space (non-secure)
Note 2. CM33 address space (secure)

Table 9.10-2 List of FDP1B Registers

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
FDP1 Start Register	FD1_CTL_CMD	0000_0000h	0000h	32
Sync Generator Register	FD1_CTL_SGCMD	0000_0000h	0004h	32
Register Set End Register	FD1_CTL_REGEND	0000_0000h	0008h	32
Channel Activation Register	FD1_CTL_CHACT	0000_0000h	000Ch	32
Operation Mode Register	FD1_CTL_OPMODE	0000_0000h	0010h	32
V-Period Register	FD1_CTL_VPERIOD	0000_0000h	0014h	32
Clock Control Register	FD1_CTL_CLKCTRL	0000_0000h	0018h	32
Software Reset Register	FD1_CTL_SRESET	0000_0000h	001Ch	32
Reserve	-	-	0020h to 0023h	-
Operating Status Register	FD1_CTL_STATUS	0000_0000h	0024h	32
V-Cycles Status Register	FD1_CTL_VCYCLE_STAT	0000_0000h	0028h	32
Reserve	-	-	002Ch to 0037h	-
Interrupt Enable Register	FD1_CTL_IRQENB	0000_0000h	0038h	32
Interrupt Status Register	FD1_CTL_IRQSTA	0000_0000h	003Ch	32
Interrupt Control Register	FD1_CTL_IRQFSET	0000_0000h	0040h	32
Reserve	-	-	0044h to 005Fh	-
Source Picture Size Register	FD1_RPF1_SIZE	0000_0000h	0060h	32
Source Picture Format Register	FD1_RPF1_FORMAT	0000_0000h	0064h	32
Source Picture Stride Register	FD1_RPF1_PSTRIDE	0000_0000h	0068h	32
Reserve	-	-	006Ch to 0077h	-
RPF1 Source Component Y Address Register	FD1_RPF1_ADDR_Y	0000_0000h	0078h	32
RPF1 Source Component C0 Address Register	FD1_RPF1_ADDR_C0	0000_0000h	007Ch	32
RPF1 Source Component C1 Address Register	FD1_RPF1_ADDR_C1	0000_0000h	0080h	32
Reserve	-	-	0084h to 0090h	-
RPF Data Swap Register	FD1_RPF_SWAP	0000_0000h	0094h	32
Reserve	-	-	0098h to 00BFh	-
Destination Picture Format Register	FD1_WPF_FORMAT	0000_0000h	00C0h	32
Destination Picture Rounding Control Register	FD1_WPF_RNDCTL	0000_0000h	00C4h	32

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
Destination Picture Stride Register	FD1_WPF_PSTRIDE	0000_0000h	00C8h	32
Destination Component Y Address Register	FD1_WPF_ADDR_Y	0000_0000h	00CCh	32
Destination Component C0 Address Register	FD1_WPF_ADDR_C0	0000_0000h	00D0h	32
Destination Component C1 Address Register	FD1_WPF_ADDR_C1	0000_0000h	00D4h	32
WPF Data Swap Register	FD1_WPF_SWAP	0000_0000h	00D8h	32
Reserve	-	-	00DCh to 00FFh	-
IPC Mode Register	FD1_RPF1_MODE	0000_0000h	0100h	32
Still Mask Threshold Register	FD1_RPF1_SMSK_THRESH	0000_0000h	0104h	32
Comb Detection Parameter Register	FD1_RPF1_COMB_DET	0000_0000h	0108h	32
Motion Decision Parameter Register	FD1_RPF1_MOTDEC	0000_0000h	010Ch	32
Reserve	-	-	0110h to 011Fh	-
DLI Blend Parameter Register	FD1_RPF1_DLI_BLEND	0000_0000h	0120h	32
DLI Horizontal Frequency Gain Register	FD1_RPF1_DLI_HGAIN	0000_0000h	0124h	32
Reserve	-	-	0128h	32
DLI Angle Parameter Register	FD1_RPF1_DLI_ANGLE	0000_0000h	012Ch	32
DLI Isolated Pixel Parameter Register0	FD1_RPF1_DLI_ISOPIX0	0000_0000h	0130h	32
DLI Isolated Pixel Parameter Register1	FD1_RPF1_DLI_ISOPIX1	0000_0000h	0134h	32
Reserve	-	-	0138h to 013Fh	-
IPC Sensor Threshold Register0	FD1_RPF1_SENSOR_TH0	0000_0000h	0140h	32
IPC Sensor Threshold Register1	FD1_RPF1_SENSOR_TH1	0000_0000h	0144h	32
Reserve	-	-	0148h to 016Fh	-
Sensor Control Register0	FD1_SENSOR_CTL0	0000_0000h	0170h	32
Sensor Control Register1	FD1_SENSOR_CTL1	0000_0000h	0174h	32
Reserve	-	-	0178h to 01DFh	-
Line Memory Pixel Number Register	FD1_IPC_LMEM	0000_0000h	01E0h	32
Reserve	-	-	01E4h to 07FFh	-
IP Internal Data	FD1_IP_INTDATA	0202_0202h	0800h	32

9.10.1.4 Register Description

9.10.1.4.1 Access Restrictions

FDP1B supports only 32-bit access to read or write all addresses in FDP1B area. Do not access any of otherwise access units. Therefore, read-modify-write is needed to change partial bits in 32bits register.

9.10.1.4.2 Register Type Definitions

There are 3 types of registers in FDP1B: immediate register, V-update Register and V-update Status Register. The values of immediate registers are reflected to H/W behavior immediately after the changing its values. Hence, do not change these immediate registers while FDP1B is processing unless this manual allows it explicitly. The values of V-update registers are reflected to H/W behavior at the timing of V-interruption (at the timing of FD1_CTL_IRQSTA.VINT is set to 1). Users can change these V-update registers while FDP1B is processing. To reflect V-update registers to the H/W, users have to set FD1_CTL_REGEND.REGEND to 1 after the all V-update registers which are intended to be changed are completely set. The values of V-update Status Registers are updated at the timing of V-interruption. Therefore, the status (FD1_CTL_STATUS) of frame process which is conducted before a V-interruption must be referred after that V-interruption.

To confirm the types of registers, refer to **Table 9.10-2**.

9.10.1.4.3 Notational Conventions for Registers and Bit Fields

This document uses the following notational conventions for the FDP1B registers and bit fields.

1. The names of registers and bits are written in uppercase.
2. A bit or bit field in a register is indicated as [register name.bit name]. For example, the STRCMD bit in the FD1_CTL_CMD register is indicated as FD1_CTL_CMD.STRCMD.
3. In each subsection for register description in **9.10.1.4 Register Description**, when only a bit name is written without showing its register name, the bit is in the register described in that subsection.

9.10.1.4.4 Register Classification

The FDP1B registers are arranged in the following order; the general control registers control the operation of the entire FDP1B, and the other registers control each image processing and specify parameters for the processing. The functions of the registers are described in this order starting from **9.10.1.4.6 General Control Registers**.

1. General control registers (FD1_CTL_*)
2. RPF control registers (FD1_RPF_*)
3. WPF control registers (FD1_WPF_*)

9.10.1.4.5 Restrictions on Access to Registers and Lookup Tables

The FDP1B has control registers. When accessing the addresses where these registers are allocated, the following restrictions should be satisfied. If any restriction is violated, the FDP1B will not operate correctly.

1. For the read-only bits and reserved bits in all FDP1B registers, writing 1 is prohibited unless otherwise specified.
2. Addresses undefined in **9.10.1.3 Register Configuration** are reserved areas and write access is prohibited in these areas.
3. For all immediate registers, except FD1_CTL_SGCMD, FD1_CTL_SRESET and FD1_CTL_IRQ*, modifying register values during operation of the module is prohibited. Modify registers while the FDP1B is stopped. For the operating status of the FDP1B, refer to **9.10.1.4.6 (9) Operating Status Register (FD1_CTL_STATUS)**.

9.10.1.4.6 General Control Registers

(1) FDP1B Start Register (FD1_CTL_CMD)

Access Size : 32 bits

Address : <FDP1_base> + 0000h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STRCMD
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	STRCMD	0h	RW	Start Command FDP1 starts the FD1 process if this bit is set to 1. Set this bit after all relevant registers are surely set except for FD1_CTL_REGEND and FD1_CTL_SGCMD. Set this bit to 1 before the FD1_CTL_REGEND.REGEND and FD1_CTL_SGCMD if FDP1 has to execute process. This bit will keep the value 1 after the set. To clear this bit, write the value 0 to this bit. 0: NOP or stops process at the next timing of V-Interruption. 1: Start FDP1 process at the next timing of V-Interruption. Note that setting of this register is valid if the register FD1_CTL_REGEND.REGEND is set to the value 1 at the timing of next V-Interruption.

(2) Sync Generator Register (FD1_CTL_SGCMD)

Access Size : 32 bits
Address : <FDP1_base> + 0004h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SGEN
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SGEN	0h	RW	V-Interruption Generator Enable This bit controls V-Interruption. This bit should be set to the value 1 after other all registers are set, and only at the first field of the sequence. Set this bit to the value 1 to execute process in all V-Interruption modes described in **9.10.1.4.6 (5)**. 0: Disable V-Interruption 1: Enable V-Interruption [Important] In case of software reset (FD1_CTL_SRESET.SRST), set this SGEN bit to the value 0. If this bit is cleared during process V-interruption will not occur after FDP1 finished current frame. As a result, FDP1 will not start the next frame because STRCMD is not captured and V-update status registers will not be updated about process information of current frame.

(3) Register Set End Register (FD1_CTL_REGEND)

Access Size : 32 bits
Address : <FDP1_base> + 0008h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	REGEND
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	REGEND	0h	RW	Register Setting Ready Flag Set the value 1 to this bit after completing the setting of parameter registers. Do not write the value 0 to this bit. This bit will be cleared by FDP1 automatically. 0: Writing 0 is prohibited except for after the completion of S/W reset sequence. 1: Registers which are set by SW is ready to be reflected for Hardware reference. The V-update status registers is ready to be reflected for Software reference. The second or later V-interruption in the case No interrupt mode can be generated.

(4) Channel Activation Register (FD1_CTL_CHACT)

Access Size : 32 bits
Address : <FDP1_base> + 000Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	SMW	WR	-	-	-	-	SMR	RD[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0
9	SMW	0h	RW	Refer to **Table 9.10-3** for setting this bit.
8	WR	0h	RW	Refer to **Table 9.10-3** for setting this bit.
7 to 4	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0
3	SMR	0h	RW	Refer to **Table 9.10-3** for setting this bit.
2 to 0	RD[2:0]	0h	RW	Refer to **Table 9.10-3** for setting these bits.

Table 9.10-3 FD1_CTL_CHACT Setting for DIM Parameter

Use Case	PRG*1	DIM [2:0]*2	FD1_CTL_CHACT						
			SMW	WR	SMR	RD2	RD1	RD0	Word Expression
Progressive Mode	1	1	0	1	0	0	1	0	0000_0102h

Note 1. FD1_CTL_OPMODE.PRG

Note 2. FD1_IPC_MODE.DIM [2:0]

(5) Operation Mode Register (FD1_CTL_OPMODE)

Access Size : 32 bits
Address : <FDP1_base> + 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	PRG	-	-	VIMD[1:0]	
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PRG	0h	RW	Progressive Mode Set the value 1 to this bit as default.
3, 2	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	VIMD[1:0]	0h	RW	V-Interruption Mode This register field sets V-Sync interrupt mode of FDP1. Refer to **Table 9.10.1-4** for detail. 0: Interrupt mode 1: Best Effort mode 2: No interrupt mode

Table 9.10-4 V-Interrupt Mode

VIMD[1:0]	V-Interruption Timing		
	First Time	Second Time or Later	STRCMD Captured Timing
0 (Interrupt mode)	V-Interruption occurs when changes FD1_CTL_SGCMD.SGEN from 0 to 1 and FDP1B is not doing frame process.	Fixed period interrupt specified by FD1_CTL_VPERIOD in spite of the frame processing.	Captured at V-Interruption
1 (Best effort)		Fixed period interrupt specified by FD1_CTL_VPERIOD. If the frame processing is not completed at the timing specified by FD1_CTL_VPERIOD, V-Interruption is deferred to the end of the processing.	
2 (No interrupt)		• If FD1_CTL_REGEND is set 1 before the end of the processing (frame end interruption FD1_CTL_IRQSTA .FRE is set as 1) V-Interruption occurs right after the end of the processing. • If FD1_CTL_REGEND is set 1 after the end of processing (frame end interruption FD1_CTL_IRQSTA .FRE is set as 1) V-Interruption occurs right after FD1_CTL_REGEND is set. • If FD1_CTL_REGEND is not set 1 V-Interruption does not occur.	

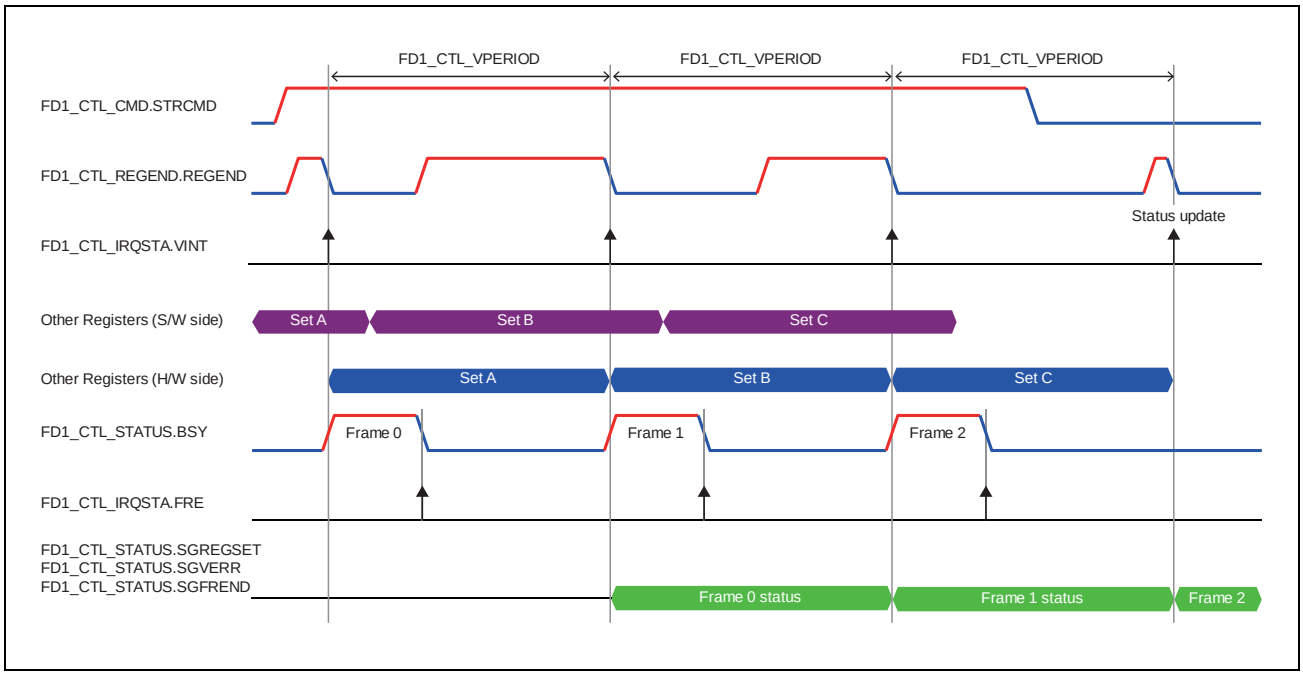


Figure 9.10-5 Timing Chart of Interrupt Mode (VIMD = 0) and Best Effort Mode (VIMD = 1)

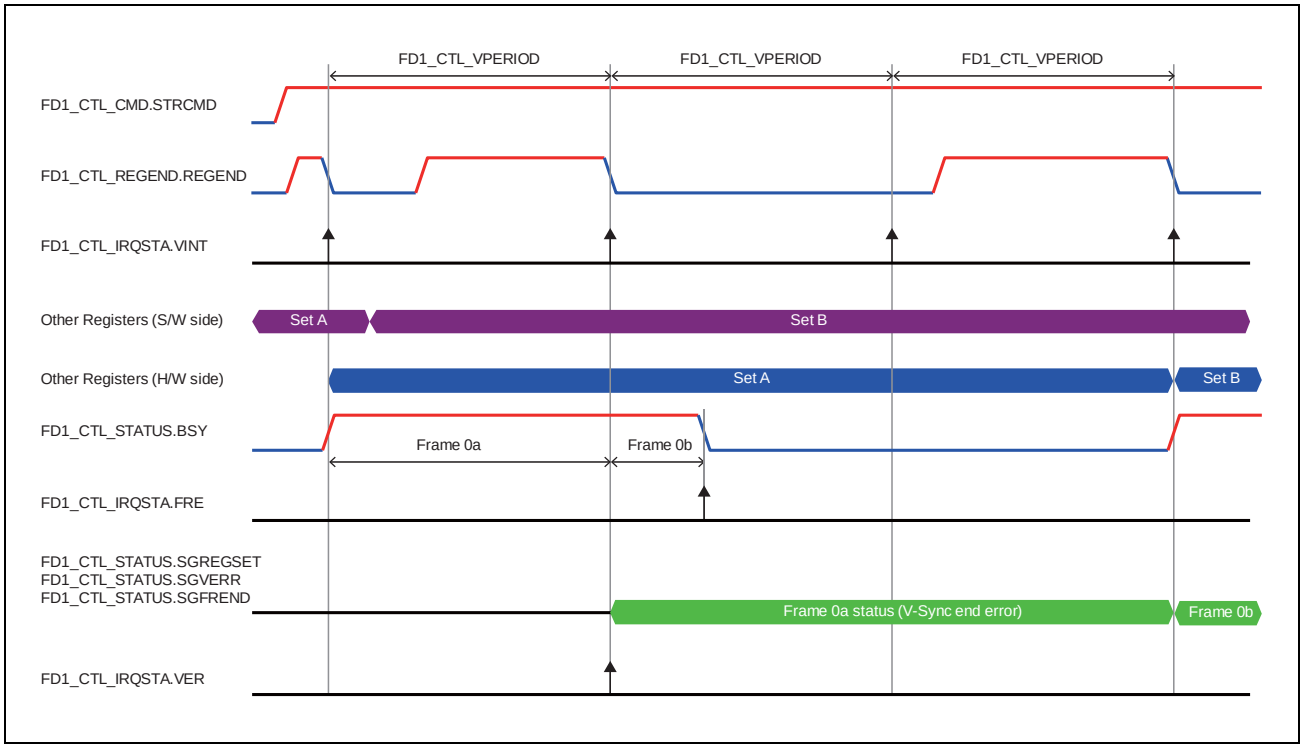


Figure 9.10-6 Timing Chart of Interrupt Mode (VIMD = 0) with Error

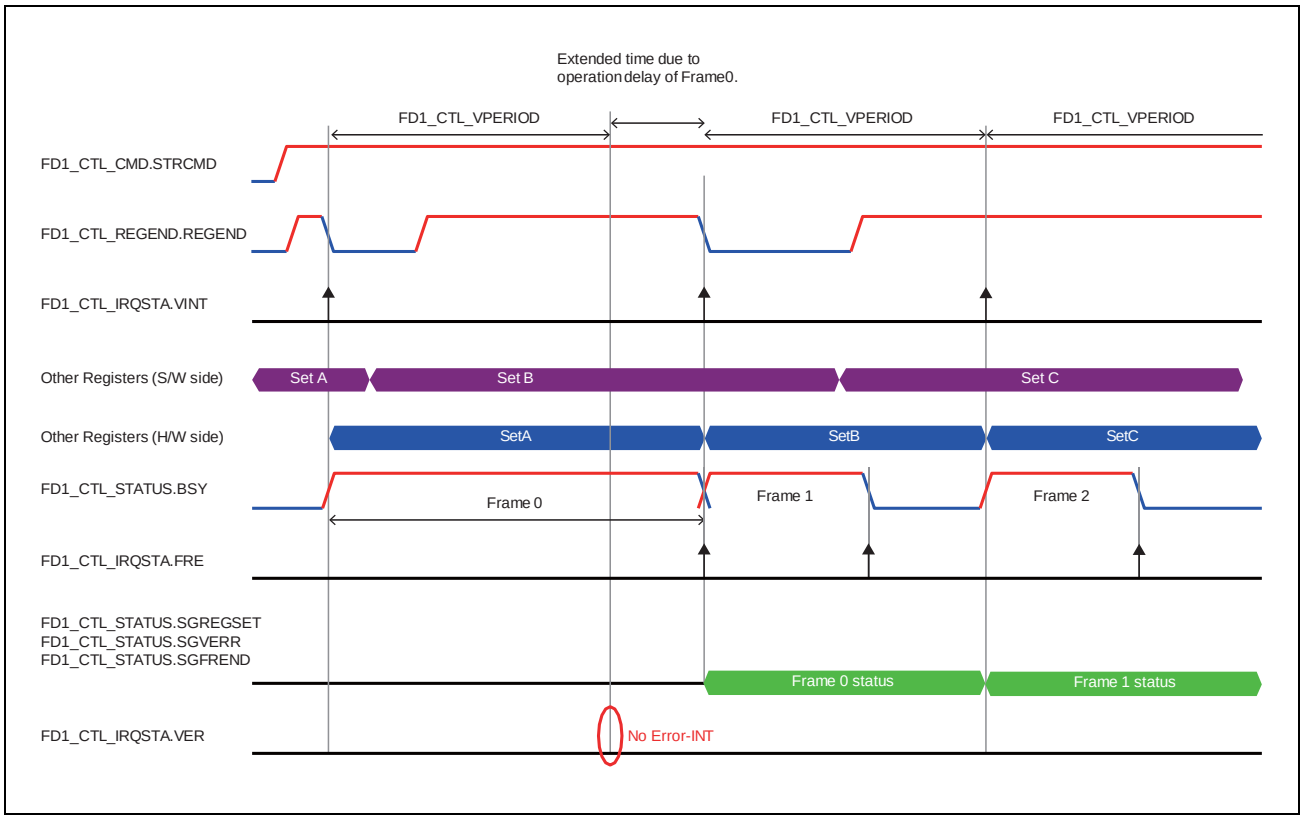


Figure 9.10-7 Timing Chart of Best Effort Mode (VIMD = 1) (In case of Error at Interrupt Mode)

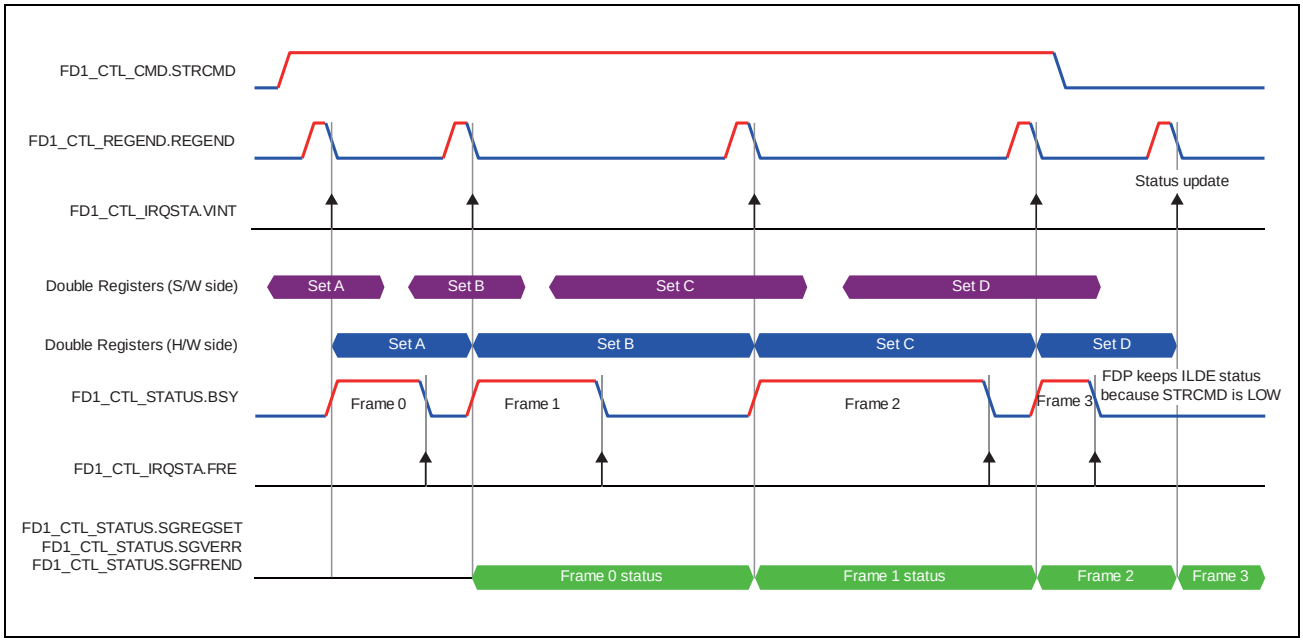


Figure 9.10-8 Timing Chart of No Interrupt Mode (VIMD = 2)

(6) V-Period Register (FD1_CTL_VPERIOD)

Access Size : 32 bits
Address : <FDP1_base> + 0014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VPERIOD[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VPERIOD[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	VPERIOD [31:0]	00000000h	RW	V-Sync Period Set the period from the V-Sync to the next V-Sync in the unit of FDP1 clock frequency (800MHz) cycles.

(7) Clock Control Register (FD1_CTL_CLKCTRL)

Access Size : 32 bits

Address : <FDP1_base> + 0018h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CSTP_N
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	CSTP_N	0h	RW	Set the value 1 to this bit.

(8) Software Reset Register (FD1_CTL_SRESET)

Access Size : 32 bits

Address : <FDP1_base> + 001Ch

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SRST
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SRST	0h	RW	FDP1 Software Reset FDP1 stops it operation after this bit is set to 1. FDP1 aborts all processing and quit bus/ memory accesses. Do not start next process by FD1_CTL_CMD until the software processing is completed. When this processing is completed, the FD1_ CTL_IRQSTA.FRE interrupt source bit is set to 1, when the FRE interrupt is enabled. This interrupt notifies the end of the reset processing. The end of software reset processing is notified through the FRE bit, but the software reset issued while FDP1 is stopped is ignored as NOP. As it takes a while until the reset is actually issued after the reset bit is set, the FDP1 may complete operation before the reset is actually issued. In this case, no interrupt is output for the software reset that is issued after the FDP1 completes operation. But even through in this case, not that the output image may be destroyed due to software reset process. This bit is always read as 0. 0: NOP 1: Software Reset [Important] Set FD1_CTL_SGCMD.SGEN to the value 0 before write 1 to this SRST bit.

(9) Operating Status Register (FD1_CTL_STATUS)

Access Size : 32 bits
Address : <FDP1_base> + 0024h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VINT_CNT[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	SGREGSET	SGVERR	SGFREND	-	-	-	-	-	-	-	BSY
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	VINT_CNT[15:0]	0000h	R	V-Sync Interrupt Counter Status This bit field shows the number of interruption times of V-Sync interrupt. Hence, the value of this bit indicates the number of progressive output frames which is generated by FDP1. This counter value is reset to 0 when FD1_CTL_SGCMD.SGEN is set to 1.
15 to 11	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	SGREGSET	0h	R	Register Set End Status This bit represents the register set status updated at V-Sync. The status is controlled by FD1_CTL_REGEND.REGEND bit and the internal timing. 0: Registers are not set for the operation 1: Registers are set for the operation
9	SGVERR	0h	R	V-Sync End Error Status This bit represents the V-Sync end error status updated at V-Sync. 0: No error 1: V-Sync end error (Fram processor cannot be finished by the timing of V-Sync)
8	SGFREND	0h	R	Frame End Status This bit represents the frame end status updated at V-Sync. 0: The frame processor is not finished 1: Frame end (The frame processor is finished)
7 to 1	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	BSY	0h	R	FDP1 Operating Status This bit indicates the operating or stopped state of control FDP1. 0: FDP1 is stopped. 1: FDP1 is operating.

(10) V-Cycles Status Register (FD1_CTL_VCYCLE_STAT)

Access Size : 32 bits
Address : <FDP1_base> + 0028h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VCYC[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VCYC[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	VCYC[31:0]	00000000h	R	Number of Cycles of the Previous Frame Processing This status register shows the number of cycles of previous frame processing. FDP1 updates this register at the timing of V-Sync interrupt.

(11) Interrupt Enable Register (FD1_CTL_IRQENB)

Access Size : 32 bits
Address : <FDP1_base> + 0038h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	VERE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	VINTE	-	-	-	FREE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	VERE	0h	RW	Interrupt Enable for V-Sync End Error The interrupt of V-Sync end error is asserted at the timing that FDP1 does not complete the frame processing at the V-Sync.
15 to 5	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	VINTE	0h	RW	Interrupt Enable for V-Sync The interrupt of V-Sync is asserted at the timing of V-Sync specified by FD1_CTL_OPMODE.VIMD register field. If VIMD is set to 2, this interrupt is not asserted.
3 to 1	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	FREE	0h	RW	Interrupt Enable for Frame End The interrupt of frame end is asserted at the timing that FDP1 completes the frame processing.

Each bit controls the interrupt enable of the corresponding interrupt source.

0: Interrupt Disabled

1: Interrupt Enabled

Each bit in FD1_CTL_IRQSTA is set to 1 when the corresponding interrupt source is generated. FD1_CTL_IRQENB specifies whether to output an interrupt signal for the generated source. When an interrupt is disabled in this register, no interrupt signal is generated even when the corresponding bit in FD1_CTL_IRQSTA is set to 1. When an interrupt is enabled in this register, an interrupt signal is output when the corresponding bit in FD1_CTL_IRQSTA is set to 1.

(12) Interrupt Status Register (FD1_CTL_IRQSTA)

Access Size : 32 bits
Address : <FDP1_base> + 003Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	VER
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	VINT	-	-	-	FRE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	VER	0h	RW	Interrupt Status and Clear for V-Sync End Error
15 to 5	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	VINT	0h	RW	Interrupt Status and Clear for V-Sync
3 to 1	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	FRE	0h	RW	Interrupt Status and Clear for Frame End

The read value from each bit is the status of the interrupt source, and the write access to each bit controls the interrupt status.

[Read Access] Interrupt Status

0: No interrupt

1: Interrupt activated

[Write Access] Interrupt Clear

0: The interrupt status is cleared to 0

1: Hold the interrupt status value

FD1_CTL_IRQSTA indicates the state of the interrupt sources generated in the FDP1B. Whether to output a FDP1B interrupt when an interrupt source is generated and the corresponding bit is set to 1 is determined by the corresponding bit setting in FD1_CTL_IRQENB.

While an interrupt is disabled in FD1_CTL_IRQENB, the FDP1B does not output an interrupt signal even when an interrupt source is generated, but the source flag in this register is set to 1.

(13) Interrupt Control Register (FD1_CTL_IRQFSET)

Access Size : 32 bits
Address : <FDP1_base> + 0040h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	FRES
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	FRES	0h	RW	This bit is used to issue the dummy frame end interruption. [Read Access] Always 0 value is read. [Write Access] 0: Hold the current interrupt status value 1: The interrupt status of frame end (bit FRE of register FD1_CTL_IRQSTA) is forced to 1. After being forced to 1, the status value 1 of interrupt status bit FRE is hold until it is cleared by being written 0. The bit FRES turns to 0 automatically after being written 1.

9.10.1.4.7 RPF Control Registers

(1) Source Picture Size Register (FD1_RPF1_SIZE)

Access Size : 32 bits
Address : <FDP1_base> + 0060h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	HSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VSIZE[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
28 to 16	HSIZE[12:0]	0000h	RW	Horizontal Source Picture Size Specify the horizontal picture size of input frame in unit of pixels.
15 to 13	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
12 to 0	VSIZE[12:0]	0000h	RW	Vertical Source Picture Size Specify the vertical picture size of input frame in unit of pixels.

(2) Source Picture Format Register (FD1_RPF1_FORMAT)

Access Size : 32 bits
Address : <FDP1_base> + 0064h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CIPM
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	RSPYCS	RSPUVS	-	-	-	-	-	RDFMT[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
16	CIPM	0h	RW	Horizontal Chrominance Interpolation Method Setting Image data is processed in the YCbCr444 format inside RPF in case of YCbCr color space. When the chrominance format of the input image is YCbCr422 or YCbCr420, data is up-sampled as shown in **Figure 9.10-9**. This bit specifies the method of up-sampling for this purpose. 0: The nearest-neighbor method is used for horizontal chrominance interpolation. 1: The bilinear method is used for horizontal chrominance interpolation. In the case all following conditions are true set 0 to this bit in order to prevent the image difference between input and output. If set 1 to this bit in this case the output image may be different from the input image, but it is not broken. 1) Input format is either YCbCr4:2:2 or YCbCr4:2:0 2) Output format is either YCbCr4:2:2 or YCbCr4:2:0
15, 14	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
13	RSPYCS	0h	RW	RPF Input Mode Setting 1 When the input format is YUY2, set this bit to 1 and set the RDFM T bits to 71 (47h). When the input format is YVYU, set this bit and the RSPUVS bit to 1 and set the RDFM T bits to 71 (47h). In other cases, set this bit to 0.
12	RSPUVS	0h	RW	RPF Input Mode Setting 2 When the input format is NV61, set this bit to 1 and set the RDFM T bits to 65 (41h). When the input format is NV21, set this bit to 1 and set the RDFM T bits to 66 (42h). When the input format is YVYU, set this bit and the RSPYCS bit to 1 and set the RDFM T bits to 71 (47h). In other cases, set this bit to 0.
11 to 9	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
8	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.
7	-	0h	R	Reserved These bits are read as 0b. The written value will be ignored.

Bit	Bit Name	Initial Value	R/W	Description
6 to 0	RDFMT[6:0]	00h	RW	RPF Input Image Format Setting These bits select the format of the image input from the external SDRAM to the RPF1. Select a value corresponding to the desired format from those shown in **Table 9.10-5**. [Note 1] Number of input pixels When YCbCr4:2:2 is selected through the RDFMT bits, the horizontal size of the input image should be specified in 2-pixel units. When YCbCr4:2:0 is selected, the vertical and horizontal sizes should be specified in 2-pixel units. [Note 2] Input chroma horizontal up-sampling When the input format is YCbCr4:2:2 or YCbCr4:2:0 the chroma component is up-sampled horizontally to YCbCr4:4:4 (the format used for image process) with the method specified by the CIPM bit. [Note 3] Input chroma vertical up-sampling When the input format is YCbCr4:2:0 the chroma component is up-sampled vertically to YCbCr4:4:4 (the format used for image process) by copying the neighbor upper input line.

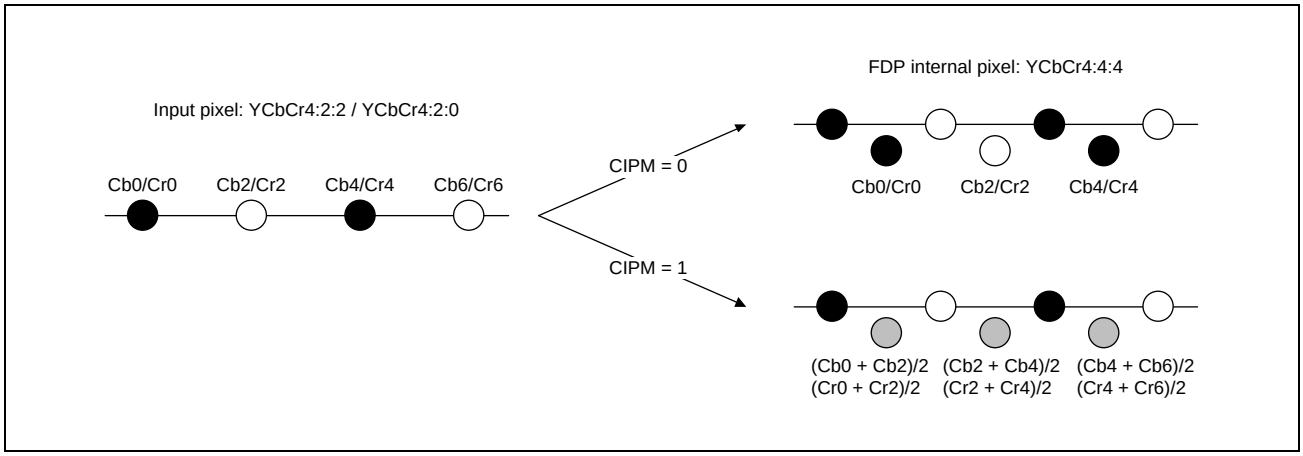


Figure 9.10-9 Chrominance Interpolation Methods Selectable through CIPM Setting

Table 9.10-5 YCbCr Formats for RPF Input and WPF Output

RDFMT [6:0], WRFMT[6:0]	Packed YcbCr input/output Format	Reference
00h to 3Fh	Reserved	—
40h	YcbCr4:4:4 semi-planar	**Figure 9.10-10**
41h	YCbCr4:2:2 semi-planar (NV16, NV61*1)	
42h	YCbCr4:2:0 semi-planar (NV12, NV21*1)	
43h to 45h	Reserved	—
46h	YcbCr4:4:4 interleaved	**Figure 9.10-11**
47h	YcbCr4:2:2 interleaved type 0 (UYVY, YUY2*2, YVYU*3)	
48h	YcbCr4:2:2 interleaved type 1	
49h	Reserved	
4Ah	YcbCr4:4:4 planar	**Figure 9.10-12**
4Bh	YcbCr4:2:2 planar (YV16)	
4Ch	YcbCr4:2:0 planar (YV12, I420)	
4Dh to 7Fh	Reserved	—

Note 1. When the input format is NV61 or NV21, set the RSPUVS bit to 1.

Note 2. When the input format is YUY2, set the RSPYCS bit to 1.

Note 3. When the input format is YVYU, set the RSPUVS bit to 1 and RSPYCS bit to 1.

In write format case shown in **9.10.1.4.8 (1) Destination Picture Format Register (FD1_WPF_FORMAT)**, FD1_WPF_FORMAT.WSPYCS, WSPUVS are corresponds to RSPYCS and RSPUVS, respectively.

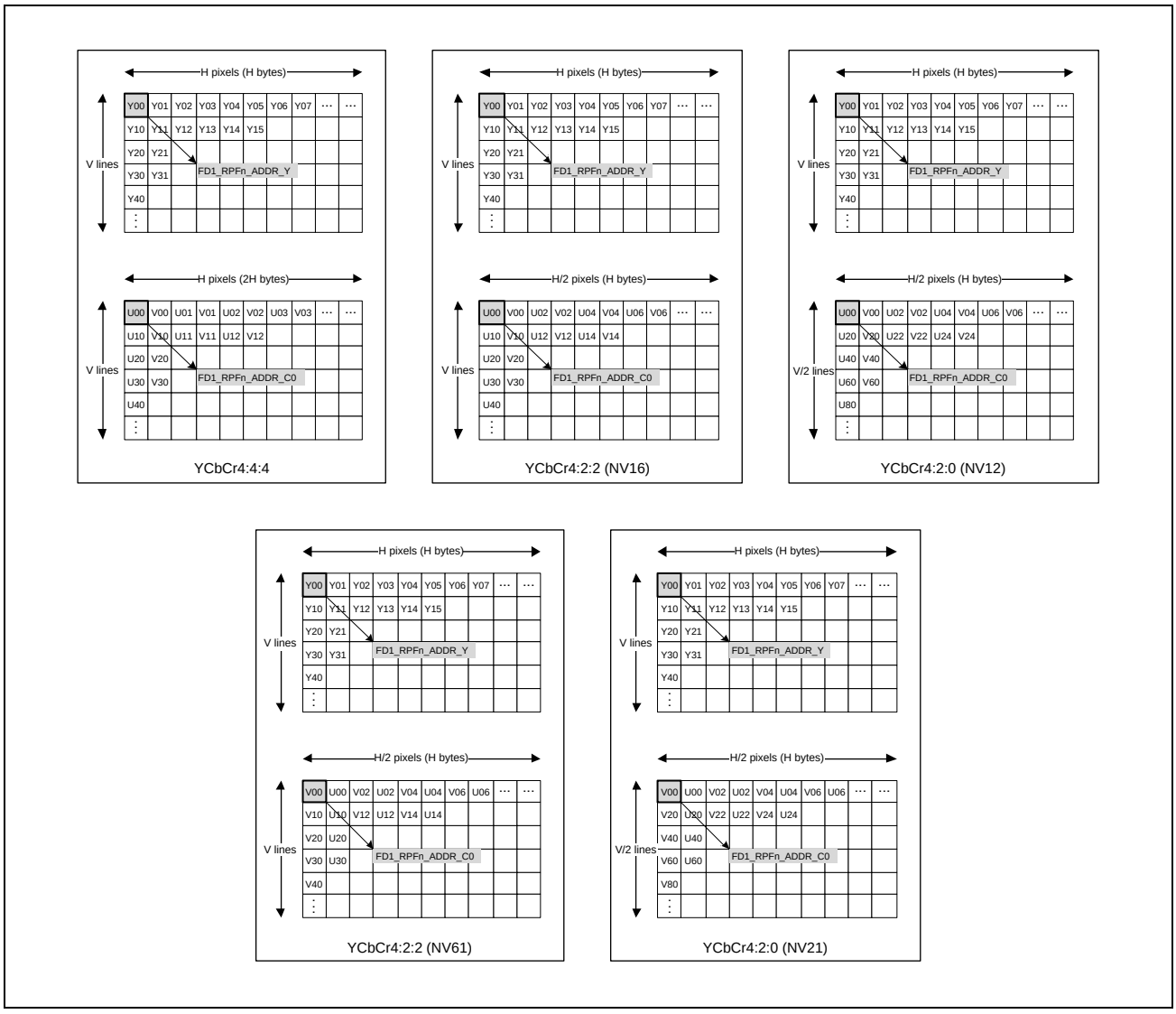


Figure 9.10-10 YcbCr Semi-Planar Formats

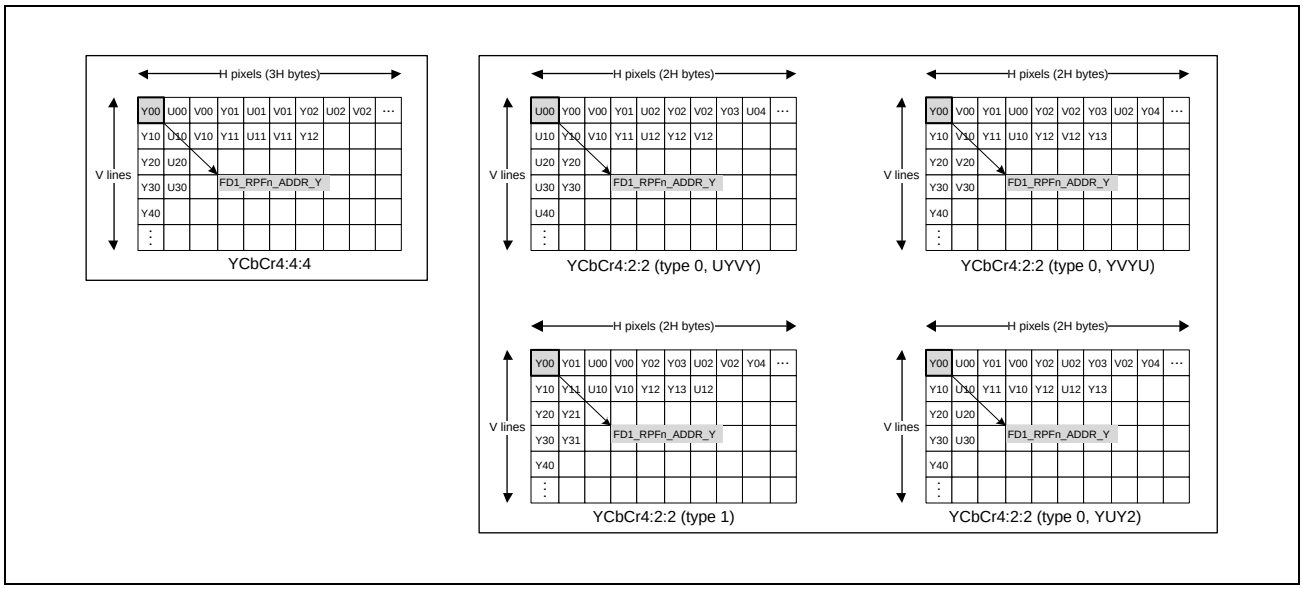


Figure 9.10-11 YcbCr Interleaved Formats

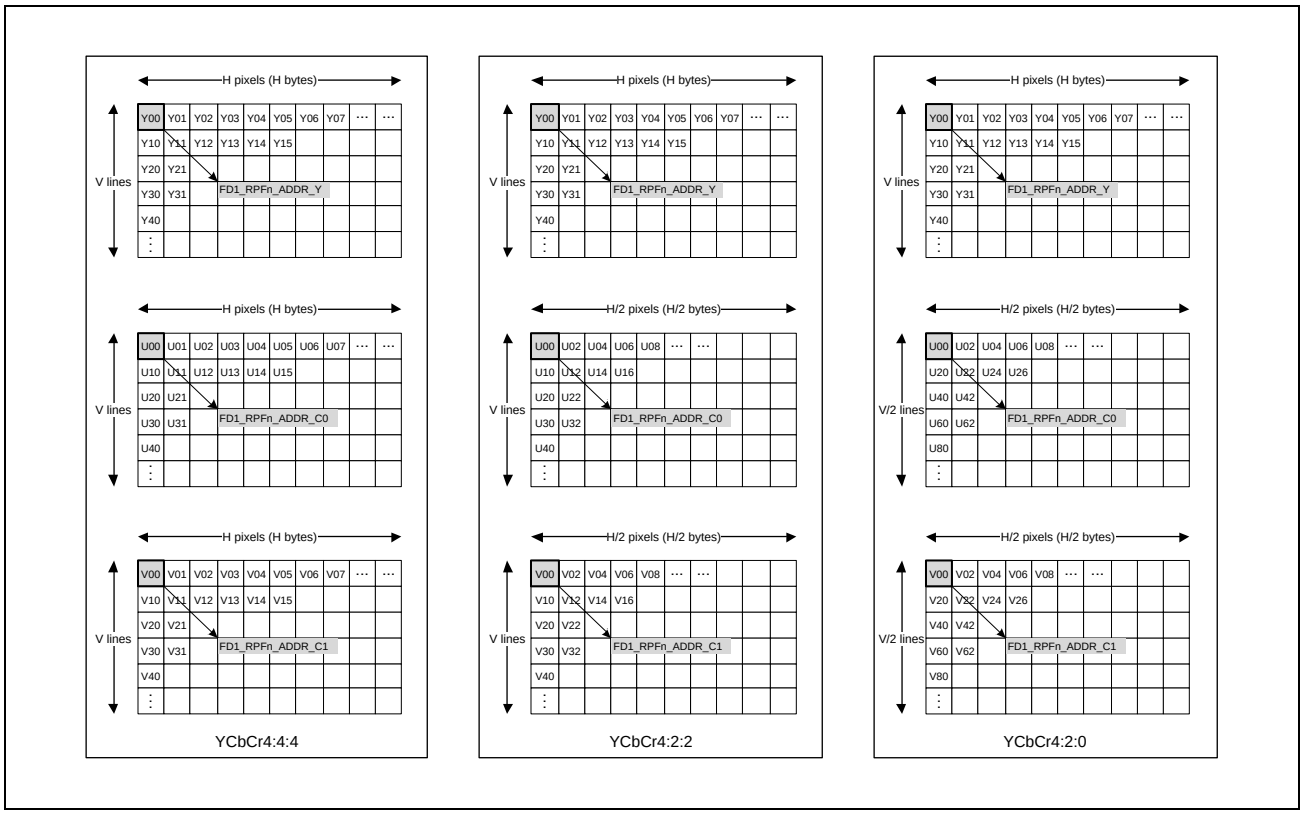


Figure 9.10-12 YcbCr Planar Formats

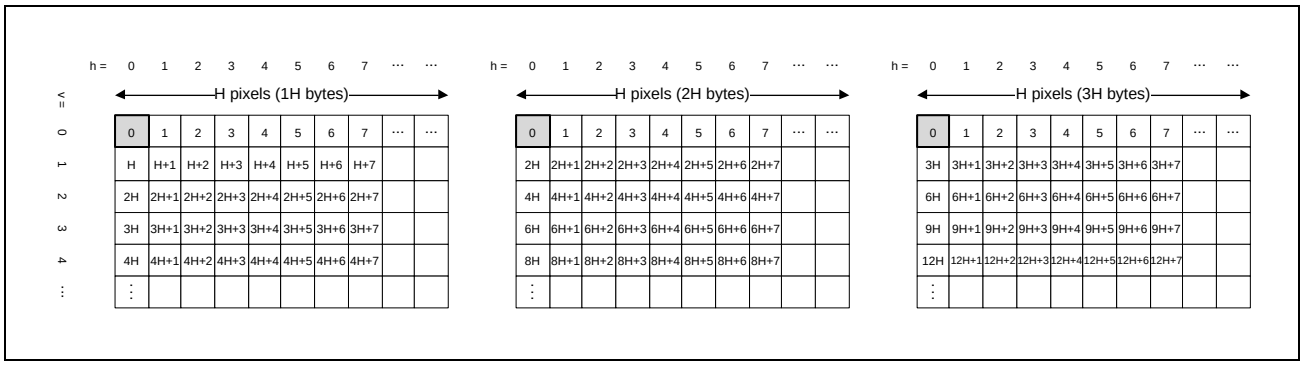


Figure 9.10-13 Memory Address for the Corresponding Pixel's Position

Legend of **Figure 9.10-13**, H is horizontal image size in pixel unit. 1H, 2H and 3H is corresponding to the case 1 pixel has 1 byte, 2 bytes, 3 bytes of data respectively.

(3) Source Picture Stride Register (FD1_RPF1_PSTRIDE)

Access Size : 32 bits
Address : <FDP1_base> + 0068h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRC_STRD_Y[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRC_STRD_C[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	SRC_STRD_Y [15:0]	0000h	RW	Memory Stride of Source Picture Y Plane These bits specify in 1-byte units the memory stride of the source picture Y plane read by the RPF1). A value from 0020h to 1FFEh can be specified.
15 to 0	SRC_STRD_C [15:0]	0000h	RW	Memory Stride of Source Picture C Plane These bits specify in 1-byte units the memory stride of the source picture C plane read by the RPF1. A value from 0020h to 1FFEh can be specified. In the YCbCr planar format, this setting is used as the memory stride of the Cb and Cr planes.

(4) RPF1 Source Component-m Address Register (FD1_RPF1_ADDR_m) (m = Y, C0, C1)

Access Size : 32 bits

Address : <FDP1_base> + 0078h / 007Ch / 0080h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PSRC_ADDR_m[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PSRC_ADDR_m[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PSRC_ADDR_m[31:0]	00000000h	RW	RPF1 Source Field Address for Component-m (m = Y,C0, C1) These bits specify in 1-byte units the start address of the source component-m plane read by the RPF1. A value from 0000_0000h to FFFF_FFFFh can be specified. Set same value with LSB 4bit of RPF1 source field address of component C0 to LSB 4bit of RPF1 source field address of component C1, when input image format is Planar(FD1_RPF_FORMAT.RDFMT = 74,75,76).

(5) RPF Data Swap Register (FD1_RPF1_SWAP)

Access Size : 32 bits
Address : <FDP1_base> + 0094h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	ISWAP[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	ISWAP[3:0]	0h	RW	Input Swap Setting for Image Data This bit field specifies the data swapping function. Each bit corresponds to the following data swapping. Bit[3]: Double Word (64-bit) Swap Bit[2]: Word (32-bit) Swap Bit[1]: Half Word (16-bit) Swap Bit[0]: Byte (8-bit) Swap

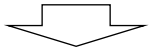
When data order in memory for each format is defined as same as **Figure 9.10-10** to **Figure 9.10-12** set 1111b to ISWAP[3:0]. If data order in memory is different from definitions of these figures, it can be adjusted to definitions of these figures in 16 bytes unit as shown in **Table 9.10-6**.

NOTE

Follow **9.10.2 FCPF** about setting this swap register.

Table 9.10-6 Changing Data Order According to Swap Register

Data Order in Memory																	ISWAP[3:0]/OSWAP[3:0]			
Byte Address	16n+0	16n+1	16n+2	16n+3	16n+4	16n+5	16n+6	16n+7	16n+8	16n+9	16n+10	16n+11	16n+12	16n+13	16n+14	16n+15	BIT3	BIT2	BIT1	BIT0
Data	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	1	1	1	1
	1	0	3	2	5	4	7	6	9	8	11	10	13	12	15	14	1	1	1	0
	2	3	0	1	6	7	4	5	10	11	8	9	14	15	12	13	1	1	0	1
	3	2	1	0	7	6	5	4	11	10	9	8	15	14	13	12	1	1	0	0
	4	5	6	7	0	1	2	3	12	13	14	15	8	9	10	11	1	0	1	1
	5	4	7	6	1	0	3	2	13	12	15	14	9	8	11	10	1	0	1	0
	6	7	4	5	2	3	0	1	14	15	12	13	10	11	8	9	1	0	0	1
	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	1	0	0	0
	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	0	1	1	1
	9	8	11	10	13	12	15	14	1	0	3	2	5	4	7	6	0	1	1	0
	10	11	8	9	14	15	12	13	2	3	0	1	6	7	4	5	0	1	0	1
	11	10	9	8	15	14	13	12	3	2	1	0	7	6	5	4	0	1	0	0
	12	13	14	15	8	9	10	11	4	5	6	7	0	1	2	3	0	0	1	1
	13	12	15	14	9	8	11	10	5	4	7	6	1	0	3	2	0	0	1	0
	14	15	12	13	10	11	8	9	6	7	4	5	2	3	0	1	0	0	0	1
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	0	0	0	0



Data Order Defined in **Table 9.10-7** for RGB Format and **Figure 9.10-10** to **Figure 9.10-12** for YCbCr Format																
Byte Address	16n+0	16n+1	16n+2	16n+3	16n+4	16n+5	16n+6	16n+7	16n+8	16n+9	16n+10	16n+11	16n+12	16n+13	16n+14	16n+15
Data	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

9.10.1.4.8 WPF Control Registers

(1) Destination Picture Format Register (FD1_WPF_FORMAT)

Access Size : 32 bits
Address : <FDP1_base> + 00C0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PDV[7:0]								-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	R	R	RW	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WSPYCS	WSPUVS	DITH[1:0]		WRTM[2:0]			CSC	-	WRFMT[6:0]						
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	R	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	PDV[7:0]	00h	RW	PAD Value in Output Packed Data These bits specify the value to be stored in the bit field indicated as PAD or P in the output formats shown in **Table 9.10-7**. A value from 0 to 255 can be specified. Note: Case that PAD data size less than 8-bit, fill the most significant bit for the empty higher of bits. When the PAD is 1 bit: Output the MSB bit (PDV[7]) When the PAD is 4 bits: Output the upper 4 bits (PDV[7:4]) When the PAD is 8 bits: Output 8 bits (PDV[7:0])
23 to 21	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	-	0h	RW	Reserved When read, the initial value is read. When writing, be sure to write the initial value. Operation is not guaranteed if a value other than the initial value is written.
19 to 16	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	WSPYCS	0h	RW	WPF Output Mode Setting 1 When the output format is YUY2, set this bit to 1 and set the WRFMT bits to 71 (47h). When the output format is YVYU, set this bit and the WSPUVS bit to 1 and set the WRFMT bits to 71 (47h). In other cases, set this bit to 0.
14	WSPUVS	0h	RW	WPF Output Mode Setting 2 When the output format is NV61, set this bit to 1 and set the WRFMT bits to 65 (41h). When the output format is NV21, set this bit to 1 and set the WRFMT bits to 66 (42h). When the output format is YVYU, set this bit and the WSPYCS bit to 1 and set the WRFMT bits to 71 (47h). In other cases, set this bit to 0.
13, 12	DITH[1:0]	0h	RW	Dithering Enable/Disable When the output format specified through the WRFMT bits is RGB with 18 bpp (262144 colors) or less, the color reduction processing is applied to match the number of colors. The color reduction processing may generate the artifacts of pseudo gradation, which can be suppressed through dithering. The DITH bits enable or disable dithering during color reduction. When the output format specified through the WRFMT bits is YCbCr, specify 0 in these bits. 0: Dithering is disabled 3: Dithering is enabled 1, 2: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
11 to 9	WRTM[2:0]	0h	RW	CSC Conversion Expression Setting These bits select the expression for color space conversion. The conversion direction is RGB to YCbCr when the format specified in the WRFMT bits is RGB, or YCbCr to RGB when the format is YCbCr. 0: BT.601 YCbCr [16,235/240] → RGB [0,255] 1: BT.601 YCbCr [0,255] → RGB [0,255] 2: BT.709 YCbCr [16,235/240] → RGB [0,255] 3: BT.709 YCbCr [16,235/240] → RGB [16,235] 4 to 7: Setting prohibited
8	CSC	0h	RW	Color Space Conversion Setting Enables or disables YCbCr ↔ RGB color space conversion to be executed in the WPF. The characteristics of color space conversion are determined by the WRTM setting. 0: Color space is not converted. 1: Color space is converted.
7	-	0h	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	WRFMT[6:0]	00h	RW	WPF Output Image Format Setting These bits select the format of the image output from the WPF to the external memory from among those listed in **Table 9.10-7** and **Table 9.10-5**. [Note 1] Number of output pixels When YCbCr4:2:2 is specified through WRFMT, the horizontal size of the output image should be a multiple of 2 pixels. When YCbCr4:2:0 is specified, the vertical and horizontal sizes of the output image should be multiples of 2 pixels. [Note 2] Output chroma horizontal down-sampling When the output format is YCbCr4:2:2 or YCbCr4:2:0 the chroma component is down-sampled horizontally from YCbCr4:4:4 (the format of image process result) by averaging the values of neighbor two pixels. [Note 3] Output chroma vertical down-sampling When the output format is YCbCr4:2:0 the chroma component is down-sampled vertically from YCbCr4:4:4 (the format of image process result) by outputting only the even number line (line 0, 2, etc).

Table 9.10-7 Packed RGB Formats for WPF Output (1/2)

WRFMT[6:0]	Bit per Pixel	Phase	Upper Row - Address / Bottom Row - Bit Field																													
			n								n+1								n+2								n+3							
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
00h	8	—	R0	R0	R0	G0	G0	G0	B0	B0	R1	R1	R1	G1	G1	G1	B1	B1	R2	R2	R2	G2	G2	G2	B2	B2	R3	R3	R3	G3	G3	G3	B3	B3
01h	12	—	0	0	0	0	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
02h			R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	0	0	0	0
03h	—	—	Reserved								Reserved								Reserved								Reserved							
04h	15	—	0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
05h			R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	0
06h	16	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
07h	18	—	PAD								0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
08h			R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	PAD							
09h			0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	PAD							
0Ah			PAD								R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0
0Bh			PAD								0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0
0Ch			0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	PAD							
0Dh			PAD								R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0
0Eh			R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	PAD							
0Fh		0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	R1	R1
		1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	0	0	0	0	0	0	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2
		2	G2	G2	B2	B2	B2	B2	B2	B2	0	0	0	0	0	0	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3
10h		0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	R1	R1	R1	R1	R1	R1	G1	G1
		1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	0	0	0	0	0	0	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	B2	B2	B2	B2
		2	B2	B2	0	0	0	0	0	0	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	0	0	0	0	0	0
11h		0	0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1
		1	0	0	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2
		2	0	0	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3
12h		0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1	0	0
		1	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2	0	0
		2	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3	0	0
13h	24	—	PAD								R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
14h			R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	PAD							
15h		0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
		1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
		2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
16h	18	—	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	0	0	0	0	0	0	0	G0	G0	G0	B0	B0	B0	B0	B0	B0
17h			0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
18h	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1
		1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2
		2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3
19h	12	—	PAD				R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	PAD				R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
1Ah		—	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	PAD				R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	PAD			
1Bh	15	—	P	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	P	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
1Ch		—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	P	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	P
1Dh	12	—	PAD				B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	PAD				B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1
1Eh		—	B0	B0	B0	B0	G0	G0	G0	G0	R0	R0	R0	R0	PAD				B1	B1	B1	B1	G1	G1	G1	G1	R1	R1	R1	R1	PAD			
1Fh	15	—	P	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	P	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1
20h		—	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	P	B1	B1	B1	B1	B1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	P
21h	18	0	0	0	B0	B0	B0	B0	B0	B0	0	0	G0	G0	G0	G0	G0	G0	0	0	R0	R0	R0	R0	R0	R0	0	0	B1	B1	B1	B1	B1	B1
		1	0	0	G1	G1	G1	G1	G1	G1	0	0	R1	R1	R1	R1	R1	R1	0	0	B2	B2	B2	B2	B2	B2	0	0	G2	G2	G2	G2	G2	G2
		2	0	0	R2	R2	R2	R2	R2	R2	0	0	B3	B3	B3	B3	B3	B3	0	0	G3	G3	G3	G3	G3	G3	0	0	R3	R3	R3	R3	R3	R3
22h	24	—	PAD								B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0

Table 9.10-7 Packed RGB Formats for WPF Output t (2/2)

			Upper Row - Address / Bottom Row - Bit Field																											
			n								n+1								n+2								n+3							
WRFMT[6:0]	Bit per Pixel	Phase	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
23h	16	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0
24h to 3Fh	—	—	Reserved								Reserved								Reserved								Reserved							

Refer to **Table 9.10-5** for YCbCr formats.

(2) Destination Picture Rounding Control Register (FD1_WPF_RNDCTL)

Access Size : 32 bits
Address : <FDP1_base> + 00C4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	CBRM	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	CLMD[1:0]		-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	RW	RW	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	CBRM	0h	RW	Bit Count Reduction Selection for Data Storage in Packed RGB This bit specifies the method for reducing the number of bits when data is stored in the bit fields indicated as R, G, and B in **Table 9.10-7** and the target bit fields are not eight bits. 0: Bit count conversion: The lower-order bits are truncated 1: Bit count conversion: Rounding (rounding off)
27 to 14	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	CLMD[1:0]	0h	RW	Color Data Clipping These bits specify the method for clipping the YCbCr color data output from the WPF. When RGB color data is output from the WPF, specify 0 in these bits. 00b: Output value is not clipped (0-255) 01b: Output value is clipped: YCbCr mode 1 (16-235 (Y), 16-240 (Cb/Cr)) 10b: Output value is clipped: YCbCr mode 2 (Y/Cb/Cr = 1-254) 11b: Setting prohibited
11 to 0	-	0h	R	Reserved These bits are always read as 0. The write value should always be 0.

(3) Destination Picture Stride Register (FD1_WPF_PSTRIDE)

Access Size : 32 bits
Address : <FDP1_base> + 00C8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DST_STRD_Y[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DST_STRD_C[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	DST_STRD_Y [15:0]	0000h	RW	Memory Stride of Destination Picture Y/RGB Plane These bits specify in 1-byte units the memory stride of the destination picture in the external memory to be written to by the WPF. A value from 0020h to 1FFEh can be specified.
15 to 0	DST_STRD_C [15:0]	0000h	RW	Memory Stride of Destination Picture C Plane These bits specify in 1-byte units the memory stride for the C plane of the destination picture in the external memory to be written to by the WPF. When the WPF outputs images in an RGB format, this setting is not used. When the WPF outputs images in YCbCr planar format, this setting is applied to both the Cb and Cr planes. A value from 0020h to 1FFEh can be specified.

(4) Destination Component-m Address Register (FD1_WPF_ADDR_m) (m = Y, C0, C1)

Access Size : 32 bits

Address : <FDP1_base> + 00CCh / 00D0h / 00D4h

Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DST_ADDR_m[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DST_ADDR_m[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DST_ADDR_m [31:0]	00000000h	RW	Destination Address for Component-m (m = Y, C0, C1) These bits specify in 1-byte units the address for storing the destination component-m plane to be written to the external memory by the WPF. A value from 0000_0000h to FFFF_FFFFh can be specified.

(5) WPF Data Swap Register (FD1_WPF_SWAP)

Access Size : 32 bits
Address : <FDP1_base> + 00D8h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SSWAP[3:0]				OSWAP[3:0]			
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 4	SSWAP[3:0]	0h	RW	Set the same value with FD1_RPF_SWAP.ISWAP.
3 to 0	OSWAP[3:0]	0h	RW	Output Swap Setting for Image Data This bit field specifies the data swapping function. Each bit corresponds to the following data swapping. Bit[3]: Double Word (64-bit) Swap Bit[2]: Word (32-bit) Swap Bit[1]: Half Word (16-bit) Swap Bit[0]: Byte (8-bit) Swap

When data order in memory for each format is defined as same as **Table 9.10-7** for RGB format and **Figure 9.10-10** to **Figure 9.10-12** for YCbCr format set 1111b to OSWAP[3:0]. If data order in memory is different from definitions of these table and figures, it can be adjusted to definitions of these table and figures in 16 bytes unit as shown in **Table 9.10-6**.

NOTE

Follow **9.10.2 FCPF** about setting this swap register.

9.10.1.4.9 Other Control Registers

(1) IPC Mode Register (FD1_IPC_MODE)

Access Size : 32 bits
Address : <FDP1_base> + 0100h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	DLI	-	-	-	-	-	DIM[2:0]		
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	RW	R	R	R	R	R	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	DLI	0h	RW	Set the value 1 to this bit.
7 to 3	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	DIM[2:0]	0h	RW	001b: Set the value 001b to this bit as default.

(2) Line Memory Pixel Number Register (FD1_IPC_LMEM)

Access Size : 32 bits
Address : <FDP1_base> + 01E0h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	PNUM[11:0]											
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	PNUM[11:0]	000h	RW	Specify 1024 in case of linear addressing access. In case of tile addressing access, set 960 to this bit. Refer to **9.10.2.5 Operation**.

(3) IP Internal Data (FD1_IP_INTDATA)

Access Size : 32 bits
Address : <FDP1_base> + 0800h
Initial Value : 0202_0202h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IP_INTERNAL_DATA[31:16]															
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP_INTERNAL_DATA[15:0]															
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IP_INTERNAL_DATA[31:0]	02020202h	R	Internal data specifies version of FDP 0202_0202h: FDP1 version of this LSI.

9.10.1.5 Operation

Procedure of FDP1B Start and Stop operations are described for Register usage and Register settings.

9.10.1.5.1 Register Usages for FDP1B Start and Stop

Registers FD1_CTL_CMD, FD1_CTL_SGCMD, and FD1_CTL_REGEND must be used correctly in the procedures of FDP1B start and stop as specified in **Table 9.10-8** and **Table 9.10-8**.

Table 9.10-8 Register Usages for Starting FDP1B

Register	Start the First Frame (FR[0])	Start the Second or Later Frame (FR[f], f = 1, 2, 3…)
FD1_CTL_SGCMD.SGEN	After all registers have been set completely, it must be set as 1 to generate V-Interruption.	It must be kept as 1 from the previous frame FR [f-1].
FD1_CTL_REGEND.REGEND	After all registers except FD1_CTL_SGCMD.SGEN have been set completely, it must be set 1. As the results, at the first V-interruption, all V-update registers are reflected for hardware reference, FD1_CTL_CMD.STRCMD is referred to start the first frame.	After V-interruption of starting the previous frame FR [f-1], it must be set 1 after all registers about FR[f] have been set completely. As the results, at the V-interruption of the end of frame FR [f-1], the V-update registers will be reflected to hardware reference, FD1_CTL_CMD.STRCMD is referred to start the frame FR[f].
FD1_CTL_CMD.STRCMD	It must be set 1 when all registers except FD1_CTL_SGCMD.SGEN, FD1_CTL_REGEND.REGEND are set. At the first V-Interruption it is referred by hardware to start the frame.	It must be set 1 when all registers except FD1_CTL_REGEND.REGEND are set at the next V-Interruption (boundary between FR [t-1] and FR[t]) it is referred by hardware to start the frame.

Table 9.10-9 Register Usages for Stopping FDP1B

Register	Stopping Immediately by Software Reset	Stopping after Current Process Finished
FD1_CTL_SGCMD.SGEN	It must be cleared before setting the value of FD1_CTL_SRESET.SRST as 1 so that no more V-Interruption after FDP1B stopped.	After V-interruption at the final frame end (FD1_CTL_IRQSTA.FRE is set as 1) occurred, it must be cleared so that no more V-Interruption is generated after FDP1B stopped.
FD1_CTL_REGEND.REGEND	—	1) No Interrupt Mode After V-interruption of starting the final frame, it must be set 1 right after clearing FD1_CTL_CMD.STRCMD so that the final V-interruption occurs and operation status of the final frame is updated to V-update status registers for Software reference. 2) Other modes Right after V-interruption at starting the final frame, it must be set 1 right after clearing FD1_CTL_CMD.STRCMD so that operation status of the final frame is updated to V-update status registers for Software reference.
FD1_CTL_CMD.STRCMD	After FDP1B stopped successfully (FD1_CTL_IRQSTA.FRE is set as 1), it must be cleared to prevent wrong operation when restart FDP1B.	After V-interruption of starting the final frame, it must be cleared to prevent wrong operation when restart FDP1B.

9.10.1.5.2 Register Setting Order for FDP1B Start and Stop

(1) Interrupt mode or best effort mode

(1a) To start the first frame, set registers in following order.

1. Set 1 to FD1_CTL_CLKCTRL.CSTP_N
2. Set registers other than FD1_CTL_CMD, FD1_CTL_SGCMD, and FD1_CTL_REGEND.
3. Set FD1_CTL_CMD.STRCMD to the value 1. Set FD1_CTL_REGEND.REGEND to the value 1.
4. Set FD1_CTL_SGCMD.SGEN to the value 1, then FDP1 starts its process. FD1_CTL_SGCMD.SGEN must be not cleared if continue to the second or later frames.

(1b) To start the second or later frame, set registers in following order.

1. After V-interruption of the previous frame start timing occurred, set registers other than FD1_CTL_CMD and FD1_CTL_REGEND.
2. Set FD1_CTL_CMD.STRCMD to the value 1.
3. Set FD1_CTL_REGEND.REGEND to the value 1.
4. At the V-interruption of the previous frame end timing, FDP1B starts the target frame. The V-interruption of the previous frame end timing and the one of the target frame start timing are the same.

(1c) To stop FDP1B operation after current frame finished, set registers as the following.

1. After V-interruption of the current frame start timing occurred, set FD1_CTL_CMD.STRCMD to the value 0.
2. Set FD1_CTL_REGEND.REGEND to the value 1. If this register was completed before the V-Interruption of frame end timing, FDP1B stops its operation and current frame becomes the last frame of the image sequence. If these register setting could not be completed before the V-Interruption of frame end timing, FDP1B stops its operation in the next V-interruption. In the latter case, V-update status registers are not updated at V-interruption of frame end but at the next V-interruption at which FD1_CTL_REGEND.REGEND is set as 1.
3. Set FD1_CTL_SGCMD.SGEN to the value 0, then FDP1B will stop generating V-interruption.

(1d) To stop FDP1B operation immediately, set registers as the following.

1. Set FD1_CTL_SGCMD.SGEN to the value 0, then FDP1B will stop generating V-interruption.
2. Set FD1_CTL_SRESET.SRST to the value 1. FDP1B will invoke termination process immediately.
3. Wait frame end interrupt from FDP1B or until the register bit FD1_CTL_IRQSTA.FRE is set to 1. After it occurs, FDP1B have finished its processing. With this operation, FDP1B can stop its process quickly, but the output frame of the last frame is corrupted.

(2) No interrupt mode

(2a) To start the first frame, set registers in following order.

Same as (1a).

(2b) To start the second or later frame, set registers in following order.

Same as (1b).

(2c) To stop FDP1B operation after current frame finished, set registers as the following.

1. After V-interruption of the current frame start timing occurred, set FD1_CTL_CMD.STRCMD to the value 0.
2. Set FD1_CTL_REGEND.REGEND to the value 1. At the V-Interruption of frame end timing, FDP1B stops its operation and current frame becomes the last frame of the image sequence.
3. Set FD1_CTL_SGCMD.SGEN to the value 0, then FDP1B will stop generating V-interruption.

(2d) To stop FDP1B operation immediately, set registers as the following.

Same as (1d).

9.10.1.6 Usage Notes

9.10.1.6.1 Assignment in Memory Space

Make sure that FDP1B memory space shall be mapped to Non-Cache region.

9.10.1.6.2 FCPF Connection

Bus access of FDP is carried out through FCPF module in this LSI. So, refer to also **9.10.2 FCPF** in this manual.

There are some notices in FCPF connection system below.

(1) FDP1B must be started, stopped in companion with FCPF as guided in **9.10.2.5 Operation**.

(2) There are some restrictions about register setting in FCPF connection system. Refer to **9.10.2 FCPF** about restrictions of register setting.

9.10.1.6.3 Manipulate FDP with Software Reset of CPG

Please refer to Usage Notes of **9.10.2.6.1 Manipulating FDP with CPG Software Reset** about manipulating FDP with CPG software reset.

9.10.1.6.4 Terminology

(1) Definition of operators and functions

The following operators, notations are defined for explaining the functions of FDP1B.

clip0 (*x*)

Clip to 0 value if x is less than 0. [$x = (x < 0)? 0: x$]

clip3 (min, max, *x*)

The value *x* shall be clipped so that *x* shall be min < *x* < max. [$x = (x < \text{min})? \text{min}: ((x > \text{max})? \text{max}:)$]

(2) Unit

The unit is defined here for explaining FDP1B's functions.

[bpp]: Bits per pixel

FDP1B can handle a number of image formats. Each image format has different number of data bits. The unit [bpp] shows number of data bits for one pixel. For example, RGB 8 bpp means that its format is RGB and the pixel consists of 8 data bits.

9.10.2 FCPF

The FCP is a companion module of a video processing module in this LSI. It provides data caching, and converting of AXI transaction in order to reduce the memory bandwidth.

The FCPF is a module that reads tile addressing image data or linear addressing image data from external memory, converts it to linear addressing image data, and transfers the data to FDP1B. The FCPF also outputs images processed by FDP1B in linear addressing image data to external memory. FCPF access with 256 bits AXI bus for the FDP1B. This module is under control of software for the FDP with the control function of the FCPF executed on the host CPU.

9.10.2.1 Features

The FCPF has the following features.

- Support 3840 × 3840 resolution
- Support converting linear addressing access from FDP to tile addressing image
- Support out-of-order for the whole outstanding transactions

Table 9.10-10 Main Function of the FCPF

Item	Description
Picture size supported	Tile addressing image: Min.: 32 pixels × 32 lines Max.: Horizontal 3840 pixels × vertical 3840 lines Others: Support all picture sizes specified in **9.10.1 FDP1B** (**9.10.1.1 Features**)
Color format	Tile addressing image: YCbCr420 semi-planar Support all formats specified I in **9.10.1 FDP1B** (**9.10.1.4.7 (2) Source Picture Format Register (FD1_RPF1_FORMAT)** and **9.10.1.4.8 (1) Destination Picture Format Register (FD1_WPF_FORMAT)**)
AXI transaction	• Read access: Support out-of-order for all outstanding transactions Support data interleaving Tile/linear address conversion (format YCbCr420 semi-planar only) • Write access: Support out-of-order for all outstanding transactions

9.10.2.2 Block Diagram

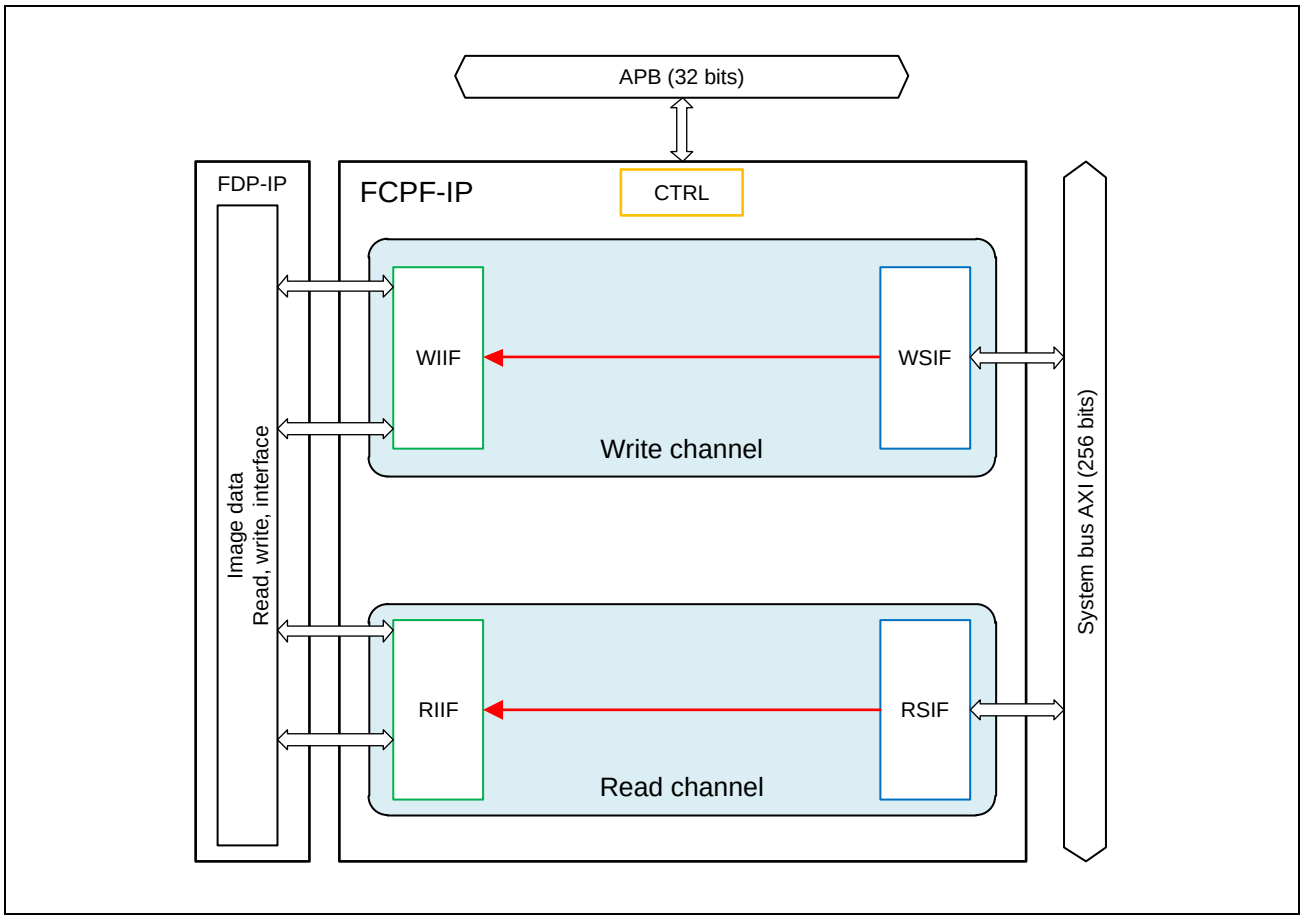


Figure 9.10-14 Overall Configuration of the FCPF

Table 9.10-11 Sub-module Description

Abbr.	Description
WIIF	Connected IP write channel interface
WSIF	System AXI write channel interface
CTRL	FCPF controller

Abbr.	Description
RIIF	Connected IP read channel interface
RSIF	System AXI read channel interface

9.10.2.2.1 IP Set and Bus Access Functions

FCPF works in companion with other IPs (FDP) in an IP set. As shown in **Table 9.10-12** the bus access function conducted by FCPF is different according to each IP set.

Table 9.10-12 Bus Access Function for Each IP Set

IP Set	Bus Access
FDP + FCPF	Following functions are conducted by FCPF • Read linear addressing image data • Read tile addressing image data • Write image data

9.10.2.3 Register Configuration

The FCPF contains the registers which can directly be read and written to by the host CPU. The host can access any word (32 bits) in the registers. Therefore, read-modify-write is needed to change partial bits in 32 bits register.

There are 2 types of registers in FCPF: Immediate register, V-update Register. The values of immediate registers are reflected to H/W behavior immediately after changing its values. Hence, do not change these immediate registers while FCPF is processing (FCP_STA.ACT is set to 1) unless this manual allows it explicitly. The values of V-update registers are reflected to H/W behavior at the next V-interruption of FDP. For FDP connection case users can set V-update registers of the next frame/field while FDP + FCPF is processing the current frame/field.

Refer to **9.10.2.5.1 Operation Control Setting for the IP Set FDP** to grasp the flow of setting registers of FCPF and connected IP.

The grayed portions represent reserved areas for debugging or future functional expansion. Do not access the reserved areas. Never access the registers when the FCPF is in the operation clock placed in hold status.

The following are the base addresses of the FCPF channel.

FCPF: 0_1631_0000h (5631_0000h*1, 4631_0000h*2)

Note 1. CM33 address space (non-secure)
Note 2. CM33 address space (secure)

Table 9.10-13 FCPF Registers

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
FCP version control register	FCPF0_FCP_VCR	0000_0202h	0000h	32
Reserve	-	-	0004h to 0017h	-
FCP status register	FCPF0_FCP_STA	0000_0000h	0018h	32
Reserve	-	-	001Ch to 006Fh	-
FCP Tile/Linear address conversion control	FCPF0_FCP_TL_CTRL	0000_0000h	0070h	32
Reserve	-	-	0074h to 00C3h	-
FCP picture information register 1	FCPF0_FCP_PICINFO1	0000_0000h	00C4h	32
Reserve	-	-	00C8h to 0113h	-
FCP base address of reference picture of plane Y	FCPF0_FCP_BA_REF_Y	0000_0000h	0114h	32
Reserve	-	-	0118h to 011Bh	-
FCP base address of reference picture of plane C	FCPF0_FCP_BA_REF_C	0000_0000h	011Ch	32

Remarks Immediate register: FCPF0_FCP_VCR, FCPF0_FCP_STA
V-update register: FCPF0_FCP_TL_CTRL, FCPF0_FCP_PICINFO1, FCPF0_FCP_BA_REF_Y, FCPF0_FCP_BA_REF_C

9.10.2.4 Register Descriptions

9.10.2.4.1 FCP Version Control Register (FCP_VCR)

FCP_VCR is a version control register of the FCPF. The value read is fixed.

Access Size : 32 bits
Address : <FCPF0_base> + 0000h
Initial Value : 0000_0202h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CATEGORY[7:0]								REVISION[7:0]							
Initial Value	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	-	All 0	R	Reserved These bits are read as 0b. The written value will be ignored.
15 to 8	CATEGORY [7:0]	2h	R	FCP use case category. 02h: This LSI
7 to 0	REVISION[7:0]	2h	R	Version of LSI product 02h: FCP revision of this LSI

9.10.2.4.2 FCP Status Register (FCP_STA)

FCP_STA indicates the FCPF status.

Access Size : 32 bits
Address : <FCPF0_base> + 0018h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ACT
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	-	All 0	R	Reserved These bits are always read as 0 or 1. Do not care about this value. The write value should always be 0.
0	ACT	0h	R	FCPF Active Flag 0: Indicates the FCPF is not active. 1: Indicates the FCPF is active.

9.10.2.4.3 FCP Tile/Linear Control Register (FCP_TL_CTRL)

FCP_TL_CTRL specifies the control of Tile/Linear address conversion on FCPF. Refer to **9.10.2.5.4 (2) Tile/linear addressing conversion** to grasp how to set registers when reading a tile addressing image.

Access Size : 32 bits
Address : <FCPF0_base> + 0070h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TLEN	-	-	VPOS_C[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VPOS_Y[12:0]												
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	TLEN	0h	RW	Tile Linear Enable 0: Disable Tile/Linear address conversion 1: Enable Tile/Linear address conversion
30, 29	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 16	VPOS_C[12:0]	0h	RW	Vertical POSition of Chroma Vertical position of tile address chroma plane which FDP start to read. This register must be set value from 0 to FFEh.
15 to 13	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	VPOS_Y[12:0]	0h	RW	Vertical POSition of Y plane Vertical position of tile address luma plane which FDP start to read. This register must be set value from 0 to 1FFDh.

NOTE

This register is only effective for the Tile data. VPOS_C value should be set the half of the VPOS_Y value.

9.10.2.4.4 FCP Picture Information 1 (FCP_PICINFO1)

FCP_PICINFO1 specifies the picture information of the tile address image. Refer to **9.10.2.5.4 (2) Tile/linear addressing conversion** to grasp how to set registers when reading a tile addressing image.

Access Size : 32 bits
Address : <FCPF0_base> + 00C4h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	STRIDE_DIV16[10:0]										
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	-	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	STRIDE_DIV16 [10:0]	0h	RW	STRIDE DIVided by 16 Memory stride of the tile addressing image. Common for all luma and chroma planes Y and C. It must be specified so that STRIDE_DIV16*16 is exponent of 2 and larger than or equal to 128. STRIDE_DIV16*16 must be the same with memory stride of Y plane, C plane set by FDP registers. This bit must get value from 8h to 200h.

9.10.2.4.5 FCP Base Address Reference picture of plane m (FCP_BA_REF_m) (m = Y, C)

FCP_BA_REF_m specifies the base address of the reference picture of the plane Y, C which is located in tile addressing. Those registers are different from the source address registers set in FDP. Refer to **9.10.2.5.4 (1) Tile addressing image data** to grasp the way of setting address registers.

Access Size : 32 bits
Address : <FCPF0_base> + 0114h / 011Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BA[31:16]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BA[15:0]															
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BA[31:0]	0h	RW	Base Address of reference picture The base address of the tile addressing image of the plane Y, C. It must be specified in 16384 bytes unit. In case of FDP connection Y are corresponding to the luma plane of previous, current, next field (frame in case of progressive input) respectively. C is corresponding to the chroma plane of the current field (frame). This bit must be set value from 0 to FFFF_4000h

9.10.2.5 Operation

This section explains how to control operation of an IP set which includes FCPF.

9.10.2.5.1 Operation Control Setting for the IP Set FDP

(1) The case FDP in interrupt mode or best effort mode

(1a) To start the first frame, set registers of FDP, FCPF as following order.

1. Set value 1 to register FD1_CTL_CLKCTRL.CSTP_N
2. Set FDP registers other than FD1_CTL_CMD, FD1_CTL_SGCMD, and FD1_CTL_REGEND. FDP registers must follow restriction at **9.10.2.5.3 Register Settings and Restrictions on Connected IP (FDP)**.
3. Set FCPF registers (V-update registers and other registers if necessary).
   In case of reading tile addressing image set 1 to FCP_TL_CTRL.TLEN.
4. Set FD1_CTL_CMD.STRCMD to the value 1. Set FD1_CTL_REGEND.REGEND to the value 1.
5. Set FD1_CTL_SGCMD.SGEN to the value 1, then FDP starts its process. FD1_CTL_SGCMD.SGEN must be not cleared if continue to the second or later frames.

(1b) To start the second or later frame, set registers in following order.

1. After V-interruption of the previous frame start timing occurred, set all FDP registers other than FD1_CTL_CMD and FD1_CTL_REGEND. FDP registers must follow restriction at **9.10.2.5.3 Register Settings and Restrictions on Connected IP (FDP)**.
2. Set FCPF V-update registers.
   In case of reading tile addressing image set 1 to FCP_TL_CTRL.TLEN.
3. Set FD1_CTL_CMD.STRCMD to the value 1.
4. Set FD1_CTL_REGEND.REGEND to the value 1.
5. At the V-interruption of the previous frame end timing, FDP starts the target frame. The V-interruption of the previous frame end timing and the one of the target frame start timing are the same.

(1c) to stop FDP, FCPF operation after current frame finished, set registers as the following.

1. After V-interruption of the current frame start timing occurred, set FD1_CTL_CMD.STRCMD to the value 0.
2. Set FD1_CTL_REGEND.REGEND to the value 1. If this register was completed before the V-Interruption of frame end timing, FDP stops its operation and current frame becomes the last frame of the image sequence. If these register setting could not be completed before the V-Interruption of frame end timing, FDP stops its operation in the next V-interruption. In the latter case, V-update status registers are not updated at V-interruption of frame end but at the next V-interruption at which FD1_CTL_REGEND.REGEND is set as 1.
3. Set FD1_CTL_SGCMD.SGEN to the value 0, then FDP will stop generating V-interruption.

(1d) To stop FDP operation immediately, set registers as the following.

1. Set FD1_CTL_SGCMD.SGEN to the value 0, then FDP will stop generating V-interruption.
2. Set FD1_CTL_SRESET.SRST to the value 1. FDP will invoke termination process immediately.
3. Wait frame end interrupt from FDP or until the register bit FD1_CTL_IRQSTA.FRE is set to 1. After it occurs, both FDP and FCPF have finished its processing.
4. Set CPG_RST_18.{RSTB4, RSTB5, RSTB6} to the value 1. FCPF will invoke termination process immediately.

5. Wait until ACT bit of the FCP_STA register become 0. After it occurs, FCPF has finished its processing.

With this operation, FDP can stop its process quickly, but the output frame of the last frame is corrupted.

(2) The case FDP in no interrupt mode

(2a) To start the first frame, set registers in following order.

Same as (1a).

(2b) To start the second or later frame, set registers in following order.

Same as (1b).

(2c) To stop FDP operation after current frame finished, set registers as the following.

1. After V-interruption of the current frame start timing occurred, set FD1_CTL_CMD.STRCMD to the value 0.
2. Set FD1_CTL_REGEND.REGEND to the value 1. At the V-Interruption of frame end timing, FDP stops its operation and current frame becomes the last frame of the image sequence.
3. Set FD1_CTL_SGCMD.SGEN to the value 0, then FDP will stop generating V-interruption.

(2d) To stop FDP operation immediately, set registers as the following.

Same as (1d).

9.10.2.5.2 Reset Operation

FCPF reset can be conducted by software reset and power on reset. The software reset which is controlled by software, is used to stop the FCPF operation. The software reset operation is not executed immediately; it is executed on completion of the bus transaction. The other is power on reset which initialize the whole FCPF logic immediately.

(1) Software reset

The procedure of software reset of the FCPF is as following:

1. Activate software reset sequence
   Set CPG_RST_18.{RSTB4, RSTB5, RSTB6} to the value 1.
2. Confirm software reset sequence finished:
   FCP_STA.ACT becomes 0. All remain transactions with the bus system finished.

The procedure to stop operation of an IP set including FCPF is descripted at **9.10.2.5.1 Operation Control Setting for the IP Set FDP**.

(2) Power-on reset

The power-on reset is controlled by an externally input signal called hardware reset. This reset initializes the whole internal logic of FCPF.

9.10.2.5.3 Register Settings and Restrictions on Connected IP (FDP)

The connected FDP has to follow the setting and restrictions in following tables.

(1) Restrictions on FDP read side

In case of reading tile addressing image (FCP_TL_CTRL.TLEN = 1), the setting of input data swap (FD1_RPF1_SWAP.ISWAP [3:0]), line memory pixel number (FD1_RPF1_LMEM.PNUM [11:0]), read color format (FD1_RPF1_FORMAT.RDFMT [6:0]), read memory stride (FD1_RPF1_PSTRIDE.SRC_STRD_Y [15:0], FD1_RPF1_PSTRIDE.SRC_STRD_C [15:0]) must follow the **Table 9.10-14**.

Table 9.10-14 Setting Restriction on FDP Read Side

Register	Value
FD1_RPF1_SWAP.ISWAP[3:0]	Fh
Max value of FD1_RPF1_LMEM.PNUM[11:0]	960
FD1_RPF1_FORMAT.RDFMT[6:0]	42h (YCbCr420 semi-planar)
FD1_RPF1_PSTRIDE.SRC_STRD_Y[15:0]	STRIDE*1
FD1_RPF1_PSTRIDE.SRC_STRD_C[15:0]	STRIDE*1

Note 1. STRIDE = FCP_PICINFO1.STRIDE_DIV16 [10:0] × 16

9.10.2.5.4 Decode Data

Decode data consists of tile addressing image data

(1) Tile addressing image data

Image data is located on memory in tile addressing as shown in **Figure 9.10-15**. Although the valid image data exists in the green area only, the tile addressing area is extended to horizontal and vertical direction. The tile size is 128 bytes × 32 lines. The stride of allocation memory area is exponent of 2.

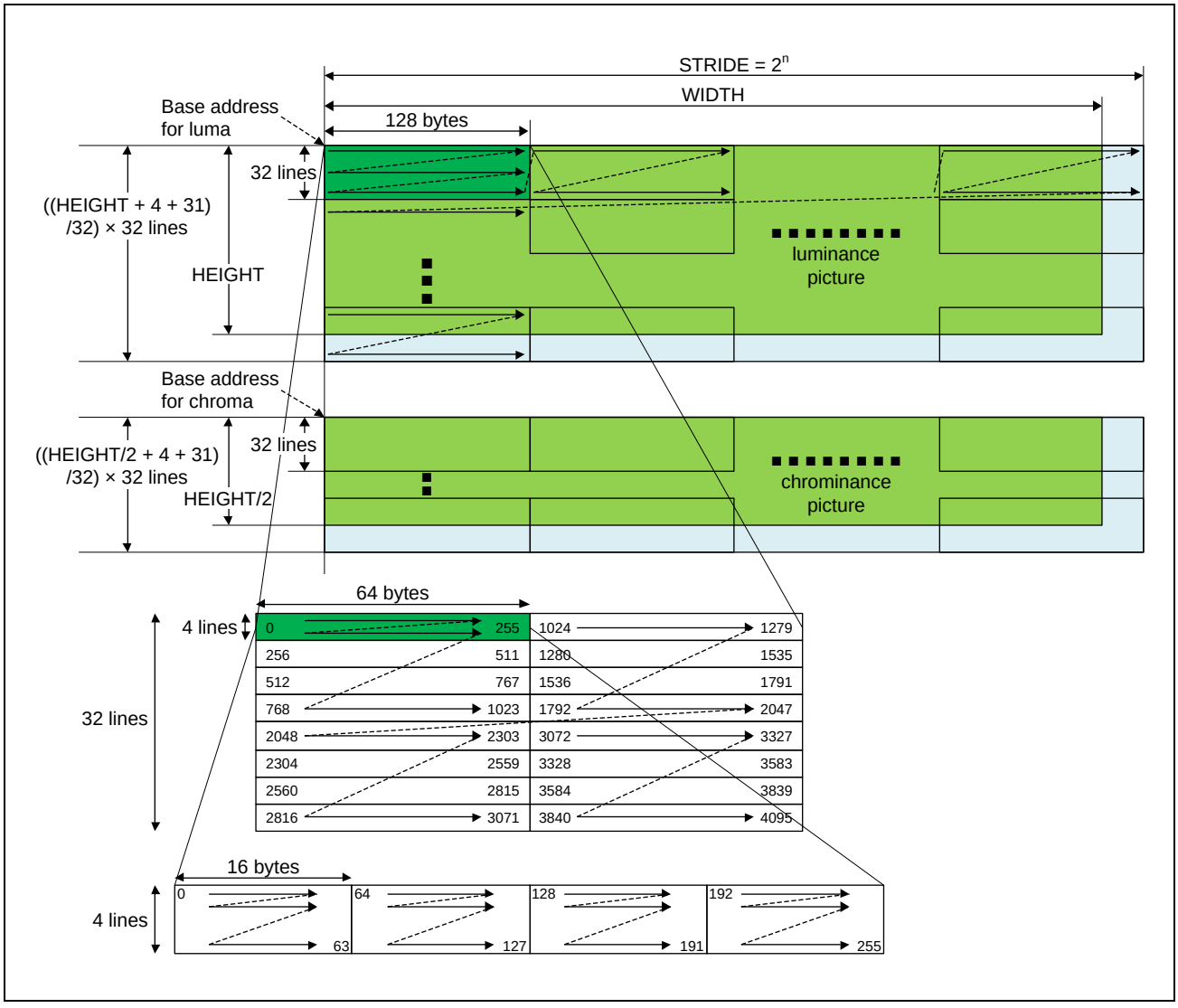


Figure 9.10-15 Memory Allocation of Tile Addressing Image

Parameters in **Figure 9.10-15**:

- STRIDE = FCP_PICINFO1. STRIDE_DIV16 × 16
- HEIGHT, WIDTH: The height and width of the tile addressing image
- Base address for luma: Value of register FCP_BA_REF_Y.BA[31:0]
- Base address for chroma: Value of register FCP_BA_REF_C.BA[31:0]

The buffer size needed for a tile addressing image is shown in **Table 9.10-15**.

Table 9.10-15 Data Size of a Tile Addressing Image

Number of Pic	Data Size per Picture
Maximum number of frames	Luminance: STRIDE × CEIL32 (BLOCK_HEIGHT + 4) bytes*1 Chrominance: STRIDE × CEIL32 (BLOCK_HEIGHT/2 + 4) bytes where STRIDE = FCP_PICINFO1. STRIDE_DIV16 × 16 BLOCK_HEIGHT = pixel height of picture in the size of MB or CTB boundary [pixel height of picture in the size of MB or CTB boundary] MB: CEIL16 (pixel height) CtbSizeY = 16: CEIL16 (pixel height) CtbSizeY = 32: CEIL32 (pixel height) CtbSizeY = 64: CEIL64 (pixel height)

Note 1. CEILn (a) = ((a + (n – 1)) / n) × n

(2) Tile/linear addressing conversion

In order to enable the FDP which supports only linear addressing to access the tile addressing image data, the software needs to set the registers following to **Figure 9.10-16**. Note that the value of VPOS_{Y/C} is set basing on base address regardless of 2 lines of line-shift because reading access to an address is converted to the address of 2 lines below.

In order to enable tile/linear addressing conversion, set 1 to the register FCP_TL_CTRL.TLEN.

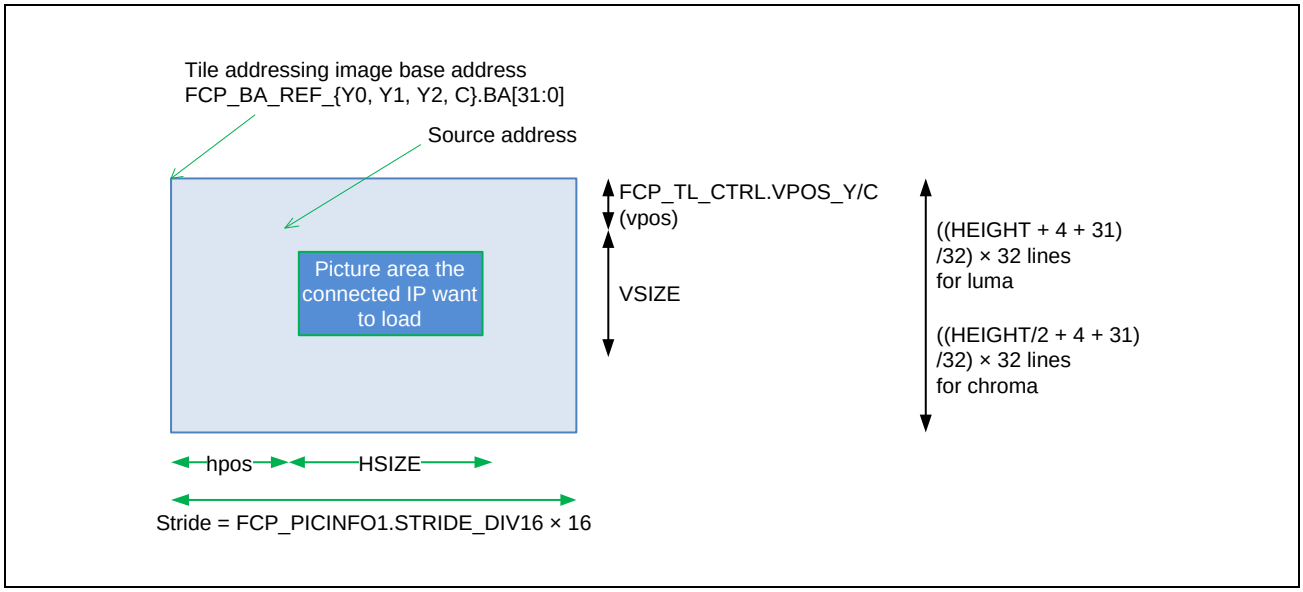


Figure 9.10-16 Reading Tile Addressing Image by Linear Access

Parameters in **Figure 9.10-16**:

- "Source address": FD1_RPF1_ADDR_m.PSRC_ADDR_m[31:0], m = Y,C0
  "Source address" = "Tile addressing image base address" + vpos × stride + hpos
- hpos: the start reading byte position for each line in linear addressing. This value is automatically calculated by FCP and not needed to be set as register.
- VSIZE: FD1_RPF_SIZE.VSIZE[12:0]
- HSIZE: FD1_RPF_SIZE.HSIZE[12:0]

9.10.2.5.5 Operation Status

Operation status of FDP, FCPF is defined for the whole IP set as follows.

(1) If both register bits FD1_CTL_STATUS.BSY and FCP_STA.ACT are 0, both of module FDP, FCPF are stopping.

(2) If either register bit FD1_CTL_STATUS.BSY or FCP_STA.ACT is 1, both of module FDP, FCPF are operating.

9.10.2.6 Usage Notes

9.10.2.6.1 Manipulating FDP with CPG Software Reset

The following notes 1. to 2. are to prevent FDP from the unstable status which may cause hang or wrong operation.

NOTES

1. In the case of stopping FDP using the corresponding register bit of software reset registers (CPG_RST_18) of the module CPG, do following steps.

   Step 1) Stop FDP with the software reset sequence following to the guidance in **9.10.2.5.1 Operation Control Setting for the IP Set FDP**.

   Step 2) Do the following power off sequence b1 to b6.

   b1) Write 0110_1111h to the CPG register (Offset address: CA4h). Transition the Q-ch for all power supply control to Q_STOPPED.

   b2) Read 0330_3333h from the CPG register (Offset address: CA4h). Confirm the Q-ch for all power supply control has transitioned to Q_STOPPED.

   b3) Write 0060_0000h to the CPG register (Offset address: 948h). Assert reset_fcpf_p and reset_fdp1_p.

   b4) Read 0000_006xh from the CPG register (Offset address: A24h). Confirm reset_fcpf_p and reset_fdp1_p asserted.

   b5) Write 0010_0000h to the CPG register (Offset address: 948h). Assert reset_brdg_n.

   b6) Read 0000_007xh from the CPG register (Offset address: A24h). Confirm reset_brdg_n already asserted.

   b7) Power off all the power depending on **10.1.3 Power-On/Off Sequence**.

2. In the case of re-starting FDP after stopping those modules with CPG software reset, do following steps.

   Step 1) Do the following power on sequence a2 to a8.

   a1) Power on all the power depending on **10.1.3 Power-On Sequence**.

   a2) Write 000E_000Eh to the CPG register (Offset address: 66Ch). Supply all clocks.

   a3) Read 0000_0380h from the CPG register (Offset address: 82Ch). Confirm all clocks to started.

   a4) Write 3333_3333h to the CPG register (Offset address: CA4h) Transition all Q-ch for all Q_RUN.

   a5) Write 0010_0010h to the CPG register (Offset address: 948h). Negate the reset_brdg_n.

   a6) Read 0000_006xh from the CPG register (Offset address: A24h). Confirm the reset_brdg_n was already negated.

   a7) Write 0060_0060h hto the CPG register (Offset address: 948h). Negate the reset_fcpf_p and reset_fdp1_p.

   a8) Read 0000_000xh from the CPG register (Offset address: A24h). Confirm the reset_fcpf_p and reset_fdp1_p already negated.

   Step2) Re-start FDP with the software reset sequence following to the guidance in **9.10.2.5.1 Operation Control Setting for the IP Set FDP**.

SECTION 9 IMAGE

9.11 Low Voltage Differential Signaling (LVDS)

9.11.1 Overview

This module is equipped with LVDS_CORE, APB IF section, and Renesas' PHY IP (PHY section), which are collectively provided as LVDS. This section describes the specifications of this LSI.

It complies with the following standards:

- FPD-link Standards Compliant (TIA/EIA-644)
- VESA Standards Compliant

9.11.1.1 Feature

Table 9.11-1 shows the main features of this module.

Table 9.11-1 Overview of Functions

Function	Description
Single-link Target	• RGB888 640 × 480 (VGA) • RGB888 800 × 480 (QVGA) • RGB888 1366 × 768 60 Hz (WXGA) • RGB888 1280 × 800 60 Hz (WXGA)
Dual-link Target	• RGB888 1920 × 1200 60 fps (up to 3.32 Gbps) • RGB888 1920 × 1080 60 fps (up to 2.99 Gbps)
Number of support lanes	• Single-link: 4 lanes (Data) + 1 lane (Clock) • Single-link (2ch): 8 lanes (Data) + 2 lanes (Clock) • Dual-link: 8 lanes (Data) + 2 lanes (Clock)
Link function	• Video mode supports RGB888 • Output Mode Select of 8 Format • Dual-Link Even/Odd Swap • Lane Select (In a channel) • Supports 8 lanes transfers. (4 lanes × 2ch)
PHY function	• Parallel-Serial Converter • LVDS IO Buffer Control • Clock Rate: 25 to 87 MHz (Up to 87 MHz even with SSC ON) • Data Rate: 175 to 609 Mbps/Lane
Register interface	• AMBA® APB (v1.0) compliant Slave interface • Data bus width 32 bits • Two systems of LVDS_CORE register / PHY register
SSC	• Modulation Frequency: 30 to 150 kHz • Center Spread: −1 to +1 [%] • Down Spread: 0 to −2 [%]

9.11.1.2 Block Diagram

Figure 9.11-1 shows a block diagram of LVDS.

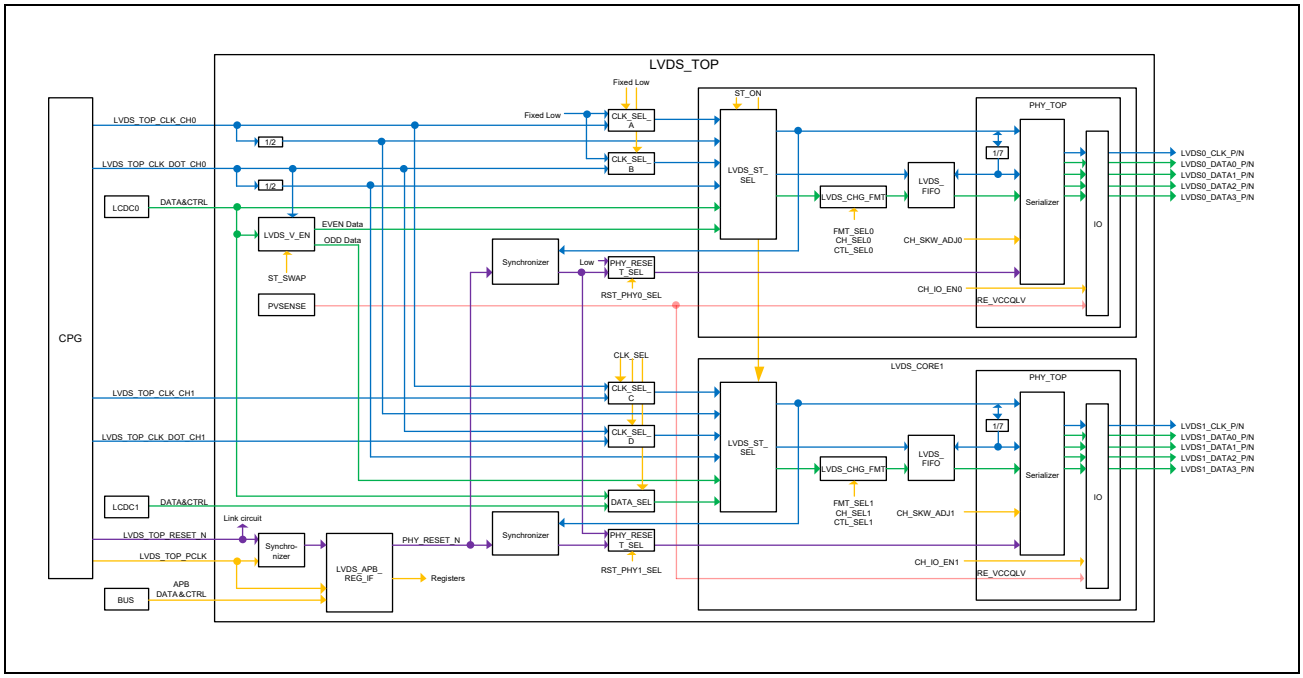


Figure 9.11-1 Block Diagram of LVDS

LVDS_TOP

This is the top circuit of LVDS consisting of the LVDS_CORE section for LVDS output control and the PHY section. This module is equipped with two core circuits for LVDS 2ch support, a Dual-link control circuit and an APB IF circuit. In addition, it is configured to implement multiple Dividers, Selectors, and Synchronizers to match the monitor specifications.

LVDS_CORE section

It implements functions such as single-link and dual-link switching, output format switching, and lane switching. The PHY section is mounted on the LVDS_CORE section.

PHY section

It consists of a Serializer and LVDS IO. Renesas IP is used.

APB IF section

The APB IF section implements registers for controlling the LVDS_CORE and PHY sections.

9.11.1.3 External Pins

Table 9.11-2 shows the pin configuration.

Table 9.11-2 Pin Configuration

Pin Name	I/O	Function	Pin State when not in Use
LVDS0_DATA0_P	Out	LVDS0 TX differential data lane 0 (positive)	Open
LVDS0_DATA0_N	Out	LVDS0 TX differential data lane 0 (negative)	Open
LVDS0_DATA 1_P	Out	LVDS0 TX differential data lane 1 (positive)	Open
LVDS0_DATA 1_N	Out	LVDS0 TX differential data lane 1 (negative)	Open
LVDS0_DATA 2_P	Out	LVDS0 TX differential data lane 2 (positive)	Open
LVDS0_DATA 2_N	Out	LVDS0 TX differential data lane 2 (negative)	Open
LVDS0_DATA 3_P	Out	LVDS0 TX differential data lane 3 (positive)	Open
LVDS0_DATA3_N	Out	LVDS0 TX differential data lane 4 (negative)	Open
LVDS0_CLK_P	Out	LVDS0 TX differential clock lane (positive)	Open
LVDS0_CLK_N	Out	LVDS0 TX differential clock lane (negative)	Open
LVDS1_DATA0_P	Out	LVDS1 TX differential data lane 0 (positive)	Open
LVDS1_DATA0_N	Out	LVDS1 TX differential data lane 0 (negative)	Open
LVDS1_DATA1_P	Out	LVDS1 TX differential data lane 1 (positive)	Open
LVDS1_DATA1_N	Out	LVDS1 TX differential data lane 1 (negative)	Open
LVDS1_DATA2_P	Out	LVDS1 TX differential data lane 2 (positive)	Open
LVDS1_DATA2_N	Out	LVDS1 TX differential data lane 2 (negative)	Open
LVDS1_DATA3_P	Out	LVDS1 TX differential data lane 3 (positive)	Open
LVDS1_DATA3_N	Out	LVDS1 TX differential data lane 4 (negative)	Open
LVDS1_CLK_P	Out	LVDS1 TX differential clock lane (positive)	Open
LVDS1_CLK_N	Out	LVDS1 TX differential clock lane (negative)	Open

Note: LVDS is a differential output interface. The channel supports 2 channels, TX0 = Ch0 and TX1 = Ch1. Each channel consists of 1 Clock Lane and 4 Data Lanes.

9.11.1.4 Connected Module

Table 9.11-3 shows modules connected to this module.

Table 9.11-3 Connected Module

Module Name	Connected Module Name	Function of Connected Module
LVDS	CPG	Clock Pulse Generator
	BUS	Interconnect BUS
	LCDC0	LCD Controller0
	LCDC1	LCD Controller1
	External pin	External pin

9.11.2 Registers

The LVDS register address is <LVDS_base> given as an offset from the base address. The register base addresses of LVDS are listed in the following table.

Table 9.11-4 Register Base Addresses

Base Address Name	Base Address	Function
<LVDS_base>	0_164C_0000h (564C_0000h*1, 464C_0000h*2)	

Note 1. CM33 address space (non-secure)

Note 2. CM33 address space (secure)

9.11.2.1 List of Registers

The LVDS registers and the memory addresses are listed in the following table.

For the actual addresses, the offset values indicated in this table are added to the base addresses.

Register Name	Abbreviation	Initial Value	Offset Address	Access Size [bits]
LVDS Common Register	LVDS_CMN	0000_0000h	0000h	32
Reserve	-	-	0004h to 000Fh	-
LVDS0 PHY Register	LVDS_0_PHY	0000_0000h	0010h	32
LVDS0 Control Register	LVDS_0_CTL	0000_0000h	0014h	32
LVDS1 PHY Register	LVDS_1_PHY	0000_0000h	0018h	32
LVDS1 Control Register	LVDS_1_CTL	0000_0000h	001Ch	32

Note: Addresses other than those listed above are Reserved. Access to the reserved area is prohibited. The operation when accessed is not guaranteed. Access is 32 bits. Byte access is not allowed.

9.11.2.2 LVDS Common Register (LVDS_CMN)

This is a common register for LVDS modules.

Registers are implemented to select the clock, reset, and data paths according to the operating mode.

Access Size : 32 bits
Address : <LVDS_base> + 0000h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	RST_PHY1_SEL	RST_PHY0_SEL	-	-	-	-	-	-	-	CLK_SEL
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	R	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	ST_SWAP	ST_ON	-	-	-	-	-	-	-	PHY_RESET_N
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R	R	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
25	RST_PHY1_SEL	0h	RW	PHY1 Reset Signal Select 0: CH0 1: CH1
24	RST_PHY0_SEL	0h	RW	PHY0 Reset Signal Select 0: Low 1: CH0
23 to 17	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
16	CLK_SEL	0h	RW	Clock Select (LVDS_TOP_CLK_CHx, LVDS_TOP_CLK_DOT_CHx): x = 0, 1 CLK_SEL_C 0: LVDS_TOP_CLK_CH0 1: LVDS_TOP_CLK_CH1 CLK_SEL_D 0: LVDS_TOP_CLK_DOT_CH0 1: LVDS_TOP_CLK_DOT_CH1
15 to 10	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
9	ST_SWAP	0h	RW	LVDS Dual-Link Even/Odd Data Stripe Swap 0: ch0 = Even, ch1 = Odd 1: ch0 = Odd, ch1 = Even
8	ST_ON	0h	RW	Dual-link Select (Stripe-ON) 0: Single-link 1: Dual-link
7 to 1	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
0	PHY_RESET_N	0h	RW	LVDS PHY0 & PHY1 Reset 0: Reset, 1: Normal

9.11.2.3 LVDS0 PHY Register (LVDS_0_PHY)

This is the PHY control register for LVDS0 (ch0).

Register is implemented to adjust the LVDS output clock and data skew and to enable the LVDS output IO.

Access Size : 32 bits
Address : <LVDS_base> + 0010h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	CH_SKW_ADJ0[14:0]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	CH_IO_EN0[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 16	CH_SKW_ADJ0 [14:0]	0h	RW	LVDS0 Skew Adjust 000: +0 ps, 100: -0 ps 001: +30 ps, 101: -30 ps 010: +60 ps, 110: -60 ps 011: +90 ps, 111: -90 ps [30:28]: Lane4, [27:15]: Lane3, [24:22]: Lane2, [21:19]: Lane1, [18:16]: Lane0
15 to 5	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4 to 0	CH_IO_EN0 [4:0]	0h	RW	LVDS0 DRV Enable (Power Cut) 0: OutputBuffer, 1: Normal Operation [4]: Lane4, [3]: Lane3, [2]: Lane2, [1]: Lane1, [0]: Lane0

9.11.2.4 LVDS0 Control Register (LVDS_0_CTL)

This is the control register of the LVDS_CORE part of LVDS0 (ch0).

This is a register that controls the functions of the LVDS_CORE section (Control bit selection, output format selection, and output lane switching).

Access Size : 32 bits
Address : <LVDS_base> + 0014h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	CTL_SEL0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	FMT_SEL0[3:0]				CH_SEL0[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	CTL_SEL0[7:0]	0h	RW	LVDS0 Control bit Select [17:16]: ctrl0 00: HSYNC, 01: VSYNC, 10: DE, 11: Low [19:18]: ctrl1 00: VSYNC, 01: DE, 10: Low, 11: HSYNC [21:20]: ctrl2 00: DE, 01: Low, 10: HSYNC, 11: VSYNC [23:22]: ctrl3 00: Low, 01: HSYNC, 10: VSYNC, 11: DE
15 to 12	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	FMT_SEL0[3:0]	0h	RW	LVDS0 Output Format Variation 0000: Mode0, 0001: Mode1, 0010: Mode2, 0011: Mode3 0100: Mode4, 0101: Mode5, 0110: Mode6, 0111: Mode7
7 to 0	CH_SEL0[7:0]	0h	RW	LVDS0 Data Lane Select [1:0]: Lane0 00: Lane0, 01: Lane1, 10: Lane2, 11: Lane3 [3:2]: Lane1 00: Lane1, 01: Lane2, 10: Lane3, 11: Lane0 [5:4]: Lane2 00: Lane2, 01: Lane3, 10: Lane0, 11: Lane1 [7:6]: Lane3 00: Lane3, 01: Lane0, 10: Lane1, 11: Lane2

9.11.2.5 LVDS1 PHY Register (LVDS_1_PHY)

This is the PHY control register for LVDS1 (ch1).

Register is implemented to adjust the LVDS output clock and data skew and to enable the LVDS output IO.

Access Size : 32 bits
Address : <LVDS_base> + 0018h
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	CH_SKW_ADJ1[14:0]														
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	CH_IO_EN1[4:0]				
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
30 to 16	CH_SKW_ADJ1 [14:0]	0h	RW	LVDS1 Skew Adjust 000: +0 ps, 100: -0 ps 001: +30 ps, 101: -30 ps 010: +60 ps, 110: -60 ps 011: +90 ps, 111: -90 ps [30:28]: Lane4, [27:15]: Lane3, [24:22]: Lane2, [21:19]: Lane1, [18:16]: Lane0
15 to 5	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
4 to 0	CH_IO_EN1[4:0]	0h	RW	LVDS1 DRV Enable (Power Cut) 0: OutputBuffer, 1: Normal Operation [4]: Lane4, [3]: Lane3, [2]: Lane2, [1]: Lane1, [0]: Lane0

9.11.2.6 LVDS1 Control Register (LVDS_1_CTL)

This is the control register of the LVDS_CORE part of LVDS1 (ch1).

This is a register that controls the functions of the LVDS_CORE section (Control bit selection, output format selection, and output lane switching).

Access Size : 32 bits
Address : <LVDS_base> + 001Ch
Initial Value : 0000_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	CTL_SEL1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	FMT_SEL1[3:0]				CH_SEL1[7:0]							
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
23 to 16	CTL_SEL1[7:0]	0h	RW	LVDS1 Control bit Select [17:16]: ctrl0 00: HSYNC, 01: VSYNC, 10: DE, 11: Low [19:18]: ctrl1 00: VSYNC, 01: DE, 10: Low, 11: HSYNC [21:20]: ctrl2 00: DE, 01: Low, 10: HSYNC, 11: VSYNC [23:22]: ctrl3 00: Low, 01: HSYNC, 10: VSYNC, 11: DE
15 to 12	-	0h	R	Reserved Whenever it is read, 0 is read. The written value will be ignored.
11 to 8	FMT_SEL1[3:0]	0h	RW	LVDS1 Output Format Variation 0000: Mode0, 0001: Mode1, 0010: Mode2, 0011: Mode3 0100: Mode4. 0101: Mode5, 0110: Mode6, 0111: Mode7
7 to 0	CH_SEL1[7:0]	0h	RW	LVDS1 Data Lane Select [1:0]: Lane0 00: Lane0, 01: Lane1, 10: Lane2, 11: Lane3 [3:2]: Lane1 00: Lane1, 01: Lane2, 10: Lane3, 11: Lane0 [5:4]: Lane2 00: Lane2, 01: Lane3, 10: Lane0, 11: Lane1 [7:6]: Lane3 00: Lane3, 01: Lane0, 10: Lane1, 11: Lane2

9.11.3 Operation

9.11.3.1 Power On Sequence

The initialization procedure for this module is shown in **Figure 9.11-2**.

The basic sequence is as follows.

All operating modes (Single-link, Dual-link) can be activated by following the steps below. For details on LVDS operating modes, refer to **9.11.3.3 LVDS Operating Modes**.

1. Power-on
2. Unreset LVDS (RESET_N)
3. Supplying the APB clock to LVDS (LVDS_TOP_PCLK)
4. LVDS_CORE Register Settings (Excludes PHY_RESET)
5. Provides LVDS PHY and DOT clocks (LVDS_TOP_CLK_CH0, LVDS_TOP_CLK_CH1, LVDS_TOP_CLK_DOT_CH0, LVDS_TOP_CLK_DOT_CH1)
6. PHY register setting
7. Unreset PHY

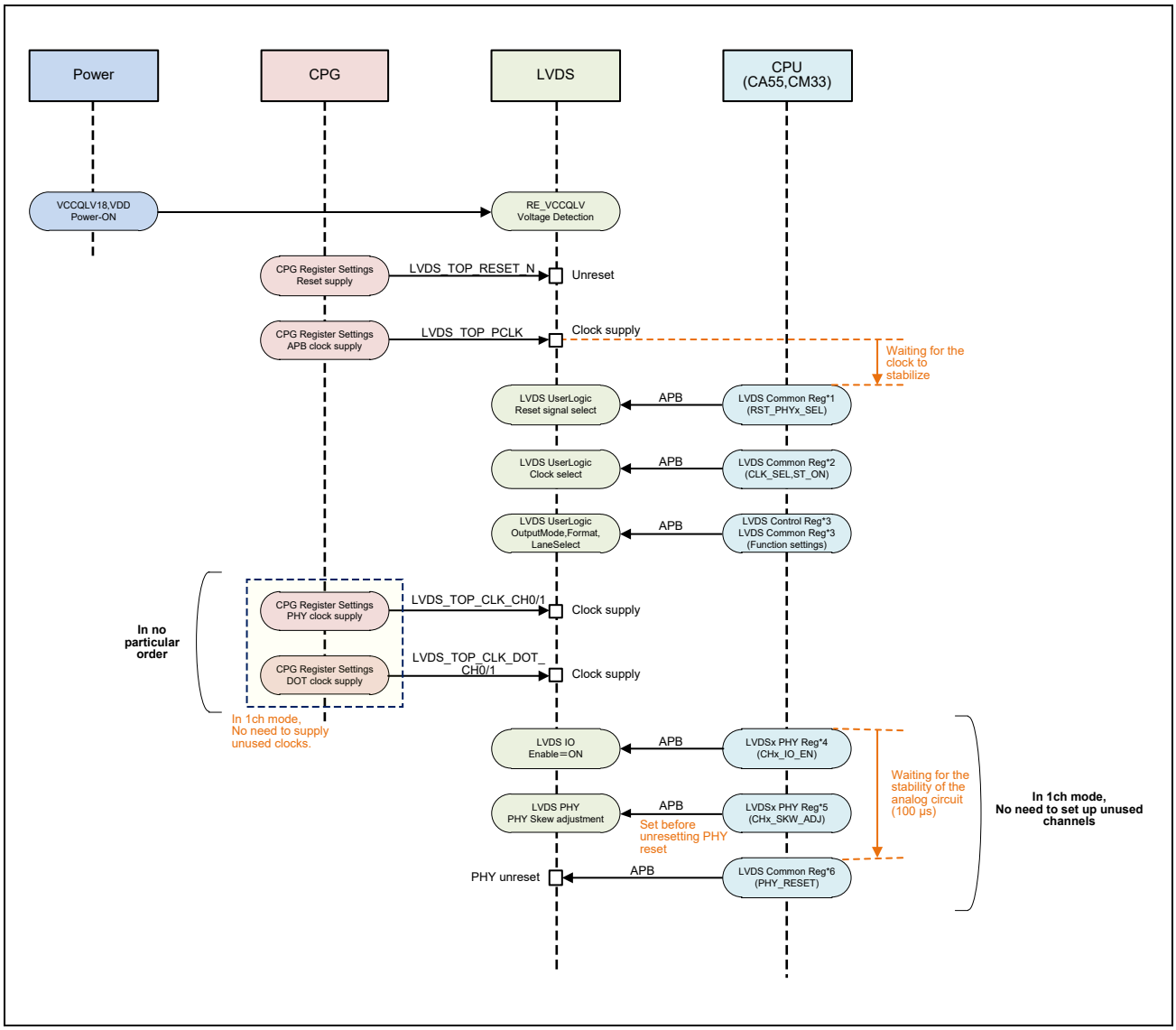


Figure 9.11-2 Power On Sequence

Remark Refer to **Table 9.11-5** for *1 to *6 in the figure.
For the relationship between each operation mode and the register settings, refer to **Table 9.11-6**.

(1) Power on, unreset

After the power is turned on (VCCQLV18, VDD), the reset (RESET_N) is canceled for LVDS startup. The LVDS is unreset by the CPG register control. For more information about the CPG registers, refer to CPG chapter.

(2) Clock supply

The clock input to the LVDS can be frequency-set, supplied, and deactivated by the CPG. For more information about the CPG registers, refer to CPG chapter.

Also refer to “3. Supplying the APB clock to LVDS (LVDS_TOP_PCLK)”, “4. LVDS_CORE Register Settings (Excludes PHY_RESET)”, and **9.11.3.7 Output Pattern**, as LVDS requires a reset and clock to be selected according to the mode of operation.

It is recommended that the following clock frequency settings be set before clocking the LVDS.

- LVDS_TOP_CLK_CH0/1 and LVDS_TOP_CLK_DOT_CH0/1 must be set to the required frequency based on the specifications of the panel to be connected.

(3) Register settings (Common, PHY, LVDS_CORE)

After the APB clock (LVDS_TOP_PCLK) is stabilized, the PHY and LVDS_CORE register settings are configured. Depending on the mode of operation, the initial settings of the registers must be made.

<Common Register Settings>

LVDS_CMN is the register of common settings. Depending on the channel to be used, it is necessary to set the LVDS0, 1 registers. There is no difference in the register settings depending on the channel.

For details on how to set it, refer to (1) of **9.11.3.7 Output Pattern**.

- Register function
  1) CLK_SEL
  2) RST_PHYx_SEL
  3) PHY_RESET
  4) ST_ON
  5) ST_SWAP

<PHY Register Settings>

LVDS_0_PHY and LVDS_1_PHY are the registers for the PHY configuration. Depending on the channel to be used, it is necessary to set the LVDS0, 1 registers. There is no difference in the register settings depending on the channel.

- Setup Procedure
  1) Set the CHx_IO_EN according to the lane to be used.
  2) If you want to adjust Skew for each channel, set the CHx_SKW_ADJ.
  3) After the stabilization time (100 µs) of the Analog circuit, including the PHY clock (LVDS_TOP_CLK_CH0,1) and “1)”, set the PHY_RESET register and release the PHY reset.

  Note: x = 0 or 1

<LVDS_CORE Register Settings>

LVDS_0_CTL and LVDS_1_CTL are the registers for LVDS_CORE configuration. Depending on the channel to be used, it is necessary to set the LVDS0, 1 registers. Make the appropriate settings according to the single-link and dual-link.

For details on how to do this, refer to **9.11.3.5 Output Format Selection** (FMT_SEL), **9.11.3.8 Control Bit Switching** (CTL_SEL), and **9.11.3.9 Output Lane Selection** (CH_SEL).

- Setting conditions

  Before configuring the PHY register, configure the LVDS_CORE register.
  The settings for the various LVDS_CORE registers are in no particular order.
  The registers that require initial settings are as follows. The registers used by default do not need to be initialized.

Table 9.11-5 List of Registers that Require Initial Setting

No.	Numbers in the figure	Register Name	Address	Bit	I/O	Function
1	*1	LVDS_CMN	000h	25	RST_PHY1_SEL	LVDS PHY1 Reset Signal Select
2	*1			24	RST_PHY0_SEL	LVDS PHY0 Reset Signal Select
3	*2			16	CLK_SEL	LVDS Clock Select
4	*3			9	ST_SWAP	LVDS Dual-Link Even/Odd Data Swap
5	*2			8	ST_ON	LVDS Dual-Link Clock and Data Select
6	*6			0	PHY_RESET	LVDS0,1 PHY Reset
7	*5	LVDS_0_PHY	010h	30-16	CH_SKW_ADJ0[14:0]	LVDS0 Skew Adjust
8	*4			4-0	CH_IO_EN0 [4:0]	LVDS0 DRV Enable (Power Cut)
9	*3	LVDS_0_CTL	014h	23-16	CTL_SEL0 [7:0]	LVDS0 Control bit Select
10	*3			11-8	FMT_SEL0 [3:0]	LVDS0 Output Format Variation
11	*3			7-0	CH_SEL [7:0]	LVDS0 Data Lane Select
12	*5	LVDS_1_PHY	018h	30-16	CH_SKW_ADJ1 [14:0]	LVDS1 Skew Adjust
13	*4			4-0	CH_IO_EN1 [4:0]	LVDS1 DRV Enable (Power Cut)
14	*3	LVDS_1_CTL	01Ch	23-16	CTL_SEL1 [7:0]	LVDS1 Control bit Select
15	*3			11-8	FMT_SEL1 [3:0]	LVDS1 Output Format Variation
16	*3			7-0	CH_SEL1 [7:0]	LVDS1 Data Lane Select

(4) Relationship between operating mode and register setting

The initial setting is performed according to the procedure in **Figure 9.11-2**, regardless of whether it is a single-link or a dual-link.

If only one channel is used, it is not necessary to set the CPG and LVDS registers for the unused channel.

Table 9.11-6 shows the relationship between the operating mode and the register settings.

Note 1. Refer to the CPG chapter for details on the clock settings (LVDS_TOP_CLK_CH0, LVDS_TOP_CLK_CH1, LVDS_TOP_CLK_DOT_CH 0, LVDS_TOP_CLK_DOT_CH 1) of CPG Reg.

Note 2. For LVDS Reg., refer to **9.11.2 Registers**.

Table 9.11-6 Relationship between Operating Mode and Register Setting

Operation Mode	CPG Reg.				LVDS Reg.	
	LVDS_TOP_CLK_CH0	LVDS_TOP_CLK_CH1	LVDS_TOP_CLK_DOT_CH0	LVDS_TOP_CLK_DOT_CH1	LVDS0	LVDS1
Single-link (ch1)	✓	—	✓	—	✓	—
Single-link (ch2)	—	✓	—	✓	—	✓
Single-link (2ch)	✓	✓	✓	✓	✓	✓
Single-link (Multi)	✓	—	✓	—	✓	—
Dual-link	✓	—	✓	—	✓	—

9.11.3.2 Power Off Sequence

The flow of the power off sequence in this module is shown in **Figure 9.11-3**. The Power off sequence is the reverse of the Power on sequence.

The basic sequence is as follows.

1) PHY Reset
2) LVDS IO Disable
3) LVDS reset, clock stop
4) Power OFF

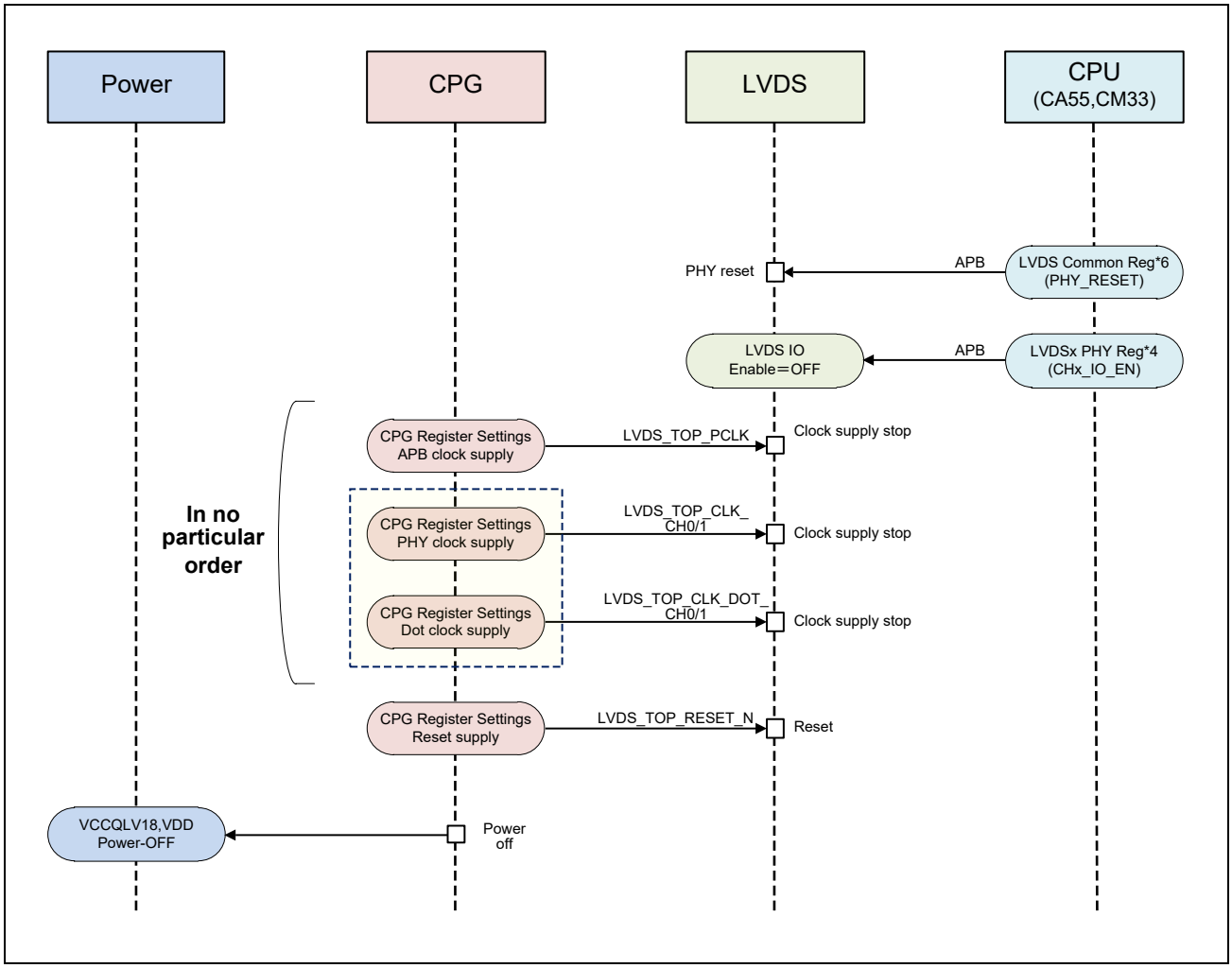


Figure 9.11-3 Power Off Sequence

Remark Refer to **Table 9.11-5** for *4 and *6 in the figure.

9.11.3.3 LVDS Operating Modes

The module supports five different modes of operation. This chapter describes each mode of operation. For details on setting the operation mode, refer to **9.11.3.7 Output Pattern**.

(1) Single-link (ch0 only)

This mode outputs the image data of LCDC0 to LVDS (ch0) as shown below. In this mode, LVDS (ch1) is not used, so the ch1 side is used in the reset state and clock stop in consideration of power saving.

Figure 9.11-4 shows an image of the single-link (ch0 only) clock and data path.

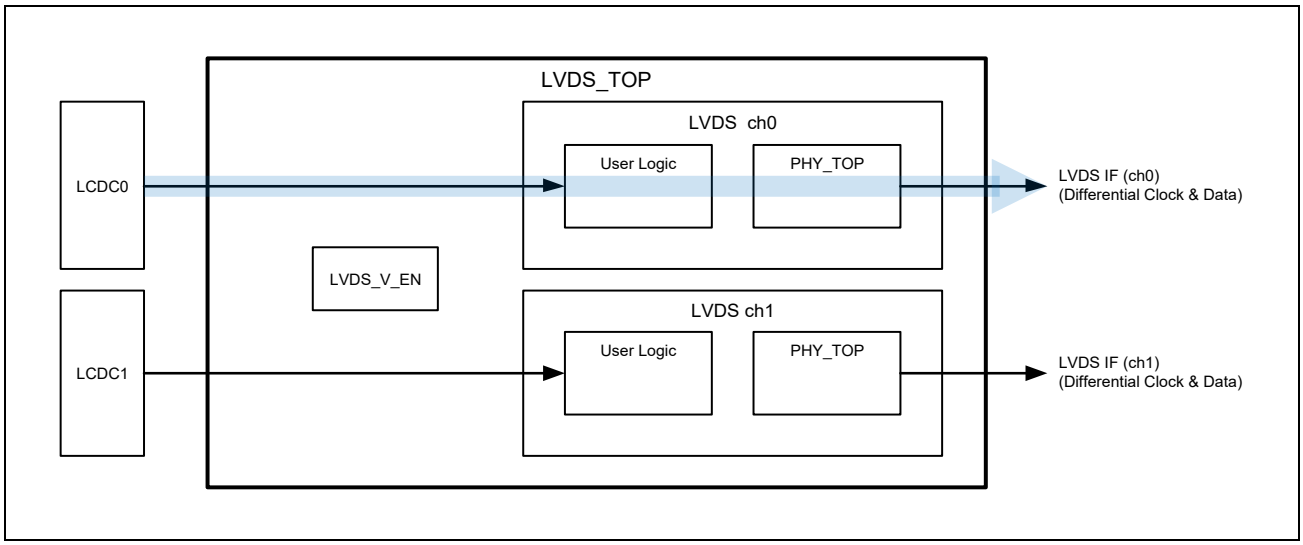


Figure 9.11-4 Single-link (Ch0 Only) Data Path Image

(2) Single-link (ch1 only)

This mode outputs the image data of LCDC1 to LVDS (ch1) as shown below. In this mode, LVDS (ch0) is not used, so the ch0 side is used in the reset state and clock stop in consideration of power saving.

Figure 9.11-5 shows an image of the single-link (ch1 only) clock and data path.

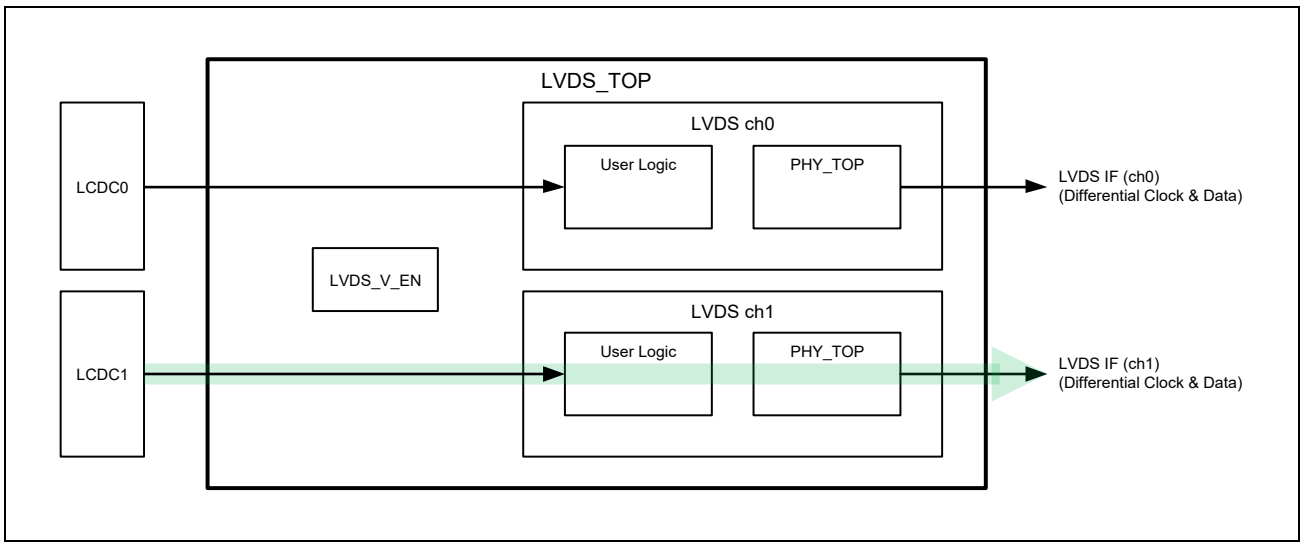


Figure 9.11-5 Single-link (Ch1 Only) Data Path Image

(3) Single-link (2ch)

In this mode, the image data of LCDC0 is output to LVDS (ch0) and the image data of LCDC1 is output to LVDS (ch1). Since LVDS (ch0) and LVDS (ch1) are not synchronously related, they can be output in different image formats and can be operated asynchronously. In this mode, two channels are used, so it is necessary to cancel the reset and supply the clock for each channel.

Figure 9.11-6 shows an image of the single-link (2ch) clock and data path.

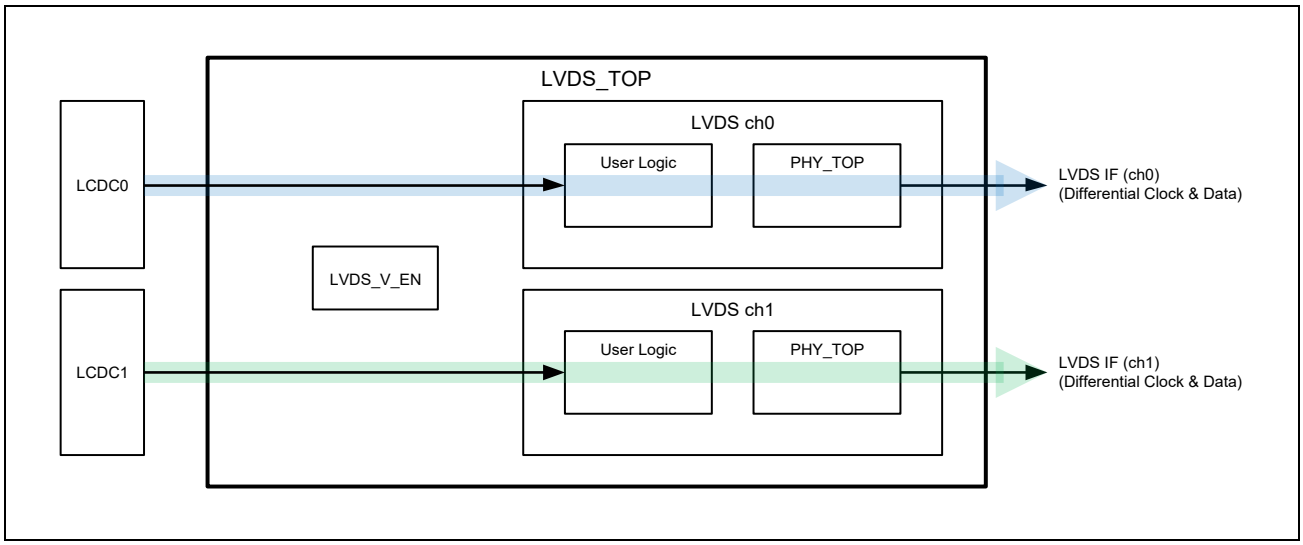


Figure 9.11-6 Single-link (2ch) Data Path Image

(4) Single-link (Multi)

In this mode, the image data of LCDC0 is output to both LVDS (ch0) and LVDS (ch1). LVDS (ch0) and LVDS (ch1) operate synchronously. In this mode, two channels are used, so it is necessary to cancel the reset and supply the clock for each channel.

Figure 9.11-7 shows an image of the single-link (Multi) clock and data path.

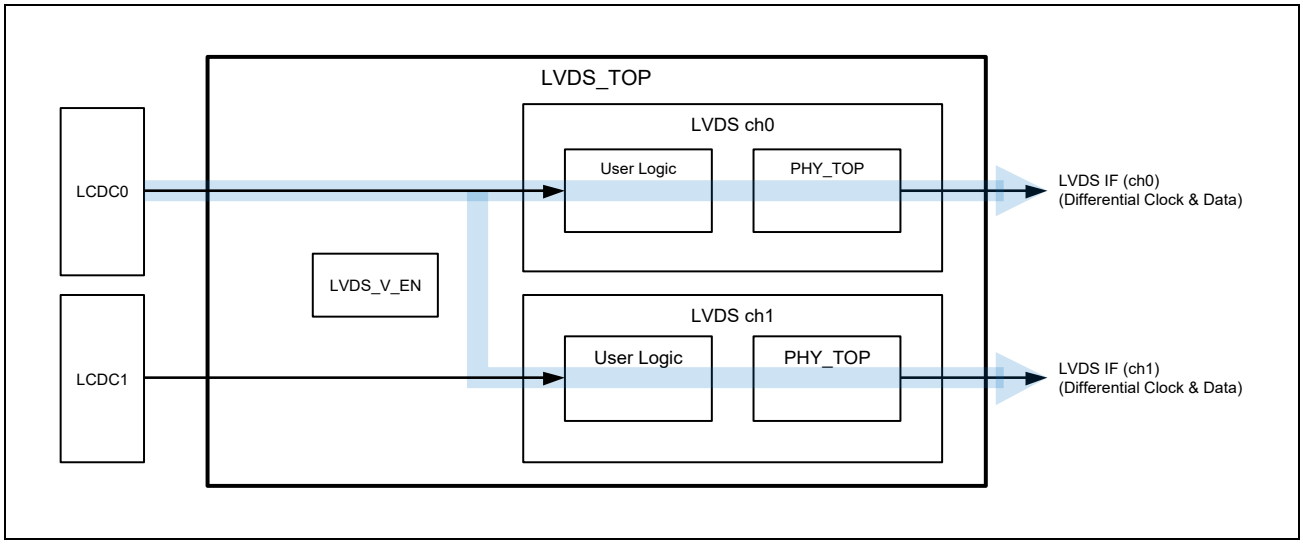


Figure 9.11-7 Single-link (Multi) Data Path Image

(5) Dual-link

In this mode, the input image data is separated into Even pixels and Odd pixels, and the output is distributed to LVDS ch0 and ch1. The destination channel of the Even pixel and Odd pixel can be changed by setting the LVDS registers. In this mode, two channels are used, so it is necessary to cancel the reset and supply the clock for each channel.

Figure 9.11-8 shows an image of the dual-link clock and data path.

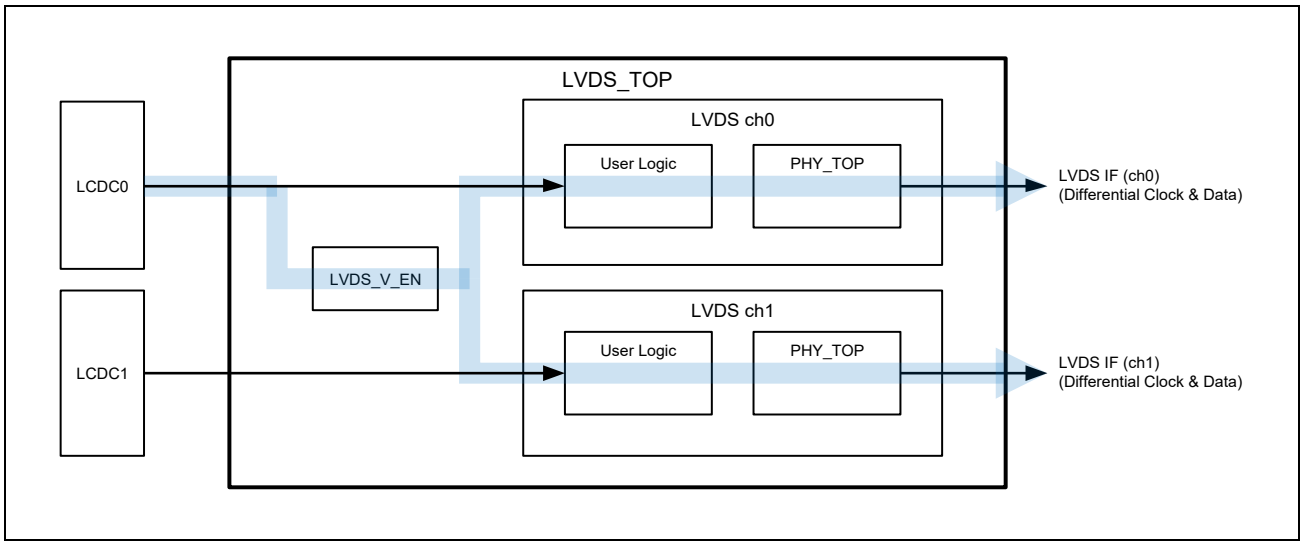


Figure 9.11-8 Dual-link Data Path Image

9.11.3.4 Input Format

For the input format to this module, please refer to **9.4 LCD Controller (LCDC)**.

9.11.3.5 Output Format Selection

This module supports 8 types of output formats (Modes). Switch the output format by setting the register (LVDS_0_CTL.FMT_SEL[3:0]/ LVDS_1_CTL.FMT_SEL[3:0]). The output image data is 24 bits for RGB (8 bits each) and 4 bits for Control bits. The control bits input from the LCDC consist of 4 bits: HSYNC, VSYNC, DE, and Low(fixed).

Refer to **Figure 9.11-13** for the data format conversion from LCDC to LVDS.

The output formats for each mode are shown in **Figure 9.11-9**, **Figure 9.11-10**, **Figure 9.11-11**, and **Figure 9.11-12**.

Remark R = Red Data, G = Green Data, B = Blue Data, CTL = Control bit

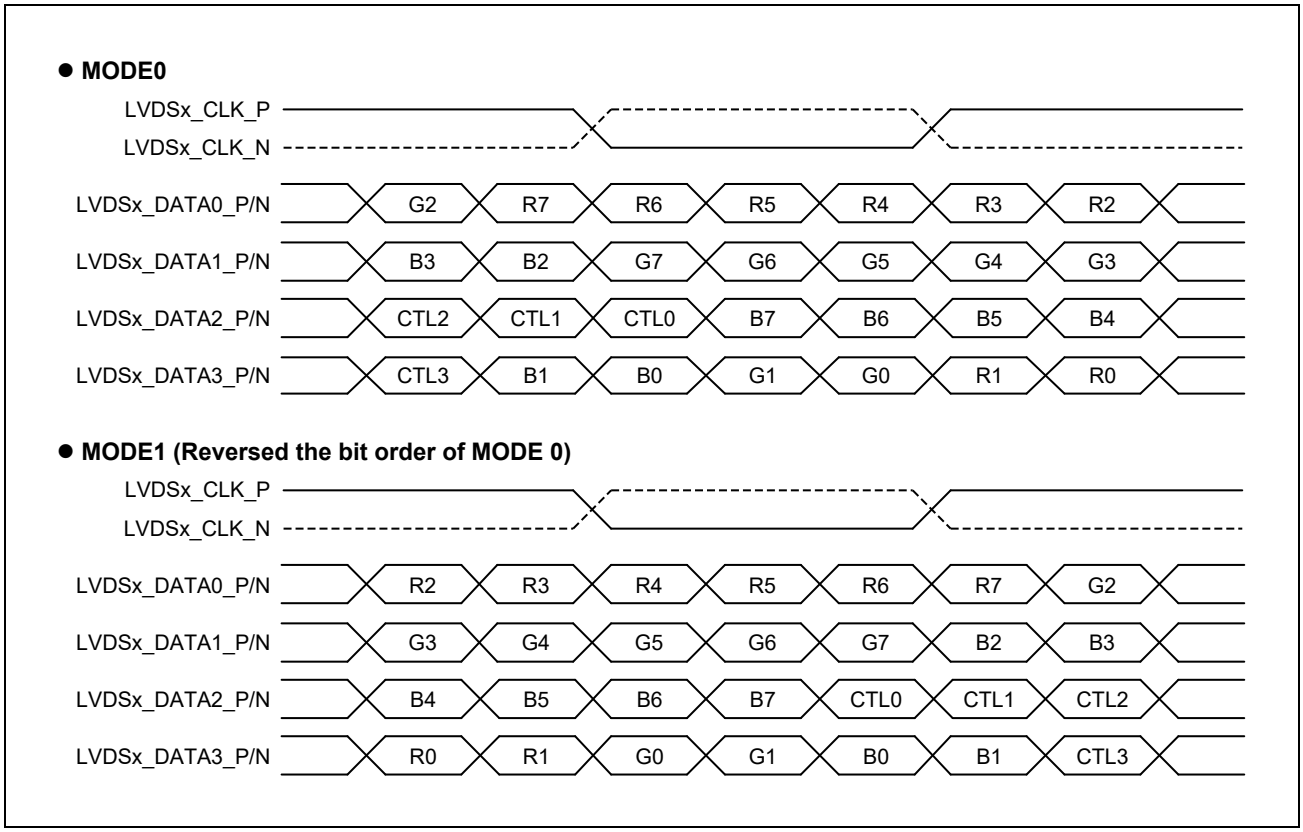


Figure 9.11-9 Mode0, Mode1 Output Format

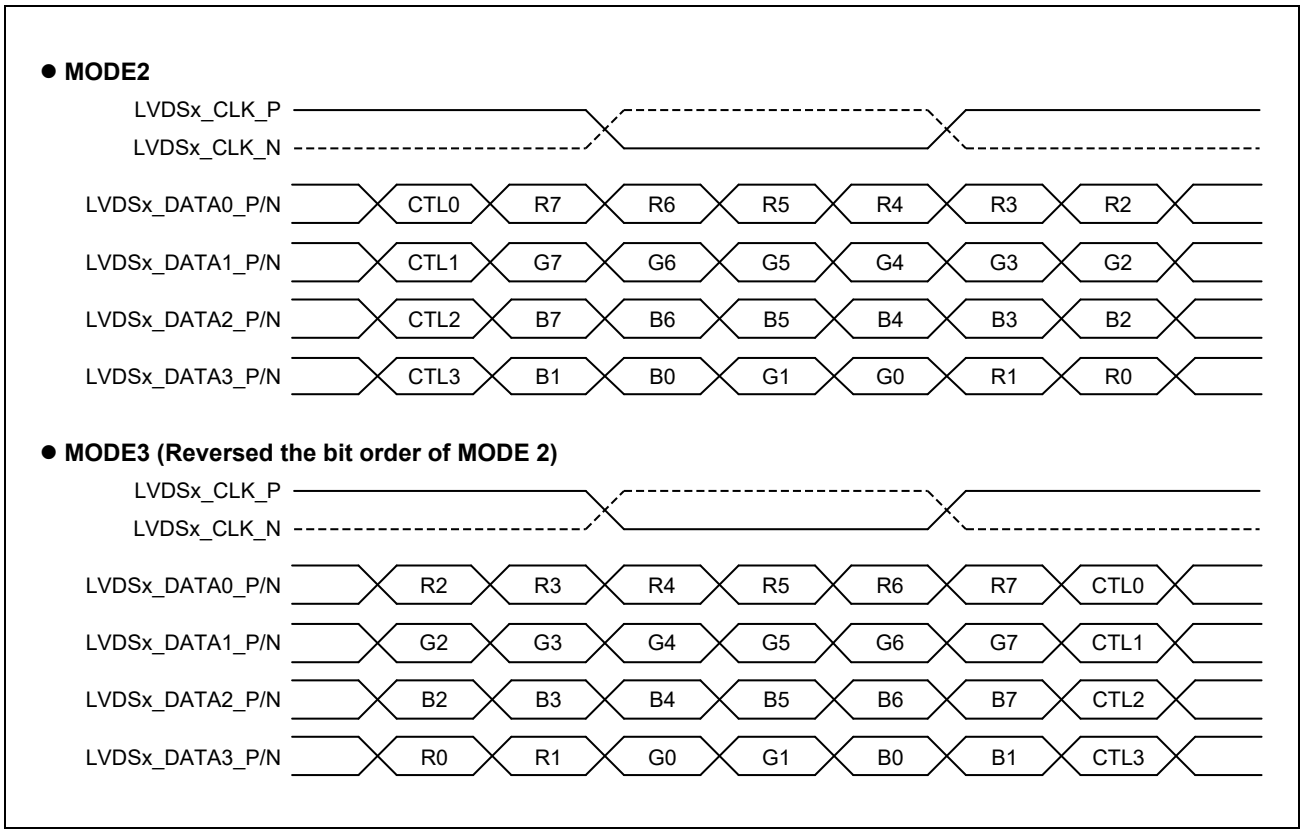


Figure 9.11-10 Mode2, Mode3 Output Format

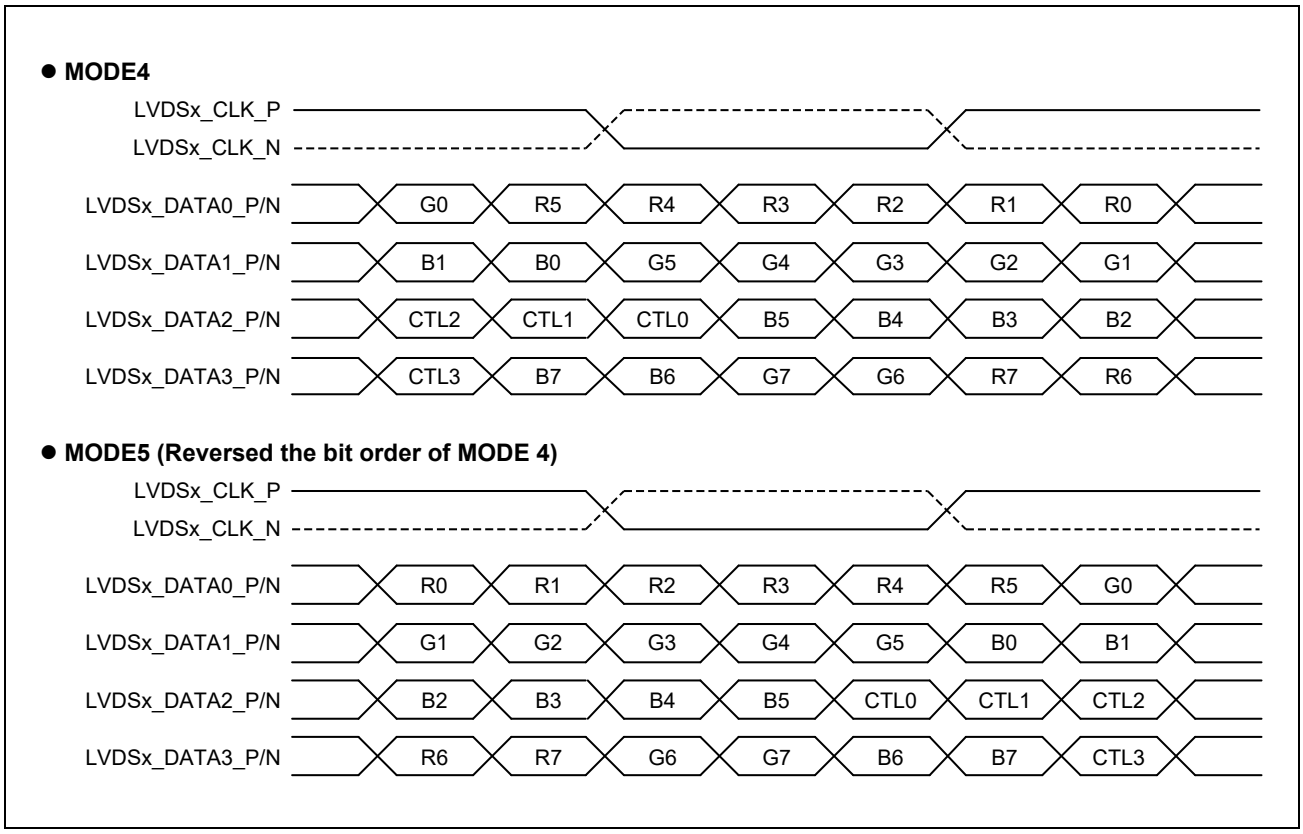


Figure 9.11-11 Mode4, Mode5 Output Format

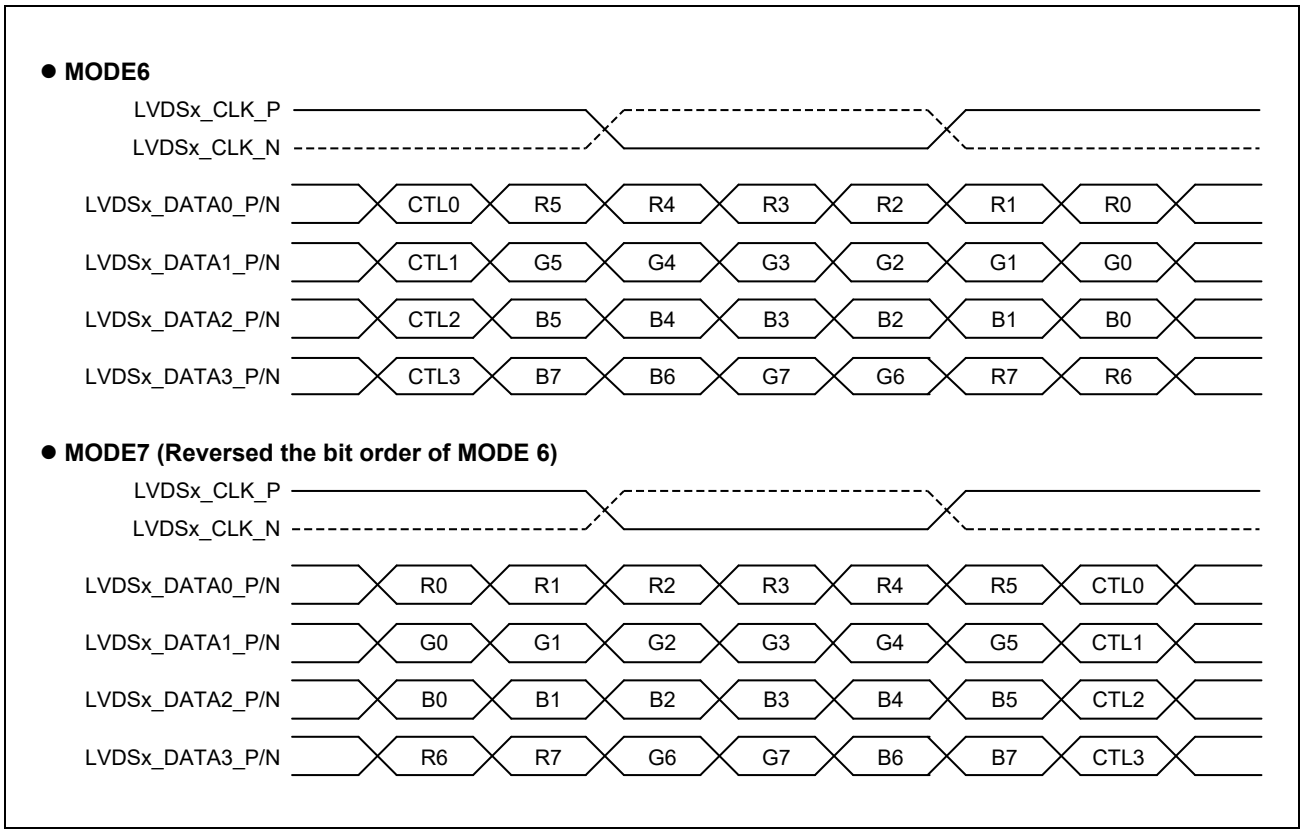


Figure 9.11-12 Mode6, Mode7 Output Format

9.11.3.6 Input/Output Data Description

This module converts the input data from the LCDC to the selected output format (refer to **9.11.3.5 Output Format Selection**) and outputs it. The clock must be output with a duty ratio of 4:3 (High:Low) according to the TIA/EIA-644 standard and is generated by the PHY. As for image data, the data equivalent to one clock of the LCDC is output by the Serializer at 7 times the speed. This section describes how to convert input data and control bits at the time of output (bit assignment).

(1) Data Conversion

Describes data transformation. The figure below is an example of conversion. The input data is converted into the selected data according to the output format (refer to **9.11.3.5 Output Format Selection**) and output. For details on the register settings for selecting the output format, refer to LVDS_0_CTL and LVDS_1_CTL.

The upper part of the figure below is the input data from the LCDC. 24 bits of data are input per clock.

As shown at the bottom of the figure below, the input parallel data is sorted into serial data and output.

The conversion of parallel data to serial data is done by the PHY's Serializer.

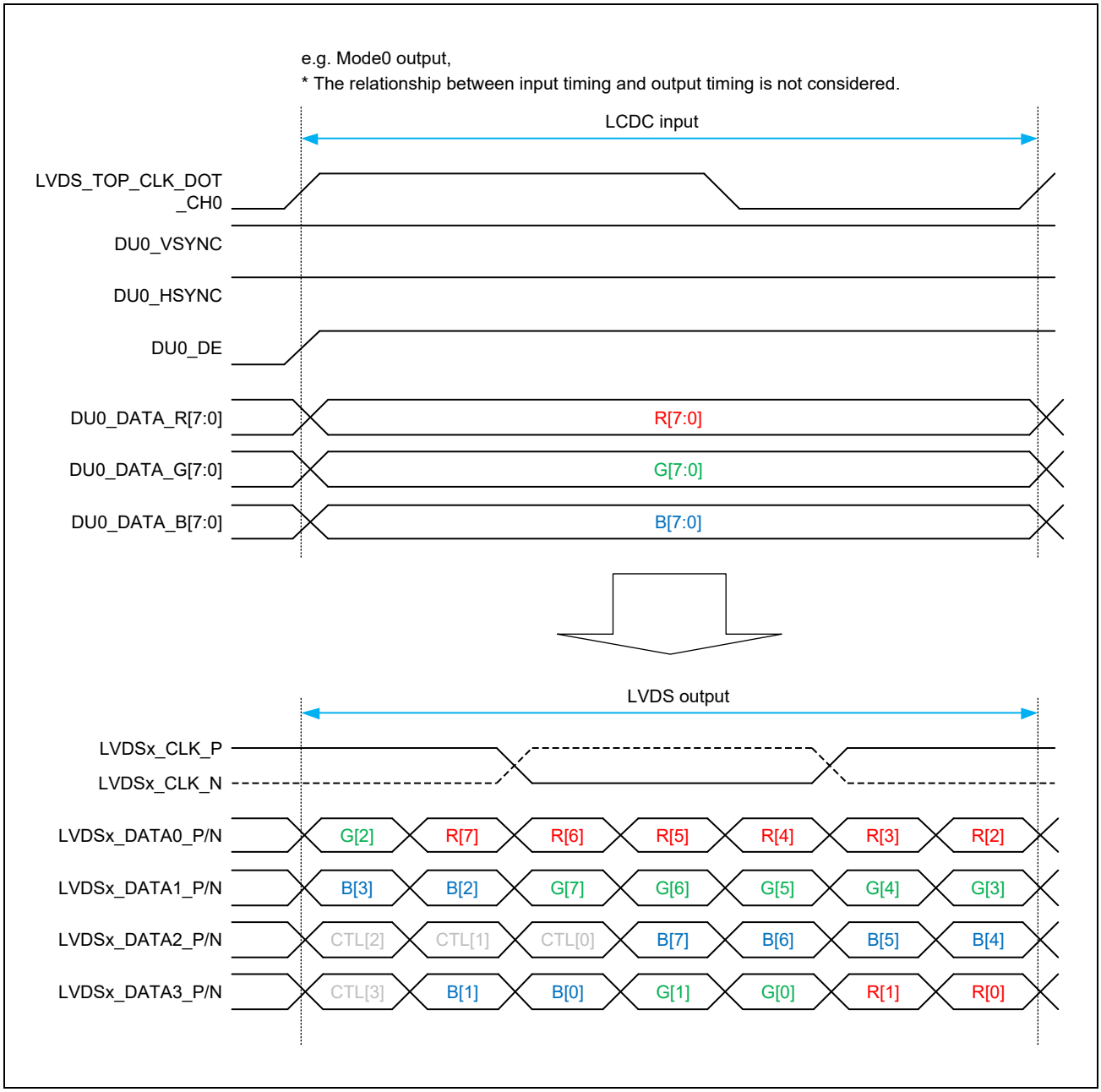


Figure 9.11-13 Data Conversion

(2) Control Bit Conversion

This section describes the Control bit transformation. The synchronization signal (DUx_HSYNC, DUx_VSYNC, DUx_DE) from the LCDC and the low(fixed) bit are reflected in the output data by the selected Control bit.

For the register settings for Control bit conversion, refer to LVDS_0_CTL and LVDS_1_CTL.

The upper part of the figure below shows the three types of synchronization signals input from the LCDC.

As shown in the lower part of the figure below, three types of synchronization signals and Low fixation are reflected in the Control bit.

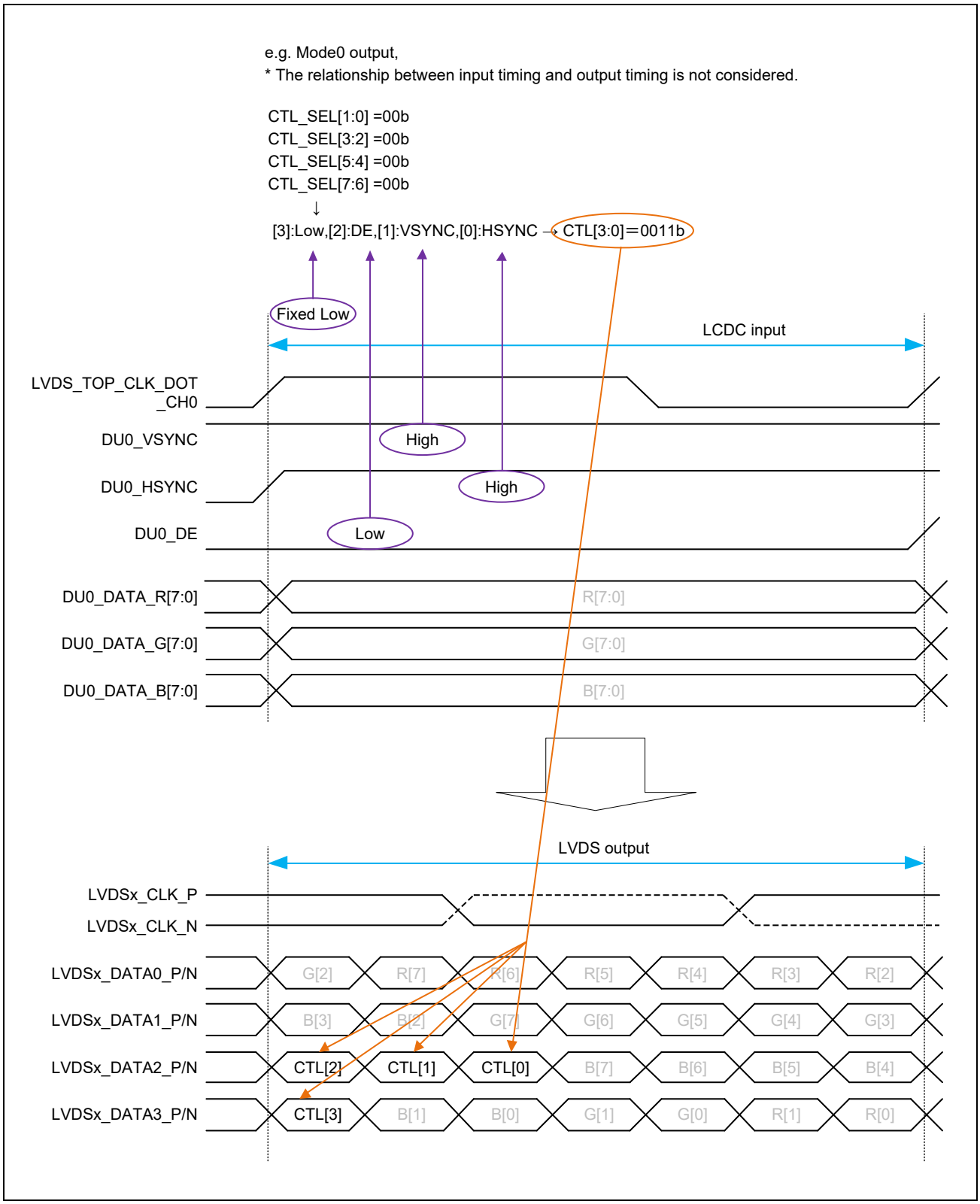


Figure 9.11-14 Control Bit Conversion

9.11.3.7 Output Pattern

Depending on your use case (output pattern), you need to set the LVDS operating mode. The output pattern is a combination of MIPI-DSI and Parallel output. **Table 9.11-7** shows the relationship between LCDC0,1 and LVDS0, 1 in the output pattern and LVDS operating mode, limited to LVDS.

For an image of the LVDS operating mode, refer to **9.11.3.3 LVDS Operating Modes**. Refer to (1) for the setting method.

The color coding of the operating modes is as follows:

- Grey: LVDS module unused
- Blue: Single-link (ch0 only)
- Orange: Single-link (ch1 only)
- Yellow: Single-link (2ch)
- Green: Single-link (Multi)
- Violet: Dual-link

Table 9.11-7 Relationship between Operating Modes and Output Variation Patterns

Output Pattern	1-0	1-1	1-2	1-3	1-4	1-5
LVDS0	LCDC0	—	—	—	—	LCDC0
LVDS1	—	LCDC1	—	—	—	LCDC0

Output Pattern	2-0	2-1	2-2	2-3	2-4	2-5	2-6
LVDS0	LCDC0	LCDC0	LCDC0	LCDC0	LCDC0	LCDC0	—
LVDS1	LCDC0	LCDC1	—	—	LCDC0	LCDC0	—

Output Pattern	3-0	3-1
LVDS0	LCDC0	LCDC0
LVDS1	LCDC0	LCDC0

(1) How to set up

Register settings (CLK_SEL, ST_ON, RST_PHY0_SEL, RST_PHY1_SEL, and PHY_RESET) must be made to determine the clock, reset, and data paths.

Table 9.11-8 shows the relationship between LVDS operating modes and register settings.

Table 9.11-8 List of LVDS Operation Mode Settings

Operation Mode	CLK_SEL	ST_ON	RST_PHY0_SEL	RST_PHY1_SEL	PHY_RESET
Unused	0	0	0	0	0
Single-link (ch0 only)	1	0	1	0	1
Single-link (ch1 only)	1	0	0	1	1
Single-link (2ch)	1	0	1	1	1
Single-link (Multi)	0	0	1	0	1
Dual-link	1	1	1	0	1

Note: If only one LVDS channel is used, turn off the CPG clock gating enable for unused clocks. The target clocks are LVDS_TOP_CLK_CH0, LVDS_TOP_CLK_DOT_CH0 and LVDS_TOP_CLK_CH1, LVDS_TOP_CLK_DOT_CH1. The data is switched along with the associated clock.

9.11.3.8 Control Bit Switching

Switching the Control bit by setting the register (LVDS_0_CTL.CTL_SEL0[7:0] / LVDS_1_CTL.CTL_SEL1[7:0]) of this module.

The output synchronization signal can be assigned to the 4 bits of the Control bit [3:0] included in the output format with a register. The target synchronization signal is VSYNC/HSYNC/DE/Low (Fixed).

The relationship between CTL_SELx[7:0] and the Control bit is as follows.

- CTL_SELx[1:0]: CTL[0]
- CTL_SELx[2:3]: CTL[1]
- CTL_SELx[4:5]: CTL[2]
- CTL_SELx[6:7]: CTL[3]

Table 9.11-9 shows a list of Control bit assignment selection.

Refer to **Figure 9.11-14** Control Bit Conversion for an image of the control bit switching.

Table 9.11-9 Control bit Assignment Selection List

Register Setting Value	CTL[3]	CTL[2]	CTL[1]	CTL[0]
00b	Low	DE	VSYNC	HSYNC
01b	HSYNC	Low	DE	VSYNC
10b	VSYNC	HSYNC	Low	DE
11b	DE	VSYNC	HSYNC	Low

Note: CTL_SEL When “00_00_00_00b” is set to [7:0], it is as follows.
CTL[3] = Low, CTL[2] = DE, CTL[1] = VSYNC, CTL[0] = HSYNC.

9.11.3.9 Output Lane Selection

This module is a function to swap the input and output DataLanes in the register (LVDS_0_CTL.CH_SEL0[7:0] / LVDS_1_CTL.CH_SEL1[7:0]) settings.

Clock Lanes are not swapped. In addition, DataLanes cannot be swapped across channels. Register settings can be set for 4 lanes at once (8 bits).

Table 9.11-10 shows the output lane selection list.

Table 9.11-10 Output Lane Selection List

Each CH_SELx (2bit)	O_Lane3	O_Lane2	O_Lane1	O_Lane0
00b	I_Lane3	I_Lane2	I_Lane1	I_Lane0
01b	I_Lane0	I_Lane3	I_Lane2	I_Lane1
10b	I_Lane1	I_Lane0	I_Lane3	I_Lane2
11b	I_Lane2	I_Lane1	I_Lane0	I_Lane3

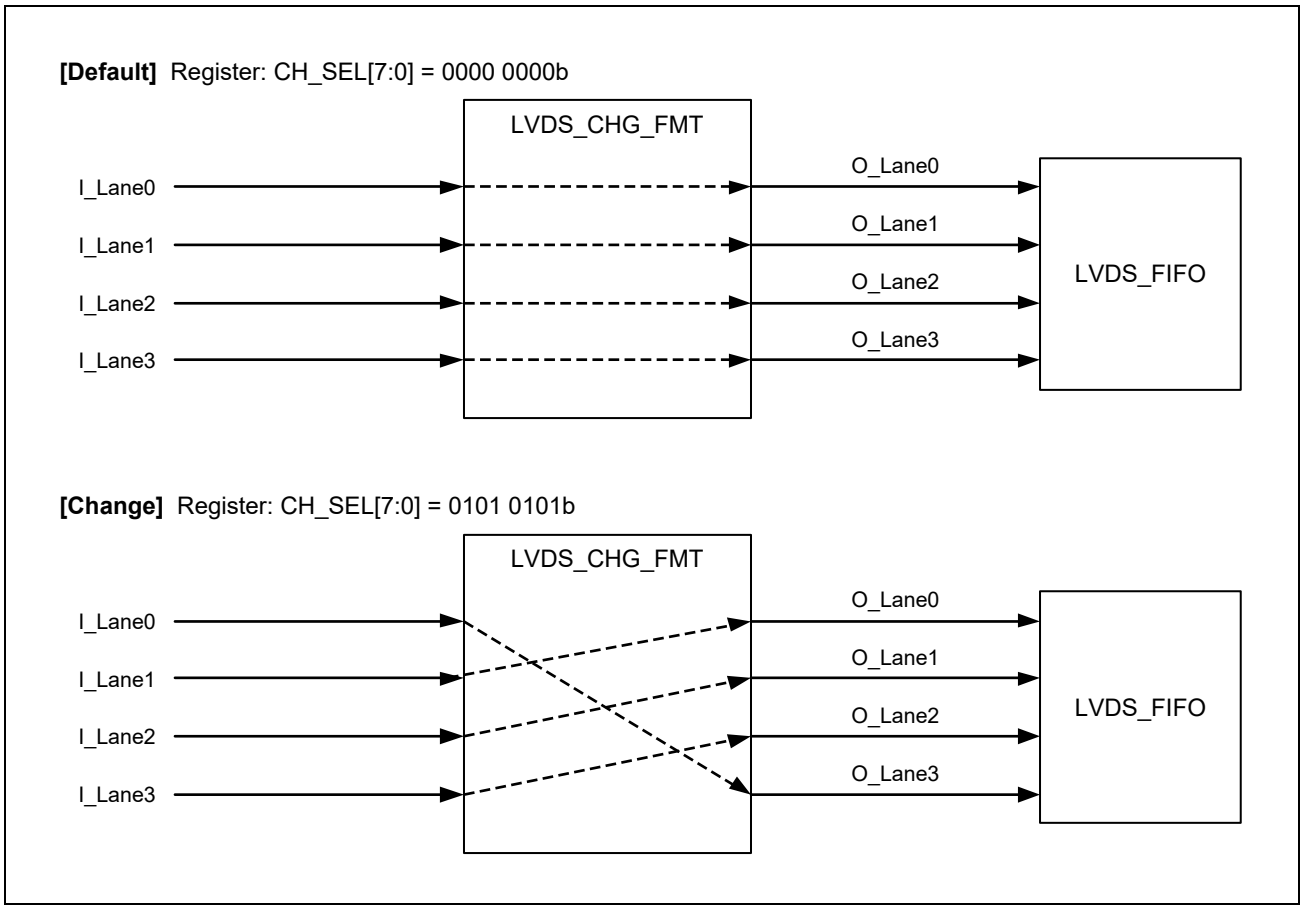


Figure 9.11-15 Image of Data Lane Swapped

9.11.3.10 Dual-link Output Data Swap

This function selects whether to output Even Data or Odd Data of image data to each of LVDS Ch0 and 1.

Even Data/Odd Data is judged and sorted based on the ODDP signal input from the LCDC.

In addition, the register (refer to LVDS_CMN.ST_SWAP) setting allows you to select the output destination of Even data and Odd data. Default outputs Even Data to Ch0 and Odd Data to Ch1.

Figure 9.11-16 shows the timing of Even/Odd Data distribution.

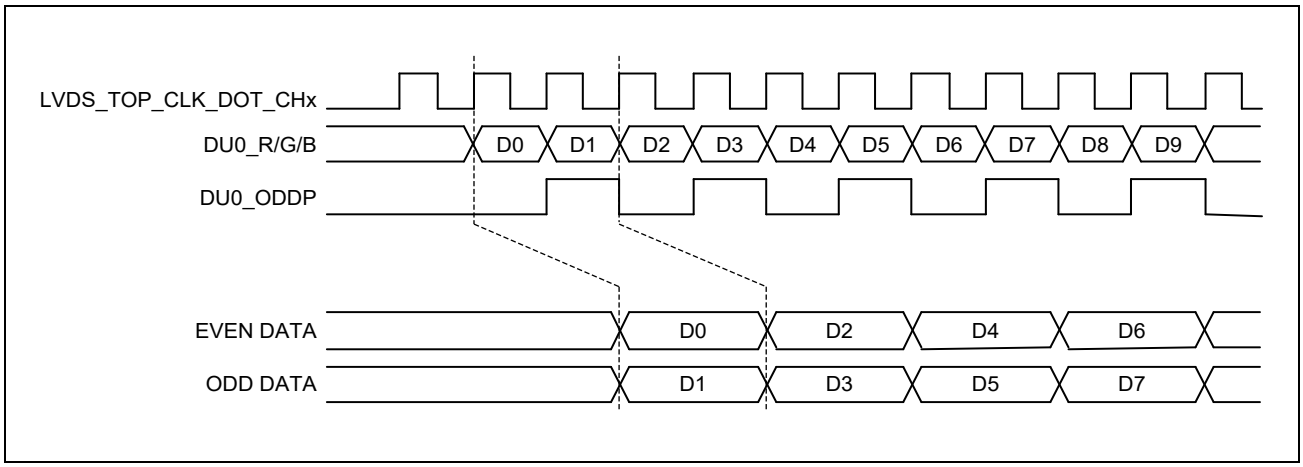


Figure 9.11-16 Timing of Even/Odd Data Distribution

9.11.3.11 Asynchronous FIFO

In this module, an asynchronous FIFO is implemented between the LVDS_CORE section and the PHY.

9.11.3.12 PHY Reset

The reset of the LVDS PHY can be controlled by setting the APB registers.

The PHY reset must be entered synchronously with the PHY clock.

For the control procedure, refer to Power on sequence (**9.11.3.1 Power On Sequence**).

9.11.3.13 PHY Skew Adjustment

This is a function to adjust the LVDS output data and the Skew for each lane of the clock.

Skew adjustment can be adjusted from ±30 ps per lane and can be set up to ±90 ps.

For register settings, refer to LVDS_0_PHY and LVDS_1_PHY.

9.11.3.14 PHY IO Buffer Enable

This function controls the LVDS IO buffer.

The output of the IO buffer of an unused lane can be controlled by setting it to Disable (CH_IO_EN is Normal Operation) in the register.

When Disable (CH_IO_EN is Normal Operation), Hi-Z is output.

For register settings, refer to LVDS_0_PHY and LVDS_1_PHY.

9.11.4 Interrupt

There are no interrupt signals in this module.

9.11.5 Usage Note

This module does not support dynamic register settings.

Set the appropriate register settings at the time of initial setup, and if you want to change the settings, perform the Power On sequence again after performing the Power Off sequence. (No need to turn the power on or off)

The clocks of the unused channels (LVDS_TOP_CLK_CH0, LVDS_TOP_CLK_DOT_CH0, LVDS_TOP_CLK_CH1, LVDS_TOP_CLK_DOT_CH1) should be stopped at the CPG register.

For more information, please refer to **4.4 Clock Pulse Generator (CPG)**.

SECTION 10 ELECTRICAL CHARACTERISTICS

10.1 Electrical Characteristics

This section describes the electrical characteristics of this LSI.

10.1.1 Absolute Maximum Ratings

Permanent damage to the LSI may result if absolute maximum ratings are exceeded.

Table 10.1-1 Absolute Maximum Ratings (1/2)

Unit Name	Item	Symbol	Min.	Max.	Unit
CA55	VDD09_CA55	CA55_V_{DD09}	−0.4	1.2	V
PD_OTHERS	VDD08_OTHERS	OTHERS_V_{DD08}	−0.4	1.2	V
	VDD1833_OTHERS	OTHERS_V_{DD33}	−0.4	3.8	V
	VDD1833_PRE18_OTHERS_1	OTHERS1_PRE18V_{DD1833}	−0.4	2.5	V
PD_AWO	VDD08_AWO	AWO_V_{DD08}	−0.4	1.2	V
	VDD18_AWO	AWO_V_{DD18}	−0.4	2.5	V
	VDD1833_AWO	AWO_V_{DD1833}	−0.4	3.8	V
	VDD1833_PRE18_AWO	AWO_PRE18V_{DD1833}	−0.4	2.5	V
USB3 Ch0	USB30_USVPH	USB30_USV_{PH}	−0.4	2.5	V
	USB30_USVPTX	USB30_USV_{PTX}	−0.4	1.2	V
	USB30_USVDD18	USB30_USV_{DD18}	−0.4	2.5	V
USB2 Ch0	USB20_USVDD18	USB20_USV_{DD18}	−0.4	2.5	V
USB2 Ch1	USB21_USVDD18	USB21_USV_{DD18}	−0.4	2.5	V
USB3 Ch0 (for USB2), USB2 Ch0, USB2 Ch1 common	USB2_USVDD33	USB2_USV_{DD33}	−0.4	3.8	V
	USB2_USDVDD	USB2_$USDV_{DD}$	−0.4	1.2	V
xSPI	VDD1833_XSPI	XSPI_V_{DD1833}	−0.4	3.8	V
SD/MMC Ch0	VDD1833_SD0	SD0_V_{DD1833}	−0.4	3.8	V
SD/MMC Ch1	VDD1833_SD1	SD1_V_{DD1833}	−0.4	3.8	V
SD Ch2	VDD1833_SD2	SD2_V_{DD1833}	−0.4	3.8	V
GBETH Ch0	VDD1833_ET0	ET0_V_{DD1833}	−0.4	3.8	V
GBETH Ch1	VDD1833_ET1	ET1_V_{DD1833}	−0.4	3.8	V
SD/MMC Ch0, SD/MMC Ch1, SD Ch2, GBETH Ch0, GBETH Ch1 common	VDD1833_PRE18_OTHERS_2	OTHERS2_PRE18V_{DD1833}	−0.4	2.5	V
OTP	OTPVDD18	OTP_V_{DD18}	−0.4	2.5	V
LPDDR4/LPDDR4X Ch0	DDR0_VDDQ	DDR0_V_{DDQ}	−0.4	1.5	V
	DDR0_VDDQLP	DDR0_V_{DDQLP}	−0.4	1.5	V
	DDR0_VAA	DDR0_V_{AA}	−0.4	2.5	V

Table 10.1-1 Absolute Maximum Ratings (2/2)

Unit Name	Item	Symbol	Min.	Max.	Unit
MIPI CSI Ch0	CSI0_MSVDD18	CSI0_MSV_{DD18}	−0.4	2.5	V
	CSI0_MSVDD0P8	CSI0_MSV_{DD0P8}	−0.4	1.2	V
MIPI DSI	DSI_VDD0P8	DSI_V_{DD0P8}	−0.4	1.2	V
	DSI_VDD12	DSI_V_{DD12}	−0.4	2.5	V
	DSI_VDD18	DSI_V_{DD18}	−0.4	2.5	V
PCIE	PCIE_VCC18ACMN	PCIE_$V_{CC18ACMN}$	−0.4	2.5	V
	PCIE_VCC18AL01	PCIE_$V_{CC18AL01}$	−0.4	2.5	V
	PCIE_VCC08AL01	PCIE_$V_{CC08AL01}$	−0.4	1.2	V
I3C	VDD1218_I3C	I3C_V_{DD1218}	−0.4	2.5	V
ADC	ADAVDD18	$ADAV_{DD18}$	−0.4	2.5	V
LVDS Ch0, LVDS Ch1 common	LVDS_VCCQ	$LVDSV_{DD18}$	−0.4	2.5	V
CPG	PLVDD_PLLCM33	PLLCM33_PLV_{DD}	−0.4	2.5	V
	PLVDD_PLLCA55	PLLCA55_PLV_{DD}	−0.4	2.5	V
	PLVDD_OTHERS	PLLOTHERS_PLV_{DD}	−0.4	2.5	V
	PLDVDD08_PLLCM33	PLLCM33_$PLDV_{DD08}$	−0.4	1.2	V
	PLDVDD09_PLLCA55	PLLCA55_$PLDV_{DD09}$	−0.4	1.2	V
	PLDVDD08_OTHERS	PLLOTHERS_$PLDV_{DD08}$	−0.4	1.2	V
PWC	VDD18_PWC	PWC_V_{DD18}	−0.4	2.5	V
—	Input voltage (0.6-V I/O)	V_{in06}	−0.4	DDR0_V_{DDQLP} + 0.3*1	V
—	Input voltage (1.1-V I/O)	V_{in11}	−0.4	DDR0_V_{DDQ} + 0.3*1	V
—	Input voltage (1.2-V I/O)	V_{in12}	−0.4	V_{12} + 0.3*2	V
—	Input voltage (1.8-V I/O)	V_{in18}	−0.4	V_{18} + 0.3*3	V
—	Input voltage (1.8-V I/O (3.3-V tolerant))*4	V_{in18_tol}	−0.4	3.6	V
—	Input voltage (3.3-V I/O)	V_{in33}	−0.4	V_{33} + 0.3*5	V
—	Analog input voltage (ADC I/O)	V_{ain18}	0	$ADAV_{DD18}$	V
—	Junction temperature	T_j	−40	125	°C
—	Storage temperature	T_{stg}	−40	150	°C

Note 1. The voltage to be applied must be within the absolute maximum rating (1.5 V).

Note 2. The voltage to be applied must be within the absolute maximum rating (2.5 V). V_{12} indicates the power supply voltage for 1.2-V I/O pins.

Note 3. The voltage to be applied must be within the absolute maximum rating (2.5 V). V_{18} indicates the power supply voltage for 1.8-V I/O pins. When 1.8-V is used for the 3.3/1.8-V switching I/O, this specification is applied.

Note 4. PS0 to PS3, PJ0 to PJ4

Note 5. The voltage to be applied must be within the absolute maximum rating (3.8 V). V_{33} indicates the power supply voltage for 3.3-V I/O pins. When 3.3-V is used for the 3.3/1.8-V switching I/O, this specification is applied.

10.1.2 Recommended Operating Range

Table 10.1-2 Recommended Operating Range (1/2)

Unit Name	Item	Symbol	Min.	Typ.	Max.	Unit	Note
CA55	VDD09_CA55	CA55_V_{DD09}	0.86	0.9	0.94	V	0.9 V: OD*1
			0.76	0.8	0.84	V	0.8 V: ND*1
PD_OTHERS	VDD08_OTHERS	OTHERS_V_{DD08}	0.76	0.8	0.84	V	*2
	VDD1833_OTHERS	OTHERS_V_{DD1833}	3.14	3.3	3.46	V	
			1.71	1.8	1.89	V	
	VDD1833_PRE18_OTHERS_1	OTHERS1_PRE18V_{DD1833}	1.71	1.8	1.89	V	
PD_AWO	VDD08_AWO	AWO_V_{DD08}	0.76	0.8	0.84	V	
	VDD18_AWO	AWO_V_{DD18}	1.71	1.8	1.89	V	
	VDD1833_AWO	AWO_V_{DD1833}	3.14	3.3	3.46	V	
			1.71	1.8	1.89	V	
	VDD1833_PRE18_AWO	AWO_PRE18V_{DD1833}	1.71	1.8	1.89	V	
USB3 Ch0	USB30_USVPH	USB30_USV_{PH}	1.71	1.8	1.89	V	
	USB30_USVPTX	USB30_USV_{PTX}	0.76	0.8	0.84	V	
	USB30_USVDD18	USB30_USV_{DD18}	1.71	1.8	1.89	V	
USB2 Ch0	USB20_USVDD18	USB20_USV_{DD18}	1.71	1.8	1.89	V	
USB2 Ch1	USB21_USVDD18	USB21_USV_{DD18}	1.71	1.8	1.89	V	
USB3 Ch0 (for USB2), USB2 Ch0, USB2 Ch1 common	USB2_USVDD33	USB2_USV_{DD33}	3.14	3.3	3.46	V	
	USB2_USDVDD	USB2_$USDV_{DD}$	0.76	0.8	0.84	V	
xSPI	VDD1833_XSPI	XSPI_V_{DD1833}	3.14	3.3	3.46	V	
			1.71	1.8	1.89	V	
SD/MMC Ch0	VDD1833_SD0	SD0_V_{DD1833}	3.14	3.3	3.46	V	
			1.71	1.8	1.89	V	
SD/MMC Ch1	VDD1833_SD1	SD1_V_{DD1833}	3.14	3.3	3.46	V	
			1.71	1.8	1.89	V	
SD Ch2	VDD1833_SD2	SD2_V_{DD1833}	3.14	3.3	3.46	V	
			1.71	1.8	1.89	V	
GBETH Ch0	VDD1833_ET0	ET0_V_{DD1833}	3.14	3.3	3.46	V	
			1.71	1.8	1.89	V	
GBETH Ch1	VDD1833_ET1	ET1_V_{DD1833}	3.14	3.3	3.46	V	
			1.71	1.8	1.89	V	
SD/MMC Ch0, SD/MMC Ch1, SD Ch2, GBETH Ch0, GBETH Ch1 common	VDD1833_PRE18_OTHERS_2	OTHERS2_PRE18V_{DD1833}	1.71	1.8	1.89	V	
OTP	OTPVDD18	OTP_V_{DD18}	1.71	1.8	1.89	V	
LPDDR4/LPDDR4X Ch0	DDR0_VDDQ	DDR0_V_{DDQ}	1.06	1.1	1.17	V	
	DDR0_VDDQLP	DDR0_V_{DDQLP}	0.57	0.6	0.65	V	0.6 V: LPDDR4X
			1.06	1.1	1.17	V	1.1 V: LPDDR4
	DDR0_VAA	DDR0_V_{AA}	1.71	1.8	1.89	V	

Table 10.1-2 Recommended Operating Range (2/2)

Unit Name	Item	Symbol	Min.	Typ.	Max.	Unit	Note
MIPI CSI Ch0	CSI0_MSVDD18	CSI0_MSV_{DD18}	1.71	1.8	1.89	V	
	CSI0_MSVDD0P8	CSI0_MSV_{DD0P8}	0.76	0.8	0.84	V	
MIPI DSI	DSI_VDD0P8	DSI_V_{DD0P8}	0.76	0.8	0.84	V	
	DSI_VDD12	DSI_V_{DD12}	1.14	1.2	1.26	V	
	DSI_VDD18	DSI_V_{DD18}	1.71	1.8	1.89	V	
PCIE	PCIE_VCC18ACMN	PCIE_$V_{CC18ACMN}$	1.71	1.8	1.89	V	
	PCIE_VCC18AL01	PCIE_$V_{CC18AL01}$	1.71	1.8	1.89	V	
	PCIE_VCC08AL01	PCIE_$V_{CC08AL01}$	0.76	0.8	0.84	V	
I3C	VDD1218_I3C	I3C_V_{DD1218}	1.71	1.8	1.89	V	
			1.14	1.2	1.26	V	
ADC	ADAVDD18	$ADAV_{DD18}$	1.71	1.8	1.89	V	
LVDS Ch0, LVDS Ch1 common	LVDS_VCCQ	$LVDSV_{DD18}$	1.71	1.8	1.89	V	
CPG	PLVDD_PLLCM33	PLLCM33_PLV_{DD}	1.71	1.8	1.89	V	
	PLVDD_PLLCA55	PLLCA55_PLV_{DD}	1.71	1.8	1.89	V	
	PLVDD_OTHERS	PLLOTHERS_PLV_{DD}	1.71	1.8	1.89	V	
	PLDVDD08_PLLCM33	PLLCM33_$PLDV_{DD08}$	0.76	0.8	0.84	V	
	PLDVDD09_PLLCA55	PLLCA55_$PLDV_{DD09}$	0.86	0.9	0.94	V	0.9 V: OD*1
			0.76	0.8	0.84	V	0.8 V: ND*1
	PLDVDD08_OTHERS	PLLOTHERS_$PLDV_{DD08}$	0.76	0.8	0.84	V	
PWC	VDD18_PWC	PWC_V_{DD18}	1.71	1.8	1.89	V	

Note 1. OD: Over drive (up to 1.8-GHz operation frequency)
ND: Normal drive (up to 1.1-GHz operation frequency)

Note 2. To avoid the possibility of noise, separating this power supply from other power supply terminals is recommended.

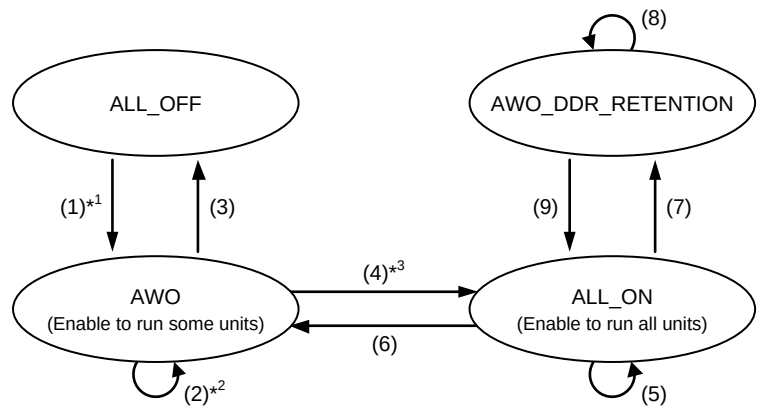
10.1.3 Power-On/Off Sequence

10.1.3.1 CM33 Boot Mode (State Diagram)

The state diagram of CM33 cold boot is shown in **Figure 10.1-1**. The boot mode states (1) to (9) refer to the sequence of (1) to (9) in **Figure 10.1-2** to **Figure 10.1-5**.

For the details of boot mode, refer to **4.4.7 Control of Booting**.

For the details of power mode, refer to **4.5.3 Power Domains**.



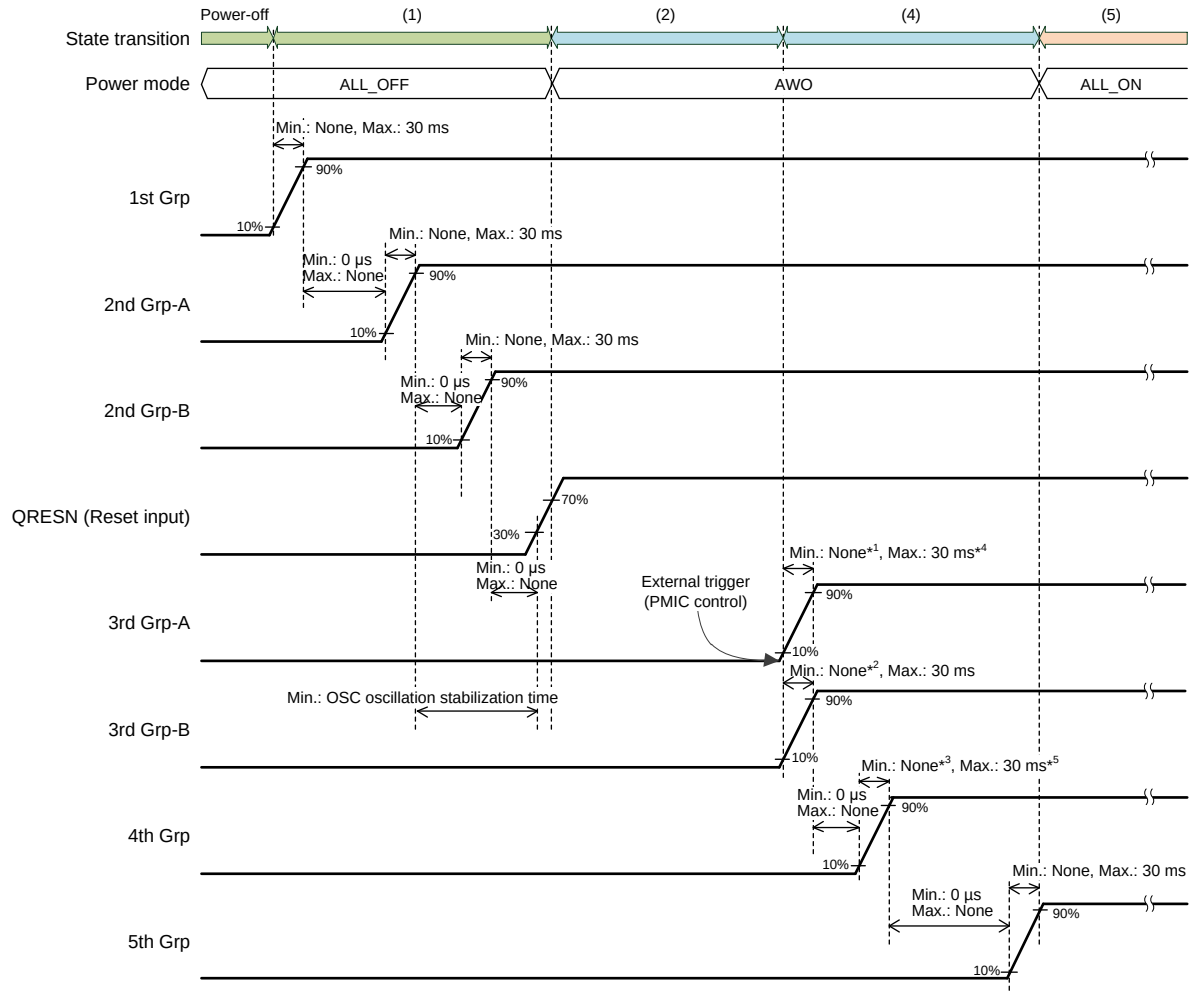
Note 1. CM33 cold boot sequence.

Note 2. CM33 operates the system managements.

Note 3. When the power mode switches from AWO to ALL_ON depending on the user system, CM33 sets the LSI registers and PMIC registers to supply power to the OTHERS domain.

Figure 10.1-1 CM33 Boot State Diagram

10.1.3.2 CM33 Boot Mode (Power-On)



1st Grp power supply
0.8 V
VDD08_AWO
PLDVDD08_PLLCM33

2nd Grp-A power supply
1.8 V
ADAVDD18
OTPVDD18
VDD1833_PRE18_AWO
VDD18_AWO
VDD18_PWC
PLVDD_PLLCM33

(When used as 1.8 V)
VDD1833_AWO
VDD1833_XSPI

2nd Grp-B power supply
(When used as 3.3 V)
VDD1833_AWO
VDD1833_XSPI

3rd Grp-A power supply
0.8 V
USB2_USDVDD
USB30_USVPTX
PCIE_VCC08AL01
DSI_VDD0P8
CSI0_MSVDD0P8
PLDVDD08_OTHERS
VDD08_OTHERS

0.8/0.9 V
VDD09_CA55
PLDVDD09_PLLCA55

3rd Grp-B power supply
1.1 V
DDR0_VDDQ

0.6/1.1 V
DDR0_VDDQLP

4th Grp power supply
1.8 V
VDD1833_PRE18_OTHERS_1
VDD1833_PRE18_OTHERS_2
DSI_VDD18
CSI0_MSVDD18
LVDS_VCCQ
USB20_USVDD18
USB21_USVDD18
USB30_USVPH
USB30_USVDD18
DDR0_VAA
PLVDD_PLLCA55
PLVDD_OTHERS
PCIE_VCC18ACMN
PCIE_VCC18AL01

(When used as 1.8 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

1.2/1.8 V
VDD1218_I3C

3.3 V
USB2_USVDD33

5th Grp power supply
1.2 V
DSI_VDD12
(When used as 3.3 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

Note 1. VDD08_OTHERS, USB30_USVPTX, USB2_USDVDD: Min. 10 µs

Note 2. DDR0_VDDQ, DDR0_VDDQLP (1.1 V): Min. 180 µs
DDR0_VDDQLP (0.6 V): Min. 100 µs

Note 3. VDD1833_PRE18_OTHERS_1: Min. 10 µs
DDR0_VAA: Min. 290 µs
USB2_USVDD33: Min.30 µs
USB20_USVDD18, USB21_USVDD18, USB30_USVDD18: Min. 20 µs
USB30_USVPH: Min. 10 µs

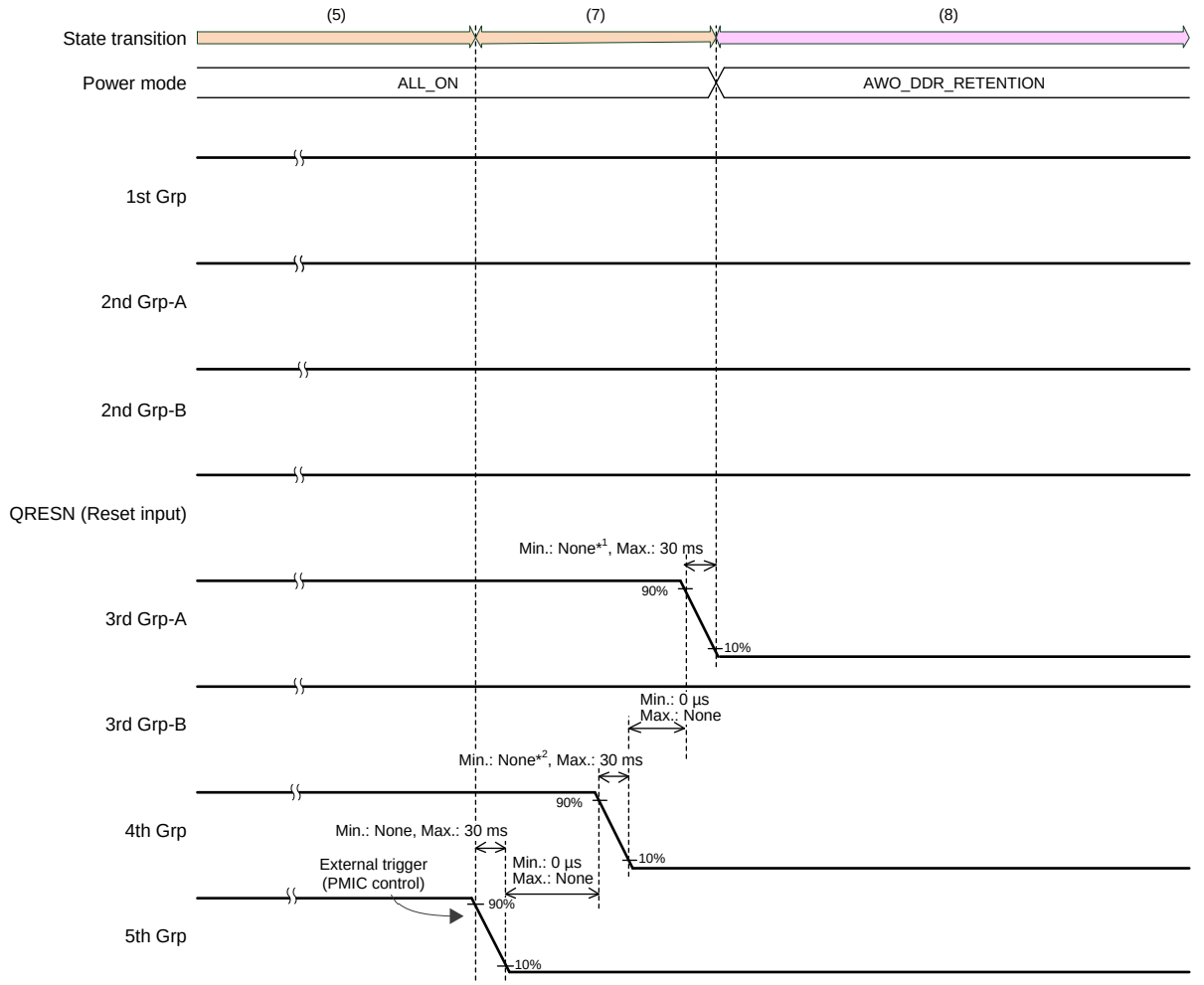
Note 4. USB2_USDVDD: Max. 10 ms

Note 5. USB20_USVDD18, USB21_USVDD18, USB30_USVDD18, USB2_USVDD33: Max. 10 ms

Note: The clock stabilization time depends on the board design. Make the setting according to the results of evaluation.

Figure 10.1-2 Power-On Sequence (CM33 Boot)

10.1.3.3 CM33 Boot Mode (DDR Retention Enter)



1st Grp power supply
0.8 V
VDD08_AWO
PLDVDD08_PLLCM33

2nd Grp-A power supply
1.8 V
ADAVDD18
OTPVDD18
VDD1833_PRE18_AWO
VDD18_AWO
VDD18_PWC
PLVDD_PLLCM33

(When used as 1.8 V)
VDD1833_AWO
VDD1833_XSPI

2nd Grp-B power supply
(When used as 3.3 V)
VDD1833_AWO
VDD1833_XSPI

3rd Grp-A power supply
0.8 V
USB2_USDVDD
USB30_USVPTX
PCIE_VCC08AL01
DSI_VDD0P8
CSI0_MSVDD0P8
PLDVDD08_OTHERS
VDD08_OTHERS

0.8/0.9 V
VDD09_CA55
PLDVDD09_PLLCA55

3rd Grp-B power supply
1.1 V
DDR0_VDDQ

0.6/1.1 V
DDR0_VDDQLP

4th Grp power supply
1.8 V
VDD1833_PRE18_OTHERS_1
VDD1833_PRE18_OTHERS_2
DSI_VDD18
CSI0_MSVDD18
LVDS_VCCQ
USB20_USVDD18
USB21_USVDD18
USB30_USVPH
USB30_USVDD18
DDR0_VAA
PLVDD_PLLCA55
PLVDD_OTHERS
PCIE_VCC18ACMN
PCIE_VCC18AL01

(When used as 1.8 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

1.2/1.8 V
VDD1218_I3C

3.3 V
USB2_USVDD33

5th Grp power supply
1.2 V
DSI_VDD12
(When used as 3.3 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

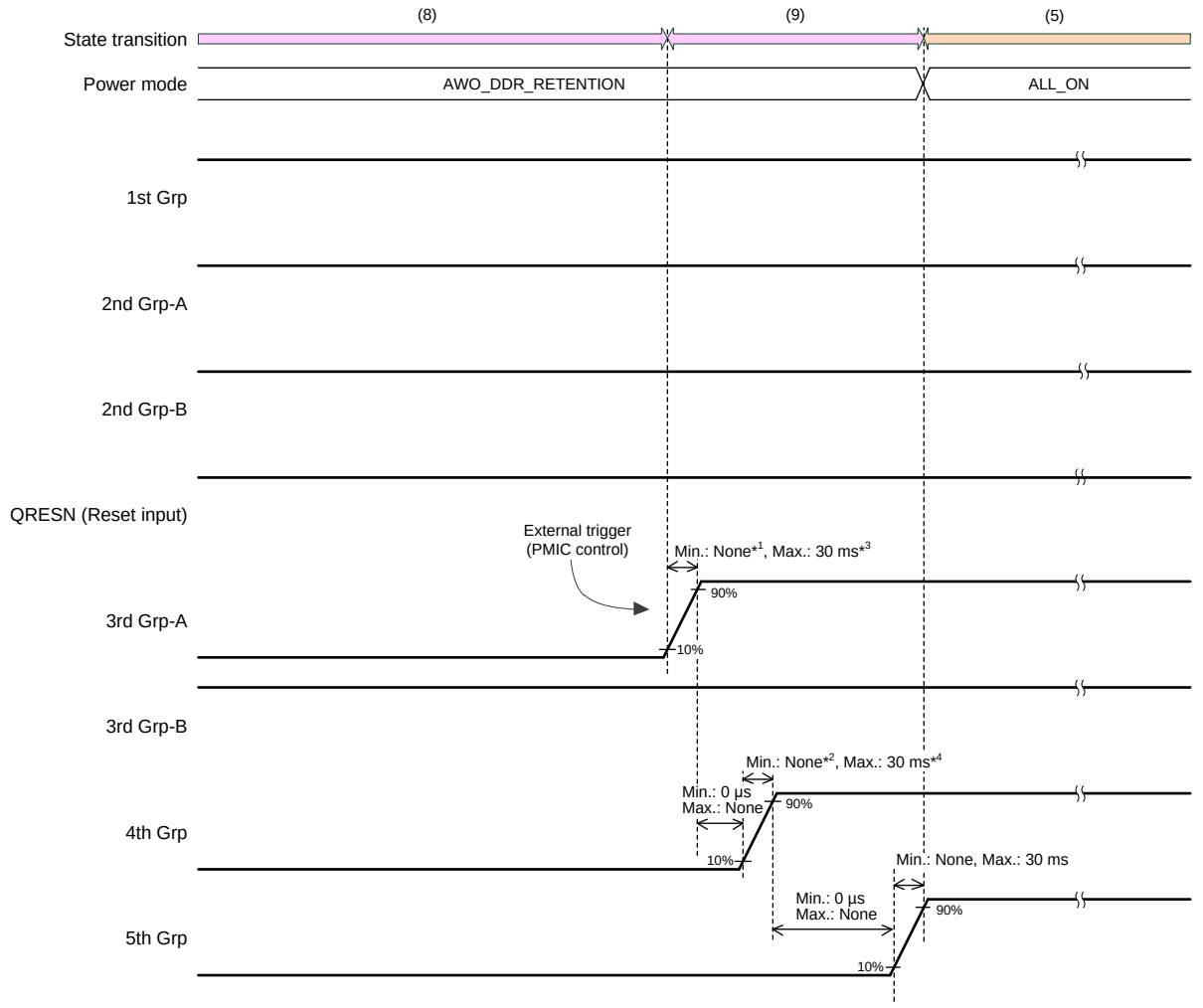
Note 1. VDD08_OTHERS: Min. 10 µs

Note 2. VDD1833_PRE18_OTHERS_1: Min. 10 µs

Note: The clock stabilization time depends on the board design. Make the setting according to the results of evaluation.

Figure 10.1-3 DDR Retention Enter Sequence (CM33 Boot Mode)

10.1.3.4 CM33 Boot Mode (DDR Retention Exit)



1st Grp power supply
0.8 V
VDD08_AWO
PLDVDD08_PLLCM33

2nd Grp-A power supply
1.8 V
ADAVDD18
OTPVDD18
VDD1833_PRE18_AWO
VDD18_AWO
VDD18_PWC
PLVDD_PLLCM33

(When used as 1.8 V)
VDD1833_AWO
VDD1833_XSPI

2nd Grp-B power supply
(When used as 3.3 V)
VDD1833_AWO
VDD1833_XSPI

3rd Grp-A power supply
0.8 V
USB2_USDVDD
USB30_USVPTX
PCIE_VCC08AL01
DSI_VDD0P8
CSI0_MSVDD0P8
PLDVDD08_OTHERS
VDD08_OTHERS

0.8/0.9 V
VDD09_CA55
PLDVDD09_PLLCA55

3rd Grp-B power supply
1.1 V
DDR0_VDDQ

0.6/1.1 V
DDR0_VDDQLP

4th Grp power supply
1.8 V
VDD1833_PRE18_OTHERS_1
VDD1833_PRE18_OTHERS_2
DSI_VDD18
CSI0_MSVDD18
LVDS_VCCQ
USB20_USVDD18
USB21_USVDD18
USB30_USVPH
USB30_USVDD18
DDR0_VAA
PLVDD_PLLCA55
PLVDD_OTHERS
PCIE_VCC18ACMN
PCIE_VCC18AL01

(When used as 1.8 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

1.2/1.8 V
VDD1218_I3C

3.3 V
USB2_USVDD33

5th Grp power supply
1.2 V
DSI_VDD12
(When used as 3.3 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

Note 1. VDD08_OTHERS, USB30_USVPTX, USB2_USDVDD: Min. 10 µs

Note 2. VDD1833_PRE18_OTHERS_1: Min. 10 µs
DDR0_VAA: Min. 290 µs
USB2_USVDD33: Min.30 µs
USB20_USVDD18, USB21_USVDD18, USB30_USVDD18: Min. 20 µs
USB30_USVPH: Min. 10 µs

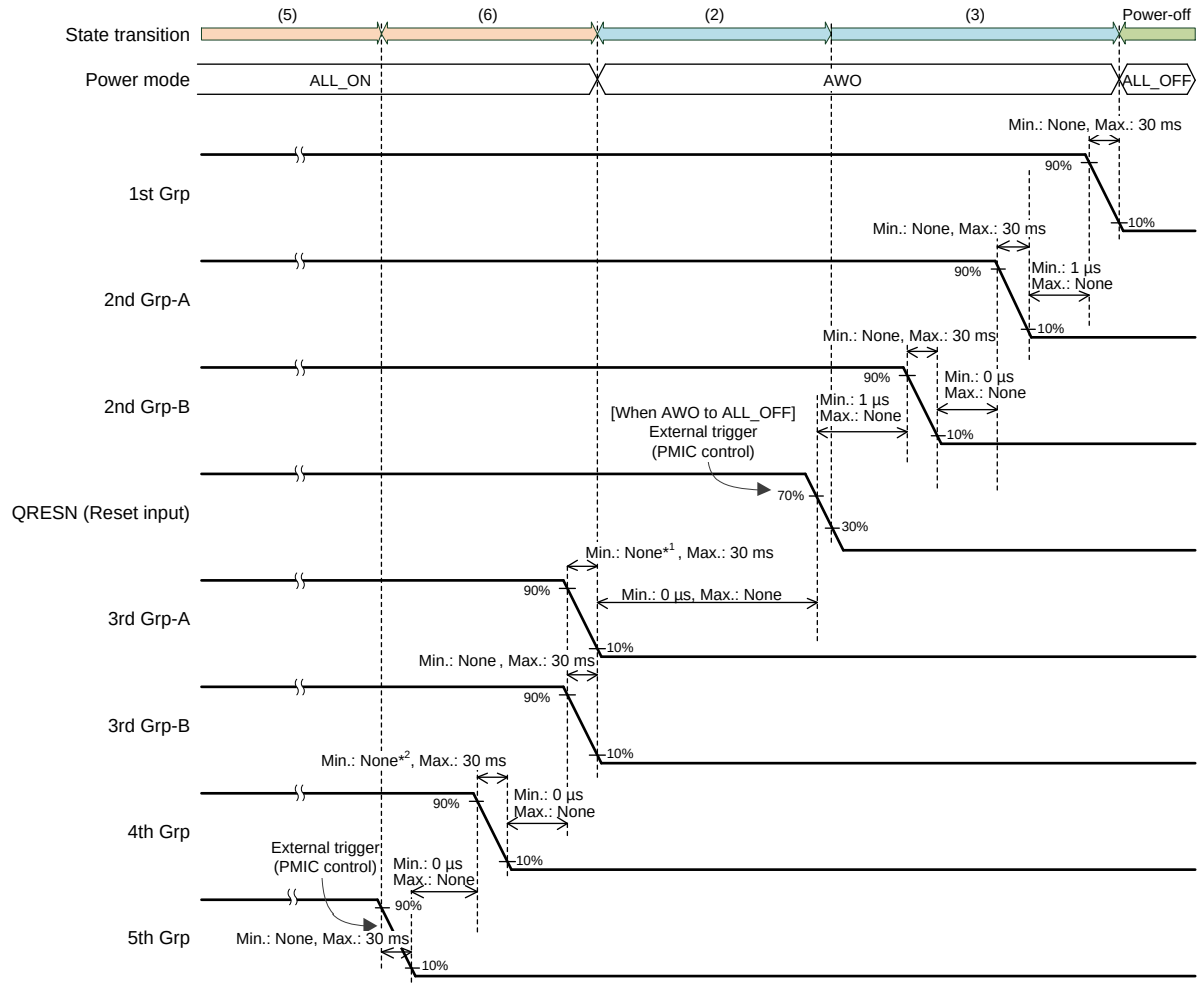
Note 3. USB2_USDVDD: Max. 10 ms

Note 4. USB20_USVDD18, USB21_USVDD18, USB30_USVDD18, USB2_USVDD33: Max. 10 ms

Note: The clock stabilization time depends on the board design. Make the setting according to the results of evaluation.

Figure 10.1-4 DDR Retention Exit Sequence (CM33 Boot Mode)

10.1.3.5 CM33 Boot Mode (Power-Off)



1st Grp power supply
0.8 V
VDD08_AWO
PLDVDD08_PLLCM33

2nd Grp-A power supply
1.8 V
ADAVDD18
OTPVDD18
VDD1833_PRE18_AWO
VDD18_AWO
VDD18_PWC
PLVDD_PLLCM33

(When used as 1.8 V)
VDD1833_AWO
VDD1833_XSPI

3rd Grp-A power supply
0.8 V
USB2_USDVDD
USB30_USVPTX
PCIE_VCC08AL01
DSI_VDD0P8
CSI0_MSVDD0P8
PLDVDD08_OTHERS
VDD08_OTHERS

0.8/0.9 V
VDD09_CA55
PLDVDD09_PLLCA55

4th Grp power supply
1.8 V
VDD1833_PRE18_OTHERS_1
VDD1833_PRE18_OTHERS_2
DSI_VDD18
CSI0_MSVDD18
LVDS_VCCQ
USB20_USVDD18
USB21_USVDD18
USB30_USVPH
USB30_USVDD18
DDR0_VAA
PLVDD_PLLCA55
PLVDD_OTHERS
PCIE_VCC18ACMN
PCIE_VCC18AL01

(When used as 1.8 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

1.2/1.8 V
VDD1218_I3C

3.3 V
USB2_USVDD33

5th Grp power supply
1.2 V
DSI_VDD12
(When used as 3.3 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

2nd Grp-B power supply
(When used as 3.3 V)
VDD1833_AWO
VDD1833_XSPI

3rd Grp-B power supply
1.1 V
DDR0_VDDQ

0.6/1.1 V
DDR0_VDDQLP

Note 1. VDD08_OTHERS: Min. 10 µs

Note 2. VDD1833_PRE18_OTHERS_1: Min. 10 µs

Note: The clock stabilization time depends on the board design. Make the setting according to the results of evaluation.

Note: The fall time of each power supply must be compliant with the maximum 30 ms regulation.

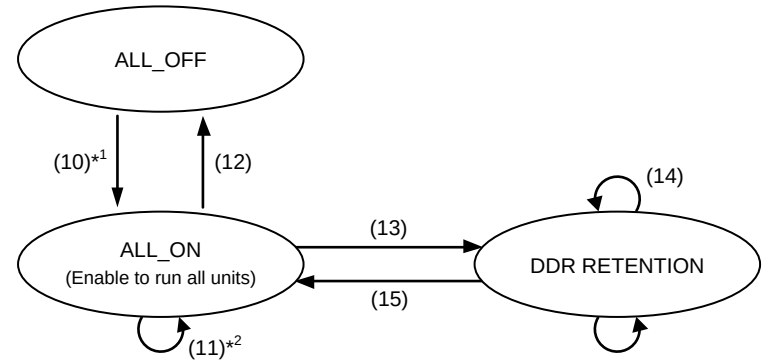
Figure 10.1-5 Power-Off Sequence (CM33 Boot Mode)

10.1.3.6 CA55 Boot Mode (State Diagram)

The state diagram of CA55 cold boot is shown in **Figure 10.1-6**. The boot mode states (10) to (15) refer to the sequence of (10) to (15) in **Figure 10.1-7** to **Figure 10.1-10**.

For the details of boot mode, refer to **4.4.7 Control of Booting**.

For the details of power mode, refer to **4.5.3 Power Domains**.

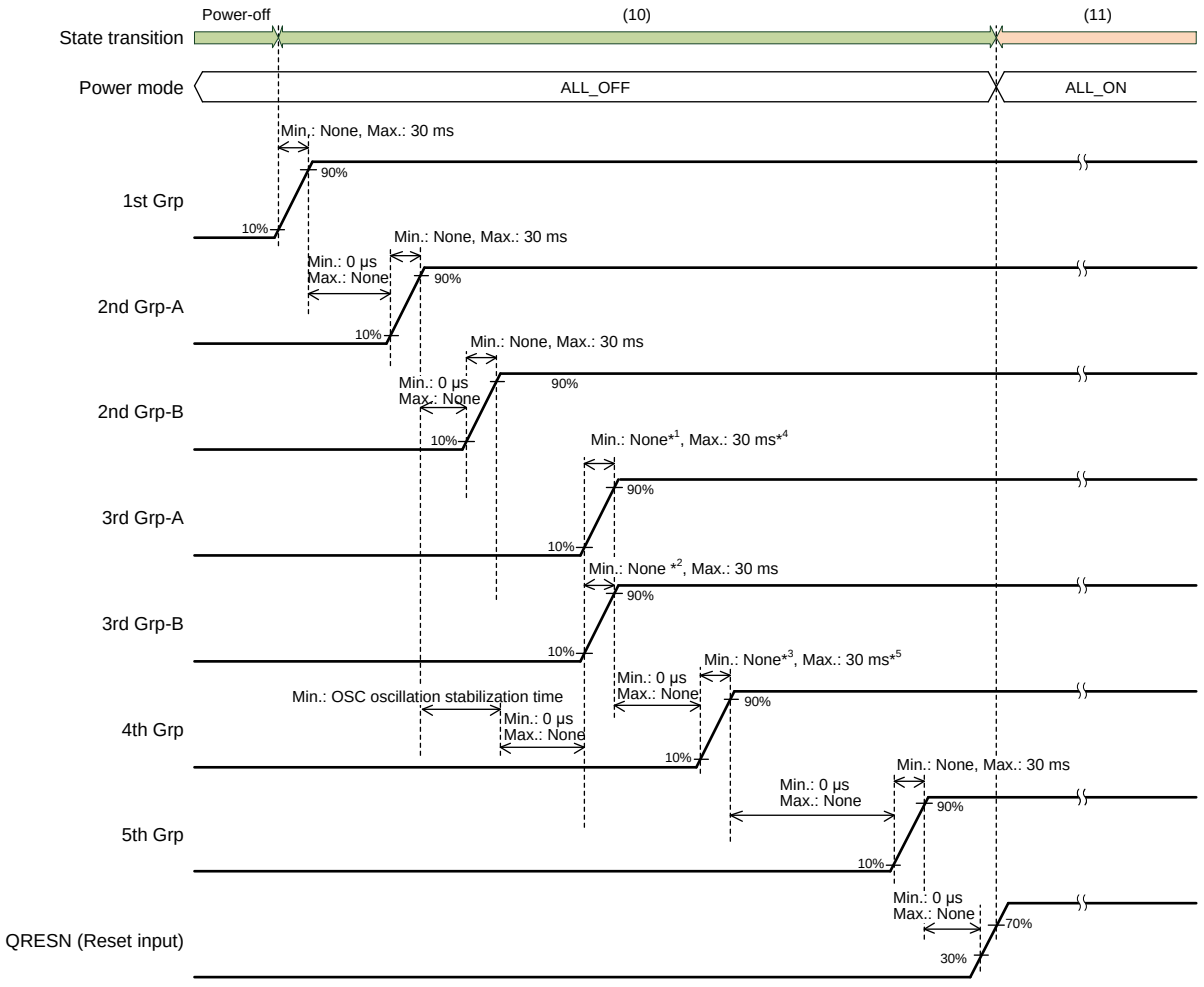


Note 1. CA55 cold boot sequence.

Note 2. CA55 operates the system managements.

Figure 10.1-6 CA55 Boot State Diagram

10.1.3.7 CA55 Boot Mode (Power-On)



1st Grp power supply
0.8 V
VDD08_AWO
PLDVDD08_PLLCM33

2nd Grp-A power supply
1.8 V
ADAVDD18
OTPVDD18
VDD1833_PRE18_AWO
VDD18_AWO
VDD18_PWC
PLVDD_PLLCM33

(When used as 1.8 V)
VDD1833_AWO
VDD1833_XSPI

2nd Grp-B power supply
(When used as 3.3 V)
VDD1833_AWO
VDD1833_XSPI

3rd Grp-A power supply
0.8 V
USB2_USDVDD
USB30_USVPTX
PCIE_VCC08AL01
DSI_VDD0P8
CSI0_MSVDD0P8
PLDVDD08_OTHERS
VDD08_OTHERS

0.8/0.9 V
VDD09_CA55
PLDVDD09_PLLCA55

3rd Grp-B power supply
1.1 V
DDR0_VDDQ

0.6/1.1 V
DDR0_VDDQLP

4th Grp power supply
1.8 V
VDD1833_PRE18_OTHERS_1
VDD1833_PRE18_OTHERS_2
DSI_VDD18
CSI0_MSVDD18
LVDS_VCCQ
USB20_USVDD18
USB21_USVDD18
USB30_USVPH
USB30_USVDD18
DDR0_VAA
PLVDD_PLLCA55
PLVDD_OTHERS
PCIE_VCC18ACMN
PCIE_VCC18AL01

(When used as 1.8 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

1.2/1.8 V
VDD1218_I3C

3.3 V
USB2_USVDD33

5th Grp power supply
1.2 V
DSI_VDD12
(When used as 3.3 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

Note 1. VDD08_OTHERS, USB30_USVPTX, USB2_USDVDD: Min. 10 µs

Note 2. DDR0_VDDQ, DDR0_VDDQLP (1.1 V): Min. 180 µs

Note 3. VDD1833_PRE18_OTHERS_1: Min. 10 µs
DDR0_VAA: Min. 290 µs
USB2_USVDD33: Min.30 µs
USB20_USVDD18, USB21_USVDD18, USB30_USVDD18: Min. 20 µs
USB30_USVPH: Min. 10 µs

Note 4. USB2_USDVDD: Max. 10 ms

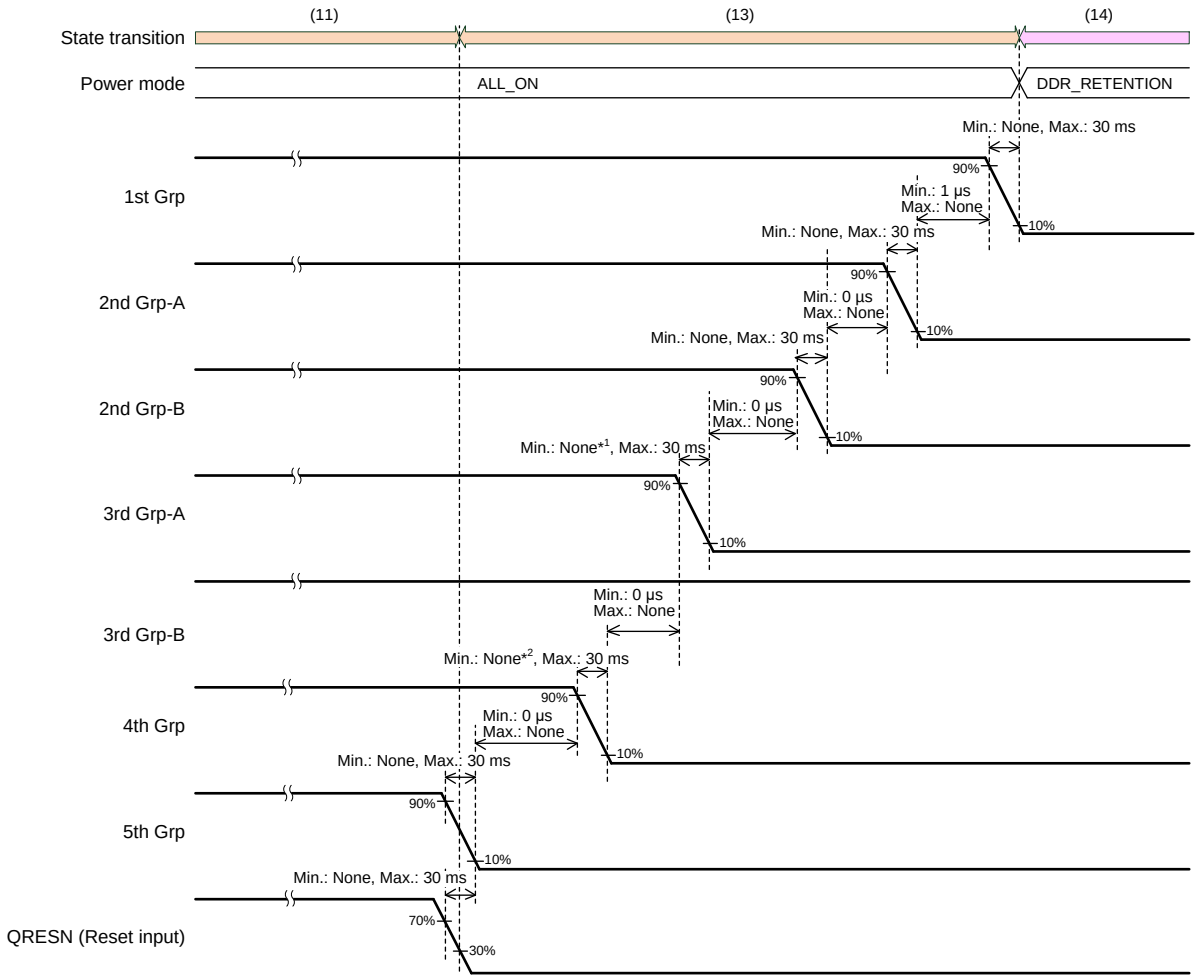
Note 5. USB20_USVDD18, USB21_USVDD18, USB30_USVDD18, USB2_USVDD33: Max. 10 ms

Note: The clock stabilization time depends on the board design. Make the setting according to the results of evaluation.

Note: The rise time of each power supply must be compliant with the maximum 30 ms regulation.

Figure 10.1-7 Power-On Sequence (CA55 Boot Mode)

10.1.3.8 CA55 Boot Mode (DDR Retention Enter)



1st Grp power supply
0.8 V
VDD08_AWO
PLDVDD08_PLLCM33

2nd Grp-A power supply
1.8 V
ADAVDD18
OTPVDD18
VDD1833_PRE18_AWO
VDD18_AWO
VDD18_PWC
PLVDD_PLLCM33

(When used as 1.8 V)
VDD1833_AWO
VDD1833_XSPI

2nd Grp-B power supply
(When used as 3.3 V)
VDD1833_AWO
VDD1833_XSPI

3rd Grp-A power supply
0.8 V
USB2_USDVDD
USB30_USVPTX
PCIE_VCC08AL01
DSI_VDD0P8
CSI0_MSVDD0P8
PLDVDD08_OTHERS
VDD08_OTHERS

0.8/0.9 V
VDD09_CA55
PLDVDD09_PLLCA55

3rd Grp-B power supply
1.1 V
DDR0_VDDQ

0.6/1.1 V
DDR0_VDDQLP

4th Grp power supply
1.8 V
VDD1833_PRE18_OTHERS_1
VDD1833_PRE18_OTHERS_2
DSI_VDD18
CSI0_MSVDD18
LVDS_VCCQ
USB20_USVDD18
USB21_USVDD18
USB30_USVPH
USB30_USVDD18
DDR0_VAA
PLVDD_PLLCA55
PLVDD_OTHERS
PCIE_VCC18ACMN
PCIE_VCC18AL01

(When used as 1.8 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

1.2/1.8 V
VDD1218_I3C

3.3 V
USB2_USVDD33

5th Grp power supply
1.2 V
DSI_VDD12
(When used as 3.3 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

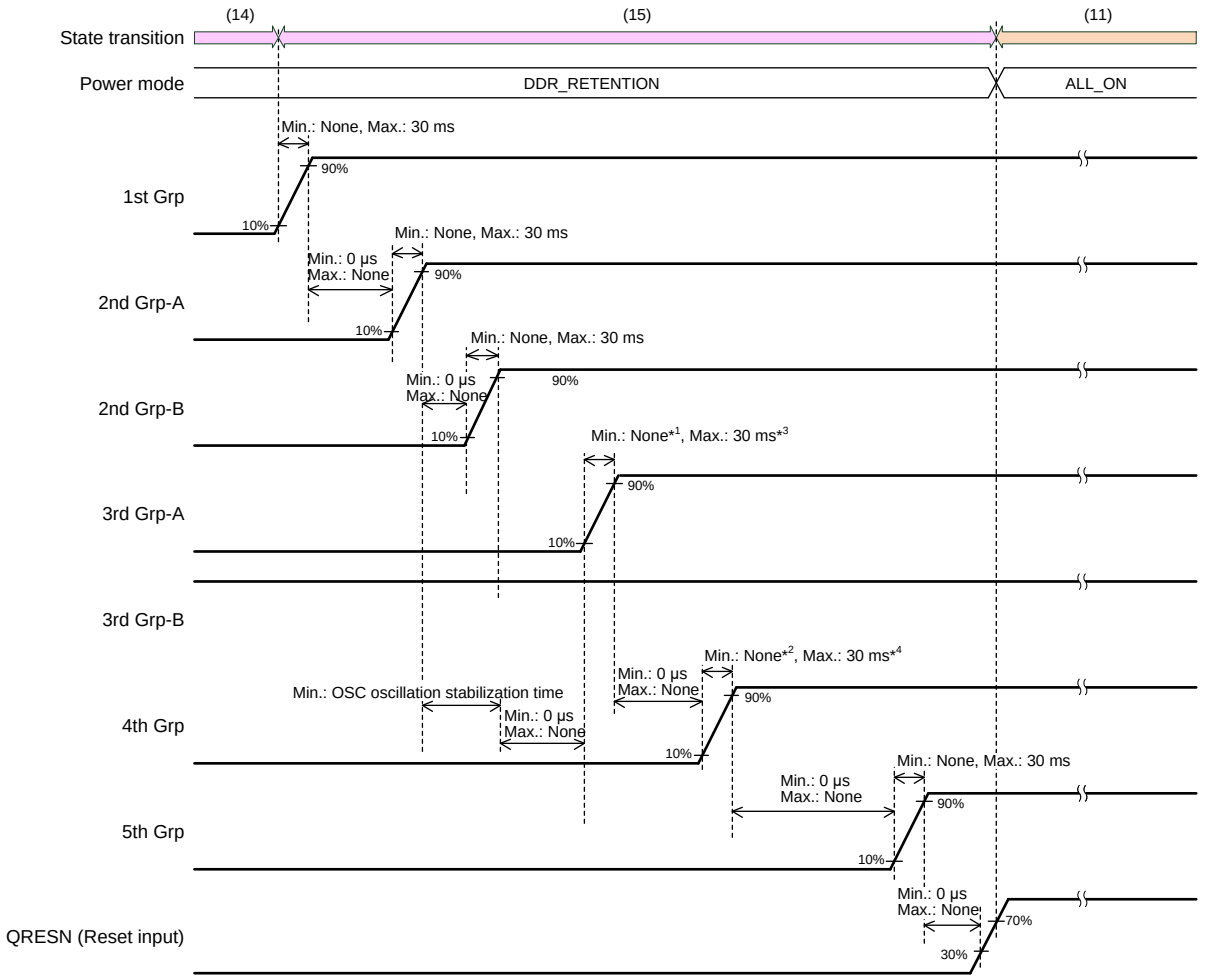
Note 1. VDD08_OTHERS: Min. 10 µs

Note 2. VDD1833_PRE18_OTHERS_1: Min. 10 µs

Note: The fall time of each power supply must be compliant with the maximum 30 ms regulation.

Figure 10.1-8 DDR Retention Enter Sequence (CA55 Boot Mode)

10.1.3.9 CA55 Boot Mode (DDR retention exit)



1st Grp power supply
0.8 V
VDD08_AWO
PLDVDD08_PLLCM33

2nd Grp-A power supply
1.8 V
ADAVDD18
OTPVDD18
VDD1833_PRE18_AWO
VDD18_AWO
VDD18_PWC
PLVDD_PLLCM33

(When used as 1.8 V)
VDD1833_AWO
VDD1833_XSPI

2nd Grp-B power supply
(When used as 3.3 V)
VDD1833_AWO
VDD1833_XSPI

3rd Grp-A power supply
0.8 V
USB2_USDVDD
USB30_USVPTX
PCIE_VCC08AL01
DSI_VDD0P8
CSI0_MSVDD0P8
PLDVDD08_OTHERS
VDD08_OTHERS

0.8/0.9 V
VDD09_CA55
PLDVDD09_PLLCA55

3rd Grp-B power supply
1.1 V
DDR0_VDDQ

0.6/1.1 V
DDR0_VDDQLP

4th Grp power supply
1.8 V
VDD1833_PRE18_OTHERS_1
VDD1833_PRE18_OTHERS_2
DSI_VDD18
CSI0_MSVDD18
LVDS_VCCQ
USB20_USVDD18
USB21_USVDD18
USB30_USVPH
USB30_USVDD18
DDR0_VAA
PLVDD_PLLCA55
PLVDD_OTHERS
PCIE_VCC18ACMN
PCIE_VCC18AL01

(When used as 1.8 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

1.2/1.8 V
VDD1218_I3C

3.3 V
USB2_USVDD33

5th Grp power supply
1.2 V
DSI_VDD12
(When used as 3.3 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

Note 1. VDD08_OTHERS, USB30_USVPTX, USB2_USDVDD: Min. 10 µs

Note 2. VDD1833_PRE18_OTHERS_1: Min. 10 µs
DDR0_VAA: Min. 290 µs
USB2_USVDD33: Min.30 µs
USB20_USVDD18, USB21_USVDD18, USB30_USVDD18: Min. 20 µs
USB30_USVPH: Min. 10 µs

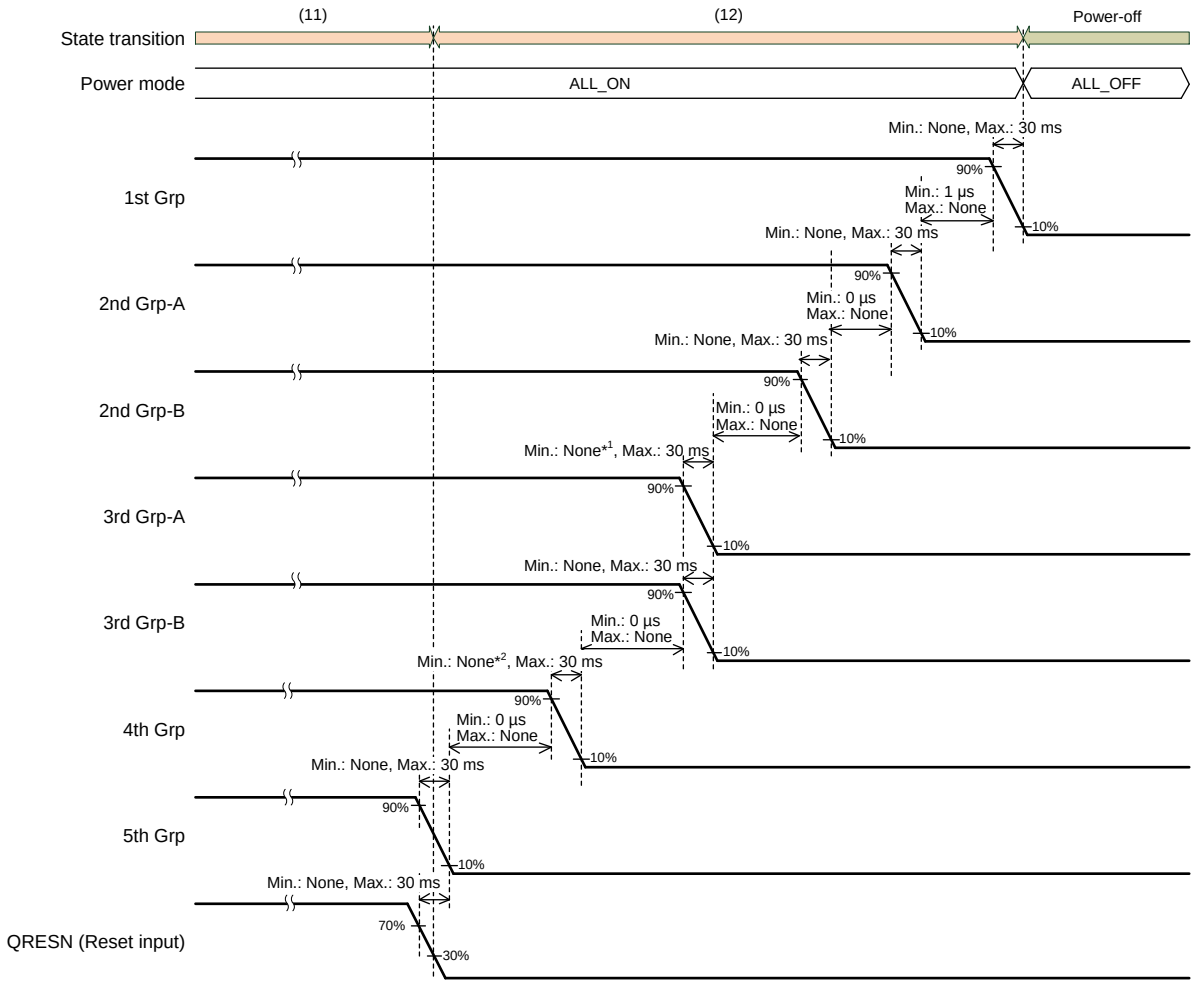
Note 3. USB2_USDVDD: Max. 10 ms

Note 4. USB20_USVDD18, USB21_USVDD18, USB30_USVDD18, USB2_USVDD33: Max. 10 ms

Note: The clock stabilization time depends on the board design. Make the setting according to the results of evaluation.

Figure 10.1-9 DDR Retention Exit Sequence (CA55 Boot Mode)

10.1.3.10 CA55 Boot Mode (Power-Off)



1st Grp power supply
0.8 V
VDD08_AWO
PLDVDD08_PLLCM33

2nd Grp-A power supply
1.8 V
ADAVDD18
OTPVDD18
VDD1833_PRE18_AWO
VDD18_AWO
VDD18_PWC
PLVDD_PLLCM33

(When used as 1.8 V)
VDD1833_AWO
VDD1833_XSPI

2nd Grp-B power supply
(When used as 3.3 V)
VDD1833_AWO
VDD1833_XSPI

3rd Grp-A power supply
0.8 V
USB2_USDVDD
USB30_USVPTX
PCIE_VCC08AL01
DSI_VDD0P8
CSI0_MSVDD0P8
PLDVDD08_OTHERS
VDD08_OTHERS

0.8/0.9 V
VDD09_CA55
PLDVDD09_PLLCA55

3rd Grp-B power supply
1.1 V
DDR0_VDDQ

0.6/1.1 V
DDR0_VDDQLP

4th Grp power supply
1.8 V
VDD1833_PRE18_OTHERS_1
VDD1833_PRE18_OTHERS_2
DSI_VDD18
CSI0_MSVDD18
LVDS_VCCQ
USB20_USVDD18
USB21_USVDD18
USB30_USVPH
USB30_USVDD18
DDR0_VAA
PLVDD_PLLCA55
PLVDD_OTHERS
PCIE_VCC18ACMN
PCIE_VCC18AL01

(When used as 1.8 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

1.2/1.8 V
VDD1218_I3C

3.3 V
USB2_USVDD33

5th Grp power supply
1.2 V
DSI_VDD12
(When used as 3.3 V)
VDD1833_OTHERS
VDD1833_SD0
VDD1833_SD1
VDD1833_SD2
VDD1833_ET0
VDD1833_ET1

Note 1. VDD08_OTHERS: Min. 10 µs

Note 2. VDD1833_PRE18_OTHERS_1: Min. 10 µs

Note: The clock stabilization time depends on the board design. Make the setting according to the results of evaluation.

Figure 10.1-10 Power-Off Sequence (CA55 Boot Mode)

10.1.4 DC Characteristics

10.1.4.1 Maximum Supply Current

Conditions for the supply current: Power supply voltage = Max. value, T_j = −40 to 125ºC

Table 10.1-3 Max. Supply Currents during Operation (1/2)

Unit Name	Item	Symbol	Max.	Unit	Note
CA55	0.8-V (or 0.9-V) power supply current	I_{DD09_CA55}	3014	mA	VDD09_CA55
PD_OTHERS	0.8-V core power supply current	I_{DD08_OTHERS}	3484	mA	VDD08_OTHERS
	I/O power supply current	I_{DD1833_OTHERS}	75	mA	VDD1833_OTHERS
	pre-driver power supply	$I_{DD1833PRE18_OTHERS1}$	4	mA	VDD1833_PRE18_OTHERS_1
PD_AWO	0.8-V core power supply current	I_{DD08_AWO}	192	mA	VDD08_AWO
	1.8-V core power supply current	I_{DD18_AWO}	4	mA	VDD18_AWO
	I/O power supply current	I_{DD1833_AWO}	13	mA	VDD1833_AWO
	Pre-driver power supply current	$I_{DD1833PRE18_AWO}$	2	mA	VDD1833_PRE18_AWO
USB3 Ch0	1.8-V PHY power supply current	$I_{DDUSB30_USVPH}$	33	mA	USB30_USVPH
	0.8-V PHY power supply current	$I_{DDUSB30_USVPTX}$	59	mA	USB30_USVPTX
	1.8-V PHY power supply current*1	$I_{DDUSB30_USVDD18}$	20	mA	USB30_USVDD18
USB2 Ch0	1.8-V PHY power supply current	$I_{DDUSB20_USVDD18}$	20	mA	USB20_USVDD18
USB2 Ch1	1.8-V PHY power supply current	$I_{DDUSB21_USVDD18}$	20	mA	USB21_USVDD18
USB3 Ch0 (for USB2), USB2 Ch0, USB2 Ch1 common	3.3-V PHY power supply current	$I_{DDUSB2_USVDD33}$	39	mA	USB2_USVDD33
	0.8-V PHY power supply current	I_{DDUSB2_USDVDD}	30	mA	USB2_USDVDD
xSPI	I/O power supply current	I_{DD1833_XSPI}	25	mA	VDD1833_XSPI
SD/MMC Ch0	I/O power supply current	I_{DD1833_SD0}	17	mA	VDD1833_SD0
SD/MMC Ch1	I/O power supply current	I_{DD1833_SD1}	10	mA	VDD1833_SD1
SD Ch2	I/O power supply current	I_{DD1833_SD2}	10	mA	VDD1833_SD2
SD/MMC Ch0, SD/MMC Ch1, SD Ch2, GBETH Ch0, GBETH Ch1 common	pre-driver power supply	$I_{DD1833PRE18_OTHERS2}$	7	mA	VDD1833_PRE18_OTHERS_2
OTP	1.8-V power supply current	$I_{DDOTPVDD18}$	6	mA	OTPVDD18
LPDDR4/LPDDR4X Ch0	1.1-V PHY power supply current	I_{DDQ_DDR0}	760	mA	DDR0_VDDQ
	PHY power supply current	I_{DDQLP_DDR0}	253	mA	DDR0_VDDQLP
	1.8-V PLL power supply current	I_{DDVAA_DDR0}	5	mA	DDR0_VAA
GBETH Ch0	I/O power supply current	I_{DD1833_ET0}	9	mA	VDD1833_ET0
GBETH Ch1	I/O power supply current	I_{DD1833_ET1}	9	mA	VDD1833_ET1
MIPI CSI Ch0	1.8-V PHY power supply current	$I_{DDMSVDD18_CSI0}$	11	mA	CSI0_MSVDD18
	0.8-V core power supply current	$I_{DDMSVDD0P8_CS0}$	39	mA	CSI0_MSVDD0P8
MIPI DSI	0.8-V core power supply current	I_{DD0P8_DSI}	112	mA	DSI_VDD0P8
	1.2-V PHY power supply current	I_{DD12_DSI}	8	mA	DSI_VDD12
	1.8-V PHY power supply current	I_{DD18_DSI}	11	mA	DSI_VDD18

Table 10.1-3 Max. Supply Currents during Operation (2/2)

Unit Name	Item	Symbol	Max.	Unit	Note
PCIE	1.8-V power supply current	$I_{DDPCIEVCC18ACMN}$	18	mA	PCIE_VCC18ACMN
	1.8-V PHY power supply current	$I_{DDPCIEVCC18AL01}$	53	mA	PCIE_VCC18AL01
	0.8-V PHY power supply current	$I_{DDPCIEVCC08AL01}$	83	mA	PCIE_VCC08AL01
I3C	I/O power supply current	I_{DD1218_I3C}	2	mA	VDD1218_I3C
ADC	1.8-V analog power supply current	$I_{DDADAVDD18}$	2	mA	ADAVDD18
LVDS Ch0, LVDS Ch1 common	LVDS 1.8V power supply current	$I_{DDLVDSVDD18}$	91	mA	LVDS_VCCQ
CPG	PLLCM33 1.8-V power supply current	$I_{DDPLVDD_PLLCM33}$	2	mA	PLVDD_PLLCM33
	PLLCA55 1.8-V power supply current	$I_{DDPLVDD_PLLCA55}$	2	mA	PLVDD_PLLCA55
	PLLVDD_OTHERS 1.8-V power supply current	$I_{DDPLVDD_PLLOTHERS}$	15	mA	PLVDD_OTHERS
	PLLCM33 0.8-V power supply current	$I_{DDPLVDD08_PLLCM33}$	3	mA	PLDVDD08_PLLCM33
	PLLCA55 0.8-V power supply current	$I_{DDPLVDD08_PLLCA55}$	3	mA	PLDVDD09_PLLCA55
	PLLVDD_OTHERS 0.8-V power supply current	$I_{DDPLVDD08_PLLOTHERS}$	18	mA	PLDVDD08_OTHERS
PWC	1.8-V I/O power supply current	I_{DD18_PWC}	2	mA	VDD18_PWC

10.1.4.2 Standard I/O Characteristics

For the I/O types, refer to the external pin list in **1.2 Pin**.

Table 10.1-4 DC Characteristics (Standard I/O)

V_{DD} = 1.14 V to 1.89 V (1.8/1.2-V switching I/O type), V_{DD} = 1.71 V to 1.89 V (3.3/1.8-V switching I/O types 3*[13], 4*[13] and 1.8-V OSC I/O type), V_{DD} = 1.71 V to 3.46 V (3.3/1.8-V switching I/O types 1, 2)

Item	I/O Type	Symbol	Min.	Typ.	Max.	Unit	Condition
External voltage tolerance	3.3/1.8-V switching I/O type 3	V_{TOL}	—	—	3.6	V	V_{DD} power-off & on
High-level input voltage	1.8/1.2-V switching I/O type (1.2 V)	V_{IH}	0.8 × V_{DD}	—	V_{DD} + 0.3	V	—
	1.8/1.2-V switching I/O type (1.8 V)	V_{IH}	0.7 × V_{DD}	—	V_{DD} + 0.3	V	—
	1.8-V OSC I/O type 3.3/1.8-V switching I/O type 2 (1.8 V)	V_{IH}	0.7 × V_{DD}	—	V_{DD} + 0.3	V	—
	3.3/1.8-V switching I/O type 1 3.3/1.8-V switching I/O type 3 3.3/1.8-V switching I/O type 4 (1.8 V)	V_{IH}	0.75 × V_{DD}	—	V_{DD} + 0.3	V	—
	3.3/1.8-V switching I/O type 1 3.3/1.8-V switching I/O type 2 (3.3 V)	V_{IH}	0.7 × V_{DD}	—	V_{DD} + 0.3	V	—
Low-level input voltage	1.8/1.2-V switching I/O type (1.2 V)	V_{IL}	−0.3	—	0.2 × V_{DD}	V	—
	1.8/1.2-V switching I/O type (1.8 V)	V_{IL}	−0.3	—	0.3 × V_{DD}	V	—
	1.8-V OSC I/O type 3.3/1.8-V switching I/O type 1 3.3/1.8-V switching I/O type 2 3.3/1.8-V switching I/O type 3 3.3/1.8-V switching I/O type 4 (1.8 V)	V_{IL}	−0.3	—	0.3 × V_{DD}	V	—
	3.3/1.8-V switching I/O type 1 3.3/1.8-V switching I/O type 2 (3.3 V)	V_{IL}	−0.3	—	0.3 × V_{DD}	V	—
Hysteresis voltage	1.8/1.2-V switching I/O type	ΔV	0.1 × V_{DD}	—	—	V	—
	3.3/1.8-V switching I/O type 3*[2] 3.3/1.8-V switching I/O type 4*[1]	ΔV	0.08 × V_{DD}	—	—	V	—
	3.3/1.8-V switching I/O type 2*[14]	ΔV	0.1	—	—	V	—

Item	I/O Type	Symbol	Min.	Typ.	Max.	Unit	Condition
Input leakage current	1.8/1.2-V switching I/O type (1.2 V)	I_I	−10	—	10	µA	V_{in} = V_{SS} or V_{DD} max & V_{DD} power-on
			−10	—	10	µA	V_{in} = V_{SS} or V_{DD} max & V_{DD} power-off
	1.8/1.2-V switching I/O type (1.8 V)	I_I	−18	—	18	µA	V_{in} = V_{DD} max & V_{DD} power-on
			−18	—	18	µA	V_{in} = V_{SS} or V_{DD} max & V_{DD} power-off
	1.8-V OSC I/O type 3.3/1.8-V switching I/O type 1	I_I	−12	—	12	µA	V_{in} = V_{DD} max & V_{DD} power-on
	3.3/1.8-V switching I/O type 3	I_I	−18	—	18	µA	V_{in} = V_{DD} max & V_{DD} power-on
			−18	—	18	µA	V_{in} = V_{SS} or V_{DD} max & V_{DD} power-off
	3.3/1.8-V switching I/O type 2 3.3/1.8-V switching I/O type 4	I_I	−12	—	12	µA	V_{in} = V_{DD} max & V_{DD} power-on
Input pull-down resistor current	1.8/1.2-V switching I/O type*3 (1.2 V)	I_{RPU}	10	—	100	µA	V_{in} = V_{DD} max
	1.8/1.2-V switching I/O type*3 (1.8 V)	I_{RPU}	25	—	130	µA	V_{in} = V_{DD} max
	3.3/1.8-V switching I/O type 1*7 3.3/1.8-V switching I/O type 3*9 3.3/1.8-V switching I/O type 4*5	I_{RPU}	25	—	200	µA	V_{in} = V_{DD} max
	3.3/1.8-V switching I/O type 2*11	I_{RPU}	18	—	148	µA	V_{in} = V_{DD} max
Input pull-up resistor current	1.8/1.2-V switching I/O type*4 (1.2 V)	I_{RPD}	−10	—	−100	µA	V_{in} = V_{SS}
	1.8/1.2-V switching I/O type*4 (1.8 V)	I_{RPD}	−35	—	−185	µA	V_{in} = V_{SS}
	3.3/1.8-V switching I/O type 1*8 3.3/1.8-V switching I/O type 3*10 3.3/1.8-V switching I/O type 4*6	I_{RPD}	−25	—	−200	µA	V_{in} = V_{SS}
	3.3/1.8-V switching I/O type 2*12	I_{RPD}	−18	—	−192	µA	V_{in} = V_{SS}

Item	I/O Type	Symbol	Min.	Typ.	Max.	Unit	Condition
High-level output voltage	1.8/1.2-V switching I/O type (1.2 V)	V_{OH}	0.8 × V_{DD}	—	V_{DD}	V	I_{OH} = −1/−2/−4/−6 mA (drive strength X1/X2/X4/X6)
	1.8/1.2-V switching I/O type (1.8 V)	V_{OH}	0.8 × V_{DD}	—	V_{DD}	V	I_{OH} = −2/−4/−8/−12 mA (drive strength X1/X2/X4/X6)
	1.8-V OSC I/O type	V_{OH}	0.8 × V_{DD}	—	V_{DD}	V	I_{OH} = −2/−4/−8/−12 mA (drive strength X1/X2/X4/X6)
	3.3/1.8-V switching I/O type 1 (1.8 V) 3.3/1.8-V switching I/O type 3 (1.8V) 3.3/1.8-V switching I/O type 4 (1.8V)	V_{OH}	0.8 × V_{DD}	—	V_{DD}	V	I_{OH} = −1.6/−3.2/−6.4/−9.6 mA (drive strength X1/X2/X4/X6)
	3.3/1.8-V switching I/O type 1 (3.3 V)	V_{OH}	0.8 × V_{DD}	—	V_{DD}	V	I_{OH} = −2/−4/−8/−12 mA (drive strength X1/X2/X4/X6)
	3.3/1.8-V switching I/O type 2 (1.8 V)	V_{OH}	0.8 × V_{DD}	—	V_{DD}	V	I_{OH} = −5/−6/−7/−10 mA (drive strength X1/X2/X4/X6)
	3.3/1.8-V switching I/O type 2 (3.3 V)	V_{OH}	0.8 × V_{DD}	—	V_{DD}	V	I_{OH} = −9/−11/−13/−18 mA (drive strength X1/X2/X4/X6)
Low-level output voltage	1.8/1.2-V switching I/O type (1.2 V)	V_{OL}	0	—	0.2 × V_{DD}	V	I_{OL} = 1/2/4/6 mA (drive strength X1/X2/X4/X6)
	1.8/1.2-V switching I/O type (1.8 V)	V_{OL}	0	—	0.2 × V_{DD}	V	I_{OL} = 2/4/8/12 mA (drive strength X1/X2/X4/X6)
	1.8-V OSC I/O type	V_{OL}	0	—	0.2 × V_{DD}	V	I_{OL} = 2/4/8/12 mA (drive strength X1/X2/X4/X6)
	3.3/1.8-V switching I/O type 1 (1.8 V) 3.3/1.8-V switching I/O type 3 (1.8V) 3.3/1.8-V switching I/O type 4 (1.8V)	V_{OL}	0	—	0.2 × V_{DD}	V	I_{OL} = 1.6/3.2/6.4/9.6 mA (drive strength X1/X2/X4/X6)
	3.3/1.8-V switching I/O type 1 (3.3 V)	V_{OL}	0	—	0.2 × V_{DD}	V	I_{OL} = 2/4/8/12 mA (drive strength X1/X2/X4/X6)
	3.3/1.8-V switching I/O type 2 (1.8 V)	V_{OL}	0	—	0.2 × V_{DD}	V	I_{OL} = 5/6/7/10 mA (drive strength X1/X2/X4/X6)
	3.3/1.8-V switching I/O type 2 (3.3 V)	V_{OL}	0	—	0.2 × V_{DD}	V	I_{OL} = 9/11/13/18 mA (drive strength X1/X2/X4/X6)

Item	I/O Type	Symbol	Min.	Typ.	Max.	Unit	Condition
Pull-up resistance	1.8/1.2-V switching I/O type*4 (1.2 V)	R_{PU}	15	—	160	kΩ	—
	1.8/1.2-V switching I/O type*4 (1.8 V)	R_{PU}	10	—	50	kΩ	—
	3.3/1.8-V switching I/O type 1*8 (1.8 V) 3.3/1.8-V switching I/O type 3*10 (1.8 V) 3.3/1.8-V switching I/O type 4*6 (1.8 V)	R_{PU}	10	—	50	kΩ	—
	3.3/1.8-V switching I/O type 1*8 (3.3 V)	R_{PU}	10	—	100	kΩ	—
	3.3/1.8-V switching I/O type 2*12	R_{PU}	12	—	92	kΩ	—
Pull-down resistance	1.8/1.2-V switching I/O type*3 (1.2 V)	R_{PD}	15	—	160	kΩ	—
	1.8/1.2-V switching I/O type*3 (1.8 V)	R_{PD}	15	—	60	kΩ	—
	3.3/1.8-V switching I/O type 1*7 (1.8 V) 3.3/1.8-V switching I/O type 3*9 (1.8 V) 3.3/1.8-V switching I/O type 4*5 (1.8 V)	R_{PD}	10	—	50	kΩ	—
	3.3/1.8-V switching I/O type 1*7 (3.3 V)	R_{PD}	10	—	100	kΩ	—
	3.3/1.8-V switching I/O type 2*11	R_{PD}	13	—	92	kΩ	—
Input capacitance	—	C_{in}	—	—	10	pF	

Note 1. Only for the QRESN and QRESNSEL pin

Note 2. When the RIIC function is in use or the value of the PFC_SMT_mn register is 1

Note 3. Only for the P20 and P21 pins (when the value of the PFC_PUPD_mn register is 10b)

Note 4. Only for the P20 and P21 pins (when the value of the PFC_PUPD_mn register is 11b)

Note 5. Only for the QBYPASS, BSCANP, MD_BOOT0, MD_BOOT3, and BOOTSELCPU pins

Note 6. Only for the MD_BOOT1, MD_BOOT2 and MD_CLKS pins

Note 7. Only for the WDTUDFCA, WDTUDFCM, and SCIF_RXD, SCIF_TXD pins (when the value of the PFC_PUPD_mn register is 10b)

Note 8. Only for the WDTUDFCA, WDTUDFCM, and SCIF_RXD, SCIF_TXD pins (when the value of the PFC_PUPD_mn register is 11b)

Note 9. When the value of the PFC_PUPD_mn register is 10b

Note 10. When the value of the PFC_PUPD_mn register is 11b

Note 11. Only for the pins other than SD0CLK, SD0RSTN, SD0PWEN, and SD0IOVS (when the value of the PFC_PUPD_mn register is 10b)

Note 12. Only for the pins other than SD0CLK, SD0RSTN, SD0PWEN, and SD0IOVS (when the value of the PFC_PUPD_mn register is 11b)

Note 13. This type IO can be available both 3.3 V and 1.8 V buffers, but it is fixed at 1.8 V buffer.

Note 14. Only for the pins other than SD0CLK, SD0CMD, SD0DAT0-SD0DAT7, SD0PWEN, SD0IOVS, and SD0RSTN (when the RIIC function is in use or the value of the PFC_SMT_mn register is 1)

Table 10.1-5 DC Characteristics (LVDS)

V_{DD} = 1.71 V to 1.89 V (I/O type[LVDS])

Item	I/O Type	Symbol	Min.	Typ.	Max.	Unit	Condition
Output differential voltage	LVDS	|VOD|	250	—	450	mV	TIA-EIA-644 Termination load = 100Ω
Output offset voltage	LVDS	VOS	1.125	—	1.375	V	TIA-EIA-644 Termination load = 100Ω
Change in |VOD| between 0 and 1	LVDS	ΔVOD	—	—	50	mV	TIA-EIA-644 Termination load = 100Ω
Change in VOS between 0 and 1	LVDS	ΔVOS	—	—	50	mV	TIA-EIA-644 Termination load = 100Ω

10.1.5 AC Characteristics

Conditions:

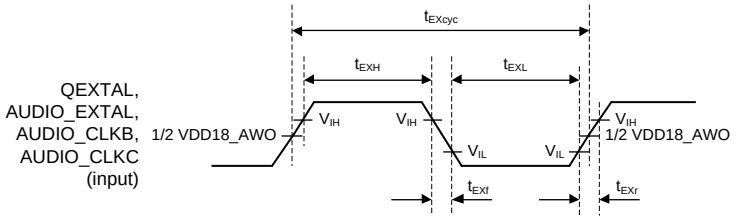
VDD18 = VDD18_AWO = ADAVDD18 = VDD1833_* (1.8 V mode)

VDD33 = VDD1833_* (3.3 V mode)

10.1.5.1 Clock Timing

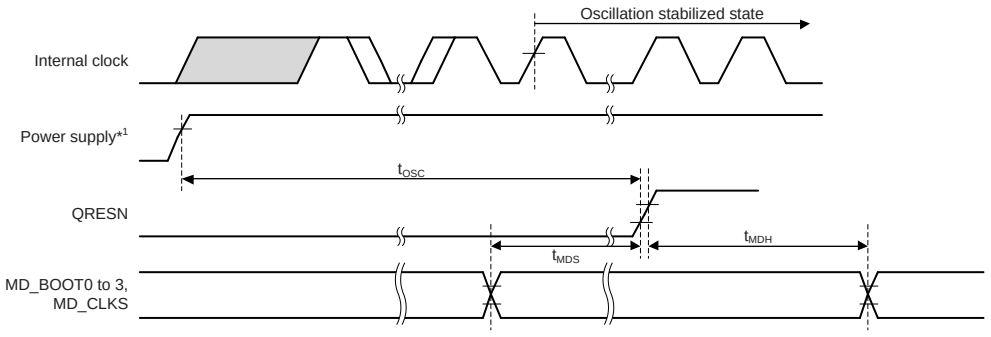
Table 10.1-6 Clock Timing Table

Item	Symbol	Min.	Max.	Unit	Figures
QEXTAL clock input frequency	f_{EX}	24 −50 ppm	24 +50 ppm	MHz	**Figure 10.1-11**
QEXTAL clock input cycle time	t_{EXcyc}	41.67	41.67	ns	
AUDIO_EXTAL clock input frequency	f_{EX}	4	48	MHz	
AUDIO_EXTAL clock input cycle time	t_{EXcyc}	20.83	250	ns	
AUDIO_CLKB, AUDIO_CLKC clock input frequency (external clock is input)	f_{EX}	4	50	MHz	
AUDIO_CLKB, AUDIO_CLKC clock input cycle time (external clock is input)	t_{EXcyc}	20	250	ns	
QEXTAL clock input low-level pulse width	t_{EXL}	0.4	0.6	t_{EXcyc}	
QEXTAL clock input high-level pulse width	t_{EXH}	0.4	0.6	t_{EXcyc}	
AUDIO_XTAL, AUDIO_CLKB, AUDIO_CLKC clock input low-level pulse width	t_{EXL}	0.45	0.55	t_{EXcyc}	
AUDIO_EXTAL, AUDIO_CLKB, AUDIO_CLKC clock input high-level pulse width	t_{EXH}	0.45	0.55	t_{EXcyc}	
QEXTAL, AUDIO_EXTAL, AUDIO_CLKB, AUDIO_CLKC clock input rise time	t_{EXr}	—	4	ns	
QEXTAL, AUDIO_EXTAL, AUDIO_CLKB, AUDIO_CLKC clock input fall time	t_{EXf}	—	4	ns	
Mode hold time	t_{MDH}	—	100	ns	**Figure 10.1-12**
Mode setup time	t_{MDS}	—	100	ns	



Note: When the clock is input on the QEXTAL, AUDIO_EXTAL, AUDIO_CLKB, or AUDIO_CLKC

Figure 10.1-11 Clock Input Timing



Note 1. 1st Grp power supply (refer to **10.1.3 Power-On/Off Sequence**)

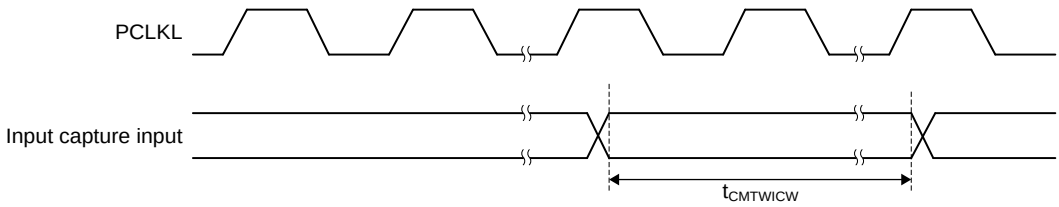
Figure 10.1-12 Power-On Oscillation Settling Time

10.1.5.2 CMTW Timing

Table 10.1-7 CMTW Timing

Parameter			Symbol	Min.	Max.	Unit	Figure
CMTW	Input capture input pulse width	Single-edge setting	$t_{CMTWICW}$	1.5	—	t_{PLcyc}*1	**Figure 10.1-13**
		Both-edge setting		2.5	—		

Note 1. t_{PLcyc}: PCLKL cycle



Note: PCLKL: CMTWn_clkm (n = 0 to 7)

Figure 10.1-13 CMTW Input Capture Input Timing

10.1.5.3 POEG and GPT Trigger Timings

GPT Conditions: High-drive output is selected in the Port Drive Capability bit in the PFC_IOLH_mn register.

Table 10.1-8 POEG and GPT Trigger Timings

Parameter			Symbol	Min.	Max.	Unit	Figure
POEG	POEG input trigger pulse width		t_{POEW}	1.5	—	t_{Pcyc}*1	**Figure 10.1-14**
GPT	Input capture pulse width	Single edge	t_{GTICW}	1.5	—	t_{PDcyc}*2	**Figure 10.1-15**
		Dual edge	t_{GTICW}	2.5	—		

Note 1. t_{Pcyc}: POEGx_n_PCLK cycle (x = A to D, n = 0, 1)

Note 2. t_{PDcyc}: GPT_n_clks_gpt cycle (n = 0, 1)

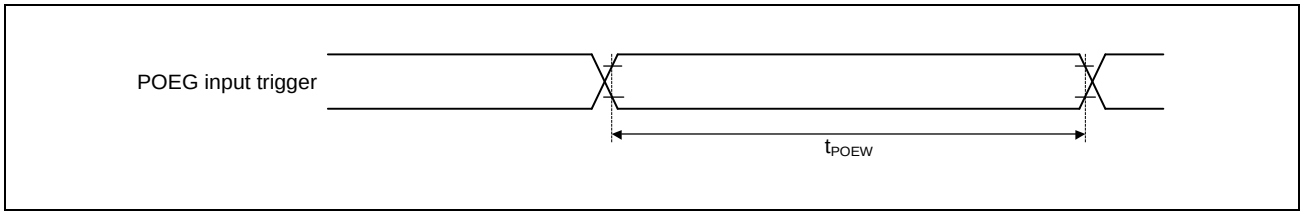


Figure 10.1-14 POEG Input Trigger Timing

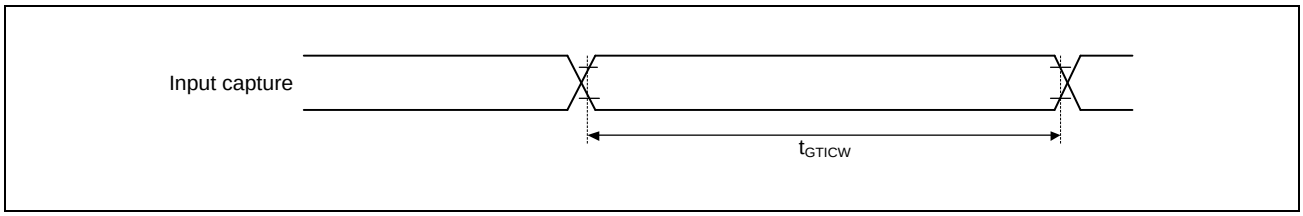


Figure 10.1-15 GPT Input Capture Timing

10.1.5.4 Watchdog Timer Access Timing

Table 10.1-9 Watchdog Timer Timing

Item	Symbol	Min.	Max.	Unit	Figures
WDTUDFCM / WDTUDFCA output time	t_L	64	64	t_{P1cyc}*1	**Figure 10.1-16**

Note 1. t_{P1cyc} indicates peripheral clock WDT_n_clk_loco (n = 0 to 3).

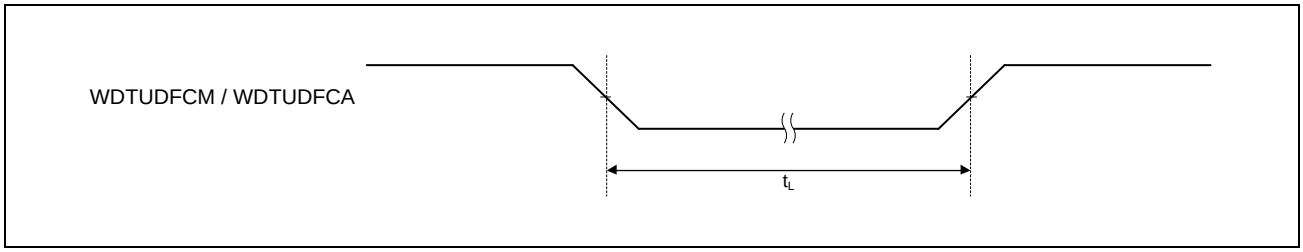


Figure 10.1-16 Watchdog Timer Output Timing

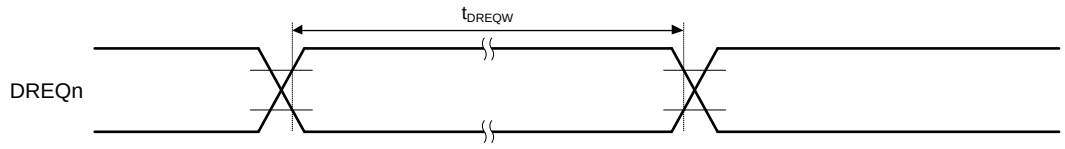
10.1.5.5 DMAC Timing

Table 10.1-10 DMAC Timing

Item	Symbol	Min.	Max.	Unit	Figures
DREQn pulse width	t_{DREQW}	20	—	t_{cyc}*1	**Figure 10.1-17**
TENDn pulse width	t_{TENDW}	16	16	$t_{PCLKcyc}$*2	**Figure 10.1-18**

Note 1. t_{cyc} = 41.666 ns (24 MHz)

Note 2. $t_{PCLKcyc}$ = 10 ns (100 MHz): ICU_0_PCLK_I cycle



Note: This specification “(Min. 20 t_{cyc})” is the min. pulse width when the digital noise filter is off. Refer to **4.2 Pin Function Controller (PFC)** for the noise filter settings in detail.

Figure 10.1-17 DMAC DREQn Timing

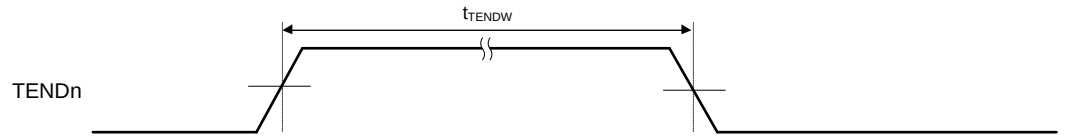


Figure 10.1-18 DMAC TENDn Timing

10.1.5.6 LPDDR4 PHY Characteristics

The LPDDR4 PHY of this LSI is compliant with the JEDEC 209-4B / JEDEC 209-4-1 standard.

10.1.5.7 SD Access Timing

Conditions:

V_{OH} = VDD33 × 0.7

V_{OL} = VDD33 × 0.3

C = 40 pF (3.3 V)

Drive strength: ×6

10.1.5.7.1 SD Access Timing (SDR 3.3-V)

Table 10.1-11 SD AC Access Timing (SDR at 3.3-V Operation)

Item	Symbol	Default Speed Mode (25 MHz)		High Speed Mode (50 MHz)		Unit	Figures
		Min.	Max.	Min.	Max.		
SDnCLK clock cycle	t_{SDCYC}	40.0	—	20.0	—	ns	**Figure 10.1-19**
SDnCLK clock high level width	t_{SDWH}	10	—	7	—	ns	
SDnCLK clock low level width	t_{SDWL}	10	—	7	—	ns	
SDnCLK clock rise time	t_{SDLH}	—	10	—	3	ns	
SDnCLK clock fall time	t_{SDHL}	—	10	—	3	ns	
SDnCMD,SDnDATm output delay	t_{SDODLY}	−7.5	2.5	−6.2	2.5	ns	
SDnCMD,SDnDATm input set up time	t_{SDIS}	4.0	—	4.0	—	ns	
SDnCMD,SDnDATm input hold time	t_{SDIH}	2.0	—	2.0	—	ns	
SDnCMD,SDnDATm input data width	t_{SDIDW}	—	—	—	—	ns	

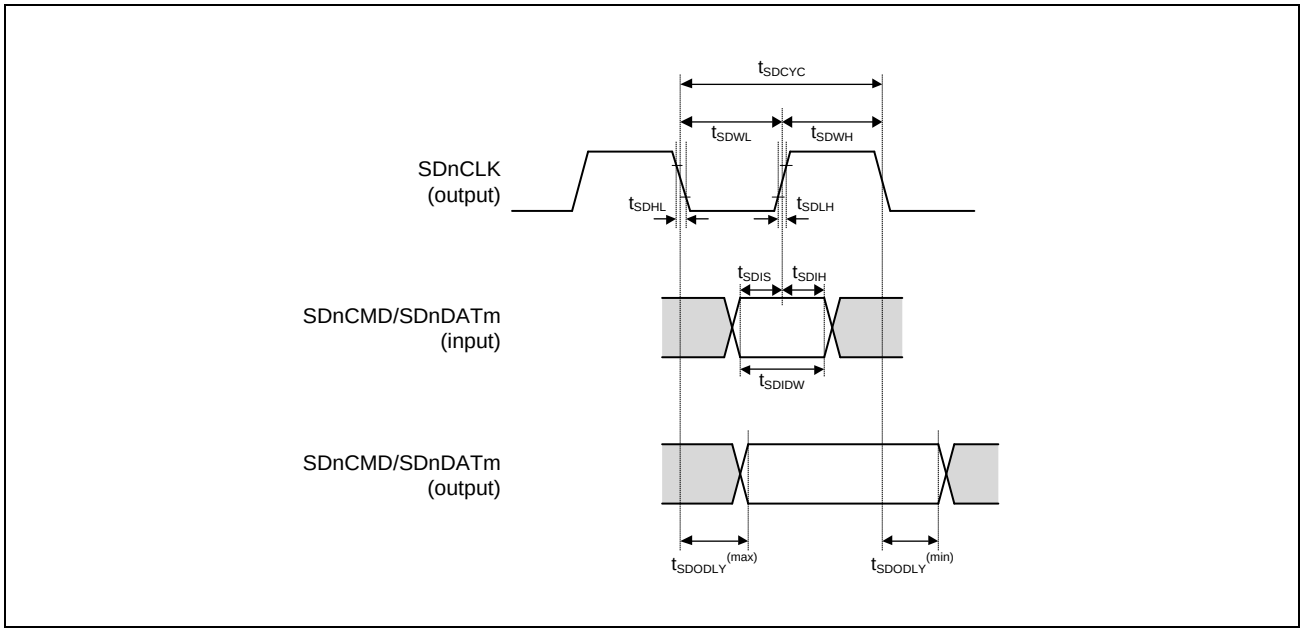


Figure 10.1-19 SDHC Interface Timing (SDR 3.3-V Power Supply)

NOTE

The disclosure of other characteristics of the SD interface needs the conclusion of the following agreement.

- SD Host/Ancillary Product License Agreement (SD HALA)

For details, contact Renesas sales representatives.

10.1.5.8 eMMC Access Timing

Conditions:

V_{OH} = VDD18 × 0.7, V_{OL} = VDD18 × 0.3, C = 15 pF (1.8 V)

V_{OH} = VDD33 × 0.7, V_{OL} = VDD33 × 0.3, C = 30 pF (3.3 V)

Drive strength: ×6

10.1.5.8.1 eMMC host interface timing (default)

Table 10.1-12 eMMC Host Interface Timing (MMC Default 3.3-V Power Supply)

Item	Symbol	Min.	Max.	Unit	Figures
SDnCLK clock cycle	t_{MMCPP}	20.0	—	ns	**Figure 10.1-20**
SDnCLK clock high level width	t_{MMCWH}	7	—	ns	
SDnCLK clock low level width	t_{MMCWL}	7	—	ns	
SDnCLK clock rise time	t_{MMCLH}	—	3	ns	
SDnCLK clock fall time	t_{MMCHL}	—	3	ns	
SDnCMD/SDnDATm output delay	$t_{MMCODLY}$	−6.2	2.5	ns	
SDnCMD/SDnDATm input setup time	t_{MMCISU}	4.0	—	ns	
SDnCMD/SDnDATm input hold time	t_{MMCIH}	2.0	—	ns	
SDnCMD/SDnDATm input data width	t_{MMCIDW}	—	—	ns	

Table 10.1-13 eMMC Host Interface Timing (MMC Default 1.8-V Power Supply)

Item	Symbol	Min.	Max.	Unit	Figures
SDnCLK clock cycle	t_{MMCPP}	20.0	—	ns	**Figure 10.1-20**
SDnCLK clock high level width	t_{MMCWH}	7	—	ns	
SDnCLK clock low level width	t_{MMCWL}	7	—	ns	
SDnCLK clock rise time	t_{MMCLH}	—	3	ns	
SDnCLK clock fall time	t_{MMCHL}	—	3	ns	
SDnCMD/SDnDATm output delay	$t_{MMCODLY}$	−4.2	1.6	ns	
SDnCMD/SDnDATm input setup time	t_{MMCISU}	1.3	—	ns	
SDnCMD/SDnDATm input hold time	t_{MMCIH}	1.878	—	ns	
SDnCMD/SDnDATm input data width	t_{MMCIDW}	—	—	ns	

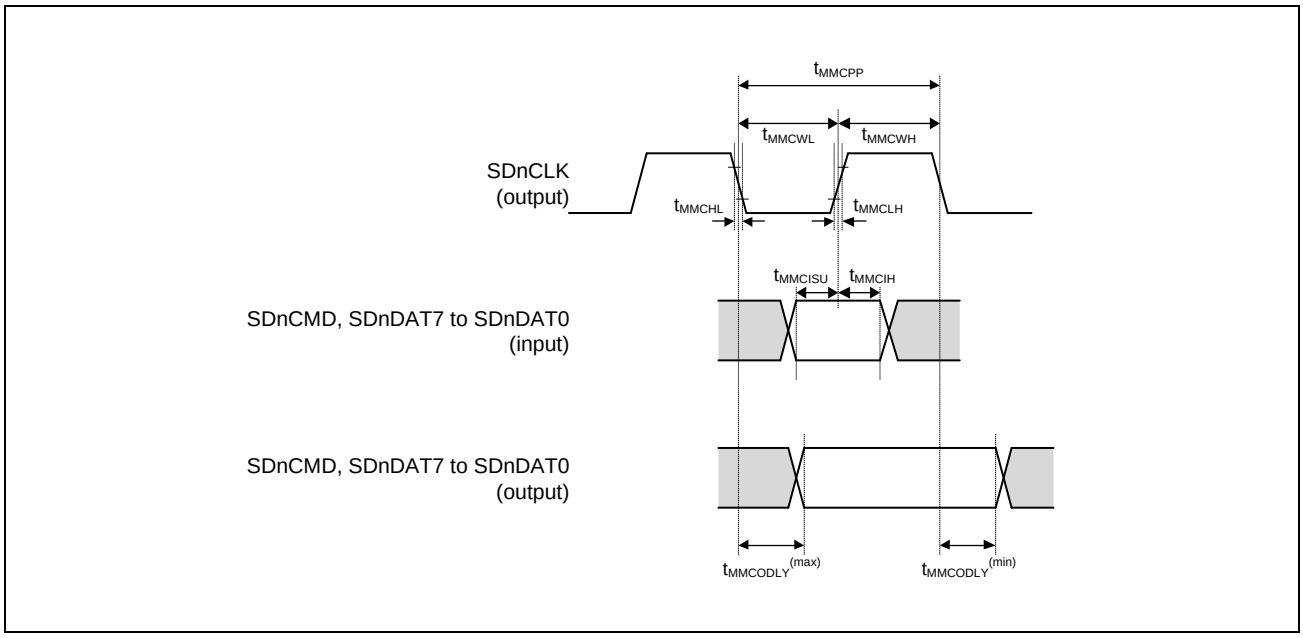


Figure 10.1-20 eMMC Host Interface Timing (MMC Default 1.8-V/3.3-V Power Supply)

10.1.5.8.2 eMMC host interface timing (HS-SDR)

NOTES

1. The spec of eMMC host interface timing (HS-SDR 3.3-V power supply) is the same as **Table 10.1-12 eMMC Host Interface Timing (MMC Default 3.3-V Power Supply)**.
2. The spec of eMMC host interface timing (HS-SDR 1.8V power supply) is the same as **Table 10.1-13 eMMC Host Interface Timing (MMC Default 1.8-V Power Supply)**.

10.1.5.8.3 eMMC host interface timing (HS-DDR)

Table 10.1-14 eMMC Host Interface Timing (HS-DDR 3.3-V Power Supply Operation)

Item	Symbol	High Speed Mode (50 MHz)		Unit	Figures
		Min.	Max.		
SDnCLK clock cycle	t_{SDCYC}	20.0	—	ns	**Figure 10.1-21**
SDnCLK clock high level width	t_{SDWH}	9.0	11.0	ns	
SDnCLK clock low level width	t_{SDWL}	9.0	11.0	ns	
SDnCLK clock rise time	t_{SDLH}	—	3.0	ns	
SDnCLK clock fall time	t_{SDHL}	—	3.0	ns	
SDnCMD output delay	t_{SDODLY}	−6.0	6.0	ns	
SDnCMD input set up time	t_{SDIS}	4.8	—	ns	
SDnCMD input hold time	t_{SDIH}	2.5	—	ns	
SDnDATm output delay	t_{SDODLY_DDR}	2.5	6.5	ns	
SDnDATm input set up time	t_{SDIS_DDR}	1.768	—	ns	
SDnDATm input hold time	t_{SDIH_DDR}	1.5	—	ns	

Table 10.1-15 eMMC Host Interface Timing (HS-DDR 1.8-V Power Supply Operation)

Item	Symbol	High Speed Mode (50 MHz)		Unit	Figures
		Min.	Max.		
SDnCLK clock cycle	t_{MMCCYC}	20.0	—	ns	**Figure 10.1-21**
SDnCLK clock high level width	t_{MMCWH}	9.0	11.0	ns	
SDnCLK clock low level width	t_{MMCWL}	9.0	11.0	ns	
SDnCLK clock rise time	t_{MMCLH}	—	3.0	ns	
SDnCLK clock fall time	t_{MMCHL}	—	3.0	ns	
SDnCMD output delay	$t_{MMCODLY}$	−6.0	3.0	ns	
SDnCMD input set up time	t_{MMCIS}	4.8	—	ns	
SDnCMD input hold time	t_{MMCIH}	2.5	—	ns	
SDnDATm output delay	$t_{MMCODLY_DDR}$	2.5	6.5	ns	
SDnDATm input set up time	t_{MMCIS_DDR}	1.768	—	ns	
SDnDATm input hold time	t_{SMMCIH_DDR}	1.5	—	ns	

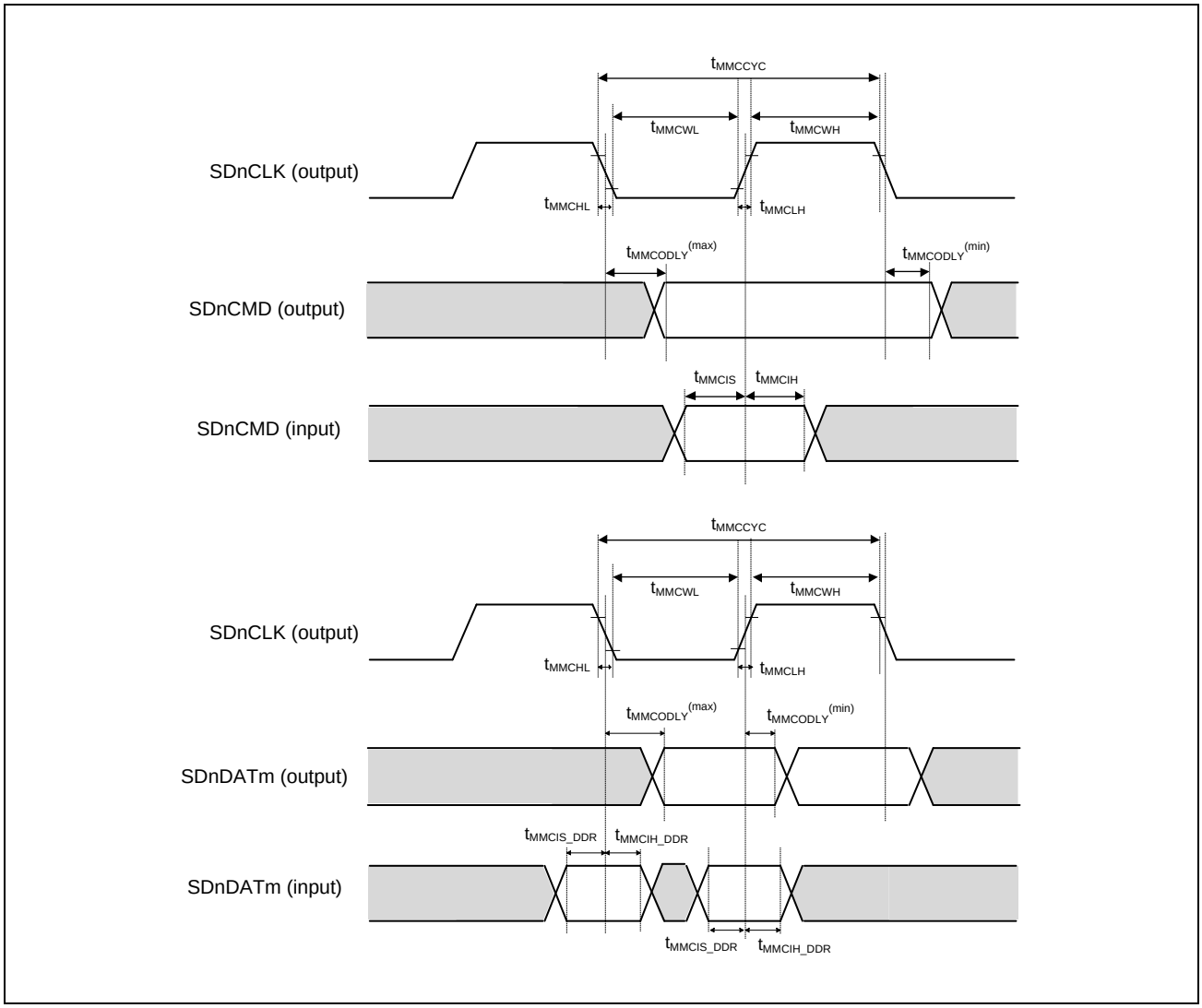


Figure 10.1-21 eMMC Host Interface (MMC Interface HS-DDR Mode 1.8/3.3-V Power Supply Selection)

10.1.5.8.4 eMMC host interface timing (HS200)

Table 10.1-16 eMMC Host Interface Timing (HS200 1.8-V Power Supply Operation, Output Load 15 pF)

Item	Symbol	Min.	Max.	Unit	Figures
SDnCLK clock cycle	t_{MMCPP}	5.0	10.0	ns	**Figure 10.1-22**
SDnCLK clock high level width	t_{MMCWH}	1.5	—	ns	
SDnCLK clock low level width	t_{MMCWL}	1.5	—	ns	
SDnCLK clock rise time	t_{MMCLH}	—	1.0	ns	
SDnCLK clock fall time	t_{MMCHL}	—	1.0	ns	
SDnCMD/SDnDATm output delay	$t_{MMCODLY}$	−1.7	0.9	ns	
SDnCMD/SDnDATm input setup time	t_{MMCISU}	—	—	ns	
SDnCMD/SDnDATm input hold time	t_{MMCIH}	—	—	ns	
SDnCMD/SDnDATm input data width	t_{MMCIDW}	2.88	—	ns	

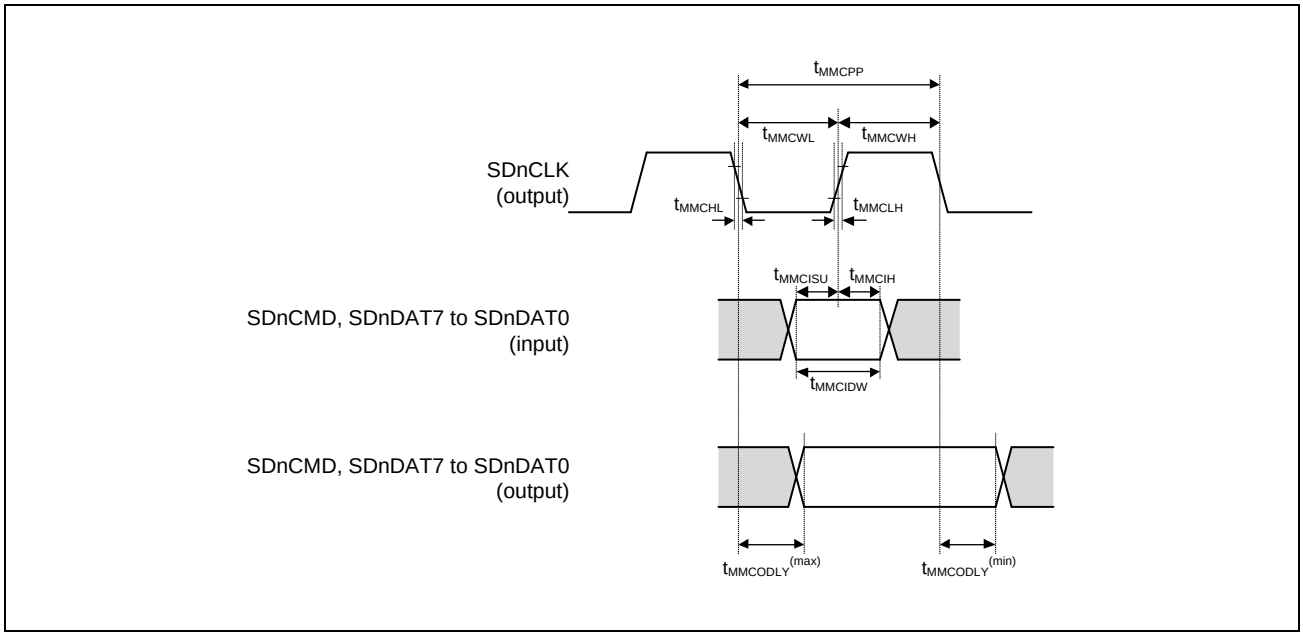


Figure 10.1-22 eMMC Host Interface (MMC Interface HS200 Mode 1.8-V Power Supply Selection)

10.1.5.9 Ethernet Interface Timing

Conditions:

V_{OH} = VDD18 × 0.5, V_{OL} = VDD18 × 0.5

V_{OH} = VDD33 × 0.5, V_{OL} = VDD33 × 0.5

C = 15 pF (RGMII), 30 pF (MII)

Drive strength: ×2, ×4

Table 10.1-17 Ethernet Interface Timing (n = 0, 1)

Parameter			Symbol	Min.	Max.	Unit	Figure
Ethernet (RGMII)	ETn_TXC_TXCLK, ETn_RXC_RXCLK cycle time duration	1 Gbps	$t_{RGMIIck}$	7.2	8.8	ns	**Figure 10.1-23**
		100 Mbps	$t_{RGMIIck}$	36	44	ns	
		10 Mbps	$t_{RGMIIck}$	360	440	ns	
	ETn_TXC_TXCLK, ETn_RXC_RXCLK frequency	1 Gbps	—	125 − 50 ppm	125 + 50 ppm	MHz	
		100 Mbps	—	25 − 50 ppm	25 + 50 ppm	MHz	
		10 Mbps	—	2.5 − 50 ppm	2.5 + 50 ppm	MHz	
	ETn_TXC_TXCLK, ETn_RXC_RXCLK duty cycle	1 Gbps	—	45	55	%	
		100 Mbps 10 Mbps	—	40	60	%	
	ETn_TXC_TXCLK, ETn_TXD0 to ETn_TXD3, ETn_TXCTL_TXEN, ETn_RXC_RXCLK, ETn_RXD0 to ETn_RXD3, ETn_RXCTL_RXDV rise/fall time		t_{RGMIIr}, t_{RGMIIf}	—	0.75*1	ns	
	ETn_TXD0 to ETn_TXD3, ETn_TXCTL_TXEN, ETn_TXC_TXCLK output skew		$t_{RGMIIos}$	−0.5	0.5	ns	
	ETn_RXD0 to ETn_RXD3, ETn_RXCTL_RXDV setup time		t_{RGMIIs}	1	—	ns	
	ETn_RXD0 to ETn_RXD3, ETn_RXCTL_RXDV hold time		t_{RGMIIh}	1	—	ns	
Ethernet (MII)	ETn_TXCLK, ETn_RXCLK cycle time	100 Mbps	t_{MIIck}	40	—	ns	**Figure 10.1-24**
		10 Mbps	t_{MIIck}	400	—	ns	
	ETn_TXC_TXCLK, ETn_RXC_RXCLK frequency	100 Mbps	—	25 − 50 ppm	25 + 50 ppm	MHz	
		10 Mbps	—	2.5 − 50 ppm	2.5 + 50 ppm	MHz	
	ETn_TXD0 to ETn_TXD3, ETn_TXCTL_TXEN, ETn_TXER output delay time		t_{MIId}	0	20	ns	
	ETn_RXD0 to ETn_RXD3, ETn_RXCTL_RXDV, ETn_RXER setup time		t_{MIIs}	10	—	ns	
	ETn_RXD0 to ETn_RXD3, ETn_RXCTL_RXDV, ETn_RXER hold time		t_{MIIh}	10	—	ns	

Note 1. The measurement condition of t_{RGMIIr} and t_{RGMIIf} is in FIGURE 3 in Reduced Gigabit Media Independent Interface (RGMII) 12/10/2000 Version 1.3.

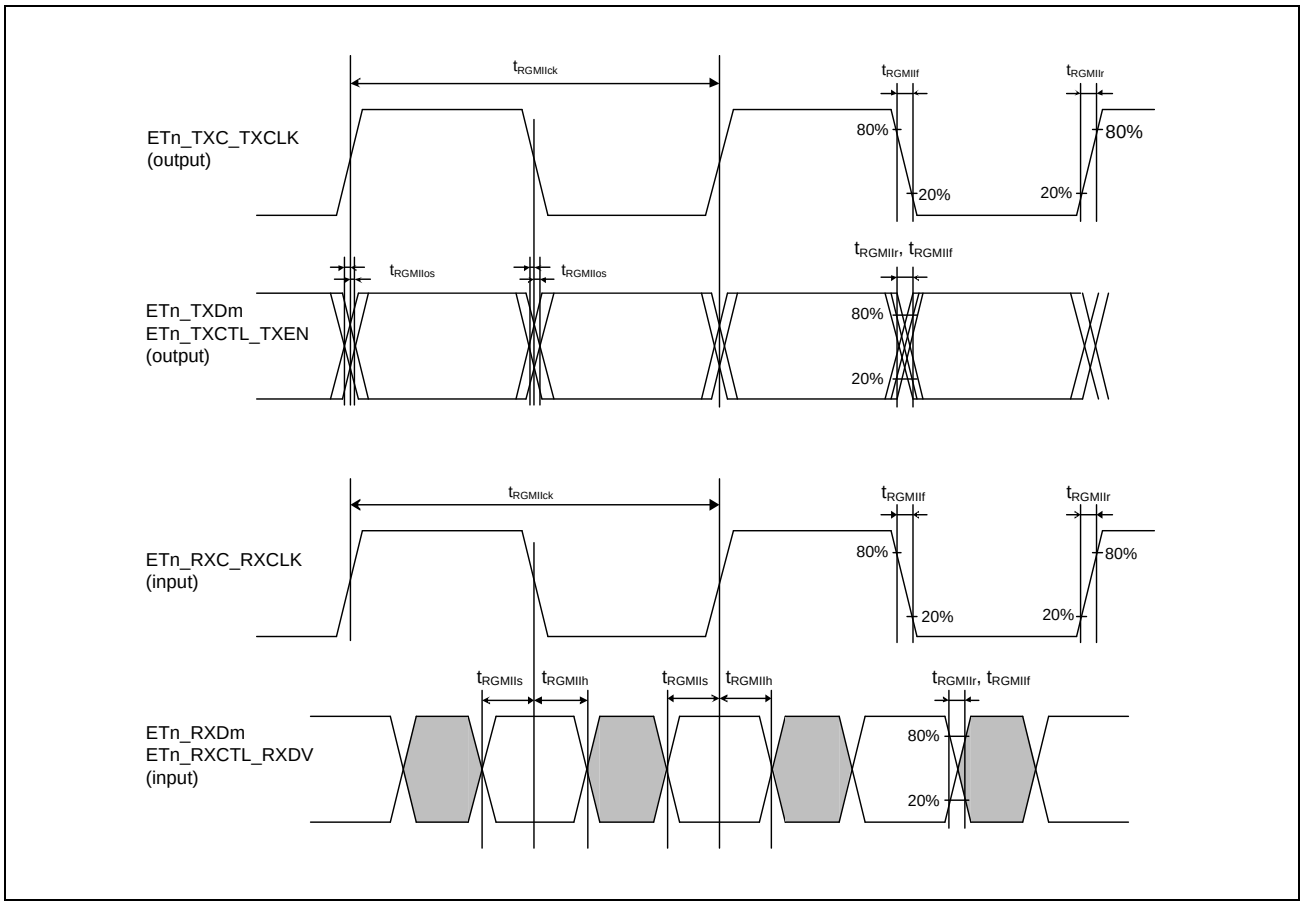


Figure 10.1-23 RGMII Transmission and Reception Timing (n = 0, 1)

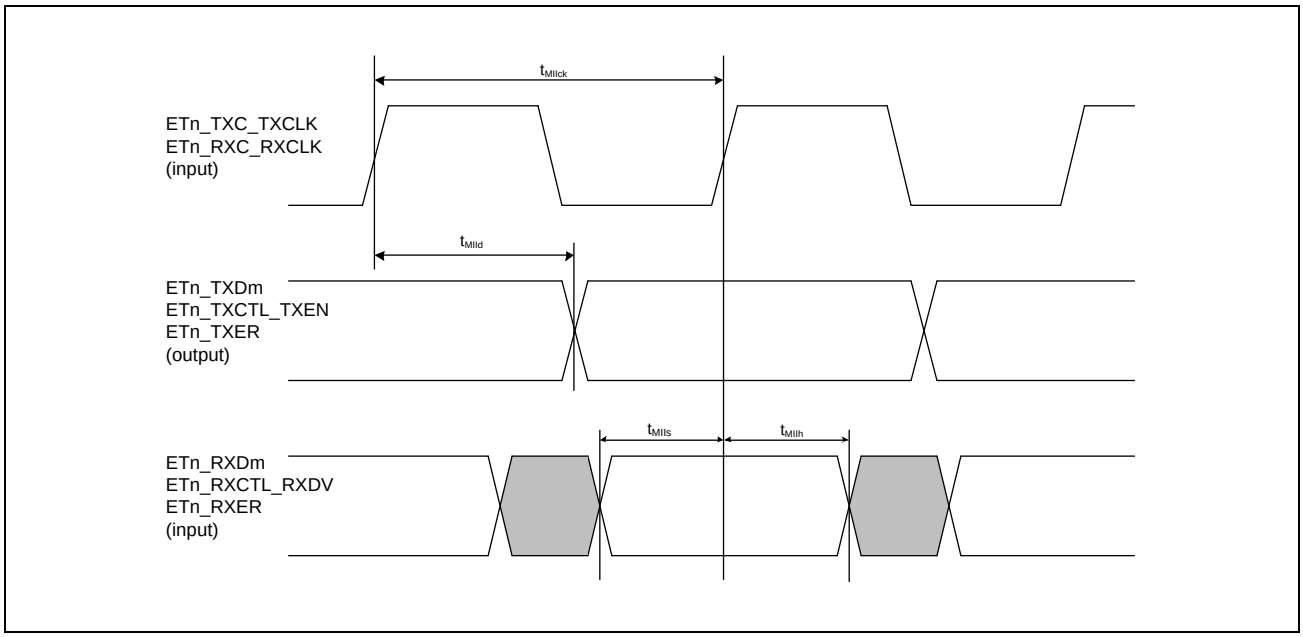


Figure 10.1-24 MII Transmission and Reception Timing (n = 0, 1)

10.1.5.10 USB 3.2 PHY Characteristics

The USB3 PHY of this LSI is compliant with the following USB 3.2 Gen2x1 standard:

Universal Serial Bus 3.2 Specification

10.1.5.11 USB 2.0 PHY Characteristics

The USB2 PHY of this LSI is compliant with the following USB 2.0 standard:

Universal Serial Bus 2.0 Specification

10.1.5.12 PCI Express PHY Characteristics

The PCI Express PHY of this LSI is compliant with the following PCIe standard:

Revision 4.0 of the PCI Express® Base Specification for Gen1/Gen 2/ Gen 3

10.1.5.13 xSPI Timing

Conditions:

- Single-end clock

  V_{OH} = VDD18 × 0.8, V_{OL} = VDD18 × 0.2, C = 15 pF (1.8 V)
  V_{OH} = VDD33 × 0.8, V_{OL} = VDD33 × 0.2, C = 15 pF (3.3 V)

- Data

  V_{OH} = VDD18 × 0.8, V_{OL} = VDD18 × 0.2, C = 15 pF (1.8 V)
  V_{OH} = VDD33 × 0.8, V_{OL} = VDD33 × 0.2, C = 15 pF (3.3 V)

Drive strength: ×6

Table 10.1-18 xSPI Timing (1/2)

Parameter		Symbol	1.8V		3.3V		Unit	Figure
			Min.	Max.	Min.	Max.		
Cycle time	SDR	t_{PERIOD}	7.5	—	12.5	—	ns	**Figure 10.1-25**
	DDR	t_{PERIOD}	7.5	—	12.5	—	ns	
Clock output slew rate		t_{SRck}	0.75/0.56*1	—	1.03	—	V/ns	
Clock duty cycle distortion		t_{CKDCD}	0.0	t_{PERIOD} × 0.05	0.0	t_{PERIOD} × 0.05	ns	
Clock minimum pulse width		t_{CKMPW}	t_{PERIOD} × 0.45	—	t_{PERIOD} × 0.45	—	ns	
Differential clock crossing voltage		$V_{OX(AC)}$	0.4 × VDD18	0.6 × VDD18	—	—	V	
DS duty cycle distortion		t_{DSDCD}	0.0	t_{PERIOD} × 0.04	0.0	t_{PERIOD} × 0.04	ns	
DS minimum pulse width		t_{DSMPW}	t_{PERIOD} × 0.41	—	t_{PERIOD} × 0.41	—	ns	
Data input/output slew rate		t_{SR}	0.75/0.56*1	—	1.03	—	V/ns	
Data input setup time (to CK)	SDR	t_{SU}	2.0	—	2.4	—	ns	**Figure 10.1-26**
Data input hold time (to CK)		t_{H}	1.0	—	1.0	—	ns	
Data output delay time		t_{OD}	—	1.6*2	—	1.8*2	ns	
Data output hold time		t_{OH}	−1.5	—	−2.3	—	ns	
Data output buffer off time		t_{BOFF}	−1.5	—	−2.3	—	ns	
Data input setup time (to DS)	DDR*2	t_{SU}	−0.6/−0.8*1	—	−0.6/−0.8*1	—	ns	**Figure 10.1-27**, **Figure 10.1-28**
Data input hold time (to DS)		t_{H}	t_{PERIOD} × 0.41 − 0.6/0.8*1	—	t_{PERIOD} × 0.41 – 0.6/0.8*1	—	ns	
Data output setup time (to CK)		t_{SUO}	0.6 / 1.0*1,*4	—	1.0	—	ns	
Data output hold time (to CK)		t_{HO}	0.6 / 1.0*1,*4	—	1.0	—	ns	
CS low to clock high		t_{CSLCKH}	6.0/8.0*1,*3	—	8.0*3	—	ns	**Figure 10.1-26** to **Figure 10.1-28**
Clock low to CS high		t_{CKLCSH}	6.0/8.0*1	—	8.0	—	ns	
CS high time		t_{CSTD}	1	16	1	16	t_{PERIOD}	

Table 10.1-18 xSPI Timing (2/2)

Parameter	Symbol	1.8 V		3.3 V		Unit	Figure
		Min.	Max.	Min.	Max.		
DS low to CS high	t_{DSLCSH}	t_{PERIOD} × 0.8*7	—	t_{PERIOD} × 0.8*7	—	ns	**Figure 10.1-29**
CS high to DS Tri-state	t_{CSHDST}	0.0	t_{PERIOD}	0.0	t_{PERIOD}	ns	
CS low to DS low*5	t_{CSLDSL}	0.0	12.5*6	0.0	17.4*6	ns	
DS Tri-state to CS low	t_{DSTCSL}	0.0	—	0.0	—	ns	
CK low to DS low*8	t_{CKLDSL}	—	(0.45 + e) × t_{PERIOD} – 2.0*9	—	(0.45 + e) × t_{PERIOD} – 2.0*9	ns	

Note: CK: XSPI0_CKP (XSPI0_CKN)
DS: XSPI0_DS
CS: XSPI0_CS0N, XSPI0_CS1N

Note 1. Specification at 133 MHz / Specification at 100 MHz

Note 2. These are values when the OEN assertion is extended in the Output Enable Asserting extension bit (COMCFG.OEASTEX = 1b).

Note 3. These are the values when the CS assertion is extended in the CS asserting extension bit (LIOCFGCSn.CSASTEX = 1b).

Note 4. The standard value for xSPI266 is 0.8 ns.

Note 5. If the DS is high during the command & modifier phase when using JESD251 Profile 2.0 memory, the time from CS low to DS high must also meet this specification.

Note 6. When using JESD251 Profile 1.0 memory or JESD251 Profile 2.0 memory with LIOCFGCSn.LATEMD set to 0, this constraint does not apply if the internal pull-down resistor of the DS pin is enabled.

Note 7. These are the values when the t_{CKLDSL} constraint is satisfied.

Note 8. This constraint is necessary only to satisfy the t_{DSLCSH} requirement specified in JESD251, which specifies that t_{DSLCSH} must be at least 80% of t_{PERIOD}. Set LIOCFGCSn.CSNEGEX to the appropriate value to ensure the memory specification complies with this constraint.

Note 9. e: LIOCFGCSn.CSNEGEX (e = 0, 1)

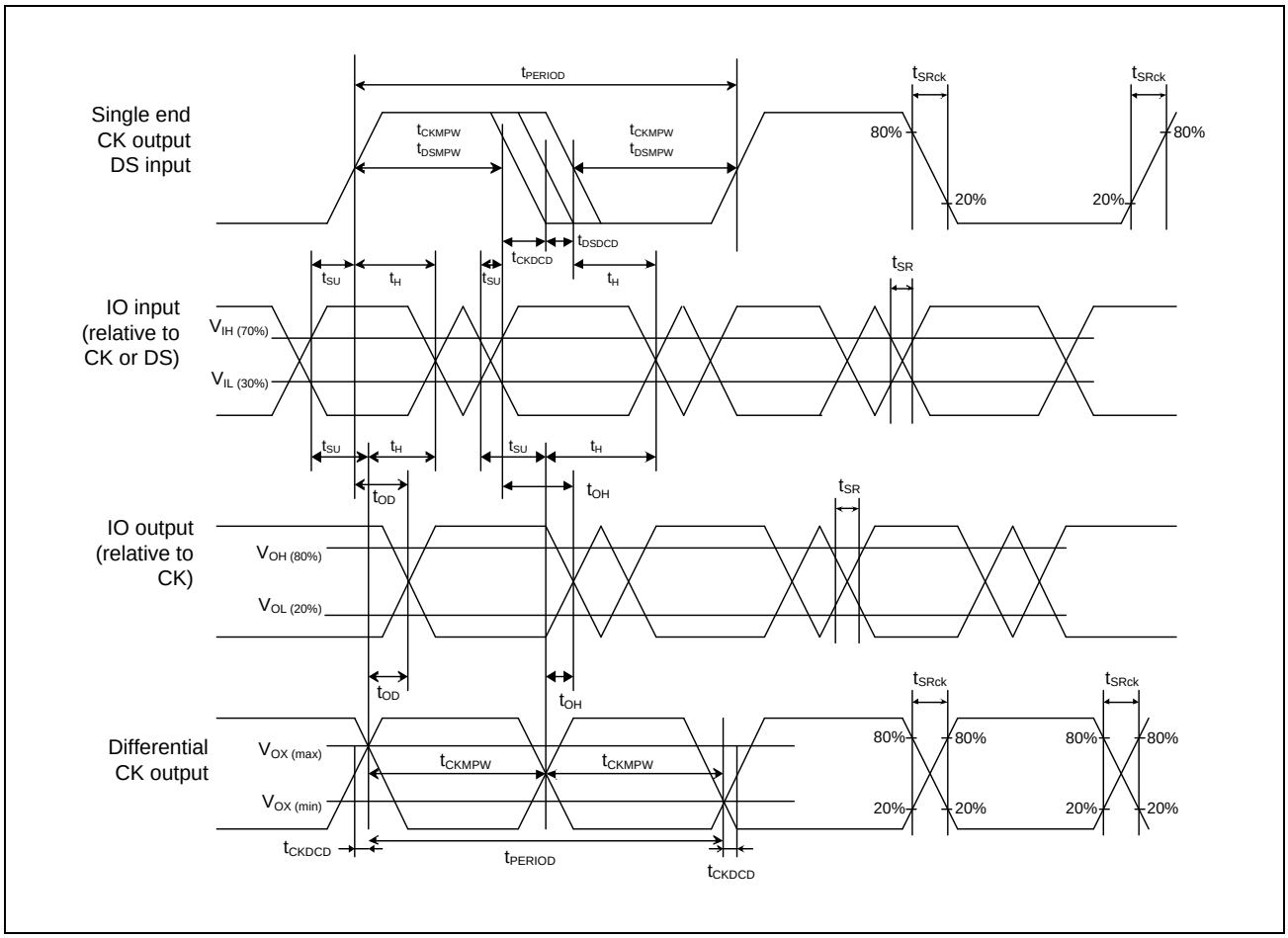


Figure 10.1-25 xSPI Clock / DS Timing

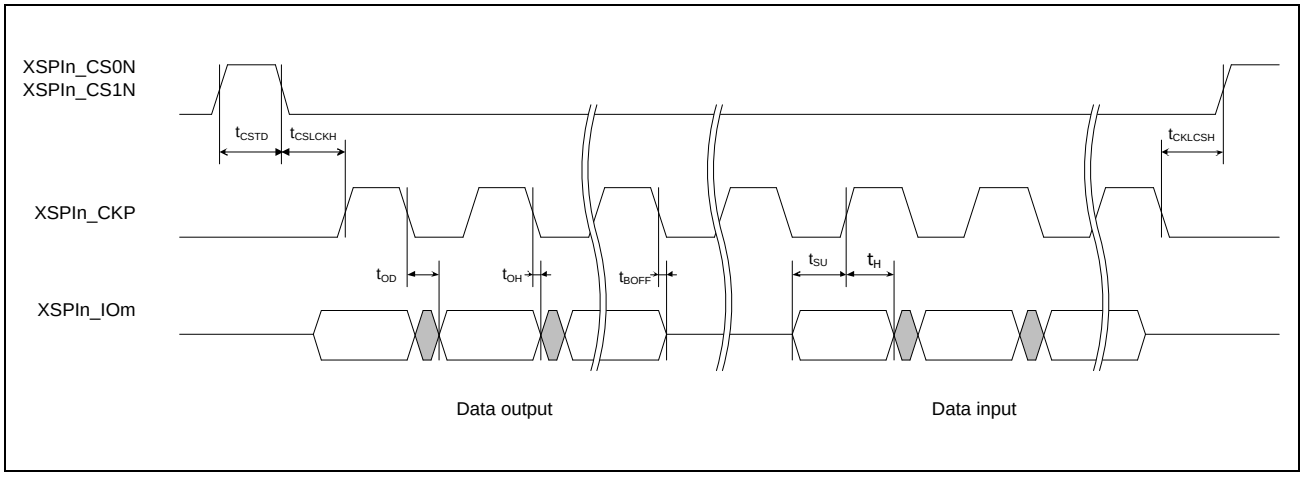


Figure 10.1-26 SDR Transmission and Reception Timing (1S-1S-1S, 1S-2S-2S, 2S-2S-2S, 1S-4S-4S, 4S-4S-4S)

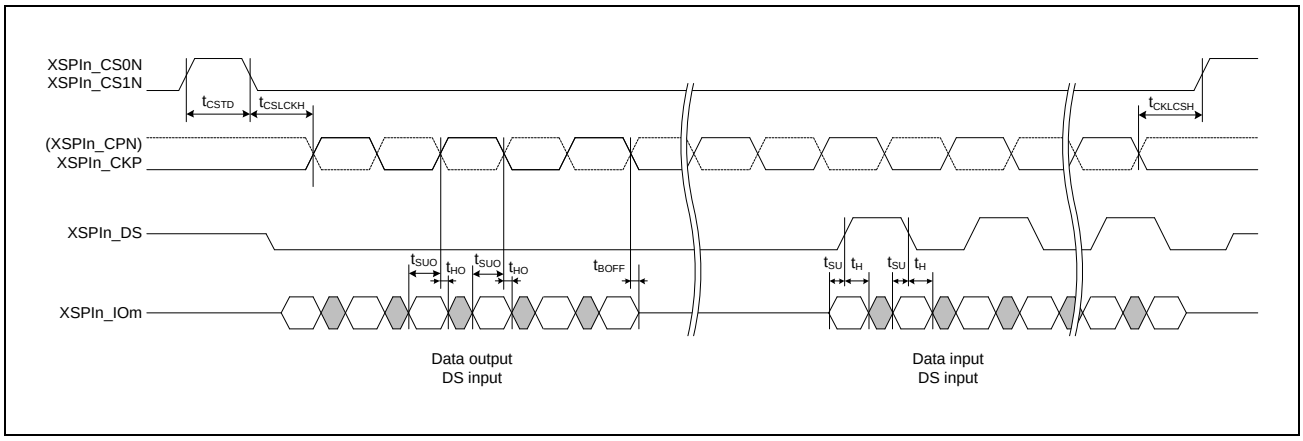


Figure 10.1-27 DDR Transmission and Reception Timing (4S-4D-4D, 8D-8D-8D)

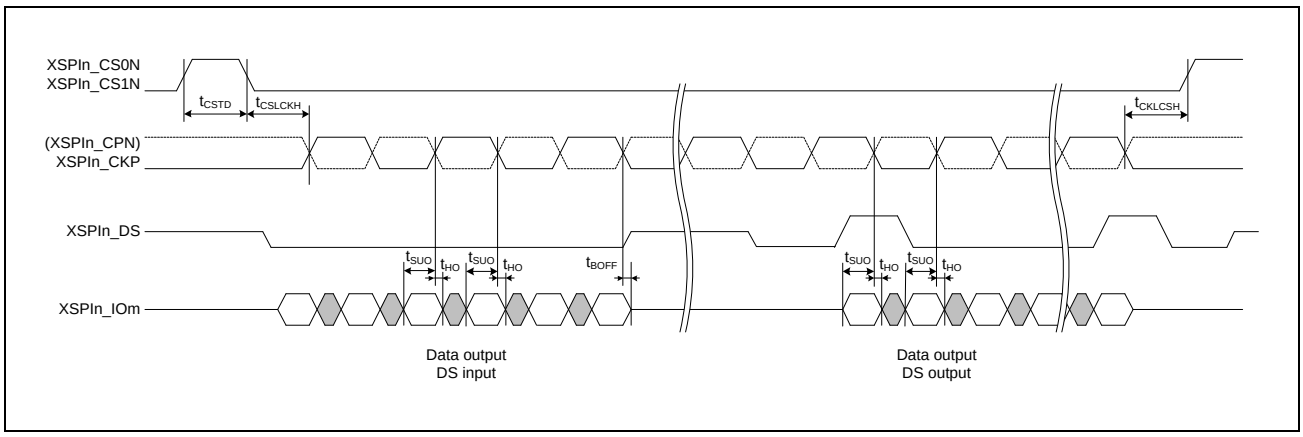


Figure 10.1-28 DDR Transmission and Reception Timing (HyperRAM write)

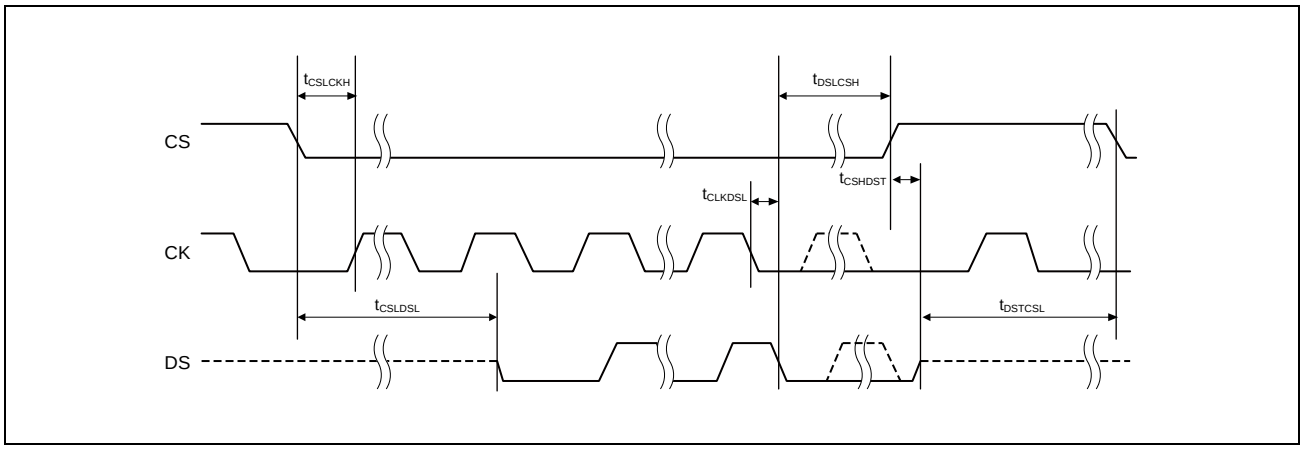


Figure 10.1-29 DS to CS Signal Timing

10.1.5.14 Serial Communications Interface (RSCI) Access Timing

Conditions:

V_{OH} = VDD18 × 0.5, V_{OL} = VDD18 × 0.5, C = 30 pF (1.8 V)

V_{OH} = VDD33 × 0.5, V_{OL} = VDD33 × 0.5, C = 30 pF (3.3 V)

Drive strength: ×2, ×4 (However, ×6 only for SCL (P00) and SDA (P01) of RSCI0 in simple I2C mode)

Table 10.1-19 RSCI Timing (1/2)

Parameter			Symbol	Min.	Max.	Unit	Figure
RSCI (Asynchronous)	Input clock cycle		t_{Scyc}	4	—	$t_{PSCIcyc}$	**Figure 10.1-30**
	Input clock pulse width		t_{SCKW}	0.4	0.6	t_{Scyc}	
	Input clock rise time		t_{SCKr}	—	3	ns	
	Input clock fall time		t_{SCKf}	—	3	ns	
	Output clock cycle		t_{Scyc}	6	—	$t_{PSCIcyc}$	
	Output clock pulse width		t_{SCKW}	0.4	0.6	t_{Scyc}	
	Output clock rise time	V_{DD1833} = 1.8 V	t_{SCKr}	—	6.18*2	ns	
		V_{DD1833} = 3.3 V		—	7.9*2	ns	
	Output clock fall time	V_{DD1833} = 1.8 V	t_{SCKf}	—	6.18*2	ns	
		V_{DD1833} = 3.3 V		—	7.9*2	ns	
RSCI (Simple I2C, Standard mode)	SDA input rise time		t_{Sr}	—	1000	ns	**Figure 10.1-31**
	SDA input fall time		t_{Sf}	—	300	ns	
	SCL, SDA input spike pulse removal time		t_{SP}	0	2 × NFcyc*1	ns	
	Data input setup time		t_{SDAS}	250	—	ns	
	Data input hold time		t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load		C_b	—	400	pF	
RSCI (Simple I2C, Fast mode)	SDA input rise time		t_{Sr}	—	300	ns	**Figure 10.1-31**
	SDA input fall time		t_{Sf}	—	300	ns	
	SCL, SDA input spike pulse removal time		t_{SP}	0	2 × NFcyc*1	ns	
	Data input setup time		t_{SDAS}	100	—	ns	
	Data input hold time		t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load		C_b	—	400	pF	

Table 10.1-19 RSCI Timing (2/2)

Parameter			Symbol	Min.	Max.	Unit	Figure
RSCI (Clock sync, Simple SPI)	SCK output clock cycle (master)		t_{SPcyc}	4	65536	$t_{PSCIcyc}$	**Figure 10.1-32** to **Figure 10.1-37**
	SCK input clock cycle (slave)			4	65536	$t_{PSCIcyc}$	
	SCK clock high-level pulse width		t_{SPCKWH}	0.4	0.6	t_{SPcyc}	
	SCK clock low-level pulse width		t_{SPCKWL}	0.4	0.6	t_{SPcyc}	
	Input clock rise time		t_{SPCKR}	—	3	ns	
	Input clock fall time		t_{SPCKF}	—	3	ns	
	Output clock rise time	V_{DD1833} = 1.8 V	t_{SPCKR}	—	6.18*2	ns	
		V_{DD1833} = 3.3 V		—	7.9*2	ns	
	Output clock fall time	V_{DD1833} = 1.8 V	t_{SPCKF}	—	6.18*2	ns	
		V_{DD1833} = 3.3 V		—	7.9*2	ns	
	Data input setup time	Internal clock	t_{SU}	7	—	ns	
		External clock		3	—	ns	
	Data input hold time	Internal clock	t_H	3	—	ns	
		External clock		3	—	ns	
	Data output delay time	Internal clock	t_{OD}	—	3	ns	
		External clock		—	12	ns	
	Data output hold time	Internal clock	t_{OH}	−3	—	ns	
		External clock		0	—	ns	
	Data rise/fall time	V_{DD1833} = 1.8 V	t_{DR}, t_{DF}	—	6.18*2	ns	
		V_{DD1833} = 3.3 V		—	7.9*2	ns	
	Slave access time	Internal clock	t_{SA}	—	3 × $t_{PSCIcyc}$ + 12	ns	
		External clock		—	3 × $t_{PSCIcyc}$ + 12	ns	
	Slave output release time	Internal clock	t_{REL}	—	3 × $t_{PSCIcyc}$ + 12	ns	
		External clock		—	3 × $t_{PSCIcyc}$ + 12	ns	
RSCI (Simple SPI)	SS input setup time		t_{LEAD}	1	—	t_{SPcyc}	**Figure 10.1-32** to **Figure 10.1-37**
	SS input hold time		t_{LAG}	1	—	t_{SPcyc}	
	SS input rise/fall time		t_{SSR}, t_{SSF}	—	3	ns	

Note: $t_{PSCIcyc}$: RSCI_n_TCLK cycle (n = 1 to 9)

Note 1. NFcyc = 4p × 2q − 1 × $t_{PSCIcyc}$
p: CCR2.CKS[1:0] (p = 0, 1, 2, 3)
q: CCR1.NFCS[2:0] (q = 1, 2, 3, 4)

Note 2. Output transition time from 20% to 80%

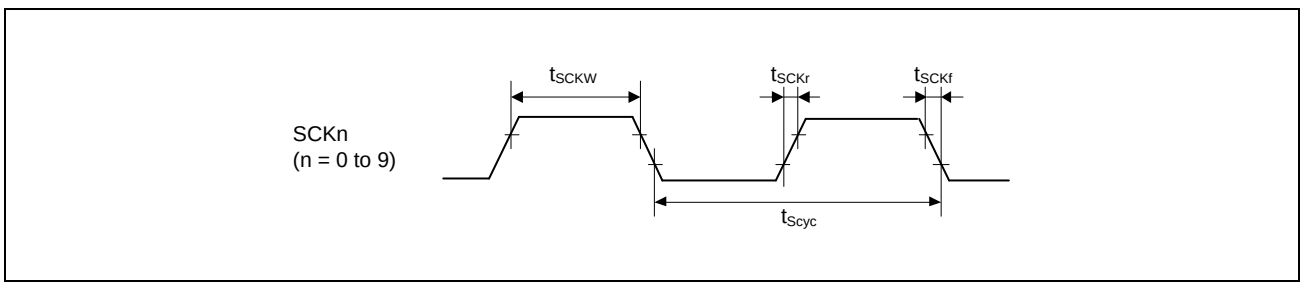
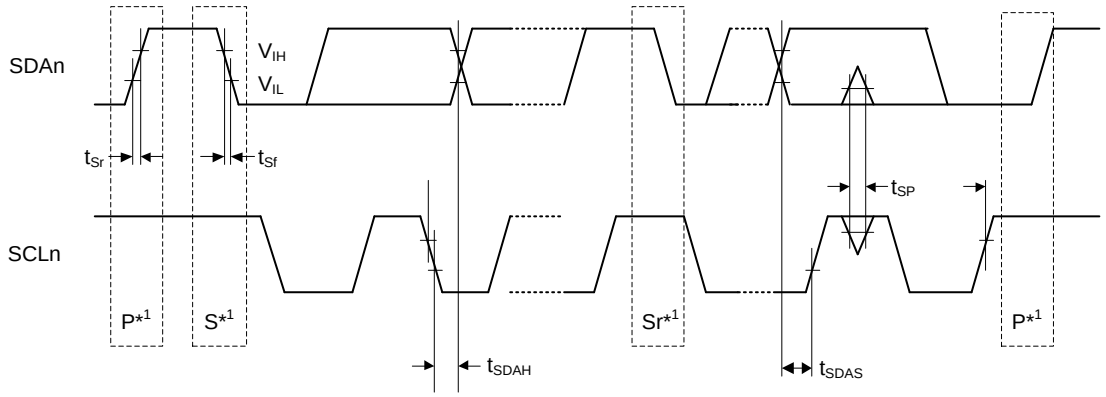


Figure 10.1-30 SCK Clock Input/Output Timing



Note 1. S, P, and Sr indicate the following conditions.
S: Start condition
P: Stop condition
Sr: Restart condition

Figure 10.1-31 RSCI Simple I2C Mode Timing

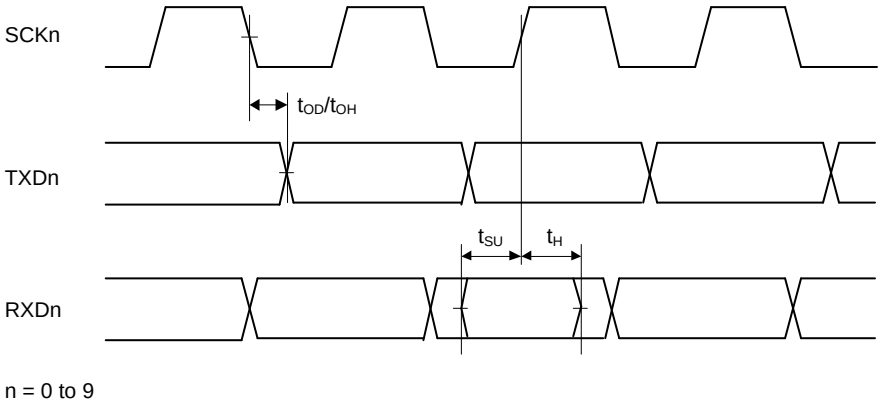


Figure 10.1-32 RSCI Input/Output Timing in Clock Synchronous Mode

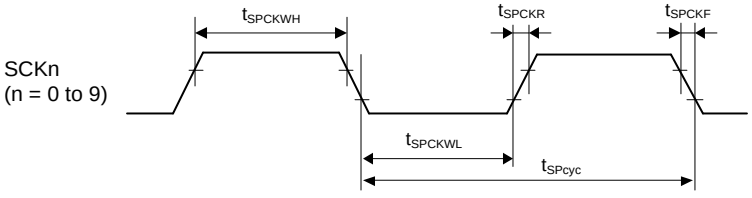


Figure 10.1-33 RSCI Simple SPI Mode Clock Timing

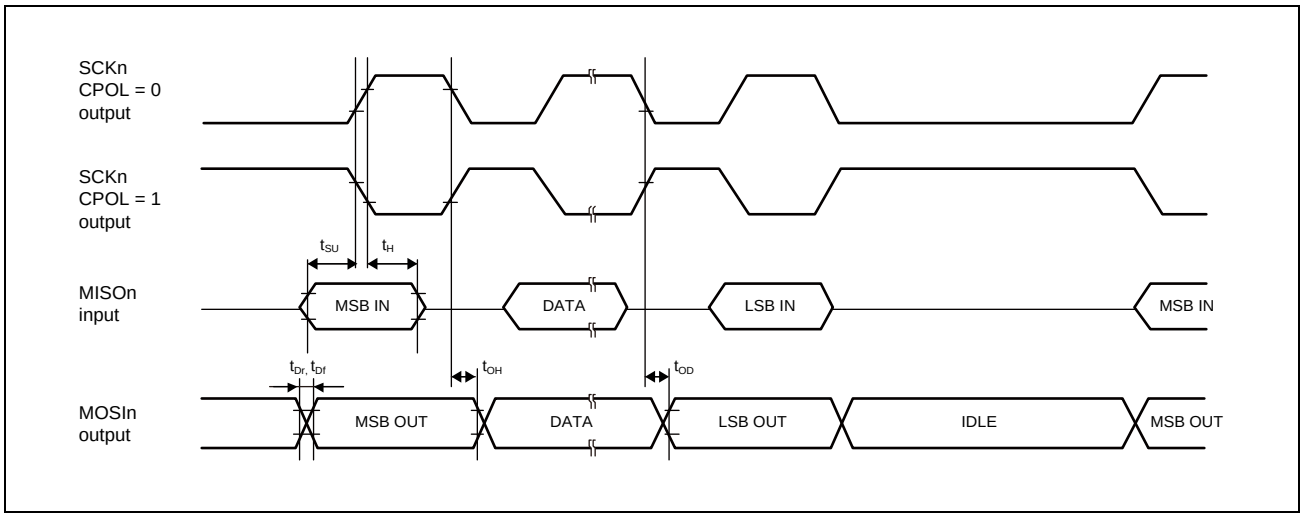


Figure 10.1-34 RSCI Simple SPI Mode Timing for Master when CPHA = 0

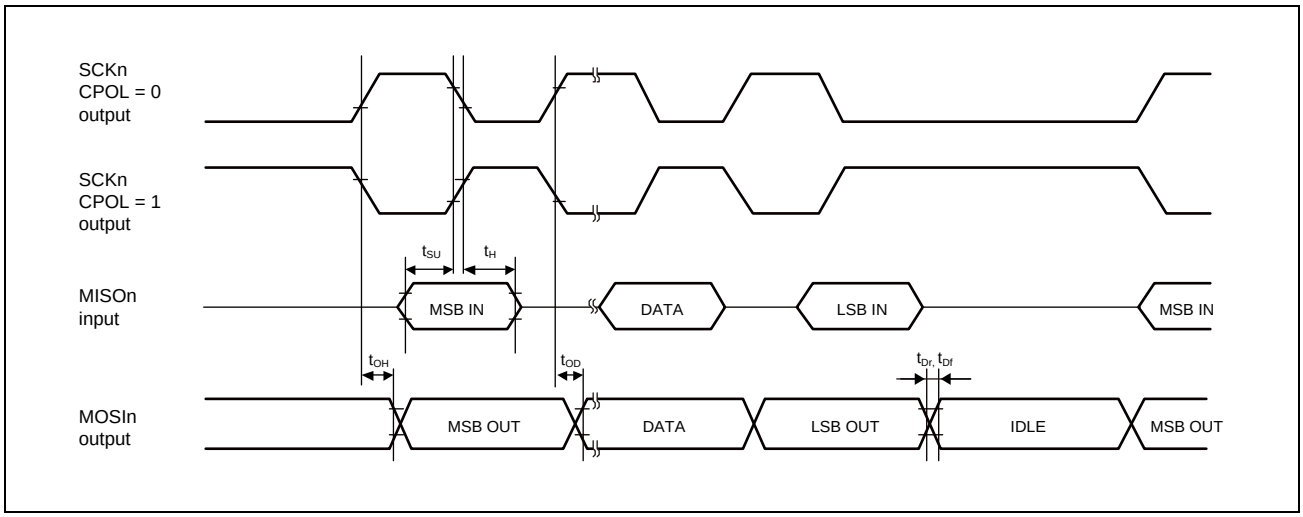


Figure 10.1-35 RSCI Simple SPI Mode Timing for Master when CPHA = 1

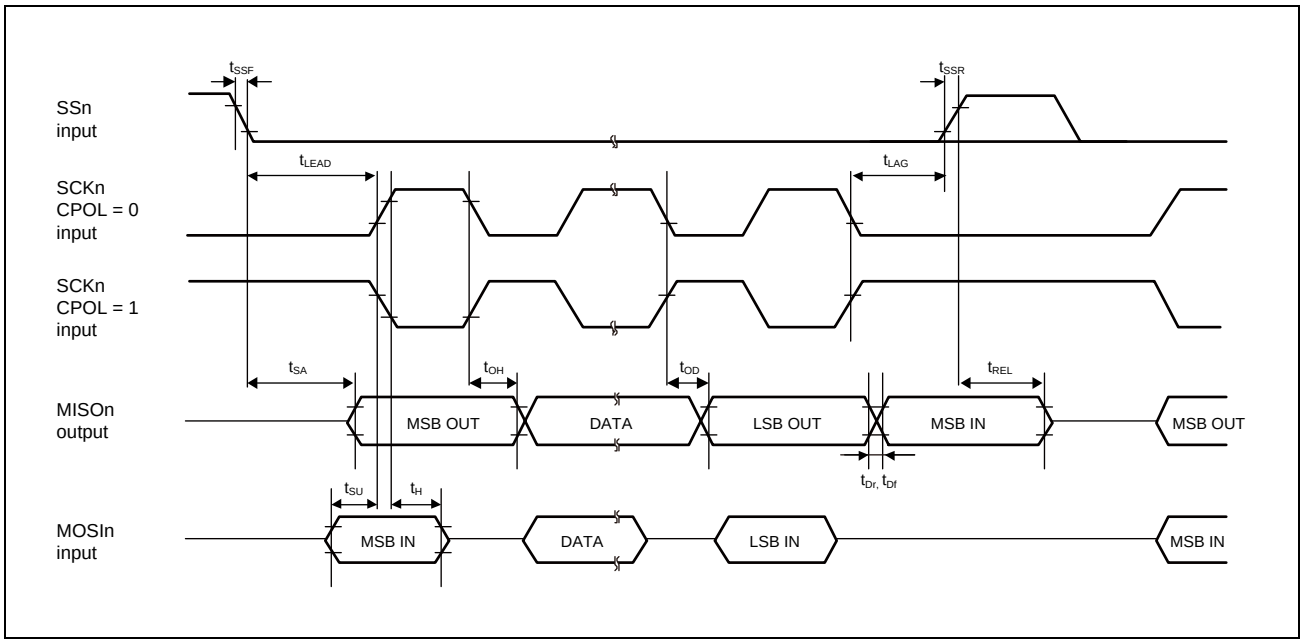


Figure 10.1-36 RSCI Simple SPI Mode Timing for Slave when CPHA = 0

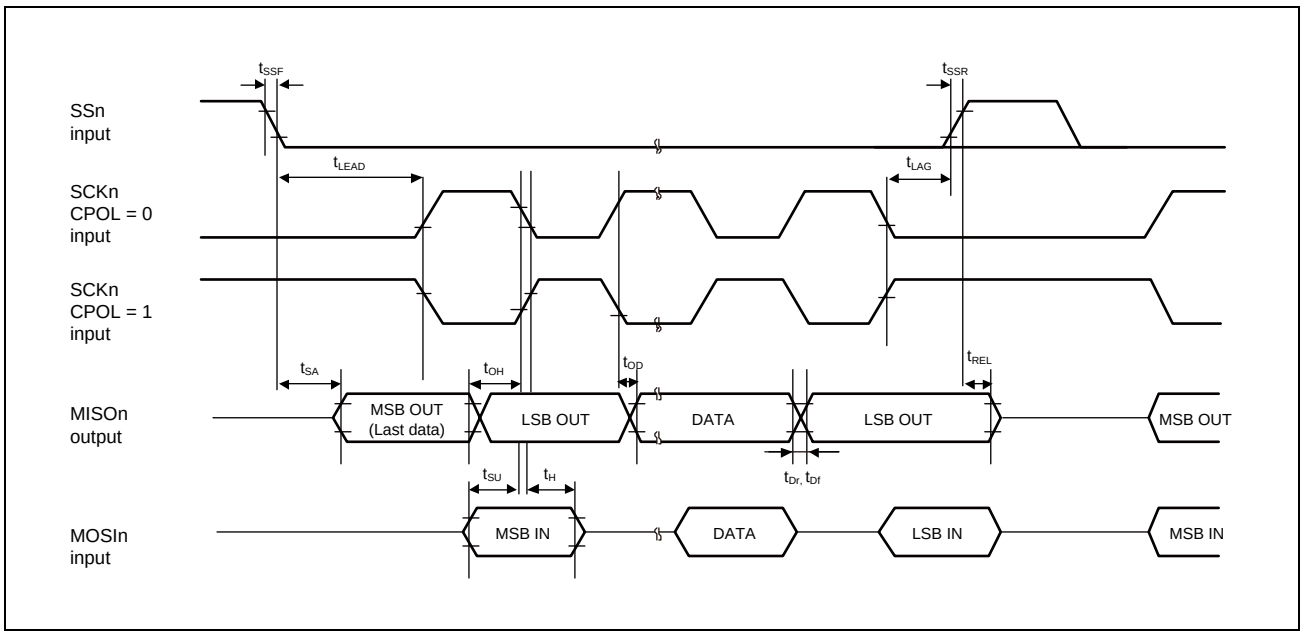


Figure 10.1-37 RSCI Simple SPI Mode Timing for Slave when CPHA = 1

10.1.5.15 Renesas Serial Peripheral Interface (RSPI) Access Timing

Conditions:

V_{OH} = VDD18 × 0.5, V_{OL} = VDD18 × 0.5, C = 30 pF (1.8 V)

V_{OH} = VDD33 × 0.5, V_{OL} = VDD33 × 0.5, C = 30 pF (3.3 V)

Drive strength: ×6

Table 10.1-20 RSPI Timing (1/2)

Parameter		Symbol	Min.*1	Max.*1	Unit	Figure
RSPCK clock cycle	Master	t_{SPcyc}	4	4096	t_{SPIcyc}	**Figure 10.1-38**
	Slave		4 / 8*6	4096	t_{SPIcyc}	
RSPCK clock high-level pulse width	Master	t_{SPCKWH}	$(t_{SPcyc} - t_{SPCKr} - t_{SPCKf})/2 - 2.5$	—	ns	
	Slave		1	—	t_{SPIcyc}	
RSPCK clock low-level pulse width	Master	t_{SPCKWL}	$(t_{SPcyc} - t_{SPCKr} - t_{SPCKf})/2 - 2.5$	—	ns	
	Slave		1	—	t_{SPIcyc}	
RSPCK clock rise/fall time	Output	t_{SPCKr}, t_{SPCKf}	—	3*5	ns	
	Input		—	3*5	ns	
Data input setup time	Master	t_{SU}	5	—	ns	**Figure 10.1-39** to **Figure 10.1-45**
	Slave		3	—	ns	
Data input hold time	Master	t_H	3	—	ns	
	Slave		3	—	ns	
SSL setup time	Master	t_{LEAD}	$N \times t_{SPcyc} - 3$*2	$N \times t_{SPcyc} + 3$*2	ns	**Figure 10.1-39** to **Figure 10.1-42**
	Slave		5	—	t_{SPIcyc}	
SSL hold time	Master	t_{LAG}	$N \times t_{SPcyc} - 3$*3	$N \times t_{SPcyc} + 3$*3	ns	
	Slave		5	—	t_{SPIcyc}	
Continuous transmission delay	Master	t_{TD}	$t_{SPcyc} + 2 \times t_{SPIcyc}$	$8 \times t_{SPcyc} + 2 \times t_{SPIcyc}$	ns	
	Slave		$t_{SPcyc} + 5 \times t_{SPIcyc}$	—	ns	
TI-SSP SS input setup time		t_{TISS}	3	—	ns	**Figure 10.1-43** to **Figure 10.1-45**
TI-SSP SS input hold time		t_{TISH}	3	—	ns	
TI-SSP next access time		t_{TIND}	M*4	—	t_{SPIcyc}	
TI-SSP Master SS output delay		t_{TISSOD}	−3	3	ns	
TI-SSP Master OE delay 1		$t_{TIMOED1}$	—	2	ns	
TI-SSP Master OE delay 2		$t_{TIMOED2}$	—	2	ns	
TI-SSP Slave OE delay 1		$t_{TISOED1}$	—	7.5 / 10*6	ns	
TI-SSP Slave OE delay 2		$t_{TISOED2}$	—	7.5 / 10*6	ns	
SSL Activation to Data Output Delay		t_{OD1}	—	3	ns	**Figure 10.1-39**
Data output delay time	Master	t_{OD}	—	3	ns	**Figure 10.1-39** to **Figure 10.1-45**
	Slave		—	7.5 / 10*6	ns	
Data output hold time	Master	t_{OH}	−3	—	ns	
	Slave		3	—	ns	
MOSI, MISO rise/fall time	Output	t_{Dr}, t_{Df}	—	3*5	ns	
	Input		—	1	µs	
SSL rise/fall time	Output	t_{SSLr}, t_{SSLf}	—	3*5	ns	**Figure 10.1-39**, **Figure 10.1-40**
	Input		—	1	µs	

Table 10.1-20 RSPI Timing (2/2)

Parameter	Symbol	Min.*1	Max.*1	Unit	Figure
Slave access time	t_{SA}	—	8 / 10*6	ns	**Figure 10.1-41**, **Figure 10.1-42**
Slave output release time	t_{REL}	—	8 / 10*6	ns	

Note 1. t_{SPIcyc}: RSPI_n_TCLK cycle (n = 0 to 2)

Note 2. RSPIm_SPCKD.SCKDL[2:0] set value + 1 (1 to 8)

Note 3. RSPIm_SSLND.SLNDL[2:0] set value + 1 (1 to 8)

Note 4. RSPIm_SSLND.SLNDL[2:0] set value + 2 (2 to 9)

Note 5. Output transition time from 20% to 80%

Note 6. This value when PMx, P3x pins are used as RSPI (x = 0 to 7)

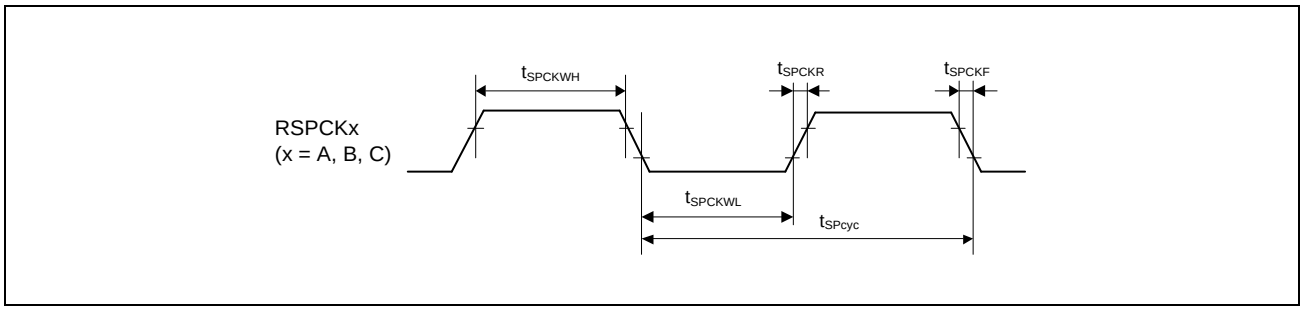


Figure 10.1-38 RSPI Clock Timing

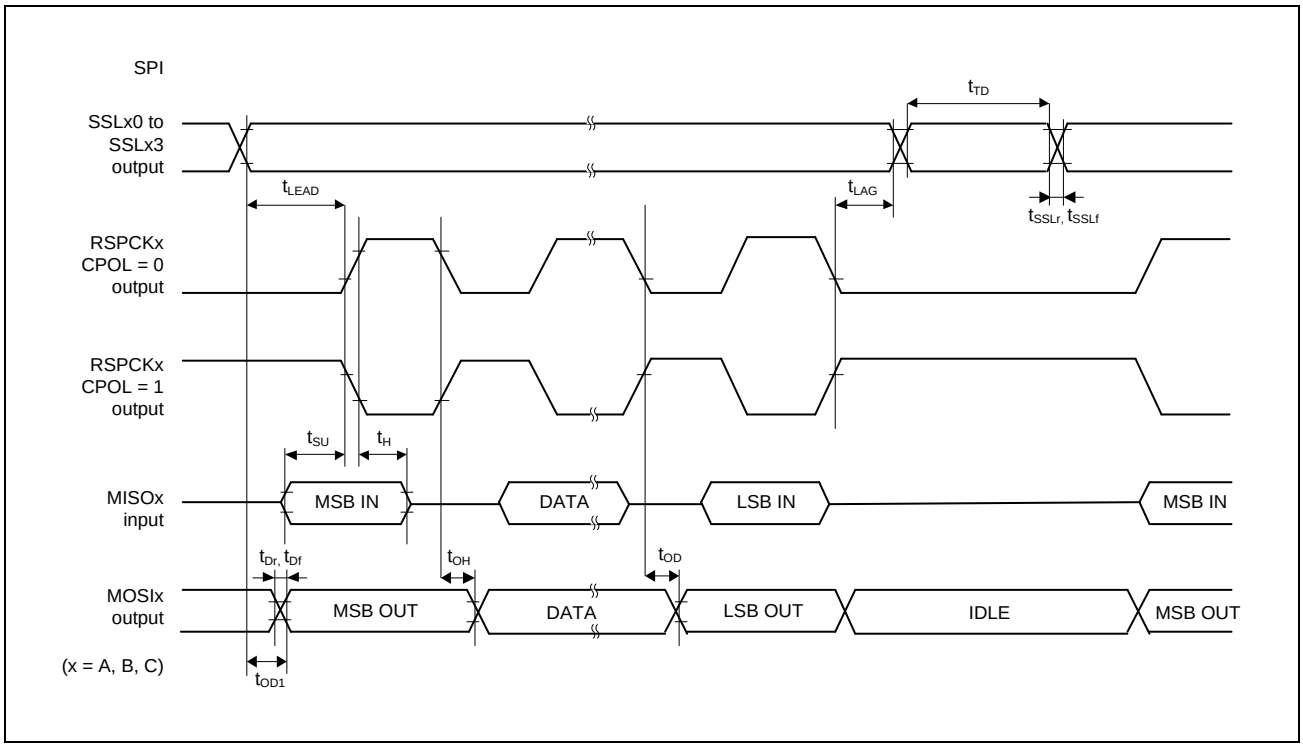


Figure 10.1-39 RSPI Timing (Master, Motorola RSPI, CPHA = 0)

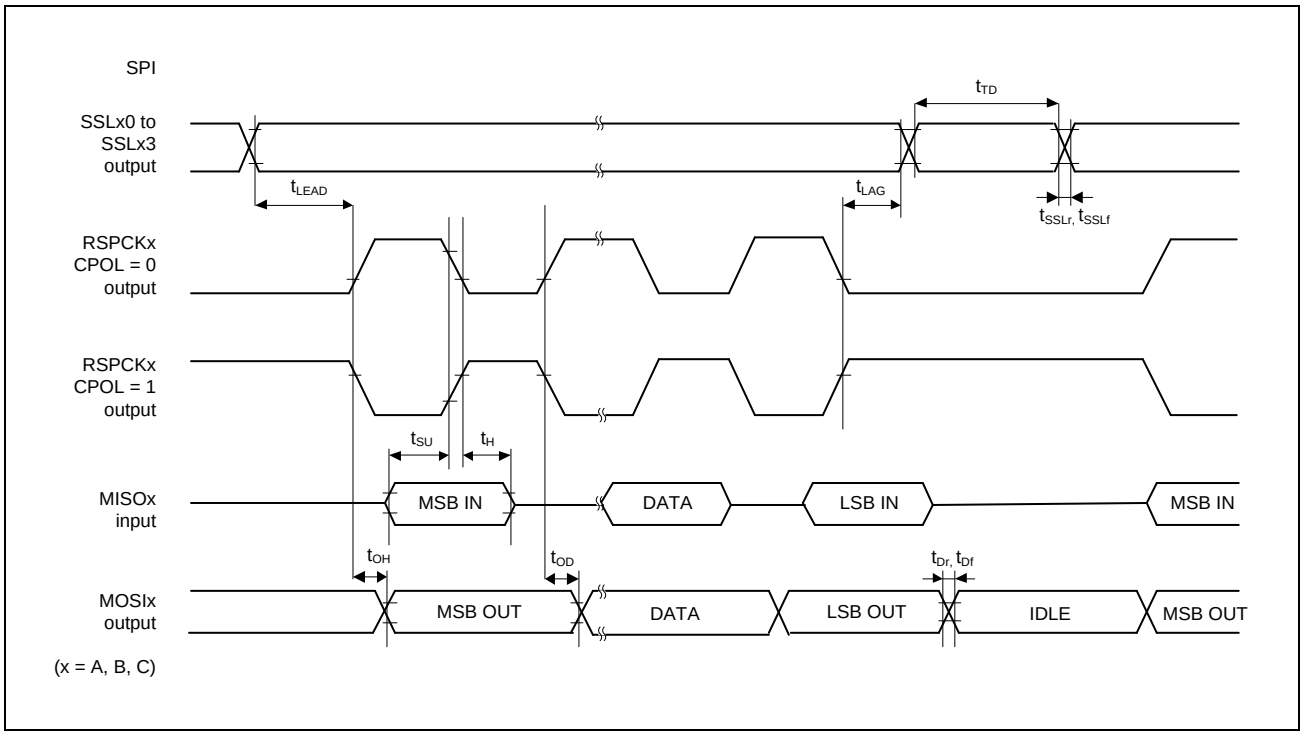


Figure 10.1-40 RSPI Timing (Master, Motorola RSPI, CPHA = 1)

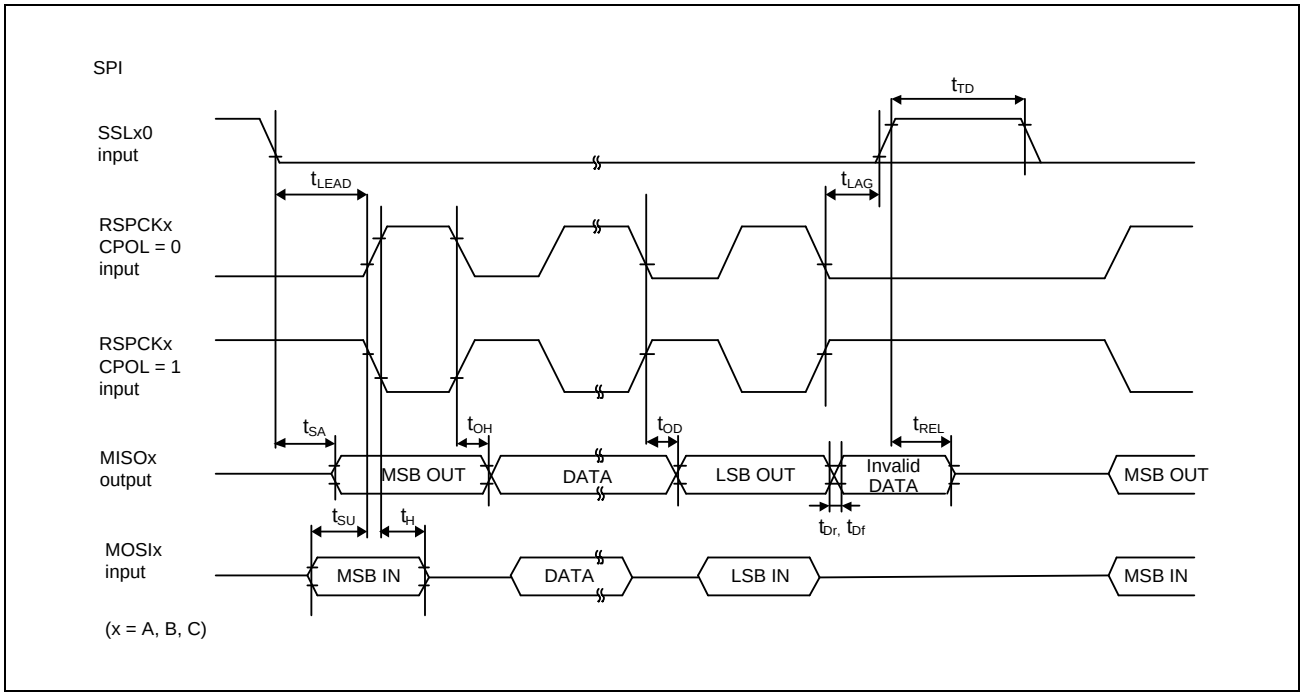


Figure 10.1-41 RSPI Timing (Slave, Motorola RSPI, CPHA = 0)

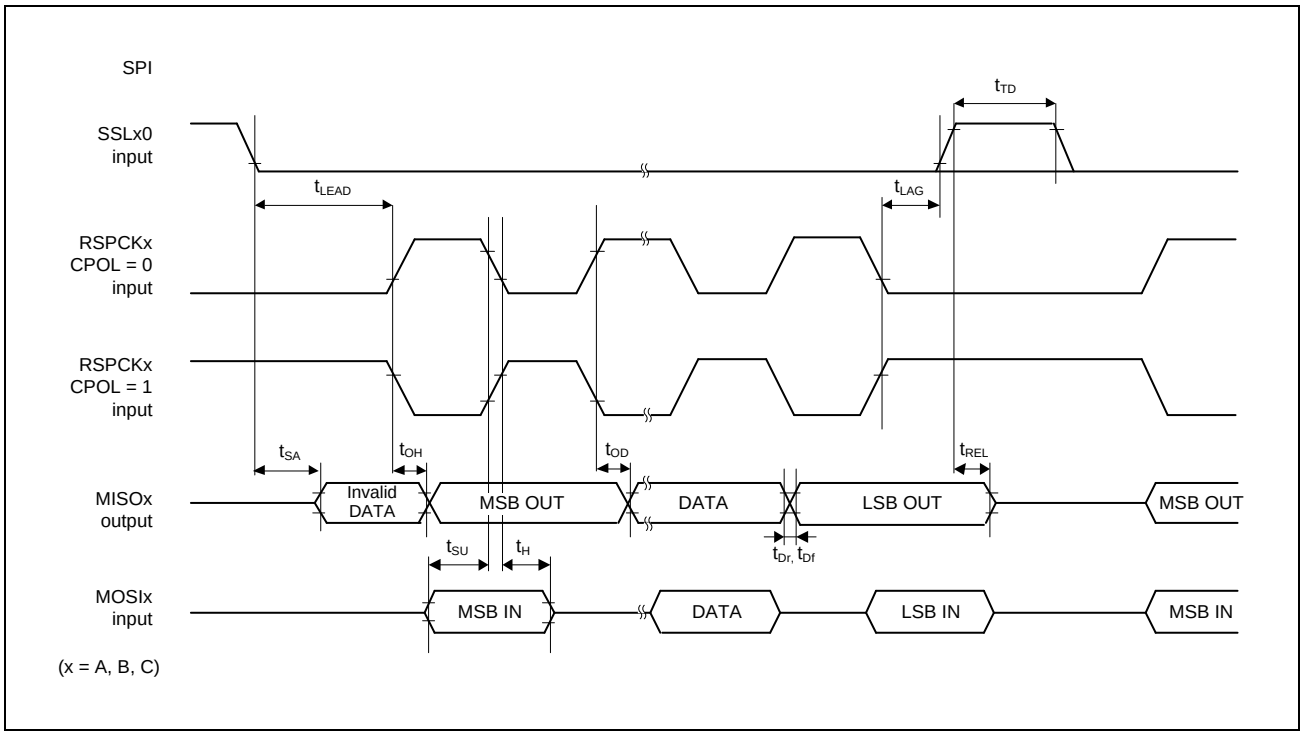


Figure 10.1-42 RSPI Timing (Slave, Motorola RSPI, CPHA = 1)

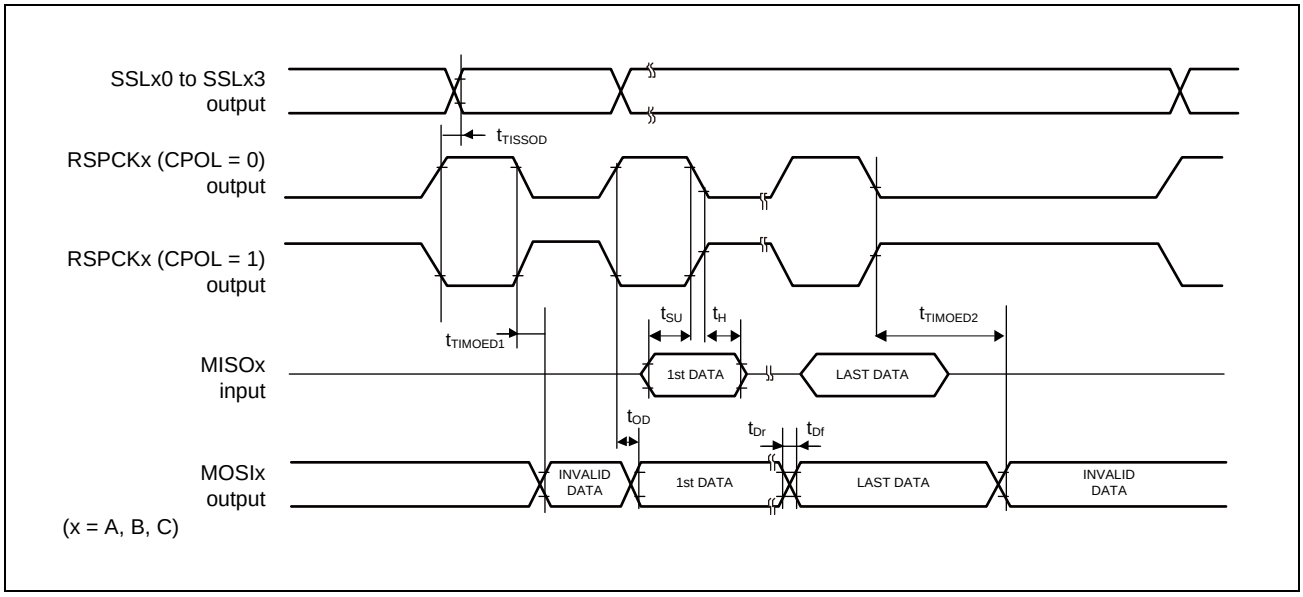


Figure 10.1-43 RSPI Timing (Master, TI SSP)

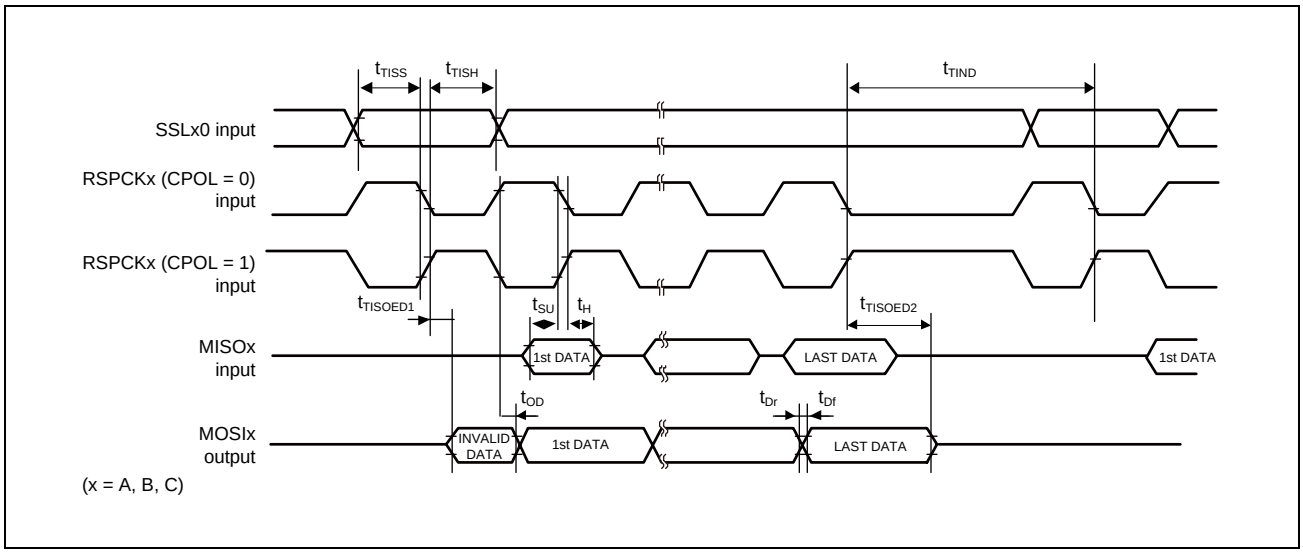


Figure 10.1-44 RSPI Timing (Slave, TI-SSP, with delay in burst transfer)

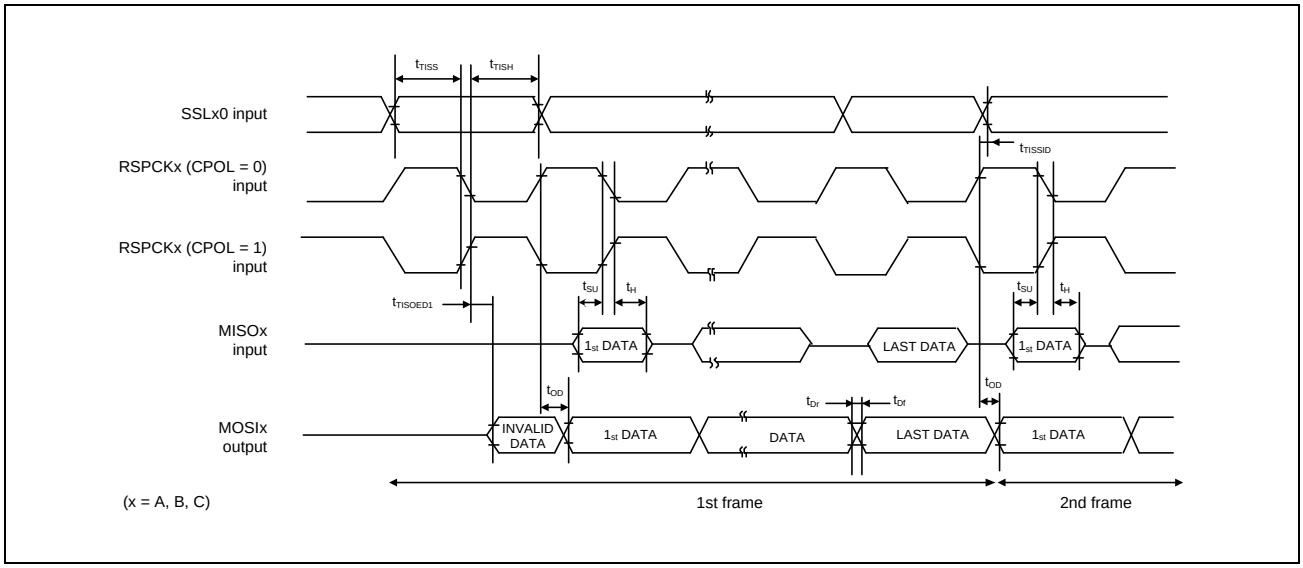


Figure 10.1-45 RSPI Timing (Slave, TI-SSP, without delay in burst transfer)

10.1.5.16 Renesas IIC Bus Interface (RIIC) Access Timing

Conditions: V_{OL} = 0.4 V

Drive strength: ×6

Table 10.1-21 RIIC Timing

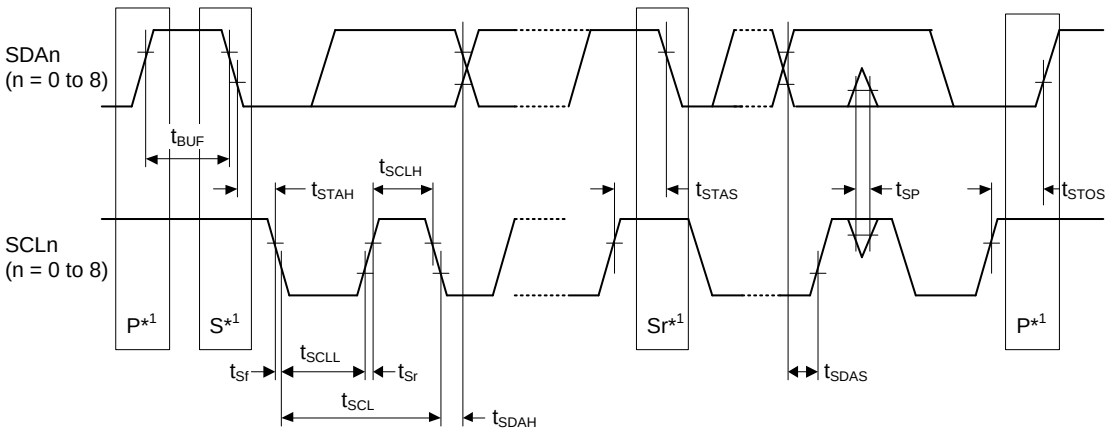
Parameter		Symbol	Min.*1,*2	Max.*1,*2	Unit	Figure
RIIC (Standard-mode)	SCL input cycle time	t_{SCL}	6(12) × t_{IICcyc} + 1300	—	ns	**Figure 10.1-46**
	SCL input high-level pulse width	t_{SCLH}	3(6) × t_{IICcyc} + 300	—	ns	
	SCL input low-level pulse width	t_{SCLL}	3(6) × t_{IICcyc} + 300	—	ns	
	SCL, SDA input rise time	t_{sr}	—	1000	ns	
	SCL, SDA input fall time	t_{sf}	—	300	ns	
	SCL, SDA input spike pulse removal time	t_{SP}	0	1(4) × t_{IICcyc}	ns	
	SDA input bus free time	t_{BUF}	3(6) × t_{IICcyc} + 300	—	ns	
	Start condition input hold time	t_{STAH}	t_{IICcyc} + 300	—	ns	
	Restart condition input setup time	t_{STAS}	1000	—	ns	
	Stop condition input setup time	t_{STOS}	1000	—	ns	
	Data input setup time	t_{SDAS}	t_{IICcyc} + 50	—	ns	
	Data input hold time	t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load	C_b	—	400	pF	
RIIC (Fast-mode)	SCL input cycle time	t_{SCL}	6(12) × t_{IICcyc} + 600	—	ns	**Figure 10.1-46**
	SCL input high-level pulse width	t_{SCLH}	3(6) × t_{IICcyc} + 300	—	ns	
	SCL input low-level pulse width	t_{SCLL}	3(6) × t_{IICcyc} + 300	—	ns	
	SCL, SDA input rise time	t_{sr}	—*4	300	ns	
	SCL, SDA input fall time	t_{sf}	—*4	300	ns	
	SCL, SDA input spike pulse removal time	t_{SP}	0	1(4) × t_{IICcyc}	ns	
	SDA input bus free time	t_{BUF}	3(6) × t_{IICcyc} + 300	—	ns	
	Start condition input hold time	t_{STAH}	t_{IICcyc} + 300	—	ns	
	Restart condition input setup time	t_{STAS}	300	—	ns	
	Stop condition input setup time	t_{STOS}	300	—	ns	
	Data input setup time	t_{SDAS}	t_{IICcyc} + 50	—	ns	
	Data input hold time	t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load*3	C_b	—	400	pF	
RIIC (Fast-mode Plus)	SCL input cycle time	t_{SCL}	6(12) × t_{IICcyc} + 240	—	ns	**Figure 10.1-46**
	SCL input high-level pulse width	t_{SCLH}	3(6) × t_{IICcyc} + 120	—	ns	
	SCL input low-level pulse width	t_{SCLL}	3(6) × t_{IICcyc} + 120	—	ns	
	SCL, SDA input rise time	t_{sr}	—*4	120	ns	
	SCL, SDA input fall time	t_{sf}	—*4	120	ns	
	SCL, SDA input spike pulse removal time	t_{SP}	0	1(4) × t_{IICcyc}	ns	
	SDA input bus free time	t_{BUF}	3(6) × t_{IICcyc} + 120	—	ns	
	Start condition input hold time	t_{STAH}	t_{IICcyc} + 300	—	ns	
	Restart condition input setup time	t_{STAS}	300	—	ns	
	Stop condition input setup time	t_{STOS}	300	—	ns	
	Data input setup time	t_{SDAS}	t_{IICcyc} + 50	—	ns	
	Data input hold time	t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load*3	C_b	—	550	pF	

Note 1. t_{IICcyc}: RIIC internal reference clock (IICϕ) cycle

Note 2. The values outside parentheses apply when the value of the ICMR3.NF[1:0] bits is 00b while the digital filter is enabled by setting ICFER.NFE = 1. The values within parentheses apply when the value of the ICMR3.NF[1:0] bits is 11b while the digital filter is enabled by setting ICFER.NFE = 1.

Note 3. C_b is the total capacitance of the bus lines.

Note 4. The minimum values are not specified for t_{sr} and t_{sf} in Fast-mode or Fast-mode Plus.



Note 1. S, P, and Sr indicate the following conditions.
S: Start condition
P: Stop condition
Sr: Restart condition

Figure 10.1-46 RIIC Bus Interface Input/Output Timing

10.1.5.17 I3C Timing

Conditions: V_{OH} = VDD1218_I3C × 0.5, V_{OL} = VDD1218_I3C × 0.5, C = 30 pF (1.2 V or 1.8 V)

Drive strength: ×6

Table 10.1-22 I3C Timing

Parameter		Symbol	Min.*1	Max.	Unit	Figure
IIC (Standard mode, SMBus)	SCL3n cycle time	t_{SCL}	4(36) × t_{IICcyc} + 4 × t_{Pcyc} + 1300	—	ns	**Figure 10.1-47**
	SCL3n high-level pulse width	t_{SCLH}	2(18) × t_{IICcyc} + 2 × t_{Pcyc} + 300	—	ns	
	SCL3n low-level pulse width	t_{SCLL}	2(18) × t_{IICcyc} + 2 × t_{Pcyc} + 800	—	ns	
	SCL3n, SDA3n rise time	t_{Sr}	—	1000	ns	
	SCL3n, SDA3n fall time	t_{Sf}	—	300	ns	
	SCL3n, SDA3n spike pulse removal time	t_{SP}	0	1(16) × t_{IICcyc}	ns	
	SDA3n bus free time	t_{BUF}	3(20) × t_{IICcyc} + 300	—	ns	
	Hold time for START condition	t_{STAH}	t_{IICcyc} + 300	—	ns	
	Setup time for repeated START condition	t_{STAS}	1000	—	ns	
	Setup time for STOP condition	t_{STOS}	1000	—	ns	
	Data setup time	t_{SDAS}	t_{IICcyc} + 50	—	ns	
	Data hold time	t_{SDAH}	0	—	ns	
	SCL3n, SDA3n capacitive load	C_b	—	400	pF	
IIC (Fast mode)	SCL3n cycle time	t_{SCL}	4(36) × t_{IICcyc} + 4 × t_{Pcyc} + 600	—	ns	**Figure 10.1-47**
	SCL3n high-level pulse width	t_{SCLH}	2(18) × t_{IICcyc} + 2 × t_{Pcyc} + 300	—	ns	
	SCL3n low-level pulse width	t_{SCLL}	2(18) × t_{IICcyc} + 2 × t_{Pcyc} + 300	—	ns	
	SCL3n, SDA3n rise time	t_{Sr}	—	300	ns	
	SCL3n, SDA3n fall time	t_{Sf}	—	300	ns	
	SCL3n, SDA3n spike pulse removal time	t_{SP}	0	1(16) × t_{IICcyc}	ns	
	SDA3n bus free time	t_{BUF}	3(20) × t_{IICcyc} + 300	—	ns	
	Hold time for START condition	t_{STAH}	t_{IICcyc} + 300	—	ns	
	Setup time for repeated START condition	t_{STAS}	300	—	ns	
	Setup time for STOP condition	t_{STOS}	300	—	ns	
	Data setup time	t_{SDAS}	t_{IICcyc} + 50	—	ns	
	Data hold time	t_{SDAH}	0	—	ns	
	SCL3n, SDA3n capacitive load	C_b	—	400	pF	

Note 1. t_{IICcyc}: I3C internal reference clock (I3Cϕ) cycle, t_{Pcyc}: I3C_0_TCLK cycle. Refer to **7.8.3.2.14 Reference Clock Control Register** for details. Values in parentheses apply when INCTL.DNFS[3:0] is set to 1111b while the digital filter is enabled with DNFE.DNFE set to 1b.

Table 10.1-23 IIC Timing (Fast-mode+)

Parameter		Symbol	Min.*1	Max.	Unit	Figure
IIC (Fast-mode+)	SCL3n cycle time	t_{SCL}	4(26) × t_{IICcyc} + 4 × t_{Pcyc} + 240	—	ns	**Figure 10.1-47**
	SCL3n high-level pulse width	t_{SCLH}	2(18) × t_{IICcyc} + 2 × t_{Pcyc} + 120	—	ns	
	SCL3n low-level pulse width	t_{SCLL}	2(18) × t_{IICcyc} + 2 × t_{Pcyc} + 120	—	ns	
	SCL3n, SDA3n rise time	t_{Sr}	—	120	ns	
	SCL3n, SDA3n fall time	t_{Sf}	—	120	ns	
	SCL3n, SDA3n spike pulse removal time	t_{SP}	—	1(16) × t_{IICcyc}	ns	
	SDA3n bus free time	t_{BUF}	3(20) × t_{IICcyc} + 120	—	ns	
	Hold time for START condition	t_{STAH}	t_{IICcyc} + 135	—	ns	
	Setup time for repeated START condition	t_{STAS}	260	—	ns	
	Setup time for STOP condition	t_{STOS}	260	—	ns	
	Data setup time	t_{SDAS}	50	—	ns	
	Data hold time	t_{SDAH}	0	—	ns	
	SCL3n, SDA3n capacitive load	C_b	—	550	pF	

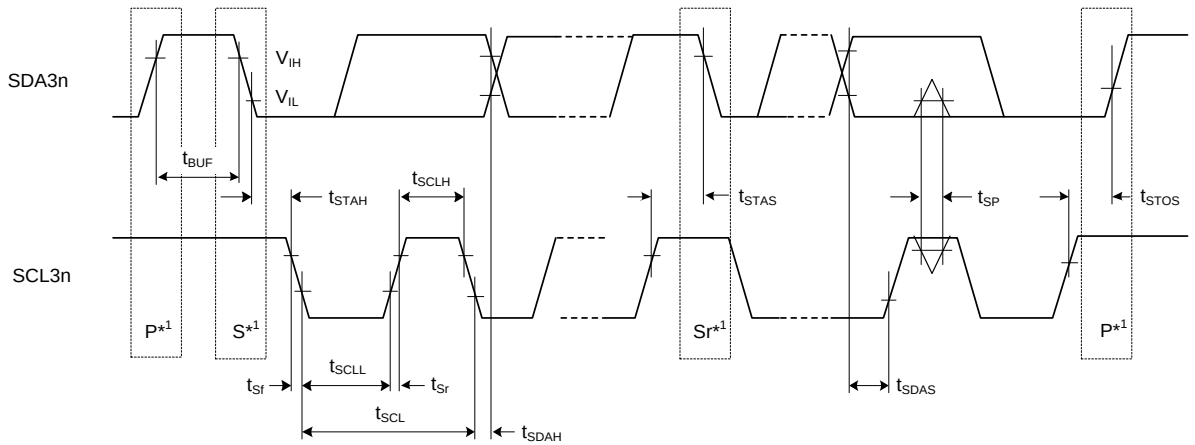
Note 1. t_{IICcyc}: I3C internal reference clock (I3Cϕ) cycle, t_{Pcyc}: I3C_0_TCLK cycle. Refer to **7.8.3.2.14 Reference Clock Control Register** for details. Values in parentheses apply when INCTL.DNFS[3:0] is set to 1111b while the digital filter is enabled with INCTL.DNFE set to 1b.

Table 10.1-24 IIC Timing (HS mode)

Parameter		Symbol	Cb = 100 pF		Cb = 400 pF		Unit	Figure
			Min.*1	Max.	Min.*1	Max.		
IIC (HS mode)	SCL3n cycle time	t_{SCL}	3(36) × t_{IICcyc} + 4 × t_{Pcyc} + 240	—	3(36) × t_{IICcyc} + 4 × t_{Pcyc} + 240	—	ns	**Figure 10.1-47**
	SCL3n high-level pulse width	t_{SCLH}	2(18) × t_{IICcyc} + 2 × t_{Pcyc} + 120	—	2(18) × t_{IICcyc} + 2 × t_{Pcyc} + 120	—	ns	
	SCL3n low-level pulse width	t_{SCLL}	2(18) × t_{IICcyc} + 2 × t_{Pcyc} + 120	—	2(18) × t_{IICcyc} + 2 × t_{Pcyc} + 120	—	ns	
	SCL3n rise time	t_{Sr}	—	40	—	80	ns	
	SCL3n rise time after a repeated START condition and after an acknowledge bit	t_{Sr}	—	80	—	160	ns	
	SDA3n rise time	t_{Sr}	—	80	—	160	ns	
	SCL3n fall time	t_{Sf}	—	40	—	80	ns	
	SDA3n fall time	t_{Sf}	—	80	—	160	ns	
	SCL3n, SDA3n spike pulse removal time	t_{SP}	0	1(16) × t_{IICcyc}	0	1(16) × t_{IICcyc}	ns	
	Hold time for START condition	t_{STAH}	t_{IICcyc} + 135	—	t_{IICcyc} + 135	—	ns	
	Setup time for repeated START condition	t_{STAS}	160	—	160	—	ns	
	Setup time for STOP condition	t_{STOS}	160	—	160	—	ns	
	Data setup time	t_{SDAS}	10	—	10	—	ns	
	Data hold time	t_{SDAH}	0	80	0	150	ns	
	SCL3n, SDA3n capacitive load	C_b	—	100	—	400	pF	

Note 1. t_{IICcyc}: I3C internal reference clock (I3Cϕ) cycle, t_{Pcyc}: I3C_0_TCLK cycle. Refer to **7.8.3.2.14 Reference Clock Control Register** for details. Values in parentheses apply when INCTL.DNFS[3:0] is set to 1111b while the digital filter is enabled with INCTL.DNFE set to 1b

Note 2. The maximum SCL clock frequency is 1.7 MHz.



Note: n = 0

Note 1. S, P, and Sr indicate the following conditions
S: Start condition
P: Stop condition
Sr: Restart condition

Figure 10.1-47 I3C Bus Interface Input/Output Timing

Table 10.1-25 I3C Timing (Open Drain Timing Parameters)

Parameter	Symbol	Min.*1	Max.	Unit	Figure	Notes
SCL3n clock Low period	t_{LOW_OD}	200	—	ns	**Figure 10.1-50**	1, 2
	$t_{DIG_OD_L}$	t_{LOW_ODmin} + t_{fDA_ODmin}	—	ns	**Figure 10.1-50**	—
SCL3n clock High period	t_{HIGH}	—	41	ns	**Figure 10.1-50**	3, 4
	t_{DIG_H}	36 (when 1.8 V) 40 (when 1.2 V)	t_{HIGH} + t_{CF}	ns	**Figure 10.1-50**	
SDA3n signal fall time	t_{fDA_OD}	t_{CF}	33	ns	**Figure 10.1-50**	—
SDA3n data setup time open drain mode V_{DD1218} = 1.8 V	t_{SU_OD}	12	—	ns	**Figure 10.1-49**, **Figure 10.1-50**	1
SDA3n data setup time open drain mode V_{DD1218} = 1.2 V		13.9	—	ns		
Clock after START (S) condition	t_{CAS}	38.4	For ENTAS0: 1 µ	seconds	**Figure 10.1-50**	5, 6
			For ENTAS1: 100 µ			
			For ENTAS2: 2 m			
			For ENTAS3: 50 m			
Clock before STOP (P) condition	t_{CBP}	t_{CASmin}/2	—	seconds	**Figure 10.1-51**	—
Current master to secondary master overlap time during handoff	$t_{MMOverlap}$	$t_{DIG_OD_Lmin}$	—	ns	**Figure 10.1-56**	—
Bus available condition	t_{AVAL}	1	—	µs	—	7
Bus idle condition	t_{IDLE}	1	—	ms	—	—
Time internal where new master not driving SDA3n low	t_{MMLock}	$t_{AVALmin}$	—	µs	**Figure 10.1-56**	—

Note 1. This is approximately equal to t_{LOWmin} + t_{DS_ODmin} + t_{rDA_ODtyp} + t_{SU_ODmin}.

Note 2. The Master may use a shorter Low period if it knows that this is safe, i.e., that SDA is already above VIH.

Note 3. This is based on t_{SPIKE}, rise and fall times, and interconnect.

Note 4. This maximum High period may be exceeded when the signals can be safely seen by Legacy I^2C Devices, and/or in consideration of the interconnect (e.g., a short bus).

Note 5. On a Legacy Bus where I^2C Devices need to see Start.

Note 6. Slaves that do not support the optional ENTASx CCCs shall use the t_{CAS} Max value shown for ENTAS3

Note 7. On a Mixed Bus with Fm Legacy I^2C Devices, t_{AVAL} is 300 ns shorter than the Fm Bus Free Condition time (t_{BUF})

Table 10.1-26 I3C Timing (Push-Pull Timing Parameters for SDR)

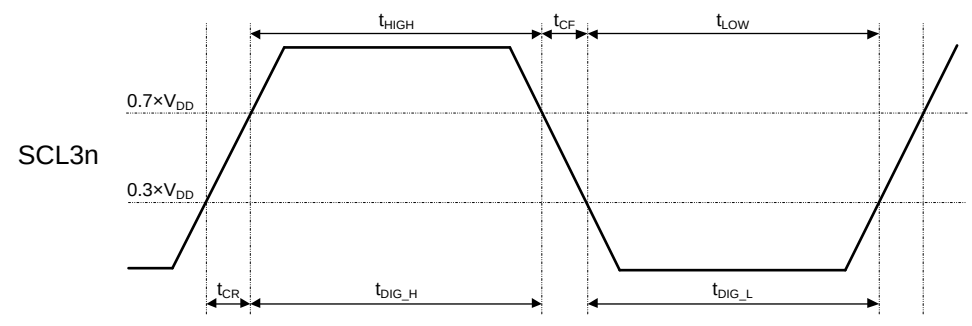
Parameter		Symbol	Min.*1	Max.	Unit	Figure	Notes
SCL3n clock frequency	V_{DD1218} = 1.8 V	t_{SCL}	0.01	12.5	MHz	—	1
	V_{DD1218} = 1.2 V		0.01	12.39	MHz	—	
SCL3n clock Low period		t_{LOW}	24	—	ns	**Figure 10.1-48**	—
		t_{DIG_L}	32	—	ns	**Figure 10.1-48**	2, 4
SCL3n clock High period for Mixed Bus		t_{HIGH}	24	—	ns	**Figure 10.1-48**	—
		t_{DIG_H}	32	45	ns	**Figure 10.1-48**	2, 3
SCL3n clock High period		t_{HIGH}	24	—	ns	**Figure 10.1-48**	—
		t_{DIG_H}	32	45	ns	**Figure 10.1-48**	2
Clock in to data out for a slave	V_{DD1218} = 1.8 V	t_{SCO}	—	12	ns	**Figure 10.1-53**	—
	V_{DD1218} = 1.2 V		—	12.7	ns		
SCL3n clock rise time		t_{CR}	—	150 × 1/f_{SCL} (capped at 60)	ns	**Figure 10.1-48**	—
SCL3n clock fall time		t_{CF}	—	150 × 1/f_{SCL} (capped at 60)	ns	**Figure 10.1-48**	—
SDA3n signal data hold in push-pull mode	Master	t_{HD_PP}	t_{CR} + 3 and t_{CF} + 3	—	—	**Figure 10.1-52**	4
	Slave		0	—	—	**Figure 10.1-54**	—
SDA3n signal data setup in push-pull mode	V_{DD1218} = 1.8 V	t_{SU_PP}	12	N/A	ns	**Figure 10.1-52**, **Figure 10.1-53**, **Figure 10.1-54**	—
	V_{DD1218} = 1.2 V		13.9	N/A	ns		
Clock after repeated START (Sr)		t_{CASr}	t_{CASmin}	N/A	ns	**Figure 10.1-55**	—
Clock before repeated START (Sr)		t_{CBSr}	t_{CASmin}/2	N/A	ns	**Figure 10.1-55**	—
Capacitive load per bus line (SDA3n / SCL3n)		C_b	—	50	pF	—	—

Note 1. FSCL = 1/(t_{DIG_L} + t_{DIG_H})

Note 2. t_{DIG_L} and t_{DIG_H} are the clock Low and High periods as seen at the receiver end of the I3C Bus using VIL and VIH (see **Figure 10.1-48**)

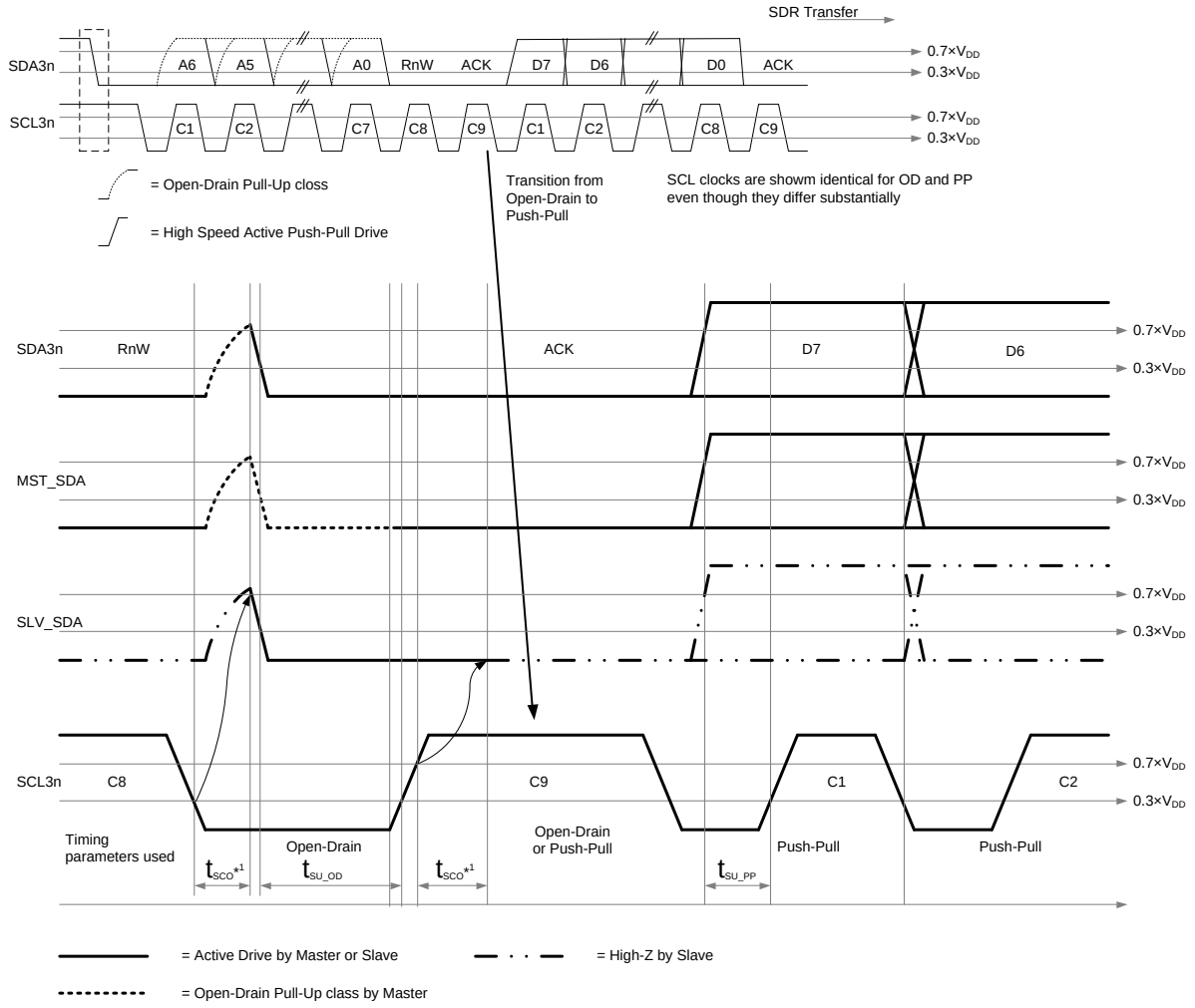
Note 3. When communicating with an I3C Device on a mixed bus, the $t_{DIG_H_MIXED}$ period must be constrained in order to make sure that I^2C Devices do not interpret I3C signaling as valid I^2C signaling.

Note 4. As both edges are used, the hold time must be satisfied for the respective edges, for example, t_{CF} + 3 for falling edge clocks, and t_{CR} + 3 for rising edge clocks.



Note: V_{DD}: VDD1218_I3C

Figure 10.1-48 t_{DIG_H} and t_{DIG_L}

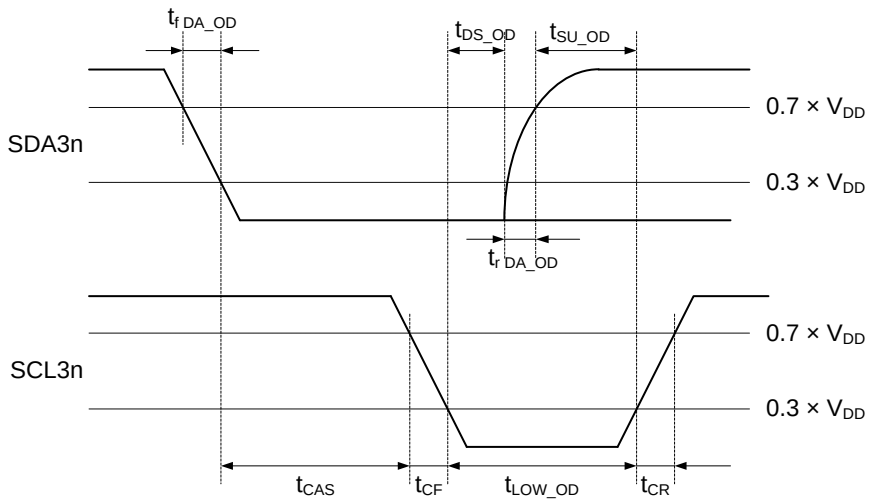


Note: V_{DD}: VDD1218_I3C

Note: Open-Drain: Master releases SDA after a suitable delay from SCL edges
Push-Pull: Master drives SDA after a suitable delay from SCL edges

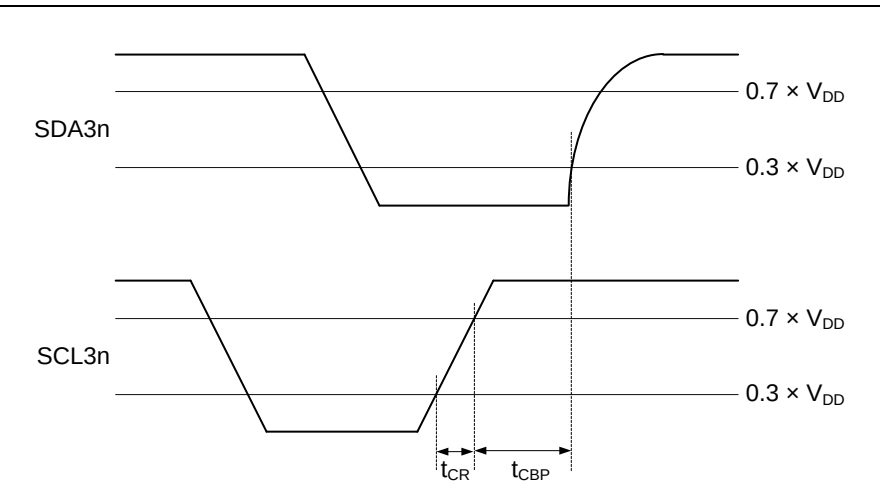
Note 1. t_{SCO} is depicted for informative purposes only

Figure 10.1-49 I3C Data Transfer – ACK by Slave



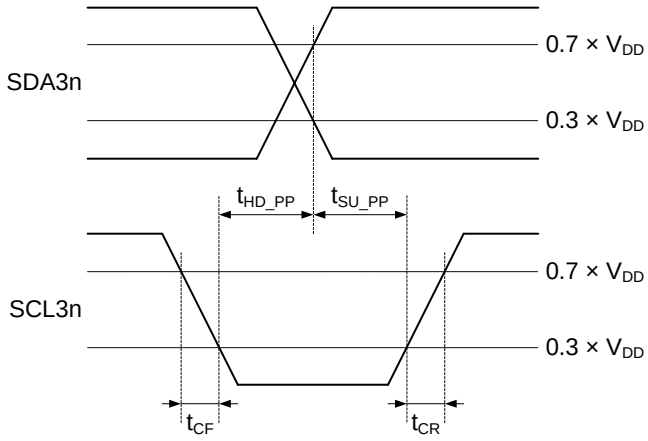
Note: V_{DD}: VDD1218_I3C

Figure 10.1-50 I3C START Condition Timing



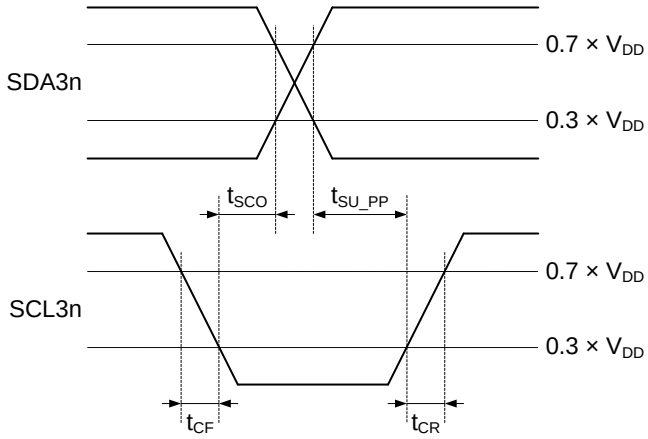
Note: V_{DD}: VDD1218_I3C

Figure 10.1-51 I3C STOP Condition Timing



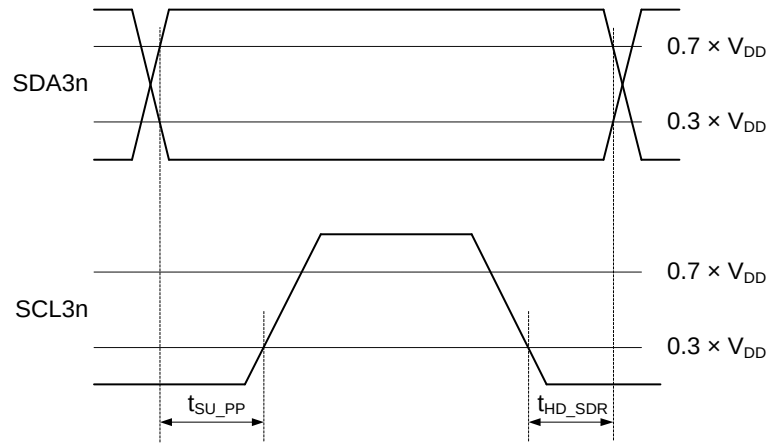
Note: V_{DD}: VDD1218_I3C

Figure 10.1-52 I3C Master Out Timing



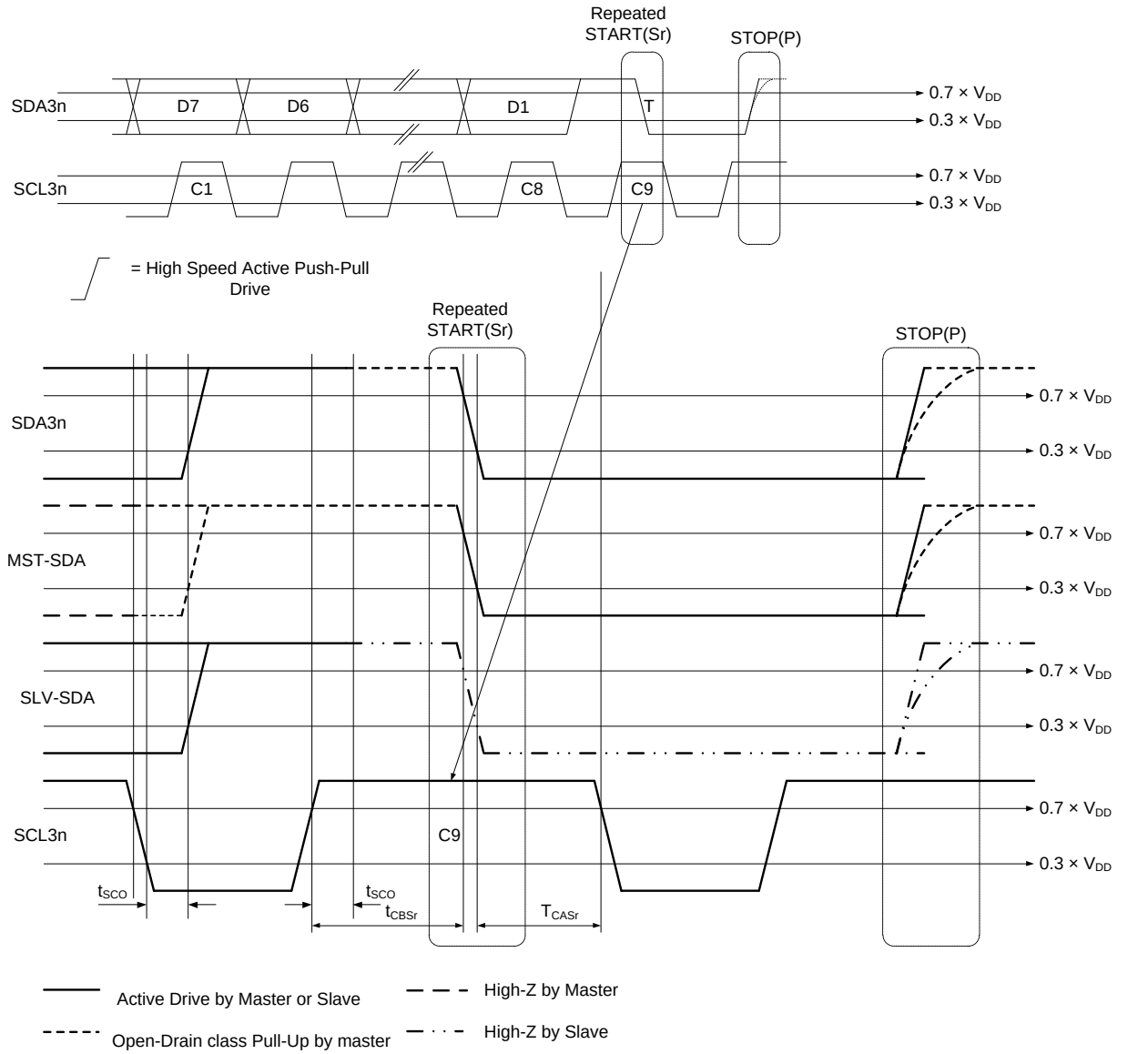
Note: V_{DD}: VDD1218_I3C

Figure 10.1-53 I3C Slave Out Timing



Note: V_{DD}: VDD1218_I3C

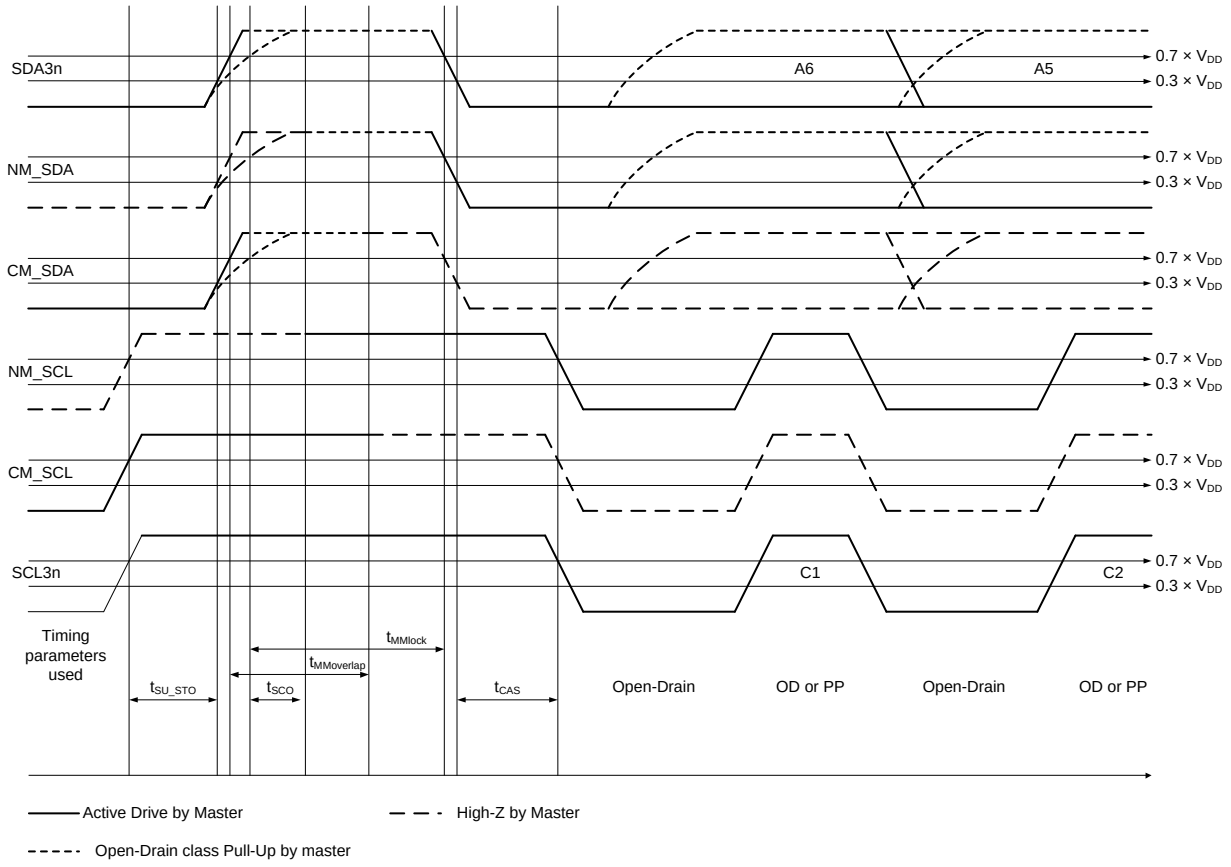
Figure 10.1-54 Master SDR Timing



Note: V_{DD}: VDD1218_I3C

Note: Sr immediately followed by P is illegal for I^2C Devices; however, many devices are not adversely affected by it.The condition is not illegal for I3C Devices.

Figure 10.1-55 T-Bit When Master Ends Read with Repeated START and STOP



Note: V_{DD}: VDD1218_I3C

Note: Master releases or drives SDA after a suitable delay from SCL edges

Figure 10.1-56 I3C Timing

10.1.5.18 CANFD Interface Access Timing

Table 10.1-27 CANFD Interface Timing

Parameter		Symbol	CAN		CANFD		Unit	Figures
			Min.	Max.	Min.	Max.		
CANFD	Internal delay time	t_{node}	—	100	—	50	ns	**Figure 10.1-57**
	Transmission rate	—	—	1	—	8	Mbps	

Note 1. Internal delay time (t_{node}) = Internal transmission delay time (t_{output}) + Internal reception delay time (t_{input})

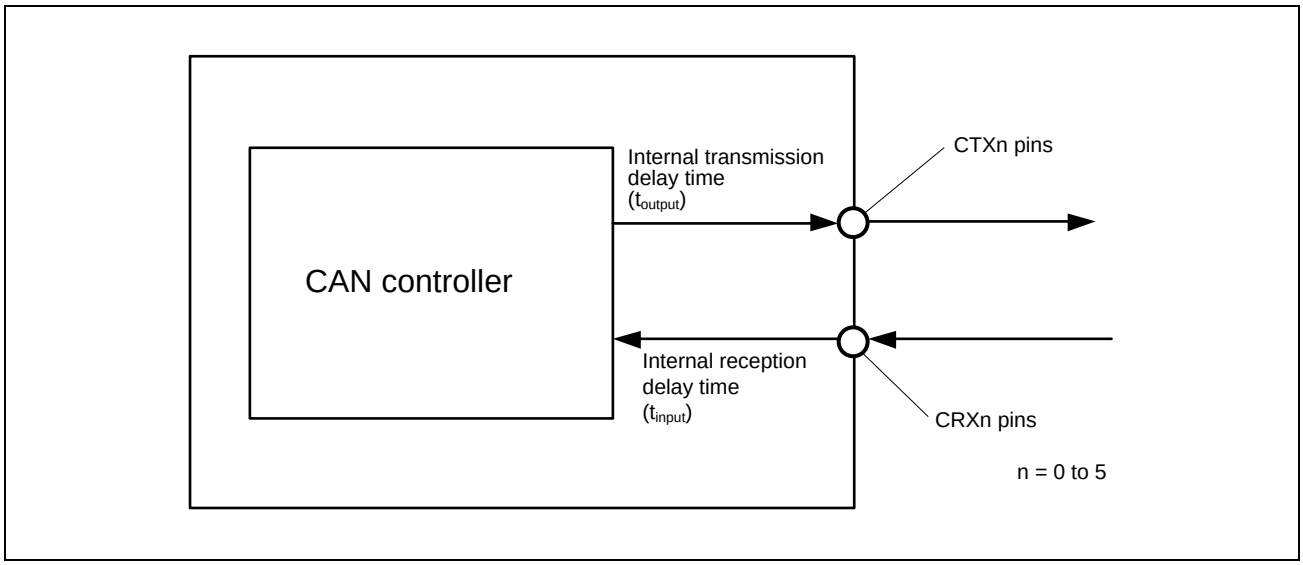


Figure 10.1-57 CANFD Interface Condition

10.1.5.19 A/D Converter Access Timing

Table 10.1-28 A/D Converter Trigger Timing

Parameter			Symbol	Min.	Max.	Unit*1	Figure
A/D converter	A/D converter trigger input pulse width	ADTRG	t_{TRGW}	1.5	—	$t_{PADCcyc}$	**Figure 10.1-58**

Note 1. $t_{PADCcyc}$: ADC_0_PCLK cycle

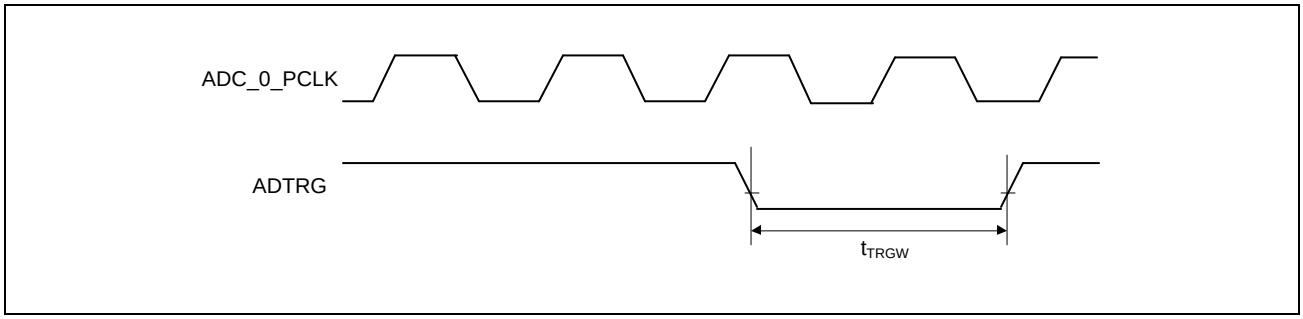


Figure 10.1-58 A/D Converter Trigger Input Timing (ADTRG)

10.1.5.20 SSIU Timing

Condition:

V_{OH} = VDD18 × 0.5, V_{OL} = VDD18 × 0.5, C = 30 pF*1 (1.8 V)
V_{OH} = VDD33 × 0.5, V_{OL} = VDD33 × 0.5, C = 30 pF*1 (3.3 V)

Drive strength: ×1, ×2, ×4, ×6

Note 1. Other than t_{RC}: Rise-edge clock timing

Table 10.1-29 SSIU Signal Timing

Parameter	Symbol	Min.	Max.	Unit	Note	Figure
Output clock cycle	t_O	80	15625	ns	—	**Figure 10.1-59**
Input clock cycle	t_I	80	15625	ns	—	
Output clock high-cycle	t_{HC}	35*1	—	ns	—	
Output clock low-cycle	t_{LC}	35*1	—	ns	—	
Input clock high-cycle	t_{HC}	35	—	ns	—	
Input clock low-cycle	t_{LC}	35	—	ns	—	
Rise-edge clock timing	t_{RC}	—	20*2	ns	Output (100 pF)	
Output delay	t_D	-5	19	ns	—	**Figure 10.1-60** to **Figure 10.1-63**
Setup time	t_S	15	—	ns	—	
Hold time	t_H	5	—	ns	—	

Note 1. The width at high or low level of the clock signal when the input on AUDIO_CLKA, AUDIO_CLKB, or AUDIO_CLKC is output from SCK without frequency division in master mode is min. 30 ns.
The relevant registers are as follows:
ADG: Audio Clock Select Register m (AUDIO_CLK_SELm (m = 0 to 2))
SSIU: Control Register (SSICRn (n = 0 to 9))

Note 2. Output transition time from 20% to 80%

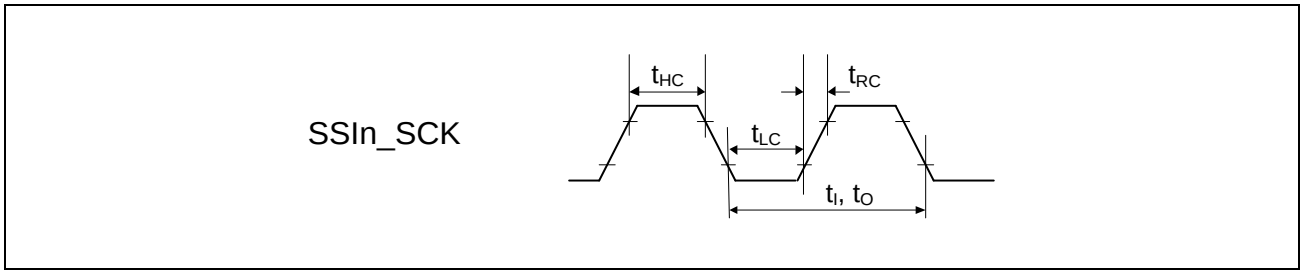


Figure 10.1-59 SCK Clock Input/Output Timing

SSICR*.SCKD = Don’t care, SSICR*.SWSD=1, SSICR*.SCKP=1, SSICR*.SWSP = Don’t care

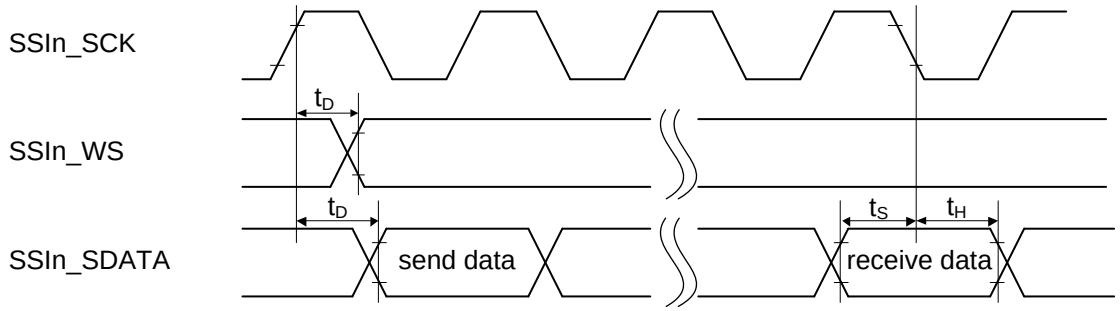


Figure 10.1-60 SSI Timing (1)

SSICR*.SCKD = Don’t care, SSICR*.SWSD=1, SSICR*.SCKP=0, SSICR*.SWSP = Don’t care

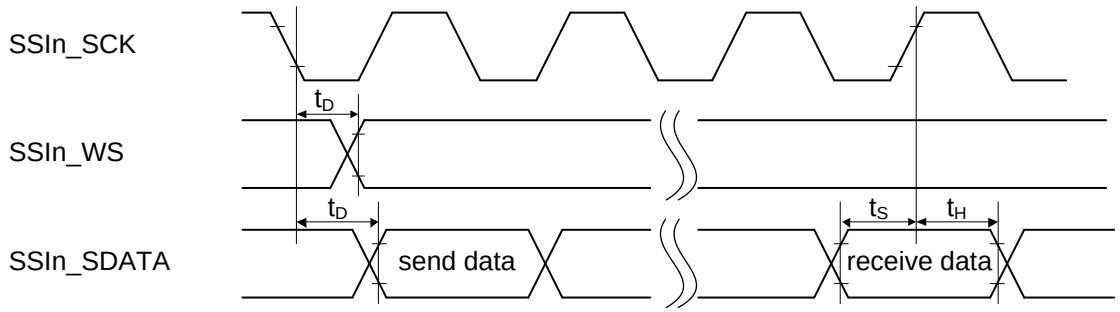


Figure 10.1-61 SSI Timing (2)

SSICR*.SCKD = Don’t care, SSICR*.SWSD=0, SSICR*.SCKP=1, SSICR*.SWSP = Don’t care

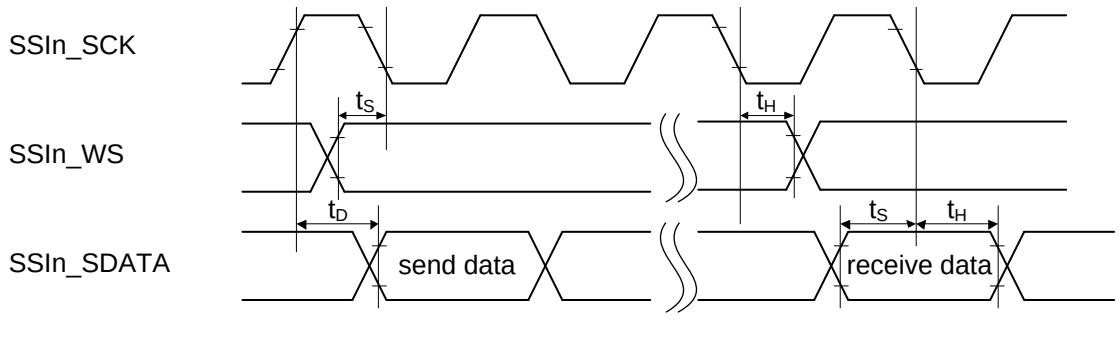


Figure 10.1-62 SSI Timing (3)

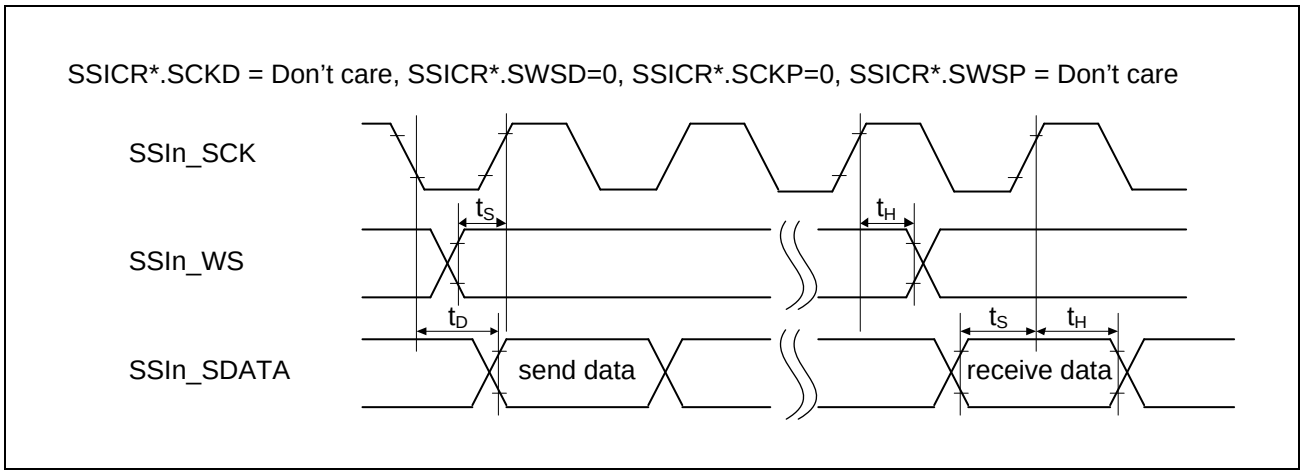


Figure 10.1-63 SSI Timing (4)

10.1.5.21 PDM Timing

Conditions:

V_{OH} = VDD18 × 0.5, V_{OL} = VDD18 × 0.5, C = 30 pF (1.8 V)

V_{OH} = VDD33 × 0.5, V_{OL} = VDD33 × 0.5, C = 30 pF (3.3 V)

Drive strength: ×1, ×2, ×4, ×6

Table 10.1-30 PDM Interface Timing

Parameter	Symbol	Min.	Max.	Unit	Figure
Clock period	t_{PSYNC}	2	32	t_{CCcyc} = 208.33 ns (4.8 MHz)*1	**Figure 10.1-64**
Clock high-level period	t_{PDCKWH}	t_{PSYNC} × 0.45	t_{PSYNC} × 0.55	ns	
Clock low-level period	t_{PDCKWL}	t_{PSYNC} × 0.45	t_{PSYNC} × 0.55	ns	
Clock rise time	t_{R-EDGE}	—	3*2	ns	
Clock fall time	t_{F-EDGE}	—	3*2	ns	
Setup time	t_{su}	15	—	ns	**Figure 10.1-65**, **Figure 10.1-66**
Hold time	t_h	0	—	ns	

Note 1. t_{CCcyc} is the period of PDM_n_CCLK (n = 0, 1).

Note 2. Output transition time from 20% to 80%

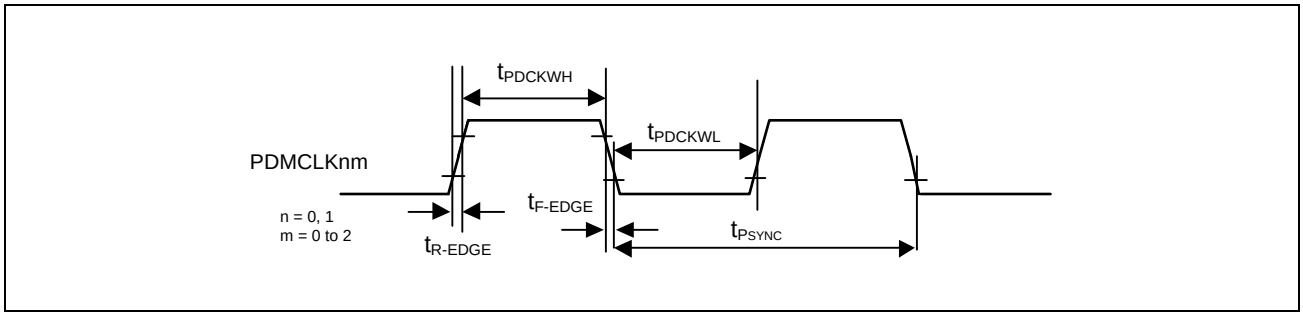


Figure 10.1-64 Timing of Clock Output (PDMCLKnm)

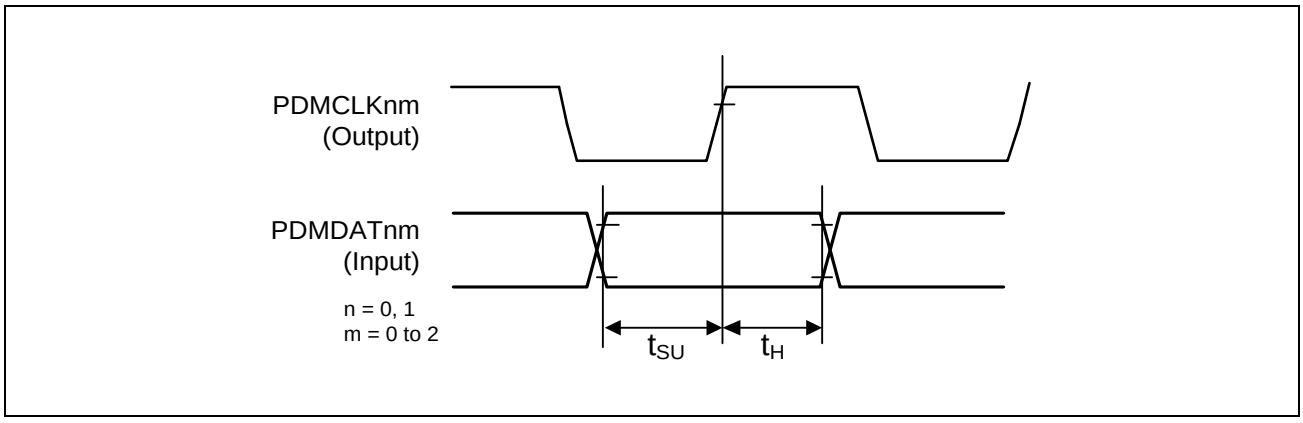


Figure 10.1-65 Timing of Clock Output (Synchronized with the rise of PDMCLKnm)

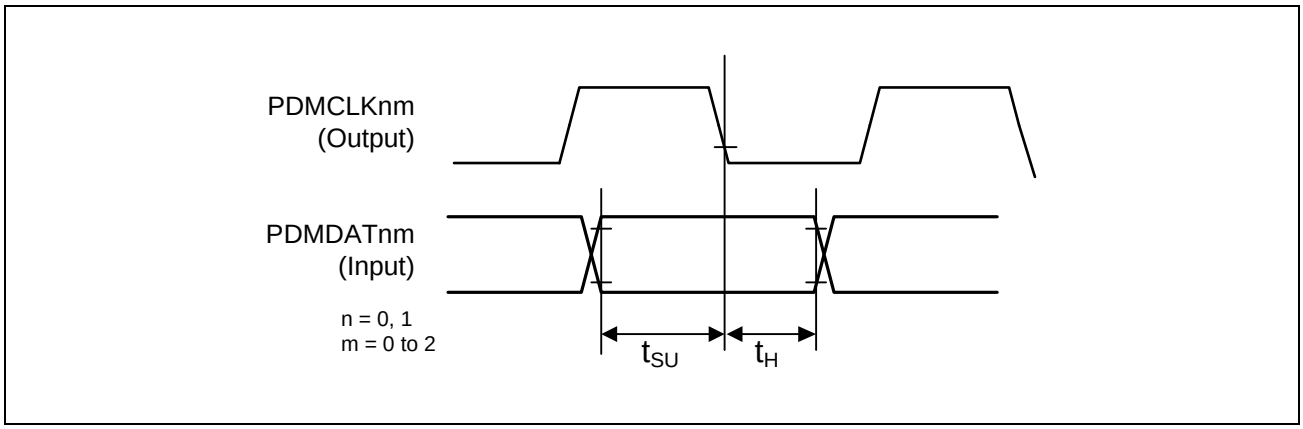


Figure 10.1-66 Timing of Clock Output (Synchronized with the fall of PDMCLKnm)

10.1.5.22 MIPI CSI-2 PHY Characteristics

The MIPI CSI-2 Rx D-PHY of this LSI is equivalent to the MIPI D-PHY Version 1.2.

For details, refer to the MIPI specification.

10.1.5.23 MIPI DSI Tx D-PHY Characteristics

The MIPI DSI Tx D-PHY of this LSI is compliant with the MIPI D-PHY Version 1.2.

For details, refer to the MIPI specification.

10.1.5.24 Control Signal Access Timing

Table 10.1-31 Control Signal Timing

Item	Symbol	Min.	Max.	Unit	Figures
QRESN pulse width	t_{RESW}	1	—	µs	**Figure 10.1-67**
TRSTN pulse width	t_{TRSW}	1	—	µs	
NMI pulse width	t_{NMIW}	20	—	t_{cyc}*1	**Figure 10.1-68**
IRQ pulse width	t_{IRQW}	20	—	t_{cyc}*1	
TINT pulse width	t_{TINTW}	20	—	t_{cyc}*1	

Note 1. t_{cyc} = 41.666 ns (24 MHz)

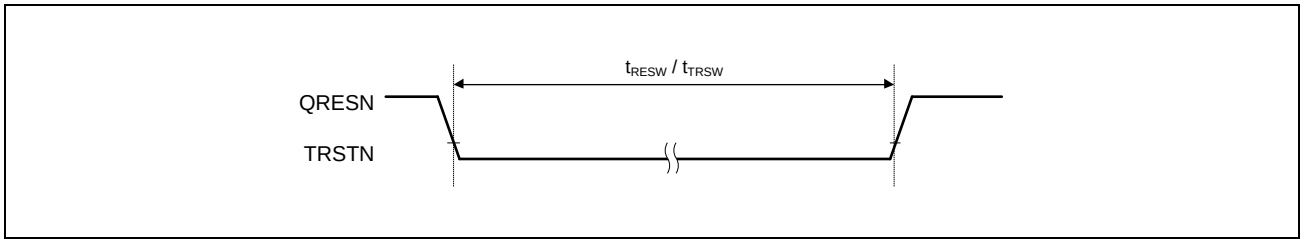
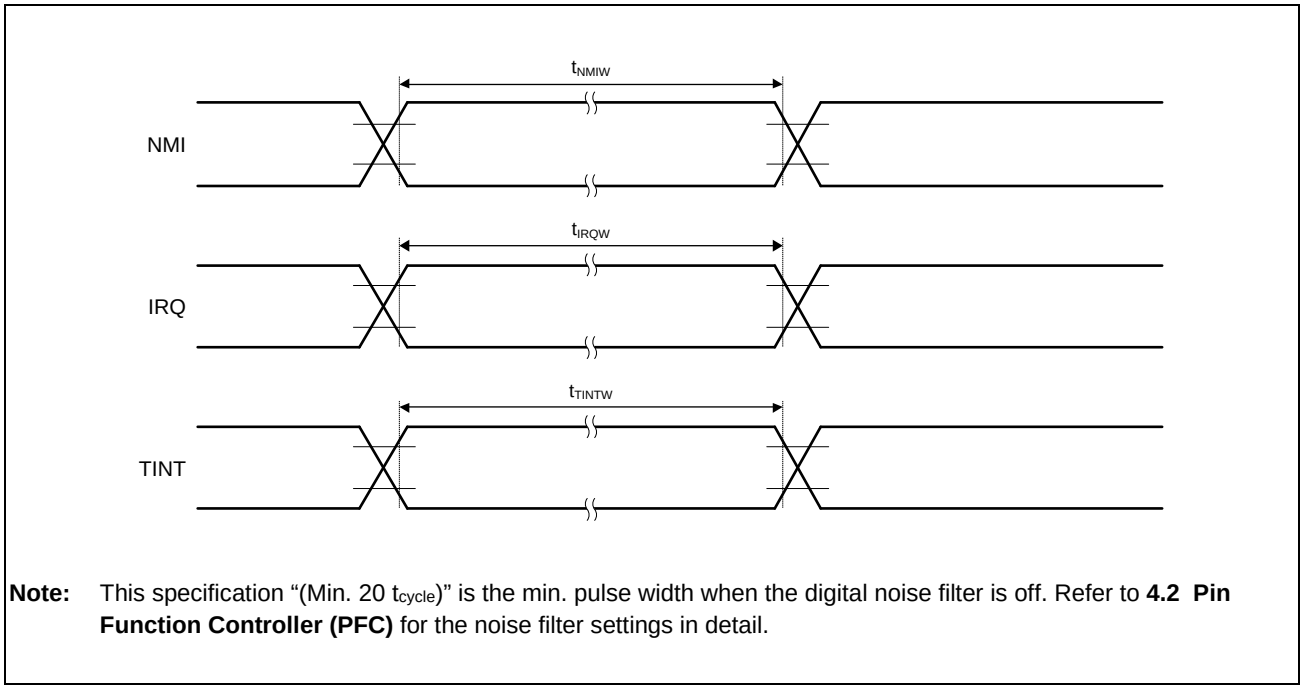


Figure 10.1-67 Reset Input Timing



Note: This specification “(Min. 20 t_{cycle})” is the min. pulse width when the digital noise filter is off. Refer to **4.2 Pin Function Controller (PFC)** for the noise filter settings in detail.

Figure 10.1-68 Interrupt Signal Input Timing

10.1.5.25 JTAG Debugger Interface Access Timing

Table 10.1-32 Debugger IF Timing

Item	Symbol	Min.	Max.	Unit	Figures
TCK_SWCLK cycle time	t_{TCKcyc}	50	—	ns	**Figure 10.1-69**
TCK_SWCLK high-level pulse width	t_{TCKH}	20	—	ns	**Figure 10.1-70**
TCK_SWCLK low-level pulse width	t_{TCKL}	20	—	ns	
TDI setup time	t_{TDIS}	15	—	ns	
TDI hold time	t_{TDIH}	15	—	ns	
TMS_SWDIO setup time	t_{TMSS}	15	—	ns	
TMS_SWDIO hold time	t_{TMSH}	15	—	ns	
TMS_SWDIO delay time	t_{SWDO}	—	14	ns	
TDO delay time	t_{TDOD}	—	14	ns	
Capture register setup time	t_{CAPTS}	10	—	ns	**Figure 10.1-71**
Capture register hold time	t_{CAPTH}	10	—	ns	
Update register delay time	$t_{UPDATED}$	—	20	ns	

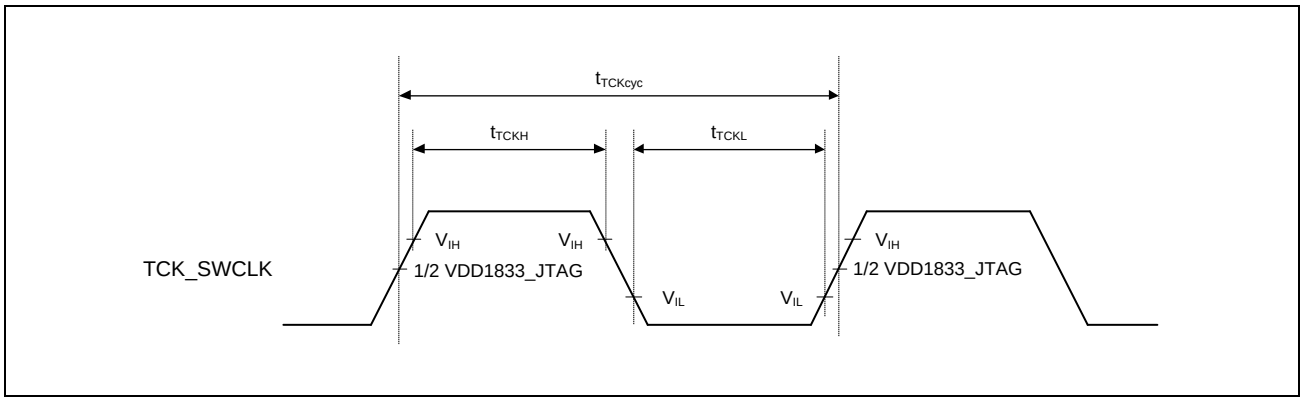


Figure 10.1-69 TCK_SWCLK Input Timing

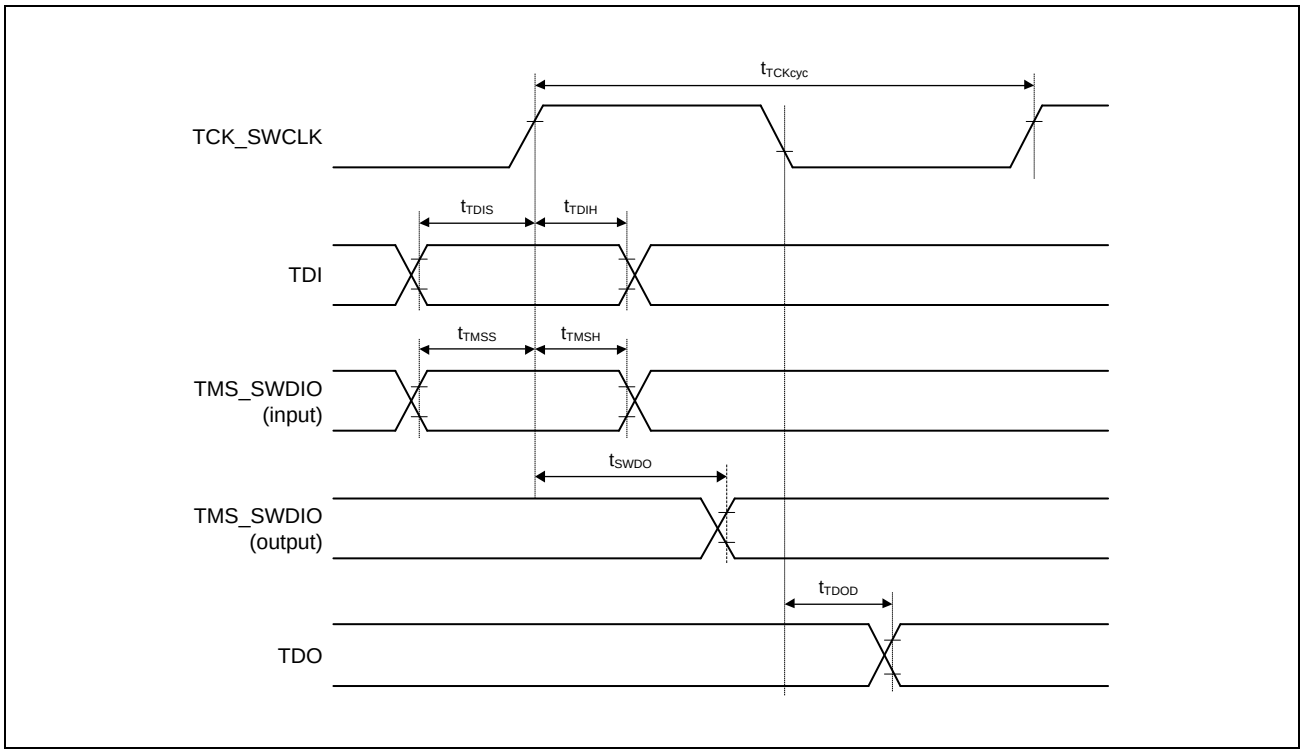


Figure 10.1-70 Data Transfer Timing

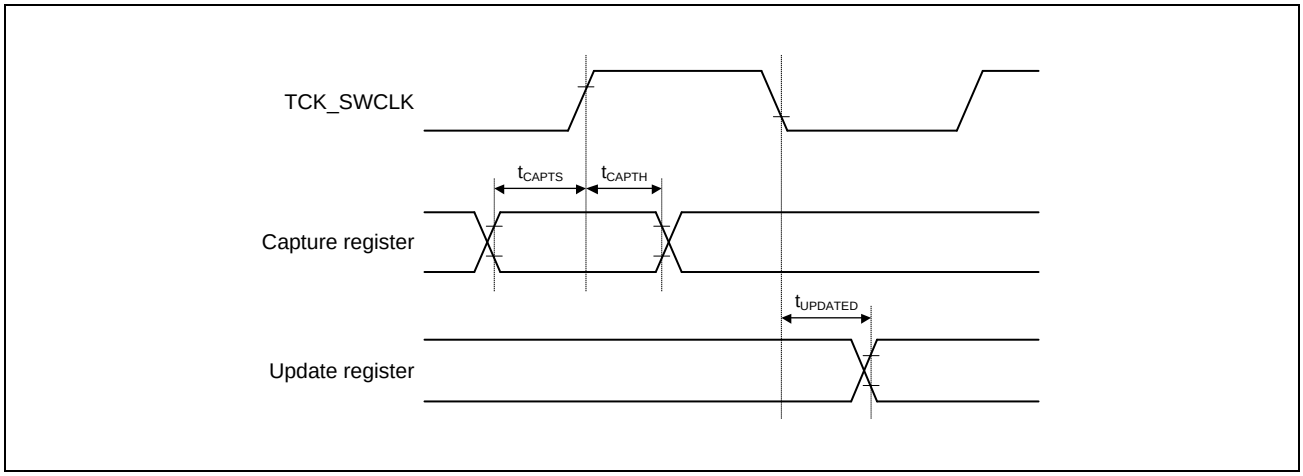


Figure 10.1-71 Boundary Scan Input/Output I/O Timing

10.1.5.26 LCDC

Conditions:

V_{OH} = VDD18 × 0.7, V_{OL} = VDD18 × 0.3, C = 30 pF

V_{OH} = VDD33 × 0.7, V_{OL} = VDD33 × 0.3, C = 30 pF

Drive strength: ×6

Table 10.1-33 LCDC IF Timing

Parameter	Symbol	Min.	Typ.	Max.	Unit
LCD_CLK period	t_{Lcyc}	11.49	—	185.19	ns
LCD_CLK low pulse width	t_{LOL}	$t_{Lcyc}/2 - 1.06$	—	$t_{Lcyc}/2 + 1.06$	ns
LCD_CLK high pulse width	t_{LOH}	$t_{Lcyc}/2 - 1.06$	—	$t_{Lcyc}/2 + 1.06$	ns
LCD_CLK rise time	t_{LOR}	—	—	3	ns
LCD_CLK fall time	t_{LOF}	—	—	3	ns
Data output dealy	t_{DD}	−1.5	—	1.5	ns

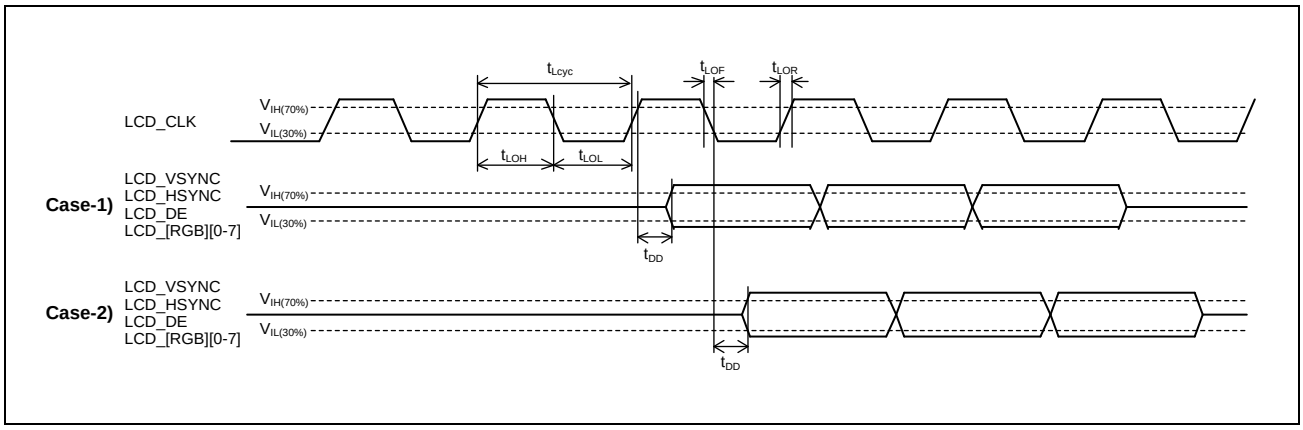


Figure 10.1-72 Boundary Scan Input/Output I/O Timing

10.1.5.27 LVDS Interface

Table 10.1-34 LVDS IF Timing

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
Output Clock Frequency of LVDS	$F_{CLKP/CLKN}$	25	—	87	MHz	**Figure 10.1-73**
Output Clock of LVDS	T_{CP}	11.494	—	40	ns	
Output Data Rate of LVDS	$F_{DAP/DAN}$	175	—	609	Mbps	
LVDS Output Pulse Position 0	T_{PPOS0}	−0.200	0	0.200	ns	
LVDS Output Pulse Position 1	T_{PPOS1}	$(T_{CP}/7) - 0.200$	$(T_{CP}/7)$	$(T_{CP}/7) + 0.200$	ns	
LVDS Output Pulse Position 2	T_{PPOS2}	$2(T_{CP}/7) - 0.200$	$2(T_{CP}/7)$	$2(T_{CP}/7) + 0.200$	ns	
LVDS Output Pulse Position 3	T_{PPOS3}	$3(T_{CP}/7) - 0.200$	$3(T_{CP}/7)$	$3(T_{CP}/7) + 0.200$	ns	
LVDS Output Pulse Position 4	T_{PPOS4}	$4(T_{CP}/7) - 0.200$	$4(T_{CP}/7)$	$4(T_{CP}/7) + 0.200$	ns	
LVDS Output Pulse Position 5	T_{PPOS5}	$5(T_{CP}/7) - 0.200$	$5(T_{CP}/7)$	$5(T_{CP}/7) + 0.200$	ns	
LVDS Output Pulse Position 6	T_{PPOS6}	$6(T_{CP}/7) - 0.200$	$6(T_{CP}/7)$	$6(T_{CP}/7) + 0.200$	ns	

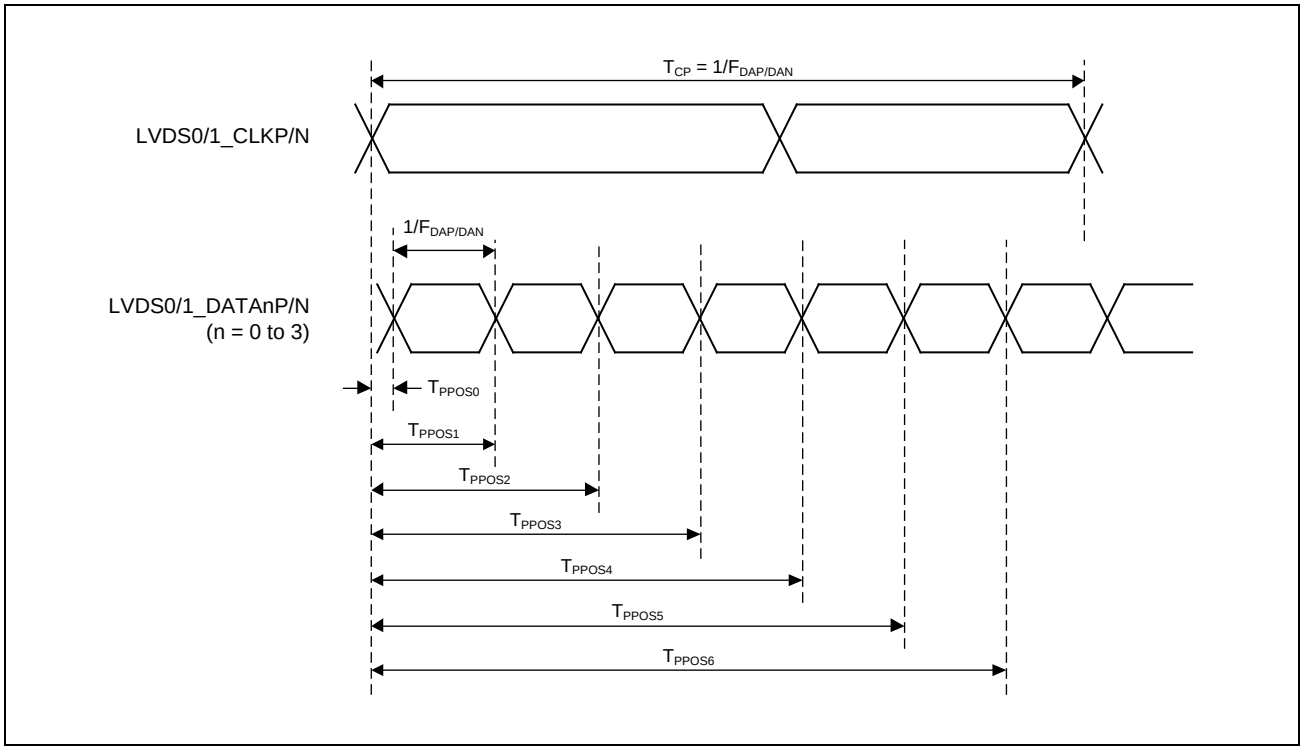


Figure 10.1-73 T_{CP}, $F_{DAP/DAN}$, T_{PPOS}

10.1.6 Analog Characteristics

10.1.6.1 ADC Characteristics

Table 10.1-35 DC Characteristics

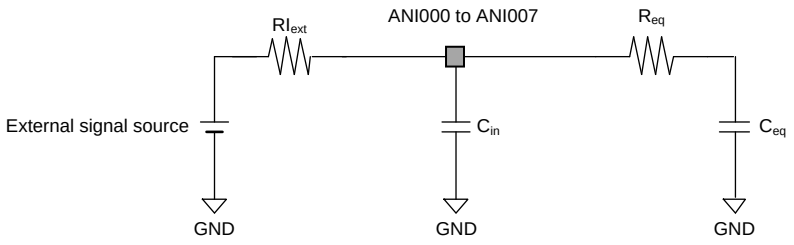
Item	Min.	Typ.	Max.	Unit
Resolution	—	12	—	Bit
Analog input capacitance	—	—	13	pF
Analog input range	0	—	ADAVDD18	V
Conversion time*1 Permissible signal source impedance Max. = 1.0 kΩ	0.4	—	4.0	µs
Offset error	0	—	100	LSB
Full-scale error	−100	—	0	LSB
Quantization error	—	±0.5	—	LSB
DNL differential non-linearity error	—	±1.0	±3.0	LSB
INL integral nonlinearity error	—	±2.0	±6.0	LSB

Note 1. The conversion time is the total of the sampling time and the comparison time.

Table 10.1-36 Recommended External Input Resistance

Item	Symbol	Min.	Typ.	Max.	Unit
External input resistance*1 (ANI000-ANI007)	RI_{ext}	—	—	1	kΩ

Note 1. Output resistance of signal generator + Series parasitic resistance between signal source and ADC input.



Note: Analog input equivalent resistance (R_{eq}): 500Ω
Analog input equivalent capacitance (C_{eq}): 10 pF
Analog input pin capacitance (C_{in}): 3 pF

Figure 10.1-74 A/D Converter Equivalent Circuit and Peripheral Configuration Diagram

10.1.6.2 Temperature Sensor Characteristics

Table 10.1-37 Temperature Sensor Characteristics

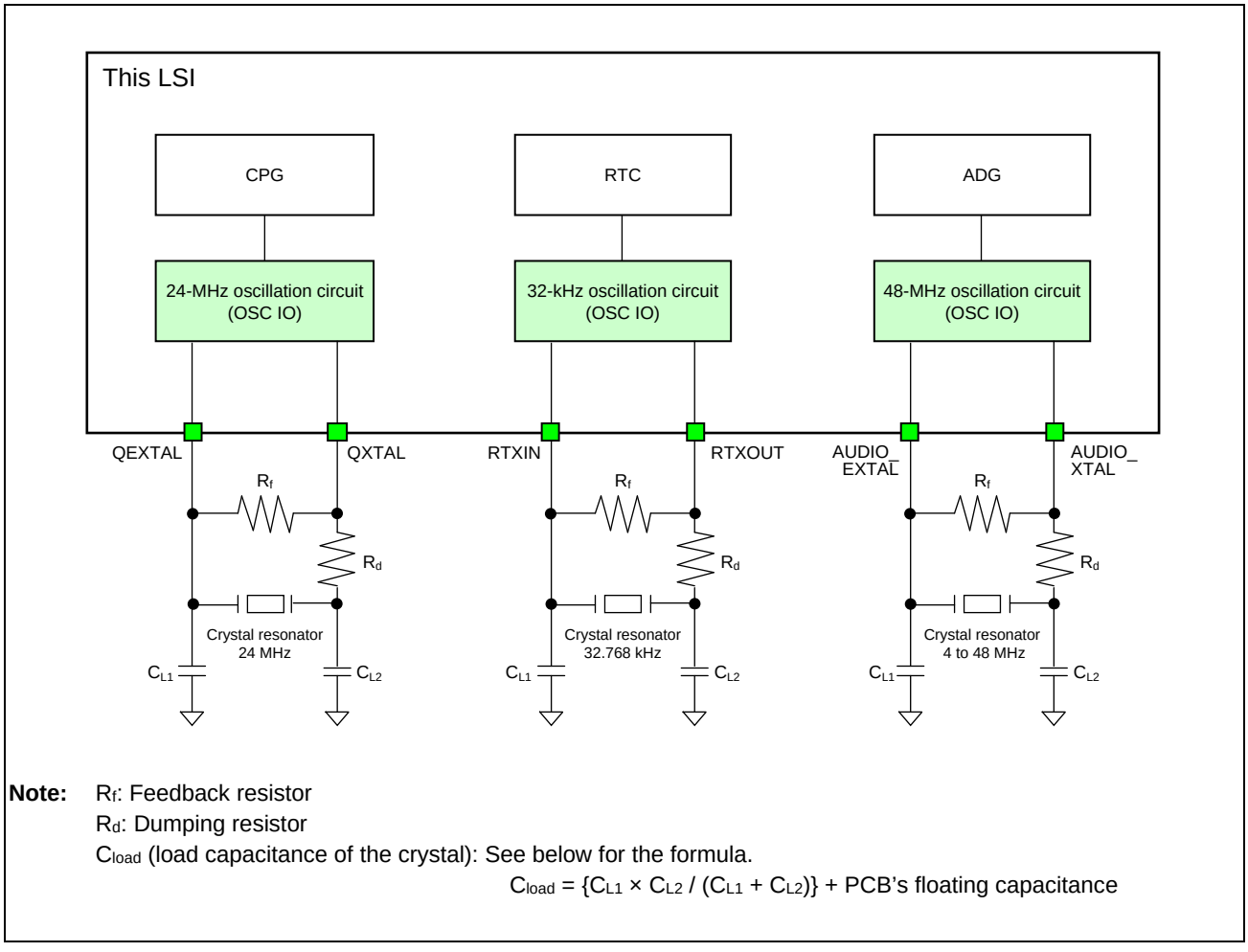
Item	Symbol	Min.	Typ.	Max.	Unit	Test Condition
Accuracy from −40°C to 125°C	Accm40_125	—	±3.0	±5.0	°C	

10.1.7 Oscillation Circuits for Connecting Crystal Resonators (OSC)

This LSI chip includes three oscillation circuits (OSC) for connection to crystal resonators, specifically a 24-MHz crystal resonator for the system clock, a frequency range between 4 and 48-MHz crystal resonator for the audio clocks and a 32.768-kHz crystal resonator for the real-time clock. **Table 10.1-38** lists the pins for connecting the crystal resonators and the clock frequencies. **Figure 10.1-75** shows an example of the connections with crystal resonators.

Table 10.1-38 Pins for Connecting Crystal Resonators and Clock Frequency

External Pin Name	I/O	Clock Frequency
For the system clock		
QEXTAL	Input	24 MHz (frequency deviation: ±50 ppm)
QXTAL	Output	24 MHz
For the real-time clock		
RTXIN	Input	32.768 kHz (frequency deviation: ±50 ppm)
RTXOUT	Output	32.768 kHz
For the audio clocks		
AUDIO_EXTAL	Input	4 to 48 MHz
AUDIO_XTAL	Output	4 to 48 MHz



Note: R_f: Feedback resistor
R_d: Dumping resistor
C_{load} (load capacitance of the crystal): See below for the formula.

$$C_{load} = \{C_{L1} \times C_{L2} / (C_{L1} + C_{L2})\} + \text{PCB's floating capacitance}$$

Figure 10.1-75 Example of Connections with Crystal Resonators

Place the crystal resonators and the capacitors C_{L1} and C_{L2} as close as possible to the pins to connect crystal resonators. To avoid interference and to ensure correct oscillation, the grounding points of the capacitors appended to the crystal resonators should be shared, and no wiring patterns should be placed near these components.

The characteristics of the crystal resonators are closely related to the design of the user board. Therefore, the user should sufficiently evaluate them with reference to the example of connection of crystal resonators in **Figure 10.1-75**.

The circuit rating of a crystal resonator depends on the crystal resonator and the stray capacitance of the mounting circuit. Therefore, contact the manufacturer of the crystal resonator before deciding upon the circuit rating. The user should thoroughly evaluate and then set the parameters (resistor and capacitor values).

Table 10.1-39 is a list of recommended values for the crystal resonators.

Table 10.1-39 Recommended Model Values for the Crystal Resonators

	Model Values for the Crystal Resonators			
Clock Frequency	Max. ESR*1	Max. C_L*2	Max. C_0*3	Max. Drive Level
32.768 kHz	70 kΩ	12.5 pF	1.4 pF	1 µW
24 MHz	60 Ω	12 pF	7 pF	100 µW
48 MHz	50 Ω	10 pF	7 pF	100 µW

Note 1. ESR means the equivalent series resistor of the crystal resonator.

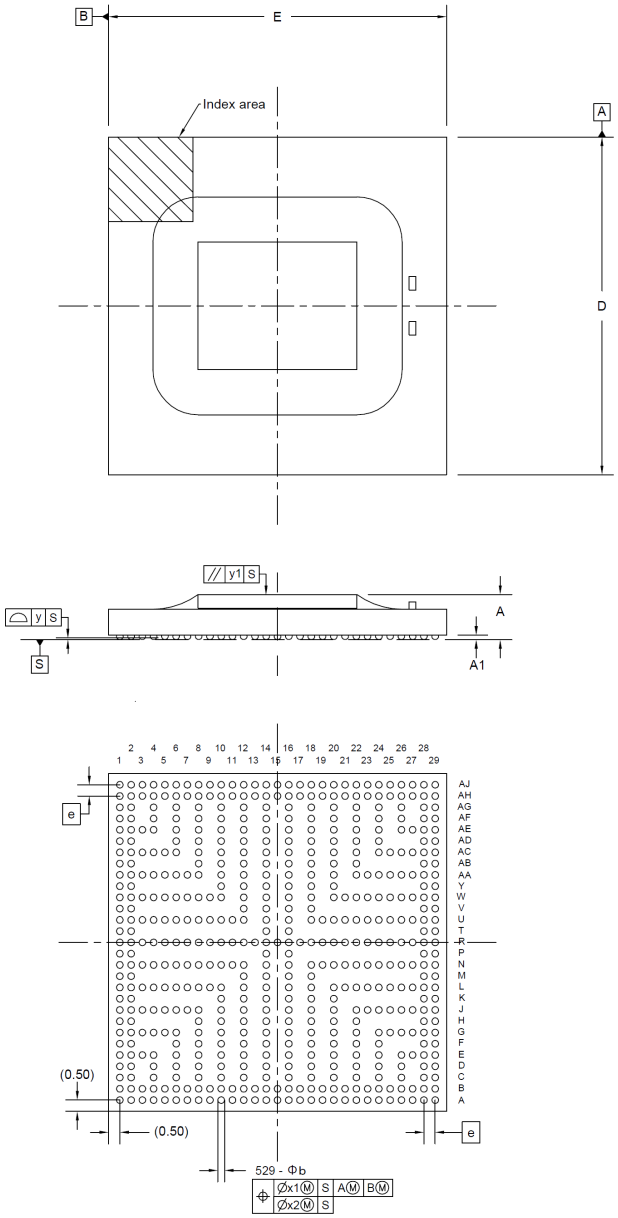
Note 2. C_L is the load capacitance of the crystal resonator.

Note 3. C_0 is the parallel capacitance of the crystal resonator.

Appendix A PACKAGE DIMENSIONS

A.1 Package Dimensions

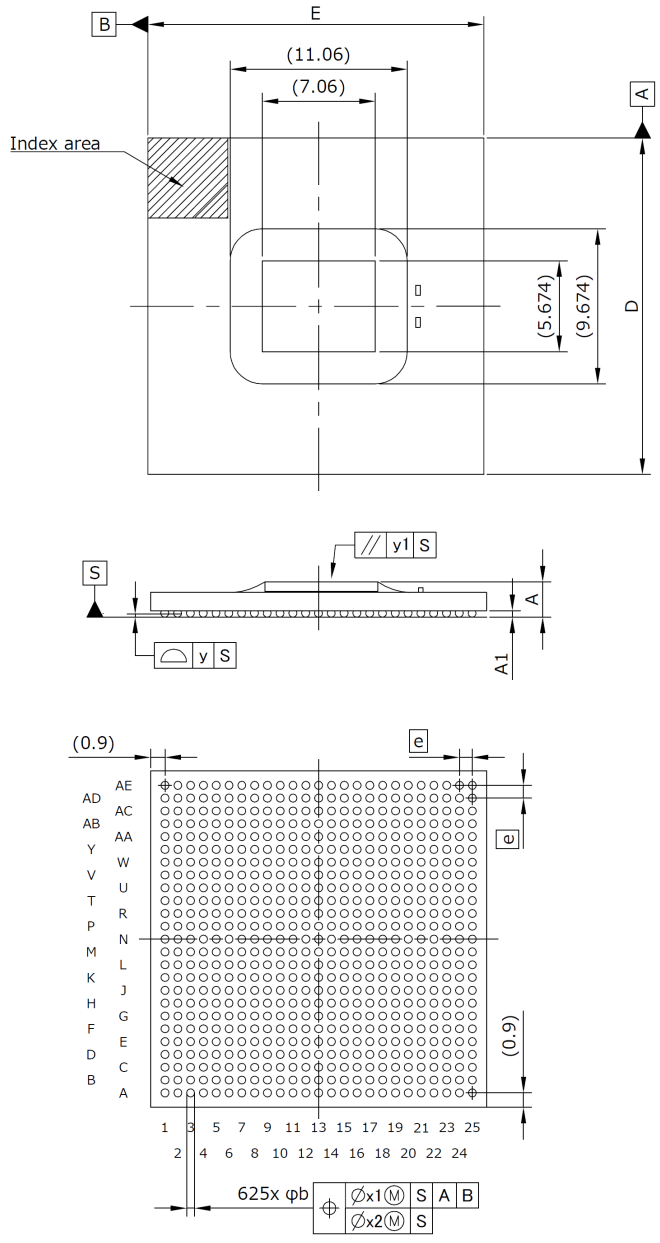
JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-FBGA529-15x15-0.50	PRBG0529KA-A	0.82



Reference Symbol	Dimension in Millimeters		
	MIN.	NOM.	MAX.
D	14.85	15.00	15.15
E	14.85	15.00	15.15
e	—	0.50	—
A	(1.90)	(2.10)	2.30
A1	0.15	(0.25)	—
b	0.25	0.30	0.35
x1	—	—	0.20
x2	—	—	0.05
y	—	—	0.12
y1	—	—	0.20

Figure A.1-1 Package Dimensions (529-pin FCBGA, 15 mm□ BGA 0.5 mm pin-pitch)

JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-FBGA625-21x21-0.80	PRBG0625GA-A	1.7



Reference Symbol	Dimension in Millimeters		
	MIN.	NOM.	MAX.
D	20.85	21.00	21.15
E	20.85	21.00	21.15
A	—	—	2.45
A1	0.35	0.40	0.45
e	—	0.80	—
b	0.45	0.50	0.55
x1	—	—	0.20
x2	—	—	0.08
y	—	—	0.15
y1	—	—	0.20

Figure A.1-2 Package Dimensions (625-pin FCBGA, 21 mm□ BGA 0.8 mm pin-pitch)

REVISION HISTORY	RZ/G3E Group User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
1.00	Oct 31, 2024	—	First edition issued
1.10	Jun 30, 2025	SECTION 1 OVERVIEW	
		1.1 Overview	
		75	1.1.1 Features: NOTE added
		76	Table 1.1-1 Product Lineup: Note 1 and Note, added
		78	1.1.3.5 Video Processing Unit: The functional description (I-/P-slice supported, I-/P-/B-slice supported) was added to the Video Codec Unit (VCD)
		79	1.1.3.7 External Memory Interface: The functional description of External Bus Controller for LPDDR4/4X SDRAM (DDR), modified (16 ECC regions, deleted)
		80	1.1.3.8 CPU Peripheral: The external Interrupt pins of the Interrupt Controller, modified
		89	1.1.3.21 Package, modified
		90	Figure 1.1-2 Block Diagram: An arrow was added from the ACPU Bus to the NPU Bus
		91	Table 1.1-2 List of Units (1/2): GPV Unit, deleted
		1.2 Pin	
		93	1.2.2 External Pins and Multiplexed Functional Pins: Text, modified (External Pins → External Pins and Multiplexed Functional Pins)
		—	1.2.2.1 List of External Pins, deleted
		—	1.2.2.2 List of Multiplexed Functional Pins, deleted
		1.3 Clocks	
		97	Table 1.3-3 List of Pin Functions (2/2): The frequency of the RSPCKn pin, modified (100 MHz → 50 MHz)
		1.7 Internal Bus	
		110 to 112	Table 1.7-1 Accessible Areas (1/3) to Table 1.7-1 Accessible Areas (3/3) CRU (Video) and CRU (Statistics) were deleted, and CRU0 (Video, Statistics) was added in the "Master Unit" column
		1.8 Address Map	
		123	Table 1.8-1 Detailed Address Space (2/6): The "Space" column of 0_1600_0000h - 0_1600_FFFFh, modified (CRU → CRU0)
		130	Table 1.8-2 Detailed Address Space of CM33 (2/13): The "Space" column of 5600_0000h - 5600_FFFFh, modified (CRU (Non-Secure) → CRU0 (Non-Secure))
		131	Table 1.8-2 Detailed Address Space of CM33 (8/13): The "Space" column of 4600_0000h - 4600_FFFFh, modified (CRU (Secure) → CRU0 (Secure))
		SECTION 2 PROCESSORS	
		2.2 CPU	
		186	2.2.3 System CPU Cortex-M33 (CM33): The reference in the main text, modified (1.8.2 CM33 Address Space)
		SECTION 4 SYSTEM	
		4.1 System Overview	
		221	Trusted Secure IP → Renesas Secure IP (RSIP-E50B)
		4.2 Pin Function Controller (PFC)	
		222	4.2.1 Overview, modified ((P00 to PB4) → (P00 to PS3))
		247	Table 4.2-13 Drive Strength Setting Values on Respective Pins and Values after Reset (1/4) Table heading: IOLH_mn bits = 00b/01b/10b/11b → 00b/10b/01b/11b WDTUDF_CA → WDTUDFCA, WDTUDF_CM → WDTUDFCM The Value after Reset of WDTUDFCA, WDTUDFCM, SCIF_RXD, SCIF_TXD, and PS0 to PJ4, modified (×2 → ×4)

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	248	Table 4.2-13 Drive Strength Setting Values on Respective Pins and Values after Reset (2/4) Table heading: IOLH_mn bits = 00b/01b/10b/11b → 00b/10b/01b/11b The Value after Reset of PG0 to PD6, modified (×2 → ×4)
		249	Table 4.2-13 Drive Strength Setting Values on Respective Pins and Values after Reset (3/4) Table heading: IOLH_mn bits = 00b/01b/10b/11b → 00b/10b/01b/11b The Value after Reset of PD7 to P60, modified (×2 → ×4)
		250	Table 4.2-13 Drive Strength Setting Values on Respective Pins and Values after Reset (4/4) Table heading: IOLH_mn bits = 00b/01b/10b/11b → 00b/10b/01b/11b The Value after Reset of P61 to P37, modified (×2 → ×4)
		250	Note 2 to Note 4: Table heading, modified (00b 01b 10b 11b → 00b 10b 01b 11b)
		279	4.2.2.16 Schmitt Control Registers (PFC_SMT_mn): Note 1, deleted
		311	4.2.4.3 Multiplexed Pins (in Control Mode): Switching between Functions Multiplexed on the Pins and Control of the Pins, modified (the external P00, P30, P50, P70, P90, or PB1 pin → the external P13, P33, P81, PA5, PC0, PG7, PL0, or PS0 pin, P02, P03, or P15 → P44, PC1, or PJ3)
		317	Figure 4.2-10 Output Port Group Operation, modified (P6* → P1*)
		4.3 System Controller (SYS)	
		320	4.3.1.1 Functions, modified (Trust zone control → TrustZone control)
		367	4.3.3.2.33 Slave Access Control Register 21 (SYS_SLVACCCTL21): The main text, added
		387	4.3.3.2.53 Slave Access Control Register 57 (SYS_SLVACCCTL57): The main text, added
		4.4 Clock Pulse Generator (CPG)	
		560	Table 4.4-38 Target Units for Control by the CPG_BUS_m_MSTOP Registers (m = 1 to 4) The control target of the CPG_BUS_3_MSTOP register in Spec. MSTOP6_ON, modified (GPV_ACPU → Reserved)
		560	Table 4.4-39 Target Units for Control by the CPG_BUS_m_MSTOP Registers (m = 5 to 8) The control target of the CPG_BUS_8_MSTOP register in Spec. MSTOP7_ON, modified (GPV_COM → Reserved) The control target of the CPG_BUS_5_MSTOP register in Spec. MSTOP8_ON, modified (GPV_MCPU → Reserved) The control target of the CPG_BUS_8_MSTOP register in Spec. MSTOP11_ON, modified (NPU GPV → Reserved)
		561	Table 4.4-40 Target Units for Control by the CPG_BUS_m_MSTOP Registers (m = 9 to 13) The control target of the CPG_BUS_10_MSTOP register in Spec. MSTOP4_ON, modified (GPV_VIDEO1 → Reserved) The control target of the CPG_BUS_9_MSTOP register in Spec. MSTOP10_ON, modified (GPV_VIDEO0 → Reserved)
		4.5 Power Management Unit (PMU)	
		618	Table 4.5-3 Relationship between the Power Domain and the Respective Unit: The Power Domain of Unit: SRAM1, modified
		621	Table 4.5-4 Switching Procedure from AWO/AWO_DDR_RETENTION to ALL_ON: The Remarks in Step 2 in the Pre-process column, modified
		4.6 Interrupt Controller	
		627	Figure 4.6-1 Block Diagram of the ICU, modified
		668	(27) GPT Interrupt CA55 Status Register (ICU_GPTINTA55CTL) The main text, modified The description of bits 31 to 0, modified (GPT CH1 → GPT1), (GPT CH0 → GPT0)
		680	(35) DMAC TEND Selection Register k (ICU_DMTENDSELk) (k = 0, 1): The main text, modified (Table 4.6-14 → Table 4.6-12)
		691	(1) Input event normalization/synchronization and event output: The description on the Register access security control, added

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	715 to 770	Table 4.6-24 List of Input Events: The “Active” column, deleted
		782	4.6.3.1 Features The description of the Non-secure interrupt control register (Channels NS0-41), deleted The description of the interrupt control register (Channels S0-41), deleted
		807	4.6.3.4.2 Software interrupt, added
		810	4.6.3.5.3 Software interrupt sequence, added
		4.7 DMA Controller (DMAC)	
		831	4.7.5.2.10 Channel Status Register n (DMACm_CHSTAT_n) (m = 0 to 4, n = 0 to 15): The description of bits 31 to 17, and 15 to 12, modified
		844	4.7.5.2.18 DMA Status EN Register (DMACm_DST_EN) (m = 0 to 4): The description of bits 31 to 8, modified
		845	4.7.5.2.19 DMA Status ER Register (DMACm_DST_ER) (m = 0 to 4): The description of bits 31 to 8, modified
		846	4.7.5.2.20 DMA Status END Register (DMACm_DST_END) (m = 0 to 4): The description of bits 31 to 8, modified
		847	4.7.5.2.21 DMA Status TC Register (DMACm_DST_TC) (m = 0 to 4): The description of bits 31 to 8, modified
		848	4.7.5.2.22 DMA Status SUS Register (DMACm_DST_SUS) (m = 0 to 4): The description of bits 31 to 8, modified
		887	Figure 4.7-18 Level Detection Timing (HIEN = 1b, REQD = 0b, AM[2:0] = 001b), modified (HCLK → ACLK)
		4.9 Debug Interface	
		937	4.9.2.2.1 ID Register (CST_BSID) The Initial Value of bits 31 to 28, modified The description of bits 3 to 0, modified
		SECTION 5 TIMER	
		5.3 Realtime Clock (RTC)	
		957	5.3.2.2.2 Second Counter (RTC_RSECCNT)/Binary Counter 0 (RTC_BCNT0): RW and the description of bit 7, modified
		959	5.3.2.2.3 Minute Counter (RTC_RMINCNT)/Binary Counter 1 (RTC_BCNT1): RW and the description of bit 7, modified
		961	5.3.2.2.4 Hour Counter (RTC_RHRCNT)/Binary Counter 2 (RTC_BCNT2): RW and the description of bit 7, modified
		963	5.3.2.2.5 Day-of-Week Counter (RTC_RWKCNT)/Binary Counter 3 (RTC_BCNT3): RW and the description of bits 7 to 3, modified
		965	5.3.2.2.6 Date Counter (RTC_RDAYCNT): RW and the description of bits 7 and 6, modified
		966	5.3.2.2.7 Month Counter (RTC_RMONCNT): RW and the description of bits 7 to 5, modified
		967	5.3.2.2.8 Year Counter (RTC_RYRCNT): RW and the description of bits 15 to 8, modified
		974	5.3.2.2.12 Day-of-Week Alarm Register (RTC_RWKAR)/Binary Counter 3 Alarm Register (RTC_BCNT3AR): RW and the description of bits 6 to 3, modified
		976	5.3.2.2.13 Date Alarm Register (RTC_RDAYAR)/Binary Counter 0 Alarm Enable Register (RTC_BCNT0AER): RW and the description of bit 6, modified
		978	5.3.2.2.14 Month Alarm Register (RTC_RMONAR)/Binary Counter 1 Alarm Enable Register (RTC_BCNT1AER): RW and the description of bits 6 and 5, modified
		980	5.3.2.2.15 Year Alarm Register (RTC_RYRAR)/Binary Counter 2 Alarm Enable Register (RTC_BCNT2AER): RW and the description of bits 15 to 8, modified
		982	5.3.2.2.16 Year Alarm Enable Register (RTC_RYRAREN)/Binary Counter 3 Alarm Enable Register (RTC_BCNT3AER): RW and the description of bits 6 to 0, modified
		984	5.3.2.2.17 RTC Status Register (RTC_RSR): RW and the description of bits 7 to 3, modified
		985	5.3.2.2.18 RTC Control Register 1 (RTC_RCR1): RW and the description of bit 3, modified

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	986	5.3.2.2.19 RTC Control Register 2 (RTC_RCR2): RW and the description of bit 3, modified
		5.4 Watchdog Timer (WDT)	
		1003	Table 5.4-1 WDT Specifications: The Specifications of Parameter: Count source, modified (clk_loco → LOCO clock)
		1004	Figure 5.4-1 WDT Block Diagram, modified (clk_loco → LOCO clock)
		1007	5.4.2.2.2 WDT Control Register (WDTm_WDTCR) The description of bits 7 to 4, modified (clk_loco → LOCO clock) The description of the RPES[1:0] bits (Window End Position Select), modified The description of the RPSS[1:0] bits (Window Start Position Select), modified
		1010	5.4.2.2.3 WDT Status Register (WDTm_WDTSR) The description of the UNDFF flag (Underflow Flag), modified (bus clock → WDT_n_clkp (n = 0 to 3)) The description of the REFEF flag (Refresh Error Flag), modified (IWDTCLK → LOCO clock, clk_loco → WDT_n_clkp (n = 0 to 3))
		1013	5.4.3.1 Register Start Mode: The main text, modified
		1016	5.4.3.3 Refresh Operation: The main text, modified
		1017	Figure 5.4-5 WDT Refresh Operation Waveforms when WDTCR.CKS[3:0] = 4h and WDTCR.TOPS[1:0] = 01b, modified (clk_loco → LOCO clock, clkp → WDT_n_clkp)
		1019	5.4.3.6 Reading the Down-Counter Value: The main text, modified (clk_loco → LOCO clock)
		1019	Figure 5.4-6 Processing for Reading WDT Down-Counter Value when WDTCR.CKS[3:0] = 4h and WDTCR.TOPS[1:0] = 01b, modified (clk_loco → LOCO clock, clkp → WDT_n_clkp)
		1020	Table 5.4-6 WDT1 Setting Procedure for Error System Reset: The description of Step 6, modified (clk_loco → LOCO clock)
		5.6 Compare Match Timer W (CMTW)	
		1045	5.6.2.2.3 Timer I/O Control Register (CMTWm_CMWIOR): The description of bit 14, and 7 and 6, modified
		5.7 General-Purpose Timer (GPT)	
		1075	5.7.2.2.1 General-Purpose Timer Write-Protection Register (GPTm_n_GTWP): The description of bits 31 to 16 and 7 to 5, modified
		1083	5.7.2.2.5 General-Purpose Timer Start Source Select Register (GPTm_n_GTSSR): R/W and the description of bits 30 to 24, modified
		1087	5.7.2.2.6 General-Purpose Timer Stop Source Select Register (GPTm_n_GTPSR): R/W and the description of bits 30 to 24, modified
		1091	5.7.2.2.7 General-Purpose Timer Clear Source Select Register (GPTm_n_GTCSR): R/W and the description of bits 30 to 24, modified
		1095	5.7.2.2.8 General-Purpose Timer Up Count Source Select Register (GPTm_n_GTUPSR): R/W and the description of bits 31 to 24, modified
		1099	5.7.2.2.9 General-Purpose Timer Down Count Source Select Register (GPTm_n_GTDNSR): R/W and the description of bits 31 to 24, modified
		1103	5.7.2.2.10 General-Purpose Timer Input Capture Source Select Register A (GPTm_n_GTICASR): R/W and the description of bits 31 to 24, modified
		1107	5.7.2.2.11 General-Purpose Timer Input Capture Source Select Register B (GPTm_n_GTICBSR): R/W and the description of bits 31 to 24, modified
		1111	5.7.2.2.12 General-Purpose Timer Control Register (GPTm_n_GTCR): R/W and the description of bits 31 to 27, 22 to 19, and 15 to 1, modified
		1113	5.7.2.2.13 General-Purpose Timer Count Direction and Duty Setting Register (GPTm_n_GTUDDTYC): R/W and the description of bits 31 to 28, 23 to 20, and 15 to 2, modified
		1116	5.7.2.2.14 General-Purpose Timer I/O Control Register (GPTm_n_GTIOR): R/W and the description of bits 28, 27, 21, 12, 11, and 5, modified
		1120	5.7.2.2.15 General-Purpose Timer Interrupt Output Setting Register (GPTm_n_GTINTAD): R/W and the description of bits 31, 27, 26, 23 to 20, and 15 to 8, modified

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	1123	5.7.2.2.16 General-Purpose Timer Status Register (GPTm_n_GTST): R/W and the description of bits 27 to 25, 23 to 20, and 14 to 11, modified
		1130	5.7.2.2.17 General-Purpose Timer Buffer Enable Register (GPTm_n_GTBER): R/W and the description of bits 31, 27, 23, and 7 to 4, modified
		1135	5.7.2.2.18 General-Purpose Timer Interrupt and A/D Converter Start Request Setting Register (GPTm_n_GTITC): R/W and the description of bits 31 to 15, 13, and 11, modified
		1146	5.7.2.2.27 General-Purpose Timer Dead Time Control Register (GPTm_n_GTDTCR): R/W and the description of bits 31 to 9, 7, 6, and 3 to 1, modified
		1151	5.7.2.2.31 General-Purpose Timer Output Protection Function Temporary Release Register (GPTm_n_GTSOTR): R/W and the description of bits 31 to 1, modified
		1152	5.7.2.2.32 General-Purpose Timer Extended Interrupt Skipping Counter Control Register (GPTm_n_GTEITC): R/W and the description of bits 19, 18, 11 to 8, 3, and 2, modified
		1158	5.7.2.2.34 General-Purpose Timer Extended Interrupt Skipping Setting Register 2 (GPTm_n_GTEITLI2): R/W and the description of bits 31 to 8, modified
		1160	5.7.2.2.35 General-Purpose Timer Extended Buffer Transfer Skipping Setting Register (GPTm_n_GTEITLB): R/W and the description of bits 31, 27, 23, 19, 15 to 11, 7, and 3, modified
		1162	5.7.2.2.36 General-Purpose Timer Inter Channel Logical Operation Function Setting Register (GPTm_n_GTICLF): R/W and the description of bits 31 to 26, 19, 15 to 10, and 3, modified
		1167	5.7.2.2.38 General-Purpose Timer Operation Enable Bit Simultaneous Control Register (GPTm_n_GTSECR): R/W and the description of bits 31 to 25, 23 to 17, 15 to 10, and 7 to 2, modified
		1188	Table 5.7-22 Example for Setting GTCCRA and GTCCRB Buffer Operation for Output Compare: The description of item No. 6, modified (Figure 5.7-15 → Figure 5.7-16)
		5.8 Port Output Enable for GPT (POEG)	
		1292	5.8.2.2.1 POEG Group n Setting Register (POEG_POEGGn) (n = A to H) The description of bits 6 and 2, modified
		1296	5.8.3.4 Release from Output-Disable The main text, modified (GTST.DTEF, added)
		SECTION 6 HIGH-SPEED INTERFACE	
		6.1 High-Speed Interface Overview	
		1300	6.1 High-Speed Interface Overview: The description of PCI Express 3.0 Interface (PCIe) (See 6.6), modified
		6.2 SD/MMC Host Interface (SD)	
		1361	6.2.3.1.2 Bus signal voltage switch: The description of item No. 4, modified
		6.4 USB3.2 Gen2x1 Interface (USB3)	
		1422	Figure 6.4-1 USB Connection Configuration: The title, modified
		1431	6.4.4.3.2 Register List USB3m_HOST_U2HCSPC2: The register name, modified USB3m_HOST_DCDn: The register name and offset address, modified
		1454	(1) U3P1 Port Status and Control Register (USB3m_HOST_U3P1PORTSC) The description of bit 22, modified The description of bits 8 to 5, modified (the U0 state → the U3 state)
		1458	(3) U3P1 Port Link Info Register (USB3m_HOST_U3P1PORTLI): R/W and the description of bits 15 to 0, modified
		1463	(6) U2P1 Port Power Management Status and Control Register (USB3m_HOST_U2P1PORTPMSC): The description of bits 31 to 28, 16, 7 to 4, and 2 to 0, modified (xHCI 1.1 → xHCI 1.2)
		1466	(8) U2P1 Port Hardware LPM Power Control Register (USB3m_HOST_U2P1PORTHLPMC): The description of bits 13 to 10 and 9 to 2, modified (xHCI 1.1 → xHCI 1.2)

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	1471	(c) Offset 08h - xHCI Supported Protocol Capability Field Definitions (USB3.2) Register (USB3m_HOST_U3HCSPC3): The description of bits 27 to 25, modified (USB2 → USB3)
		1477	(c) Offset 08h - xHCI Supported Protocol Capability Field Definitions (USB2.0) Register (USB3m_HOST_U2HCSPC3): The description of bit 20, modified (xHCI 1.1 → xHCI 1.2)
		1494	(2) Device Context Doorbell Register (Slot #n) (USB3m_HOST_DCDn) (n = 1 to 32): The address, modified
		1496	(b) Configuration Status1 Register (USB3m_HOST_CFGSTS1): The initial value of bits 27 and 26, modified
		1510	(n) PHY Control Register (USB3m_HOST_PHYCTRL): The description of bits 19, 18, and 17, modified
		1517	(a) Battery Charging Control Register (USB3m_HOST_BCCTRL): The description of bits 7 to 2, modified
		1521	(4) PIPE Status and Control Register 3 (USB3m_HOST_U3P0PIPESC3): The description of bits 21 to 16, modified
		6.5 USB2.0 Interface	
		1541	Table 6.5-3 External Signal Pins: Name: The Input/Output column of USB20_DP, USB20_DM, USB21_DP, and USB21_DM, modified
		1547	(3) OTG function: The description of the OTG_PERI bit, modified
		1552	(1) HcRevision Register (USB2m_HOST_HcRevision): The description of bits 31 to 8, modified
		1553	(2) HcControl Register (USB2m_HOST_HcControl): The description of bits 31 to 10, and 8, modified
		1555	(3) HcCommandStatus Register (USB2m_HOST_HcCommandStatus): The description of bits 31 to 18, and 15 to 3, modified
		1556	(4) HcInterruptStatus Register (USB2m_HOST_HcInterruptStatus): The description of bits 31 to 7, modified
		1558	(5) HcInterruptEnable Register (USB2m_HOST_HcInterruptEnable): The description of bits 29 to 7, modified
		1560	(6) HcInterruptDisable Register (USB2m_HOST_HcInterruptDisable): The description of bits 29 to 7, modified
		1561	(7) HcHCCA Register (USB2m_HOST_HcHCCA): The description of bits t 7 to 0, modified
		1562	(8) HcPeriodicCurrentED Register (USB2m_HOST_HcPeriodCurrentED): The description of bits 3 to 0, modified
		1563	(9) HcControlHeadED Register (USB2m_HOST_HcControlHeadED): The description of bits 3 to 0, modified
		1564	(10) HcControlCurrentED Register (USB2m_HOST_HcControlCurrentED): The description of bits 3 to 0, modified
		1565	(11) HcBulkHeadED Register (USB2m_HOST_HcBulkHeadED): The description of bits 3 to 0, modified
		1566	(12) HcBulkCurrentED Register (USB2m_HOST_HcBulkCurrentED): The description of bits 3 to 0, modified
		1567	(13) HcDoneHead Register (USB2m_HOST_HcDoneHead): The description of bits 3 to 0, modified
		1569	(15) HcFmRemaining Register (USB2m_HOST_HcFmRemaining): The description of bits 30 to 14, modified
		1571	(17) HcPeriodicStart Register (USB2m_HOST_HcPeriodicStart): The description of bits 31 to 14, modified
		1572	(18) HcLSThreshold Register (USB2m_HOST_HcLSThreshold): The description of bits 31 to 12, modified
		1573	(19) HcRhDescriptorA Register (USB2m_HOST_HcRhDescriptorA): The description of bits 23 to 13, modified
		1574	(20) HcRhDescriptorB Register (USB2m_HOST_HcRhDescriptorB): The description of bits 31 to 18, 16 to 2, and 0, modified

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	1575	(21) HcRhStatus Register (USB2m_HOST_HcRhStatus): The description of bits 30 to 18, and 14 to 2, modified
		1577	(22) HcRhPortStatus[1:NDP] Register (USB2m_HOST_HcRhPortStatus1): The description of bits 31 to 21, 15 to 10, and 7 to 5, modified
		1580	(1) HCIVERSION/CAPLENGTH Register (USB2m_HOST_CAPL_VERSION): The description of bits 15 to 8, modified
		1581	(2) HCSPARAMS Register (USB2m_HOST_HCSPARAMS): The description of bits 31 to 24, 19 to 17, 6, and 5, modified
		1582	(3) HCCPARAMS Register (USB2m_HOST_HCCPARAMS): The description of bits 31 to 20, and 3, modified
		1585	(1) USBCMD Register (USB2m_HOST_USBCMD): The description of bits 31 to 28, 14 to 12, 10, and 7, modified
		1587	(2) USBSTS Register (USB2m_HOST_USBSTS): The description of bits 31 to 17, and 11 to 6, modified
		1589	(3) SBINTR Register (USB2m_HOST_USBINTR): The description of bits 31 to 17, and 15 to 6, modified
		1590	(4) FRINDEX Register (USB2m_HOST_FRINDEX): The description of bits 31 to 14, modified
		1592	(6) PERIODICLISTBASE Register (USB2m_HOST_PERIODICLISTBASE): The description of bits 11 to 0, modified
		1593	(7) ASYNCLISTADDR Register (USB2m_HOST_ASYNCLISTADDR): The description of bits 4 to 0, modified
		1594	(8) CONFIGFLAG Register (USB2m_HOST_CONFIGFLAG): The description of bits 31 to 1, modified
		1595	(9) PORTSC1Register (USB2m_HOST_PORTSC1): The description of bits 15 and 14, modified
		1599	(1) INT_ENABLE Register (USB2m_HOST_INT_ENABLE): The description of bits 31 to 5, modified
		1600	(2) INT_STATUS Register (USB2m_HOST_INT_STATUS): The description of bits 31 to 5, modified
		1601	(3) AHB_BUS_CTR Register (USB2m_HOST_AHB_BUS_CTR): The description of bits 31 to 16, 11 to 9, 7, 6, 3, and 2, modified
		1602	(4) USBCTR Register (USB2m_HOST_USBCTR): The description of bits 31 to 3, modified
		1603	(5) Register Enable/Clock Gating Control Register (USB2m_HOST_REGEN_CG_CTRL): The description of bits 30, and 27 to 1, modified
		1604	(6) Suspend Control Register (USB2m_HOST_SPD_CTRL): The description of bits 29 to 24, and 22 to 1, modified
		1607	(8) Overcurrent Detection/Sleep Timer Setting Register (USB2m_HOST_OC_SLP_TIMSET): The description of bits 31 to 29, modified
		1609	(10) UTMI+PHY Control Register (USB2m_HOST_UTMI_CTRL): The description of bits 30 to 9, modified
		1610	(11) PORT_LPM_CTR1 Register (USB2m_HOST_PORT_LPM_CTRL1): The description of bits 31 to 8, modified
		1611	(1) Common Control Register (USB2m_HOST_COMMCTRL): The description of bits 30 to 0, modified
		1612	(2) OTG-BC Interrupt Status Register (USB2m_HOST_OBINTSTA) The description of bits 31 to 18, 15 to 7, and 5, modified The bit name, R/W, and description of bits 3 and 2, modified
		1613	(3) OTG-BC Interrupt Enable Register (USB2m_HOST_OBINTEN) The description of bits 31 to 18, 15 to 7, and 5, modified The bit name, R/W, and description of bits 3 and 2, modified
		1614	(4) VBUS Control Register (USB2m_HOST_VBCTRL): The description of bits 31 to 5, and 3 to 1, modified

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	1615	(5) Line Control Port 1 Register (USB2m_HOST_LINECTRL1) The description of bits 31 to 26, 23, 22, 15 to 4, and 1, modified The bit name and description of bits 25 and 24, modified
		1616	(6) BC Control Port 1 Register (USB2m_HOST_BCCTRL1): The description of bits 31 to 26, 23 to 10, 7, and 6, modified
		1621	(3) U2H_EHCI_INT assertion source and control The description of “Interrupt enable control by register”, modified (bit[17:16] → bit[16]) The “Interrupt source” table, modified
		1623	(5) U2H_OBINT assertion source and control The description of “Interrupt enable control by register”, modified (bit[27:24], bit[18:16], bit[12:8], bit[3:0] → bit[17:16], bit[6], bit[4], bit[1:0]) The “Interrupt source” table, modified The description of “Clearing the interrupt”, modified (bit[27:24], bit[18:16], bit[12:8], bit[3:0] → bit[17:16], bit[6], bit[4], bit[1:0])
		1630	6.5.2.8.3 Port Power (VBUS) control specifications: The main text, modified
		1644	(1) System Configuration Control Register 0 (USB20_FUNC_SYSCFG0): The description of bits 15 to 9, 6, and 3 to 1, modified
		1646	(2) System Configuration Control Register 1 (USB20_FUNC_SYSCFG1): The description of bits t 15 to 12, 11 to 8, 7, and 6, modified
		1648	(1) Device State Control Register 0 (USB20_FUNC_DVSTCTR0): The description of bits 15 to 9, and 7 to 3, modified
		1650	(1) USB Test Mode Register (USB20_FUNC_TESTMODE): The description of bits 15 to 4, modified
		1653	(2) CFIFO Port Select Register (USB20_FUNC_CFIFOSEL): The description of bits 13, 12, 9, 7, 6, and 4, modified
		1655	(3) DnFIFO Port Select Register (USB20_FUNC_DnFIFOSEL) (n = 0, 1): The description of bits 9 to 4, modified
		1657	(4) CFIFO Port Control Register (USB20_FUNC_CFIFOCTR) D0FIFO Port Control Register (USB20_FUNC_D0FIFOCTR) D1FIFO Port Control Register (USB20_FUNC_D1FIFOCTR) The description of bit 12, modified
		1659	(1) Interrupt Enable Register 0 (USB20_FUNC_INTENB0): The description of bits 7 to 0, modified
		1663	(1) SOF Pin Configuration Register (USB20_FUNC_SOFCFG): The description of bits 15 to 7, 5, 4, 1, and 0, modified
		1673	(1) Frame Number Register (USB20_FUNC_FRMNUM): The description of bits 13 to 11, modified
		1675	(2) Micro Frame Number Register (USB20_FUNC_UFRMNUM): The description of bits 15 to 3, modified
		1676	(1) USB Address Register (USB20_FUNC_USBADDR): The description of bits 15 to 7, modified
		1681	(1) DCP Configuration Register (USB20_FUNC_DCPCFG): The description of bits 15 to 9, and 6 to 0, modified
		1682	(2) DCP Max. Packet Size Register (USB20_FUNC_DCPMAXP): The description of bits 15 to 7, modified
		1683	(3) DCP Control Register (USB20_FUNC_DCPCTR): The description of bits 14 to 9, 4, and 3, modified
		1686	(1) Pipe Window Select Register (USB20_FUNC_PIPESEL): The description of bits 15 to 4, modified
		1687	(2) Pipe Configuration Register (USB20_FUNC_PIPECFG): The description of bits 13 to 11, 6, and 5, modified
		1691	(3) Pipe Buffer Setting Register (USB20_FUNC_PIPEBUF): The description of bits 15, 9, and 8, modified

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	1693	(4) Pipe Maximum Packet Size Register (USB20_FUNC_PIPEMAXP): The description of bits 15 to 11, modified
		1694	(5) Pipe Cycle Control Register (USB20_FUNC_PIPEPERI): The description of bits 15 to 13, and 11 to 3, modified
		1697	(1) PIPEn Control Register (USB20_FUNC_PIPEnCTR) (n = 1 to 5): The description of bits 13 to 11, and 4 to 2, modified
		1702	(2) PIPEn Control Register (USB20_FUNC_PIPEnCTR) (n = 6 to 9): The description of bits 14 to 10, and 4 to 2, modified
		1704	(1) PIPEn Transaction Counter Enable Register (USB20_FUNC_PIPEnTRE) (n = 1 to 5): The description of bits 15 to 10, and 7 to 0, modified
		1707	(1) Low Power Control Register (USB20_FUNC_LPCTRL): The description of bits 15 to 8, and 6 to 0, modified
		1708	(1) Low Power Status Register (USB20_FUNC_LPSTS): The description of bits 15, and 13 to 0, modified
		1709	(1) PHY Function Control Register (USB20_FUNC_PHYFUNCTR): The description of bits 15, and 13 to 0, modified
		1710	(1) PHY_OTG Control Register (USB20_FUNC_PHYOTGCTR): The description of bits 15 to 11, and 8 to 0, modified
		1711	(1) Peripheral L1 Control Register 1 (USB20_FUNC_PL1CTRL1): The description of bits 15, 13, and 12, modified
		1713	(1) Peripheral L1 Control Register 2 (USB20_FUNC_PL1CTRL2): The description of bits 15 to 13, and 7 to 0, modified
		1720	(a) Channel Status Register ch n (USB20_FUNC_CHSTAT_n) (n = 0, 1): The description of bits 23 to 19, and 15 to 12, modified
		1724	(a) Channel Control Register ch n (USB20_FUNC_CHCTRL_n) (n = 0, 1): The description of bits 31 to 20, 15, 13, 11, and 10, modified
		1727	(a) Channel Configuration Register ch n (USB20_FUNC_CHCFG_n) (n = 0, 1): The description of bits 22, 10 to 4, 2, and 1, modified
		1730	(a) Channel Interval Register ch n (USB20_FUNC_CHITVL_n) (n = 0, 1): The description of bits 31 to 16, modified
		1731	(a) Channel Extension Register ch n (USB20_FUNC_CHEXT_n) (n = 0, 1) The description of bits 31 to 20, modified RW and the description of bits 7 to 4, modified
		1740	(a) DMA Control Register (USB20_FUNC_DCTRL): The description of bits 31 to 28, 23 to 20, and 15 to 2, modified
		1741	(a) Descriptor Interval Register (USB20_FUNC_DSCITVL): The description of bits 31 to 16, and 7 to 0, modified
		1742	(a) DMA Status EN Register (USB20_FUNC_DSTAT_EN): The description of bits 31 to 2, modified
		1743	(a) DMA Status ER Register (USB20_FUNC_DSTAT_ER): The description of bits 31 to 2, modified
		1744	(a) DMA Status END Register (USB20_FUNC_DSTAT_END): The description of bits 31 to 2, modified
		1745	(a) DMA Status TC Register (USB20_FUNC_DSTAT_TC): The description of bits 31 to 2, modified
		1746	(a) DMA Status SUS Register (USB20_FUNC_DSTAT_SUS): The description of bits 31 to 2, modified
		6.6 PCI Express 3.0 Interface (PCIe)	
		1810	6.6.1 Overview: PCI Express Specification (Compliant with the PCI Express Base Specification 4.0): The description, modified
		1812	Table 6.6-1 List of External Pins: PCIE0_RSTOUTB: The function, modified
		1819	Table 6.6-4 Register Base Addresses: Base Address Name: <PCI_PHY_base>, deleted

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	1820	6.6.4.1 Register Descriptions (Root Complex Mode): Registers with the HwInit attribute: The table, modified
		1871	(38) PCIe Core Control 1 Register (PCI_RC_PCCTRL1): The bit name and description of bit 11, modified
		1878	(42) PCIe Core Status 5 Register (PCI_RC_PCSTAT5): The description of bit 0, modified
		1885	(49) DMAC Control Register (PCI_RC_DMACTRL): The initial value of bits 2 to 0, modified
		1923	(1) Vendor and Device ID Register (PCI_RC_VID) The main text, modified The description of bits 31 to 16, and 15 to 0, modified
		1942	(18) PM Status/Control Register (PCI_RC_PMSC): R/W of bit 15, modified
		1945	(21) Device Control/Status Register (PCI_RC_DEVCS): The description of bits 11 to 8, modified
		1958	(29) Device Control 2/Status 2 Register (PCI_RC_DEVCS2): The description of bits 31 to 5, modified
		1977	Table 6.6-45 Valid Reset Signal, deleted
		1997	(63) Link Control 3 Register (PCI_RC_LINC3): The bit name and description of bits 15 to 9, modified
		2004	6.6.4.2 Register Descriptions (Endpoint Mode): The table, modified
		2053	(35) PCIe Core Control 1 Register (PCI_EP_PCCTRL1): The bit name and description of bit 11, modified
		2102	(1) Vendor and Device ID (Function #n) Register (PCI_EP_VID_Fn) (n = 0, 1) The main text, modified The description of bits 31 to 16, and 15 to 0, modified
		2186	(68) Link Control 3 Register (Function #0) (PCI_EP_LINC3_F0): The bit name and description of bits15 to 9, modified
		2188	(70) Lane Equalization Control Register Lane #n (Function #0) (PCI_EP_LEQCTL_F0): R/W(PCIe) and R/W(UDL) of bits 30 to 24, 22 to 16, 14 to 8, and 6 to 0, modified
		2231, 2232	Figure 6.6-4 Legacy Interrupt Issuing Flow, added Figure 6.6-5 MSI Issuing Flow, added
		2233	Table 6.6-127 External Interrupt Notification: Interrupt Notification Name: UI_EXTMSI_FUNC0/1/2/3/4 [2:0], modified
		2252	6.6.6.1 Setting Up: The main text, modified
		2253	6.6.6.1.1 Changing the Initial Values of the Registers (1) Setting the Initial Values of the Registers, deleted (3) Physical Layer Control/Monitor Registers, added (1), (2) Setting the initial values of the registers, added
		2271	Table 6.6-129 Initialization Procedure (RC) Step 8: The description, modified Note 3, added
		2273	Table 6.6-130 Initialization Procedure (EP) Step 8, 9: The description, modified Note 3, added
		SECTION 7 LOW-SPEED INTERFACE	
		7.2 Expanded Serial Peripheral Interface (xSPI)	
		2281	Table 7.2-1 xSPI Specifications xSPI function: The description, modified Note 1, added
		2284	7.2.2.1 List of Registers: The table, modified
		2321	(2) xSPI Common Status Register (XSPIm_COMSTT): The initial value of bits 22 to 20, modified
		2351	7.2.3.4.2 Reset pattern: The main text, modified

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	7.3 Serial Communications Interface (RSCI)	
		2409	7.3.2.2.15 Simple-LIN (SCIX) Control Register 0 (RSCIm_XCR0): The description of bits 25 and 24, modified (Note 2, deleted)
		7.4 Serial Communications Interface with FIFO (SCIF)	
		2604	Figure 7.4-12 Receive Data Sampling Timing, modified
		7.5 Serial Peripheral Interface (RSPI)	
		2607	Table 7.5-1 SPI Specifications (2/3): SSL control function: The description, modified
		2608	Table 7.5-1 SPI Specifications (3/3): Others: The description, modified
		2609	7.5.1.1 Features: The main text, modified
		2611	7.5.2.1 List of Registers RSPIm_SPSCR, RSPIm_SPDCR2: The access size, modified RSPIm_SPTFSR: The initial value, modified
		2626	7.5.2.2.6 SPI Control Register (RSPIm_SPCR): SPFRF bit (SPI Frame Format Select): The description, modified
		2629	7.5.2.2.9 SPI Pin Control Register (RSPIm_SPPCR) The bit name and description of bit 2, modified SPOM bit (SPI Output Pin Mode): The description, deleted
		2631	7.5.2.2.11 SPI Slave Select Polarity Register (RSPIm_SSLP): SSLnP bits (SSLn Signal Polarity Setting): The description, modified
		2633	7.5.2.2.13 SPI Sequence Control Register (RSPIm_SPSCR): The access size, modified
		2636	7.5.2.2.14 SPI Command Register n (RSPIm_SPCMDn) (n = 0 to 7): SSLA[1:0] bits (SSL Signal Assertion): The description, modified
		2637	7.5.2.2.15 SPI Data Control Register (RSPIm_SPDCR) The bit name and description of bits 2 and 1, modified SLSEL[1:0] bits (SSL Pin Output Select): The description, deleted
		2639	7.5.2.2.16 SPI Data Control Register 2 (RSPIm_SPDCR2): The access size, modified
		2643	7.5.2.2.18 SPI Status Register (RSPIm_SPSR): MODF bit (Mode Fault Error Flag): The description, modified (the description of [Multi-master mode], deleted)
		2647	7.5.2.2.19 SPI Transfer FIFO Status Register (RSPIm_SPTFSR): The initial value of bits 4 and 2, modified
		2652	7.5.3.1 Overview of SPI Operations: The main text, modified
		2652	Table 7.5-7 Relationship between SPI Modes and SPCR Settings and Description of Each Mode, modified (Single-Master → Master) (Multi-Master(SPI Operation), deleted) (Output pin mode, deleted)
		2653	7.5.3.2 SPI Pin Control: The main text, modified
		2653	Table 7.5-8 Relationship between Pin States and Set Control Bit Values, modified
		2655	7.5.3.3 SPI System Configuration Examples: The main text, modified
		2655	7.5.3.3.1 Master/single slave (this LSI = master): The title, modified The main text, modified
		2655	Figure 7.5-7 Master/Single-Slave Configuration Example (This LSI = Master): The title, modified
		2655	7.5.3.3.2 Master/single slave (this LSI = slave): The title, modified The main text, modified
		2656	Figure 7.5-8 Master/Single-Slave Configuration Example (This LSI = Slave, CPHA = 0b): The title, modified Figure 7.5-9 Master/Single-Slave Configuration Example (This LSI = Slave, CPHA = 1b): The title, modified
		2656	7.5.3.3.3 Master/multi-slave (this LSI = master): The title, modified The main text, modified

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	2657	Figure 7.5-10 Master/Multi-Slave Configuration Example (This LSI = Master): The title, modified
		2657	7.5.3.3.4 Master/multi-slave (with this LSI acting as slave): The title, modified The main text, modified
		2658	Figure 7.5-11 Master/Multi-Slave Configuration Example (This LSI = Slave): The title, modified SPI master, modified (RSPCK → SPCK)
		—	7.5.3.3.5 Multi-master/multi-slave (with this LSI acting as master), deleted
		2691	Table 7.5-10 Abnormal Transfer Occurrence Conditions and Error Detection Function of SPI, modified
		2697	7.5.3.10.3 Mode fault error The main text, modified (2) In the TI-SSP case: The description, modified
		2699	7.5.3.13.1 Master mode operation: The main text, modified
		2710	Figure 7.5-63 Example of Initialization Flowchart in Master Mode (SPI Operation), modified (“Sets output mode (CMOS/open-drain)”, deleted)
		2717	Figure 7.5-68 Example of Initialization Flow in Slave Mode, modified (“Set SPI pin control register (SPPCR)”, deleted)
		2725	Figure 7.5-76 Example of Initialization Flowchart in Master Mode (Clock Synchronous Operation), modified (“Sets output mode (CMOS/open-drain)”, deleted)
		2727	Figure 7.5-77 Example of Initialization Flow in Slave Mode, modified (“Set SPI pin control register (SPPCR)”, deleted)
		2732	Table 7.5-16 Communication End Event Generating Conditions (Receive Only Slave Mode): Bits SPOM and SLSEL[1:0], deleted
		7.7 I^2C Bus Interface (RIIC)	
		2754	7.7.2.2.1 I2C Bus Control Register 1 (RIICm_ICCR1) The description of bit 2, modified
		7.8 I3C Bus Interface (I3C)	
		3112	Figure 7.8-111 Example of I^2C Initialization Flowchart (Single Buffer Transfer), modified
		3113	Figure 7.8-112 Example of I3C initialization flowchart, modified
		7.9 CAN-FD Interface (CANFD)	
		3144	Table 7.9-1 CANFD Specifications (2/2): Interrupt source: The description, modified
		3155	7.9.2.1 List of Registers: Note 1, added
		3156	7.9.2.2.1 Channel n Nominal Bit Rate Configuration Register (CFDCnNCFG) (n = 0 to 5): The description of bits 16 to 10, modified
		3204	7.9.2.2.18 Global Acceptance Filter List Entry Control Register (CFDGAFLECTR): AFLPN[5:0] bits (Acceptance Filter List Page Number): The description, modified (0h and Fh → 0h and 2Fh)
		3255	7.9.2.2.48 TX Message Buffer Control Register n (CFDTMCn) (n = 0 to 383): The main text, modified
		3257	7.9.2.2.49 TX Message Buffer Status Register n (CFDTMSTSn) (n = 0 to 383): The main text, modified
		3259	Table 7.9-7 Alignment of TMTRSTS[15:0] Mirror Bits, modified
		3261	Table 7.9-8 Alignment of TMTARSTS[15:0] Mirror Bits, modified
		3263	Table 7.9-9 Alignment of TMTCSTS[15:0] Mirror Bits, modified
		3265	Table 7.9-10 Alignment of TMTASTS[15:0] Mirror Bits, modified
		3267	Table 7.9-11 Alignment of TMIE[15:0] bits, modified
		3269	7.9.2.2.55 TX Queue Configuration/Control Register 0n (CFDTXQCC0n) (n = 0 to 5) The main text, modified (TXMB31 → TXMB15) TXQDC[4:0] bits (TX Queue Depth Configuration): The description, modified

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	3272	7.9.2.2.56 TX Queue Configuration/Control Register 1n (CFDTXQCC1n) (n = 0 to 5) The main text, modified (TXMB31 → TXMB15) TXQDC[4:0] bits (TX Queue Depth Configuration): The description, modified
		3275	7.9.2.2.57 TX Queue Configuration/Control Register 2n (CFDTXQCC2n) (n = 0 to 5) The main text, modified (TXMB63 → TXMB47) TXQDC[4:0] bits (TX Queue Depth Configuration): The description, modified
		3278	7.9.2.2.58 TX Queue Configuration/Control Register 3n (CFDTXQCC3n) (n = 0 to 5) The main text, modified (TXMB63 → TXMB47) TXQDC[4:0] bits (TX Queue Depth Configuration): The description, modified
		3319	7.9.2.2.80 Global Flexible CAN Mode Configuration Register (CFDGFCMC) The bit name of bit 2, modified FLXC1 bit (Flexible CAN Mode between Channel 2 and Channel 3): The description, added FLXC2 bit (Flexible CAN Mode between Channel 4 and Channel 5): The description, added
		3322	7.9.2.2.82 Global Test Configuration Register (CFDGTSTCFG): RTMPS[9:0] bits (RAM Test Mode Page Select): The description, modified
		3393	Figure 7.9-28 Message Buffer Configuration, modified
		3394	7.9.6.1.1 Normal RX Message Buffer Configuration: The main text, modified
		3395	7.9.6.2.1 FIFO Buffers Configuration: The main text, modified
		3397	(2) FIFO TX Message Buffer Link Configuration: The main text, modified (1Fh: TX Message Buffer 63 → 0Fh: TX Message Buffer 47)
		3405	7.9.7.2 DMA Transfer: Reception DMA, Transmission DMA: The description, modified
		3415	7.9.8.2 Transmission: The main text, modified
		3416	Figure 7.9-39 Channel Transmission Message Buffer Configuration, modified
		3427, 3428	7.9.8.2.4 TX Queue: The main text, modified
		3432	7.9.8.2.4 TX Queue: The table, modified (Range Width, Direction, and Note, modified)
		3434	Table 7.9-29 TX History List Buffer Number Entry, modified
		3441	7.9.9.2.1 RAM Test Mode: The main text, modified
		3441	Figure 7.9-55 RAM Page Structure, modified
		3447	Figure 7.9-59 Diagram of the Flexible CAN for Channel 0 and Channel 1: The title, modified: The text, modified (CH1 → LSI)
		3448	7.9.12 Flexible Transmission Buffer Assignment: The main text, modified (64 transmit buffers → 32 transmit buffers)
		7.10 12-Bit A/D Converter (ADC)	
		3454	Figure 7.10-1 Block Diagram of 12-Bit A/D Converter (Unit0), modified (ADTRG# → ADTRG)
		3508	Figure 7.10-9 Example of Operation of Compare Function (Compare Target: ANI000-ANI003): The title, modified
		3510	Figure 7.10-11 Example of Operation for Event Output of Compare Function, modified (AN → ANI)
		SECTION 9 IMAGE	
		9.1 Image Overview	
		3736	9.1.1 Variation of Display Output, added
		9.2 Camera Data Receiver Unit (CRU)	
		3740	9.2.3.1.1 Features COMMON: 1) Data input range: The description of [Pixel size], modified Note 1, modified
		3742	9.2.3.1.1 Features: The description of item No. 6 in “LINK”, modified
		3757	(9) Receive Data Type Enable Low Register (CRUm_CSI2nDTEL) (m = 0): The description of bits 31 to 0, modified (ICnMC → ICnIPMC)

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	3758	Table 9.2-4 Receive Data Type Enable Low Register (CSI2nDTEL): The data type and description of bits 18 to 16, modified
		3762	(12) Receive Status Clear Register (CRUm_CSI2nRXSC) (m = 0): The description of bit 17, modified
		3784	(30) Slave D-PHY Control Register (MSB) (CRUm_S_DPHYCTL_MSB) (m = 0): The description of bit 0, modified
		3800	Figure 9.2-7 Block Diagram of Image Converter, modified Note 2, added
		3804	9.2.4.2.1 List of Registers (m = 0): Register Name: The AXI-VD Bus Master FIFO Setting Register and the AXI-VD Bus Master Transfer Resume Register, added
		3809	Figure 9.2-8 Bypass Setting for Each Function, modified
		3812	(4) CRU Interrupt Enable Register2 (CRUm_CRUnIE2) (m = 0): The bit name and description of bit 11, modified
		3814	(5) CRU Interrupt Status Register1 (CRUm_CRUnINTS1) (m = 0): R/W and the description of bits 31, 11, and 3, modified
		3816	(6) CRU Interrupt Status Register2 (CRUm_CRUnINTS2) (m = 0): R/W and the description of bits 31 and 30, modified
		3855	(44) AXI-VD Bus Master Transfer Setting Register (CRUm_AMnAXIATTR) (m = 0): The description of bits 3 to 0, modified
		3856	(45) AXI-VD Bus Master FIFO Setting Register (m = 0) (CRUm_AMnFIFO), added
		3857	(46) AXI-VD Bus Master Transfer Resume Register (m = 0) (CRUm_AMnFIFOTRST), added
		3935	(120) CRU Test Image Generation Control 1 Register (CRUm_ICnTICTRL1) (m = 0): The description of bits 20 to 16, modified
		3960	Figure 9.2-17 Image of Data Storage (Example of Division by VC), modified
		3961	Figure 9.2-18 Image of Data Storage (Example of Division by MB), modified
		3972	9.2.4.3.4 Pattern Generator (for Debugging): The main text, modified (Specify V → Specify Y, Specify Y → Specify V)
		3974	Figure 9.2-31 Data Generation Circuit (in the Case of Pattern Generation): The description in the square of YU, modified (V LSB 4-bit → U LSB 4-bit)
		3989	Figure 9.2-43 Example of VC Specification and Selection of Several Data Types for Non-Image Processing, modified
		3990	Figure 9.2-44 Example of Settings for Full Bypass, modified
		3998	Table 9.2-23 Interrupt Sources and Registers, and Interrupt Signals [n = 0] The “Interrupt Source Status/Clear” column of Internal Signal Name: axi_mst_err_int, modified The “Conditions for Occurrence” column of cru_vd_addr_wend, cru_sd_addr_wend, and cru_vsd_addr_wend, modified
		4004	Figure 9.2-53 Reception Start Flow for the MIPI CSI-2 Input, modified (CPG_RST_13 and CPG_RSTMON_6, deleted)
		4006	Figure 9.2-55 Initializing the AXI Master, modified
		4010	Figure 9.2-58 Initializing the LINK, modified (CSI2nDTEL = F778_FF0Fh → CSI2nDTEL = F77F_FF0Fh)
		4012	Figure 9.2-61 Stopping the D-PHY, modified (CPG_RST_13, deleted)
		4018	Figure 9.2-65 Reception Start Flow for the Pattern Generator Input (CPG_RSTMON_6, deleted)
		9.4 LCD Controller (LCDC)	
		4037	9.4.3.1 List of VSPD Registers: Register Name: Display List-0 Wake Up Counter Register, added
		4069	(7) Display List-0 Wake Up Counter Register (VI6_DL_WUPCNT0), added

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	4074	Figure 9.4-8 Chrominance Interpolation Methods Selectable through CIPM Setting, modified (VSP → VSPD)
		4147	(6) BRS Blend Control Register A (LCDC_VSPD_VI6_BRSA_BLD): The description of bit 23, modified ((m = A, B, C, D, E) → (m = A, B))
		4149	(7) BRS Control Register B (LCDC_VSPD_VI6_BRSB_CTRL): The bit name and description of bits 18 to 16, modified
		4150	(8) BRS Blend Control Register B (LCDC_VSPD_VI6_BRSB_BLD): The description of bit 23, modified ((m = A, B, C, D, E) → (m = A, B))
		4161	9.4.4.2.1 DU Module Control Register 0 (LCDC_DU_MCR0): The bit name and description of bit 0, modified
		9.5 MIPI DSI Interface (DSI)	
		4202	Table 9.5-2 Clocks and the Frequency Range: Clock Name: The description of “hsclk”, modified (1/16 for MIPI_DSI_PLLCLK → 1/8 for the PLL clock frequency (FFOUT))
		4269	9.5.2.2.55 Sequence Channel x Set 0 Register (DSI_LINK_SQCHxSET0R): R/W of bit 23, modified
		4284	9.5.2.3.3 DSI PLL Clock Setting 0 Register (DSI_DPHY_PLLCLKSET0R): The main text, modified (F_{FOUT}, deleted)
		9.10 Frame Data Processor (FDP)	
		4577	Table 9.10-2 List of FDP1B Registers: The initial value of Abbreviation: FD1_CTL_CHACT, modified
		4584	(4) Channel Activation Register (FD1_CTL_CHACT) The initial value of bits 8 and 1, modified The description of bits 9, 8, 3, and 2 to 0, modified
		4584	Table 9.10-3 FD1_CTL_CHACT Setting for DIM Parameter: Note, added
		SECTION 10 ELECTRICAL CHARACTERISTICS	
		10.1 Electrical Characteristics	
		4669	Table 10.1-1 Absolute Maximum Ratings The Max value of Item: USB30_USVDD18, modified Notes 3 and 5, modified
		4675	Figure 10.1-2 Power-On Sequence (CM33 Boot): Note, modified (The clock settling time → The clock stabilization time)
		4677	Figure 10.1-3 DDR Retention Enter Sequence (CM33 Boot Mode): Note, modified (The clock settling time → The clock stabilization time)
		4679	Figure 10.1-4 DDR Retention Exit Sequence (CM33 Boot Mode): Note, modified (The clock settling time → The clock stabilization time)
		4681	Figure 10.1-5 Power-Off Sequence (CM33 Boot Mode): Note, modified (The clock settling time → The clock stabilization time)
		4684	Figure 10.1-7 Power-On Sequence (CA55 Boot Mode): Note, modified (The clock settling time → The clock stabilization time)
		4688	Figure 10.1-9 DDR Retention Exit Sequence (CA55 Boot Mode): Note, modified (The clock settling time → The clock stabilization time)
		4690	Figure 10.1-10 Power-Off Sequence (CA55 Boot Mode): Note, modified (The clock settling time → The clock stabilization time)
		4701	10.1.5.5 DMAC Timing, added
		4707	10.1.5.9 Ethernet Interface Timing: Conditions, modified
		4707	Table 10.1-17 Ethernet Interface Timing (n = 0, 1) The Min value of ETn_TXD0 to ETn_TXD3, ETn_TXCTL_TXEN, and ETn_TXER output delay time in Parameter: Ethernet(MII), modified Note 1, added

Rev.	Date	Description	
		Page	Summary
1.10	Jun 30, 2025	4711	Table 10.1-18 xSPI Timing (2/2) The Min value in Parameter: DS low to CS high, modified The Max value in Parameter: CS low to DS low, modified Parameter: CK low to DS low, added Notes 5 to 9, added
		4713	Figure 10.1-29 DS to CS Signal Timing, modified
		4714	10.1.5.14 Serial Communications Interface (RSCI) Access Timing: Drive strength, modified
		4715	Table 10.1-19 RSCI Timing (2/2) Parameter: SCK clock rise/fall time, deleted Parameter: Input clock rise time, Input clock fall time, Output clock rise time, and Output clock fall time, added
		4716	Figure 10.1-33 RSCI Simple SPI Mode Clock Timing, modified
		4719	Table 10.1-20 RSPI Timing Parameter: SSL Activation to Data Output Delay, added Notes 1 to 4, modified
		4720	Figure 10.1-38 RSPI Clock Timing, modified
		4720	Figure 10.1-39 RSPI Timing (Master, Motorola RSPI, CPHA = 0), modified
		4726	Table 10.1-22 I3C Timing: Note 1, modified
		4727	Table 10.1-23 IIC Timing (Fast-mode+): Note 1, modified
		4728	Table 10.1-24 IIC Timing (HS mode) The Min value in Parameter: SCL3n cycle time, modified Note 1, modified
		4730	Table 10.1-25 I3C Timing (Open Drain Timing Parameters) The Min value and Notes in Parameter: SCL3n clock High period, modified Notes in Parameter: Clock after START (S) condition, modified The Min value in Parameter: Clock before STOP (P) condition, modified Notes in Parameter: Bus available condition, modified Notes 3 and 4, added
		4731	Table 10.1-26 I3C Timing (Push-Pull Timing Parameters for SDR) Notes in Parameter: SCL3n clock Low period, modified Parameter: SCL3n clock High period for Mixed Bus, added Notes in Parameter: SDA3n signal data hold in push-pull mode, modified Note 3, added
		4740	10.1.5.20 SSIU Timing: Condition, modified
		4740	Table 10.1-29 SSIU Signal Timing: Note 2, added
		4743	10.1.5.21 PDM Timing: Condition, modified
		4743	Table 10.1-30 PDM Interface Timing: Note 2, added
		4752	10.1.7 Oscillation Circuits for Connecting Crystal Resonators (OSC): The main text, modified
		4752	Table 10.1-38 Pins for Connecting Crystal Resonators and Clock Frequency: For the emergency clock, deleted

RZ/G3E Group User's Manual: Hardware

Publication Date: Rev.1.00 Oct 31, 2024
Rev.1.10 Jun 30, 2025

Published by: Renesas Electronics Corporation

RZ/G3E Group

Renesas Electronics Corporation

R01UH1069EJ0110