

nRF54L15, nRF54L10, and nRF54L05

Wireless SoCs

nRF54L15, nRF54L10, and nRF54L05 make up the nRF54L Series. All wireless System-on-Chip (SoC) options in the series integrate an ultra-low power multiprotocol 2.4 GHz radio and MCU (Microcontroller Unit) functionality featuring a 128 MHz Arm Cortex-M33 processor, comprehensive peripheral set, and scalable memory configurations up to 1.5 MB NVM and up to 256 KB RAM.

Ultra-low power consumption is enabled with Nordic proprietary technologies such as low-leakage RAM and design expertise utilized in the advanced multiprotocol radio. With lower power consumption, each wireless SoC in the nRF54L Series enables improved battery lifetimes or reduced battery size.

Designed with versatility in mind, the nRF54L Series SoCs are suited to enable a broad range of applications. The multiprotocol 2.4 GHz radio supports the latest Bluetooth[®] 6.0 features including Bluetooth Channel Sounding, as well as 802.15.4-2020 for standards such as Thread, Matter, and Zigbee, and a proprietary 2.4 GHz mode supporting up to 4 Mbps for higher throughput. The devices integrate the peripherals expected in a wireless microcontroller enabling many products to be implemented with a single chip. An integrated RISC-V coprocessor further reduces the need for external ICs.

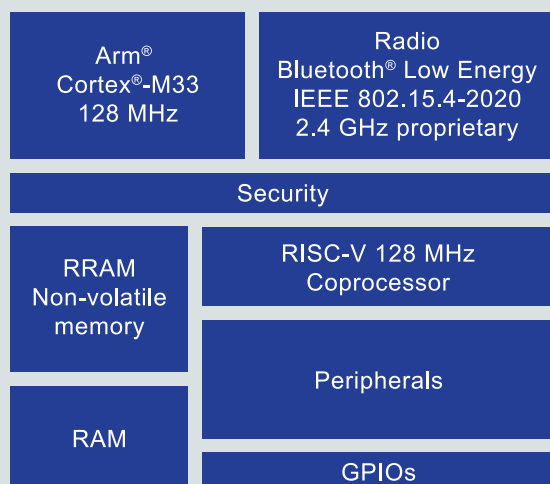
nRF54L Series wireless SoCs are available in a range of memory and package configurations, including pin-to-pin compatible options. With several memory options, finding the right device to fit an application optimizes cost and flexibility in design.

Key features

- 128 MHz Arm[®] Cortex[®]-M33 processor
- Scalable memory configurations up to 1.5 MB NVM and up to 256 KB RAM
- Multiprotocol 2.4 GHz radio supporting Bluetooth Low Energy, 802.15.4-2020, and 2.4 GHz proprietary modes (up to 4 Mbps)
- Comprehensive set of peripherals including new Global RTC available in System OFF, 14-bit ADC, and high-speed serial interfaces
- 128 MHz RISC-V coprocessor
- Advanced security including TrustZone[®] isolation, tamper detection, and cryptographic engine side-channel leakage protection
- Ultra-compact WLCSP (2.4x2.2 mm) and QFN (6.0x6.0 mm) packages



nRF54L15 | nRF54L10 | nRF54L05



Power consumption highlights

Power mode	Current @ 3.0 V
Active with radio	
Bluetooth LE TX 1 Mbps at 0 dBm	5.0 mA
Bluetooth LE TX 1 Mbps at +4 dBm	6.8 mA
Bluetooth LE TX 1 Mbps at +8 dBm	10.0 mA
Bluetooth LE RX 1 Mbps	3.2 mA
Active with processing	
CPU CoreMark from RRAM with cache	2.4 mA
Sleep	
System ON IDLE with GRTC (XOSC) and 256 KB RAM	3.0 µA
System ON IDLE with GRTC (XOSC) and 192 KB RAM	2.6 µA
System ON IDLE with GRTC (XOSC) and 96 KB RAM	2.0 µA
System OFF with GRTC wakeup	0.8 µA
System OFF	0.6 µA

Product variants

Part number	NVM	RAM
nRF54L15	1.5 MB	256 KB
nRF54L10	1.0 MB	192 KB
nRF54L05	0.5 MB	96 KB

Key features

Features

Multiprotocol radio

- *Bluetooth*[®] 6.0, IEEE 802.15.4-2020, and 2.4 GHz enabled transceiver
 - -96 dBm sensitivity in 1 Mbps Bluetooth Low Energy mode, 0.1% bit error rate
 - -104 dBm sensitivity in 125 kbps Bluetooth Low Energy mode (long range) with a 0.1% bit error rate
 - -102 dBm typical sensitivity in IEEE 802.15.4
 - Up to +8 dBm configurable output power; 1 dB step size from -8 dBm to +8 dBm
 - Supported data rates:
 - Bluetooth 6.0 – 2 Mbps, 1 Mbps, 500 kbps, and 125 kbps
 - IEEE 802.15.4-2020 – 250 kbps
 - Proprietary 2.4 GHz – 4 Mbps, 2 Mbps, and 1 Mbps
 - Single-ended antenna output (on-chip balun)
 - 128-bit AES/ECB/CCM/AAR coprocessor (on-the-fly operation)
 - RSSI (1 dB resolution)

Platform security

- Secure/non-secure memory protection
- Symmetric and asymmetric key crypto accelerator
- Secure key management
- Tamper detection
- Immutable boot partition
- Debug access port protection
- Two watchdog timers for secure and non-secure access

Memory

- nRF54L15 – 1524 KB non-volatile memory (RRAM) and 256 KB RAM
- nRF54L10 – 1022 KB non-volatile memory (RRAM) and 192 KB RAM
- nRF54L05 – 500 KB non-volatile memory (RRAM) and 96 KB RAM

Operating values

- 1.7 V to 3.6 V supply and I/O voltage
- Single 32 MHz crystal operation
- Optional 32.768 kHz clock
- Operating temperature from -40°C to 105°C

Arm[®] Cortex[®]-M33 with TrustZone[®] technology, 128 MHz

- 500 EEMBC CoreMark[®] score running from non-volatile memory, 3.90 CoreMark per MHz
- Single-precision floating-point unit (FPU)
- Memory protection unit (MPU)
- Digital signal processing (DSP) instructions
- Data watchpoint and trace (DWT), embedded trace macrocell (ETM), instrumentation trace macrocell (ITM), and cross trigger interface (CTI)
- Serial wire debug (SWD)
- Trace port interface unit (TPIU)
 - 4-bit parallel trace of ITM and ETM trace data
 - Serial wire output (SWO) trace of ITM data

Peripherals

- RISC-V Coprocessor
- Global RTC (GRTC) that can run in System OFF mode and implement a shared system timer
- Seven 32-bit timers with counter mode
- Up to five fully featured serial interfaces with EasyDMA, supporting I²C, SPI controller/peripheral, and UART
 - One high-speed SPIM up to 32 MHz, four up to 8 MHz
 - One high-speed UARTE up to 4 Mbps, four up to 1 Mbps
 - I²C up to 400 kHz
- Three pulse width modulator (PWM) units with EasyDMA
- I²S two channel Inter-IC sound interface
- ADC with up to eight programmable gain channels. 14-bit at 31.25 kbps, 12-bit at 250 kbps, and up to 10-bit at 2 Msps.
- Pulse density modulation (PDM) interface
- Near field communication (NFC)
- Up to two quadrature decoders (QDEC)
- Comparator and low-power comparator with wake-up from System OFF mode
- Temperature sensor

Package variants

- QFN48 6.0x6.0 mm with 31 GPIO pins
- WLCSP 2.4x2.2 mm with 32 GPIO pins
 - 300 µm pitch

Contents

	Key features.	ii
1	Revision history.	12
2	About this document.	13
	2.1 Document status.	13
	2.2 Peripheral chapters.	13
	2.2.1 Peripheral naming conventions.	13
3	Product overview.	15
	3.1 Block diagram.	15
	3.2 Memory and package overview.	16
	3.3 Power domains.	17
	3.4 Address format.	17
	3.5 Memory.	19
	3.5.1 RAM — Random access memory.	19
	3.5.2 NVM — Non-volatile memory.	19
	3.5.3 Memory map.	20
	3.5.4 Instantiation.	21
4	Application core.	26
	4.1 Arm Cortex-M33 CPU.	26
	4.1.1 CPU.	26
	4.1.2 CPUC — CPU control.	27
	4.1.3 Arm Cortex-M33 Peripherals.	33
	4.2 Core components.	33
	4.2.1 AMBA interconnect (AMBIX).	33
	4.2.2 EasyDMA.	34
	4.2.3 CACHE — Instruction/data cache.	36
	4.2.4 FICR — Factory information configuration registers.	44
	4.2.5 MEMCONF — Memory configuration.	50
	4.2.6 RRAMC — Resistive random access memory controller.	53
	4.2.7 SICR — Secure information configuration region.	66
	4.2.8 SWI — Software interrupts.	66
	4.2.9 UICR — User information configuration registers.	66
5	Power and clock management.	73
	5.1 System ON mode.	73
	5.1.1 Sub-power modes.	73
	5.2 System OFF mode.	74
	5.2.1 Emulated System OFF mode.	74
	5.3 Power supply supervisors.	75
	5.3.1 Power-fail comparator.	75
	5.4 CLOCK — Clock control.	76
	5.4.1 HFCLK controller.	77
	5.4.2 LFCLK controller.	78
	5.4.3 Registers.	79
	5.5 OSCILLATORS — Oscillator control.	91
	5.5.1 High-frequency (32 MHz) crystal oscillator (HFXO).	91

5.5.2 Low-frequency (32.768 kHz) crystal oscillator (LFXO).	94
5.5.3 CPU clock frequency selection.	95
5.5.4 Registers.	96
5.6 POWER — Power control.	97
5.6.1 Registers.	98
5.7 REGULATORS — Regulator control.	102
5.7.1 VREGMAIN — Main regulator.	103
5.7.2 Registers.	103
5.8 RESET — Reset control.	106
5.8.1 Power-on reset.	106
5.8.2 Pin reset.	106
5.8.3 Brownout reset.	107
5.8.4 Glitch detector.	107
5.8.5 Wakeup from System OFF mode reset.	107
5.8.6 Soft reset.	107
5.8.7 CTRL-AP resets.	107
5.8.8 Watchdog timer reset.	107
5.8.9 Retained registers.	108
5.8.10 Reset behavior.	108
5.8.11 Registers.	109
6 Event system.	111
6.1 DPPI latencies.	111
6.2 DPPI — Distributed programmable peripheral interconnect.	112
6.2.1 Subscribing to and publishing on channels.	113
6.2.2 DPPI controller (DPPIC).	115
6.2.3 Connection examples.	115
6.2.4 Split security.	116
6.2.5 Registers.	117
6.3 PPIB — PPI Bridge.	120
6.3.1 PPIB connections.	120
6.3.2 Handshake and overflow.	121
6.3.3 Connection examples.	121
6.3.4 Ownership and security attributes.	122
6.3.5 Registers.	123
7 Security.	126
7.1 Memory and peripheral access permissions.	126
7.2 TrustZone security.	128
7.3 Immutable boot region.	131
7.4 Security attributes.	131
7.5 Peripherals with split security access.	131
7.5.1 CRACEN.	132
7.5.2 DPPIC.	132
7.5.3 GPIO.	133
7.5.4 GPIOTE.	134
7.5.5 GRTC.	135
7.6 Physical security.	135
7.7 Security components.	136
7.7.1 CRACEN — Cryptographic accelerator engine.	136
7.7.2 GLITCHDET — Voltage glitch detectors.	165
7.7.3 KMU — Key management unit.	166
7.7.4 MPC — Memory Privilege Controller.	175

7.7.5 SPU — System protection unit.	180
7.7.6 TAMPC — Tamper controller.	192
8 Peripherals.	213
8.1 Peripheral interface.	213
8.1.2 Peripheral ID.	214
8.1.3 Peripherals with shared ID.	214
8.1.4 Peripheral registers.	214
8.1.5 Bit set and clear.	215
8.1.6 Tasks.	215
8.1.7 Events.	215
8.1.8 Publish and subscribe.	215
8.1.9 Shortcuts.	215
8.1.10 Interrupts.	216
8.2 Instantiation.	216
8.3 AAR — Accelerated address resolver.	220
8.3.1 Shared resources.	220
8.3.2 Resolving a resolvable address.	221
8.3.3 EasyDMA.	222
8.3.4 Use case example for chaining RADIO packet reception with address resolution using AAR.	224
8.3.5 Registers.	225
8.4 CCM — AES CCM mode encryption.	232
8.4.1 Shared resources.	233
8.4.2 Encryption and decryption.	233
8.4.3 Encrypting packets in radio transmit mode.	236
8.4.4 Decrypting packets received by the radio.	237
8.4.5 CCM data structure.	238
8.4.6 EasyDMA and ERROR event.	239
8.4.7 Registers.	239
8.5 COMP — Comparator.	247
8.5.1 START and STOP tasks.	248
8.5.2 Operation modes.	248
8.5.3 Current source on analog input.	249
8.5.4 Shared resources.	249
8.5.5 Differential mode.	249
8.5.6 Single-ended mode.	251
8.5.7 Registers.	253
8.6 ECB — AES electronic codebook mode encryption.	262
8.6.1 Shared resources.	264
8.6.2 EasyDMA.	264
8.6.3 Registers.	265
8.7 EGU — Event generator unit.	270
8.7.1 Registers.	270
8.8 GPIO — General purpose input/output.	273
8.8.1 Pin configuration.	274
8.8.2 Pin sense mechanism.	274
8.8.3 Port capabilities.	275
8.8.4 Peripheral and subsystem assignment.	276
8.8.5 Clock pins.	277
8.8.6 Registers.	278
8.9 GPIOTE — GPIO tasks and events.	282
8.9.1 Pin events and tasks.	283
8.9.2 Port event.	284

8.9.3 Tasks and events pin configuration.	284
8.9.4 Split security attribute.	284
8.9.5 Registers.	285
8.10 GRTC — Global real-time counter.	291
8.10.1 GRTC clock sources.	292
8.10.2 SYSCOUNTER.	292
8.10.3 Pulse Width Modulation (PWM).	295
8.10.4 Clock output.	296
8.10.5 Split Security.	296
8.10.6 Registers.	297
8.11 I ² S — Inter-IC sound interface.	313
8.11.1 Mode.	314
8.11.2 Transmitting and receiving.	314
8.11.3 Left right clock (LRCK).	315
8.11.4 Serial clock (SCK).	316
8.11.5 Master clock (MCK).	316
8.11.6 Width, alignment and format.	318
8.11.7 EasyDMA.	320
8.11.8 Module operation.	324
8.11.9 Pin configuration.	326
8.11.10 Registers.	326
8.12 LPCOMP — Low-power comparator.	339
8.12.1 Shared resources.	341
8.12.2 Pin configuration.	341
8.12.3 Registers.	341
8.13 NFCT — Near field communication tag.	350
8.13.1 Overview.	351
8.13.2 Operating states.	353
8.13.3 Pin configuration.	354
8.13.4 EasyDMA.	355
8.13.5 Frame assembler.	355
8.13.6 Frame disassembler.	357
8.13.7 Frame timing controller.	358
8.13.8 Collision resolution.	359
8.13.9 Antenna interface.	360
8.13.10 NFCT antenna recommendations.	360
8.13.11 Battery protection.	361
8.13.12 Digital Modulation Signal.	361
8.13.13 References.	362
8.13.14 Registers.	362
8.14 PDM — Pulse density modulation interface.	388
8.14.1 Master clock generator.	389
8.14.2 Module operation.	389
8.14.3 Decimation filter.	390
8.14.4 EasyDMA.	390
8.14.5 Hardware example.	391
8.14.6 Pin configuration.	393
8.14.7 Registers.	394
8.15 PWM — Pulse width modulation.	405
8.15.1 Wave counter.	406
8.15.2 Decoder with EasyDMA.	409
8.15.3 Limitations.	416
8.15.4 Pin configuration.	416
8.15.5 Registers.	416

8.16 QDEC — Quadrature decoder.	436
8.16.1 Sampling and decoding.	437
8.16.2 LED output.	438
8.16.3 Debounce filters.	438
8.16.4 Accumulators.	438
8.16.5 Output/input pins.	439
8.16.6 Pin configuration.	439
8.16.7 Registers.	440
8.17 RADIO — 2.4 GHz radio.	454
8.17.1 Packet configuration.	455
8.17.2 Address configuration.	456
8.17.3 Data whitening.	457
8.17.4 CRC.	457
8.17.5 Radio states.	458
8.17.6 Transmit sequence.	459
8.17.7 Receive sequence.	461
8.17.8 Received signal strength indicator (RSSI).	462
8.17.9 Interframe spacing (IFS).	463
8.17.10 Device address match.	463
8.17.11 Bit counter.	464
8.17.12 IEEE 802.15.4 operation.	464
8.17.13 EasyDMA.	474
8.17.14 Registers.	475
8.18 SAADC — Successive approximation analog-to-digital converter.	532
8.18.1 Shared resources.	533
8.18.2 Overview.	533
8.18.3 Digital output.	534
8.18.4 Analog inputs and channels.	534
8.18.5 Operation modes.	535
8.18.6 EasyDMA.	536
8.18.7 Reference.	537
8.18.8 Acquisition time.	538
8.18.9 Limits event monitoring.	538
8.18.10 Performance factors.	539
8.18.11 Registers.	540
8.19 SPIM — Serial peripheral interface controller with EasyDMA.	558
8.19.1 SPIM transaction sequence.	559
8.19.2 Pin configuration.	560
8.19.3 Shared resources.	561
8.19.4 EasyDMA.	561
8.19.5 Low power.	562
8.19.6 Timing diagrams.	562
8.19.7 Registers.	563
8.20 SPIS — Serial peripheral interface target with EasyDMA.	585
8.20.1 SPI modes.	586
8.20.2 Shared resources.	586
8.20.3 EasyDMA.	587
8.20.4 SPIS operation.	587
8.20.5 Semaphore operation.	589
8.20.6 Pin configuration.	590
8.20.7 Registers.	591
8.21 TEMP — Temperature sensor.	609
8.21.1 Registers.	610
8.22 TIMER — Timer/counter.	618

8.22.1 Capture.	619
8.22.2 Compare.	619
8.22.3 Task delays.	619
8.22.4 Task priority.	619
8.22.5 Registers.	620
8.23 TWIM — I ² C compatible two-wire interface master with EasyDMA.	627
8.23.1 Shared resources.	629
8.23.2 EasyDMA.	629
8.23.3 Master write sequence.	629
8.23.4 Master read sequence.	630
8.23.5 Master repeated start sequence.	631
8.23.6 Low power.	632
8.23.7 Master mode pin configuration.	632
8.23.8 Registers.	633
8.23.9 Pullup resistor.	657
8.24 TWIS — I ² C compatible two-wire interface target with EasyDMA.	657
8.24.1 State machine.	658
8.24.2 Shared resources.	659
8.24.3 EasyDMA.	659
8.24.4 Read command response.	660
8.24.5 Write command response.	661
8.24.6 TWI controller repeated start sequence.	662
8.24.7 Terminate an ongoing TWI transaction.	663
8.24.8 Low power.	663
8.24.9 Target mode pin configuration.	663
8.24.10 Registers.	664
8.25 UARTE — Universal asynchronous receiver/transmitter with EasyDMA.	686
8.25.1 EasyDMA.	687
8.25.2 Transmission.	687
8.25.3 Reception.	688
8.25.4 Data frame size.	690
8.25.5 Frame timeout interrupt.	691
8.25.6 Error conditions.	692
8.25.7 Using the UARTE without flow control.	692
8.25.8 Parity and stop bit configuration.	692
8.25.9 Compare match filter.	692
8.25.10 Low power.	693
8.25.11 Pin configuration.	693
8.25.12 Registers.	694
8.26 VPR — RISC-V CPU.	721
8.26.1 Registers.	722
8.26.2 Registers.	738
8.27 WDT — Watchdog timer.	740
8.27.1 Reload criteria.	741
8.27.2 Temporarily pausing the watchdog.	741
8.27.3 Watchdog reset.	741
8.27.4 Stopping the watchdog.	741
8.27.5 Registers.	742

9 Debug and trace.	749
9.1 Debug access port.	749
9.1.1 Registers.	750
9.2 Access port protection.	751
9.3 Debug interface mode.	753

9.4 Real-time debug.	754
9.5 Multidrop serial wire debug.	754
9.6 Trace.	754
9.6.1 Enabling the trace port.	754
9.7 CTRL-AP - Control access port.	755
9.7.1 Reset request.	757
9.7.2 Erase all.	757
9.7.3 Mailbox interface.	759
9.7.4 Device information.	760
9.7.5 Debugger registers.	760
9.7.6 Registers.	765
9.8 TAD - Trace and debug control.	769
9.8.1 Registers.	769
9.9 ETM — Embedded trace macrocell.	771
9.9.1 Registers.	771
9.10 TPIU — Trace port interface unit.	792
9.10.1 Registers.	792
10 Hardware and layout.	793
10.1 Pin assignments.	793
10.1.1 Dedicated pins.	793
10.1.2 Clock pins.	794
10.1.3 QFN48 pin assignments.	795
10.1.4 WLCSP300 pin assignments.	801
10.2 Mechanical specifications.	806
10.2.1 QFN48 6x6 mm package.	806
10.2.2 WLCSP 300 µm pitch package.	807
10.3 Reference circuitry.	808
10.3.1 Circuit configuration 1 for QFAA QFN-48.	808
10.3.2 Circuit configuration 1 for CAAA WLCSP 300 µm pitch.	810
10.3.3 PCB layout example.	812
10.3.4 PMIC support.	813
10.4 Reflow conditions.	813
11 Electrical specification.	814
11.1 Current consumption.	814
11.1.1 Conditions.	814
11.1.2 CURRENT Electrical specification.	815
11.2 CLOCK Electrical specification.	816
11.2.1 High frequency clock source (HFCLK).	816
11.2.2 32.768 kHz clock source (LFCLK).	816
11.3 COMP Electrical specification.	816
11.3.1 COMP Electrical Specification.	816
11.4 CPU Electrical specification.	816
11.4.1 CPU performance.	816
11.4.2 CPU wakeup times.	817
11.5 GPIO Electrical specification.	817
11.5.1 GPIO Electrical Specification.	817
11.6 I2S Electrical specification.	818
11.6.1 I2S timing specification.	818
11.7 LPCOMP Electrical specification.	819
11.7.1 LPCOMP Electrical Specification.	819
11.8 NFCT Electrical specification.	819

11.8.1 NFCT Electrical Specification.	819
11.8.2 NFCT Timing Parameters.	819
11.9 OSCILLATORS Electrical specification.	820
11.9.1 32 MHz crystal oscillator (HFXO).	820
11.9.2 32.768 kHz crystal oscillator (LFXO).	820
11.10 PPI Electrical specification.	820
11.10.1 Typical PPI latencies.	820
11.11 PDM Electrical specification.	820
11.11.1 PDM Electrical Specification.	820
11.12 QDEC Electrical specification.	821
11.12.1 QDEC Electrical Specification.	821
11.13 RADIO Electrical specification.	821
11.13.1 General radio characteristics.	821
11.13.2 Radio current consumption (transmitter).	822
11.13.3 Radio current consumption (Receiver).	822
11.13.4 Transmitter specification.	822
11.13.5 Receiver operation.	824
11.13.6 RX selectivity.	824
11.13.7 RX intermodulation.	826
11.13.8 Radio timing.	826
11.13.9 Received signal strength indicator (RSSI) specifications.	827
11.13.10 Jitter.	827
11.13.11 IEEE 802.15.4 mode energy detection constants.	827
11.14 REGULATORS Electrical specification.	827
11.14.1 Recommended operating conditions.	827
11.14.2 Power-fail comparator.	827
11.15 RESET Electrical specification.	828
11.15.1 Startup times.	828
11.16 RRAMC Electrical specification.	828
11.16.1 RRAM programming.	828
11.17 SAADC Electrical specification.	828
11.17.1 SAADC Electrical Specification.	828
11.18 SPIM Electrical specification.	828
11.18.1 Timing specifications.	828
11.19 SPIS Electrical specification.	830
11.19.1 SPIS slave interface electrical specifications.	830
11.19.2 Serial Peripheral Interface Slave (SPIS) timing specifications.	830
11.20 SWDP Electrical specification.	831
11.20.1 SW-DP.	831
11.20.2 Trace port.	832
11.21 TEMP Electrical specification.	832
11.21.1 Temperature Sensor Electrical Specification.	832
11.22 TWIM Electrical specification.	832
11.22.1 TWIM interface electrical specifications.	832
11.22.2 Two Wire Interface Master (TWIM) timing specifications.	832
11.23 TWIS Electrical specification.	833
11.23.1 TWIS slave timing specifications.	833
11.24 UARTE Electrical specification.	834
11.24.1 UARTE electrical specification.	834
11.25 WDT Electrical specification.	834
11.25.1 Watchdog Timer Electrical Specification.	834

12 Recommended operating conditions.	835
12.1 Extended Operating Temperature.	835

12.2 WLCSP light sensitivity.	835
13 Absolute maximum ratings.	836
14 Ordering information.	838
14.1 Device marking.	838
14.2 Box labels.	838
14.3 Order code.	839
14.4 Code ranges and values.	840
14.5 Product options.	842
14.6 Preprogrammed firmware.	842
15 Legal notices.	844

1 Revision history

Date	Version	Description
December 2024	0.8	<p>The following content has been updated:</p> <ul style="list-style-type: none"> • Absolute maximum ratings • CACHE – Fixed offset addresses for registers LINEADDR, PROFILING, DEBUGLOCK, and WRITELOCK • Clock – Removed registers SUBSCRIBE_XOTUNE, SUBSCRIBE_XOTUNEABORT, PUBLISH_XOSTARTED, PUBLISH_PLLSTARTED, PUBLISH_LFCLKSTARTED, PUBLISH_DONE, PUBLISH_XOTUNED, PUBLISH_XOTUNEERROR, PUBLISH_XOTUNEFAILED, and SHORTS • ECB – REFLECTOR/INITIATOR registers swapped • GRTC – WAKETIME/TIMEOUT • Order codes – Added nRF54L10-QFAA-R and nRF54L05-QFAA-R ordering codes • PDN – MOD.EDGE register selection, enumerators LeftFalling and LeftRising swapped. FDBYPASS register removed. • Power – Removed register PUBLISH_POFWARN • Recommended operating conditions • RRAMC – Buffered writes must be in sequential ascending address order • OSCILLATORS – Crystal selection chapter added • SAADC – Internal reference voltage is 0.9 V • TPIU – Updated chapter text to point to Arm Cortex-M33 TPIU • TWIM and TWIS – Removed clock pin requirement for SDA signal • PSEL.PORT min/max values aligned with existing Nordic devices (using oversized value range) • Electrical specifications • Editorial
October 2024	0.7	Preliminary release

2 About this document

This document is organized into chapters that are based on the modules and peripherals available in the IC.

2.1 Document status

The document status reflects the level of maturity of the document.

Document name	Description
Preliminary Datasheet	Applies to document versions up to 1.0. This document contains target specifications for product development.
Datasheet	Applies to document versions 1.0 and higher. This document contains final product specifications. Nordic Semiconductor ASA reserves the right to make changes at any time without notice in order to improve design and supply the best possible product.

Table 1: Defined document names

2.2 Peripheral chapters

The chapters describing peripherals include the following information:

- A description of the peripheral.
- The electrical specification tables, containing performance data which applies for the operating conditions described in [Recommended operating conditions](#) on page 835.

2.2.1 Peripheral naming conventions

Every peripheral has a unique capitalized name or an abbreviation of its name, such as TIMER, that is used for identification and reference.

This name is used in chapter headings and references, and it will appear in the Arm Cortex Microcontroller Software Interface Standard (CMSIS) hardware abstraction layer to identify the peripheral.

When there is more than one instance of a peripheral in a power domain, a two digit number D_n is added as a suffix to the peripheral name when constructing the peripheral instance name. For example, a peripheral named PERI with instance name "PERID_n" is located in power domain D, and is instance number n in that domain. For a list of power domains, see [Power domains](#) on page 17.

The following are additional examples of peripheral instance names:

- PPIB00 is in the MCU domain (0), and is the first PPIB instance in the MCU domain (0).
- SPIS21 is in the PERI domain (2), and is the second SPIS instance of the PERI domain (1).

The peripheral instance name is also used in the CMSIS to identify the peripheral instance.

The domain digits x are listed in the following table.

Domain digit	Power domain
0	MCU
1	RADIO
2	PERI
3	LP

Table 2: Domain digit overview

3 Product overview

This document is applicable for the nRF54L15, nRF54L10, and nRF54L05 System on Chip devices. The main differences are memory, GPIO pin count, and package options, which are detailed in their respective sections.

The device is an ultra-low power System on Chip (SoC) with advanced security features, a range of peripherals, and a multiprotocol 2.4 GHz transceiver. It supports Bluetooth Low Energy, IEEE 802.15.4 for Thread and Zigbee protocols, and allows for the implementation of proprietary 2.4 GHz protocols.

The main processing unit is an Arm Cortex-M33 processor running at up to 128 MHz, supported by non-volatile RRAM and RAM memory.

The Arm Cortex-M33 has a full set of digital signal processing (DSP) instructions and a memory protection unit (MPU) for application security. The full-featured single-precision floating-point unit (FPU) supports all single-precision instructions.

The peripheral set offers a variety of analog and digital functionality enabling single-chip implementation of a wide range of applications.

Hardware isolation between the secure and non-secure resources, as defined by Arm TrustZone, is implemented in the device. The hardware peripherals can be configured as secure or non-secure.

A key management unit (KMU) provides key storage, that when combined with a cryptographic accelerator (CRACEN), ensures discretion of encryption keys even within the secure world. The cryptographic accelerator has protection against differential power analysis (DPA) attacks.

The device has measures to protect against physical security attacks. It can detect and report fault injection attacks such as voltage glitching or electromagnetic fault injection. An external active shield I/O interface provides PCB or product level security for the detection of a product's encapsulation being opened, or product tampering.

The device non-volatile memory has a boot region that can be made immutable before the CPU starts up. Boot initiated from an immutable source allows subsequent boot steps to be performed by authenticated code.

The debug access port can be enabled or disabled to allow either non-intrusive or intrusive debugging, from secure- or non-secure worlds. The non-volatile memory can be protected against erasing, providing protection from unauthenticated repurposing. Authenticated debug access control, such as facilitating the Arm ADAC architecture, is supported through a hardware mailbox. The mailbox allows on-chip firmware to authenticate the debug host before enabling the device debug interface.

The device has a dedicated RISC-V CPU (VPR), which is a fast, lightweight peripheral processor (FLPR) dedicated for software defined peripherals.

3.1 Block diagram

The block diagram illustrates the overall system.

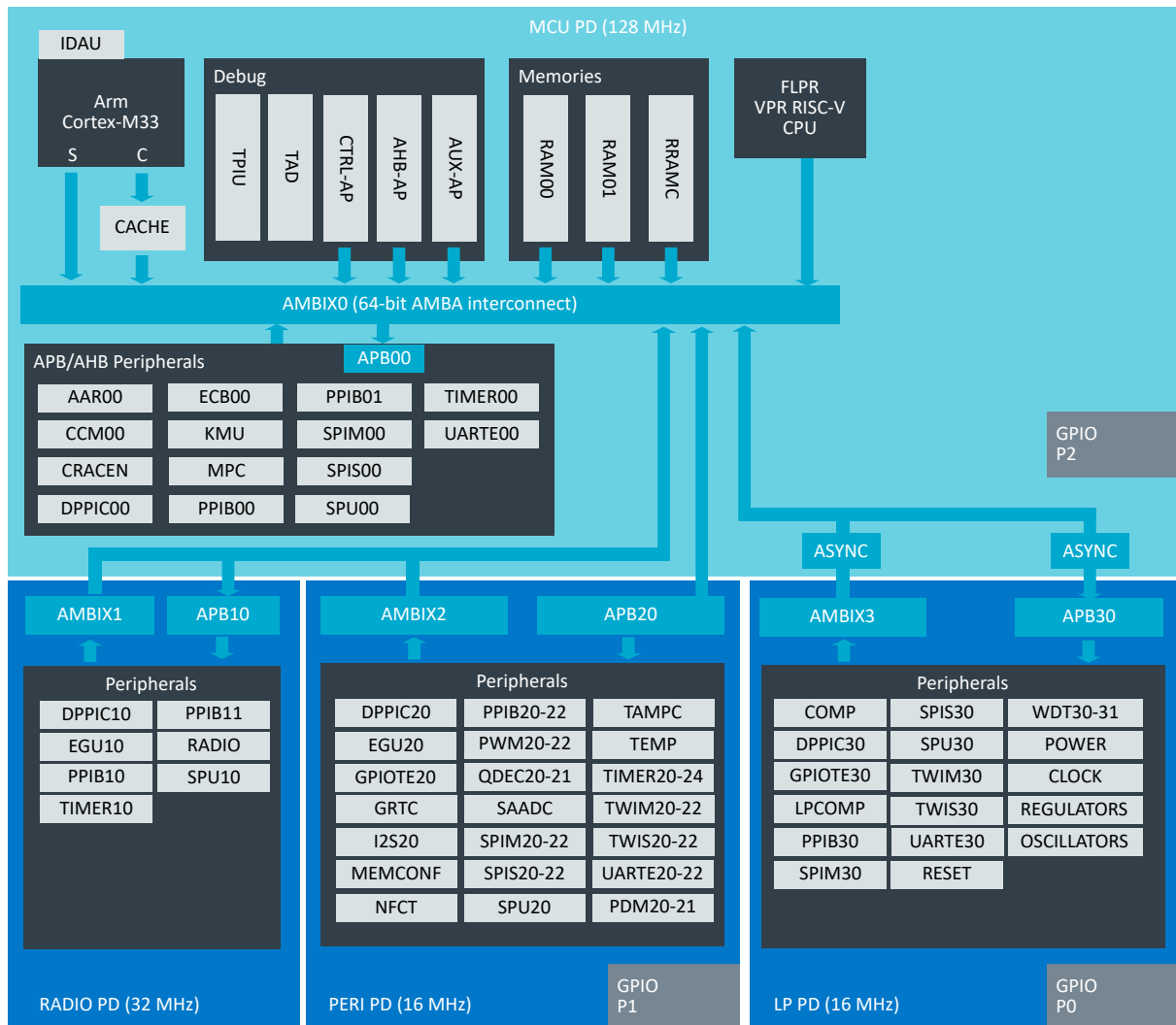


Figure 1: Block diagram

3.2 Memory and package overview

Memory	Device		
	nRF54L15	nRF54L10	nRF54L05
Non-volatile memory (RRAM)	1524 KB	1022 KB	500 KB
Random access memory (RAM)	256 KB	192 KB	96 KB

Table 3: Device memory options

		Package	
Feature		QFN48	CSP 300 µm
Pins	GPIO pins	31	32
	Wakeup-pins	20	21
	Analog input pins	8	8
Security	Active tamper shield pin pairs (in/out)	4	4
Debug	ITM parallel trace	Yes	No

Table 4: Package options

3.3 Power domains

Multiple power domains ensure low-power operation.

The MCU domain contains an Arm Cortex-M33. The CPU is connected to a debug system, allowing debug and ETM trace. The CPU executes program code from RRAM through an instruction cache. Data is stored in single-cycle RAM that is divided into multiple bus subordinates, but it forms a continuous RAM space in the memory map. High-speed peripherals are also found in the MCU domain.

There are three additional domains that have peripherals allocated to them. They are the following:

- Radio domain – Contains the short-range radio and supporting peripherals used by the radio protocol stack. It runs at 32 MHz synchronously with the MCU domain.
- Peripheral domain – Contains most peripherals. It runs at 16 MHz synchronously with the MCU domain.
- Low-power domain – Contains peripherals for ultra-low power modes and can be used to wake the rest of the system even when the peripheral domain is powered off. It runs at 16 MHz asynchronously to the MCU domain.

Each domain is mapped to one APB bus and can be powered independently. EasyDMA traffic from each domain is aggregated in a local AMBIX interconnect and can access RAM in the MCU domain.

Three of the power domains have their own GPIO ports. GPIO pins can be used by peripherals in the same power domain. For exceptions, see [GPIO — General purpose input/output](#) on page 273 and [pin assignments](#).

3.4 Address format

Addresses in the system memory map follow the address format described in the following tables.

Address bits	Description	Enumeration
[28:0]	Address space	
[31:29]	Address regions	0: Program memory 1: Data memory 2: Peripherals/APB space 7: CPU internal peripherals, like Arm Cortex private peripheral bus (PPB)

Table 5: Address regions format

The program and data memory address format is described in the following table.

Address bits	Description	Enumeration
[23:0]	Address space	
[28:24]	Reserved	Set to zero
[31:29]	Address regions	0: Program memory 1: Data memory

Table 6: Program memory and data memory address format

The peripheral address format is described in the following table.

Address bits	Description	Enumeration
[11:0]	Peripheral address space	
[17:12]	Peripheral subordinate index	Used for configuring the SPU — System protection unit on page 180, as the index n of register <code>SPU.PERIPH[n].PERM</code> .
[23:18]	Peripheral APB bus number	1: APB peripherals in MCU power domain 2: APB peripherals in RADIO power domain 3: APB peripherals in PERI power domain 4: APB peripherals in LP power domain
[27:24]	Reserved	Set to zero
[28]	Security	0: Non-secure 1: Secure
[31:29]	Address regions	2: Peripherals on APB bus

Table 7: Peripheral address format

3.5 Memory

The CPU and peripherals with EasyDMA can access memory through the AMBIX interconnects. The same interconnect is also used for CPU access to peripheral registers. The following figure is a simplified interconnect diagram.

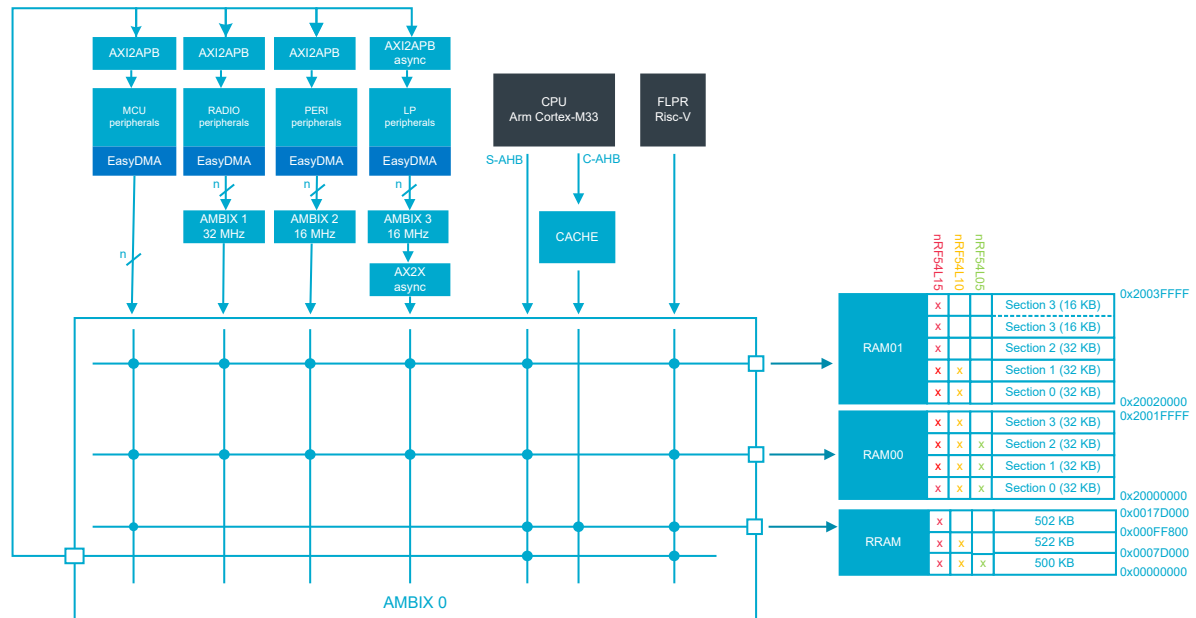


Figure 2: Memory layout

See [Block diagram](#) on page 15, [AMBA interconnect \(AMBIX\)](#) on page 33, and [EasyDMA](#) on page 34 for more information about the AMBIX interconnects and EasyDMA.

RAM and RRAM memory regions are protected with TrustZone security, and are secure after reset. Memory regions can be configured to be non-secure by using [MPC — Memory Privilege Controller](#) on page 175.

3.5.1 RAM — Random access memory

The device RAM has regions arranged in one contiguous memory range, accessible from both the CPU and peripherals.

Arm TrustZone security ensures all RAM regions are secure after reset. RAM regions can be configured as non-secure by using [MPC — Memory Privilege Controller](#) on page 175.

Each RAM region has separate power control for System ON and System OFF mode. This preserves RAM contents in sleep modes or powers off RAM to save power. The sections are illustrated in [Memory layout](#) on page 19 and the register interface is described in [MEMCONF — Memory configuration](#) on page 50.

3.5.2 NVM — Non-volatile memory

The CPU can read from non-volatile memory (RRAM) an unlimited number of times, but is restricted in how it writes to memory and the number of writes it can perform.

Writing to RRAM is managed by the RRAM controller (RRAMC), see [RRAMC — Resistive random access memory controller](#) on page 53.

Arm TrustZone security ensures the entire RRAM is secure after reset. RRAM can be configured to have multiple non-secure regions by using [MPC — Memory Privilege Controller](#) on page 175.

RRAM can be accessed by the Arm Cortex-M33 CPU via the C-AHB (code) and S-AHB (system) buses as shown in [Memory layout](#) on page 19. The code bus (C-AHB) interface is used for any instruction fetch or data access fetch to the code region of the Arm memory model. All access from C-AHB bus are cached, see [CACHE — Instruction/data cache](#) on page 36.

RRAM can also be accessed by FLPR which has a built-in cache (meaning it does not share the instruction cache with the Cortex-M33).

3.5.3 Memory map

The complete memory map is shown in the following figure.

Legend:

Configurable resource

Non-secure resource

Secure resource

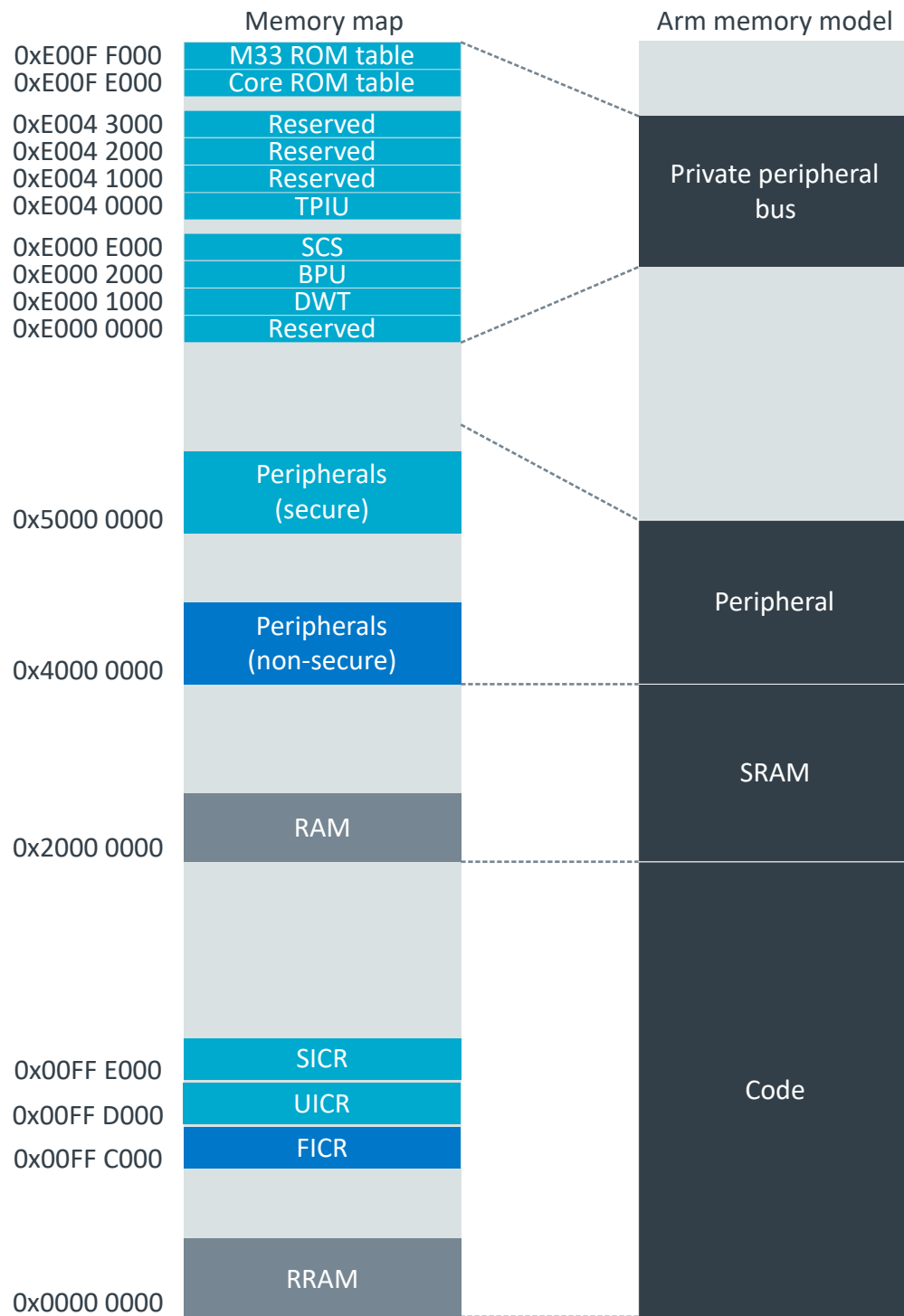


Figure 3: Memory map

3.5.4 Instantiation

ID	Base address	Instance	TrustZone			Split access	Description
			Map	Att	DMA		
64	0x50040000	SPU00	HF	S	NA	No	System protection unit SPU00

ID	Base address	Instance	TrustZone			Split access	Description
			Map	Att	DMA		
65	0x50041000	MPC00	HF	S	NA	No	Memory privilege controller MPC00
66	0x50042000	DPPIC00 : S	US	S	NA	Yes	DPPI controller DPPIC00
	0x40042000	DPPIC00 : NS					
67	0x50043000	PPIB00 : S	US	S	NA	No	PPI bridge PPIB00
	0x40043000	PPIB00 : NS					
68	0x50044000	PPIB01 : S	US	S	NA	No	PPI bridge PPIB01
	0x40044000	PPIB01 : NS					
69	0x50045000	KMU	HF	S	NSA	No	Key management unit
70	0x50046000	AAR00 : S	US	S	SA	No	Accelerated address resolver 00
	0x40046000	AAR00 : NS					
70	0x50046000	CCM00 : S	US	S	SA	No	AES CCM mode encryption CCM00, running of HCLK128M
	0x40046000	CCM00 : NS					
71	0x50047000	ECB00 : S	US	S	SA	No	When configuring this peripheral's DMA security using SPU configuration (DMASEC field of SPU->PERIPH[apb_slave_index]), use apb_slave_index 6 (same as AAR00 and CCM00)
	0x40047000	ECB00 : NS					
72	0x50048000	CRACEN	HF	S	NSA	No	Crypto accelerator
74	0x5004A000	SPIM00 : S	US	S	SA	No	SPI controller SPIM00
	0x4004A000	SPIM00 : NS					
74	0x5004A000	SPIS00 : S	US	S	SA	No	SPI peripheral SPIS00
	0x4004A000	SPIS00 : NS					
74	0x5004A000	UARTE00 : S	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE00
	0x4004A000	UARTE00 : NS					
75	0x5004B000	GLITCHDET	HF	S	NA	No	Glitch detectors
75	0x5004B000	RRAMC	HF	S	NA	No	RRAM Non-Volatile Memory Controller
76	0x5004C000	VPR00 : S	US	NS	NSA	No	FLPR - VPR peripheral registers
	0x4004C000	VPR00 : NS					
80	0x50050400	P2 : S	US	S	NA	Yes	General purpose input and output, port P2 Does not support pin sense mechanism, and DETECTMODE register has no effect. Supports extra high drive (DRIVE0=E0, DRIVE1=E1).
	0x40050400	P2 : NS					
82	0x50052000	CTRLAP : S	US	S	NSA	No	Control access port CPU side
	0x40052000	CTRLAP : NS					
83	0x50053000	TAD : S	US	S	NA	No	Empty instance abstract
	0x40053000	TAD : NS					
85	0x50055000	TIMER00 : S	US	S	NA	No	Timer TIMER00
	0x40055000	TIMER00 : NS					
128	0x50080000	SPU10	HF	S	NA	No	System protection unit SPU10
130	0x50082000	DPPIC10 : S	US	S	NA	Yes	DPPI controller DPPIC10
	0x40082000	DPPIC10 : NS					
131	0x50083000	PPIB10 : S	US	S	NA	No	PPI bridge PPIB10
	0x40083000	PPIB10 : NS					
132	0x50084000	PPIB11 : S	US	S	NA	No	PPI bridge PPIB11
	0x40084000	PPIB11 : NS					
133	0x50085000	TIMER10 : S	US	S	NA	No	Timer TIMER10
	0x40085000	TIMER10 : NS					
135	0x50087000	EGU10 : S	US	S	NA	No	Event generator unit EGU10
	0x40087000	EGU10 : NS					
138	0x5008A000	RADIO : S	US	S	SA	No	2.4 GHz radio RADIO
	0x4008A000	RADIO : NS					
192	0x500C0000	SPU20	HF	S	NA	No	System protection unit SPU20

ID	Base address	Instance	TrustZone			Split access	Description
			Map	Att	DMA		
194	0x500C2000	DPPIC20 : S	US	S	NA	Yes	DPPI controller DPPIC20
	0x400C2000	DPPIC20 : NS					
195	0x500C3000	PPIB20 : S	US	S	NA	No	PPI bridge PPIB20
	0x400C3000	PPIB20 : NS					
196	0x500C4000	PPIB21 : S	US	S	NA	No	PPI bridge PPIB21
	0x400C4000	PPIB21 : NS					
197	0x500C5000	PPIB22 : S	US	S	NA	No	PPI bridge PPIB22
	0x400C5000	PPIB22 : NS					
198	0x500C6000	SPIM20 : S	US	S	SA	No	SPI controller SPIM20
	0x400C6000	SPIM20 : NS					
198	0x500C6000	SPIS20 : S	US	S	SA	No	SPI peripheral SPIS20
	0x400C6000	SPIS20 : NS					
198	0x500C6000	TWIM20 : S	US	S	SA	No	Two-wire interface controller TWIM20
	0x400C6000	TWIM20 : NS					
198	0x500C6000	TWIS20 : S	US	S	SA	No	Two-wire interface target TWIS20
	0x400C6000	TWIS20 : NS					
198	0x500C6000	UARTE20 : S	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE20
	0x400C6000	UARTE20 : NS					
199	0x500C7000	SPIM21 : S	US	S	SA	No	SPI controller SPIM21
	0x400C7000	SPIM21 : NS					
199	0x500C7000	SPIS21 : S	US	S	SA	No	SPI peripheral SPIS21
	0x400C7000	SPIS21 : NS					
199	0x500C7000	TWIM21 : S	US	S	SA	No	Two-wire interface controller TWIM21
	0x400C7000	TWIM21 : NS					
199	0x500C7000	TWIS21 : S	US	S	SA	No	Two-wire interface target TWIS21
	0x400C7000	TWIS21 : NS					
199	0x500C7000	UARTE21 : S	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE21
	0x400C7000	UARTE21 : NS					
200	0x500C8000	SPIM22 : S	US	S	SA	No	SPI controller SPIM22
	0x400C8000	SPIM22 : NS					
200	0x500C8000	SPIS22 : S	US	S	SA	No	SPI peripheral SPIS22
	0x400C8000	SPIS22 : NS					
200	0x500C8000	TWIM22 : S	US	S	SA	No	Two-wire interface controller TWIM22
	0x400C8000	TWIM22 : NS					
200	0x500C8000	TWIS22 : S	US	S	SA	No	Two-wire interface target TWIS22
	0x400C8000	TWIS22 : NS					
200	0x500C8000	UARTE22 : S	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE22
	0x400C8000	UARTE22 : NS					
201	0x500C9000	EGU20 : S	US	S	NA	No	Event generator unit EGU20
	0x400C9000	EGU20 : NS					
202	0x500CA000	TIMER20 : S	US	S	NA	No	Timer TIMER20
	0x400CA000	TIMER20 : NS					
203	0x500CB000	TIMER21 : S	US	S	NA	No	Timer TIMER21
	0x400CB000	TIMER21 : NS					
204	0x500CC000	TIMER22 : S	US	S	NA	No	Timer TIMER22
	0x400CC000	TIMER22 : NS					
205	0x500CD000	TIMER23 : S	US	S	NA	No	Timer TIMER23
	0x400CD000	TIMER23 : NS					
206	0x500CE000	TIMER24 : S	US	S	NA	No	Timer TIMER24
	0x400CE000	TIMER24 : NS					
207	0x500CF000	MEMCONF : S	US	S	NA	No	Memory Configuration MEMCONF
	0x400CF000	MEMCONF : NS					

ID	Base address	Instance	TrustZone			Split access	Description
			Map	Att	DMA		
208	0x500D0000	PDM20 : S	US	S	SA	No	Pulse density modulation (digital microphone) interface PDM20
	0x400D0000	PDM20 : NS					
209	0x500D1000	PDM21 : S	US	S	SA	No	Pulse density modulation (digital microphone) interface PDM21
	0x400D1000	PDM21 : NS					
210	0x500D2000	PWM20 : S	US	S	SA	No	Pulse width modulation unit PWM20
	0x400D2000	PWM20 : NS					
211	0x500D3000	PWM21 : S	US	S	SA	No	Pulse width modulation unit PWM21
	0x400D3000	PWM21 : NS					
212	0x500D4000	PWM22 : S	US	S	SA	No	Pulse width modulation unit PWM22
	0x400D4000	PWM22 : NS					
213	0x500D5000	SAADC : S	US	S	SA	No	Successive approximation analog-to-digital converter SAADC
	0x400D5000	SAADC : NS					
214	0x500D6000	NFCT : S	US	S	SA	No	Near field communication tag NFCT
	0x400D6000	NFCT : NS					
215	0x500D7000	TEMP : S	US	S	NA	No	Temperature sensor TEMP
	0x400D7000	TEMP : NS					
216	0x500D8200	P1 : S	US	S	NA	Yes	General purpose input and output, port P1
	0x400D8200	P1 : NS					
218	0x500DA000	GPIOTE20 : S	US	S	NA	Yes	8 channels and 2 interrupts for GPIO port P1 GPIO tasks and events GPIOTE20
	0x400DA000	GPIOTE20 : NS					
220	0x500DC000	TAMPC	HF	S	NA	No	Tamper controller TAMPC
221	0x500DD000	I2S20 : S	US	S	SA	No	Inter-IC sound interface I2S20
	0x400DD000	I2S20 : NS					
224	0x500E0000	QDEC20 : S	US	S	NA	No	Quadrature decoder QDEC20
	0x400E0000	QDEC20 : NS					
225	0x500E1000	QDEC21 : S	US	S	NA	No	Quadrature decoder QDEC21
	0x400E1000	QDEC21 : NS					
226	0x500E2000	GRTC : S	US	S	NA	Yes	Global RTC GRTC
	0x400E2000	GRTC : NS					
256	0x50100000	SPU30	HF	S	NA	No	System protection unit SPU30
258	0x50102000	DPPIC30 : S	US	S	NA	Yes	DPPI controller DPPIC30
	0x40102000	DPPIC30 : NS					
259	0x50103000	PPIB30 : S	US	S	NA	No	PPI bridge PPIB30
	0x40103000	PPIB30 : NS					
260	0x50104000	SPIM30 : S	US	S	SA	No	SPI controller SPIM30
	0x40104000	SPIM30 : NS					
260	0x50104000	SPIS30 : S	US	S	SA	No	SPI peripheral SPIS30
	0x40104000	SPIS30 : NS					
260	0x50104000	TWIM30 : S	US	S	SA	No	Two-wire interface controller TWIM30
	0x40104000	TWIM30 : NS					
260	0x50104000	TWIS30 : S	US	S	SA	No	Two-wire interface target TWIS30
	0x40104000	TWIS30 : NS					
260	0x50104000	UARTE30 : S	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE30
	0x40104000	UARTE30 : NS					
262	0x50106000	COMP : S	US	S	NA	No	Comparator COMP
	0x40106000	COMP : NS					
262	0x50106000	LPCOMP : S	US	S	NA	No	Low-power comparator LPCOMP
	0x40106000	LPCOMP : NS					
264	0x50108000	WDT30	HF	S	NA	No	Watchdog timer WDT30
265	0x50109000	WDT31 : S	US	S	NA	No	Watchdog timer WDT31
	0x40109000	WDT31 : NS					

ID	Base address	Instance	TrustZone			Split access	Description
			Map	Att	DMA		
266	0x5010A000	PO : S	US	S	NA	Yes	General purpose input and output, port P0
	0x4010A000	PO : NS					
268	0x5010C000	GPIOE30 : S	US	S	NA	Yes	4 channels and 2 interrupts for GPIO port P0 GPIO tasks and events GPIOE30
	0x4010C000	GPIOE30 : NS					
270	0x5010E000	CLOCK : S	US	S	NA	No	Clock control
	0x4010E000	CLOCK : NS					
270	0x5010E000	POWER : S	US	S	NA	No	Power control
	0x4010E000	POWER : NS					
270	0x5010E000	RESET : S	US	S	NA	No	Reset status
	0x4010E000	RESET : NS					
288	0x50120000	OSCILLATORS : S	US	S	NA	No	Oscillator control
	0x40120000	OSCILLATORS : NS					
288	0x50120000	REGULATORS : S	US	S	NA	No	Regulator control
	0x40120000	REGULATORS : NS					
N/A	0x00FFC000	FICR	HF	NS	NA	No	Factory information configuration
N/A	0x00FFD000	UICR	HF	S	NA	No	User information configuration
N/A	0x00FFE000	SICR	HF	S	NA	No	Secure information configuration region
N/A	0x51800000	CRACENCORE	HF	S	NSA	No	CRACEN core

Table 8: Instantiation table

4 Application core

4.1 Arm Cortex-M33 CPU

4.1.1 CPU

The Arm Cortex-M33 processor has a 32-bit instruction set (Thumb-2 technology) that implements a super set of 16- and 32-bit instructions to maximize code density and performance.

This processor implements several features that enable energy-efficient arithmetic and high-performance signal processing including the following:

- Digital signal processing (DSP) instructions:
 - Single-cycle multiply and accumulate (MAC) instructions
 - 8- and 16-bit single instruction multiple data (SIMD) instructions
- Hardware divide
- Single-precision floating-point unit (FPU)
- Memory Protection Unit (MPU)
- Arm TrustZone for Armv8-M
- Stack limit checking

The [Arm Cortex Microcontroller Software Interface Standard \(CMSIS\)](#) is implemented and available for the processor.

Real-time execution is highly deterministic in Thread mode, to and from sleep modes, and when handling events at configurable priority levels via the Nested Vectored Interrupt Controller (NVIC).

An instruction cache is introduced on the C-bus (code bus) of the Cortex-M33 CPU to improve performance when fetching instructions (or data) from internal non-volatile memory. For more information on cache, see [CACHE — Instruction/data cache](#) on page 36. CPU performance parameters including wait states for configurations, CPU current consumption and efficiency, and processing power and efficiency based on the CoreMark benchmark can be found in [CPU Electrical specification](#) on page 816.

4.1.1.1 Floating point interrupt

The floating point unit (FPU) may generate exceptions, for example, due to overflow or underflow. These exceptions may trigger interrupts when enabled in the FPU peripheral. For information on the FPU interrupts, see [CPUC — CPU control](#) on page 27.

4.1.1.2 CPU and support module configuration

The Arm Cortex-M33 processor has a number of CPU options and support modules implemented on the device.

Option	Description	Implemented
WIC	Wakeup Interrupt Controller	No
Endianness	Memory system endianness	Little endian
DWT	Data Watchpoint and Trace	Yes

Table 9: Core options

Module	Description	Implemented
MPU	Number of non-secure MPU regions	16
	Number of secure MPU regions	16
SAU	Number of SAU regions	4
FPU	Floating-point unit	Yes
DSP	Digital Signal Processing Extension	Yes
Arm TrustZone for Armv8-M	Armv8-M Security Extensions	Yes
CPIF	Coprocessor interface	No
ETM	Embedded Trace Macrocell	Yes
ITM	Instrumentation Trace Macrocell	Yes
MTB	Micro Trace Buffer	No
CTI	Cross Trigger Interface	No
BPU	Breakpoint Unit	Yes
INITSVTOR	System reset secure vector table address after reset	0x00000000
INITNSVTOR	System reset non-secure vector table address after reset	0x00000000

Table 10: Modules

4.1.2 CPUC — CPU control

CPUC controls elements of the Arm Cortex-M33 processor such as enabling floating-point exceptions. It is also able to lock certain features of the CPU and prevent them from being modified.

CPUC can generate events for exceptions in the floating point unit (FPU), as shown in the following block diagram. Examples of such exceptions are divide-by-zero, or floating-point overflow.

These exceptions can trigger interrupts when enabled using registers [INTEN](#) on page 30 or [INTENSET](#) on page 30.

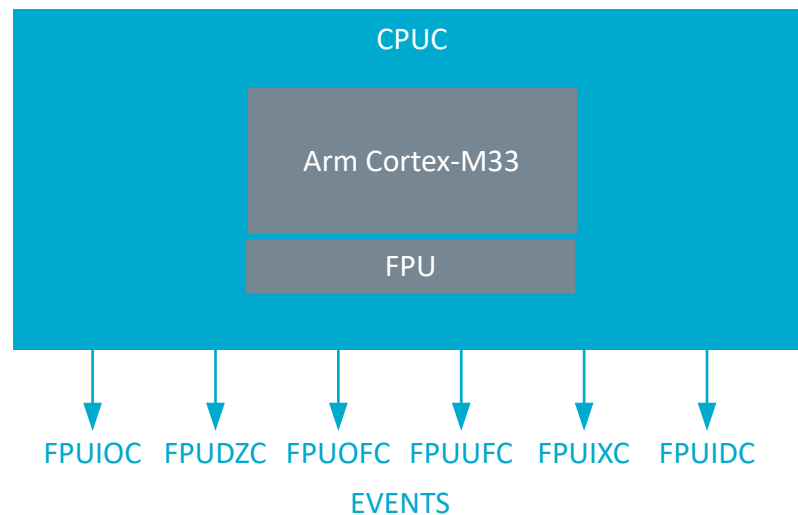


Figure 4: Block diagram

CPUC holds a CPU identifier **CPUID**, used in the system to uniquely identify the processing unit of a core.

In addition, CPUC holds a **LOCK** register, which is used to lock certain CPU features and prevent them from being modified. One example is the **LOCK.LOCKSAU** field. When set to **Locked**, this prevents further modifications to the SAU registers.

4.1.2.1 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
CPUC	APPLICATION	0xE0080000	HF	S	NA	No	Cortex-M33 configuration

Register overview

Register	Offset	TZ	Description
EVENTS_FPUIOC	0x100		An invalid operation exception has occurred in the FPU.
EVENTS_FPUDZC	0x104		A floating-point divide-by-zero exception has occurred in the FPU.
EVENTS_FPUOFC	0x108		A floating-point overflow exception has occurred in the FPU.
EVENTS_FPUUFC	0x10C		A floating-point underflow exception has occurred in the FPU.
EVENTS_FPUIXC	0x110		A floating-point inexact exception has occurred in the FPU.
EVENTS_FPUIDC	0x114		A floating-point input denormal exception has occurred in the FPU.
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
LOCK	0x500		Register to lock the certain parts of the CPU from being modified.
CPUID	0x504		The identifier for the CPU in this subsystem.

4.1.2.1.1 EVENTS_FPUIOC

Address offset: 0x100

An invalid operation exception has occurred in the FPU.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	EVENTS_FPUIOC				An invalid operation exception has occurred in the FPU.																													
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

4.1.2.1.2 EVENTS_FPUDZC

Address offset: 0x104

A floating-point divide-by-zero exception has occurred in the FPU.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																																							A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	EVENTS_FPUDZC				A floating-point divide-by-zero exception has occurred in the FPU.																																	
			NotGenerated	0	Event not generated																																		
			Generated	1	Event generated																																		

4.1.2.1.3 EVENTS_FPUOFC

Address offset: 0x108

A floating-point overflow exception has occurred in the FPU.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_FPUOFC			A floating-point overflow exception has occurred in the FPU.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

4.1.2.1.4 EVENTS_FPUUFC

Address offset: 0x10C

A floating-point underflow exception has occurred in the FPU.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				A																																			
Reset 0x00000000				0 0																																			
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	EVENTS_FPUUFC				A floating-point underflow exception has occurred in the FPU.																																	
			NotGenerated	0		Event not generated																																	
			Generated	1		Event generated																																	

4.1.2.1.5 EVENTS_FPUIXC

Address offset: 0x110

A floating-point inexact exception has occurred in the FPU.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				A																																		
Reset 0x00000000				0 0																																		
ID	R/W	Field	Value ID	Value		Description																																
A	RW	EVENTS_FPUIC				A floating-point inexact exception has occurred in the FPU.																																
			NotGenerated	0	Event not generated																																	
			Generated	1	Event generated																																	

4.1.2.1.6 EVENTS_FPUIDC

Address offset: 0x114

A floating-point input denormal exception has occurred in the FPU.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																																							A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	EVENTS_FPUIDC				A floating-point input denormal exception has occurred in the FPU.																																	
			NotGenerated	0	Event not generated																																		
			Generated	1	Event generated																																		

4.1.2.1.7 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	FPUIOC			Enable or disable interrupt for event FPUIOC																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
B	RW	FPUDZC			Enable or disable interrupt for event FPUDZC																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
C	RW	FPUOFC			Enable or disable interrupt for event FPUOFC																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
D	RW	FPUUFC			Enable or disable interrupt for event FPUUFC																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
E	RW	FPUIXC			Enable or disable interrupt for event FPUIXC																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
F	RW	FPUIDC			Enable or disable interrupt for event FPUIDC																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

4.1.2.1.8 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	FPUIOC			Write '1' to enable interrupt for event FPUIOC																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	FPUDZC			Write '1' to enable interrupt for event FPUDZC																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	FPUOFC			Write '1' to enable interrupt for event FPUOFC																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	FPUUFC			Write '1' to enable interrupt for event FPUUFC																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
E	RW	FPUIXC			Write '1' to enable interrupt for event FPUIXC																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
F	RW	FPUIDC			Write '1' to enable interrupt for event FPUIDC																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

4.1.2.1.9 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	FPUIOC			Write '1' to disable interrupt for event FPUIOC																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	FPUDZC			Write '1' to disable interrupt for event FPUDZC																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	FPUOFC			Write '1' to disable interrupt for event FPUOFC																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	FPUUFC			Write '1' to disable interrupt for event FPUUFC																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																					
ID																																				F	E	D	C	B	A
Reset 0x00000000				0 0																																					
ID	R/W	Field	Value ID	Value	Description																																				
			Enabled	1	Read: Enabled																																				
E	RW	FPUIXC			Write '1' to disable interrupt for event FPUIXC																																				
			Clear	1	Disable																																				
			Disabled	0	Read: Disabled																																				
			Enabled	1	Read: Enabled																																				
F	RW	FPUIDC			Write '1' to disable interrupt for event FPUIDC																																				
			Clear	1	Disable																																				
			Disabled	0	Read: Disabled																																				
			Enabled	1	Read: Enabled																																				

4.1.2.1.10 LOCK

Address offset: 0x500

Register to lock the certain parts of the CPU from being modified.

Each bit can only be written once and can only be changed from 0 to 1.

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			E D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	LOCKVTORAIRCRS			Locks both the Vector table Offset Register (VTOR) and Application Interrupt and Reset Control Register (AIRCR) for secure mode.																													
			NotLocked	0	Both VTOR and AIRCR can be changed.																													
			Locked	1	Prevents changes to both VTOR and AIRCR.																													
B	RW	LOCKVTORNS			Locks the Vector table Offset Register (VTOR) for non-secure mode.																													
			NotLocked	0	VTOR can be changed.																													
			Locked	1	Prevents changes to VTOR.																													
C	RW	LOCKMPUS			Locks the Memory Protection Unit (MPU) for secure mode.																													
			NotLocked	0	MPU registers can be changed.																													
			Locked	1	Prevents changes to MPU registers.																													
D	RW	LOCKMPUNS			Locks the Memory Protection Unit (MPU) for non secure mode.																													
			NotLocked	0	MPU registers can be changed.																													
			Locked	1	Prevents changes to MPU registers.																													
E	RW	LOCKS AU			Locks the Security Attribution Unit (SAU)																													
			NotLocked	0	SAU registers can be changed.																													
			Locked	1	Prevents changes to SAU registers.																													

4.1.2.1.11 CPUID

Address offset: 0x504

The identifier for the CPU in this subsystem.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	CPUID										The CPU identifier.																							

4.1.3 Arm Cortex-M33 Peripherals

4.1.3.1 Instantiation

ID	Base address	Instance	TrustZone			Split access	Description
			Map	Att	DMA		
28	0x5001C000	SWI00	HF	S	NA	No	Software interrupt SWI00
29	0x5001D000	SWI01	HF	S	NA	No	Software interrupt SWI01
30	0x5001E000	SWI02	HF	S	NA	No	Software interrupt SWI02
31	0x5001F000	SWI03	HF	S	NA	No	Software interrupt SWI03
N/A	0x02F00000	ICACHEDATA	HF	S	NA	No	Instruction cache data
N/A	0x02F10000	ICACHEINFO	HF	S	NA	No	Instruction cache info
N/A	0xE0040000	TPIU	HF	NS	NA	No	Trace port interface unit (Trace and Debug)
N/A	0xE0041000	ETM	HF	NS	NA	No	Embedded trace macrocell
N/A	0xE0080000	CPUC	HF	S	NA	No	Cortex-M33 configuration
N/A	0xE0082000	ICACHE	HF	S	NA	No	Instruction cache

Table 11: Instantiation table

4.2 Core components

4.2.1 AMBA interconnect (AMBIX)

The AMBA interconnect (AMBIX) is a multilayer capable bus interconnect that provides low latency access from Managers to Subordinates.

Manager and Subordinate connections are arranged in Manager and Subordinate pairs, allowing for a sparse bus matrix. The interconnect supports multiple concurrent transactions when targeting different Subordinates.

The interconnect also enforces the TrustZone secure/non-secure attributes and is configured using [MPC — Memory Privilege Controller](#) on page 175.

4.2.1.1 AMBIX0 bus Managers and priority handling

The main interconnect (AMBIX0) has a bus matrix that handles bus arbitration.

AMBIX0 uses a round robin bus Manager arbitration algorithm.

Some peripherals are not able to pause incoming data. As a low priority bus Manager, data loss is possible for these peripherals when bus contention occurs. To avoid bus contention when using multiple bus Managers, follow these guidelines:

- Avoid situations where more than one bus Manager is accessing the same RAM Subordinate.
- If more than one bus Manager is accessing the same Subordinate, make sure that the bus bandwidth is not exhausted.

4.2.1.2 AMBIX0 override configuration

The main interconnect (AMBIX0) has a configurable bus matrix.

The AMBIX0 override configuration is done using the [MPC.OVERRIDE](#) registers in [MPC — Memory Privilege Controller](#) on page 175.

The overrides are used to configure the secure and non-secure memory regions in the device. They are also used to prevent or grant access to read, write, or execute from the memory region. For more details, see [MPC — Memory Privilege Controller](#) on page 175.

4.2.2 EasyDMA

EasyDMA is a module implemented by some peripherals as a bus manager for direct access to RAM. It cannot access non-volatile memory,

A peripheral can implement multiple EasyDMA instances to provide dedicated channels. For example, a channel can be dedicated for reading and writing data between the peripheral and RAM. This concept is illustrated in the following figure, where READER is reading data from RAM2, while WRITER is writing data to RAM0 and RAM1.

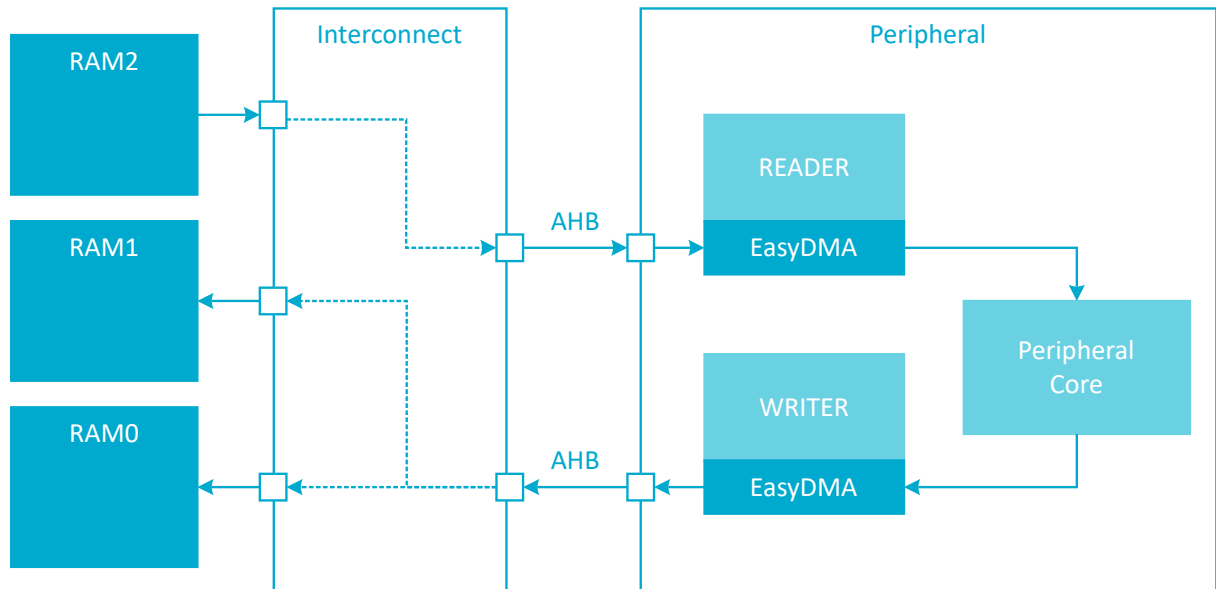


Figure 5: EasyDMA example

4.2.2.1 EasyDMA channel implementation

A typical EasyDMA channel is implemented in the following way.

```

READERBUFFER_SIZE 5
WRITERBUFFER_SIZE 6

uint8_t readerBuffer[READERBUFFER_SIZE] __at__ 0x20000000;
uint8_t writerBuffer[WRITERBUFFER_SIZE] __at__ 0x20000005;

// Configuring the READER channel
MYPERIPHERAL->READER.MAXCNT = READERBUFFER_SIZE;
MYPERIPHERAL->READER.PTR = &readerBuffer;

// Configure the WRITER channel
MYPERIPHERAL->WRITER.MAXCNT = WRITERBUFFER_SIZE;
MYPERIPHERAL->WRITER.PTR = &writerBuffer;

```

This example shows a peripheral called MYPERIPHERAL that implements two EasyDMA channels. One channel is for reading called READER, and one for writing called WRITER. When the peripheral starts, it performs the following tasks.

1. Reads 5 B from the readerBuffer located in RAM at address 0x20000000.
2. Processes the data.
3. Writes up to 6 B back to the writerBuffer located in RAM at address 0x20000005.

The memory layout of these buffers is illustrated is shown in the following figure.

0x20000000	readerBuffer[0]	readerBuffer[1]	readerBuffer[2]	readerBuffer[3]
0x20000004	readerBuffer[4]	writerBuffer[0]	writerBuffer[1]	writerBuffer[2]
0x20000008	writerBuffer[3]	writerBuffer[4]	writerBuffer[5]	

Figure 6: EasyDMA memory layout

The specified size of the WRITER.MAXCNT register must not be larger than the actual size of the buffer (writerBuffer). This prevents the channel from overflowing the writerBuffer.

Once an EasyDMA transfer is complete, the CPU reads the AMOUNT register to see how many bytes were transferred. For example, the CPU can read the MYPERIPHERAL.WRITER.AMOUNT register to see how many bytes WRITER wrote to RAM.

Note: A READER or WRITER PTR register must point to a valid memory region before using EasyDMA. The reset value of a PTR register is not guaranteed to point to valid memory. See [Memory](#) on page 19 for more information about the memory regions and EasyDMA connectivity.

4.2.2.2 EasyDMA error handling

Errors can occur during DMA handling.

If READER.PTR or WRITER.PTR is not pointing to a valid memory region, an EasyDMA transfer could HardFault or cause RAM corruption. See [Memory](#) on page 19 for more information about the different memory regions.

An EasyDMA channel is an AHB bus Manager. If several AHB Managers try to access the same AHB Subordinate at the same time, AHB bus congestion can occur. Depending on the peripheral, the peripheral could either stall and wait for access to be granted, or lose data.

4.2.2.3 Array list

EasyDMA can operate in Array List mode.

The Array List mode is implemented in channels where the LIST register is available.

The array list is not able to specify where the next item in the list is located. Instead, it assumes that the list is organized as a linear array where items are located one after the other in RAM.

The EasyDMA array list is implemented with the data structure `ArrayList_type`. This is illustrated in the following code example using a READER EasyDMA channel as an example.

```
#define BUFFER_SIZE 4

typedef struct ArrayList
{
    uint8_t buffer[BUFFER_SIZE];
} ArrayList_type;

ArrayList_type ReaderList[3] __at__ 0x20000000;

MYPERIPHERAL->READER.MAXCNT = BUFFER_SIZE;
MYPERIPHERAL->READER.PTR = &ReaderList;
MYPERIPHERAL->READER.LIST = MYPERIPHERAL_READER_LIST_ArrayList;
```

The data structure includes a buffer that is equal in size to the READER.MAXCNT register. EasyDMA uses the READER.MAXCNT register to determine when the buffer is full.

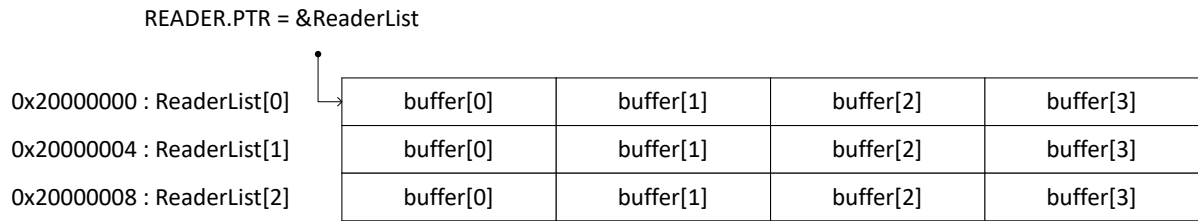


Figure 7: EasyDMA array list

4.2.3 CACHE — Instruction/data cache

The cache is two-way set associative with a least recently used (LRU) replacement policy. Both instruction and data accesses towards NVM memory are cached.

The cache has the following features:

- 4x64-bit cache line
- Ability to enable/disable cache at run-time
- Writes to cached memory are write-around and invalidate the cache line
- Manual invalidation and erase support
- Locking cache updates on cache misses
- Performance hit/miss counter registers for profiling CACHE operations
- Optional readable cache content for profiling
 - Data, tag, valid, and most recently used (MRU) bits
 - Can be disabled when not in use

The cache must be enabled by the [ENABLE](#) register.

4.2.3.1 Architecture

The following figure shows the cache architecture.

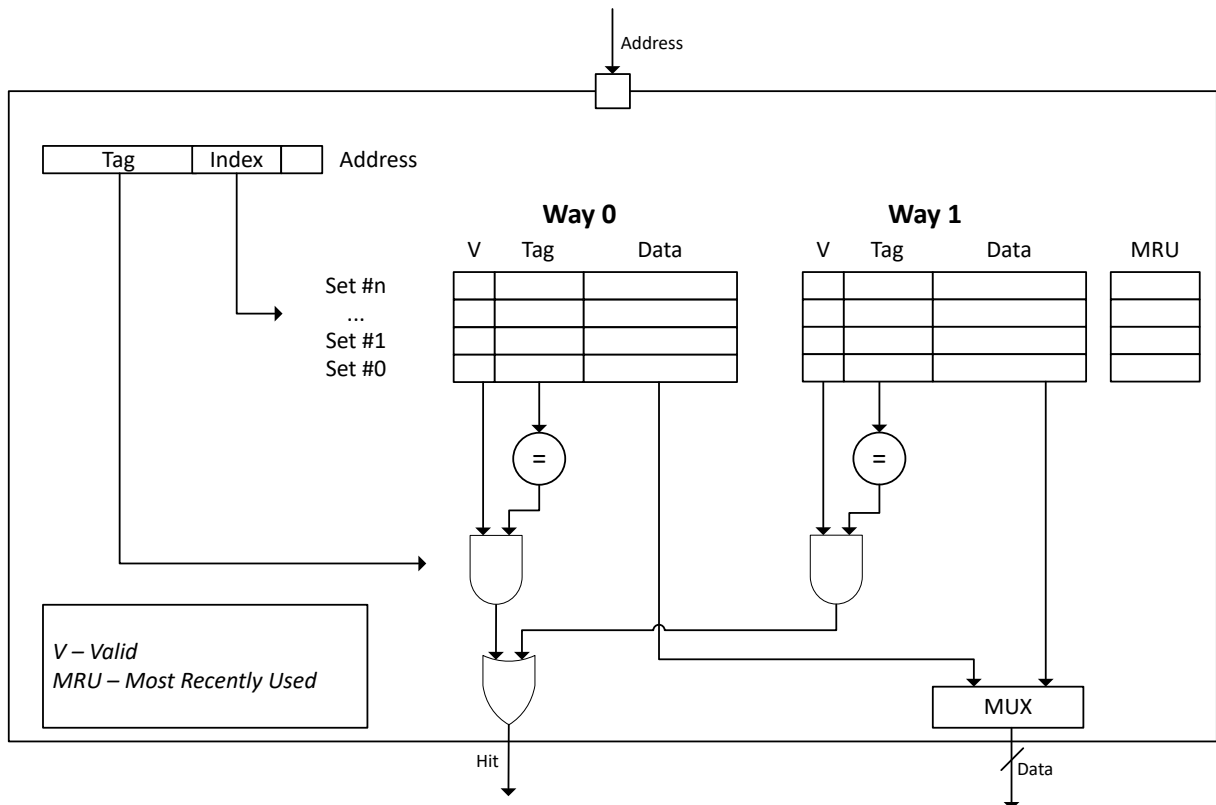


Figure 8: Cache overview

Bit	Name	Description
V	Valid	Indicates if a cache entry is valid. All V fields are cleared when enabling the cache, invalidating the cache, or when changing CACHE mode.
MRU	Most Recently Used	Updated on each fetch from the cache to indicate which Way was used most recently. Used to drive the cache replacement policy.

4.2.3.2 Profiling

The cache provides a scoreboard that tracks the hits and misses within the cache.

The results are available through a set of registers that can be used to indicate how well the cache is performing.

Profiling is enabled using [PROFILING.ENABLE](#). All profiling counters can be cleared at any time using [PROFILING.CLEAR](#). After being cleared, the counters will increment, according to the rules in the table below, at the next instruction- or data fetch.

Profiling counter	Description
HIT	Incremented on a cache hit
MISS	Incremented on a cache miss (not counting write misses)
LMISS	Incremented on a cache line miss (accessing from a new line in the cache)
READS	Incremented on a CPU cache read
WRITES	Incremented on a CPU cache write

Table 12: Profiling counters

4.2.3.3 Debug

The CPU is able to read internal cache memories and tags for debug purposes.

The content of data and tag RAM's are accessible through registers `SET[n].WAY[o].INFO` ($n=0..127$) ($o=0..1$) on page 43 and `SET[n].WAY[o].DU[p].DATA[q]` ($n=0..127$) ($o=0..1$) ($p=0..3$) ($q=0..1$) on page 44.

Debug access is prevented by using register `DEBUGLOCK` on page 42.

4.2.3.4 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
ICACHE	APPLICATION	0xE0082000	HF	S	NA	No	Instruction cache

Configuration

Instance	Domain	Configuration
ICACHE	APPLICATION	<p>Interruptions are not connected.</p> <p>Cache size: 8 KB. Sets: 128. Data unit: 64 bits. Line width = 4 data units.</p> <p>Does not support virtual cache</p> <p>Does not support cache flush</p> <p>Does not support cache clean</p> <p>Does not support non-cacheable miss feature</p> <p>Data bus width : 0..63</p>

Register overview

Register	Offset	TZ	Description
<code>TASKS_INVALIDATECACHE</code>	0x008		Invalidate the cache.
<code>TASKS_INVALIDATELINE</code>	0x014		Invalidate the line.
<code>TASKS_ERASE</code>	0x020		Erase the cache.
<code>STATUS</code>	0x400		Status of the cache activities.
<code>ENABLE</code>	0x404		Enable cache.
<code>LINEADDR</code>	0x410		Memory address covered by the line to be maintained.
<code>PROFILING.ENABLE</code>	0x414		Enable the profiling counters.
<code>PROFILING.CLEAR</code>	0x418		Clear the profiling counters.
<code>PROFILING.HIT</code>	0x41C		The cache hit counter for cache region.
<code>PROFILING.MISS</code>	0x420		The cache miss counter for cache region.
<code>PROFILING.LMISS</code>	0x424		The cache line miss counter for cache region.
<code>PROFILING.READS</code>	0x428		Number of reads for cache region.
<code>PROFILING.WRITES</code>	0x42C		Number of writes for cache region.
<code>DEBUGLOCK</code>	0x430		Lock debug mode.
<code>WRITELOCK</code>	0x434		Lock cache updates.

4.2.3.4.1 TASKS_INVALIDATECACHE

Address offset: 0x008

Invalidate the cache.

The **STATUS** is updated for this task. This task can be triggered only when the **STATUS** is ready.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_INVALIDATECACHE							Invalidate the cache.																										
								The STATUS is updated for this task. This task can be triggered only when the STATUS is ready.																											
			Trigger	1				Trigger task																											

4.2.3.4.2 TASKS_INVALIDATELINE

Address offset: 0x014

Invalidate the line.

The **STATUS** is updated for this task. This task can be triggered only when the **STATUS** is ready.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															

4.2.3.4.3 TASKS_ERASE

Address offset: 0x020

Erase the cache.

The **STATUS** is updated for this task. This task can be triggered only when the **STATUS** is ready.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																																																																															
ID				A																																																																																															
Reset 0x00000000				0 0																																																																																															
ID	R/W	Field	Value ID	Value																																Description																																																															
A	W	TASKS_ERASE																																		Erase the cache.																																																															
																																				The STATUS is updated for this task. This task can be triggered only when the STATUS is ready.																																																															
				Trigger																																1																																Trigger task																															

4.2.3.4.4 STATUS

Address offset: 0x400

Status of the cache activities.

Indicates status of the cache/line invalidate, clean, flush and erase activities initiated using respective tasks. The status also includes for the save and restore tasks.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	READY			Ready status.																														
			Ready	0	Activity is done and ready for the next activity.																														
			Busy	1	Activity is in progress.																														

4.2.3.4.5 ENABLE

Address offset: 0x404

Enable cache.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE			Enable cache																														
			Disabled	0	Disable cache																														
			Enabled	1	Enable cache																														

4.2.3.4.6 LINEADDR

Address offset: 0x410

Memory address covered by the line to be maintained.

The line maintain activities are line invalidate, line clean and line flush.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ADDR						Address.																											

4.2.3.4.7 PROFILING.ENABLE

Address offset: 0x414

Enable the profiling counters.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE			Enable the profiling counters																														
			Disable	0	Disable profiling																														
			Enable	1	Enable profiling																														

4.2.3.4.8 PROFILING.CLEAR

Address offset: 0x418

Clear the profiling counters.

The profiling counters can be cleared at any time. When cleared, all profiling counters will be set to zero, and will increment at the next instruction- or data fetch.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	W	CLEAR						Clearing the profiling counters																											
			Clear	1				Clear the profiling counters																											

4.2.3.4.9 PROFILING.HIT

Address offset: 0x41C

The cache hit counter for cache region.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	HITS						Number of cache hits																											

4.2.3.4.10 PROFILING.MISS

Address offset: 0x420

The cache miss counter for cache region.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	MISSSES						Number of cache misses																											

4.2.3.4.11 PROFILING.LMISS

Address offset: 0x424

The cache line miss counter for cache region.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	LMISSSES						Number of cache line misses																											

4.2.3.4.12 PROFILING.READS

Address offset: 0x428

Number of reads for cache region.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	READS						Number of reads for cache region.																											

4.2.3.4.13 PROFILING.WRITES

Address offset: 0x42C

Number of writes for cache region.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	WRITES						Number of writes for cache region.																											

4.2.3.4.14 DEBUGLOCK

Address offset: 0x430

Lock debug mode.

Note: Debug mode can only be unlocked by a reset

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																	A									
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																					
A	RW1	DEBUGLOCK			Lock debug mode																																					
			Unlocked	0	Debug mode unlocked																																					
			Locked	1	Debug mode locked. Ignores any other value written.																																					

4.2.3.4.15 WRITELOCK

Address offset: 0x434

Lock cache updates.

Prevents updating of cache content on cache misses, but will continue to lookup instruction/data fetches in content already present in the cache. The write lock is applied to whole cache.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																																							A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	WRITELOCK																																					
			Unlocked	0	Cache updates unlocked																																		
			Locked	1	Cache updates locked																																		

4.2.3.5 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
ICACHEINFO	APPLICATION	0x02F10000	HF	S	NA	No	Instruction cache info

Configuration

Instance	Domain	Configuration
ICACHEINFO	APPLICATION	Number of sets : 0..127 Number of ways : 0..1 Number of data units : 0..3 Data width of a data unit : 0..1 words TAG width : 0..19

Register overview

Register	Offset	TZ	Description
SET[n].WAY[o].INFO	0x0		Cache information for SET[n], WAY[o].

4.2.3.5.1 SET[n].WAY[o].INFO (n=0..127) (o=0..1)

Address offset: $0x0 + (n \times 0x8) + (o \times 0x4)$

Cache information for SET[n], WAY[o].

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															

4.2.3.6 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
ICACHEDATA	APPLICATION	0x02F00000	HF	S	NA	No	Instruction cache data

Configuration

Instance	Domain	Configuration
ICACHEDATA	APPLICATION	Number of sets : 0..127 Number of ways : 0..1 Number of data units : 0..3 Data width of a data unit : 0..1 words

Register overview

Register	Offset	TZ	Description
SET[n].WAY[o].DU[p].DATA[q]	0x0		Cache data bits for DATA[q] in DU[p] (DataUnit) of SET[n], WAY[o].

4.2.3.6.1 SET[n].WAY[o].DU[p].DATA[q] (n=0..127) (o=0..1) (p=0..3) (q=0..1)

Address offset: $0x0 + (n \times 0x40) + (o \times 0x20) + (p \times 0x8) + (q \times 0x4)$

Cache data bits for DATA[q] in DU[p] (DataUnit) of SET[n], WAY[o].

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID										A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID				Value				Description																																	
A	R	Data									Data																																	

4.2.4 FICR — Factory information configuration registers

Factory information configuration registers (FICR) are pre-programmed in factory. These registers contain chip-specific information and configuration.

4.2.4.1 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
FICR	GLOBAL	0x00FFC000	HF	NS	NA	No	Factory information configuration

Register overview

Register	Offset	TZ	Description
INFO.CONFIGID	0x300		Configuration identifier
INFO.DEVICEID[n]	0x304		Device identifier
INFO.UUID[n]	0x30C		128-bit Universally Unique Identifier (UUID).
INFO.PART	0x31C		Part code
INFO.VARIANT	0x320		Part Variant, Hardware version and Production configuration
INFO.PACKAGE	0x324		Package option
INFO.RAM	0x328		RAM size (KB)
INFO.RRAM	0x32C		RRAM size (KB)
ER[n]	0x380		Common encryption root key, word n
IR[n]	0x390		Common identity root key, word n
DEVICEADDRTYPE	0x3A0		Device address type
DEVICEADDR[n]	0x3A4		Device address n
TRIMCNF[n].ADDR	0x400		Address of the register which will be written
TRIMCNF[n].DATA	0x404		Data to be written into the register
NFC.TAGHEADER0	0x600		Default header for NFC Tag. Software can read these values to populate NFCID1_3RD_LAST, NFCID1_2ND_LAST and NFCID1_LAST.
NFC.TAGHEADER1	0x604		Default header for NFC Tag. Software can read these values to populate NFCID1_3RD_LAST, NFCID1_2ND_LAST and NFCID1_LAST.
NFC.TAGHEADER2	0x608		Default header for NFC Tag. Software can read these values to populate NFCID1_3RD_LAST, NFCID1_2ND_LAST and NFCID1_LAST.
NFC.TAGHEADER3	0x60C		Default header for NFC Tag. Software can read these values to populate NFCID1_3RD_LAST, NFCID1_2ND_LAST and NFCID1_LAST.
XOSC32MTRIM	0x620		XOSC32M capacitor selection trim values
XOSC32KTRIM	0x624		XOSC32K capacitor selection trim values

4.2.4.1.1 INFO

Device info

4.2.4.1.1.1 INFO.CONFIGID

Address offset: 0x300

Configuration identifier

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	HWID										Identification number for the HW																							

4.2.4.1.1.2 INFO.DEVICEID[n] (n=0..1)

Address offset: 0x304 + (n × 0x4)

Device identifier

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	R	DEVICEID						64 bit unique device identifier																											
				DEVICEID[0] contains the least significant bits of the device identifier.																															
				DEVICEID[1] contains the most significant bits of the device identifier.																															

4.2.4.1.1.3 INFO.UUID[n] (n=0..3)

Address offset: 0x30C + (n × 0x4)

128-bit Universally Unique Identifier (UUID).

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	RW	UUID						Device UUID [n].																											
				The DEVICE.UUID[0] contains the least significant bits of the device identifier.																															

4.2.4.1.1.4 INFO.PART

Address offset: 0x31C

Part code

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	R	PART						Part code																											
			N54L15	0x00054B15				nRF54L15																											
			N54L10	0x00054B10				nRF54L10																											
			N54L05	0x00054B05				nRF54L05																											
			Unspecified	0xFFFFFFFF				Unspecified																											

4.2.4.1.1.5 INFO.VARIANT

Address offset: 0x320

Part Variant, Hardware version and Production configuration

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	R	VARIANT						Part Variant, Hardware version and Production configuration, encoded as ASCII																											
			Unspecified	0xFFFFFFFF				Unspecified																											

4.2.4.1.1.6 INFO.PACKAGE

Address offset: 0x324

Package option

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	PACKAGE										Package option																							
			Unspecified	0xFFFFFFFF								Unspecified																							

4.2.4.1.1.7 INFO.RAM

Address offset: 0x328

RAM size (KB)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	RAM										RAM size (KB)																							
			K256	0x100								256 kByte RAM																							
			K192	0xC0								192 kByte RAM																							
			K96	0x60								96 kByte RAM																							
			Unspecified	0xFFFFFFFF								Unspecified																							

4.2.4.1.1.8 INFO.RRAM

Address offset: 0x32C

RRAM size (KB)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	RRAM						RRAM size (KB)																											
			K1524	0x5F4				1524 KByte RRAM																											
			K1012	0x3F4				1012 KByte RRAM																											
			K500	0x1F4				500 KByte RRAM																											
			Unspecified	0xFFFFFFFF				Unspecified																											

4.2.4.1.2 ER[n] (n=0..3)

Address offset: 0x380 + (n × 0x4)

Common encryption root key, word n

Used to generate keys as recommended by the Bluetooth Core Specification

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	ER		Encryption Root, word n																															

4.2.4.1.3 IR[n] (n=0..3)

Address offset: $0x390 + (n \times 0x4)$

Common identity root key, word n

Used to generate keys as recommended by the Bluetooth Core Specification

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	R	IR						Identity Root, word n																											

4.2.4.1.4 DEVICEADDRTYPE

Address offset: $0x3A0$

Device address type

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				A																																		
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
ID	R/W	Field	Value ID	Value				Description																														
A	R	DEVICEADDRTYPE						Device address type																														
			Public	0				Public address																														
			Random	1				Random address																														

4.2.4.1.5 DEVICEADDR[n] (n=0..1)

Address offset: $0x3A4 + (n \times 0x4)$

Device address n

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	R	DEVICEADDR						48 bit device address																											
								DEVICEADDR[0] contains the least significant bits of the device address.																											
								DEVICEADDR[1] contains the most significant bits of the device address.																											
								Only bits [15:0] of DEVICEADDR[1] are used.																											

4.2.4.1.6 TRIMCNF[n].ADDR (n=0..63)

Address offset: $0x400 + (n \times 0x8)$

Address of the register which will be written

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value										Description																					
A	R	Address												Address																					

4.2.4.1.7 TRIMCNF[n].DATA (n=0..63)

Address offset: $0x404 + (n \times 0x8)$

Data to be written into the register

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	R	Data						Data																											

4.2.4.1.8 NFC.TAGHEADER0

Address offset: 0x600

Default header for NFC Tag. Software can read these values to populate NFCID1_3RD_LAST, NFCID1_2ND_LAST and NFCID1_LAST.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D	D	D	D	D	D	D	D	C	C	C	C	C	C	C	B	B	B	B	B	B	B	B	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF5F				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	R	MFGID						Default Manufacturer ID: Nordic Semiconductor ASA has ICM 0x5F																											
B	R	UD1						Unique identifier byte 1																											
C	R	UD2						Unique identifier byte 2																											
D	R	UD3						Unique identifier byte 3																											

4.2.4.1.9 NFC.TAGHEADER1

Address offset: 0x604

Default header for NFC Tag. Software can read these values to populate NFCID1_3RD_LAST, NFCID1_2ND_LAST and NFCID1_LAST.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D	D	D	D	D	D	D	C	C	C	C	C	C	C	B	B	B	B	B	B	B	B	B	A	A	A	A	A	A	A	A	
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
ID	R/W	Field	Value ID	Value				Description																											
A-D	R	UD[i] (i=4..7)						Unique identifier byte i																											

4.2.4.1.10 NFC.TAGHEADER2

Address offset: 0x608

Default header for NFC Tag. Software can read these values to populate NFCID1_3RD_LAST, NFCID1_2ND_LAST and NFCID1_LAST.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D	D	D	D	D	D	D	D	C	C	C	C	C	C	C	B	B	B	B	B	B	B	B	A	A	A	A	A	A	A	A	
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
ID	R/W	Field	Value ID	Value				Description																											
A-D	R	UD[i] (i=8..11)						Unique identifier byte i																											

4.2.4.1.11 NFC.TAGHEADER3

Address offset: 0x60C

Default header for NFC Tag. Software can read these values to populate NFCID1_3RD_LAST, NFCID1_2ND_LAST and NFCID1_LAST.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D	D	D	D	D	D	D	D	C	C	C	C	C	C	C	C	B	B	B	B	B	B	B	B	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A-D	R	UD[i] (i=12..15)						Unique identifier byte i																											

4.2.4.1.12 XOSC32MTRIM

Address offset: 0x620

XOSC32M capacitor selection trim values

Note: To enable the optional internal capacitors on XC1 and XC2 pins, see to the "Using internal capacitors" section of the OSCILLATORS chapter.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B B B B B B B B B B B B B																A A A A A A A A A A A A A A A A															
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	SLOPE						Slope trim factor on twos complement form																											
								-256 = '1_0000_0000' and +255 = '0_1111_1111'																											
B	R	OFFSET						Offset trim factor on integer form																											

4.2.4.1.13 XOSC32KTRIM

Address offset: 0x624

XOSC32K capacitor selection trim values

Note: To enable the optional internal capacitors on XL1 and XL2 pins, see to the "Using internal capacitors" section of the OSCILLATORS chapter.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																
ID					B																B																A															
Reset 0xFFFFFFFF					1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1															
ID	R/W	Field	Value ID	Value	Description																																															
A	R	SLOPE			Slope trim factor on twos complement form																																															
B	R	OFFSET			Offset trim factor on integer form																																															

4.2.5 MEMCONF — Memory configuration

MEMCONF provides power control for RAM blocks.

Each RAM block can independently power up or power down in System ON and System OFF mode. RAM blocks can contain multiple RAM sections. For more information about System ON and System OFF modes, see [Power and clock management](#) on page 73. For an overview of available RAM blocks and RAM sections, see [Memory](#) on page 19.

MEMCONF registers are used for configuring the following:

- RAM sections to be retained during System OFF mode
- RAM sections to be retained and accessible during System ON mode

In System OFF mode, a RAM section is retained by configuring the corresponding MEM[i] field of registers **RET** and **RET2**. The RET and RET2 registers control retention on half the address space within the memory block.

In System ON mode, retention and accessibility for a RAM section is configured in the corresponding MEM[i] field of register **POWER[n].CONTROL** (n=0..1) on page 52.

A complete list of blocks (RET.MEM[i], RET2.MEM[i], and CONTROL.MEM[i]) are found in the following table.

Block number (index i in MEMCONF.POWER)	RAM section	RET reset value	RET2 reset value	CONTROL reset value
0	RAM00 section 0	1	x	1
1	RAM00 section 1	1	x	1
2	RAM00 section 2	1	x	1
3	RAM00 section 3	1	x	1
4	RAM01 section 0	1	x	1
5	RAM01 section 1	1	x	1
6	RAM01 section 2	1	x	1
7	RAM01 section 3	1	1	1
33	ICACHE tag + data 1:0	1	x	1
34	CRACEN PKEcode	1	x	1
35	CRACEN KeyRAM	1	x	1

Table 13: Memory block overview with MEMCONF.POWER configuration

The following table summarizes the behavior of the CONTROL and RET/RET2 fields when a power domain is powered on or off. The RAM section can be used to read and write data when it is powered. The RAM section is retained during System OFF.

Configuration			RAM section status	
Power mode	CONTROL	RET/RET2	Powered	Retained
System OFF	Any value	Off	No	No
System OFF	Off	On	No	No
System OFF	On	On	No	Yes
System ON IDLE	Off	Any value	No	No
System ON IDLE	On	Any value	No	Yes
System ON RUN	Any value	Any value	Yes	Yes

Table 14: RAM section configuration

The advantage of not retaining RAM content is reduced overall current consumption.

See chapter [Memory](#) on page 19 for more information on RAM sections.

Note: [CACHE — Instruction/data cache](#) on page 36 must be disabled when the ICACHE memory block is turned off, and only enabled after the ICACHE memory block is turned on.

4.2.5.1 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
MEMCONF : S	GLOBAL	0x500CF000	US	S	NA	No	Memory Configuration MEMCONF
MEMCONF : NS		0x400CF000					

Configuration

Instance	Domain	Configuration
MEMCONF : S	GLOBAL	
MEMCONF : NS		

Register overview

Register	Offset	TZ	Description
POWER[n].CONTROL	0x500		Control memory block power.
POWER[n].RET	0x508		RAM retention for RAM [n].
POWER[n].RET2	0x50C		RAM retention for the second bank in the RAM block

4.2.5.1.1 POWER[n].CONTROL (n=0..1)

Address offset: $0x500 + (n \times 0x10)$

Control memory block power.

Where $n = 0$ for memory blocks 0 to 31 and $n = 1$ for memory blocks 32 to 63.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				f	e	d	c	b	a	Z	Y	X	W	V	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A			
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			
ID	R/W	Field	Value	ID	Value	Description																																
A-f	RW	MEM[i] (i=0..31)				Keep the memory block MEM[i] on or off when in System ON mode.																																
						RAM blocks powered off this way will not be retained. All RAM blocks will be off in System OFF mode.																																
			Off		0	Power down																																
			On		1	Power up																																

4.2.5.1.2 POWER[n].RET (n=0..1)

Address offset: $0x508 + (n \times 0x10)$

RAM retention for RAM [n].

Where $n = 0$ for RAM blocks 0 to 31 and $n = 1$ for RAM blocks 32 to 63.

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID	f e d c b a Z Y X W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0xFFFFFFFF	1 1																															
ID	R/W	Field	Value ID	Value	Description																											
A-f	RW	MEM[i] (i=0..31)			Keep the RAM block MEM[i] retained when in System OFF mode.																											
					All other RAM will be off in System OFF mode.																											
			Off	0	Retention off																											
			On	1	Retention on																											

4.2.5.1.3 POWER[n].RET2 (n=0..1)

Address offset: 0x50C + (n × 0x10)

RAM retention for the second bank in the RAM block

Where n = 0 for RAM blocks 0 to 31 and n = 1 for RAM blocks 32 to 63.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				f	e	d	c	b	a	Z	Y	X	W	V	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A			
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A-f	RW	MEM[i] (i=0..31)			Keep the second bank in RAM block MEM[i] retained when in System OFF mode.																																	
				All other RAM will be off in System OFF mode. Not all types of RAM blocks have two banks.																																		
			Off	0	Retention off																																	
			On	1	Retention on																																	

4.2.6 RRAMC — Resistive random access memory controller

The resistive random access memory controller (RRAMC) is used for writing the internal RRAM memory, the secure information configuration region (SICR), and the user information configuration registers (UICR).

The main features of RRAMC are:

- Write and overwrite without erasing
- 128-bit word line with built in error correction code (ECC), detecting and correcting up to two bit errors per line
- Automatic standby or power-down modes
- One-time programmable (OTP) protection for user information configuration registers (UICR)
- Optional immutable boot region protection

4.2.6.1 Reading from RRAM

RRAM can be read using any natural alignment.

Read and execute operations are cachable through [CACHE — Instruction/data cache](#) on page 36. For execution performance figures, see [CPU](#) on page 26.

Low latency mode

The low power register allows making trade-offs between latency and power consumption. By default, RRAMC goes into PowerDown mode, so the wakeup time is variable. To enable a sleep mode with faster wake-up, configure Standby mode using register [POWER.LOWPOWERCONFIG.MODE](#). In combination with Constant Latency sub-power mode, this ensures the lowest latency.

4.2.6.2 Writing to RRAM

When writing is enabled in register [CONFIG.WEN](#), and [CONFIG.WRITEBUFSIZE](#) is set to `Unbuffered`, RRAM is written using any natural alignment (byte, half-word, 32-bit, or 64-bit).

RRAMC is able to write both 0 and 1 to any bit in RRAM, even if that bit has been written before.

When writing with [CONFIG.WRITEBUFSIZE](#) set to `Unbuffered`, the written data (byte, half-word, 32-bit, or 64-bit) is committed to RRAM immediately.

4.2.6.2.1 Buffered RRAM write

RRAMC enables fast buffered writes for contiguous memory regions.

RRAMC has an internal write-buffer that can be configured using [CONFIG.WRITEBUFSIZE](#).

When buffered writes are enabled, RRAMC will collect as much data as possible in the internal write-buffer, before bulk-committing the buffer to RRAM.

When committing to RRAM, the commit operation updates only the data in RRAM memory that has modified values. Values that have not been altered remain unchanged in RRAM.

To use buffered writes, perform the following operations:

1. Enable writing using [CONFIG.WEN](#), and configure [CONFIG.WRITEBUFSIZE](#).
2. Write to RRAM memory in incrementing address order.

RRAMC commits the write-buffer when either of the following occurs:

- The write-buffer is full
- The address written is outside the buffer area
- There is a read operation from a 128-bit word line in the buffer that has already been written to

RRAMC stalls while the commit takes place, and additional wait-states can be observed for the bus access.

In addition to the automatic commit, a commit can also be triggered by the following:

- After a time-out waiting for a new write operation, configured in [READYNEXTTIMEOUT](#)
- When the [COMMITWRITEBUF](#) task is triggered

The manual triggers are useful in situations where it is crucial to ensure that the write buffer has been committed. Register [BUFSTATUS.WRITEBUFEMPTY](#) can be used to check if the write-buffer is empty, or if it contains uncommitted data.

Note: The internal write-buffer is volatile, and data loss may occur during a power failure or when entering System OFF mode with uncommitted data in the buffer. Having uncommitted data in the internal write-buffer will keep the RRAM active, and thus increase power consumption during sleep mode.

4.2.6.3 Erasing RRAM

RRAMC provides a mechanism to erase the whole RRAM in one operation by using the [ERASE.ERASEALL](#) register.

When [ERASEALL](#) is triggered, RRAMC will write `0xFFFFFFFF` to the entire RRAM memory, including secure and user information configuration registers (SICR and UICR). [ERASEALL](#) will not erase the factory information configuration registers (FICR).

This functionality can be blocked by ERASE protection. For details, see [Erase all protection](#) on page 55.

Note: Unlike the CTRL-AP [ERASEALL](#) operation that can be activated from a debugger, the RRAMC [ERASEALL](#) operation will not automatically grant access to the debug access port.

Note: The write-buffer must be committed before initiating an erase operation.

4.2.6.4 Immutable boot region

RRAMC can make a part of the RRAM code memory immutable.

The immutable boot region has configurable permissions settings. Read, write, and execute permissions are configured individually. By making the region read-execute only, that memory range of the RRAM becomes immutable.

The region starts at address 0x00000000 and the size of the region is configurable. Note that the region does not add additional storage, but enforces permission settings on the memory range.

The size and permission settings of the immutable boot region is configured in [UICR](#). If [UICR.BOOTCONF](#) is not configured, RRAMC will not enforce the protection.

Once the boot region protection is enabled, it can only be removed by ERASEALL. Erase protection can be enabled to prevent ERASEALL operation. For more information about erase protection, see [CTRL-AP - Control access port](#) on page 755.

4.2.6.5 Erase all protection

RRAMC provides functionality to protect the device against the erase all function.

The following table shows the different status of protection bits, and which operations are allowed or blocked.

Protection bit status			RRAMC protection	
SECURE APPROTECT	APPROTECT	ERASE PROTECT	CTRL-AP ERASEALL	RRAMC ERASEALL
0	0	0	Available	Available
1	X	0	Available	Blocked
X	1	0	Available	Blocked
X	X	1	Blocked ¹	Blocked
1 - Enabled, 0 - Disabled, X - Don't care				

Table 15: RRAMC protection

¹Except for ERASEPROTECT.DISABLE, see [CTRL-AP - Control access port](#) on page 755.

4.2.6.6 Power-failure protection

Power failure protection is possible by using the power-fail comparator (POF) that is monitoring power supply.

If the power-fail comparator is enabled, and the power supply voltage is below the POF threshold, the power-fail comparator will prevent RRAMC from performing write operations. For more information about POF, see [Power-fail comparator](#) on page 75.

If a power failure warning is present at the start of an RRAM write operation, RRAMC will block the operation and a bus error will be signaled.

If a power failure warning occurs during an ongoing RRAM write, RRAMC can be configured to handle this in two ways:

- If `POWER.CONFIG.POF` = `Abort`, then RRAMC will stop the commit process from the internal write-buffer as soon as the condition occurs, a bus error will be signaled, and the contents of the internal write-buffer is cleared.
- If `POWER.CONFIG.POF` = `Wait`, then RRAMC will try to complete the on-going write despite the warning of the operating voltage becoming too low.

4.2.6.7 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
RRAMC	GLOBAL	0x5004B000	HF	S	NA	No	RRAM Non-Volatile Memory Controller

Configuration

Instance	Domain	Configuration
RRAMC	GLOBAL	RRAMC.REGION[4] is available for use, other regions are reserved for the system. RRAM word size : 128 bits per wordline Maximum write buffer size : 32

Register overview

Register	Offset	TZ	Description
TASKS_WAKEUP	0x000		Wakeup the RRAM from low power mode
TASKS_COMMITWRITEBUF	0x008		Commits the data stored in internal write-buffer to RRAM
SUBSCRIBE_WAKEUP	0x080		Subscribe configuration for task WAKEUP
SUBSCRIBE_COMMITWRITEBUF	0x088		Subscribe configuration for task COMMITWRITEBUF
EVENTS_WOKENUP	0x100		RRAMC is woken up from low power mode
EVENTS_READY	0x104		RRAMC is ready
EVENTS_READYNEXT	0x108		Ready to accept a new write operation
EVENTS_ACCESSERROR	0x10C		RRAM access error
PUBLISH_WOKENUP	0x180		Publish configuration for event WOKENUP
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
INTPEND	0x30C		Pending interrupts
READY	0x400		RRAMC ready status
READYNEXT	0x404		Ready next flag
ACCESSERRORADDR	0x408		Address of the first access error
BUFSTATUS.WRITEBUFEMPTY	0x418		Internal write-buffer is empty
ECC.ERRORADDR	0x420		Address of the first ECC error that could not be corrected
CONFIG	0x500		Configuration register
READYNEXTTIMEOUT	0x50C		Configuration for ready next timeout counter, in units of AXI clock frequency
POWER.CONFIG	0x510		Power configuration
POWER.LOWPOWERCONFIG	0x518		Low power mode configuration
ERASE.ERASEALL	0x540		Register for erasing whole RRAM main block, that includes the SICR and the UICR
REGION[n].ADDRESS	0x550		Region address
REGION[n].CONFIG	0x554		Region configuration

4.2.6.7.1 TASKS_WAKEUP

Address offset: 0x000

Wakeup the RRAM from low power mode

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_WAKEUP						Wakeup the RRAM from low power mode																											
			Trigger	1				Trigger task																											

4.2.6.7.2 TASKS_COMMITWRITEBUF

Address offset: 0x008

Commits the data stored in internal write-buffer to RRAM

READY status is updated during this operation

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	W	TASKS_COMMITWRITEBUF			Commits the data stored in internal write-buffer to RRAM																														
					READY status is updated during this operation																														
			Trigger	1	Trigger task																														

4.2.6.7.3 SUBSCRIBE_WAKEUP

Address offset: 0x080

Subscribe configuration for task [WAKEUP](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CHIDX		[0..255]	DPPI channel that task WAKEUP will subscribe to																														
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

4.2.6.7.4 SUBSCRIBE_COMMITWRITEBUF

Address offset: 0x088

Subscribe configuration for task [COMMITWRITEBUF](#)

READY status is updated during this operation

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task COMMITWRITEBUF will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

4.2.6.7.5 EVENTS_WOKENUP

Address offset: 0x100

RRAMC is woken up from low power mode

This event is triggered only if waken up by the [WAKEUP](#) task

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	EVENTS_WOKENUP				RRAMC is woken up from low power mode																													
						This event is triggered only if waken up by the WAKEUP task																													
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

4.2.6.7.6 EVENTS_READY

Address offset: 0x104

RRAMC is ready

This event is not connected to PPI

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_READY						RRAMC is ready																											
								This event is not connected to PPI																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

4.2.6.7.7 EVENTS_READYNEXT

Address offset: 0x108

Ready to accept a new write operation

This event is not connected to PPI

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_READYNEXT			Ready to accept a new write operation																														
					This event is not connected to PPI																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

4.2.6.7.8 EVENTS_ACCESSERROR

Address offset: 0x10C

RRAM access error

This event is not connected to PPI

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																
ID				A																																
Reset 0x00000000				0 0																																
ID	R/W	Field	Value ID	Value				Description																												
A	RW	EVENTS_ACCESSERROR							RRAM access error																											
								This event is not connected to PPI																												
				NotGenerated	0				Event not generated																											
				Generated	1				Event generated																											

4.2.6.7.9 PUBLISH_WOKENUP

Address offset: 0x180

Publish configuration for event [WOKENUP](#)

This event is triggered only if waken up by the [WAKEUP](#) task

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																													
ID				B																								A												A	A	A	A	A	A	A																		
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																										
A	RW	CHIDX		[0..255]		DPPI channel that event WOKENUP will publish to																																																										
B	RW	EN																																																														
			Disabled	0	Disable publishing																																																											
			Enabled	1	Enable publishing																																																											

4.2.6.7.10 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																						D	C	B	A			
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID			Value			Description																																			
A	RW	WOKENUP							Enable or disable interrupt for event WOKENUP																																			
									This event is triggered only if waken up by the WAKEUP task																																			

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																																				D	C	B	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value	Description																																		
			Disabled	0	Disable																																		
			Enabled	1	Enable																																		
			This event is not connected to PPI																																				
			Disabled	0	Disable																																		
B	RW	READY	Enabled	1	Enable																																		
			Enable or disable interrupt for event READY																																				
			This event is not connected to PPI																																				
			Disabled	0	Disable																																		
C	RW	READYNEXT	Enabled	1	Enable																																		
			Enable or disable interrupt for event READYNEXT																																				
			This event is not connected to PPI																																				
			Disabled	0	Disable																																		
D	RW	ACCESSERROR	Enabled	1	Enable																																		
			Enable or disable interrupt for event ACCESSERROR																																				
			This event is not connected to PPI																																				
			Disabled	0	Disable																																		
			Enabled	1	Enable																																		

4.2.6.7.11 INTENSET

Address offset: 0x304

Enable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	WOKENUP			Write '1' to enable interrupt for event WOKENUP																													
					This event is triggered only if waken up by the WAKEUP task																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
B	RW	READY			Write '1' to enable interrupt for event READY																													
					This event is not connected to PPI																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
C	RW	READYNEXT			Write '1' to enable interrupt for event READYNEXT																													
					This event is not connected to PPI																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
D	RW	ACCESSERROR			Write '1' to enable interrupt for event ACCESSERROR																													
					This event is not connected to PPI																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

4.2.6.7.12 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	WOKENUP			Write '1' to disable interrupt for event WOKENUP																													
					This event is triggered only if waken up by the WAKEUP task																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
B	RW	READY			Write '1' to disable interrupt for event READY																													
					This event is not connected to PPI																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
C	RW	READYNEXT			Write '1' to disable interrupt for event READYNEXT																													
					This event is not connected to PPI																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
D	RW	ACCESSERROR			Write '1' to disable interrupt for event ACCESSERROR																													
					This event is not connected to PPI																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

4.2.6.7.13 INTPEND

Address offset: 0x30C

Pending interrupts

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	R	WOKENUP			Read pending status of interrupt for event WOKENUP																													
					This event is triggered only if waken up by the WAKEUP task																													
			NotPending	0	Read: Not pending																													
			Pending	1	Read: Pending																													
B	R	READY			Read pending status of interrupt for event READY																													
					This event is not connected to PPI																													
			NotPending	0	Read: Not pending																													
			Pending	1	Read: Pending																													
C	R	READYNEXT			Read pending status of interrupt for event READYNEXT																													
					This event is not connected to PPI																													
			NotPending	0	Read: Not pending																													
			Pending	1	Read: Pending																													
D	R	ACCESSERROR			Read pending status of interrupt for event ACCESSERROR																													
					This event is not connected to PPI																													

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value			Description																												
			NotPending	0			Read: Not pending																												
			Pending	1			Read: Pending																												

4.2.6.7.14 READY

Address offset: 0x400

RRAMC ready status

The event **READY** is generated when the status changes to Ready

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	R	READY																		RRAMC is ready or busy															
																				The status is updated for all RRAMC operations except during read and writes to write-buffer															
			Busy	0																RRAMC is busy															
			Ready	1																The current RRAMC operation is completed and RRAMC is ready															

4.2.6.7.15 READYNEXT

Address offset: 0x404

Ready next flag

The event **READYNEXT** is generated when the READYNEXT status changes to Ready

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	READYNEXT						RRAMC can accept a new write operation																											
			Busy	0				RRAMC cannot accept any write operation now																											
			Ready	1				RRAMC is ready to accept a new write operation																											

4.2.6.7.16 ACCESSERRORADDR

Address offset: 0x408

Address of the first access error

The event **ACCESSERROR** is generated on access error. When this event is cleared, this register is updated on the next access error

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00FFFFFF				0 0 0 0 0 0 0 0 0 1																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	ADDRESS										Access error address																							
				The most significant 8 bits are set to zero always																															

4.2.6.7.17 BUFSTATUS.WRITEBUFEMPTY

Address offset: 0x418

Internal write-buffer is empty

The internal write-buffer has been committed to RRAM and is now empty

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	EMPTY																																	
			NotEmpty	0				The internal write-buffer has data that needs committing																											
			Empty	1				The internal write-buffer is empty and has no content that needs to be committed																											

4.2.6.7.18 ECC.ERRORADDR

Address offset: 0x420

Address of the first ECC error that could not be corrected

The event ECCERROR is generated on ECC error. When this event is cleared, this register is updated on the next ECC error

Bit number									31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID									A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00FFFFFF									0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value	Description																																				
A	R	ADDRESS			ECC error address																																				
The most significant 8 bits are set to zero always																																									

4.2.6.7.19 CONFIG

Address offset: 0x500

Configuration register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B B B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	WEN				Write enable																													
			Disabled	0		Write is disabled																													
			Enabled	1		Write is enabled																													
B	RW	WRITEBUFSIZE		0..32		write-buffer size in number of 128-bit words																													
			Unbuffered	0		Disable buffering																													

4.2.6.7.20 READYNEXTTIMEOUT

Address offset: 0x50C

Configuration for ready next timeout counter, in units of AXI clock frequency

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A A A A A A A A A							
Reset 0x00000080				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	VALUE		[0..4095]		Preload value for waiting for a next write																													
B	RW	EN				Enable ready next timeout																													
						The timeout value is number of RRAMC clock cycles. The timeout starts when the READYNEXT is set to ready																													
			Disable	0		Disable ready next timeout																													
			Enable	1		Enable ready next timeout																													

4.2.6.7.21 POWER.CONFIG

Address offset: 0x510

Power configuration

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A A A A A A A A A A A A A A A																															
Reset 0x00000100				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ACCESSTIMEOUT			Access timeout, in 31.25 ns units, used for going into standby power mode or remain active on wake up																														
					The timeout counter counts down and is restarted on every RRAM access and on event WOKENUP																														
B	RW	POF			Power on failure warning handling configuration																														
			Wait	0	Wait until the current RRAM write finishes																														
			Abort	1	Abort the current RRAM write																														

4.2.6.7.22 POWER.LOWPOWERCONFIG

Address offset: 0x518

Low power mode configuration

The RRAMC low power mode is entered while the device goes into system on idle

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				A																																A																											
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																																										
A	RW	MODE			RRAM low power mode																																																										
			PowerDown	0	The RRAM goes into power down mode																																																										
			Standby	1	The RRAM automatically goes into standby mode while the RRAM is not being accessed																																																										
					This mode gives faster wake-ups, and is useful in combination with Constant Latency sub-power mode.																																																										
			NAP	2	The RRAM goes into NAP mode																																																										
			PowerOff	3	The RRAM is powered Off																																																										

4.2.6.7.23 ERASE.ERASEALL

Address offset: 0x540

Register for erasing whole RRAM main block, that includes the SICR and the UICR

The status in **READY** is updated during this operation

Writes to this register are ignored when erase protect is enabled

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ERASE						Erase whole RRAM main block																											
			NoOperation	0				No operation																											
			Erase	1				Start erase of chip																											

4.2.6.7.24 REGION[n] (n=0..4)

RRAMC can apply access privileges to regions of the RRAM. Some regions are dedicated for system use and are not available for configuration - refer to the instantiation table for details.

4.2.6.7.24.1 REGION[n].ADDRESS (n=0..4)

Address offset: $0x550 + (n \times 0x8)$

Region address

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A		
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	STARTADDR			Start address of the region [n]																																	
					Bits [31:24] and bit [9:0] are read-only and set to zero																																	

4.2.6.7.24.2 REGION[n].CONFIG (n=0..4)

Address offset: $0x554 + (n \times 0x8)$

Region configuration

The register fields READ, WRITE and EXECUTE can be written to 0, even when the LOCK field is set to Enabled.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID														H	H	H	H	H	G					F						E	E	E	E	D	C	B	A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
D	R/W	Field	Value ID	Value	Description																																	
A	RW	READ			Read access																																	
			NotAllowed	0	Read access to override region [n] is not allowed																																	
			Allowed	1	Read access to override region [n] is allowed																																	
B	RW	WRITE			Write access																																	
			NotAllowed	0	Write access to override region [n] is not allowed																																	
			Allowed	1	Write access to override region [n] is allowed																																	
C	RW	EXECUTE			Execute access																																	
			NotAllowed	0	Execute access to override region [n] is not allowed																																	
			Allowed	1	Execute access to override region [n] is allowed																																	
D	RW	SECURE			Secure access																																	
			NonSecure	0	Both Secure and non-Secure access to override region [n] is allowed																																	
			Secure	1	Only secure access to override region [n] is allowed																																	
E	RW	OWNER			Owner ID																																	

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				H H H H H																G F				E E E E D C B A											
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
			NotEnforced	0		Owner ID protection is not enforced																													
F	RW	WRITEONCE																																	
						Write-once																													
			Disabled	0		Write-once disabled																													
			Enabled	1		Write-once enabled																													
						Writes to a 32-bit word in region [n] are allowed only when the current data is 0xFFFFFFFF, else the writes are ignored																													
G	RW	LOCK W1S																																	
			Disabled	0		Lock disabled for region [n]																													
			Enabled	1		Lock enabled for region [n]																													
H	RW	SIZE			Size in KBytes of region [n]																														

4.2.7 SICR — Secure information configuration region

The secure information configuration region (SICR) is reserved for keys and device unique seed.

Access to SICR is managed by [KMU — Key management unit](#) on page 166. Bus transactions originating from CPU or other peripherals are blocked.

4.2.8 SWI — Software interrupts

A set of interrupts have been reserved for use as software interrupts.

These interrupts can be enabled and triggered by software by using the Arm Cortex-M33 NVIC registers, as described in the *Arm Cortex-M33 Processor Technical Reference Manual*.

4.2.8.1 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
SWI00	APPLICATION	0x5001C000	HF	S	NA	No	Software interrupt SWI00
SWI01	APPLICATION	0x5001D000	HF	S	NA	No	Software interrupt SWI01
SWI02	APPLICATION	0x5001E000	HF	S	NA	No	Software interrupt SWI02
SWI03	APPLICATION	0x5001F000	HF	S	NA	No	Software interrupt SWI03

4.2.9 UICR — User information configuration registers

The user information configuration registers (UICR) are non-volatile memory (NVM) registers for configuring emulated one-time programmable (OTP) user specific settings and values.

All UICR registers have a RW1 protection, which means that they can be read multiple times, but written only once when UICR has been erased by the Erase All operation.

For information on writing registers, see [RRAMC — Resistive random access memory controller](#) on page 53 and [Memory](#) on page 19.

4.2.9.1 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
UICR	GLOBAL	0x00FFD000	HF	S	NA	No	User information configuration

Register overview

Register	Offset	TZ	Description
APPROTECT[n].PROTECT0	0x000		Access port protection
APPROTECT[n].PROTECT1	0x01C		Access port protection
SECUREAPPROTECT[n].PROTECT0	0x020		Access port protection
SECUREAPPROTECT[n].PROTECT1	0x03C		Access port protection register
AUXAPPROTECT[n].PROTECT0	0x040		Access port protection
AUXAPPROTECT[n].PROTECT1	0x05C		Access port protection register
ERASEPROTECT[n].PROTECT0	0x060		Erase protection
ERASEPROTECT[n].PROTECT1	0x07C		Erase protection
BOOTCONF	0x080		Immutable boot region configuration.
USER.ROT.PUBKEY[n].DIGEST[o]	0x200		First 256 bits of SHA2-512 digest over RoT public key generation [n].
USER.ROT.PUBKEY[n].REVOKE[o]	0x220		Revocation status for RoT public key generation [n].
USER.ROT.AUTHOPKEY[n].DIGEST[o]	0x2B0		First 256 bits of SHA2-512 digest over RoT authenticated operation public key generation [n].
USER.ROT.AUTHOPKEY[n].REVOKE[o]	0x2D0		Revocation status for RoT authenticated operation public key generation [n].
OTP[n]	0x500		One time programmable memory

4.2.9.1.1 APPROTECT[n] (n=0..0)

Access Port Protection Registers

4.2.9.1.1.1 APPROTECT[n].PROTECT0 (n=0..0)

Address offset: 0x000 + (n × 0x20)

Access port protection

Any other value than Unprotected will lock TAMPC PROTECT.DOMAIN signal protectors.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value	ID	Value				Description																										
A	RW1	PALL		Unprotected	0xFFFFFFFF				Leaves TAMPC PROTECT.DOMAIN DBGEN and NIDEN signal protectors unlocked and under CPU control.																										

4.2.9.1.1.2 APPROTECT[n].PROTECT1 (n=0..0)

Address offset: 0x01C + (n × 0x20)

Access port protection

Any other value than Unprotected will lock TAMPC PROTECT.DOMAIN signal protectors.

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																													
ID	A A																													
Reset 0xFFFFFFFF	1 1																													
ID	R/W	Field	Value ID	Value	Description																									
A	RW1	PALL	Unprotected	0xFFFFFFFF	Leaves TAMPC PROTECT.DOMAIN DBGEN and NIDEN signal protectors unlocked and under CPU control.																									

4.2.9.1.2 SECUREAPPROTECT[n] (n=0..0)

Access Port Protection Registers

4.2.9.1.2.1 SECUREAPPROTECT[n].PROTECT0 (n=0..0)

Address offset: 0x020 + (n × 0x20)

Access port protection

Any other value than Unprotected will lock TAMPC PROTECT.DOMAIN signal protectors.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW1	PALL	Unprotected	0xFFFFFFFF								Leaves TAMPC PROTECT.DOMAIN SPIDEN and SPNIDEN signal protectors unlocked and under CPU control.																							

4.2.9.1.2.2 SECUREAPPROTECT[n].PROTECT1 (n=0..0)

Address offset: 0x03C + (n × 0x20)

Access port protection register

Any other value than Unprotected will lock TAMPC PROTECT.DOMAIN signal protectors.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																			
ID				A A																																			
Reset 0xFFFFFFFF				1 1																																			
ID	R/W	Field	Value ID	Value				Description																															
A	RW1	PALL		Unprotected				0xFFFFFFFF				Leaves TAMPC PROTECT.DOMAIN SPIDEN and SPNIDEN signal protectors unlocked and under CPU control.																											

4.2.9.1.3 AUXAPPROTECT[n] (n=0..0)

Access Port Protection Registers

4.2.9.1.3.1 AUXAPPROTECT[n].PROTECT0 (n=0..0)

Address offset: 0x040 + (n × 0x20)

Access port protection

Any other value than Unprotected will lock TAMPC PROTECT.AP signal protectors.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	RW1	PALL	Unprotected	0xFFFFFFFF				Leaves TAMPC PROTECT.AP DBGEN signal protector unlocked and under CPU control.																											

4.2.9.1.3.2 AUXAPPROTECT[n].PROTECT1 (n=0..0)

Address offset: 0x05C + (n × 0x20)

Access port protection register

Any other value than Unprotected will lock TAMPC PROTECT.AP signal protectors.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW1	PALL	Unprotected	0xFFFFFFFF								Leaves TAMPC PROTECT.AP DBGEN signal protector unlocked and under CPU control.																							

4.2.9.1.4 ERASEPROTECT[n] (n=0..0)

Erase Protection Registers

4.2.9.1.4.1 ERASEPROTECT[n].PROTECT0 (n=0..0)

Address offset: 0x060 + (n × 0x20)

Erase protection

Any other value than Protected will leave the TAMPC PROTECT.ERASEPROTECT signal protector unlocked, so that CPU can control its value.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

4.2.9.1.4.2 ERASEPROTECT[n].PROTECT1 (n=0..0)

Address offset: 0x07C + (n × 0x20)

Erase protection

Any other value than Protected will leave the TAMPC PROTECT.ERASEPROTECT signal protector unlocked, so that CPU can control its value.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
ID	R/W	Field	Value ID	Value				Description																											
A	RW1	PALL	Protected	0x50FA50FA				The device can be erased using the CTRL-AP Erase all function and TAMPC PROTECT.ERASEPROTECT signal protector is unlocked.																											

4.2.9.1.5 BOOTCONF

Address offset: 0x080

Immutable boot region configuration.

If this register is not equal to 0xFFFFFFFF, RRAMC applies these settings to form the immutable boot region.

Unused bits must be set to zero.

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			G G G G G																F E		D C B A													
Reset 0xFFFFFFFF			1 1																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW1	READ			Read access																													
			NotAllowed	0	Reading from the region is not allowed																													
			Allowed	1	Reading from the region is allowed																													
B	RW1	WRITE			Write access																													
			NotAllowed	0	Writing to the region is not allowed																													
			Allowed	1	Writing to the region is allowed																													
C	RW1	EXECUTE			Execute access																													
			NotAllowed	0	Executing code from the region is not allowed																													
			Allowed	1	Executing code from the region is allowed																													
D	RW1	SECURE			Secure access																													
			NonSecure	0	Both secure and non-secure access to region is allowed																													
			Secure	1	Only secure access to region is allowed																													
E	RW1	WRITEONCE			Write-once																													
			Disabled	0	Write-once disabled																													
			Enabled	1	Write-once enabled																													
					Writes to a 32-bit word in the BOOTCONF region are only when the current data is 0xFFFFFFFF, otherwise the writes are ignored																													
F	RW1	LOCK			Enable lock of configuration register																													
			Disabled	0	Lock is disabled, and the RRAMC region configuration registers for the immutable boot region are writable.																													
			Enabled	1	Lock is enabled, and the RRAMC configuration registers for the immutable boot region are read-only.																													
G	RW1	SIZE			Immutable boot region size																													
					Configures the region size in kB																													

4.2.9.1.6 USER.ROT

Assets installed to establish initial Root of Trust in the device.

User RoT key materials

4.2.9.1.6.1 USER.ROT.PUBKEY[n].DIGEST[o] (n=0..3) (o=0..7)

Address offset: 0x200 + (n × 0x2C) + (o × 0x4)

First 256 bits of SHA2-512 digest over RoT public key generation [n].

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	RW1	VALUE						Value for word [o] in the key digest [n].																											

4.2.9.1.6.2 USER.ROT.PUBKEY[n].REVOKE[o] (n=0..3) (o=0..2)

Address offset: $0x220 + (n \times 0x2C) + (o \times 0x4)$

Revocation status for RoT public key generation [n].

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

4.2.9.1.6.3 USER.ROT.AUTHOPKEY[n].DIGEST[o] (n=0..3) (o=0..7)

Address offset: $0x2B0 + (n \times 0x2C) + (o \times 0x4)$

First 256 bits of SHA2-512 digest over RoT authenticated operation public key generation [n].

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	RW1	VALUE						Value for word [o] in the key digest [n].																											

4.2.9.1.6.4 USER.ROT.AUTHOPKEY[n].REVOKE[o] (n=0..3) (o=0..2)

Address offset: $0x2D0 + (n \times 0x2C) + (o \times 0x4)$

Revocation status for RoT authenticated operation public key generation [n].

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	RW1	STATUS						Revocation status.																											
			NotRevoked	0xFFFFFFFF				Key not revoked.																											
								Any other value says the key is revoked.																											

4.2.9.1.7 OTP[n] (n=0..319)

Address offset: $0x500 + (n \times 0x4)$

One time programmable memory

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																													
ID	A A																													
Reset 0xFFFFFFFF	1 1																													
ID	R/W	Field	Value ID	Value		Description																								

A RW1 OTP

OTP word

Can only be written to a non 0xFFFFFFFF value once after Erase All operation.

5 Power and clock management

The power and clock management system is optimized for ultra-low power applications to provide maximum power efficiency.

The power and clock management system is shown in the following figure.

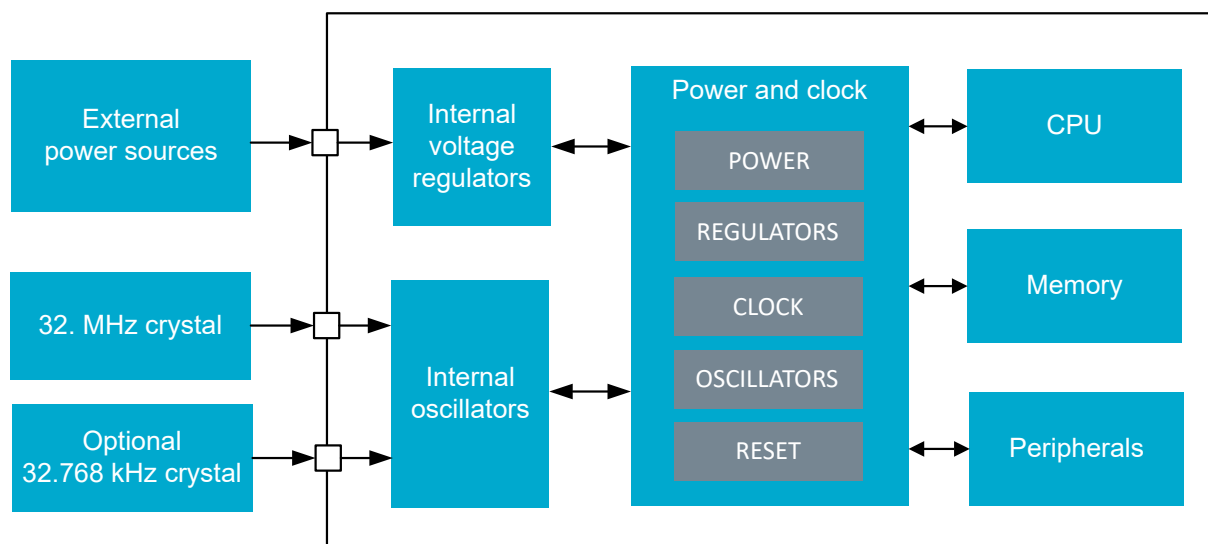


Figure 9: Power and clock management

The power and clock management system automatically tracks the power and clock resources requested by components in the system at any given time. To achieve the lowest power consumption possible, the system evaluates the requests, starts and stops clock sources, and chooses the most optimal regulator operation modes.

The device start-up sequence after reset is described in [RESET — Reset control](#) on page 106.

5.1 System ON mode

System ON is the default operation mode after power-on reset.

In System ON, all functional blocks, such as the CPU and peripherals, can be in an IDLE or RUN state depending on the configuration set by the software and the state of the executing application.

The power and clock management unit can switch the appropriate internal power domains on and off, depending on power requirements. A peripheral's power requirement is directly related to its activity level, which increases and decreases when specific tasks are triggered or events are generated.

5.1.1 Sub-power modes

In System ON mode, the system can reside in one of the two sub-power modes when the CPU and all peripherals are in an IDLE state.

The sub-power modes are:

- Constant Latency
- Low-power

In Constant Latency mode, the CPU wakeup latency and the PPI task response are constant and kept at a minimum. This is secured by forcing a set of basic resources to be turned on while in sleep mode. The cost

of constant and predictable latency is increased power consumption. Constant Latency mode is selected by triggering the task [CONSTLAT](#).

In Low-power mode, the automatic power management system described in System ON mode ensures that the most efficient supply option is chosen to save power. The cost of having the lowest possible power consumption is a varying CPU wakeup latency and PPI task response. Low-power mode is selected by triggering the task [LOWPWR](#).

When the system enters System ON mode, it is by default in the sub-power mode Low-power.

5.2 System OFF mode

System OFF is the deepest power-saving mode the system can enter. In this mode, the system's core functionality is powered down and all ongoing tasks are stopped.

Register [SYSTEMOFF](#) on page 104 sets the device into System OFF mode. The following wakeup sources will initiate a wakeup from System OFF:

- The DETECT signal generated by the GPIO peripheral
- The ANADETECT signal generated by the LPCOMP peripheral
- The SENSE signal generated by the NFCT peripheral to wake-on-field
- The SYSCOUNTER compare event generated by the GRTC peripheral
- A debug session is started
- A pin reset

When the device wakes up from System OFF, a system reset is performed. For more details, see [Reset behavior](#) on page 108.

One or more RAM sections can be retained in System OFF depending on the RAM retention settings configured in [MEMCONF — Memory configuration](#) on page 50.

Before entering System OFF mode, the following conditions must be met.

- All on-going EasyDMA transactions must finish. See peripheral specific chapters for more information about how to get the status of EasyDMA transactions.
- The register [RESET.RESETREAS](#) must be cleared. Failure to do so can make the system immediately wake up from System OFF mode.

5.2.1 Emulated System OFF mode

When the device is in Debug Interface mode, System OFF is emulated to ensure that all resources required for debugging are available during System OFF.

Resources required for debugging include the following key components:

- [Debug interface mode](#) on page 753
- [CLOCK — Clock control](#) on page 76
- [POWER — Power control](#) on page 97
- [OSCILLATORS — Oscillator control](#) on page 91
- [REGULATORS — Regulator control](#) on page 102
- [RESET — Reset control](#) on page 106
- CPU
- Memory, including RAM and RRAM

Because the CPU is kept on in an emulated System OFF mode, it is recommended to add an infinite loop directly after entering System OFF. This prevents the CPU from executing code that normally should not be executed. For more information, see [Debug and trace](#) on page 749.

5.3 Power supply supervisors

The power supply supervisors monitor the connected power supply.

The power supply supervisors provide the following functionality:

- Power-on reset — signals the circuit when a supply is connected
- Fixed brownout reset detector — holds the system in reset when the voltage is too low for safe operation
- Optional power-fail comparator (POF) — signals the application when the supply voltage drops below a configured threshold

The power supply supervisors are illustrated in the following figure.

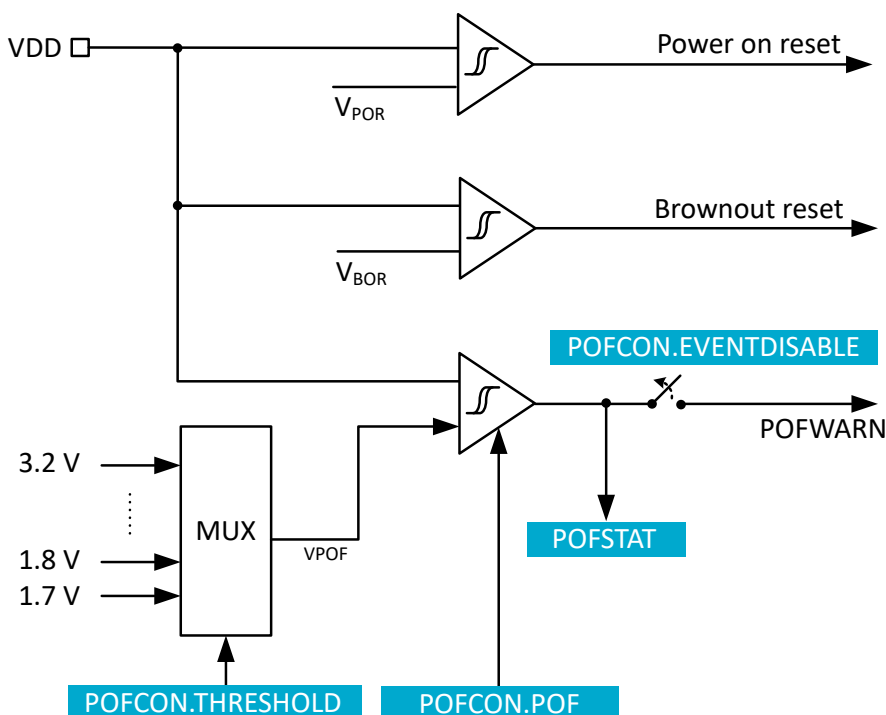


Figure 10: Power supply supervisors

5.3.1 Power-fail comparator

Using the power-fail comparator (POF) is optional. When enabled, it notifies the CPU of a potential power supply failure.

It can also be used to measure the voltage on **VDD**. To enable and configure the power-fail comparator, see register [POFCON \(Retained\)](#) on page 104.

When the supply voltage falls below the defined threshold, the power-fail comparator generates an event `POFWARN` that can be used by an application to prepare for power failure. This event is also generated when the supply voltage is already below the threshold at the time the power-fail comparator is enabled, or if the threshold is reconfigured to a level above the supply voltage. `POFWARN` is disabled using the `EVENTDISABLE` field of [REGULATORS.POFCON](#). In addition to the event, the result of the power-fail comparator is found using `POFSTAT` on page 105.

`POFWARN` prevents RRAMC from performing write operations to the non-volatile memory. See [RRAMC — Resistive random access memory controller](#) on page 53 for more information about non-volatile memory.

The power-fail comparator features a hysteresis of V_{HYST} , as illustrated in the following figure.

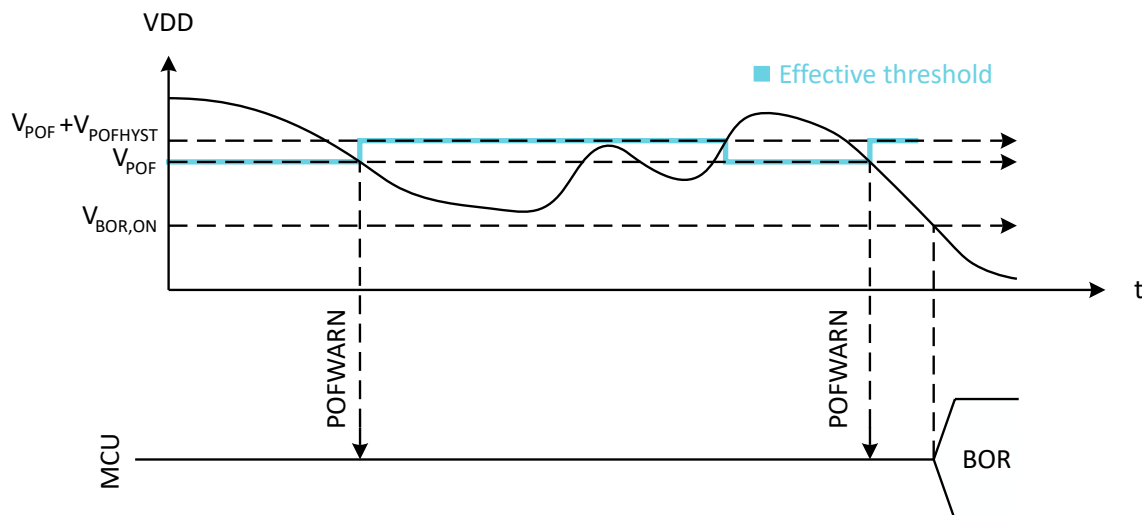


Figure 11: Power-fail comparator (BOR = Brownout reset)

To save power, the power-fail comparator is not active in System OFF, and it is not active in System ON when HFCLK is not running.

POF also supports measuring the voltage on **VDD**. To measure the voltage, perform the following steps.

1. Disable POFWARN by writing Disabled to `REGULATORS.POFCON.EVENTDISABLE`.
2. Enable POF by writing Enabled to `REGULATORS.POFCON.POF`.
3. Loop over all threshold voltages by writing a threshold voltage into register `REGULATORS.POFCON.THRESHOLD`, starting at the lowest value enumerator until `REGULATORS.POFSTAT` toggles. This toggle indicates that the voltage to measure has been found and can be read from register `REGULATORS.POFCON.THRESHOLD`.

5.4 CLOCK — Clock control

The clock control system sources the system clocks from internal or external high and low frequency oscillators. It distributes the clocks to modules based on module requirements. Clock distribution is automated and grouped independently by module to limit current consumption in unused branches of the clock tree.

The following are the main features for CLOCK:

- On-chip 128 MHz phase-locked loop (PLL) with internal oscillator
- 32 MHz crystal oscillator, when using the external 32 MHz crystal
- 32.768 kHz RC oscillator
- 32.768 kHz crystal oscillator, when using the external 32.768 kHz crystal
- Automatic clock control and distribution

The clock control system is responsible for requesting resources from the power and clock subsystem.

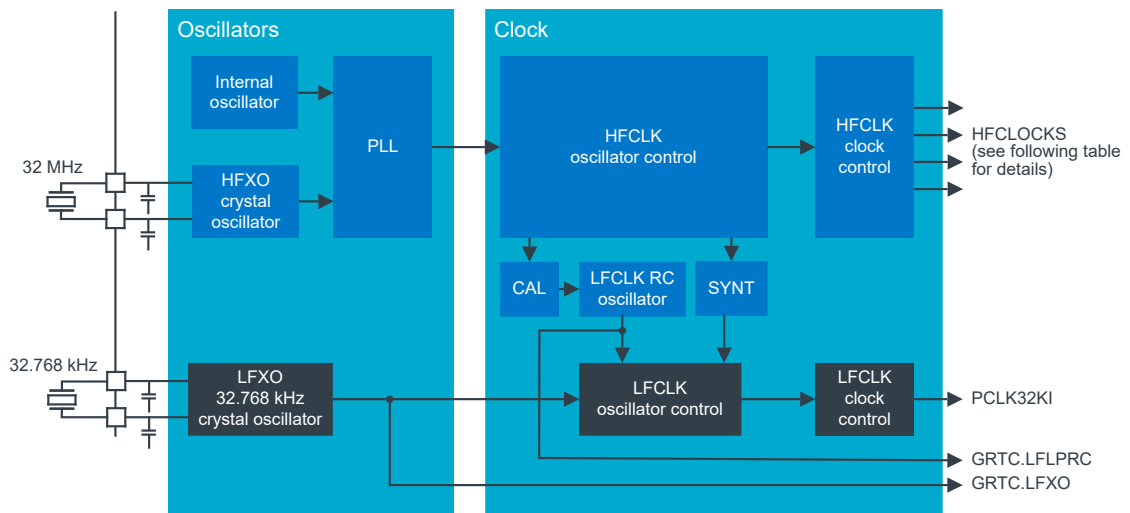


Figure 12: Clock control

5.4.1 HFCLK controller

The HFCLK clock controller provides the following clocks to the system.

Clock	Description
HCLK128M	MCU power domain and CPU clock where 64 MHz or 128 MHz can be selected
PCLK32M	32 MHz peripheral clock
PCLK16M	16 MHz peripheral clock
PCLK1M	1 MHz peripheral clock

Table 16: Clocks

Clock source	Description
HFINT	32 MHz internal oscillator
HFXO	32 MHz crystal oscillator

Table 17: Sources

The following HFCLK sources are used to generate the HFCLK clocks:

- 128 MHz internal oscillator — PLL is operating in free running mode
- 32 MHz crystal oscillator — PLL is locked on a crystal (XOSC), optionally using built-in capacitors as described in [OSCILLATORS — Oscillator control](#) on page 91.

CPUs, peripherals, and other system components automatically request clocks. The HFCLK control passes the request to the power and clock subsystem. When the clocks are running, the HFCLK control distributes them to the components. The CPU clock frequency can be selected, as described in [OSCILLATORS — Oscillator control](#) on page 91.

When all HFCLK control requests end, the HFCLK control stops requesting CLOCK from the power and clock subsystem. For example, when the CPU enters sleep or when peripherals have completed their tasks, HFCLK stops CLOCK requests. If there are no requests for HFCLK or PCLK control, the power and clock subsystem automatically stops the clock.

When the system enters System ON mode and an HFCLK clock is requested, the PLL is automatically started. When clock requests stop, the PLL automatically stops.

HFCLK clocks are only available to the HFCLK controllers when the system is in System ON mode.

An HFCLK source can run before being started by the relevant clock request. This reduces start-up time but causes increased power consumption. An example of this would be to keep the PLL running during sleep by using the task [PLLSTART](#).

The XOSC must be started when crystal clock accuracy is required. The crystal is started by triggering the task [XOSTART](#). When the crystal reaches the correct amplitude and frequency, the PLL automatically locks to the crystal and generates the event [XOSTARTED](#). At the same time, the crystal oscillator is performing an XOTUNE. When that process completes, the signal from the crystal oscillator is of its highest quality, and the event [XOTUNED](#) is generated.

Note: The crystal oscillator quality indicated by the XOSTARTED event is sufficient for all peripherals except RADIO. Before using RADIO, ensure that the event [XOTUNED](#) has been generated. This ensures the highest quality crystal signal is available.

If the crystal oscillator requires the XOTUNE process to be repeated, the device generates the event [XOTUNEERROR](#). When that happens, the [XOTUNE](#) task must be triggered. Do not trigger this task at the same time that RADIO is running (meaning RADIO must not be in the RX or TX states).

A new START task can be initiated after one has already been triggered, and before the corresponding STARTED event is generated. In this case, only one STARTED event will be generated, corresponding to the last triggered START task. Triggering a START task after the STARTED event from a previous triggered START tasks is generated, generates a new STARTED event.

Time from a START task to the corresponding STARTED event may differ depending on whether the HFCLK source is already running or in the process of starting. The amount of time before a STARTED event may vary when a different HFCLK source is configured before triggering a new START task. Different crystal types also have different start-up times, see [OSCILLATORS — Oscillator control](#) on page 91 for details.

HFXO must be running to use RADIO, NFCT, or to calibrate the 32.768 kHz RC oscillator. Using HFXO will also improve SAADC performance by reducing clock jitter.

5.4.2 LFCLK controller

The system supports the following low frequency clock sources, as described in [Clock control](#) on page 77.

- 32.768 kHz RC oscillator (LFRC)
- 32.768 kHz crystal oscillator (LFXO)
- 32.768 kHz synthesized from HFCLK (LFSYNT)

LFXO can run in System OFF mode. The other clock sources only run in System ON mode.

The following LF clocks are available in the system.

Clock	Description
PCLK32KI	32.768 kHz peripheral low-frequency clock
GRTC.LFLPRC	Direct path from 32.768 kHz internal oscillator (LFRC) to GRTC peripheral. Available in System ON modes.
GRTC.LFXO	Direct path from 32.768 kHz crystal oscillator (LFXO) to GRTC peripheral. Available in System ON or OFF modes.

Table 18: Clocks

When a peripheral requires the PCLK32KI clock, the LFCLK control automatically requests the LFCLK clock to the power and clock subsystem. The default LFCLK source is the LFRC.

To use a different LFCLK source, select the preferred clock source in register [LFCLK.SRC](#) on page 90 and then trigger the [LFCLKSTART](#) task. If LFXO is selected as the clock source, LFCLK initially starts running from the 32.768 kHz LFRC then automatically switches to the crystal once available. The [LFCLKSTARTED](#) event is then generated.

The [LFCLKSTART](#) task will request the clock to keep running until triggering the [LFCLKSTOP](#) task to stop the clock.

The LFCLK clock is stopped when there are no requests. For example, [WDT](#) is stopped, and the [LFCLKSTOP](#) task is triggered. Triggering the [LFCLKSTOP](#) task is required after the [LFCLKSTART](#) task has been triggered.

5.4.2.1 Calibrating the 32.768 kHz RC oscillator

The LFRC frequency is affected by temperature variation. LFRC can be calibrated to improve accuracy by using HFCLK as a reference oscillator during calibration.

The calibration must use the following sequence.

1. Start the LFCLK by using the [LFCLKSTART](#) task.
2. Start the HFCLK crystal oscillator HFXO by triggering the [XOSTART](#) task.
3. Wait for the [LFCLKSTARTED](#) and the HFXO [XOTUNED](#) events.
4. Trigger the [CAL](#) task to start the calibration process. The device automatically performs the calibration, adjusting the LFCLK frequency using HFCLK as reference. The [DONE](#) event is generated when calibration finishes.
5. Stop HFXO with the [XOSTOP](#) task.
6. Stop LFCLK with the [LFCLKSTOP](#) task.

LFCLK uses the calibrated value until the next calibration.

5.4.3 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
CLOCK : S	GLOBAL	0x5010E000	US	S	NA	No	Clock control
CLOCK : NS		0x4010E000					

Register overview

Register	Offset	TZ	Description
TASKS_XOSTART	0x000		Start crystal oscillator (HFXO)
TASKS_XOSTOP	0x004		Stop crystal oscillator (HFXO)
TASKS_PLLSTART	0x008		Start PLL and keep it running, regardless of the automatic clock requests
TASKS_PLLSTOP	0x00C		Stop PLL
TASKS_LFCLKSTART	0x010		Start LFCLK source as selected in LFCLK.SRC
TASKS_LFCLKSTOP	0x014		Stop LFCLK source
TASKS_CAL	0x018		Start calibration of LFRC oscillator
TASKS_XOTUNE	0x01C		Request tuning for HFXO
TASKS_XOTUNEABORT	0x020		Abort tuning for HFXO
SUBSCRIBE_XOSTART	0x080		Subscribe configuration for task XOSTART
SUBSCRIBE_XOSTOP	0x084		Subscribe configuration for task XOSTOP

Register	Offset	TZ	Description
SUBSCRIBE_PLLSTART	0x088		Subscribe configuration for task PLLSTART
SUBSCRIBE_PLLSTOP	0x08C		Subscribe configuration for task PLLSTOP
SUBSCRIBE_LFCLKSTART	0x090		Subscribe configuration for task LFCLKSTART
SUBSCRIBE_LFCLKSTOP	0x094		Subscribe configuration for task LFCLKSTOP
SUBSCRIBE_CAL	0x098		Subscribe configuration for task CAL
EVENTS_XOSTARTED	0x100		Crystal oscillator has started
EVENTS_PLLSTARTED	0x104		PLL started
EVENTS_LFCLKSTARTED	0x108		LFCLK source started
EVENTS_DONE	0x10C		Calibration of LFRC oscillator complete event
EVENTS_XOTUNED	0x110		HFXO tuning is done. XOTUNED is generated after TASKS_XOSTART or after TASKS_XOTUNE has completed
EVENTS_XOTUNEERROR	0x114		HFXO quality issue detected, XOTUNE is needed
EVENTS_XOTUNEFALIED	0x118		HFXO tuning could not be completed
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
INTPEND	0x30C		Pending interrupts
XO.RUN	0x408		Indicates that XOSTART task was triggered
XO.STAT	0x40C		XO status
PLL.RUN	0x428		Indicates that PLLSTART task was triggered
PLL.STAT	0x42C		Which PLL settings were selected when triggering START task
LFCLK.SRC	0x440		Clock source for LFCLK
LFCLK.RUN	0x448		Indicates that LFCLKSTART task was triggered
LFCLK.STAT	0x44C		Copy of LFCLK.SRCCOPY register, set when LFCLKSTARTED event is triggered.
LFCLK.SRCCOPY	0x450		Copy of LFCLK.SRC register, set when LFCLKSTART task is triggered

5.4.3.1 TASKS_XOSTART

Address offset: 0x000

Start crystal oscillator (HFXO)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_XOSTART						Start crystal oscillator (HFXO)																											
			Trigger	1				Trigger task																											

5.4.3.2 TASKS_XOSTOP

Address offset: 0x004

Stop crystal oscillator (HFXO)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_XOSTOP						Stop crystal oscillator (HFXO)																											
			Trigger	1				Trigger task																											

5.4.3.3 TASKS_PLLSTART

Address offset: 0x008

Start PLL and keep it running, regardless of the automatic clock requests

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	W	TASKS_PLLSTART			Start PLL and keep it running, regardless of the automatic clock requests																														
			Trigger	1	Trigger task																														

5.4.3.4 TASKS_PLLSTOP

Address offset: 0x00C

Stop PLL

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	TASKS_PLLSTOP			Stop PLL																															
			Trigger	1	Trigger task																															

5.4.3.5 TASKS_LFCLKSTART

Address offset: 0x010

Start LFCLK source as selected in LFCLK.SRC

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_LFCLKSTART						Start LFCLK source as selected in LFCLK.SRC																											
			Trigger	1				Trigger task																											

5.4.3.6 TASKS_LFCLKSTOP

Address offset: 0x014

Stop LFCLK source

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	TASKS_LFCLKSTOP			Stop LFCLK source																															
			Trigger	1	Trigger task																															

5.4.3.7 TASKS_CAL

Address offset: 0x018

Start calibration of LFRC oscillator

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_CAL						Start calibration of LFRC oscillator																											
			Trigger	1				Trigger task																											

5.4.3.8 TASKS_XOTUNE

Address offset: 0x01C

Request tuning for HFXO

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_XOTUNE						Request tuning for HFXO																											
			Trigger	1				Trigger task																											

5.4.3.9 TASKS_XOTUNEABORT

Address offset: 0x020

Abort tuning for HFXO

Bit number										31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																											
ID										A																											
Reset 0x00000000										0 0																											
ID	R/W	Field	Value ID	Value				Description																													
A	W	TASKS_XOTUNEABORT				Abort tuning for HFXO																															
			Trigger	1				Trigger task																													

5.4.3.10 SUBSCRIBE_XOSTART

Address offset: 0x080

Subscribe configuration for task [XOSTART](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				B																								A				A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	CHIDX		[0..255]		DPPI channel that task XOSTART will subscribe to																															
B	RW	EN																																			
			Disabled	0	Disable subscription																																
			Enabled	1	Enable subscription																																

5.4.3.11 SUBSCRIBE_XOSTOP

Address offset: 0x084

Subscribe configuration for task [XOSTOP](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task XOSTOP will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

5.4.3.12 SUBSCRIBE_PLLSTART

Address offset: 0x088

Subscribe configuration for task **PLLSTART**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task PLLSTART will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

5.4.3.13 SUBSCRIBE_PLLSTOP

Address offset: 0x08C

Subscribe configuration for task **PLLSTOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task PLLSTOP will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

5.4.3.14 SUBSCRIBE_LFCLKSTART

Address offset: 0x090

Subscribe configuration for task **LFCLKSTART**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task LFCLKSTART will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

5.4.3.15 SUBSCRIBE_LFCLKSTOP

Address offset: 0x094

Subscribe configuration for task LFCLKSTOP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task LFCLKSTOP will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

5.4.3.16 SUBSCRIBE_CAL

Address offset: 0x098

Subscribe configuration for task CAL

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task CAL will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

5.4.3.17 EVENTS_XOSTARTED

Address offset: 0x100

Crystal oscillator has started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_XOSTARTED			Crystal oscillator has started																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

5.4.3.18 EVENTS_PLLSTARTED

Address offset: 0x104

PLL started

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																	A									
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																					
A	RW	EVENTS_PLLSTARTED			PLL started																																					
			NotGenerated	0	Event not generated																																					
			Generated	1	Event generated																																					

5.4.3.19 EVENTS_LFCLKSTARTED

Address offset: 0x108

LFCLK source started

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				A																																	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	EVENTS_LFCLKSTARTED				LFCLK source started																															
			NotGenerated	0	Event not generated																																
			Generated	1	Event generated																																

5.4.3.20 EVENTS_DONE

Address offset: 0x10C

Calibration of LFRC oscillator complete event

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_DONE						Calibration of LFRC oscillator complete event																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

5.4.3.21 EVENTS_XOTUNED

Address offset: 0x110

HFXO tuning is done. XOTUNED is generated after TASKS_XOSTART or after TASKS_XOTUNE has completed

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																								A			
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																						
A	RW	EVENTS_XOTUNED			HFXO tuning is done. XOTUNED is generated after TASKS_XOSTART or after TASKS_XOTUNE has completed																																						
			NotGenerated	0	Event not generated																																						
			Generated	1	Event generated																																						

5.4.3.22 EVENTS_XOTUNEERROR

Address offset: 0x114

HFXO quality issue detected, XOTUNE is needed

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_XOTUNEERROR						HFXO quality issue detected, XOTUNE is needed																											
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

5.4.3.23 EVENTS_XOTUNEFAILED

Address offset: 0x118

HFXO tuning could not be completed

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_XOTUNEFAILED						HFXO tuning could not be completed																											
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

5.4.3.24 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				G F E D C B A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																														
A	RW	XOSTARTED			Enable or disable interrupt for event XOSTARTED																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
B	RW	PLLSTARTED			Enable or disable interrupt for event PLLSTARTED																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
C	RW	LFCLKSTARTED			Enable or disable interrupt for event LFCLKSTARTED																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
D	RW	DONE			Enable or disable interrupt for event DONE																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
E	RW	XOTUNED			Enable or disable interrupt for event XOTUNED																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
F	RW	XOTUNEERROR			Enable or disable interrupt for event XOTUNEERROR																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
G	RW	XOTUNEFAILED			Enable or disable interrupt for event XOTUNEFAILED																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

5.4.3.25 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	XOSTARTED			Write '1' to enable interrupt for event XOSTARTED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	PLLSTARTED			Write '1' to enable interrupt for event PLLSTARTED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	LFCLKSTARTED			Write '1' to enable interrupt for event LFCLKSTARTED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	DONE			Write '1' to enable interrupt for event DONE																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
E	RW	XOTUNED			Write '1' to enable interrupt for event XOTUNED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
F	RW	XOTUNEERROR			Write '1' to enable interrupt for event XOTUNEERROR																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
G	RW	XOTUNEFAILED			Write '1' to enable interrupt for event XOTUNEFAILED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

5.4.3.26 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	XOSTARTED			Write '1' to disable interrupt for event XOSTARTED																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	PLLSTARTED			Write '1' to disable interrupt for event PLLSTARTED																														
			Clear	1	Disable																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	LFCLKSTARTED																																	
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	DONE																																	
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
E	RW	XOTUNED																																	
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
F	RW	XOTUNEERROR																																	
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
G	RW	XOTUNEFAILED																																	
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

5.4.3.27 INTPEND

Address offset: 0x30C

Pending interrupts

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	XOSTARTED			Read pending status of interrupt for event XOSTARTED																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
B	R	PLLSTARTED			Read pending status of interrupt for event PLLSTARTED																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
C	R	LFCLKSTARTED			Read pending status of interrupt for event LFCLKSTARTED																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
D	R	DONE			Read pending status of interrupt for event DONE																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
E	R	XOTUNED			Read pending status of interrupt for event XOTUNED																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
F	R	XOTUNEERROR			Read pending status of interrupt for event XOTUNEERROR																														
			NotPending	0	Read: Not pending																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
			Pending	1	Read: Pending																														
G	R	XOTUNEFAILED			Read pending status of interrupt for event XOTUNEFAILED																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														

5.4.3.28 XO.RUN

Address offset: 0x408

Indicates that XOSTART task was triggered

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	STATUS			XOSTART task triggered or not																														
			NotTriggered	0	Task not triggered																														
			Triggered	1	Task triggered																														

5.4.3.29 XO.STAT

Address offset: 0x40C

XO status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	STATE						XO state (Running between START task and STOPPED event)																											
			NotRunning	0				XO is not running																											
			Running	1				XO is running																											

5.4.3.30 PLL.RUN

Address offset: 0x428

Indicates that PLLSTART task was triggered

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	STATUS			PLLSTART task triggered or not																														
			NotTriggered	0	Task not triggered																														
			Triggered	1	Task triggered																														

5.4.3.31 PLL.STAT

Address offset: 0x42C

Which PLL settings were selected when triggering START task

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A																															
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																															
A	R	STATE			PLL state (Running between START task and STOPPED event)																															
			NotRunning	0	PLL is not running																															
			Running	1	PLL is running																															

5.4.3.32 LFCLK.SRC

Address offset: 0x440

Clock source for LFCLK

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	SRC			Select which LFCLK source is started by the LFCLKSTART task																														
			LFRC	0	32.768 kHz RC oscillator																														
			LFXO	1	32.768 kHz crystal oscillator																														
			LFSYNT	2	32.768 kHz synthesized from HFCLK																														

5.4.3.33 LFCLK.RUN

Address offset: 0x448

Indicates that LFCLKSTART task was triggered

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	STATUS						LFCLKSTART task triggered or not																											
			NotTriggered	0				Task not triggered																											
			Triggered	1				Task triggered																											

5.4.3.34 LFCLK.STAT

Address offset: 0x44C

Copy of LFCLK.SRCCOPY register, set when LFCLKSTARTED event is triggered.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				C																								B				A		A				
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	R	SRC				Value of LFCLK.SRCCOPY register when LFCLKSTARTED event was triggered																																
			LFRC	0	32.768 kHz RC oscillator																																	
			LFXO	1	32.768 kHz crystal oscillator																																	
			LFSYNT	2	32.768 kHz synthesized from HFCLK																																	
B	R	ALWAYSRUNNING				ALWAYSRUN activated																																
			NotRunning	0	Automatic clock control enabled																																	
			Running	1	Oscillator is always running																																	
C	R	STATE				LFCLK state (Running between START task and STOPPED event)																																
			NotRunning	0	LFCLK not running																																	
			Running	1	LFCLK running																																	

5.4.3.35 LFCLK.SRCCOPY

Address offset: 0x450

Copy of LFCLK.SRC register, set when LFCLKSTART task is triggered

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	SRC						Value of LFCLK.SRC register when LFCLKSTART task was triggered																											
			LFRC	0				32.768 kHz RC oscillator																											
			LFXO	1				32.768 kHz crystal oscillator																											
			LFSYNT	2				32.768 kHz synthesized from HFCLK																											

5.5 OSCILLATORS — Oscillator control

The system oscillators are automatically controlled by the clock control system, see [CLOCK — Clock control](#) on page 76.

The system has the following crystal oscillators:

- High-frequency 32 MHz crystal oscillator (HFXO)
- Low-frequency 32.768 kHz crystal oscillator (LFXO)

The crystal oscillators can be configured to use either internal or external capacitors.

5.5.1 High-frequency (32 MHz) crystal oscillator (HFXO)

The high-frequency crystal oscillator (HFXO) is controlled by a 32 MHz external crystal.

The crystal oscillator is designed for use with an AT-cut quartz crystal in parallel resonant mode, and is connected between pins **XC1** and **XC2**. For correct oscillation frequency, the load capacitance must match the specification in the crystal datasheet. The following figure shows how the 32 MHz crystal is connected to the high frequency crystal oscillator.

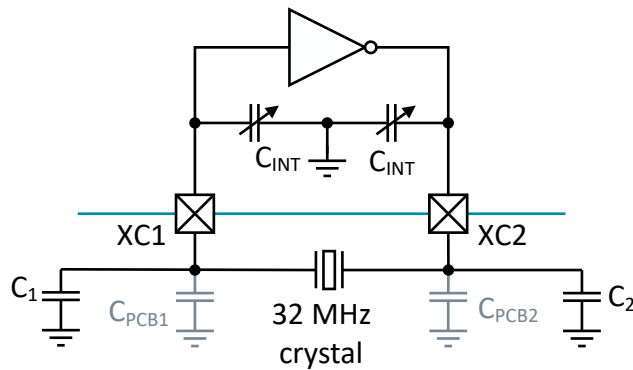


Figure 13: Circuit diagram of the high-frequency crystal oscillator

The device can be used with external capacitors C1 and C2 or the internal capacitors C_{INT} , which are configurable.

For reliable operation, the crystal load capacitance, shunt capacitance, equivalent series resistance, and drive level must comply with the specifications in [table 32 MHz crystal oscillator \(HFXO\)](#) on page 820. It is recommended to use a crystal with lower than maximum load capacitance and/or shunt capacitance. A low load capacitance reduces both start up time and current consumption.

When using internal capacitors, the load capacitance (CL) is the total capacitance seen by the crystal across its terminals and is calculated by the following equation.

$$CL = \frac{(C1' \cdot C2')}{(C1' + C2')}$$

$$C1' = C_{INT} + C_{pcb1}$$

$$C2' = C_{INT} + C_{pcb2}$$

Figure 14: Load capacitance equation for internal capacitors

C_{INT} is the value of the internal capacitors. C_{pcb1} and C_{pcb2} are stray capacitance on the PCB.

The internal capacitor must be configured before starting the high-frequency crystal oscillator using the XOSTART task. To enable the internal capacitors, find the correct value for C_{INT} in the field [OSCILLATORS.XOSC32M.CONFIG.INTCAP](#) using the following equation.

$$\text{INTCAP} = (((\text{CAPACITANCE} - 5.5) * (\text{FICR} \rightarrow \text{XOSC32MTRIM.SLOPE} + 791)) + \text{FICR} \rightarrow \text{XOSC32MTRIM.OFFSET} * 4) / 256$$

The equation has the following variables:

- CAPACITANCE is the desired capacitor value of C_{INT} in pF, holding any value between 4.0 pF and 17.0 pF in 0.25 pF steps.
- FICR→XOSC32MTRIM are factory trim values which vary between devices.

The device uses the internal capacitor together with the external crystal after configuration and HFXO starts.

5.5.1.1 Using external capacitors

It is possible to use external capacitors after disabling the internal capacitor.

When using external capacitors, the load capacitance (CL) is the total capacitance seen by the crystal across its terminals. It is calculated by the following equation.

$$CL = \frac{(C1' \cdot C2')}{(C1' + C2')}$$

$$C1' = C1 + C_{pcb1} + C_{pin}$$

$$C2' = C2 + C_{pcb2} + C_{pin}$$

Figure 15: Load capacitance equation for external capacitors

C1 and C2 are the external capacitors. C_{pcb1} and C_{pcb2} are stray capacitance on the PCB. C_{pin} is the pin input capacitance on pins **XC1** and **XC2**.

When using external capacitors, disable the internal capacitor by setting **OSCILLATORS.XOSC32M.CONFIG.INTCAP** to 0.

5.5.1.2 Crystal selection

Several crystals are supported by the 32 MHz crystal oscillator.

The following figure shows a simple model of a crystal. It has R-L-C series components, called equivalent series resistance (ESR), motional capacitance ($C_{0=N}$), and motional inductance (L_M). The capacitor in parallel, C_0 , is called the shunt capacitance, and models the package capacitance.

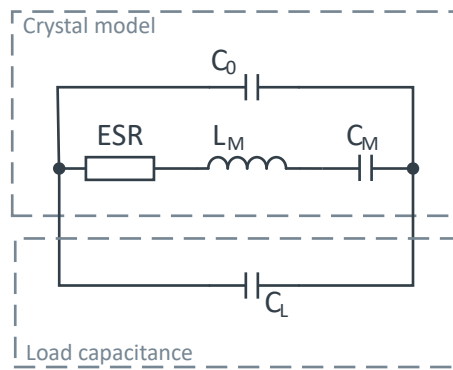


Figure 16: Simplified crystal model

The crystal selected needs to have parameters ESR, C_0 , and C_L selected to ensure stable operation of the crystal oscillator.

The following figure shows the maximum allowable combinations of ESR, C_0 and C_L for a given crystal. A crystal is supported if that crystal's parameters fall directly on the line or below it. Crystals that are above the line are not supported.

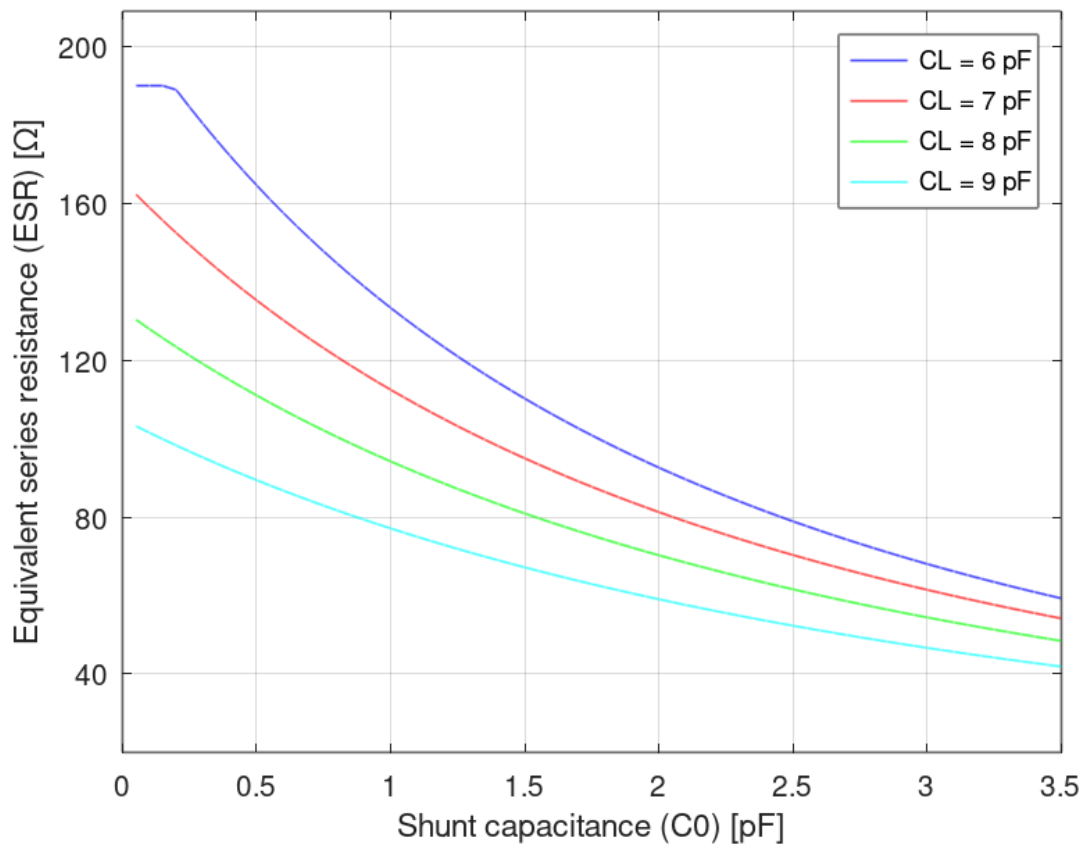


Figure 17: Maximum allowed combinations of ESR and C_0 for a given load capacitance C_L

5.5.2 Low-frequency (32.768 kHz) crystal oscillator (LFXO)

For higher clock accuracy than LFRC, the 32.768 kHz crystal oscillator (LFXO) must be used.

To use the LFXO, a 32.768 kHz crystal must be connected between the **XL1** and **XL2** pins, as shown in the following figure.

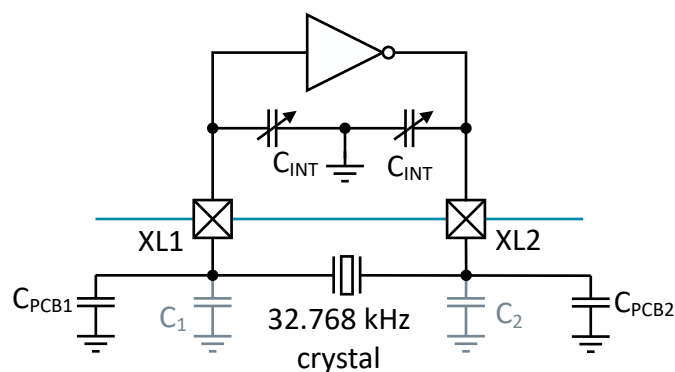


Figure 18: Circuit diagram of the low-frequency crystal oscillator

The device can be used with external capacitors C_1 and C_2 or the built-in configurable internal capacitors C_{INT} .

When using internal capacitors, the load capacitance (CL) is the total capacitance seen by the crystal across its terminals. It is calculated by the following equation.

$$CL = \frac{(C1' \cdot C2')}{(C1' + C2')}$$

$$C1' = C_{INT} + C_{pcb1}$$

$$C2' = C_{INT} + C_{pcb2}$$

Figure 19: Load capacitance equation for internal capacitors

C_{INT} is the value of the internal capacitors. C_{pcb1} and C_{pcb2} are stray capacitance on the PCB.

The internal capacitors must be configured before starting the low-frequency crystal oscillator (LFXO). To enable the internal capacitors, determine the correct field for `OSCILLATORS.XOSC32KI.INTCAP` using the following equation.

```
INTCAP = round( (2*CAPACITANCE - 12) * (FICR->XOSC32KTRIM.SLOPE + 0.765625 * 512) / 512 +
FICR->XOSC32KTRIM.OFFSET / 64 )
```

The equation has the following variables:

- CAPACITANCE is the desired capacitor value in pF, holding any value between 4 pF and 18 pF in 0.5 pF steps.
- FICR->XOSC32KTRIM are factory trim values which are device specific.

When LFXO starts, it will use the internal capacitor together with the external crystal.

5.5.2.1 Using external capacitors

When using external capacitors, the load capacitance (CL) is the total capacitance seen by the crystal across its terminals. It is calculated by the following equation.

$$CL = \frac{(C1' \cdot C2')}{(C1' + C2')}$$

$$C1' = C1 + C_{pcb1} + C_{pin}$$

$$C2' = C2 + C_{pcb2} + C_{pin}$$

Figure 20: Load capacitance equation for external capacitors

C1 and C2 are ceramic SMD capacitors connected between each crystal terminal and ground. C_{pcb1} and C_{pcb2} are stray capacitance on the PCB. C_{pin} is the pin input capacitance on pins **XL1** and **XL2**. The load capacitors C1 and C2 must have the same value.

When using external capacitors, the internal capacitor is disabled by setting `OSCILLATORS.XOSC32KI.INTCAP` to 0.

5.5.3 CPU clock frequency selection

The CPU clock frequency is configurable on boot in the register `PLL.FREQ (Retained)` on page 96.

The following speeds are supported:

- 64 MHz
- 128 MHz

The device starts at 64 MHz. For 128 MHz, it must be configured when the CPU starts and before any peripherals that use the high-frequency clock are enabled. Changing the frequency on a running system or to an unsupported value causes undefined system behavior and the device can malfunction.

5.5.4 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
OSCILLATORS : S	GLOBAL	0x50120000	US	S	NA	No	Oscillator control
OSCILLATORS : NS		0x40120000					

Register overview

Register	Offset	TZ	Description
XOSC32M.CONFIG.INTCAP	0x71C		Crystal load capacitor as seen by the crystal across its terminals, including pin capacitance but excluding PCB stray capacitance.
PLL.FREQ	0x800		Set speed of MCU power domain, including CPU This register is retained.
PLL.CURRENTFREQ	0x804		Current speed of MCU power domain, including CPU This register is retained.
XOSC32KI.INTCAP	0x904		Programmable capacitance of XL1 and XL2 This register is retained.

5.5.4.1 XOSC32M

32 MHz oscillator control

5.5.4.1.1 XOSC32M.CONFIG.INTCAP

Address offset: 0x71C

Crystal load capacitor as seen by the crystal across its terminals, including pin capacitance but excluding PCB stray capacitance.

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																												A	A	A	A	A
Reset 0x00000020	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

ID	R/W	Field	Value ID	Value	Description
A	RW	VAL			Crystal load capacitor value

Use the provided equation in [OSCILLATORS — Oscillator control](#) on page 91 to calculate the register value.

5.5.4.2 PLL

Oscillator control

5.5.4.2.1 PLL.FREQ (Retained)

Address offset: 0x800

Set speed of MCU power domain, including CPU

This register is retained.

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																						A	A				
Reset 0x00000003										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
ID	R/W	Field		Value ID		Value		Description																																			
A	RW	FREQ						Select CPU speed																																			
				CK128M		1		128 MHz																																			
				CK64M		3		64 MHz																																			

5.5.4.2.2 PLL.CURRENTFREQ (Retained)

Address offset: 0x804

Current speed of MCU power domain, including CPU

This register is retained.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000003				0 1 1																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	CURRENTFREQ			Active CPU speed																														
			CK128M	1	128 MHz																														
			CK64M	3	64 MHz																														

5.5.4.3 XOSC32KI

32.768 kHz oscillator control

5.5.4.3.1 XOSC32KI.INTCAP (Retained)

Address offset: 0x904

Programmable capacitance of XL1 and XL2

Use the provided equation in [OSCILLATORS — Oscillator control](#) on page 91 to calculate the register value.

This register is retained.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000017				0 1 0 1 1 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	VAL						Crystal load capacitor as seen by the crystal across its terminals, including pin capacitance but excluding PCB stray capacitance. Use the provided equation in "Using internal capacitors section" to calculate the register value.																											

5.6 POWER — Power control

The POWER peripheral provides an interface for the power and clock subsystem for task, event, and interrupt related settings.

The POWER peripheral requests resources from the power and clock subsystem. The power and clock subsystem makes sure that the power mode with the proper latency settings is selected when requested.

This means that the Constant Latency mode is prioritized over Low-power mode. For an overview of power modes, see [Sub-power modes](#) on page 73.

The event [POFWARN](#) is a system level event that enables the device to react quickly if there is a power failure. The power-fail comparator must be configured and enabled to receive the event, see [Power-fail comparator](#) on page 75 for more information.

Power control of the RAM blocks is controlled by the memory configuration peripheral (MEMCONF), see [MEMCONF — Memory configuration](#) on page 50.

Note: Registers [INTEN](#) on page 101, [INTENSET](#) on page 101, and [INTENCLR](#) on page 101 are shared between the POWER and CLOCK peripherals.

5.6.1 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
POWER : S	GLOBAL	0x5010E000	US	S	NA	No	Power control
POWER : NS		0x4010E000					

Register overview

Register	Offset	TZ	Description
TASKS_CONSTLAT	0x30		Enable Constant Latency mode
TASKS_LOWPWR	0x34		Enable Low-power mode (variable latency)
SUBSCRIBE_CONSTLAT	0xB0		Subscribe configuration for task CONSTLAT
SUBSCRIBE_LOWPWR	0xB4		Subscribe configuration for task LOWPWR
EVENTS_POFWARN	0x130		Power failure warning
EVENTS_SLEEPENTER	0x134		CPU entered WFI/WFE sleep
EVENTS_SLEEPEXIT	0x138		CPU exited WFI/WFE sleep
PUBLISH_SLEEPENTER	0x1B4		Publish configuration for event SLEEPENTER
PUBLISH_SLEEPEXIT	0x1B8		Publish configuration for event SLEEPEXIT
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
GPREGRET[n]	0x500		General purpose retention register
			This register is retained.
CONSTLATSTAT	0x520		Status of constant latency

5.6.1.1 TASKS_CONSTLAT

Address offset: 0x30

Enable Constant Latency mode

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																	A									
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																					
A	W	TASKS_CONSTLAT			Enable Constant Latency mode																																					
			Trigger	1	Trigger task																																					

5.6.1.2 TASKS_LOWPWR

Address offset: 0x34

Enable Low-power mode (variable latency)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_LOWPWR						Enable Low-power mode (variable latency)																											
			Trigger	1				Trigger task																											

5.6.1.3 SUBSCRIBE_CONSTLAT

Address offset: 0xB0

Subscribe configuration for task [CONSTLAT](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value				Description																														
A	RW	CHIDX		[0..255]				DPPI channel that task CONSTLAT will subscribe to																														
B	RW	EN																																				
			Disabled	0	Disable subscription																																	
			Enabled	1	Enable subscription																																	

5.6.1.4 SUBSCRIBE_LOWPWR

Address offset: 0xB4

Subscribe configuration for task [LOWPWR](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that task LOWPWR will subscribe to																																																									
B	RW	EN																																																													
			Disabled	0	Disable subscription																																																										
			Enabled	1	Enable subscription																																																										

5.6.1.5 EVENTS_POFWARN

Address offset: 0x130

Power failure warning

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	EVENTS_POFWARN				Power failure warning																													
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

5.6.1.6 EVENTS_SLEEPENTER

Address offset: 0x134

CPU entered WFI/WFE sleep

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																				A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	EVENTS_SLEEPENTER				CPU entered WFI/WFE sleep																															
			NotGenerated	0	Event not generated																																
			Generated	1	Event generated																																

5.6.1.7 EVENTS_SLEEPEXIT

Address offset: 0x138

CPU exited WFI/WFE sleep

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																																							A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	EVENTS_SLEEPEXIT				CPU exited WFI/WFE sleep																																	
			NotGenerated	0	Event not generated																																		
			Generated	1	Event generated																																		

5.6.1.8 PUBLISH_SLEEPENTER

Address offset: 0x1B4

Publish configuration for event [SLEEPENTER](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that event SLEEPENTER will publish to																																
B	RW	EN																																				
			Disabled	0	Disable publishing																																	
			Enabled	1	Enable publishing																																	

5.6.1.9 PUBLISH_SLEEPEXIT

Address offset: 0x1B8

Publish configuration for event [SLEEPEXIT](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that event SLEEPEXIT will publish to																																
B	RW	EN																																				
			Disabled	0	Disable publishing																																	
			Enabled	1	Enable publishing																																	

5.6.1.10 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				C B A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																														
A	RW	POFWARN			Enable or disable interrupt for event POFWARN																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
B	RW	SLEEPENTER			Enable or disable interrupt for event SLEEPENTER																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
C	RW	SLEEPEXIT			Enable or disable interrupt for event SLEEPEXIT																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

5.6.1.11 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	POFWARN			Write '1' to enable interrupt for event POFWARN																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	SLEEPENTER			Write '1' to enable interrupt for event SLEEPENTER																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	SLEEPEXIT			Write '1' to enable interrupt for event SLEEPEXIT																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

5.6.1.12 INTENCLR

Address offset: 0x308

All system components are powered from the main on-chip voltage regulator, VREGMAIN. The regulator converts the voltage supplied on **VDD** to internal voltage.

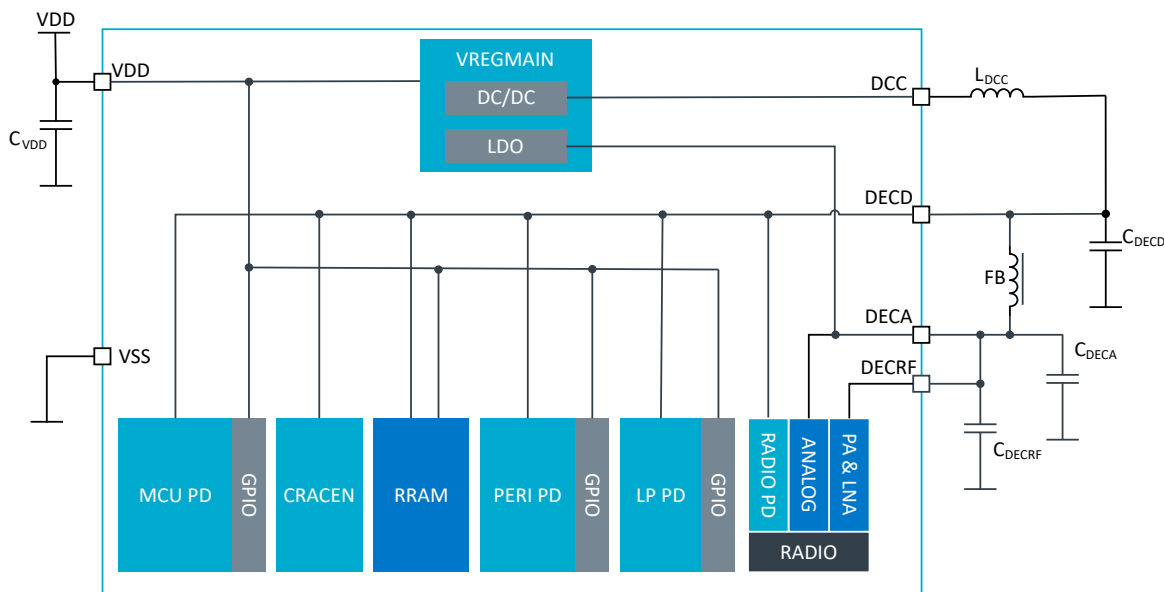


Figure 21: Regulator configuration

The main supply voltage is connected to the **VDD** pin.

After reset and device start up, VREGMAIN is enabled and operates in LDO mode. As soon as the device starts, the DC/DC regulator can be enabled using register [VREGMAIN.DCDCEN](#) on page 105.

5.7.1 VREGMAIN — Main regulator

VREGMAIN is the main regulator of the system.

After reset and device start up, VREGMAIN is enabled and in LDO mode. Once the device starts, the DC/DC regulator must be enabled using register [VREGMAIN.DCDCEN](#) on page 105. When enabling the DC/DC regulator, the device checks if an inductor is connected to the **DCC** pin. If an inductor is not detected, the device remains in LDO mode. Register [VREGMAIN.INDUCTORDET](#) on page 105 reports the inductor detection status and is used to detect inductor failure.

VREGMAIN supports DC/DC mode, which needs external components. For details, see [Reference circuitry](#) on page 808.

5.7.2 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
REGULATORS : S	GLOBAL	0x50120000	US	S	NA	No	Regulator control
REGULATORS : NS		0x40120000					

Register overview

Register	Offset	TZ	Description
SYSTEMOFF	0x500		System OFF register
POFCON	0x530		Power-fail comparator configuration This register is retained.
POFSTAT	0x534		Power-fail comparator status register
VREGMAIN.DCDCEN	0x600		Enable DC/DC converter for better power efficiency
VREGMAIN.INDUCTORDET	0x604		VREGMAIN inductor detection

5.7.2.1 SYSTEMOFF

Address offset: 0x500

System OFF register

Bit number										31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID										A																															
Reset 0x00000000										0 0																															
ID	R/W	Field	Value ID	Value	Description																																				
A	W	SYSTEMOFF			Enable System OFF mode																																				
			Enter	1	Enable System OFF mode																																				

5.7.2.2 POFCON (Retained)

Address offset: 0x530

Power-fail comparator configuration

This register is retained.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C B B B B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	POF	Disabled	0	Disable																														
			Enabled	1	Enable																														
B	RW	THRESHOLD	V17	0	Set threshold to 1.7 V																														
			V18	1	Set threshold to 1.8 V																														
			V19	2	Set threshold to 1.9 V																														
			V20	3	Set threshold to 2.0 V																														
			V21	4	Set threshold to 2.1 V																														
			V22	5	Set threshold to 2.2 V																														
			V23	6	Set threshold to 2.3 V																														
			V24	7	Set threshold to 2.4 V																														
			V25	8	Set threshold to 2.5 V																														
			V26	9	Set threshold to 2.6 V																														
			V27	10	Set threshold to 2.7 V																														
C	RW	EVENTDISABLE	Enabled	0	POFWARN event is generated																														
			Disabled	1	POFWARN event is not generated																														

5.7.2.3 POFSTAT

Address offset: 0x534

Power-fail comparator status register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	R	COMPARATOR				Power-fail comparator status																													
			Above	0		Voltage detected above VPOF threshold																													
			Below	1		Voltage detected below VPOF threshold																													

5.7.2.4 VREGMAIN

Register interface for main voltage regulator.

5.7.2.4.1 VREGMAIN.DCDCEN

Address offset: 0x600

Enable DC/DC converter for better power efficiency

Note: If inductor is not present (see register VREGMAIN.INDUCTORDET), the device will use LDO and not enable DC/DC converter

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	VAL				Enable DC/DC buck converter																													
			Disabled	0		Disable DC/DC converter and use LDO																													
			Enabled	1		Enable DC/DC converter																													
						If inductor is not present (see register VREGMAIN.INDUCTORDET), the device will use LDO mode and not enable DC/DC converter																													

5.7.2.4.2 VREGMAIN.INDUCTORDET

Address offset: 0x604

VREGMAIN inductor detection

Detect if an inductor is connected to the DCC pin. The detection can only take place if the DC/DC converter is not enabled (VREGMAIN.DCDCEN = 0)

Note: The device is only able to use the DC/DC converter if the inductor is detected

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																
ID				A																																
Reset 0x00000000				0 0																																
ID	R/W	Field	Value ID	Value				Description																												
A	R	DETECTED																																		
			InductorNotDetected0					VREGMAIN inductor not detected																												
			InductorDetected	1					VREGMAIN inductor detected																											

5.8 RESET — Reset control

A system-level reset is triggered by the following resets:

- Brownout
- Power-on
- CTRL-AP
- Watchdog
- Wakeup from System OFF
- Tamper detection
- Voltage glitch detection
- CPU Lockup
- Pin

The system reset sources are shown in the following figure.

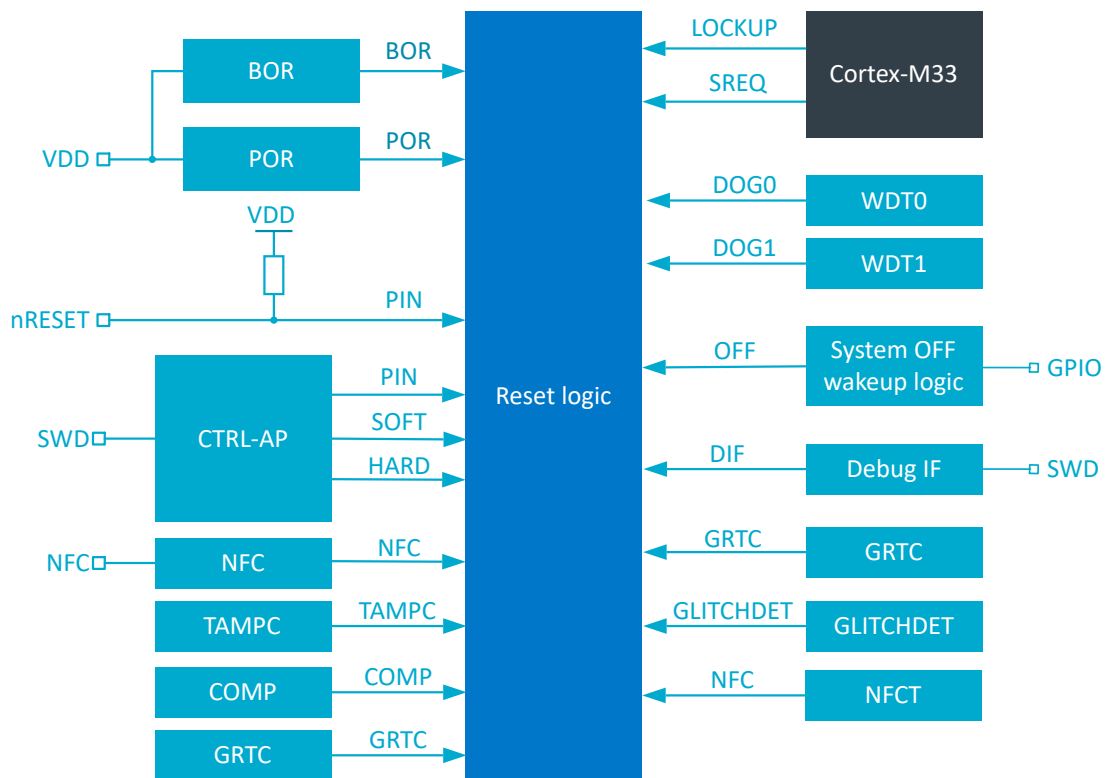


Figure 22: Reset sources

After a reset, the device automatically starts up. The register [RESETREAS](#) on page 109 can be read to determine which source generated the reset.

5.8.1 Power-on reset

The power-on reset (POR) generator initializes the system when the VDD supply voltage is above the power-on threshold.

The system is held in a reset state until the supply reaches the minimum operating voltage and the internal voltage regulators start. After a power-on reset, the device starts up.

5.8.2 Pin reset

A pin reset is generated when the physical reset pin on the device is asserted.

Pin reset is available on the reset pin **nRESET**, see [Pin assignments](#) on page 793. After **nRESET** is deasserted, the device starts.

The reset pin has an internal pull-up resistor with the same resistance as GPIO pull-ups, see [GPIO — General purpose input/output](#) on page 273.

A pin reset can also be generated using CTRL-AP, see [CTRL-AP resets](#) on page 107.

5.8.3 Brownout reset

The brownout reset (BOR) generator puts the system in RESET state if the VDD supply voltage drops below the brownout reset threshold.

Similar to a power-on reset, the device starts after BOR is deasserted.

5.8.4 Glitch detector

The glitch detector (GLITCHDET) puts the system in RESET state if either the VDD supply voltage or the device internal digital voltage drops below safe thresholds.

Similar to a power-on reset and a brownout-reset, the device starts after GLITCHDET is de-asserted.

For more information about the glitch detector, see [GLITCHDET — Voltage glitch detectors](#) on page 165.

5.8.5 Wakeup from System OFF mode reset

The device is reset when it wakes up from System OFF mode.

Similar to a power-on reset (POR), the device is started after waking up from System OFF.

If the device is in Debug interface mode, the debug access port (DAP) is not reset after a wakeup from System OFF mode. For more information, see [Debug and trace](#) on page 749.

For details on the System OFF mode, see [System OFF mode](#) on page 74.

5.8.6 Soft reset

A soft reset is generated when the SYSRESETREQ bit of the application interrupt and reset control register (AIRCR) in the Arm CPU is set. For more information, see [Arm documentation](#).

A soft reset can also be generated using CTRL-AP, see [CTRL-AP resets](#) on page 107.

Similar to a power-on reset (POR), the device is restarted after a soft reset.

5.8.7 CTRL-AP resets

CTRL-AP can generate the following resets.

- Soft reset
- Pin reset
- Hard reset. This is used during an Erase ALL operation, and is less intrusive than Pin reset. For more details, see [Reset behavior](#) on page 108.

Through the debugger interface, CTRL-AP can generate three resets using register [RESET](#) on page 761. For more details, see [CTRL-AP - Control access port](#) on page 755.

Similar to a power-on reset (POR), the device is restarted after any CTRL-AP reset.

5.8.8 Watchdog timer reset

A watchdog timer (WDT) reset is generated when the watchdog timer times out.

Similar to a power-on reset (POR), the device is started after a watchdog reset.

5.8.9 Retained registers

A retained register is one that keeps its value when entering System OFF mode. See individual peripheral chapters for information about which registers are retained.

5.8.10 Reset behavior

The reset source determines the behavior of the device after a reset.

In System OFF mode, the watchdog timer is not running and CPU lockup is not possible. RAM may be fully or partially retained, depending on RAM retention settings in [MEMCONF — Memory configuration](#) on page 50.

If the device is in Debug Interface mode, the debug components are not reset. Additionally, CPU lockup does not generate a reset. See [Debug and trace](#) on page 749 for more information about the different debug components in the system.

An 'x' in the table means that the specific module or register is reset. The table also explicitly lists which reset sources are commonly referred to as 'cold boot'.

Reset source	Cold boot	CM33	Peripherals	Debug	RAM	WDT	TAMP	REGULATOR OSCILLATORS and CPU speed	Retained registers		
									RESET REAS	POWER-RET-REG	GPIO
CPU lockup		x	x				x ²				x ¹
Soft reset and CTRL-AP soft reset		x	x								x ¹
Wakeup from System Off mode		x	x			x					
CTRL-AP hard reset		x	x		x	x		x			x
CTRL-AP pin reset		x	x	x	x	x		x			x
Watchdog timer reset		x	x	x	x	x		x			x
Pin reset		x	x	x	x	x		x			x
TAMPC reset		x	x	x	x	x		x			x
GLITCHDET reset		x	x	x	x	x		x	x	x	x
Brownout reset	x	x	x	x	x	x		x	x	x	x
Power-on reset	x	x	x	x	x	x		x	x	x	x

Table 19: Reset overview

¹Except the CTRLSEL field.

²For TAMPC reset sources, see [TAMPC — Tamper controller](#) on page 192.

5.8.11 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
RESET : S	GLOBAL	0x5010E000	US	S	NA	No	Reset status
RESET : NS		0x4010E000					

Register overview

Register	Offset	TZ	Description
RESETREAS	0x600		Reset reason

5.8.11.1 RESETREAS

Address offset: 0x600

Reset reason

Before entering System OFF mode, the RESETREAS register must be cleared.

Note: Unless cleared, the RESETREAS register will be cumulative. A field is cleared by writing 1 to it.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				N M L K J I H G F E D C B A																																
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																															
A	RW	RESETPIN			Reset from pin reset detected																															
					CTRL-AP generating a pin reset has its own bit																															
			NotDetected	0	Not detected																															
			Detected	1	Detected																															
B	RW	DOG0			Reset from watchdog timer 0 detected																															
			NotDetected	0	Not detected																															
			Detected	1	Detected																															
C	RW	DOG1			Reset from watchdog timer 1 detected																															
			NotDetected	0	Not detected																															
			Detected	1	Detected																															
D	RW	CTRLAPSOFT			Soft reset from CTRL-AP detected																															
			NotDetected	0	Not detected																															
			Detected	1	Detected																															
E	RW	CTRLAPHARD			Reset due to CTRL-AP hard reset																															
			NotDetected	0	Not detected																															
			Detected	1	Detected																															
F	RW	CTRLAPPIN			Reset due to CTRL-AP pin reset																															
			NotDetected	0	Not detected																															
			Detected	1	Detected																															
G	RW	SREQ			Reset from soft reset detected																															
			NotDetected	0	Not detected																															
			Detected	1	Detected																															
H	RW	LOCKUP			Reset from CPU lockup detected																															

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
			NotDetected	0	Not detected																														
			Detected	1	Detected																														
I	RW	OFF			Reset due to wakeup from System OFF mode when wakeup is triggered by DETECT signal from GPIO																														
			NotDetected	0	Not detected																														
			Detected	1	Detected																														
J	RW	LPCOMP			Reset due to wakeup from System OFF mode when wakeup is triggered by ANADETECT signal from LPCOMP																														
			NotDetected	0	Not detected																														
			Detected	1	Detected																														
K	RW	DIF			Reset triggered by Debug Interface																														
			NotDetected	0	Not detected																														
			Detected	1	Detected																														
L	RW	GRTC			Reset due to wakeup from GRTC																														
			NotDetected	0	Not detected																														
			Detected	1	Detected																														
M	RW	NFC			Reset after wakeup from System OFF mode due to NFC field being detected																														
			NotDetected	0	Not detected																														
			Detected	1	Detected																														
N	RW	SECTAMPER			Reset due to illegal tampering of the device																														
			NotDetected	0	Not detected																														
			Detected	1	Detected																														

6 Event system

The distributed programmable peripheral interconnect (DPPI) system enables peripherals to interact autonomously with each other through tasks and events, without intervention from the CPU.

The DPPI channels are local to each power domain, but can be transferred between power domains using PPI bridges.

The following figure shows the power domains, the PPI controllers (DPPIC), and the PPI bridges (PPIB).

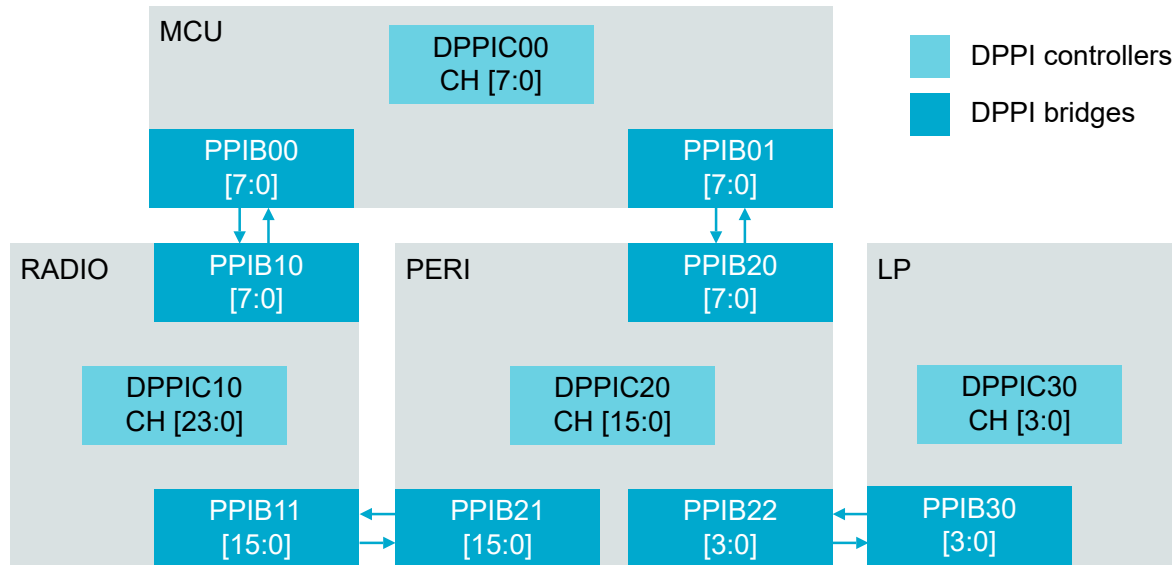


Figure 23: Power domains and PPI bridges

A subset of PPI channels from a power domain can be bridged across to a different power domain. For example, PPIB00 can bridge up to eight configurable DPPI channels to PPIB10. For more details on how to configure the PPI and bridge system, see [DPPI — Distributed programmable peripheral interconnect](#) on page 112 and [PPIB — PPI Bridge](#) on page 120.

6.1 DPPI latencies

DPPI task and event latency depends on the power domain of the source and destination peripherals.

DPPI signals operate on the HCLK128M, PCLK32M, and PCLK16M clocks.

Power domain	Clock source
MCU	HCLK128M (64 or 128 MHz)
RADIO	PCLK32M (32 MHz)
PERI	PCLK16M (16 MHz)
LP	PCLK16M (16 MHz)

Table 20: DPPI clock frequency

For peripherals in the same power domain, there is a two cycle delay from when an event is generated until a task subscribing to the same channel is triggered. Events that are generated while the system

is sleeping will have additional access latency, as the system needs to request and provide sufficient resources for PPI handling. Examples of such events are the following:

- A GPIO toggling and generating an event through GPIOTE
- Events generated by GRTC while all other clocks are stopped

To improve PPI latency, the Constant Latency mode can be used, see [Sub-power modes](#) on page 73.

For peripherals in different power domains, additional access latency will apply. Events that are generated while the generating or receiving power domains are sleeping will have additional access latency, as the system needs to request and provide sufficient resources for PPI handling.

6.2 DPPI — Distributed programmable peripheral interconnect

The distributed programmable peripheral interconnect (DPPI) enables peripherals to interact autonomously with each other by using tasks and events, without any intervention from the CPU. DPPI allows precise synchronization between peripherals when real-time application constraints exist, and eliminates the need for CPU involvement to implement behavior which can be predefined using the DPPI.

Note: For more information on tasks, events, publish/subscribe, interrupts, and other concepts, see [Peripheral interface](#) on page 213.

The main features of DPPI are:

- Peripheral tasks can subscribe to channels
- Peripheral events can be published on channels
- Publish/subscribe pattern enabling multiple connection options that include the following:
 - One-to-one
 - One-to-many
 - Many-to-one
 - Many-to-many

The DPPI consists of several PPIBus modules, which are connected to a fixed number of DPPI channels and a DPPI controller (DPPIC).

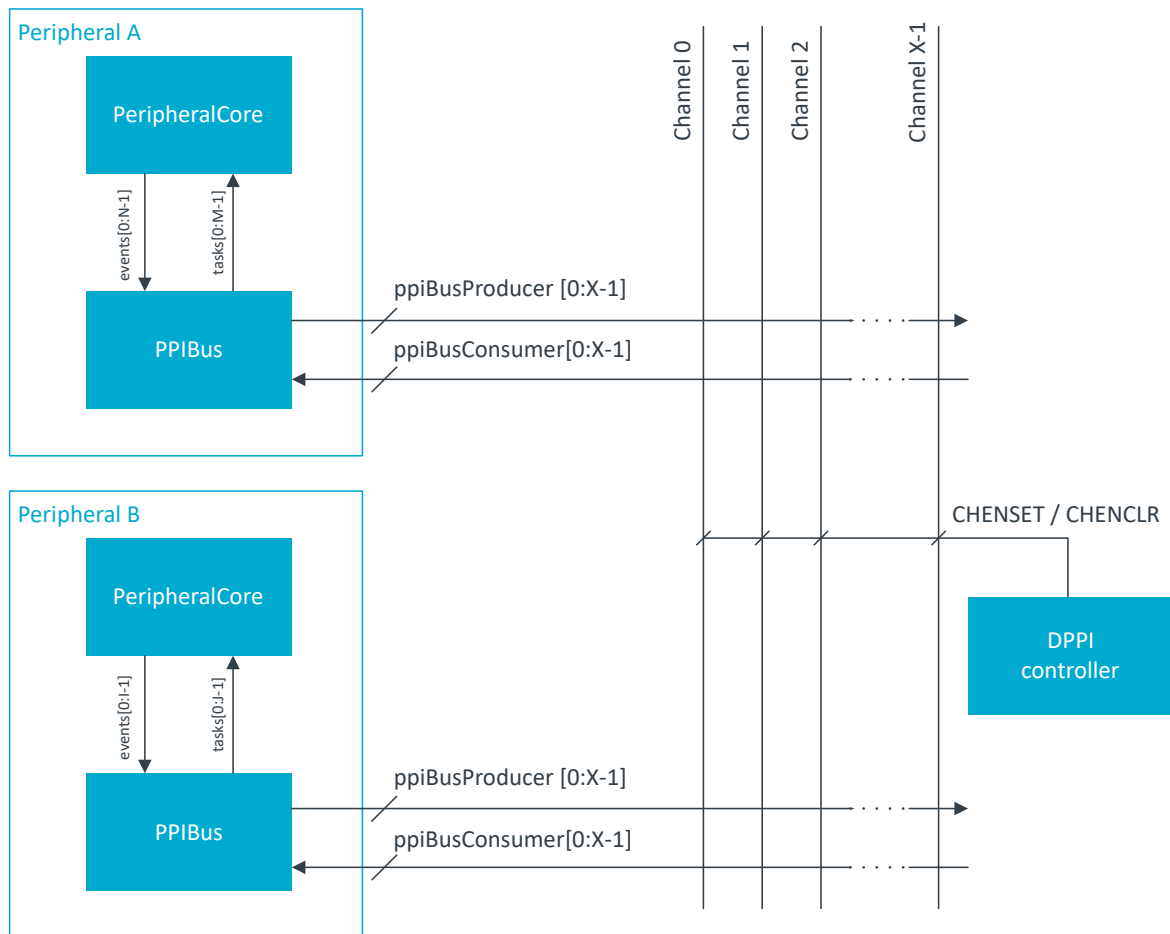


Figure 24: DPPI overview

6.2.1 Subscribing to and publishing on channels

The PPIBus can route peripheral events onto the channels (publishing), or route events from the channels into peripheral tasks (subscribing).

All peripherals include the following:

- One subscribe register per task
- One publish register per event

Publish and subscribe registers use a channel index field to determine the channel to which the event is published or tasks subscribed. In addition, there is an enable bit for the subscribe and publish registers that needs to be enabled before the subscription or publishing takes effect.

Writing non-existing channel index (CHIDX) numbers into a peripheral's publish or subscribe registers will yield unexpected results.

One event can trigger multiple tasks by subscribing different tasks to the same channel. Similarly, one task can be triggered by multiple events by publishing different events to the same channel. For advanced use cases, multiple events and multiple tasks can connect to the same channel forming a many-to-many connection. If multiple events are published on the same channel at the same time, the events are merged and only one event is routed through the DPPI.

How peripheral events are routed onto different channels based on publish registers is illustrated in the following figure.

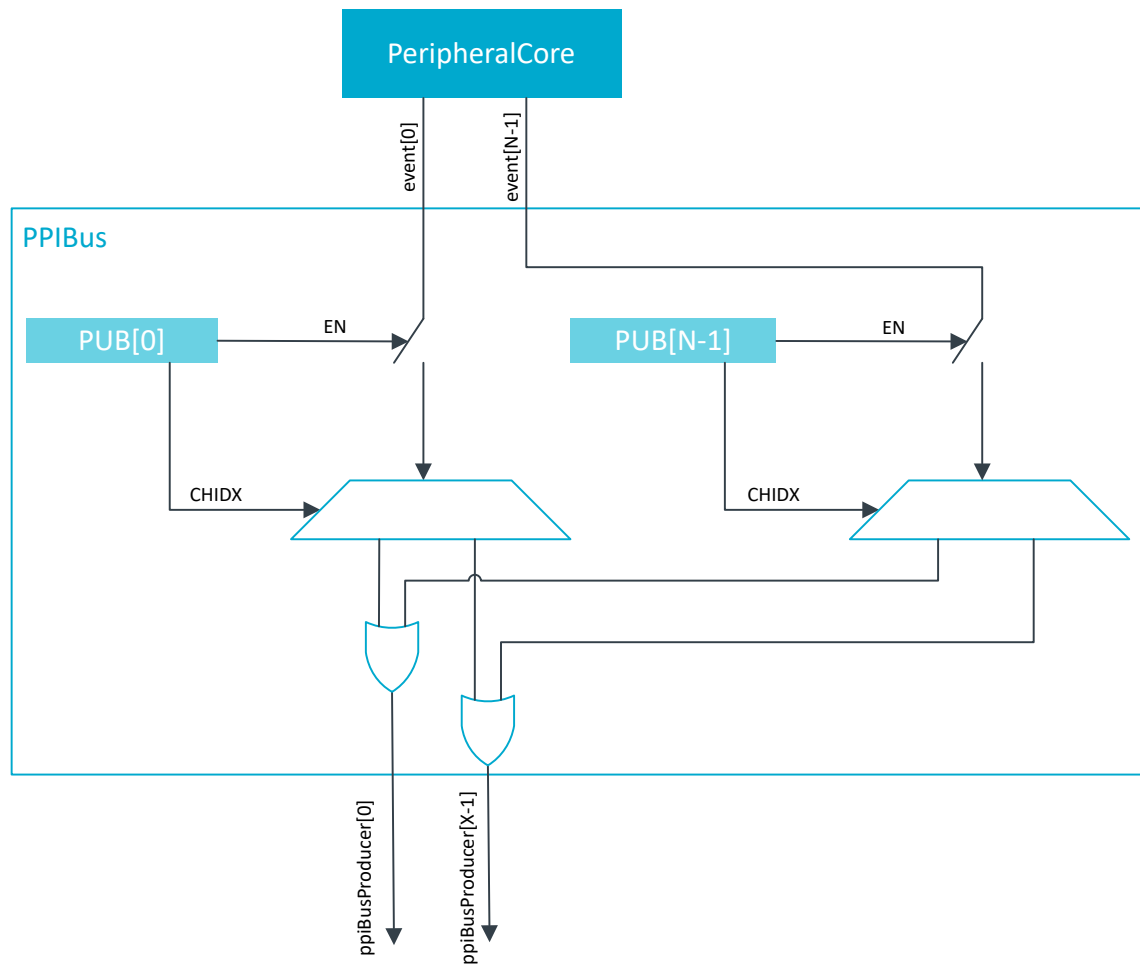


Figure 25: DPPI events flow

The following figure illustrates how peripheral tasks are triggered from different channels based on subscribe registers.

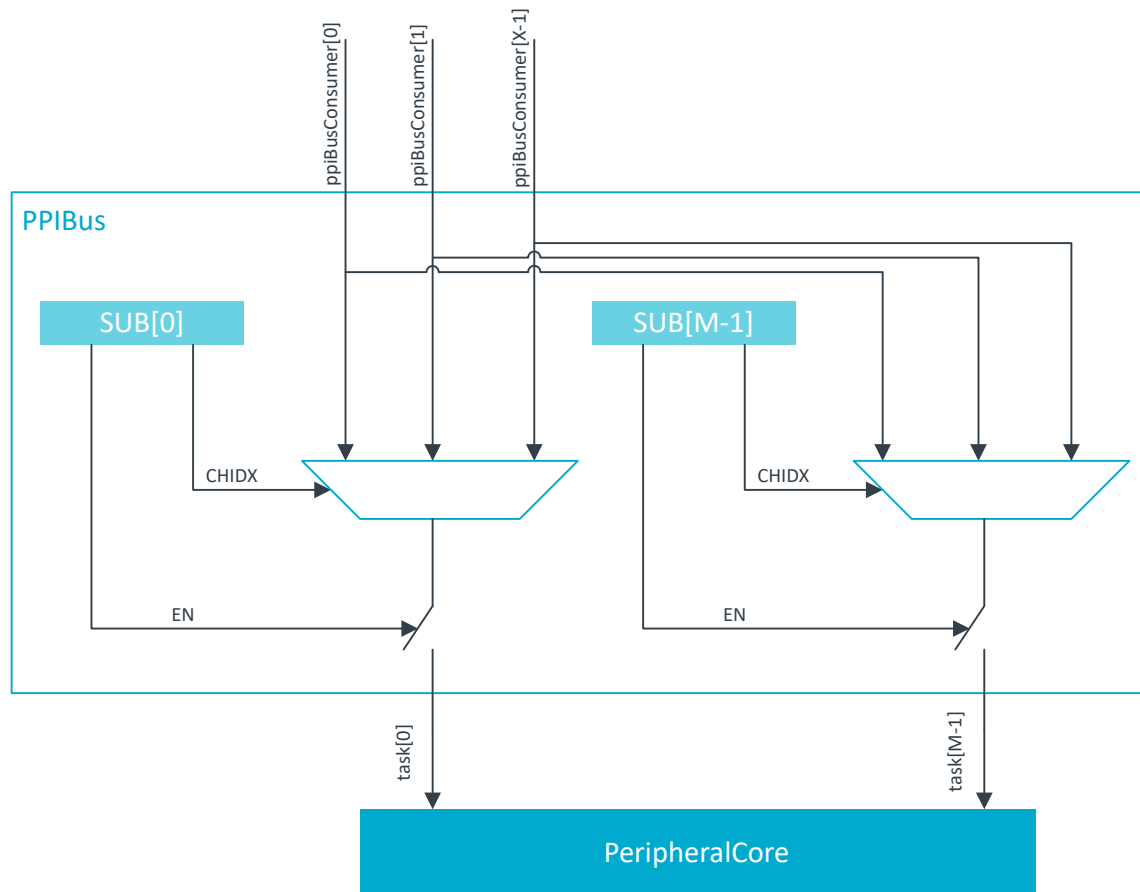


Figure 26: DPPI tasks flow

6.2.2 DPPI controller (DPPIC)

Enabling and disabling of DPPI channels is handled through DPPIC.

There are two ways of enabling or disabling a DPPI channel using DPPIC:

- Enable or disable channels individually using registers CHEN, CHENSET, and CHENCLR.
- Enable or disable channels in channel groups using the groups' tasks ENABLE and DISABLE. Channel groups should be defined via the CHG registers before these tasks are triggered.

Note: ENABLE tasks are prioritized over DISABLE tasks, i.e. in case of a simultaneously occurring TASKS_CHG[m].EN and TASKS_CHG[n].DIS (m and n can be equal or different), the CHG[m].EN task will be prioritized if the same channel subscribed to both groups.

DPPIC tasks (for example CHG[0].EN) can be triggered through DPPI like any other task, which means they can be linked to a DPPI channel through the subscribe registers.

In order to write to CHG[n], the corresponding CHG[n].EN and CHG[n].DIS subscribe registers must be disabled. Writes to CHG[n] are ignored if any of the two subscribe registers are enabled.

6.2.3 Connection examples

DPPI offers several connection options. Examples are given for how to create one-to-one and many-to-many connections.

One-to-one connection

This example shows how to create a one-to-one connection between TIMER compare register and SAADC start task.

The channel configuration is set up first. TIMER will publish its COMPARE0 event on channel 0, and SAADC will subscribe its START task to events on the same channel. After that, the channel is enabled through the DPPIC.

```
NRF_TIMER20->PUBLISH_COMPARE[0] = (0 << TIMER_PUBLISH_COMPARE_CHIDX_Pos) |
    TIMER_PUBLISH_COMPARE_EN_Msk;
NRF_SAADC->SUBSCRIBE_START = (0 << SAADC_SUBSCRIBE_START_CHIDX_Pos) |
    SAADC_SUBSCRIBE_START_EN_Msk;
NRF_DPPIC20->CHENSET = DPPIC_CHENSET_CH0_Msk;
```

Many-to-many connection

The example shows how to create a many-to-many connection, showcasing the DPPIC's channel group functionality.

A channel group that includes only channel 0 is set up first. Then the GPIOTE and TIMER configure their IN0 and COMPARE0 events respectively to be published on channel 0, while the SAADC configures its START task to subscribe to events on channel 0. Through DPPIC, the CHG0 DISABLE task is configured to subscribe to events on channel 0. After an event is received on channel 0 it will be disabled. Finally, channel 0 is enabled using the DPPIC task to enable a channel group.

```
NRF_DPPIC20->CHG[0] = (DPPIC_CHG_CH0_Included << DPPIC_CHG_CH0_Pos);
NRF_GPIOTE20->PUBLISH_IN[0] = (0 << GPIOTE_PUBLISH_IN_CHIDX_Pos) |
    GPIOTE_PUBLISH_IN_EN_Msk;
NRF_TIMER20->PUBLISH_COMPARE[0] = (0 << TIMER_PUBLISH_COMPARE_CHIDX_Pos) |
    TIMER_PUBLISH_COMPARE_EN_Msk;
NRF_SAADC->SUBSCRIBE_START = (0 << SAADC_SUBSCRIBE_START_CHIDX_Pos) |
    SAADC_SUBSCRIBE_START_EN_Msk;
NRF_DPPIC20->SUBSCRIBE_CHG[0].DIS = DPPIC_CHENSET_CH0_Msk | DPPIC_SUBSCRIBE_CHG_DIS_EN_Msk;
NRF_DPPIC20->TASK_CHG[0].EN = 1;
```

6.2.4 Split security

Individual DPPI channels and channel groups can have independent security attributes.

The split security of DPPI means it handles accesses from both secure and non-secure code. DPPI channels and channel groups can be defined as secure or non-secure.

For more information on DPPI security, see [DPPIC](#) on page 132.

6.2.5 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
DPPIC00 : S	GLOBAL	0x50042000	US	S	NA	Yes	DPPI controller DPPIC00
DPPIC00 : NS		0x40042000					
DPPIC10 : S	GLOBAL	0x50082000	US	S	NA	Yes	DPPI controller DPPIC10
DPPIC10 : NS		0x40082000					
DPPIC20 : S	GLOBAL	0x500C2000	US	S	NA	Yes	DPPI controller DPPIC20
DPPIC20 : NS		0x400C2000					
DPPIC30 : S	GLOBAL	0x50102000	US	S	NA	Yes	DPPI controller DPPIC30
DPPIC30 : NS		0x40102000					

Configuration

Instance	Domain	Configuration
DPPIC00 : S	GLOBAL	8 DPPI channels
DPPIC00 : NS		2 DPPI groups
DPPIC10 : S	GLOBAL	24 DPPI channels
DPPIC10 : NS		6 DPPI groups
DPPIC20 : S	GLOBAL	16 DPPI channels
DPPIC20 : NS		6 DPPI groups
DPPIC30 : S	GLOBAL	4 DPPI channels
DPPIC30 : NS		2 DPPI groups

Register overview

Register	Offset	TZ	Description
TASKS_CHG[n].EN	0x000		Enable channel group n
TASKS_CHG[n].DIS	0x004		Disable channel group n
SUBSCRIBE_CHG[n].EN	0x080		Subscribe configuration for task CHG[n].EN
SUBSCRIBE_CHG[n].DIS	0x084		Subscribe configuration for task CHG[n].DIS
CHEN	0x500		Channel enable register
CHENSET	0x504		Channel enable set register
CHENCLR	0x508		Channel enable clear register
CHG[n]	0x800		Channel group n
Note: Writes to this register are ignored if either SUBSCRIBE_CHG[n].EN or SUBSCRIBE_CHG[n].DIS is enabled			

6.2.5.1 TASKS_CHG[n] (n=0..5)

Channel group tasks

6.2.5.1.1 TASKS_CHG[n].EN (n=0..5)

Address offset: $0x000 + (n \times 0x8)$

Enable channel group n

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																								A				
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field		Value ID	Value					Description																																		
A	W	EN								Enable channel group n																																		
				Trigger	1					Trigger task																																		

6.2.5.1.2 TASKS_CHG[n].DIS (n=0..5)

Address offset: 0x004 + (n × 0x8)

Disable channel group n

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																				A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value				Description																												
A	W	DIS						Disable channel group n																												
			Trigger	1					Trigger task																											

6.2.5.2 SUBSCRIBE_CHG[n] (n=0..5)

Subscribe configuration for tasks

6.2.5.2.1 SUBSCRIBE_CHG[n].EN (n=0..5)

Address offset: 0x080 + (n × 0x8)

Subscribe configuration for task CHG[n].EN

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that task CHG[n].EN will subscribe to																																
B	RW	EN																																				
			Disabled	0	Disable subscription																																	
			Enabled	1	Enable subscription																																	

6.2.5.2.2 SUBSCRIBE_CHG[n].DIS (n=0..5)

Address offset: 0x084 + (n × 0x8)

Subscribe configuration for task CHG[n].DIS

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that task CHG[n].DIS will subscribe to																																
B	RW	EN																																				
			Disabled	0	Disable subscription																																	
			Enabled	1	Enable subscription																																	

6.2.5.3 CHEN

Address offset: 0x500

Channel enable register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				X W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-X	RW	CH[i] (i=0..23)			Enable or disable channel i																														
			Disabled	0	Disable channel																														
			Enabled	1	Enable channel																														

6.2.5.4 CHENSET

Address offset: 0x504

Channel enable set register

Note: Read: Reads value of CH[i] field in CHEN register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				X W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-X	RW	CH[i] (i=0..23)			Channel i enable set register. Writing 0 has no effect.																														
			Disabled	0	Read: Channel disabled																														
			Enabled	1	Read: Channel enabled																														
			Set	1	Write: Enable channel																														

6.2.5.5 CHENCLR

Address offset: 0x508

Channel enable clear register

Note: Read: Reads value of CH[i] field in CHEN register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				X W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-X	RW	CH[i] (i=0..23)			Channel i enable clear register. Writing 0 has no effect.																														
			Disabled	0	Read: Channel disabled																														
			Enabled	1	Read: Channel enabled																														
			Clear	1	Write: Disable channel																														

6.2.5.6 CHG[n] (n=0..5)

Address offset: 0x800 + (n × 0x4)

Channel group n

Note: Writes to this register are ignored if either SUBSCRIBE_CHG[n].EN or SUBSCRIBE_CHG[n].DIS is enabled

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					X W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value	ID	Value	Description																														
A-X	RW	CH[i] (i=0..23)				Include or exclude channel i																														
			Excluded	0		Exclude																														
			Included	1		Include																														

6.3 PPIB — PPI Bridge

PPIB connects tasks and events of peripherals in two different PPI systems in different power-domains.

A PPI system contains a number of peripherals that can communicate with each other by using tasks and events. This functionality is enabled by the DPPI peripheral. In a PPI system, the peripherals and DPPI are instantiated in the same APB bus.

The following figure shows a PPI system including a PPIB:

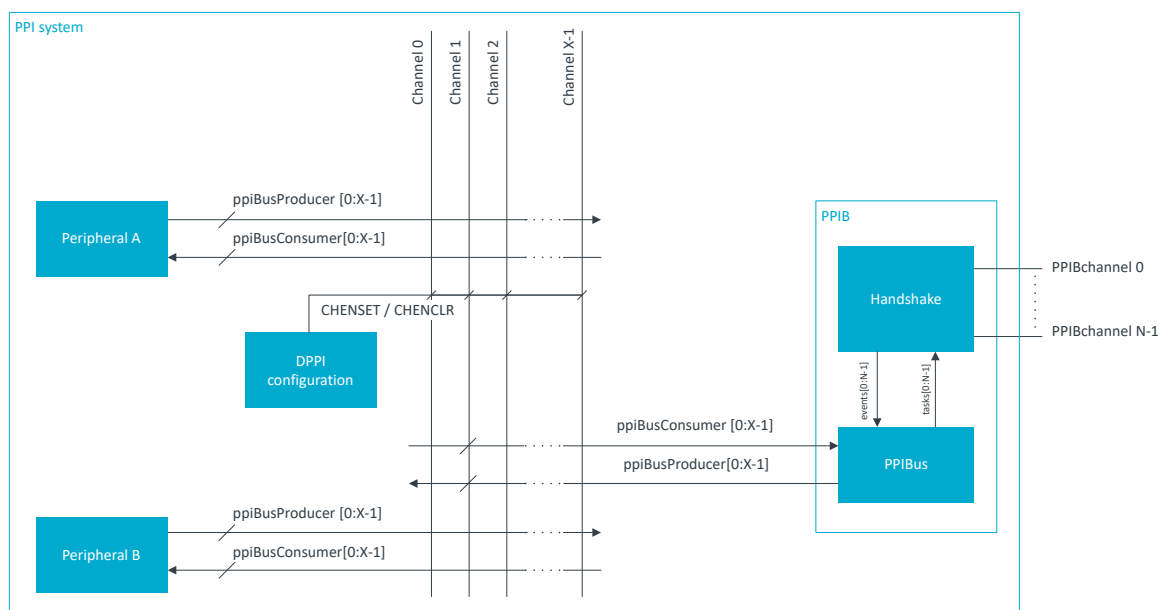


Figure 27: PPI system with PPIB

PPIB uses tasks and events like a standard peripheral, and connects to local DPPI channels via PPIBus. For more information on PPIBus module, see DPPI.

PPIB has a number of channels. Each PPIB channel connects to a single DPPI channel.

6.3.1 PPIB connections

A PPIB channel in one PPI system can be connected to a PPIB channel in another PPI system forming a PPIB connection.

A channel belonging to a PPIB instance in a PPI system is connected to a channel belonging to a PPIB instance in a different PPI system, creating a one-to-one PPIB connection between the two PPI systems. The connections are fixed and point-to-point, that is, a channel in a PPIB instance is connected only to a specific channel in another PPIB instance. For information on how the channels in the different PPIB instances are connected, see the Configuration table under the Registers section.

A PPIB channel can be configured as either source or sink. When configuring one side of the PPIB connection as source, the other side of the PPIB connection must be configured as sink, and viceversa. PPIB connections are unidirectional. Configuring both sides of a connection as source and sink at the same time will yield unexpected results.

On the source side of a PPIB connection, in order to send a (local) peripheral event to a different PPI system, the corresponding PPIB channel is configured as a consumer, subscribing to the same DPPI channel as the (local) peripheral publishes to, using the `PPIB.SUBSCRIBE_SEND[n]` register, with `n` the PPIB channel number.

On the sink side of a PPIB connection, for a (local) peripheral to be able to receive this event, the corresponding PPIB channel is configured as a producer, publishing to the same DPPI channel as the (local) peripheral subscribes to, using the `PPIB.PUBLISH_RECEIVE[n]` register, with `n` the PPIB channel number.

In a PPI system, several peripherals can publish to the same DPPI channel on the source side of a PPIB connection. Similarly, several peripherals can subscribe to the same DPPI channel on the sink side of a PPIB connection. This allows multiple connection options between peripherals in different PPI systems, same as DPPI allows in a local PPI system: one-to-one, one-to-many, many-to-one and many-to-many. However, when multiple peripherals can publish to the same DPPI channel on the source side of a PPIB connection, there is a risk of overflow. See [Handshake and overflow](#) on page 121.

6.3.2 Handshake and overflow

The two PPIB instances in a PPIB connection need a handshake to transfer a peripheral event.

This is handled by a Handshake module in the PPIB. If a handshake fails because an earlier event has not been processed completely, the new event won't be sent. Instead, bit `i` in `OVERFLOW.SEND` register on the source side will be set, with `i` the corresponding PPIB channel number.

6.3.3 Connection examples

This section contains examples on how to connect two PPI systems using PPIB.

The following example shows how to create a PPIB connection between the TIMER10 compare event in the RADIO PD and the SAADC start task in PERI PD. PPIB11 in RADIO PD is hardwired to PPIB21 in PERI PD, which allows the PPI systems in the two separate power domains to connect. DPPI channel 0 is used by both power domains. Note that it is only necessary to use the same DPPI channel within the power domain; different DPPI channels can be used across power domains. An example of this is given further down in this section.

```
// RADIO PD
NRF_TIMER10->PUBLISH_COMPARE[0] = (0<<TIMER_PUBLISH_COMPARE_CHIDX_Pos) |
TIMER_PUBLISH_COMPARE_EN_Msk;
NRF_PPIB11->SUBSCRIBE_SEND[0] = (0<<PPIB_SUBSCRIBE_SEND_CHIDX_Pos) |
PPIB_SUBSCRIBE_SEND_EN_Msk;
NRF_DPPIC10->CHENSET = DPPIC_CHENSET_CH0_Msk;

// PERI PD
NRF_SAADC->SUBSCRIBE_START = (0<<SAADC_SUBSCRIBE_START_CHIDX_Pos) |
SAADC_SUBSCRIBE_START_EN_Msk;
NRF_PPIB21->PUBLISH_RECEIVE[0] = (0<<PPIB_PUBLISH_RECEIVE_CHIDX_Pos) |
PPIB_PUBLISH_RECEIVE_EN_Msk;
NRF_DPPIC20->CHENSET = DPPIC_CHENSET_CH0_Msk;
```

The following example shows how to create a PPIB connection between the TIMER10 compare event in the RADIO PD and the COMP start task in LP PD. The two PPI systems must be connected through PPIB

instances PPIB21 and PPIB22 in PERI PD. These PPIB instances are not connected to any peripheral, only to the PPIB instances in RADIO and LP power domains. PERI PD acts as a central system that connects the two systems by means of local PPIB and DPPIC instances. This allows scaling to larger PPI systems, since multiple PPI systems can be interconnected through a central PPI system. DPPI channel 0 is used for internal RADIO PD connections, channel 1 is used by LP PD, and channel 5 is used by PERI PD. It is important that same DPPI channels are used within a power domain, but across domains the DPPI channel number does not matter.

```
// RADIO PD
NRF_TIMER10->PUBLISH_COMPARE[0] = (0<<TIMER_PUBLISH_COMPARE_CHIDX_Pos) |
TIMER_PUBLISH_COMPARE_EN_Msk;
NRF_PPIB11->SUBSCRIBE_SEND[0] = (0<<PPIB_SUBSCRIBE_SEND_CHIDX_Pos) |
PPIB_SUBSCRIBE_SEND_EN_Msk;
NRF_DPPIC10->CHENSET = DPPIC_CHENSET_CH0_Msk;

// LP PD
NRF_COMP->SUBSCRIBE_START = (1<<LPCOMP_SUBSCRIBE_SAMPLE_CHIDX_Pos) |
COMP_SUBSCRIBE_START_EN_Msk;
NRF_PPIB30->PUBLISH_RECEIVE[0] = (1<<PPIB_PUBLISH_RECEIVE_CHIDX_Pos) |
PPIB_PUBLISH_RECEIVE_EN_Msk;
NRF_DPPIC30->CHENSET = DPPIC_CHENSET_CH1_Msk;

// PERI PD
NRF_PPIB21->PUBLISH_RECEIVE[0] = (5<<PPIB_PUBLISH_RECEIVE_CHIDX_Pos) |
PPIB_PUBLISH_RECEIVE_EN_Msk;
NRF_PPIB22->SUBSCRIBE_SEND[0] = (5<<PPIB_SUBSCRIBE_SEND_CHIDX_Pos) |
PPIB_SUBSCRIBE_SEND_EN_Msk;
NRF_DPPIC20->CHENSET = DPPIC_CHENSET_CH5_Msk;
```

6.3.4 Ownership and security attributes

Each PPIB channel in a PPB instance has a specific ownership and security attribute.

The ownership and security attribute of a PPIB channel is configured with the owner and security of the DPPI channel it is connected to. For information on how PPIB channels and DPPI channels are connected in the device, see the Configuration table under the Registers section.

There is no guarantee in hardware that both PPIB channels in a PPIB connections have the correct ownership and security.

6.3.5 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
PPIB00 : S	GLOBAL	0x50043000	US	S	NA	No	PPI bridge PPIB00
PPIB00 : NS		0x40043000					
PPIB01 : S	GLOBAL	0x50044000	US	S	NA	No	PPI bridge PPIB01
PPIB01 : NS		0x40044000					
PPIB10 : S	GLOBAL	0x50083000	US	S	NA	No	PPI bridge PPIB10
PPIB10 : NS		0x40083000					
PPIB11 : S	GLOBAL	0x50084000	US	S	NA	No	PPI bridge PPIB11
PPIB11 : NS		0x40084000					
PPIB20 : S	GLOBAL	0x500C3000	US	S	NA	No	PPI bridge PPIB20
PPIB20 : NS		0x400C3000					
PPIB21 : S	GLOBAL	0x500C4000	US	S	NA	No	PPI bridge PPIB21
PPIB21 : NS		0x400C4000					
PPIB22 : S	GLOBAL	0x500C5000	US	S	NA	No	PPI bridge PPIB22
PPIB22 : NS		0x400C5000					
PPIB30 : S	GLOBAL	0x50103000	US	S	NA	No	PPI bridge PPIB30
PPIB30 : NS		0x40103000					

Configuration

Instance	Domain	Configuration
PPIB00 : S	GLOBAL	Bridges PPI channels 0-7 between PPIB_00 (MCU) and PPIB_10 (RADIO)
PPIB00 : NS		
PPIB01 : S	GLOBAL	Bridges PPI channels 0-7 between PPIB_01 (MCU) and PPIB_20 (PERI)
PPIB01 : NS		
PPIB10 : S	GLOBAL	Bridges PPI channels 0-7 between PPIB_10 (RADIO) and PPIB_00 (MCU)
PPIB10 : NS		
PPIB11 : S	GLOBAL	Bridges PPI channels 0-15 between PPIB_11 (RADIO) and PPIB_21 (PERI)
PPIB11 : NS		
PPIB20 : S	GLOBAL	Bridges PPI channels 0-7 between PPIB_20 (PERI) and PPIB_01 (MCU)
PPIB20 : NS		
PPIB21 : S	GLOBAL	Bridges PPI channels 0-15 between PPIB_21 (PERI) and PPIB_11 (RADIO)
PPIB21 : NS		
PPIB22 : S	GLOBAL	Bridges PPI channels 0-3 between PPIB_22 (PERI) and PPIB_30 (LP)
PPIB22 : NS		
PPIB30 : S	GLOBAL	Bridges PPI channels 0-3 between PPIB_30 (LP) and PPIB_22 (PERI)
PPIB30 : NS		

Register overview

Register	Offset	TZ	Description
TASKS_SEND[n]	0x000		This task is unused, but the PPIB provides the SUBSCRIBE task to connect SEND [n] task.
SUBSCRIBE_SEND[n]	0x080		Subscribe configuration for task SEND[n]
EVENTS_RECEIVE[n]	0x100		This event is unused, but the PPIB provides the PUBLISH event to connect RECEIVE [n] event.
PUBLISH_RECEIVE[n]	0x180		Publish configuration for event RECEIVE[n]
OVERFLOW_SEND	0x400		The task overflow for SEND tasks using SUBSCRIBE_SEND. Write 0 to clear.

6.3.5.1 TASKS_SEND[n] (n=0..31)

Address offset: $0x000 + (n \times 0x4)$

This task is unused, but the PPIB provides the SUBSCRIBE task to connect SEND [n] task.

Writes to SEND [n] task are ignored.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value	ID	Value	Description																													
A	W	TASKS_SEND				This task is unused, but the PPIB provides the SUBSCRIBE task to connect SEND [n] task.																													
						Writes to SEND [n] task are ignored.																													
		Trigger		1		Trigger task																													

6.3.5.2 SUBSCRIBE_SEND[n] (n=0..31)

Address offset: $0x080 + (n \times 0x4)$

Subscribe configuration for task SEND[n]

Writes to SEND [n] task are ignored.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID					B																								A					A	A	A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value	ID	Value				Description																															
A	RW	CHIDX			[0..255]				DPPI channel that task SEND[n] will subscribe to																															
B	RW	EN																																						
			Disabled	0	Disable subscription																																			
			Enabled	1	Enable subscription																																			

6.3.5.3 EVENTS_RECEIVE[n] (n=0..31)

Address offset: $0x100 + (n \times 0x4)$

This event is unused, but the PPIB provides the PUBLISH event to connect RECEIVE [n] event.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID					A																																		
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value	ID	Value	Description																																	
A	RW	EVENTS_RECEIVE				This event is unused, but the PPIB provides the PUBLISH event to connect RECEIVE [n] event.																																	
			NotGenerated	0	Event not generated																																		
			Generated	1	Event generated																																		

6.3.5.4 PUBLISH_RECEIVE[n] (n=0..31)

Address offset: $0x180 + (n \times 0x4)$

Publish configuration for event **RECEIVE[n]**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				B																								A				A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value	ID	Value		Description																														
A	RW	CHIDX			[0..255]		DPPI channel that event RECEIVE[n] will publish to																														
B	RW	EN																																			
			Disabled	0	Disable publishing																																
			Enabled	1	Enable publishing																																

6.3.5.5 OVERFLOW.SEND

Address offset: 0x400

The task overflow for SEND tasks using SUBSCRIBE_SEND.

Write 0 to clear.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				f	e	d	c	b	a	Z	Y	X	W	V	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																														
A-f	RW	SEND[i] (i=0..31)			The status for tasks overflow at SUBSCRIBE_SEND[i].																														
			Overflow	1	Task overflow is happened.																														
			NoOverflow	0	Task overflow is not happened.																														

7 Security

The device is designed with state-of-the-art security features.

The main security features of the device are the following:

- Arm TrustZone for memory, peripherals, GPIO pins, PPI channels, and interrupts
- Tamper controller to monitor and prevent physical attacks:
 - Active driven tamper switches (active shield)
 - Signal protectors for critical configuration signals
 - Glitch detectors to guard against fault injection attacks
- Crypto accelerator with built-in self-check and countermeasures
 - Masking against simple and differential power analysis
 - Protection against timing attacks
- FIPS 140-2 random number generator
- Non-volatile memory controller with built-in secure key storage (key management unit)
- Immutable boot region for establishing root of trust
- Authenticated debug to prevent unauthorized access to the debug port

7.1 Memory and peripheral access permissions

Access permissions are controlled by TrustZone, MPC, and SPU security peripherals.

The following figure shows the system security control modules for memory, peripherals, GPIO, and PPI.

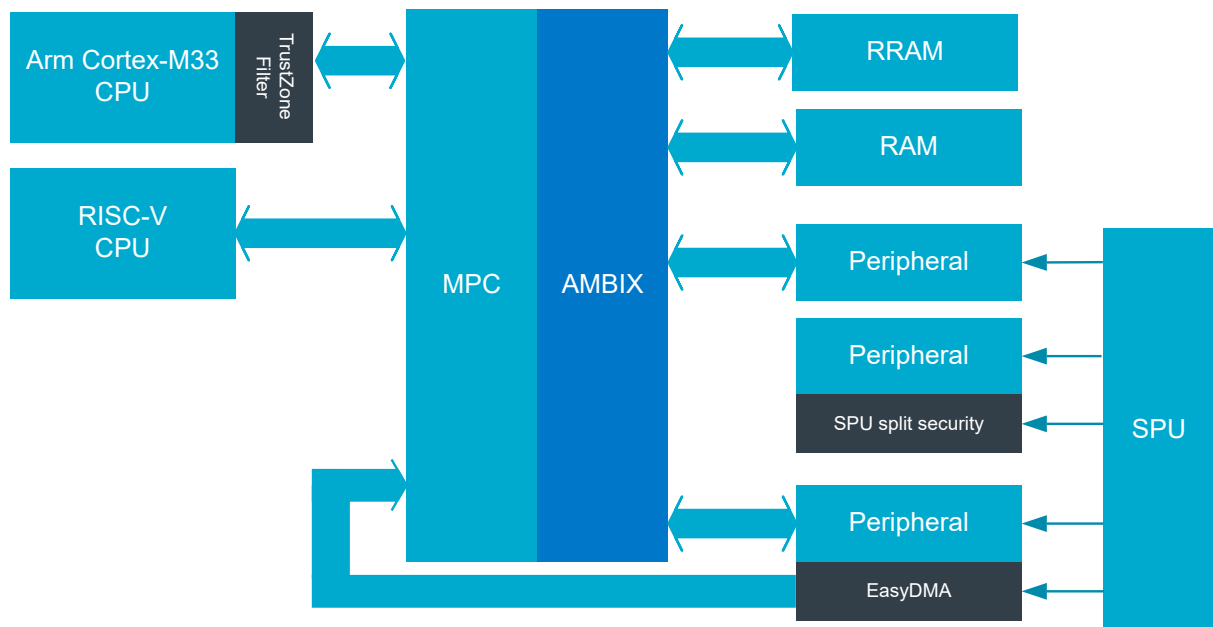


Figure 28: Modules filtering access permissions

The Arm Cortex-M33 CPU enforces TrustZone security internally, before issuing bus transactions. For security checks internal to the Arm Cortex-M33, see [TrustZone security](#) on page 128. After the internal CPU security check, the transaction is available on the bus.

Secure and non-secure memory has to be configured in the SAU and MPC.

The security attribution of a bus transaction from the Arm Cortex-M33 is determined by the CPU, SAU, and IDAU settings. See [TrustZone security](#) on page 128 for more information.

For RISC-V and peripherals, the attribution of the bus transaction is determined by the [SPU](#) settings.

The destination's security attribute is a combination of [MPC](#) and [SPU](#) configurations.

Abbreviation	Description
NS	Non-secure – TrustZone security attribute is non-secure
S	Secure – TrustZone security attribute is secure
NSC	Non-secure callable – TrustZone security attribute is non-secure callable
IDAU	Arm implementation defined attribution unit
SAU	Arm security attribution unit
SPU	Nordic system protection unit
MPC	Nordic memory privilege controller

Table 21: Abbreviations

Memory access overview

The following table lists the security attributes of the bus manager and their access to memory configured as secure and non-secure.

Security attribute of the bus manager	Security attribute of the destination memory	Access successful	MPC bus fault and error event
S	S	Yes	No
NS	S	No	Yes
S	NS	Yes	No
NS	NS	Yes	No

Table 22: Memory access overview

Peripheral access overview

Peripherals are moved in the memory map based on their security association. Non-secure peripherals can be accessed through addresses starting with 0x4 while secure peripherals are accessible in the memory region starting with 0x5.

The security association of each peripheral is controlled via the SPU. Only peripherals with programmable security association can be moved in the memory map.

Security attribute of the bus manager	Security attribute of the destination	Address region	Access successful	SPU bus fault and error event
S	S	0x5	Yes	No
S	S	0x4	No	Yes
NS	S	0x5	No	Yes
NS	S	0x4	No	Yes
S	NS	0x5	No	Yes
S	NS	0x4	Yes	No
NS	NS	0x5	No	Yes
NS	NS	0x4	Yes	No

Table 23: Peripheral access overview

In addition, the following also applies:

- For split security peripherals no bus faults are generated for blocked split security bit accesses. Reads as 0, write is ignored.
- In a split peripheral, access to secure registers using the non-secure 0x4 memory region would be through non-secure transactions and would be blocked. Make sure to use the secure memory region to access secure registers.

7.2 TrustZone security

The security architecture is based on Arm TrustZone.

The Arm Cortex-M based CPU support Arm TrustZone for secure, non-secure, and non-secure callable memory regions.

The security attribution unit (SAU) and implementation defined attribution unit (IDAU) define the access permissions based on the security state.

The IDAU configuration divides system memory space into secure (S) and non-secure (NS) regions. The SAU provides configurable regions for the Arm Cortex-M CPU, and is used to define non-secure callable (NSC) regions.

IDAU preset configuration

IDAU configuration is preset in hardware and is not available for user configuration. The security attribution follows the address map and the peripheral memory space is aliased for the secure and non-secure memory state, as defined in the following table.

Memory map name	Address map	IDAU TrustZone security attribute
Private peripheral bus	0xE0000000 - 0xFFFFFFFF	Not applicable
Device memory	0xA0000000 - 0xDFFFFFFF	NS
External memory	0x60000000 - 0xAFFFFFFF	NS
Peripheral (secure)	0x50000000 - 0x5FFFFFFF	S
Peripheral (non-secure)	0x40000000 - 0x4FFFFFFF	NS
Data memory	0x20000000 - 0x3FFFFFFF	NS
Program memory	0x00000000 - 0x1FFFFFFF	NS

Table 24: IDAU configuration

SAU configuration

The Arm Cortex-M33 CPU must configure its SAU regions when the CPU starts.

SAU configuration registers are documented in the *Arm Cortex-M33 Technical Reference Manual*, revision r1p0.

Before the SAU regions are configured, the CPU assumes the memory map is secure.

TrustZone security attributes

Based on IDAU and SAU configuration, the following table shows the TrustZone security attribute results.

IDAU security attribute	SAU security attribute	Security attribute result
S	NS, NSC, or S	S
NS, NSC, or S	S	S
NS	NS	NS
NS	NSC	NSC

Table 25: TrustZone security attributes

For the memory region that contains the secure gateway instruction branch veneers (entry points), the TrustZone security attribute seen by the Arm Cortex-M must be NSC for the secure functions that are callable from a non-secure program.

Example memory map

The following figure shows an example memory map using SAU regions to provide NS, S, and NSC regions. The figure also includes the required MPC override configuration to ensure correct secure/non-secure system partitioning.

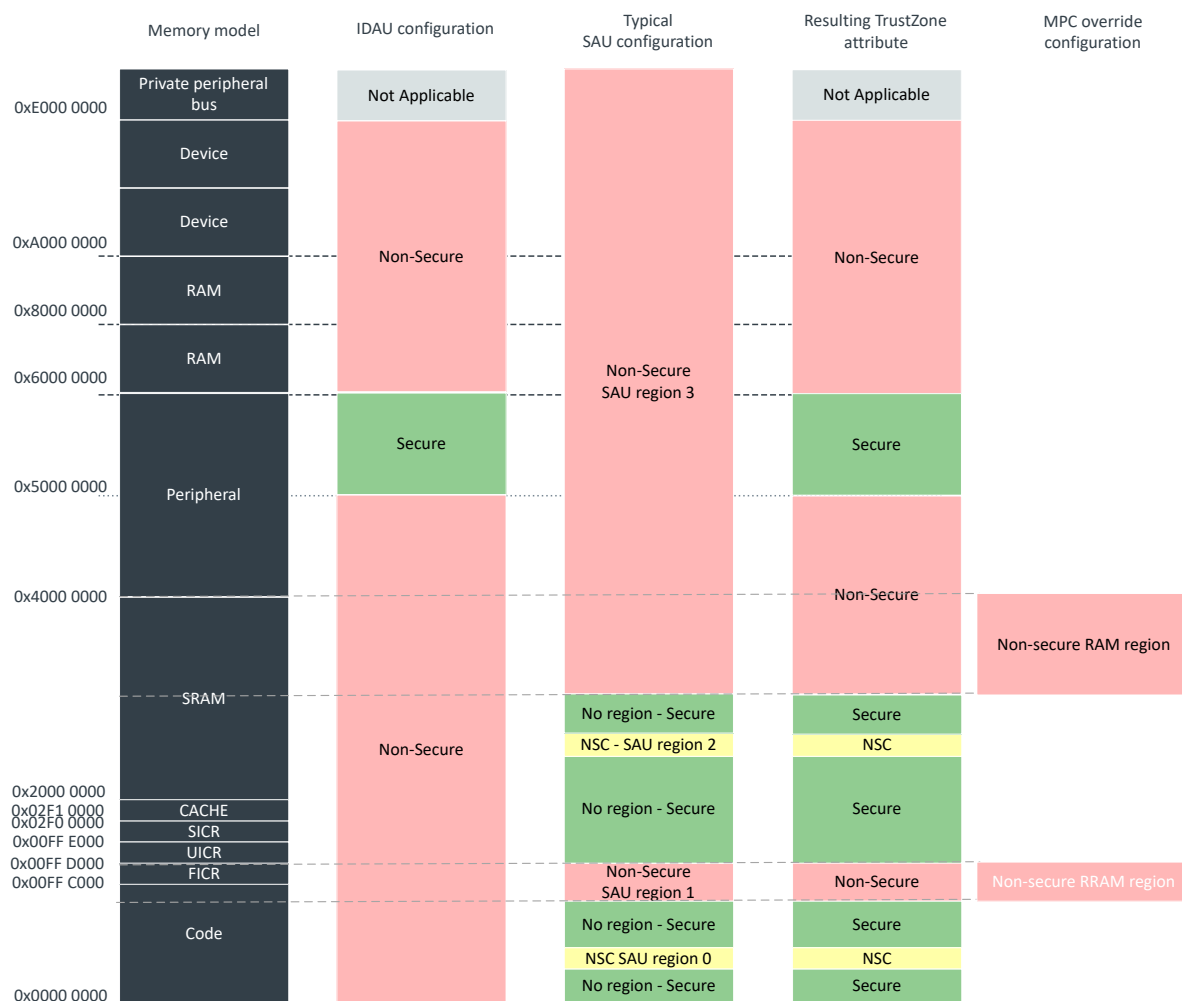


Figure 29: Example memory map security attribution

TrustZone security access

The Arm Cortex-M TrustZone security module generates a CPU SecureFault exception when access is not allowed. The following table shows various scenarios of TrustZone security attributes.

Arm Cortex-M TrustZone security attribute	Destination address security attribute	Secure fault	Access allowed
S	S	No	Yes
S	NS	No	Yes
NS	NS	No	Yes
NS	S	Yes	No

Table 26: TrustZone security access

The first two columns in the table use the TrustZone security attribute from the [TrustZone security attributes](#) table.

The Arm Cortex-M TrustZone security attribute is the TrustZone security attribute seen by the Arm Cortex-M CPU while executing a program. This shows if the Arm Cortex-M CPU program is executed from S, NS, or NSC memory. The NSC for the Arm Cortex-M TrustZone security attribute behaves same as S in the table.

The destination address security attribute is the TrustZone security attribute of the destination address lookup from the SAU and IDAU. It is used by the Arm Cortex-M CPU on the bus transaction.

7.3 Immutable boot region

The device RRAM has a boot region that can be made immutable before the CPU starts up.

Boot initiated from an immutable source allows later boot steps to be performed by authenticated code.

The boot region starts at address `0x00000000`. This address contains the initial secure program counter (PC), the stack pointer (SP), and the interrupt vectors. The size and permissions of the region are configured using UICR register [BOOTCONF](#) on page 70.

After configuration, when there is a device reset, the hardware state-machine reads UICR fields and configures [RRAMC](#). This enforces boot region protection before the Arm Cortex-M33 is released from reset.

For more information about the immutable boot region, see [RRAMC — Resistive random access memory controller](#) on page 53.

7.4 Security attributes

Bus access can have secure or non-secure attribution which follows the transaction through the system.

Non-secure peripherals use non-secure DMA bus transactions. Secure peripherals have configurable DMA security and can generate either secure or non-secure DMA bus transactions. The peripheral security is configured using [SPU — System protection unit](#) on page 180.

For Arm Cortex CPUs, see the Arm TrustZone architecture document for more details on security.

7.5 Peripherals with split security access

Some peripherals have split security access, meaning they can handle both secure and non-secure access. A subset of the peripheral's functions can be secure, while another subset is non-secure. The security is configured using SPU registers, as explained in the following sections.

The peripheral instantiation table in [Instantiation](#) on page 216 details the peripherals with split access.

Split security access is handled either on the register level or on the bit level, as explained in the following sections.

Register level split security access

For this group of peripherals, security is enforced at the register level. Split security settings apply for the entire register. Illegal access to the register will trigger a security fault. For example, if a register is configured as secure and the register is accessed from non-secure code, a security fault with a bus fault will be generated. A security fault due to an illegal access triggers the SPU event [PERIPHACCERR](#).

Bit level split security access

For this group of peripherals, security is enforced at the register bit level. Split security settings are applied to individual bits of the register. The register supports access from both secure and non-secure code.

No exceptions are triggered for the access, however the following apply:

- Writing a secure bit from non-secure code will have no effect
- Reading a register from non-secure code will return 0 for all bits that are secure.

For example, if bit i is configured as being secure, then the following apply:

- Non-secure write access to the register will not change the bit i
- Non-secure read access to the register will read 0 for the bit at position i

Interrupts

Some peripherals have split security interrupts. This means the interrupt can be configured with a security attribute.

An interrupt may be generated during secure or non-secure execution, and the interrupt handler is executed based on the interrupt's security attribute.

Interrupts implement split security at the register level (see above). For instance, if interrupt 0 is configured as secure and there is a non-secure read/write access to registers INTEN0, INTENSET0, INTENCLR0, or INTPEND0, a security fault with a bus fault will be generated.

Interrupts are generated when enabled events are triggered. When an event for a split security interrupt is triggered, the following applies.

- If an interrupt is configured as secure, an event associated to either a secure or non-secure feature can trigger the interrupt.
- If an interrupt is configured as non-secure, only events associated with non-secure features can trigger the interrupt.

An attempt to enable an interrupt for an event that does not match the ownership and security settings of the interrupt will be ignored and a security fault is not generated.

A non-secure event can be enabled to trigger a secure interrupt.

7.5.1 CRACEN

CRACEN protects the Protected RAM and the SEED register from being accessed by the CPU.

Only KMU is able to push assets to the Protected RAM and the SEED register. The CPU does not have access to these. Protection is built into the hardware and does not need configuration.

7.5.2 DPPIC

Individual DPPI channels and channel groups can have independent security attributes and are defined as either secure or non-secure. DPPI supports split security, handling both secure and non-secure access.

DPPI channels

A peripheral configured as non-secure can only subscribe to or publish on non-secure DPPI channels. A peripheral configured as secure can access all DPPI channels. An attempt by a non-secure peripheral to subscribe to or publish on a DPPI channel configured as secure is ignored and a PPI event is not issued.

DPPI channels are enabled or disabled through individual bits in registers CHEN, CHENSET, and CHENCLR.

The security of a DPPI channel is configured using [FEATURE.DPPIC.CH\[n\]](#) ($n=0..23$) on page 188.

DPPI channel groups

Channels can be grouped, which allows them to be enabled or disabled collectively.

A channel group is either secure or non-secure.

- Secure channel group – includes both secure and non-secure DPPI channels
- Non-secure channel group – only includes non-secure DPPI channels

An attempt to include a secure DPPI channel in a non-secure DPPI channel group is ignored.

Registers CHG[n], TASKS_CHG[n].EN, TASKS_CHG[n].DIS, SUBSCRIBE_CHG[n].EN, and SUBSCRIBE_CHG[n].DIS configure the DPPI channel groups. A security fault is triggered when an illegal access is made to these registers.

DPPIC subscribe to DPPIC channels through the SUBSCRIBE_CHG[] registers to trigger the task for enabling or disabling channel groups. An event from a secure channel is ignored if the group subscribing to that channel is non-secure. A secure group can subscribe to a non-secure channel or a secure channel.

The security of a DPPIC channel group is configured using [FEATURE.DPPIC.CHG\[n\] \(n=0..7\)](#) on page 188.

7.5.3 GPIO

GPIO pins can be either secure or non-secure.

GPIO supports split security, meaning the GPIO pins and registers can be accessed from both secure and non-secure peripherals.

A peripheral configured as non-secure can only access non-secure pins. A peripheral configured as secure will be able to access all pins. An attempt to access a pin configured as secure by a non-secure peripheral is ignored.

GPIO pins can be read and written through individual bits in the GPIO port registers OUT, OUTSET, OUTCLR, and IN. GPIO pin direction is configured individually using bits in registers DIR, DIRSET, and DIRCLR. An attempt to access bits with a different security setting is ignored. Writing to these bits will have no effect, and read access returns a zero value.

The LATCH register has split security. Non-secure code can only read the state of the non-secure pins, while the secure pins read as zero. Secure code is able to read the state of all pins.

The DETECTMODE register applies to the entire port (both secure and non-secure pins), and determines if the latched or non-latched signals will generate the DETECT signal.

Pin security configuration

Access to device pins can be controlled by SPU. A pin can be set as secure so that only secure peripherals or secure code can access it. Pins set as non-secure can be accessed by both secure and non-secure peripherals or code.

The security attribute of each pin can be individually configured in [FEATURE.GPIO\[n\].PIN\[o\] \(n=0..2\) \(o=0..31\)](#) on page 189. When the secure attribute (SECATTR) is set for a pin, only peripherals that have the secure attribute set will be able to read the value of the pin or change it.

Peripherals can select the pins they need access to through their PSEL registers. If a peripheral has its attribute set to non-secure, but one of its PSEL registers selects a pin with the attribute set to secure, the SPU controlled logic will ensure that the pin selection is not propagated. In addition, the pin value will always be read as zero to prevent a non-secure peripheral from obtaining a value from a secure pin. Access to other pins with attribute set as non-secure will not be blocked.

Pins can also be dedicated to peripherals by using the CTRLSEL field in the GPIO PIN_CNF[n] register. For pins controlled using CTRLSEL, the SPU PIN security setting is bypassed and pin access is controlled by the peripheral. This is illustrated in the following figure.

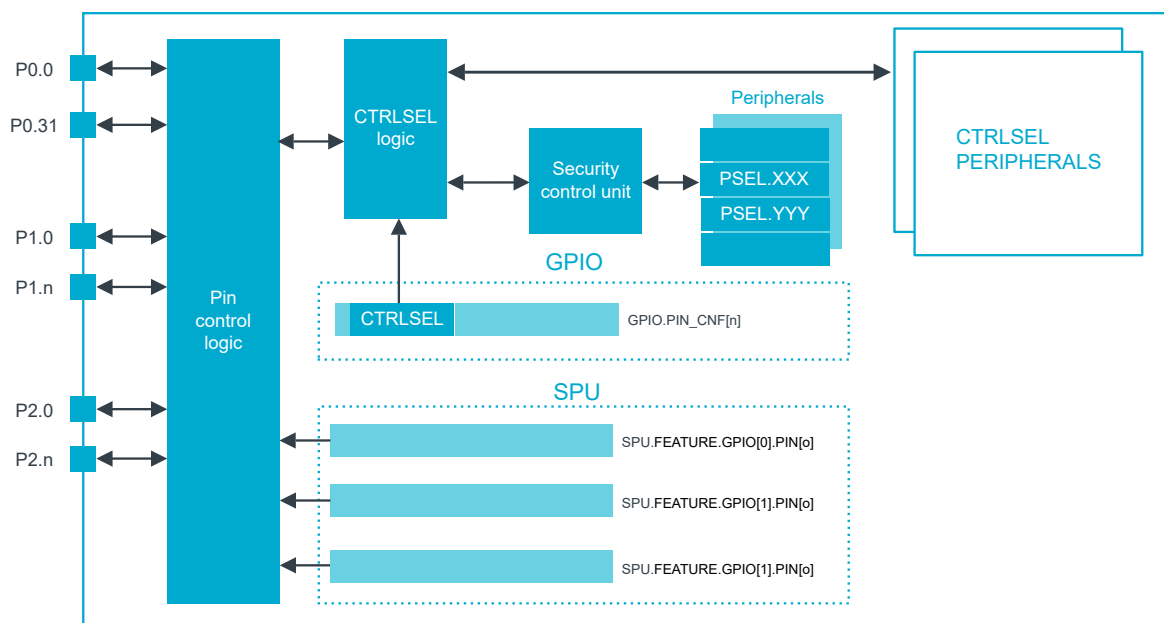


Figure 30: Pin access using CTRLSEL

7.5.4 GPIOTE

Individual GPIOTE channels and interrupts can have independent security settings and are defined as secure or non-secure.

GPIOTE channels

GPIOTE channel security is configured using `FEATURE.GPIOTE[n].CH[o]` ($n=0..1$) ($o=0..7$) on page 189.

A GPIOTE channel configured as secure can only be used by secure code to send tasks and receive events. A GPIOTE channel configured as non-secure can be used by both secure and non-secure code to send tasks and receive events.

GPIOTE channel tasks and events can be configured with a specific GPIO pin.

- A secure GPIOTE channel can be configured with both secure and non-secure GPIO pins
- A non-secure GPIOTE channel can only be configured with non-secure GPIO pins

See `CONFIG[n].PSEL` and `CONFIG[n].PORT` registers for more information.

GPIOTE channels that are not configured as described will not write to the pin when triggering the `SET[n]`, `CLR[n]`, and `OUT[n]` tasks, and will not generate the `IN[n]` event with changes in the pin polarity.

A GPIOTE channel configured as secure has the following properties:

- Can only be used during secure code execution to trigger `SET[n]`, `CLR[n]`, and `OUT[n]` tasks
- Can only generate `IN[n]` events during secure code execution
- The corresponding `CONFIG[n]` register can only be accessed during secure code execution

A GPIOTE channel n configured as non-secure has the following properties:

- Can be used during both secure and non-secure code execution to trigger `SET[n]`, `CLR[n]`, and `OUT[n]` tasks
- Can generate `IN[n]` events during both secure and non-secure code execution
- The corresponding `CONFIG[n]` register can be accessed during both secure and non-secure code execution

A security fault is triggered when there is an access violation when accessing registers `TASKS_SET[n]`, `TASKS_CLR[n]`, `TASKS_OUT[n]`, `EVENTS_IN[n]`, or `CONFIG[n]`.

GPIO channels can connect to PPI channels in order to send and receive events from other peripherals. GPIO channels can only publish or subscribe from DPPI channels that have the correct security attribute. An attempt to subscribe or publish on a DPPI channel configured as secure by a non-secure GPIO channel is ignored. A secure GPIO channel can subscribe or publish to both secure and non-secure DPPI channels.

GPIO interrupts

The security of the GPIO interrupt is configured using `FEATURE.GPIO[n].INTERRUPT[o]` ($n=0..1$) ($o=0..7$) on page 189.

A secure fault is triggered when non-secure code attempts to access registers INTENSET/INTENCLR on a secure GPIO interrupt.

GPIO interrupt i can only be generated by IN[j] event if interrupt i and channel j have the correct security attribute.

A secure GPIO interrupt can be triggered by an event generated by a secure and non-secure GPIO channel.

A non-secure GPIO interrupt can only be triggered by an event generated by non-secure GPIO channel.

7.5.5 GRTC

GRTC is implemented with split security, meaning it handles access from both secure and non-secure code. Individual GRTC SYSCOUNTER compare/capture channels and interrupts can have independent security settings that define them as secure or non-secure.

SYSCOUNTER compare/capture channels

The SYSCOUNTER compare/capture channels have the following security:

- Secure — The channel and its associated registers, trigger/subscribe tasks, and receive/publish events can only be accessed by secure code.
- Non-secure — The channel and its associated registers, trigger/subscribe tasks, and receive/publish events can be accessed by secure and non-secure code.

GRTC interrupts

GRTC interrupts can be defined as secure or non-secure.

A security fault is triggered when an invalid access targets registers INTEN/INTENSET/INTENCLR/INTPEND associated with an GRTC interrupt.

GRTC interrupt can only be generated by a COMPARE[j] event if the interrupt and channel have the correct security attribute.

A secure GRTC interrupt can be triggered by a secure or non-secure GRTC channel.

A non-secure GRTC interrupt can be triggered by an event generated by non-secure GRTC channel. An event generated by secure GRTC channel cannot trigger the interrupt.

7.6 Physical security

The device has countermeasures for physical attacks. It can detect and report fault injection attacks such as voltage glitching or electromagnetic fault injection.

The external active shield I/O interface is provided to facilitate PCB and product level security. It can detect if a product's encapsulation has been opened, or if the product has been tampered with. For more information, see [TAMPC — Tamper controller](#) on page 192.

The crypto accelerator (CRACEN) peripheral is protected against differential power analysis (DPA) attacks. The AES, SM4, and public key acceleration engines all have countermeasures against DPA attacks and will report attack attempts. For more information, see [CRACEN — Cryptographic accelerator engine](#) on page 136.

7.7 Security components

7.7.1 CRACEN — Cryptographic accelerator engine

The main features of the CRACEN peripheral are the following:

- Cryptomaster (Symmetric cryptographic engines and digest engines)
 - AES
 - Supports 128-, 192-, and 256-bit keys
 - Masking countermeasures
 - Context switching
 - HASH, including MD5, SHA1, SHA224, SHA256, SHA384, SHA512, and HMAC
 - ChaChaPoly
 - SHA3
 - SM4
- Public Key cryptographic engine (PKE) and Isolated Key Generator (IKG)
 - Modular exponentiation for RSA with and without CRT; 4096-bit maximum operand size
 - Elliptic Curve Cryptography (ECC) with 640-bit maximum operand size
 - Digital Signature Algorithm (DSA) and Elliptic Curve Digital Signature Algorithm (ECDSA, EC-KCDSA, and EdDSA), with 4096-bit maximum operand size
 - Diffie-Hellman (D-H and ECDH) key exchange
- Random Number Generator (RNG)

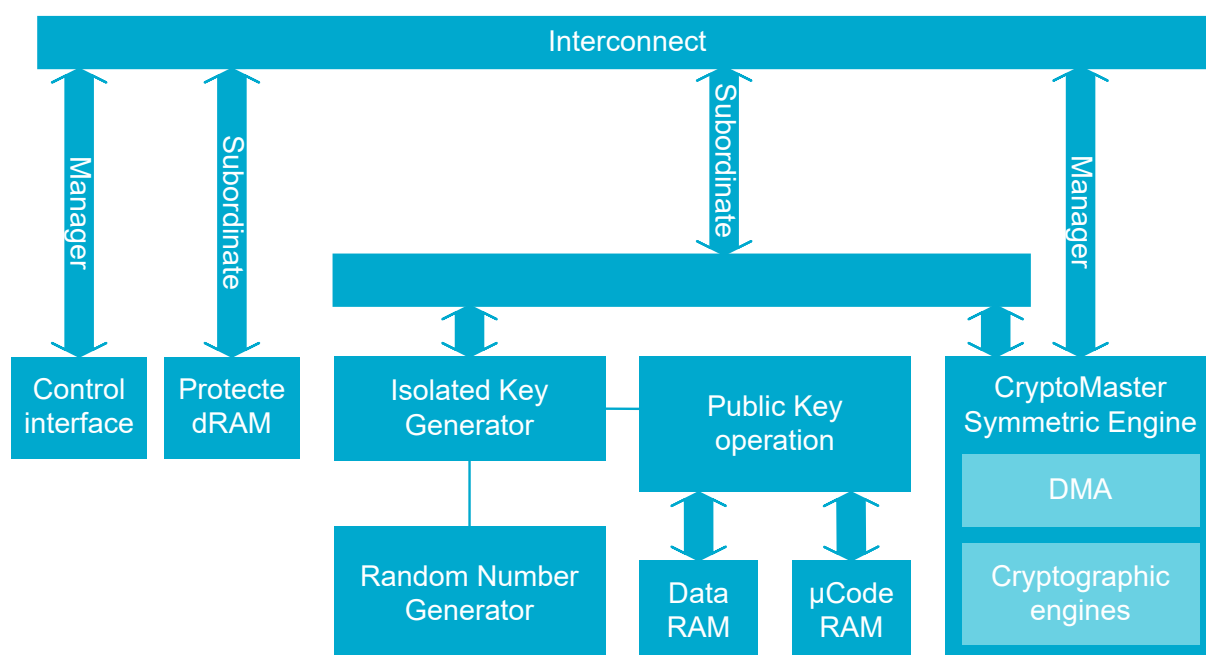


Figure 31: Cryptographic accelerator engine block diagram

7.7.1.1 Initialization

Before the CRACEN peripheral can be used, it must be configured.

At reset, each hardware crypto operation category is disabled and must be individually enabled using the [ENABLE](#) register. To use CRACEN, the desired module must first be enabled. Ongoing crypto operations will complete even if the module is disabled during the operation.

Before transferring data to CRACENCORE, CRACEN must be enabled using [ENABLE](#) on page 142.

When CRACEN is enabled, it will erase the PKE data RAM by starting a zeroization process. When the PKBUSY field of the [PK.STATUS](#) is cleared, the zeroization operation is complete. The PKE engine is not available until the zeroization process has finished.

7.7.1.2 Protected RAM

The protected RAM can be locked and is retained to store symmetric keys. Protected RAM is only writable from KMU and cannot be accessed by the CPU. After KMU has pushed keys into the protected RAM, [PROTECTEDDRAMLOCK](#) must be set to `Enabled` before CRACEN can access and use the keys.

Register [PROTECTEDDRAMLOCK](#) is a write-once register, and cannot be changed until the next device reset.

The following areas are defined for the protected RAM.

Protected RAM		
Address	End address	Description
0x51810040	0x5181005F	AES Protected key 0
0x51810060	0x5181007F	AES Protected key 1
0x51810080	0x5181008F	SM4 Protected key 0
0x51810090	0x5181009F	SM4 Protected key 1
0x518100A0	0x518100AF	SM4 Protected key 2
0x518100B0	0x518100BF	SM4 Protected key 3

7.7.1.3 Countermeasures

CRACEN contains security countermeasures to prevent malicious usage.

The following engines implement countermeasures:

- AES – Masking against Simple Power Analysis (SPA) and Differential Power Analysis (DPA)
- SM4 – Masking against SPA and DPA
- IKG/PKE – Protection against timing attacks and DPA

If CRACEN IKG/PKE is used maliciously, a [TAMPC](#) event will be generated and countermeasures enacted according to the [TAMPC](#) configuration. The countermeasures are controlled by [TAMPC](#). The bits in [TAMPC](#) that control the countermeasures have lock bits.

7.7.1.4 Isolated Key Generator

The Isolated Key Generator (IKG) is a module that derives symmetric and asymmetric keys from the unique seed and optional personalization string.

After IKG has been enabled, CRACEN performs an IKG health test. The CTRDRBGBUSY field of the [IKG.STATUS](#) is cleared when the operation has completed. IKG is started by writing to the [IKG.START](#) register. The generated IKG keys are valid as long as CRACEN remains enabled. For details on enabling and disabling CRACEN, see [ENABLE](#) on page 142.

The IKG derives the following keys from seed upon request:

- One 256-bit ECC P-256 key
- Two 256-bit AES keys

Note: The IKG generated keys are not directly accessible by CPU but are used by the PKE and AES engines. The IKG generated AES keys are not the same as protected keys in protected RAM, but can be used by the same AES engine.

7.7.1.4.1 Loading seed to IKG

The seed used by the IKG to generate keys must be pushed by the **KMU** to the **SEED** register and marked as valid before the keys can be generated.

To create and derive a seed the following sequence of operations are needed.

1. Create device unique seed:
 - a. Create 3 x 128 bit random number using CRACEN RNG
 - b. Provision random number to KMU slots, e.g. 0, 1, and 2 (128 bits in each slot)
 1. SRC.DEST = CRACEN.SEED[n], where n=0, 4, and 8.
 2. SRC.VALUE = random[i], where i=0,1, and 2 (i.e. random number results from CRACEN.RND operation above)
2. Load seed from KMU to CRACEN:
 - a. Push the KMU slots where the SEED is stored, e.g. KMU slots 0, 1, and 2
 - b. Write **CRACEN.SEEDVALID** register to mark the seed as valid for the IKG
 - c. To prevent any subsequent changes to the SEED, write **CRACEN.SEEDLOCK** register.

Note: Any IKG key generations without valid seed (**CRACEN.SEEDVALID**) will fail.

7.7.1.5 Low power

To ensure lowest possible power consumption when the peripheral is not needed, disable CRACEN.

Make sure operations are complete before disabling the peripheral with the **ENABLE** register.

7.7.1.6 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
CRACEN	GLOBAL	0x50048000	HF	S	NSA	No	Crypto accelerator

Configuration

Instance	Domain	Configuration
CRACEN	GLOBAL	<p>Access to CRACEN registers is blocked while KMU is performing a PUSH operation. CRACEN cannot write RRAM.</p> <p>CRACEN CRYPTOACCELERATOR specific configuration registers included</p> <p>PKE data (address 0x51808000) must be read and written using aligned access, i.e. using an operation where a word-aligned address is used for a word, or a halfword-aligned address is used for a halfword access.</p> <p>PKE code (address 0x5180C000) must be read and written using aligned access, i.e. using an operation where a word-aligned address is used for a word, or a halfword-aligned address is used for a halfword access.</p>

Register overview

Register	Offset	TZ	Description
EVENTS_CRYPTOMASTER	0x100		Event indicating that interrupt triggered at Cryptomaster
EVENTS_RNG	0x104		Event indicating that interrupt triggered at RNG
EVENTS_PKEIKG	0x108		Event indicating that interrupt triggered at PKE or IKG
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
INTPEND	0x30C		Pending interrupts
ENABLE	0x400		Enable CRACEN peripheral modules.
SEEDVALID	0x404		Marks the SEED register as valid.
SEED[n]	0x410		Seed word [n] for symmetric and asymmetric key generation.
			This register is only writable from KMU.
SEEDLOCK	0x440		Lock the access to the SEED register.
PROTECTEDRAMLOCK	0x444		Lock the access to the protected RAM.

7.7.1.6.1 EVENTS_CRYPTOMASTER

Address offset: 0x100

Event indicating that interrupt triggered at Cryptomaster

The interrupt source must be cleared at Cryptomaster before clearing this event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value			Description																												
A	RW	EVENTS_CRYPTOMASTER					Event indicating that interrupt triggered at Cryptomaster																												
							The interrupt source must be cleared at Cryptomaster before clearing this event.																												
			NotGenerated	0			Event not generated																												
			Generated	1			Event generated																												

7.7.1.6.2 EVENTS_RNG

Address offset: 0x104

Event indicating that interrupt triggered at RNG

The interrupt source must be cleared at RNG before clearing this event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_RNG						Event indicating that interrupt triggered at RNG																											
								The interrupt source must be cleared at RNG before clearing this event.																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

7.7.1.6.3 EVENTS_PKEIKG

Address offset: 0x108

Event indicating that interrupt triggered at PKE or IKG

The interrupt source must be cleared at PKE or IKG before clearing this event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_PKEIKG			Event indicating that interrupt triggered at PKE or IKG																														
					The interrupt source must be cleared at PKE or IKG before clearing this event.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

7.7.1.6.4 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CRYPTOMASTER			Enable or disable interrupt for event CRYPTOMASTER																														
					The interrupt source must be cleared at Cryptomaaster before clearing this event.																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
B	RW	RNG			Enable or disable interrupt for event RNG																														
					The interrupt source must be cleared at RNG before clearing this event.																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
C	RW	PKEIKG			Enable or disable interrupt for event PKEIKG																														
					The interrupt source must be cleared at PKE or IKG before clearing this event.																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

7.7.1.6.5 INTENSET

Address offset: 0x304

Enable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	CRYPTOMASTER			Write '1' to enable interrupt for event CRYPTOMASTER																													
					The interrupt source must be cleared at Cryptomaaster before clearing this event.																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
B	RW	RNG			Write '1' to enable interrupt for event RNG																													
					The interrupt source must be cleared at RNG before clearing this event.																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
C	RW	PKEIKG			Write '1' to enable interrupt for event PKEIKG																													
					The interrupt source must be cleared at PKE or IKG before clearing this event.																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

7.7.1.6.6 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CRYPTOMASTER			Write '1' to disable interrupt for event CRYPTOMASTER																														
					The interrupt source must be cleared at Cryptomaaster before clearing this event.																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	RNG			Write '1' to disable interrupt for event RNG																														
					The interrupt source must be cleared at RNG before clearing this event.																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	PKEIKG			Write '1' to disable interrupt for event PKEIKG																														
					The interrupt source must be cleared at PKE or IKG before clearing this event.																														
			Clear	1	Disable																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
			Disabled	0				Read: Disabled																											
			Enabled	1				Read: Enabled																											

7.7.1.6.7 INTPEND

Address offset: 0x30C

Pending interrupts

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																				C	B	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	R	CRYPTOMASTER			Read pending status of interrupt for event CRYPTOMASTER																																	
					The interrupt source must be cleared at Cryptomaaster before clearing this event.																																	
			NotPending	0	Read: Not pending																																	
			Pending	1	Read: Pending																																	
B	R	RNG			Read pending status of interrupt for event RNG																																	
					The interrupt source must be cleared at RNG before clearing this event.																																	
			NotPending	0	Read: Not pending																																	
			Pending	1	Read: Pending																																	
C	R	PKEIKG			Read pending status of interrupt for event PKEIKG																																	
					The interrupt source must be cleared at PKE or IKG before clearing this event.																																	
			NotPending	0	Read: Not pending																																	
			Pending	1	Read: Pending																																	

7.7.1.6.8 ENABLE

Address offset: 0x400

Enable CRACEN peripheral modules.

Each module of CRACEN can be enabled individually. When any of these modules are not in use, it can be disabled to save power. The module you want to use must first be enabled. Any ongoing crypto operations will be finished even if the module is disabled during the operation.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																				C	B	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	CRYPTOMASTER			Enable cryptomaster																																	
			Disabled	0	Cryptomaster disabled.																																	
			Enabled	1	Cryptomaster enabled.																																	
B	RW	RNG			Enable RNG																																	
			Disabled	0	RNG disabled.																																	
			Enabled	1	RNG enabled.																																	
C	RW	PKEIKG			Enable PKE and IKG																																	
			Disabled	0	PKE and IKG disabled.																																	
			Enabled	1	PKE and IKG enabled.																																	

7.7.1.6.9 SEEDVALID

Address offset: 0x404

Marks the [SEED](#) register as valid.

This register must be written after the [SEED](#) register is written, and before the IKG is started.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																					A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																
A	RW	VALID			Marks the SEED as valid																																
			Disabled	0	Valid disabled.																																
			Enabled	1	Valid enabled.																																

7.7.1.6.10 SEED[n] (n=0..11)

Address offset: 0x410 + (n × 0x4)

Seed word [n] for symmetric and asymmetric key generation.

This register is only writable from KMU.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

7.7.1.6.11 SEEDLOCK

Address offset: 0x440

Lock the access to the [SEED](#) register.

Note. If [SEEDVALID](#) was not written prior to SEEDLOCK, the write to SEEDLOCK will also mark the [SEED](#) as valid.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	ENABLE				Enable the lock																													
						Only possible to write a value 1.																													
			Disabled	0		Lock disabled.																													
			Enabled	1		Lock enabled.																													

7.7.1.6.12 PROTECTEDRAMLOCK

Address offset: 0x444

Lock the access to the protected RAM.

Bit number			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID			A																															
Reset 0x00000000			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID		Value		Description																											
A	RW	ENABLE					Enable the lock																											
							Only possible to write a value 1.																											
			Disabled		0		Lock disabled.																											
			Enabled		1		Lock enabled.																											

7.7.1.7 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
CRACENCORE	GLOBAL	0x51800000	HF	S	NSA	No	CRACEN core

Configuration

Instance	Domain	Configuration
CRACENCORE	GLOBAL	CRYPTMSTRDMA registers included CRYPTMSTRHW registers included RNGCONTROL registers included PK registers included IKG registers included RNGDATA registers included PKDATAMEMORY registers included PKUCODE registers included Using CRACENCORE configuration reset values

Register overview

Register	Offset	TZ	Description
CRYPTMSTRDMA.FETCHADRLSB	0x000		Fetch Address Least Significant Bit
CRYPTMSTRDMA.FETCHADDRMSB	0x004		Fetch Address Most Significant Bit
CRYPTMSTRDMA.FETCHLEN	0x008		Fetch Length
CRYPTMSTRDMA.FETCHTAG	0x00C		Fetch Tag
CRYPTMSTRDMA.PUSHADRLSB	0x010		Push Address Least Significant Bit
CRYPTMSTRDMA.PUSHADDRMSB	0x014		Push Address Most Significant Bit
CRYPTMSTRDMA.PUSHLEN	0x018		Push Length
CRYPTMSTRDMA.INTEN	0x01C		Interrupt Enable
CRYPTMSTRDMA.INTENSET	0x020		Interrupt Set
CRYPTMSTRDMA.INTENCLR	0x024		Interrupt Clear
CRYPTMSTRDMA.INTSTATRAW	0x028		Interrupt Status Raw
CRYPTMSTRDMA.INTSTAT	0x02C		Interrupt Status
CRYPTMSTRDMA.INTSTATCLR	0x030		Interrupt Status Clear
CRYPTMSTRDMA.CONFIG	0x034		Configuration
CRYPTMSTRDMA.START	0x038		Start
CRYPTMSTRDMA.STATUS	0x03C		Status
CRYPTMSTRHW.INCLIPSHWCFG	0x400		Included IPs Hardware configuration
CRYPTMSTRHW.BA411EAESHWCFG1	0x404		Generic g_AesModesPoss value.
CRYPTMSTRHW.BA411EAESHWCFG2	0x408		Generic g_CtrSize value.
CRYPTMSTRHW.BA413HASHHWCFG	0x40C		Generic g_Hash value
CRYPTMSTRHW.BA418SHA3HWCFG	0x410		Generic g_Sha3CtxEn value.
CRYPTMSTRHW.BA419SM4HWCFG	0x414		Generic g_SM4ModesPoss value.
CRYPTMSTRHW.BA424ARIAHWCFG	0x418		Generic g_aria_modePoss value.
RNGCONTROL.CONTROL	0x1000		Control register
RNGCONTROL.FIFOLEVEL	0x1004		FIFO level register.
RNGCONTROL.FIFOTHRESHOLD	0x1008		FIFO threshold register.
RNGCONTROL.FIFODEPTH	0x100C		FIFO depth register.
RNGCONTROL.KEY[n]	0x1010		Key register.
RNGCONTROL.TESTDATA	0x1020		Test data register.
RNGCONTROL.REPEATTHRESHOLD	0x1024		Repetition Test Count Cut-Off value.
RNGCONTROL.PROPTHRESHOLD	0x1028		Adaptive Proportion Test (1024-sample window) Cut-Off value.
RNGCONTROL.STATUS	0x1030		Status register.
RNGCONTROL.INITWAITVAL	0x1034		Initial wait counter value.
RNGCONTROL.DISABLEOSC[n]	0x1038		Disable oscillator rings $\#n \cdot 32$ to $\#((n+1) \cdot 32) - 1$.
RNGCONTROL.SWOFFTMRVAL	0x1040		Switch off timer value.
RNGCONTROL.CLKDIV	0x1044		Sample clock divider.
RNGCONTROL.AIS31CONF0	0x1048		AIS31 configuration register 0.
RNGCONTROL.AIS31CONF1	0x104C		AIS31 configuration register 1.
RNGCONTROL.AIS31CONF2	0x1050		AIS31 configuration register 2.
RNGCONTROL.AIS31STATUS	0x1054		AIS31 status register.
RNGCONTROL.HWCONFIG	0x1058		Hardware configuration register.
RNGCONTROL.FIFO[n]	0x1080		FIFO data
PK.POINTERS	0x2000		Pointers register.
PK.COMMAND	0x2004		Command register.
PK.CONTROL	0x2008		Command register.
PK.STATUS	0x200C		Status register.
PK.TIMER	0x2014		Timer register.
PK.HWCONFIG	0x2018		Hardware configuration register.
PK.OPSIZE	0x201C		Operand size register.
PK.RAMERRORINJECT	0x2040		RAM error injection register.
PK.RAMERRORSTATUS	0x2044		RAM error status register.

Register	Offset	TZ	Description
IKG.START	0x3000		Start register.
IKG.STATUS	0x3004		Status register.
IKG.INITDATA	0x3008		InitData register.
IKG.NONCE	0x300C		Nonce register.
IKG.PERSONALISATIONSTRING	0x3010		Personalisation String register.
IKG.RESEEDINTERVALLSB	0x3014		Reseed Interval LSB register.
IKG.RESEEDINTERVALMSB	0x3018		Reseed Interval MSB register.
IKG.PKECONTROL	0x301C		PKE Control register.
IKG.PKECOMMAND	0x3020		PKE Command register.
IKG.PKESTATUS	0x3024		PKE Status register.
IKG.SOFTTRST	0x3028		SoftRst register.
IKG.HWCONFIG	0x302C		HwConfig register.

7.7.1.7.1 CRYPTMSTRDMA.FETCHADDRLSB

Address offset: 0x000

Fetch Address Least Significant Bit

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value	ID	Value										Description																					
A	RW	FETCHADDRLSB																																		

7.7.1.7.2 CRYPTMSTRDMA.FETCHADDRMSB

Address offset: 0x004

Fetch Address Most Significant Bit

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value	ID	Value										Description																					
A	RW	FETCHADDRMSB																																		

7.7.1.7.3 CRYPTMSTRDMA.FETCHLEN

Address offset: 0x008

Fetch Length

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					D	C	B	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value	ID	Value	Description																														
A	RW	FETCHLEN																																		
B	RW	FETCHCSTADDR																																		
C	RW	FETCHREALIGN																																		
D	RW	FETCHZPADDING																																		

7.7.1.7.4 CRYPTMSTRDMA.FETCHTAG

Address offset: 0x00C

Fetch Tag

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																													
ID	A A																													
Reset 0x00000000	0 0																													
ID	R/W	Field	Value ID	Value	Description																									
A	RW	FETCHTAG																												

7.7.1.7.5 CRYPTMSTRDMA.PUSHADDRLSB

Address offset: 0x010

Push Address Least Significant Bit

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																													
ID	A A																													
Reset 0x00000000	0 0																													
ID	R/W	Field	Value ID	Value	Description																									
A	RW	PUSHADDRLSB																												

7.7.1.7.6 CRYPTMSTRDMA.PUSHADDRMSB

Address offset: 0x014

Push Address Most Significant Bit

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																													
ID	A A																													
Reset 0x00000000	0 0																													
ID	R/W	Field	Value ID	Value	Description																									
A	RW	PUSHADDRMSB																												

7.7.1.7.7 CRYPTMSTRDMA.PUSHLEN

Address offset: 0x018

Push Length

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																													
ID	D C B A																													
Reset 0x00000000	0 0																													
ID	R/W	Field	Value ID	Value	Description																									
A	RW	PUSHLEN																												
B	RW	PUSHCSTADDR																												
C	RW	PUSHREALIGN																												
D	RW	PUSHDISCARD																												

7.7.1.7.8 CRYPTMSTRDMA.INTEN

Address offset: 0x01C

Interrupt Enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																			
ID																																				A A A A A A			
Reset 0x00000000				0 0																																			
ID	R/W	Field	Value ID	Value				Description																															
A	RW	INTEN																																					

7.7.1.7.9 CRYPTMSTRDMA.INTENSET

Address offset: 0x020

Interrupt Set

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																												A	A	A	A	A		
Reset 0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																													
A	W	INTENSET																																

7.7.1.7.10 CRYPTMSTRDMA.INTENCLR

Address offset: 0x024

Interrupt Clear

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																																			
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	INTENCLR																																	

7.7.1.7.11 CRYPTMSTRDMA.INTSTATRAW

Address offset: 0x028

Interrupt Status Raw

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	INTSTATRAW																																	

7.7.1.7.12 CRYPTMSTRDMA.INTSTAT

Address offset: 0x02C

Interrupt Status

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	INTSTAT																																	

7.7.1.7.13 CRYPTMSTRDMA.INTSTATCLR

Address offset: 0x030

Interrupt Status Clear

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID										A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID				Value				Description																																
A	W	INTSTATCLR																																									

7.7.1.7.14 CRYPTMSTRDMA.CONFIG

Address offset: 0x034

Configuration

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	FETCHCTRLINDIRECT																																	
B	RW	PUSHCTRLINDIRECT																																	
C	RW	FETCHSTOP																																	
D	RW	PUSHSTOP																																	
E	RW	SOFTTRST																																	

7.7.1.7.15 CRYPTMSTRDMA.START

Address offset: 0x038

Start

Bit number			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																	B	A
Reset 0x00000000			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID		Value		Description																											
A	W	STARTFETCH																																
B	W	STARTPUSH																																

7.7.1.7.16 CRYPTMSTRDMA.STATUS

Address offset: 0x03C

Status

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID		F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
Reset 0x00000000		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																												
A	R	FETCHBUSY																															
B	R	PUSHBUSY																															
C	R	FETCHNOTEMPTY																															
D	R	PUSHWAITINGFIFO																															
E	R	SOFTTRSTBUSY																															
F	R	PUSHNBDATA																															

7.7.1.7.17 CRYPTMSTRHW.INCLIPSHWCFG

Address offset: 0x400

Included IPs Hardware configuration

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M L K J I H G F E D C B A																															
Reset 0x00000771				0 1 1 1 0 1 1 1 0 0 0 1																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	BA411AESINCLUDED			Generic g_IncludeAES value. BA411E–AES IP included if set																														
B	R	BA415HPAESGCMINCLUDED			Generic g_IncludeAESGCM value. BA415–HP-AES-GCM IP included if set																														
C	R	BA416HPAESXTSINCLUDED			Generic g_IncludeAESXTS value. BA416–HP-AES-XTS IP included if set																														
D	R	BA412DESINCLUDED			Generic g_IncludeDES value. BA412–3DES IP included if set																														
E	R	BA413HASHINCLUDED			Generic g_IncludeHASH value. BA413–HASH IP included if set																														
F	R	BA417CHACHAPOLYINCLUDED			Generic g_IncludeChachaPoly value. BA417–ChaChaPoly IP included if set																														
G	R	BA418SHA3INCLUDED			Generic g_IncludeSHA3 value. BA418–SHA3 IP included if set																														
H	R	BA421ZUCINCLUDED			Generic g_IncludeZUC value. BA421–ZUC IP included if set																														
I	R	BA419SM4INCLUDED			Generic g_IncludeSM4 value. BA419–SM4 IP included if set																														
J	R	BA414EPPKEINCLUDED			Generic g_IncludePKE value. BA414EP-PKE IP included if set																														
K	R	BA431NDRNGINCLUDED			Generic g_IncludeNDRNG value. BA431–NDRNG IP included if set																														
L	R	BA420HPCHACHAPOLYINCLUDED			Generic g_IncludeHPChachaPoly value. BA420–HP-ChaChaPoly IP included if set																														
M	R	BA423SNOW3GINCLUDED			Generic g_IncludeSnow3G value. BA423–Snow3G IP included if set																														
N	R	BA422KASUMIINCLUDED			Generic g_IncludeKasumi value. BA422–Kasumi IP included if set																														

7.7.1.7.18 CRYPTMSTRHW.BA411EAESHWCFG1

Address offset: 0x404

Generic g_AesModesPoss value.

Bit number		31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																						
ID		D D D										C B										A A A A A A A A A A A																		
Reset 0x070301FF		0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 1																																						
ID	R/W	Field	Value ID	Value	Description																																			
A	R	BA411EAESHWCFGMODE				Generic g_AesModesPoss value.																																		
						BA411E-AES engine configuration.																																		
B	R	BA411EAESHWCFGCS				Generic g_CS value.																																		
						BA411E-AES engine configuration.																																		
C	R	BA411EAESHWCFGMASKING				Generic g_UseMasking value.																																		
						BA411E-AES engine configuration.																																		
D	R	BA411EAESHWCFGKEYSIZE				Generic g_KeySize value.																																		
						BA411E-AES engine configuration.																																		

7.7.1.7.19 CRYPTMSTRHW.BA411EAESHWCFG2

Address offset: 0x408

Generic g_CtrSize value.

BA411E-AES engine configuration.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																				A A A A A A A A A A A A A A A A															
Reset 0x00000080				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	BA411EAESHWCFG2								Generic g_CtrSize value.																									
								BA411E-AES engine configuration.																											

7.7.1.7.20 CRYPTMSTRHW.BA413HASHHWCFG

Address offset: 0x40C

Generic g_Hash value

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B																								A A A A A A A				A A A A A A A			
Reset 0x0003003F				0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	BA413HASHHWCFGMASK			Generic g_HashMaskFunc value.																														
					BA413-Hash engine configuration.																														
B	R	BA413HASHHWCFGPPADDING			Generic g_HashPadding value.																														
					BA413-Hash engine configuration.																														
C	R	BA413HASHHWCFGHMAC			Generic g_HMAC_enabled value.																														
					BA413-Hash engine configuration.																														
D	R	BA413HASHHWCFGVERIFYDIGEST			Generic g_HashVerifyDigest value.																														
					BA413-Hash engine configuration.																														

7.7.1.7.21 CRYPTMSTRHW.BA418SHA3HWCFG

Address offset: 0x410

Generic g_Sha3CtxtEn value.

BA418-SHA3 configuration.

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID		A																															
Reset 0x00000001		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
ID	R/W	Field	Value ID	Value	Description																												
A	R	BA418SHA3HWCFG			Generic g_Sha3CtxtEn value.																												
					BA418-SHA3 configuration.																												

7.7.1.7.22 CRYPTMSTRHW.BA419SM4HWCFG

Address offset: 0x414

Generic g_SM4ModesPoss value.

BA419-SM4 engine configuration.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				B															A A A A A A A A A A																
Reset 0x000201FF				0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	R	BA419SM4HWCFG						Generic g_sm4ModesPoss value.																											
								BA419-SM4 engine configuration.																											
B	R	USEMASKING						Generic g_sm4UseMasking value.																											
								BA419-SM4 engine configuration.																											

7.7.1.7.23 CRYPTMSTRHW.BA424ARIAHWCFG

Address offset: 0x418

Generic g_aria_modePoss value.

BA424-Aria engine configuration.

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																	A A A A A A A A A A																	
Reset 0x0000017F		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	1	1	1	1	
ID	R/W	Field	Value ID	Value	Description																													
A	R	BA424ARIAHWCFG			Generic g_aria_modePoss value.																													
					BA424-Aria engine configuration.																													

7.7.1.7.24 RNGCONTROL.CONTROL

Address offset: 0x1000

Control register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				Q P P P P O N M L K J I H G F E D C B A																															
Reset 0x00040000				0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE			Enable the NDRNG.																														
B	RW	LFSREN			Select between the NDRNG with asynchronous free running oscillators (when 0) and the Pseudo-Random generator with synchronous oscillators for simulation purpose (when 1).																														
C	RW	TESTEN	NORMAL	0	Noise source (normal mode).																														
			TEST	1	Test data register (test mode).																														
D	RW	CONDBYPASS	NORMAL	0	the conditioning function is used (normal mode).																														
			BYPASS	1	the conditioning function is bypassed (to observe entropy source directly).																														
E	RW	INTENREP			Interrupt enable for Repetition Count Test failure.																														
F	RW	INTENPROP			Interrupt enable for Adaptive Proportion Test failure (1024-sample window).																														
G	RW	INTENFULL			Interrupt enable for FIFO full.																														
H	RW	SOFTRST			Software reset:																														
					This bit is not cleared automatically.																														
			NORMAL	0	Normal mode.																														
			CTEST	1	The continuous test, the conditioning function and the FIFO are reset.																														
I	RW	INTENPRE			Interrupt enable for AIS31 preliminary noise alarm.																														
J	RW	INTENALM			Interrupt enable for AIS31 noise alarm.																														
K	RW	FORCEACTIVEROS			Force oscillators to run when FIFO is full.																														
L	RW	HEALTHTESTBYPASS			Bypass NIST tests such that the results of the start-up and online test do not affect the FSM state.																														
M	RW	AIS31BYPASS			Bypass AIS31 tests such that the results of the start-up and online tests do not affect the FSM state.																														
N	RW	HEALTHTESTSEL	BEFORE	0	Before conditioning.																														
			AFTER	1	After conditioning.																														
O	RW	AIS31TESTSEL	BEFORE	0	Before conditioning.																														
			AFTER	1	After conditioning.																														
P	RW	NB128BITBLOCKS			Number of 128 bit blocks used in AES-CBCMAC post-processing.																														
					This value cannot be zero.																														
Q	RW	FIFOWRITESTARTUP			Enable write of the samples in the FIFO during start-up.																														

7.7.1.7.25 RNGCONTROL.FIFOLEVEL

Address offset: 0x1004

FIFO level register.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																															
A	RW	FIFOLEVEL			Number of 32 bits words of random values available in the FIFO.																															
					Any write to this register clears the FullInt flag in the Status register.																															

7.7.1.7.26 RNGCONTROL.FIFOTHRESHOLD

Address offset: 0x1008

FIFO threshold register.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A A																															
Reset 0x00000003					0 1 1																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	FIFOTHRESHOLD			FIFO level below which the module leaves the idle state to refill the FIFO, expressed in number of 128bit blocks.																															

7.7.1.7.27 RNGCONTROL.FIFODEPTH

Address offset: 0x100C

FIFO depth register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000010				0 1 0 0 0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	FIFODEPTH						Maximum number of 32 bits words that can be stored in the FIFO: 2**g_fifodepth.																											

7.7.1.7.28 RNGCONTROL.KEY[n] (n=0..3)

Address offset: $0x1010 + (n \times 0x4)$

Key register.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	KEY			Key register.																															
					Note: Key0 is MSB and Key3 is LSB.																															

7.7.1.7.29 RNGCONTROL.TESTDATA

Address offset: 0x1020

Test data register.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	TESTDATA			Test data register.																															

7.7.1.7.30 RNGCONTROL.REPEATTHRESHOLD

Address offset: 0x1024

Repetition Test Count Cut-Off value.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																		A	A	A	A	A	A
Reset 0x00000029					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1		
ID	R/W	Field	Value ID	Value	Description																																		
A	RW	REPEATTHRESHOLD			Repetition Test Count Cut-Off value.																																		

7.7.1.7.31 RNGCONTROL.PROPTHRESHOLD

Address offset: 0x1028

Adaptive Proportion Test (1024-sample window) Cut-Off value.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

7.7.1.7.32 RNGCONTROL.STATUS

Address offset: 0x1030

Status register.

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0															
ID																																			I	H	G	F	E		D	C	B	B	B	A		
Reset 0x00000000		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																											
A	R	TESTDATABUSY			High when data written to TestData register is being processed.																																											
B	R	STATE	RESET	0	Reset																																											
			STARTUP	1	Startup																																											
			IDLERON	2	Idle (Rings On)																																											
			IDLEROFF	3	Idle (Rings Off)																																											
			FILLFIFO	4	Fill FIFO																																											
			ERROR	5	Error																																											
C	RW	REPFAIL			NIST-800-90B repetition Count Test interrupt status.																																											
D	RW	PROPFALL			NIST-800-90B adaptive Proportion Test (1024-sample window) interrupt status.																																											
E	RW	FULLINT			FIFO full status.																																											
F	RW	PREINT			AIS31 preliminary noise alarm interrupt status.																																											
G	RW	ALMINT			AIS31 noise alarm interrupt status.																																											
H	R	STARTUPFAIL			Start-up test failure.																																											
I	RW	FIFOACCFALL			Set when a FIFO data read is performed while the NDRNG is disabled AND has its FIFO empty (FIFOLevel = 0).																																											

7.7.1.7.33 RNGCONTROL.INITWAITVAL

Address offset: 0x1034

Initial wait counter value.

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																			A	A	A	A	A	A	A	A	A	A	A	A	A	A	
Reset 0x0000FFFF		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
ID	R/W	Field	Value ID	Value	Description																												
A	RW	INITWAITVAL			Number of clock cycles to wait before sampling data from the noise source.																												

7.7.1.7.34 RNGCONTROL.DISABLEOSC[n] (n=0..1)

Address offset: 0x1038 + (n × 0x4)

Disable oscillator rings #n*32 to #((n+1)*32)-1.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A		
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	DISABLEOSC			Disable oscillator rings #n*32 to #((n+1)*32)-1.																																	

7.7.1.7.35 RNGCONTROL.SWOFFTMRVAL

Address offset: 0x1040

Switch off timer value.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x0000FFFF					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	SWOFFTMRVAL			Number of clk cycles to wait before stopping the rings after the FIFO is full.																																	

7.7.1.7.36 RNGCONTROL.CLKDIV

Address offset: 0x1044

Sample clock divider.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																													A	A	A	A	A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value	Description																																
A	RW	CLKDIV			Sample clock divider.																																

The frequency at which the outputs of the rings are sampled is given by:
 $F_s = F_{pclk} / (ClkDiv + 1)$.

7.7.1.7.37 RNGCONTROL.AIS31CONF0

Address offset: 0x1048

AIS31 configuration register 0.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B B B B B B B B B B B B A																															

7.7.1.7.38 RNGCONTROL.AIS31CONF1

Address offset: 0x104C

AIS31 configuration register 1.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B B B B B B B B B B B B A																															

7.7.1.7.39 RNGCONTROL.AIS31CONF2

Address offset: 0x1050

AIS31 configuration register 2.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B B B B B B B B B B B B A																															

7.7.1.7.40 RNGCONTROL.AIS31STATUS

Address offset: 0x1054

AIS31 status register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																
ID				C B A																																
Reset 0x00000000				0 0																																
ID	R/W	Field	Value ID	Value				Description																												
A	RW	NUMPRELIMALARMS							Number of preliminary noise alarms since counter was last cleared.																											
B	RW	PRELIMNOISEALARMRNG							Last preliminary noise alarm occurred due to history value out of range.																											
C	RW	PRELIMNOISEALARMREP							Last preliminary noise alarm occurred due to consecutive high X**2.																											

7.7.1.7.41 RNGCONTROL.HWCONFIG

Address offset: 0x1058

Hardware configuration register.

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																												C	B	A	A	A	A	A	A
Reset 0x00000337		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1	0	1	1	
ID	R/W	Field	Value ID	Value	Description																														
A	R	NUMBOFRINGS			Generic g_NumRings value.																														
B	R	AIS31			Generic g_AIS31 value.																														
C	R	AIS31FULL			Generic g_AIS31Full value.																														

7.7.1.7.42 RNGCONTROL.FIFO[n] (n=0..15)

Address offset: 0x1080 + (n × 0x4)

FIFO data

The FIFO contains the RNG output data.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

7.7.1.7.43 PK.POINTERS

Address offset: 0x2000

Pointers register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D D D D								C C C C								B B B B								A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW	OPPTRA										When executing primitive arithmetic operations, this pointer defines where operand A is located in memory (location 0x0 to 0xF).																							
B	RW	OPPTRB										When executing primitive arithmetic operations, this pointer defines where operand B is located in memory (location 0x0 to 0xF).																							
C	RW	OPPTRC										When executing primitive arithmetic operations, this pointer defines the location (0x0 to 0xF) where the result will be stored in memory.																							
D	RW	OPPTRN										When executing primitive arithmetic operations, this pointer defines the location where the modulus is located in memory (location 0x0 to 0xF).																							

7.7.1.7.44 PK.COMMAND

Address offset: 0x2004

Command register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				L	K	J	I		H	G	F		E	E	E	D		C	C	C	C	C	C	C	C	C	C	B	A	A	A	A	A	A	A	
Reset 0x0000000F				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																												
A	RW	OPEADDR						This field defines the operation to be performed.																												
				See documentation for more details.																																
B	RW	FIELDF		0: Field is GF(p) 1: Field is GF(2**m)																																

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				L K J I H G F E E E D C C C C C C C C C C B A A A A A A A																															
Reset 0x0000000F				0 1 1 1 1																															
ID	R/W	Field	Value ID	Value	Description																														
C	RW	OPBYTESM1			<p>This field defines the size (= number of bytes minus one) of the operands for the current operation.</p> <p>Possible values are limited by the maximum supported operand size.</p> <p>Examples: - 0x014 - ECC on curve K-163 - 0x01F - ECC on curve P-256 - 0x02F - ECC on curve P-384 - 0x033 - ECC on curve K-409 - 0x041 - ECC on curve P-521 - 0x07F - 1024-bit RSA - 0x09F - 1280-bit RSA - 0x1FF - 4096-bit RSA - 0x3FF - 8192-bit RSA</p>																														
D	RW	RANDMOD			Enable randomization of modulus (counter-measure).																														
E	RW	SELCURVE			<p>Enable accelerator for specific curve modulus:</p> <p>This field has no effect when the optional acceleration hardware is not included.</p> <table><tr><td>NOACCEL</td><td>0x0</td><td>No acceleration (default)</td></tr><tr><td>P256</td><td>0x1</td><td>P256</td></tr><tr><td>P384</td><td>0x2</td><td>P384</td></tr><tr><td>P521</td><td>0x3</td><td>P521</td></tr><tr><td>P192</td><td>0x4</td><td>P192</td></tr><tr><td>CURVE25519</td><td>0x5</td><td>Curve25519</td></tr><tr><td>ED25519</td><td>0x6</td><td>Ed25519.</td></tr></table>	NOACCEL	0x0	No acceleration (default)	P256	0x1	P256	P384	0x2	P384	P521	0x3	P521	P192	0x4	P192	CURVE25519	0x5	Curve25519	ED25519	0x6	Ed25519.									
NOACCEL	0x0	No acceleration (default)																																	
P256	0x1	P256																																	
P384	0x2	P384																																	
P521	0x3	P521																																	
P192	0x4	P192																																	
CURVE25519	0x5	Curve25519																																	
ED25519	0x6	Ed25519.																																	
F	RW	RANDKE			Enable randomization of exponent/scalar (counter-measure).																														
G	RW	RANDPROJ			Enable randomization of projective coordinates (counter-measure).																														
H	RW	EDWARDS			Enable Edwards curve.																														
I	RW	SWAPBYTES			<p>Swap the bytes on AHB interface:</p> <p>This bit must be programmed before writing/reading any data in data memory.</p> <table><tr><td>NATIVE</td><td>0</td><td>Native format (little endian).</td></tr><tr><td>SWAPPED</td><td>1</td><td>Byte swapped (big endian).</td></tr></table>	NATIVE	0	Native format (little endian).	SWAPPED	1	Byte swapped (big endian).																								
NATIVE	0	Native format (little endian).																																	
SWAPPED	1	Byte swapped (big endian).																																	
J	RW	FLAGA			Flag A.																														
K	RW	FLAGB			Flag B.																														
L	RW	CALCR2			<p>This bit indicates if the IP has to calculate $R^2 \bmod N$ for the next operation.</p> <p>This bit must be set to 1 when a new prime number has been programmed.</p> <p>This bit is used for primitive operations and ignored for the other operations.</p> <table><tr><td>NRECALCULATE</td><td>0</td><td>don't recalculate $R^2 \bmod N$</td></tr><tr><td>RECALCULATE</td><td>1</td><td>re-calculate $R^2 \bmod N$</td></tr></table>	NRECALCULATE	0	don't recalculate $R^2 \bmod N$	RECALCULATE	1	re-calculate $R^2 \bmod N$																								
NRECALCULATE	0	don't recalculate $R^2 \bmod N$																																	
RECALCULATE	1	re-calculate $R^2 \bmod N$																																	

7.7.1.7.45 PK.CONTROL

Address offset: 0x2008

Command register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	START						Writing a 1 starts the processing.																											
B	W	CLEARIRQ						Writing a 1 clears the IRQ output.																											

7.7.1.7.46 PK.STATUS

Address offset: 0x200C

Status register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D D D D D								C B A A A A A A A A A A A A A A A A																							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	ERRORFLAGS			These bits indicate an error condition.																														
					They are updated at the end of the operation. They are cleared when starting a new operation.																														
B	R	PKBUSY			This bit reflects the BUSY output value.																														
					It is set when the operation starts and it is cleared when the operation is finished.																														
C	R	INTRPTSTATUS			This bit reflects the IRQ output value.																														
					It is set when the operation is finished. It is cleared when the CPU writes the bit 1 of Control Register.																														
D	R	FAILPTR			These bits indicate which data location generated the error flag.																														
					They are not available for all error flags.																														

7.7.1.7.47 PK.TIMER

Address offset: 0x2014

Timer register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	RW	TIMER																		Number of core clock cycles.															

7.7.1.7.48 PK.HWCONFIG

Address offset: 0x2018

Hardware configuration register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M L			K J I H G F				E D C B B B				A A A A				A A A A				A A A A				A A A A								
Reset 0x01F72200				0 0 0 0 0 0 0 0 1 1 1 1 0 1 1 1 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	R	MAXOPSIZE				Maximum operand size (number of bytes).																													
B	R	NBMULT				Number of multipliers:																													
			MULT1	0	1 multiplier																														
			MULT4	1	4 multipliers																														
			MULT16	2	16 multipliers																														
			MULT64	4	64 multipliers																														
			MULT256	8	256 multipliers																														
C	R	PRIMEFIELD				Support prime field.																													
D	R	BINARYFIELD				Support binary field.																													
E	R	ECC				Support error correction.																													

Operand size register.

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																		
ID	A A																																		
Reset 0x00001000	0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																																		
ID	R/W	Field	Value ID	Value	Description																														
A	RW	OPSIZE			Operand size (number of bytes): This register is used when the memory is accessed via AHB Master																														
			OPSIZE256	0x0100	256 bytes.																														
			OPSIZE521	0x0209	521 bytes.																														
			OPSIZE2048	0x0800	2048 bytes.																														
			OPSIZE3072	0x0C00	3072 bytes.																														
			OPSIZE4096	0x1000	4096 bytes.																														

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0													
ID																	B	B	B	B	B	B	B	B	B											A	A	A	A	A	A	A	A	A	A
Reset 0x03FF03FF	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1												
ID	R/W	Field	Value ID		Value				Description																																				
A	RW	BITERROR1					Bit position of first error																																						
B	RW	BITERROR2					Bit position of second error																																						

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	R	RAMCORRECTION																		This bit indicates that a 1-bit error has been detected and corrected on RAM interface															
B	R	RAMFAILURE																		This bit indicates that an uncorrectable error has been detected on the data RAM interface															

7.7.1.7.52 IKG.START

Address offset: 0x3000

Start register.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	START			Start the Isolated Key Generation.																															

7.7.1.7.53 IKG.STATUS

Address offset: 0x3004

Status register.

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			G F E D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	R	SEEDERROR			Seed Error during Isolated Key Generation. When the IKG module is in error state, a reset is required to restart the module.																													
B	R	ENTROPYERROR			Entropy Error during Isolated Key Generation. When the IKG module is in error state, a reset is required to restart the module.																													
C	R	OKAY			Isolated Key Generation is okay.																													
D	R	CTRDRBGBUSY			CTR_DRBG health test is busy (only when g_hw_health_test = true).																													
E	R	CATASTROPHICERROR			Catastrophic error during CTR_DRBG health test (only when g_hw_health_test = true). When the IKG module is in error state, a reset is required to restart the module.																													
F	R	SYMKEYSTORED			Symmetric Keys are stored.																													
G	R	PRIVKEYSTORED			Private Keys are stored.																													

7.7.1.7.54 IKG.INITDATA

Address offset: 0x3008

InitData register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	INITDATA						Writing a 1 initialise Nonce and Personalisation_String registers counters, i.e. start writing from the 32 LSB.																											

7.7.1.7.55 IKG.NONCE

Address offset: 0x300C

Nonce register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	NONCE						Nonce (write/read value 32-bit by 32-bit).																											

7.7.1.7.56 IKG.PERSONALISATIONSTRING

Address offset: 0x3010

Personalisation String register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

7.7.1.7.57 IKG.RESEEDINTERVALLSB

Address offset: 0x3014

Reseed Interval LSB register.

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID										A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x80000000										1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																			
A	RW	RESEEDINTERVALLSB						Reseed Interval LSB.																																			

7.7.1.7.58 IKG.RESEEDINTERVALMSB

Address offset: 0x3018

Reseed Interval MSB register.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value	Description																																		
A	RW	RESEEDINTERVALMSB			Reseed Interval MSB.																																		

7.7.1.7.59 IKG.PKECONTROL

Address offset: 0x301C

PKE Control register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	PKESTART						Start the PKE operation or trigger for Secure mode exit.																											
B	W	CLEARIRQ						Clear the IRQ output.																											

7.7.1.7.60 IKG.PKECOMMAND

Address offset: 0x3020

PKE Command register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C C B B B B A																															
Reset 0x00000000				0 0																															
D	R/W	Field	Value ID	Value	Description																														
A	RW	SECUREMODE			Secure mode. It is activated as soon as it is set to 1.It is deactivated when it is set to 0 and PKE_Start is set to 1.																														
			DEACTIVATED	0																															
			ACTIVATED	1																															
B	RW	SELECTEDKEY			Select Generated Private Key for PKE operation. This Key Index should be between 0 and g_nb_priv_keys-1.																														
C	RW	OPSEL			Select PKE operation with Isolated Key Note: Value 3 is reserved.																														
			PUBKEY	0	Public Key Generation																														
			ECDSA	1	ECDSA Signature																														
			PTMUL	2	Point Multiplication																														

7.7.1.7.61 IKG.PKESTATUS

Address offset: 0x3024

PKE Status register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	ERROR			Error because either Private Keys are not stored or the operation is not defined.																														
B	R	STARTERROR			Error because a new operation is started while the previous one is still busy.																														
C	R	IKGPKBUSY			Busy, set when the operation starts and cleared when the operation is finished.																														
D	R	IRQSTATUS			IRQ, set when the operation is finished and cleared when the CPU writes the bit 1 of PKE_Control Register or a new operation is started.																														
E	R	ERASEBUSY			The PKE Data RAM is being erased.																														

7.7.1.7.62 IKG.SOFRST

Address offset: 0x3028

SoftRst register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	SOFRST						Software reset:																											
								This bit is not cleared automatically.																											
			NORMAL	0				Normal mode.																											
			KEY	1				The Isolated Key Generation logic and the keys are reset.																											

7.7.1.7.63 IKG.HWCONFIG

Address offset: 0x302C

HwConfig register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				K	K	K	K	J	J	J	J	I	I	I	I	H	H	H	H	G	G	G	F	E	E	D	C	B	B	B	B	A	A	A	A
Reset 0xCC4C8312				1	1	0	0	1	1	0	0	0	1	0	0	1	1	0	0	1	0	0	0	0	0	1	1	0	0	0	1	0	0	1	0
ID	R/W	Field	Value ID	Value	Description																														
A	R	NBSYMKEYS			Number of Symmetric Keys generated.																														
B	R	NBPRIVKEYS			Number of Private Keys generated.																														
C	R	IKGCM			Countermeasures for IKG operations are implemented when 1.																														
D	R	HWHEALTHTEST			CTR_DRBG health test is implemented when 1.																														
E	R	CURVE			ECC curve for IKG (input).																														
					Note: value 3 is reserved																														
			P256	0	P256.																														
			P384	1	P384.																														
			P521	2	P521.																														
F	R	DF			Derivation function is implemented in the CTR_DRBG when 1.																														
G	R	KEYSIZE			AES Key Size support for the AES Core embedded in the CTR_DRBG.																														
					[0]: supports AES128 when 1 [1]: supports AES192 when 1 [2]: supports AES256 when 1																														
			AES128	1	supports AES128																														
			AES192	2	supports AES192																														
			AES256	4	supports AES256																														
H	R	ENTROPYINPUTLENGTH			Value of g_entropy_input_length/32.																														
I	R	NONCELENGTH			Value of g_nonce_length/32.																														
J	R	PERSONALIZATIONSTRINGLENGTH			Value of g_personalization_string_length/32.																														
K	R	ADDITIONALINPUTLENGTH			Value of g_additional_input_length/32.																														

7.7.2 GLITCHDET — Voltage glitch detectors

The system has voltage glitch detectors.

The voltage glitch detectors are automatically enabled after reset. To save power, the glitch detectors must be disabled when not in use.

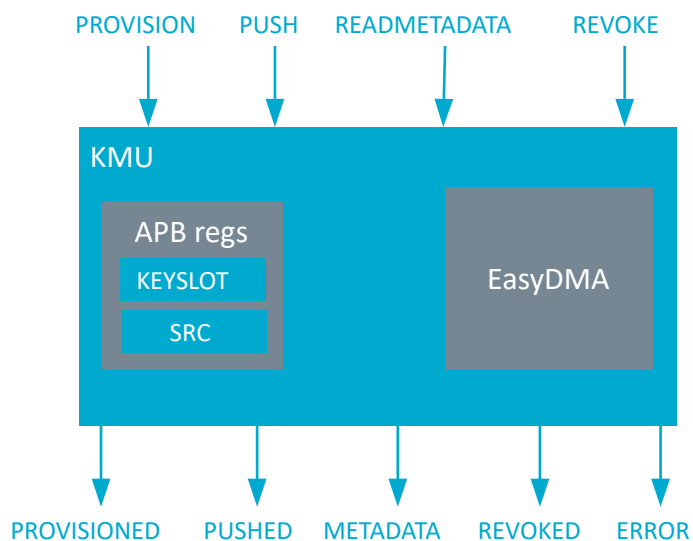


Figure 32: Block diagram

7.7.3.1 Key slot

A key slot stores secure assets and up to 32 bits of additional metadata. Assets greater than 128 bits must be divided and distributed over multiple key slot instances.

The following table summarizes what can be stored in a key slot.

Field	Size [bits]	Description
METADATA	32	A text field that can be used for any purpose. This field can be read by secure code using the READMETADATA task.
DEST	32	The destination address used for the push, and used by the PUSH operation.
VALUE	128	The secure asset. This field cannot be read, only pushed to its destination address using the PUSH task.
RPOLICY	2	The revocation policy for the key slot. See Provisioning on page 168 for a detailed definition of this field.

Table 27: Key slot contents

7.7.3.1.1 Key slot states

KMU maintains the key slot state.

The following figure shows the key slot states and how they transition through the device life cycle.

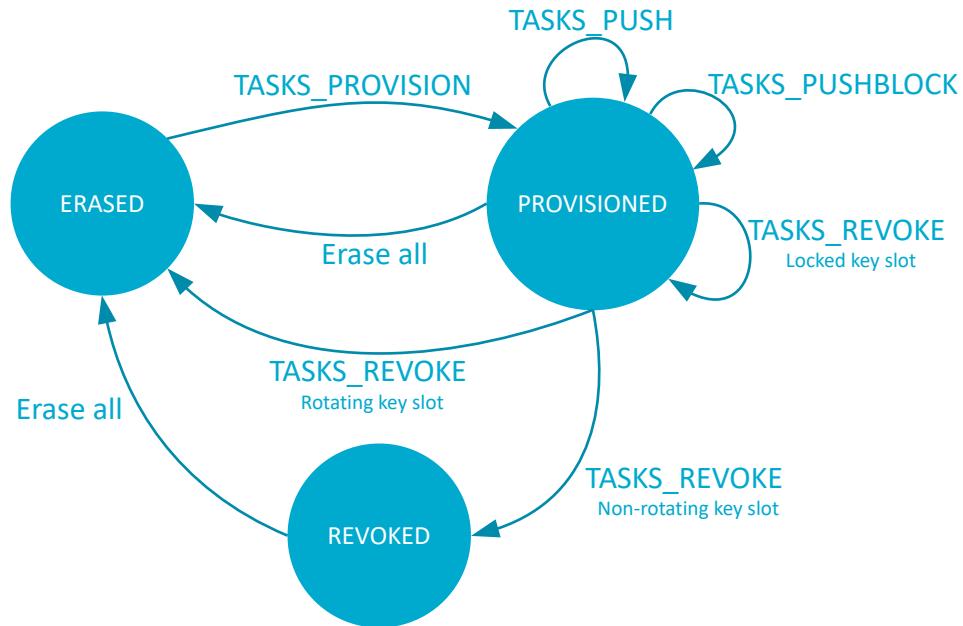


Figure 33: Key slot states

7.7.3.2 Operations

KMU has operations to store, use, and remove assets.

Operation	Description
Provision	Store assets in SICR
Push	Retrieve assets from SICR and push to write-only registers or memory for use
Read metadata	Read key slot metadata from SICR
Revoke	Remove an asset from SICR
Block	Block a keyslot from being pushed, provisioned, or revoked until next reset
Push block	Block a key slot by preventing a push until next reset

Table 28: KMU operations

KMU allows a single operation to run at a time. Once a TASK is triggered to start an operation, KMU ignores any subsequent TASK requests until the initial operation is complete.

7.7.3.2.1 Provisioning

Provisioning is the storage of an asset in SICR. During provisioning, KMU copies data and permission policy from RAM to SICR.

Provisioning a key slot is possible when the key slot is in the ERASED state.

To provision an asset, perform the following steps:

1. Populate the SRC data struct as an array in RAM.
2. Write the [SRC](#) register to the address of the SRC data in RAM. See the following table for SRC data details.
3. Configure the key slot ID in the [KEY SLOT](#) register.

4. Using the RRAM controller, enable unbuffered RRAM write using register [RRAMC.CONFIG](#). For more details on RRAMC, see [RRAMC — Resistive random access memory controller](#) on page 53.
5. Trigger the [PROVISION](#) task. KMU writes data to SICR.
If copying of data was successful, KMU generates the [PROVISIONED](#) event, otherwise KMU generates the [ERROR](#) event.
6. Disable the RRAM write operation. For details, see [RRAMC — Resistive random access memory controller](#) on page 53.

If a power failure occurs during provisioning, KMU will not write key slot data to RRAM and the key slot is not provisioned. For more details on how to detect power failures, see [Power-fail comparator](#) on page 75.

The following lists the SRC data used for provisioning.

Field	Byte offset	Size [bytes]	Description
METADATA	24	4	32 bits of any cleartext metadata that belongs with the key slot. This metadata can later be read using the READMETADATA task (for details, see Read metadata on page 170).
DEST	20	4	32-bit destination address. Note that DEST cannot point to SICR. DEST must be on a 128-bit boundary.
RPOLICY	16	4	Revocation policy (same definition as the key slot RPOLICY field). Only two LSB's of the field are used, unused bits shall be set to zero. <ul style="list-style-type: none"> '11' REVOKED: When TASKS_REVOKE is triggered, key slot ends up in the Revoked state "forever" (until Erase all). '01' ROTATING: Key Slot can be reused, and when TASKS_REVOKE is triggered, the key slot ends up in the Erased state and can be reused. '10' LOCKED: Key Slot can not be revoked (until Erase all). When TASKS_REVOKE is triggered, EVENTS_ERROR is generated. '00' RESERVED: Reserved for future use. The revocation policy affects how the key slot transitions through its states, see Key slot states on page 167.
VALUE[3:0]	0	16	Asset contents/value. This value can later be used by the PUSH task (for details, see Push on page 169).

Table 29: SRC data

7.7.3.2.2 Push

Retrieving an asset from SICR is called a push. During a push, KMU copies data from SICR to the destination address that was determined during provisioning.

A key slot can be pushed only when it is in the PROVISIONED state and if it is not push-blocked. For more details on push-block, see [Push block](#) on page 170.

To push a key slot, perform the following steps:

1. Configure the key slot ID in the [KEYSLOT](#) register.
2. Trigger the [PUSH](#) task.

KMU copies data from SICR.

If the push is successful, KMU generates the **PUSHED** event. If the keyslot is in the REVOKED state, KMU generates the **REVOKED** event. If unsuccessful, KMU generates the **ERROR** event.

Note: Some push operations generate the **PUSHED** event when data is not successfully copied to the destination. For example, when the CRACEN SEEDLOCK register is set to enabled, write operations to the CRACEN SEED register are ignored.

7.7.3.2.3 Read metadata

Each key slot has a 32-bit metadata field that can be read.

The metadata field is the same 32-bit field that is provisioned, see [Provisioning](#) on page 168.

When reading the metadata, KMU copies the key slot metadata from SICR to the **METADATA** register.

Key slot metadata can be read when the key slot is in the PROVISIONED state.

Perform the following steps to read key slot metadata.

1. Configure the key slot ID in the **KEYSLOT** register.
2. Trigger the **READMETADATA** task.

If the key slot is revoked, KMU generates the **REVOKED** event and ends the operation.

If the key slot has not been provisioned, KMU generates the **ERROR** event and ends the operation.

KMU copies data from SICR into the **METADATA** register. If copying of data was successful, KMU generates the **METADATAREAD** event. If unsuccessful, KMU generates the **ERROR** event.

7.7.3.2.4 Revoke

A key slot that is revoked it can no longer be pushed.

A key slot can be revoked when it is in the PROVISIONED state or when its revocation policy is not LOCKED.

To revoke a key slot, perform the following steps:

1. Configure the key slot ID in the **KEYSLOT** register.
2. Enable RRAM write operation in Normal write mode. For details, see [RRAMC — Resistive random access memory controller](#) on page 53.
3. Trigger the **REVOKE** task.
KMU erases the asset from SICR. If revoking the key slot is successful, KMU generates the **REVOKED** event. If unsuccessful, or the key slot is already in the REVOKED state, KMU generates the **ERROR** event.
4. Disable RRAM write operation. For details, see [RRAMC — Resistive random access memory controller](#) on page 53.

Rotating key slots are available after a successful revocation. Non-rotating key slots remain in a REVOKED state and can not be used again until SICR is erased. SICR can only be erased using the Erase All functions of [CTRL-AP - Control access port](#) on page 755 and [RRAMC — Resistive random access memory controller](#) on page 53.

7.7.3.2.5 Push block

Push block prevents a key slot from being pushed until the next device reset.

A key slot must be in the PROVISIONED state for a push block to take effect.

To block a key slot from the push operation, perform the following steps:

1. Configure the key slot ID in the **KEYSLOT** register.
2. Trigger the **PUSHBLOCK** task.

When the push block has been applied, KMU generates the **PUSHBLOCKED** event.

7.7.3.3 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
KMU	GLOBAL	0x50045000	HF	S	NSA	No	Key management unit

Configuration

Instance	Domain	Configuration
KMU	GLOBAL	Number of keyslots is 250 Number of bits per keyslot is 128

Register overview

Register	Offset	TZ	Description
TASKS_PROVISION	0x0000		Provision key slot
TASKS_PUSH	0x0004		Push key slot
TASKS_REVOKE	0x0008		Revoke key slot
TASKS_READMETADATA	0x000C		Read key slot metadata into METADATA register
TASKS_PUSHBLOCK	0x0010		Block only the PUSH operation of a key slot, preventing the key slot from being PUSHED until next reset. The task is kept for backwards compatibility.
EVENTS_PROVISIONED	0x100		Key slot successfully provisioned
EVENTS_PUSHED	0x104		Key slot successfully pushed
EVENTS_REVOKED	0x108		Key slot has been revoked and can no longer be used
EVENTS_ERROR	0x10C		Error generated during PROVISION, PUSH, READMETADATA or REVOKE operations. Triggering the PROVISION, PUSH and REVOKE tasks on a BLOCKED keyslot will also generate this event.
EVENTS_METADATAREAD	0x110		Key slot metadata has been read into METADATA register
EVENTS_PUSHBLOCKED	0x114		The PUSHBLOCK operation was successful. The event is kept for backwards compatibility.
STATUS	0x400		KMU status register
KEYSLOT	0x500		Select key slot to operate on
SRC	0x504		Source address for provisioning
METADATA	0x508		Key slot metadata as read by TASKS_READMETADATA.

7.7.3.3.1 TASKS_PROVISION

Address offset: 0x0000

Provision key slot

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A																															
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID		Value				Description																											
A	W	TASKS_PROVISION							Provision key slot																											
			Trigger		1				Trigger task																											

7.7.3.3.2 TASKS_PUSH

Address offset: 0x0004

Push key slot

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																
ID																																				A
Reset 0x00000000				0 0																																0
ID	R/W	Field	Value ID	Value				Description																												
A	W	TASKS_PUSH						Push key slot																												
			Trigger	1				Trigger task																												

7.7.3.3.3 TASKS_REVOKE

Address offset: 0x0008

Revoke key slot

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																
ID																																				A
Reset 0x00000000				0 0																																0
ID	R/W	Field	Value ID	Value				Description																												
A	W	TASKS_REVOKE						Revoke key slot																												
			Trigger	1				Trigger task																												

7.7.3.3.4 TASKS_READMETADATA

Address offset: 0x000C

Read key slot metadata into METADATA register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																
ID																																				A
Reset 0x00000000				0 0																																0
ID	R/W	Field	Value ID	Value				Description																												
A	W	TASKS_READMETADATA						Read key slot metadata into METADATA register																												
			Trigger	1				Trigger task																												

7.7.3.3.5 TASKS_PUSHBLOCK

Address offset: 0x0010

Block only the PUSH operation of a key slot, preventing the key slot from being PUSHED until next reset. The task is kept for backwards compatibility.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	TASKS_PUSHBLOCK			Block only the PUSH operation of a key slot, preventing the key slot from being PUSHED until next reset. The task is kept for backwards compatibility.																															
			Trigger	1	Trigger task																															

7.7.3.3.6 EVENTS_PROVISIONED

Address offset: 0x100

Key slot successfully provisioned

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_PROVISIONED							Key slot successfully provisioned																										
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

7.7.3.3.7 EVENTS_PUSHED

Address offset: 0x104

Key slot successfully pushed

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				A																																		
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value				Description																														
A	RW	EVENTS_PUSHED						Key slot successfully pushed																														
			NotGenerated	0				Event not generated																														
			Generated	1				Event generated																														

7.7.3.3.8 EVENTS_REVOKED

Address offset: 0x108

Key slot has been revoked and can no longer be used

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_REVOKED						Key slot has been revoked and can no longer be used																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

7.7.3.3.9 EVENTS_ERROR

Address offset: 0x10C

Error generated during PROVISION, PUSH, READMETADATA or REVOKE operations. Triggering the PROVISION, PUSH and REVOKE tasks on a BLOCKED keyslot will also generate this event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_ERROR						Error generated during PROVISION, PUSH, READMETADATA or REVOKE operations. Triggering the PROVISION, PUSH and REVOKE tasks on a BLOCKED keyslot will also generate this event.																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

7.7.3.3.10 EVENTS_METADATAREAD

Address offset: 0x110

Key slot metadata has been read into METADATA register

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	EVENTS_METADATAREAD			Key slot metadata has been read into METADATA register																																	
			NotGenerated	0	Event not generated																																	
			Generated	1	Event generated																																	

7.7.3.3.11 EVENTS_PUSHBLOCKED

Address offset: 0x114

The PUSHBLOCK operation was successful. The event is kept for backwards compatibility.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	EVENTS_PUSHBLOCKED			The PUSHBLOCK operation was successful. The event is kept for backwards compatibility.																																	
			NotGenerated	0	Event not generated																																	
			Generated	1	Event generated																																	

7.7.3.3.12 STATUS

Address offset: 0x400

KMU status register

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	R	STATUS			KMU status																																	
			Ready	0	KMU is ready for new operation																																	
			Busy	1	KMU is busy, an operation is in progress																																	

7.7.3.3.13 KEYSLOT

Address offset: 0x500

Select key slot to operate on

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ID		0..249				Select key slot ID to provision, push, read METADATA, revoke or block when the corresponding task is triggered.																											

7.7.3.3.14 SRC

Address offset: 0x504

Source address for provisioning

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A			
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value			Description																															
A	RW	SRC					Source address for TASKS_PROVISION.																															

7.7.3.3.15 METADATA

Address offset: 0x508

Key slot metadata as read by TASKS_READMETADATA.

When EVENTS_METADATA has been generated, this register holds the key slot metadata.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value			Description																												
A	RW	METADATA					Read metadata.																												

7.7.4 MPC — Memory Privilege Controller

The MPC peripheral is an address decoder with built-in security functions.

MPC enforces security for system memory access. It is used to divide the address space into smaller regions and assign permissions to these regions.

The main features of MPC are the following:

- Address decoding
- Configurable access permissions
- Error reporting

7.7.4.1 Override configuration

The MPC overrides are used to divide the address space into smaller regions and assign permissions to these regions.

When the device is reset, the memory in RAM and RRAM is secure. Only secure CPUs and peripherals can read, write, or execute from secure memory. To configure permission settings in a memory region, perform the following steps.

1. Define the memory region by configuring [STARTADDR](#) and [ENDADDR](#). The values must be multiples of the override region granularity, which is .
2. Use [PERMMASK](#) to define which of the access permissions in [PERM](#) to apply.
3. Enable and lock the override using the [CONFIG](#) register.

To prevent unintended reconfiguration of MPC, all overrides should be safeguarded by using the LOCK bit in the [CONFIG](#) register.

Note: For overlapping regions, MPC will perform a logical OR between the permission bits and the resulting overlap region will be more permissive. It is not possible to retract permissions using MPC. The logical OR applies to both PERMMASK and PERM registers.

7.7.4.2 MPC error reporting

MPC reports an error when an access violation is detected.

MPC generates an `EVENTS_MEMACCERR` event when an erroneous transaction is detected. The following errors can be detected.

- The address cannot be decoded or the bus Manager does not have permissions to access the Subordinate. When this happens, the `MEMACCERR.ADDRESS` will capture the failing address issued by the bus Manager port and `MEMACCERR.INFO` will capture additional access information for the attempted transaction.
- If a transaction is routed to a Subordinate, but the Subordinate responds with an error.

The `MEMACCERR` registers will not be updated when `EVENTS_MEMACCERR` is set.

7.7.4.3 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
MPC00	GLOBAL	0x50041000	HF	S	NA	No	Memory privilege controller MPC00

Configuration

Instance	Domain	Configuration
MPC00	GLOBAL	<p>The override region granularity is 4096 bytes</p> <p>Supports 15 Manager ports</p> <p>Supports 8 Subordinates</p> <p>Supports 8 REGIONs</p>

Register overview

Register	Offset	TZ	Description
<code>EVENTS_MEMACCERR</code>	0x100		Memory Access Error event
<code>INTEN</code>	0x300		Enable or disable interrupt
<code>INTENSET</code>	0x304		Enable interrupt
<code>INTENCLR</code>	0x308		Disable interrupt
<code>MEMACCERR.ADDRESS</code>	0x400		Target Address of Memory Access Error. Register content won't be changed as long as <code>MEMACCERR</code> event is active.
<code>MEMACCERR.INFO</code>	0x404		Access information for the transaction that triggered a memory access error. Register content won't be changed as long as <code>MEMACCERR</code> event is active.
<code>OVERRIDE[n].CONFIG</code>	0x800		Override region n Configuration register
<code>OVERRIDE[n].STARTADDR</code>	0x804		Override region n Start Address
<code>OVERRIDE[n].ENDADDR</code>	0x808		Override region n End Address
<code>OVERRIDE[n].PERM</code>	0x810		Permission settings for override region n
<code>OVERRIDE[n].PERMMASK</code>	0x814		Masks permission setting fields from register <code>OVERRIDE.PERM</code>

7.7.4.3.1 EVENTS_MEMACCERR

Address offset: 0x100

Memory Access Error event

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_MEMACCERR						Memory Access Error event																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

7.7.4.3.2 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																				A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	MEMACCERR				Enable or disable interrupt for event MEMACCERR																															
			Disabled	0		Disable																															
			Enabled	1		Enable																															

7.7.4.3.3 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	MEMACCERR						Write '1' to enable interrupt for event MEMACCERR																											
			Set	1				Enable																											
			Disabled	0				Read: Disabled																											
			Enabled	1				Read: Enabled																											

7.7.4.3.4 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	MEMACCERR			Write '1' to disable interrupt for event MEMACCERR																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

7.7.4.3.5 MEMACCERR

Memory Access Error status registers

7.7.4.3.5.1 MEMACCERR.ADDRESS

Address offset: 0x400

Target Address of Memory Access Error. Register content won't be changed as long as MEMACCERR event is active.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value												Description																			
A	R	ADDRESS														Target address for erroneous access																			

7.7.4.3.5.2 MEMACCERR.INFO

Address offset: 0x404

Access information for the transaction that triggered a memory access error. Register content won't be changed as long as MEMACCERR event is active.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				E D C B A																																
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																															
A	R	READ			Read bit of bus access																															
			Set	1	Read access bit was set																															
			NotSet	0	Read access bit was not set																															
B	R	WRITE			Write bit of bus access																															
			Set	1	Write access bit was set																															
			NotSet	0	Write access bit was not set																															
C	R	EXECUTE			Execute bit of bus access																															
			Set	1	Execute access bit was set																															
			NotSet	0	Execute access bit was not set																															
D	R	SECURE			Secure bit of bus access																															
			Set	1	Secure access bit was set																															
			NotSet	0	Secure access bit was not set																															
E	R	ERRORSOURCE			Source of memory access error																															
			MPC	1	Error was triggered by MPC module																															
			Slave	0	Error was triggered by a Subordinate																															

7.7.4.3.6 OVERRIDE[n] (n=0..4)

Special privilege tables

7.7.4.3.6.1 OVERRIDE[n].CONFIG (n=0..4)

Address offset: 0x800 + (n × 0x20)

Override region n Configuration register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
B	RW	WRITE	NotAllowed	0	Read access to override region n is not allowed																														
			Allowed	1	Read access to override region n is allowed																														
					Write access																														
			NotAllowed	0	Write access to override region n is not allowed																														
C	RW	EXECUTE	Allowed	1	Write access to override region n is allowed																														
					Software execute																														
			NotAllowed	0	Software execution from override region n is not allowed																														
			Allowed	1	Software execution from override region n is allowed																														
D	RW	SECATTR			Security mapping																														
			Secure	1	Override region n is mapped in secure memory address space																														
			NonSecure	0	Override region n is mapped in non-secure memory address space																														

7.7.4.3.6.5 OVERRIDE[n].PERMMASK (n=0..4)

Address offset: 0x814 + (n × 0x20)

Masks permission setting fields from register [OVERRIDE.PERM](#)

See section *Validate an access* above.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
D	R/W	Field	Value ID	Value	Description																														
A	RW	READ			Read mask																														
			Masked	0	Permission setting READ in OVERRIDE register will not be applied																														
			UnMasked	1	Permission setting READ in OVERRIDE register will be applied																														
B	RW	WRITE			Write mask																														
			Masked	0	Permission setting WRITE in OVERRIDE register will not be applied																														
			UnMasked	1	Permission setting WRITE in OVERRIDE register will be applied																														
C	RW	EXECUTE			Execute mask																														
			Masked	0	Permission setting EXECUTE in OVERRIDE register will not be applied																														
			UnMasked	1	Permission setting EXECUTE in OVERRIDE register will be applied																														
D	RW	SECATTR			Security mapping mask																														
			Masked	0	Permission setting SECATTR in OVERRIDE register will not be applied																														
			UnMasked	1	Permission setting SECATTR in OVERRIDE register will be applied																														

7.7.5 SPU — System protection unit

SPU configures the access privileges for a peripheral.

SPU allows configuring access controls individually for each peripheral, and for some peripheral features. For example, a DPPI channel can be configured with different access controls than the peripheral.

SPU controls access according to TrustZone security attributes. If a peripheral or feature is configured as secure, only TrustZone secure accesses are allowed. If a peripheral or feature is configured as non-secure, then accesses are allowed both from secure and non-secure masters.

For some peripherals, the peripheral's DMA has a separate security configuration. If the peripheral is configured as secure, the peripheral's DMA can be configured to perform either secure or non-secure accesses. If the peripheral is configured as non-secure, the peripheral's DMA will always perform non-secure accesses.

7.7.5.1 General concepts

The SPU provides the register interface to configure and enforce the access privileges per peripheral, and where applicable, individual features of the peripheral such as GPIO pins, DPPI channels, etc.

Any accesses to a peripheral or a peripheral feature are validated against the SPU configuration for the security attributes.

Security attributes of a peripheral normally applies to all registers of the peripheral. However, some peripherals have split security to individual features within the peripheral, such as individual pins or DPPI channels. For these split feature peripherals, access is granted on a per-bit or per-register level. Unless mentioned otherwise, the term peripheral is used in the remainder of this section to refer to both a peripheral and an individual peripheral feature.

Each APB bus has its own SPU instance that controls the resource of that bus. The SPU must be configured for security attributes of the peripherals. The SPU is always a secure peripheral.

- See [Instantiation](#) on page 216 to find the SPU instance used by the peripheral.
- The APB bus number can be extracted from the peripheral address. See [APB bus number](#) to find the APB bus number for a peripheral.
- See [Block diagram](#) on page 15 for an overview over APB buses, the peripherals on that bus, and their controlling SPU instance.

See [APB bus number](#) for information on extracting the Peripheral slave index from a peripheral address.

The following example shows which SPU instance to use for SAADC peripheral to configure the peripheral permissions using `PERIPH[n].PERM`:

```
#define SPU_CORTEX_ADDRESS_REGION    (0x50000000)

uint32_t perip_addr = NRF_SAADC_S_BASE;

uint32_t apb_bus_number = (perip_addr & 0x00FC0000);
uint32_t apb_slave_index = (perip_addr & 0x0003F000) >> 12;

// Get the address to the SPU instance
NRF_SPU_Type *p_spu = (NRF_SPU_Type*) (SPU_CORTEX_ADDRESS_REGION |
apb_bus_number);

// Configure PERIPH[n].PERM.SECATTR to secure for SAADC
p_spu->PERIPH[apb_slave_index].PERM =
(p_spu->PERIPH[apb_slave_index].PERM &
~SPU_PERIPH_PERM_SECATTR_Msk) |
(SPU_PERIPH_PERM_SECATTR_Secure <<
SPU_PERIPH_PERM_SECATTR_Pos)
```

See [Instantiation](#) on page 216 to find the value of SLAVE_BITS for each SPU instance.

SPU supports secure and non-secure accesses based on TrustZone. On each access to a peripheral address, the security state of the master initiating the transaction is verified against the SPU security attribute configuration of the peripheral. The following figure shows a simplified view of the SPU registers controlling several internal modules.

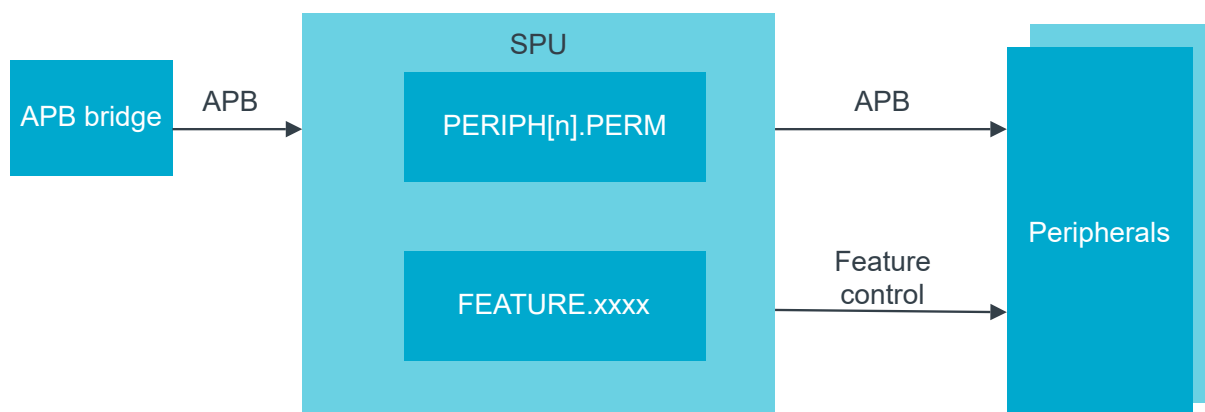


Figure 34: Simplified view of peripherals and peripheral features using SPU

The protection logic implements a read-as-zero/write-ignore (RAZ/WI) policy:

- A read operation that is not allowed by the SPU will always return a zero value on the bus, preventing information leak.
- A write operation that is not allowed by the SPU will be ignored.

An access error on peripherals managed by an SPU result in the PERIPHACCERR event on the SPU.

7.7.5.2 Peripheral access control

Peripheral access control depends on the security attributes.

Peripheral security attributes are defined in the *Peripheral Instantiation* table as one of the following:

Always Secure (HF S)

Access to the peripheral is always restricted to secure code.

Always Non-secure (HF NS)

Access to the peripheral is always allowed from both secure and non-secure code.

User selectable (US)

The security attribute can be configured for secure or non-secure access.

The full list of peripherals and their corresponding security attributes can be found in the *Instantiation* table in *Memory* section. For each peripheral with ID n , the register `PERIPH[n].PERM.SECUREMAPPING` will show whether the security attribute for this peripheral is user selectable or not.

The security attribute can be configured using the register `PERIPH[n].PERM.SECATTR`, if user selectable.

The DMA security attribute is determined as follows:

- If `PERIPH[n].PERM.DMA` is set to `NoSeparateAttribute`, then `PERIPH[n].PERM.DMASEC` cannot be configured, it has the same value as `PERIPH[n].PERM.SECATTR`.
- If `PERIPH[n].PERM.DMA` is set to `SeparateAttribute` and `PERIPH[n].PERM.SECATTR` is set to secure, then `PERIPH[n].PERM.DMASEC` is configurable. It is by default set to secure.

Secure code can access both secure peripherals and non-secure peripherals.

The DMA Privilege attribute is determined as follows:

- If `PERIPH[n].PERM.DMA` is set to `NoSeparateAttribute`, then `PERIPH[n].PERM.DMAPRIV` cannot be configured, it has the same value as `PERIPH[n].PERM.PRIVATTR`.
- If `PERIPH[n].PERM.DMA` is set to `SeparateAttribute` and `PERIPH[n].PERM.PRIVATTR` is set to `Privileged`, then `PERIPH[n].PERM.DMAPRIV` is configurable. It is by default set to `Unprivileged`.

7.7.5.2.1 Peripherals with split security

Peripherals with split security allow more detailed configuration.

When peripherals have split security, then the security of each feature in the peripheral can be configured individually using register FEATURE.

Each SPU instance can have different numbers of features. See [the instantiation table](#) for an overview of features supported by the split security peripherals.

7.7.5.2.2 Peripheral address mapping

Peripherals that have non-secure security mapping have their address starting with 0x4XXX_XXXX.

Peripherals that have secure security mapping have their address starting with 0x5XXX_XXXX.

Peripherals with a user-selectable security mapping are available at an address starting with:

- 0x4XXX_XXXX, if the peripheral security attribute is set to non-secure
- 0x5XXX_XXXX, if the peripheral security attribute is set to secure

Note: Accesses to the 0x4XXX_XXXX address range from secure or non-secure code for a peripheral marked as secure will result in a bus-error.

Secure code accessing the 0x5XXX_XXXX address range of a peripheral marked as non-secure will also result in a bus-error.

Peripherals with a split security mapping are available at an address starting with:

- 0x4XXX_XXXX for non-secure access and 0x5XXX_XXXX for secure access, if the peripheral security attribute is set to non-secure
 - Secure registers in the 0x4XXX_XXXX range are not visible for secure or non-secure code, and an attempt to access such a register will generate a peripheral access error, and result in write-ignore, read as zero behavior.
 - Secure code can access both non-secure and secure registers in the 0x5XXX_XXXX range
- 0x5XXX_XXXX, if the peripheral security attribute is set to secure

Note: An access to an address that is within the address range of an APB interconnect, but is not within the address range of a peripheral, will generate a peripheral access error, and result in write-ignore, read as zero behavior.

7.7.5.2.3 Special considerations for peripherals with DMA master

Peripherals containing a DMA master can be configured so the security attribute of the DMA transfers is different from the security attribute of the peripheral itself. This allows a secure peripheral to do non-secure data transfers to or from the system memories.

If the following conditions are met:

- The DMA field of `PERIPH[n].PERM.DMA` is "SeparateAttribute"
- The peripheral itself is secure (`PERIPH[n].PERM.SECATTR == 1`)

Then it is possible to select the security attribute of the DMA transfers using the field DMASEC (`PERIPH[n].PERM.DMASEC == Secure` and `PERIPH[n].PERM.DMASEC == NonSecure`) in `PERIPH[n].PERM`.

7.7.5.2.4 Peripheral access error reporting

The SPU generates a peripheral access error event once access violation is detected.

The following will happen if the logic controlled by the SPU detects an access violation on one of the peripherals:

- The faulty transfer will be blocked

- In case of a read transfer, the data will read as zero
- If supported by the master, feedback is sent to the master through specific bus error signals. If the master is a processor supporting Arm TrustZone for Cortex-M, a SecureFault exception will be generated for security related errors.
- The PERIPHACCERR event will be triggered.

7.7.5.3 Feature access control

Access to the features can be restricted. A feature can be declared as secure so that only secure peripherals can access it.

The security attribute of a feature is configured by using corresponding SPU's feature register. When the secure attribute is set for a feature, only peripherals that have the secure attribute will be able to access that feature. For example, register FEATURE.IPCT.DPPI.SECATTR is used to configure secure attribute for the DPPI channel in IPCT. When the secure attribute is set, only peripherals that have the secure attribute set will be able to publish events to this channel or subscribe to this channel to receive tasks.

See [the SPU configuration](#) to find the features supported by each SPU instance.

7.7.5.5 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
SPU00	GLOBAL	0x50040000	HF	S	NA	No	System protection unit SPU00
SPU10	GLOBAL	0x50080000	HF	S	NA	No	System protection unit SPU10
SPU20	GLOBAL	0x500C0000	HF	S	NA	No	System protection unit SPU20
SPU30	GLOBAL	0x50100000	HF	S	NA	No	System protection unit SPU30

Configuration

Instance	Domain	Configuration
SPU00	GLOBAL	Supports FEATURE.DPPIC[n] Supports FEATURE.GPIO[n] Supports FEATURE.CRACEN SLAVE_BITS=4 (number of address bits required to represent the peripheral slave index)
SPU10	GLOBAL	Supports FEATURE.DPPIC[n] SLAVE_BITS=4 (number of address bits required to represent the peripheral slave index)
SPU20	GLOBAL	Supports FEATURE.DPPIC[n] Supports FEATURE.GPIOTE[n] Supports FEATURE.GRTC[n] Supports FEATURE.GPIO[n] SLAVE_BITS=4 (number of address bits required to represent the peripheral slave index)
SPU30	GLOBAL	Supports FEATURE.DPPIC[n] Supports FEATURE.GPIOTE[n] Supports FEATURE.GPIO[n] SLAVE_BITS=4 (number of address bits required to represent the peripheral slave index)

Register overview

Register	Offset	TZ	Description
EVENTS_PERIPHACCERR	0x100		A security violation has been detected on one or several peripherals
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
INTPEND	0x30C		Pending interrupts
PERIPHACCERR.ADDRESS	0x404		Address of the transaction that caused first error.
PERIPH[n].PERM	0x500		Get and set the applicable access permissions for the peripheral slave index n
FEATURE.DPPIC.CH[n]	0x680		Configuration of features for channel n of DPPIC
FEATURE.DPPIC.CHG[n]	0x6E0		Configuration of features for channel group n of DPPIC
FEATURE.GPIOTE[n].CH[o]	0x700		Configuration of features for channel o of GPIOTE[n]
FEATURE.GPIOTE[n].INTERRUPT[o]	0x720		Configuration of features for interrupt o of GPIOTE[n]
FEATURE.GPIO[n].PIN[o]	0x800		Configuration of features for GPIO[n] PIN[o]
FEATURE.CRACEN.SEED	0x980		Configuration for CRACEN SEED
FEATURE.GRTC.CC[n]	0xD00		Configuration of features for CC n of GRTC
FEATURE.GRTC.PWMCONFIG	0xD74		Configuration of feature for PWMCONFIG of GRTC
FEATURE.GRTC.CLK	0xD78		Configuration of features for CLKOUT/CLKCFG of GRTC
FEATURE.GRTC.SYSCOUNTER	0xD7C		Configuration of features for SYSCOUNTERL/SYSCOUNTERH of GRTC
FEATURE.GRTC.INTERRUPT[n]	0xD80		Configuration of features for interrupt n of GRTC

7.7.5.5.1 EVENTS_PERIPHACCERR

Address offset: 0x100

A security violation has been detected on one or several peripherals

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																													
A	RW	EVENTS_PERIPHACCERR				A security violation has been detected on one or several peripherals																													
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

7.7.5.5.2 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																																							A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	PERIPHACCERR				Enable or disable interrupt for event PERIPHACCERR																																	
			Disabled	0		Disable																																	
			Enabled	1		Enable																																	

7.7.5.5.3 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	PERIPHACCERR				Write '1' to enable interrupt for event PERIPHACCERR																													
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

7.7.5.5.4 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	PERIPHACCERR			Write '1' to disable interrupt for event PERIPHACCERR																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

7.7.5.5.5 INTPEND

Address offset: 0x30C

Pending interrupts

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	R	PERIPHACCERR			Read pending status of interrupt for event PERIPHACCERR																																	
			NotPending	0	Read: Not pending																																	
			Pending	1	Read: Pending																																	

7.7.5.5.6 PERIPHACCERR.ADDRESS

Address offset: 0x404

Address of the transaction that caused first error.

The event PERIPHACCERR must be cleared to clear this register.

Note: Only the lower 16 bits of the address are captured into the register. The upper 16 bits correspond to the upper 16 bits of the SPU's base address.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	ADDRESS										Address																							

7.7.5.5.7 PERIPH[n].PERM (n=0..63)

Address offset: 0x500 + (n × 0x4)

Get and set the applicable access permissions for the peripheral slave index n

Note: Reset values are unique per peripheral instantiation. Please refer to the peripheral instantiation table. Entries not listed in the instantiation table are undefined.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																					
ID				F																E								D				C		B		B		A		A	
Reset 0x8000000A				1 0 1 0 1 0																																					
ID	R/W	Field	Value ID	Value	Description																																				
A	R	SECUREMAPPING			Read capabilities for TrustZone Cortex-M secure attribute																																				
			NonSecure	0	This peripheral is always accessible as a non-secure peripheral																																				
			Secure	1	This peripheral is always accessible as a secure peripheral																																				
			UserSelectable	2	Non-secure or secure attribute for this peripheral is defined by the PERIPH[n].PERM register																																				
			Split	3	This peripheral implements the split security mechanism.																																				
B	R	DMA			Read the peripheral DMA capabilities																																				
			NoDMA	0	Peripheral has no DMA capability																																				
			NoSeparateAttribute	1	Peripheral has DMA and DMA transfers always have the same security attribute as assigned to the peripheral																																				
			SeparateAttribute	2	Peripheral has DMA and DMA transfers can have a different security attribute than the one assigned to the peripheral																																				

Bit number		31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																							
ID		F																E								D				C				B				A			
Reset 0x8000000A		1 0 1 0 1 0																																							
ID	R/W	Field	Value ID	Value	Description																																				
C	RW	SECATTR			Peripheral security mapping																																				
					This bit has effect only if PERIPH[n].PERM.SECUREMAPPING is UserSelectable or Split																																				
			Secure	1	Peripheral is mapped in secure peripheral address space																																				
			NonSecure	0	If SECUREMAPPING == UserSelectable: Peripheral is mapped in non-secure peripheral address space.																																				
					If SECUREMAPPING == Split: Peripheral is mapped in non-secure and secure peripheral address space.																																				
D	RW	DMASEC			Security attribution for the DMA transfer																																				
					This bit has effect only if PERIPH[n].PERM.SECATTR is set to secure and PERIPH[n].PERM.DMA is set to SeparateAttribute.																																				
			Secure	1	DMA transfers initiated by this peripheral have the secure attribute set																																				
			NonSecure	0	DMA transfers initiated by this peripheral have the non-secure attribute set																																				
E	RW	LOCK W1S			Register lock																																				
			Unlocked	0	This register can be updated																																				
			Locked	1	The content of this register can not be changed until the next reset																																				
					When Locked, it remains Locked until the next reset cycle.																																				
F	R	PRESENT			Indicates if a peripheral is present with peripheral slave index n																																				
			NotPresent	0	Peripheral is not present																																				
			IsPresent	1	Peripheral is present																																				

7.7.5.5.8 FEATURE.DPPIC.CH[n] (n=0..23)

Address offset: 0x680 + (n × 0x4)

Configuration of features for channel n of DPPIC

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																				B								A							
Reset 0x00100010				0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	SECATTR			SECATTR feature																														
			NonSecure	0	Feature is available for non-secure usage																														
			Secure	1	Feature is reserved for secure usage																														
B	RW	LOCK W1S			LOCK feature																														
			Unlocked	0	Feature permissions can be updated																														
			Locked	1	Feature permissions can not be changed until the next reset																														
					When Locked, it remains Locked until the next reset cycle.																														

7.7.5.5.9 FEATURE.DPPIC.CHG[n] (n=0..7)

Address offset: 0x6E0 + (n × 0x4)

Configuration of features for channel group n of DPPIC

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																									
ID																												B				A													
Reset 0x00100010				0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0																																									
ID	R/W	Field	Value ID	Value	Description																																								
A	RW	SECATTR			SECATTR feature																																								
			NonSecure	0	Feature is available for non-secure usage																																								
			Secure	1	Feature is reserved for secure usage																																								
B	RW	LOCK W1S			LOCK feature																																								
			Unlocked	0	Feature permissions can be updated																																								
			Locked	1	Feature permissions can not be changed until the next reset																																								
					When Locked, it remains Locked until the next reset cycle.																																								

7.7.5.5.10 FEATURE.GPIOTE[n].CH[o] (n=0..1) (o=0..7)

Address offset: $0x700 + (n \times 0x40) + (o \times 0x4)$

Configuration of features for channel o of GPIOTE[n]

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0												
ID																												B				A															
Reset 0x00100010				0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0										
ID	R/W	Field	Value ID	Value	Description																																										
A	RW	SECATTR			SECATTR feature																																										
			NonSecure	0	Feature is available for non-secure usage																																										
			Secure	1	Feature is reserved for secure usage																																										
B	RW	LOCK W1S			LOCK feature																																										
			Unlocked	0	Feature permissions can be updated																																										
			Locked	1	Feature permissions can not be changed until the next reset																																										
					When Locked, it remains Locked until the next reset cycle.																																										

7.7.5.5.11 FEATURE.GPIOTE[n].INTERRUPT[o] (n=0..1) (o=0..7)

Address offset: $0x720 + (n \times 0x40) + (o \times 0x4)$

Configuration of features for interrupt o of GPIOTE[n]

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0												
ID																												B				A															
Reset 0x00100010				0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0										
ID	R/W	Field	Value ID	Value	Description																																										
A	RW	SECATTR			SECATTR feature																																										
			NonSecure	0	Feature is available for non-secure usage																																										
			Secure	1	Feature is reserved for secure usage																																										
B	RW	LOCK W1S			LOCK feature																																										
			Unlocked	0	Feature permissions can be updated																																										
			Locked	1	Feature permissions can not be changed until the next reset																																										
					When Locked, it remains Locked until the next reset cycle.																																										

7.7.5.5.12 FEATURE.GPIO[n].PIN[o] (n=0..2) (o=0..31)

Address offset: $0x800 + (n \times 0x80) + (o \times 0x4)$

Configuration of features for GPIO[n] PIN[o]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																				B								A							
Reset 0x00100010				0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	SECATTR			SECATTR feature																														
			NonSecure	0	Feature is available for non-secure usage																														
			Secure	1	Feature is reserved for secure usage																														
B	RW	LOCK W1S			LOCK feature																														
			Unlocked	0	Feature permissions can be updated																														
			Locked	1	Feature permissions can not be changed until the next reset																														
					When Locked, it remains Locked until the next reset cycle.																														

7.7.5.5.13 FEATURE.CRACEN.SEED

Address offset: 0x980

Configuration for CRACEN SEED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																				B								A							
Reset 0x00020010				0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	SECATTR			SECATTR feature																														
			NonSecure	0	Feature is available for non-secure usage																														
			Secure	1	Feature is reserved for secure usage																														
B	RW	LOCK W1S			LOCK feature																														
			Unlocked	0	Feature permissions can be updated																														
			Locked	1	Feature permissions can not be changed until the next reset																														
					When Locked, it remains Locked until the next reset cycle.																														

7.7.5.5.14 FEATURE.GRTC.CC[n] (n=0..23)

Address offset: 0xD00 + (n × 0x4)

Configuration of features for CC n of GRTC

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																				B								A							
Reset 0x00100010				0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	SECATTR			SECATTR feature																														
			NonSecure	0	Feature is available for non-secure usage																														
			Secure	1	Feature is reserved for secure usage																														
B	RW	LOCK W1S			LOCK feature																														
			Unlocked	0	Feature permissions can be updated																														
			Locked	1	Feature permissions can not be changed until the next reset																														
					When Locked, it remains Locked until the next reset cycle.																														

7.7.5.5.15 FEATURE.GRTC.PWMCONFIG

Address offset: 0xD74

Configuration of feature for PWMCONFIG of GRTC

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																												B				A			
Reset 0x00100010				0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	SECATTR			SECATTR feature																														
			NonSecure	0	Feature is available for non-secure usage																														
			Secure	1	Feature is reserved for secure usage																														
B	RW	LOCK W1S			LOCK feature																														
			Unlocked	0	Feature permissions can be updated																														
			Locked	1	Feature permissions can not be changed until the next reset																														
					When Locked, it remains Locked until the next reset cycle.																														

7.7.5.5.16 FEATURE.GRTC.CLK

Address offset: 0xD78

Configuration of features for CLKOUT/CLKCFG of GRTC

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																													B				A				
Reset 0x00100010					0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																
A	RW	SECATTR			SECATTR feature																																
			NonSecure	0	Feature is available for non-secure usage																																
			Secure	1	Feature is reserved for secure usage																																
B	RW	LOCK W1S			LOCK feature																																
			Unlocked	0	Feature permissions can be updated																																
			Locked	1	Feature permissions can not be changed until the next reset																																
					When Locked, it remains Locked until the next reset cycle.																																

7.7.5.5.17 FEATURE.GRTC.SYSCOUNTER

Address offset: 0xD7C

Configuration of features for SYSCOUNTERL/SYSCOUNTERH of GRTC

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																												B				A							
Reset 0x00100010				0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	SECATTR				SECATTR feature																																	
			NonSecure	0	Feature is available for non-secure usage																																		
			Secure	1	Feature is reserved for secure usage																																		
B	RW	LOCK W1S				LOCK feature																																	
			Unlocked	0	Feature permissions can be updated																																		
			Locked	1	Feature permissions can not be changed until the next reset																																		
				When Locked, it remains Locked until the next reset cycle.																																			

7.7.5.5.18 FEATURE.GRTC.INTERRUPT[n] (n=0..15)

Address offset: 0xD80 + (n × 0x4)

Configuration of features for interrupt n of GRTC

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																				B				A														
Reset 0x00100010				0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	SECATTR			SECATTR feature																																	
			NonSecure	0	Feature is available for non-secure usage																																	
			Secure	1	Feature is reserved for secure usage																																	
B	RW	LOCK W1S			LOCK feature																																	
			Unlocked	0	Feature permissions can be updated																																	
			Locked	1	Feature permissions can not be changed until the next reset																																	
					When Locked, it remains Locked until the next reset cycle.																																	

7.7.6 TAMPC — Tamper controller

The tamper controller handles inputs from internal and external physical attack detectors and controls the device response.

The following figure displays an overview of the TAMPC detectors, inputs and outputs.

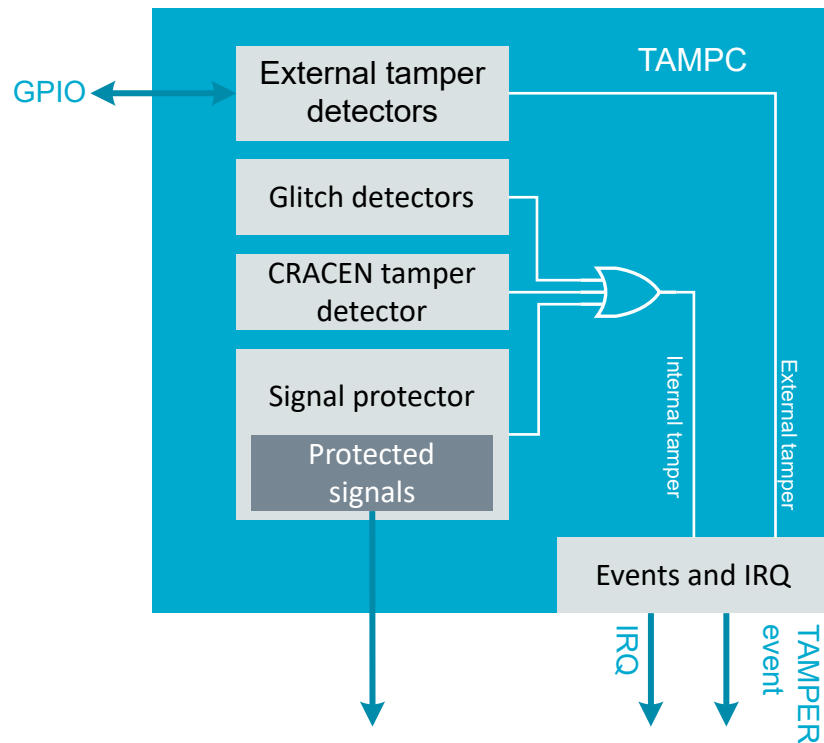


Figure 35: TAMPC overview

The Tamper Controller implements the following physical security features:

- Detection of external tampering attacks
 - Detector supporting an active driven shield mounted on PCB on top of device
- Detection of fault injection attacks (voltage glitching, electromagnetic fault injection, etc.)
 - Signal protector to guard critical configuration signals
 - Glitch detectors to detect timing violations of internal logic
 - Built-in self-check for correctness inside the CRACEN

The tamper detectors are divided into two categories, external and internal. The external detectors rely on external stimuli through dedicated GPIOs, and the internal detectors rely on internal signals not exposed outside the device package.

External tamper detectors

A tamper attack detected by any of the external tamper detectors indicates that a break-in attack is ongoing by e.g. breaking product encapsulation. This is detected through the external active shield detectors.

Internal tamper detectors

A tamper attack detected by any of the internal tamper detectors indicates that the device's internal logic could be affected by the attack, hence the system state can no longer be trusted. The default system

reaction from reset is to trigger a system wide reset if any of the internal tamper detectors detects a tamper attack.

7.7.6.1 Active shield

The TAMPC implements support for an active shield to protect against physical access to the device and its connections on the PCB level.

The active shield detector is enabled using the [PROTECT.ACTIVESHIELD.CTRL](#) on page 205 register. The active shield detector has a number of channels. The TAMPC will react depending on its configuration when a channel in the active shield is broken, i.e. there is a mismatch between the input and output signal of an enabled channel in the active shield. A detected broken channel in the active shield will either raise a TAMPC EVENT or trigger a system wide reset with reset reason `SECTAMPER` depending on the status of the [PROTECT.EXTRESETEN.CTRL](#) on page 209 register.

A channel in the active shield detector consists of a signal propagating from an output pin to an input pin. The channels are enabled using the `CH[i]` fields in the register [ACTIVESHIELD.CHEN](#) on page 200. The GPIO pins reserved for the the active shield detector channels must be configured before the channels are ready to be used. Pin direction and `CTRLSEL` must be set according to the register interface in the GPIO peripheral. See the GPIO chapter for more information about configuring the reserved active shield detector pins, and which pins are reserved for the active shield detector. Pins reserved for the active shield detector can be used as normal GPIO pins when the channel is unused.

The active shield detector contains a Pseudo-Random Bit Sequence (PRBS) generator. A PRBS signal is generated on an output pin at each rising edge of the 32 KHz clock. The signal is routed through an external shield to an input pin. The signal on the input pin is sampled on the falling edge of the 32KHz clock. If the sampled signal does not match the transmitted signal, a channel in the external shield is assumed broken, and a tamper event is produced.

7.7.6.2 CRACEN tamper detector

The cryptographic accelerator engine (CRACEN) implements a separate tamper detector mechanism.

CRACEN notifies TAMPC if tampering is detected during AES and IKG operations. The CRACEN tamper detector is always enabled. Register [PROTECT.CRACENTAMP.CTRL](#) controls if TAMPC reacts upon tampering detected by CRACEN. A detected tamper event from CRACEN will either generate a TAMPC event or trigger a system wide reset with reset reason `SECTAMPER` depending on the status of the register [PROTECT.INTRESETEN.CTRL](#) on page 210. The internal tamper reset enable signal (`INTRESETEN`) is enabled from reset.

7.7.6.3 Glitch detector

The device implements general detectors against fault injection attacks.

A grid of detectors are strategically placed among the digital logic to detect local timing glitches, i.e. timing violations. The glitch detectors monitor the effects of fault injection attack attempts, instead of monitoring the attack attempt itself. Fault injection attempts could be utilizing e.g. voltage glitches on supply or decoupling pins, or electro magnetic fault injection (EMFI) techniques. This makes the glitch detectors suited as general detectors against fault injection attacks. The detectors are designed and tuned to be more sensitive to timing violations than normal logic, making the digital logic able to react before an injected fault is propagated through the system.

The glitch detectors are enabled from reset, and can be disabled using the [PROTECT.GLITCHSLOWDOMAIN.CTRL](#) on page 207 and [PROTECT.GLITCHFASTDOMAIN.CTRL](#) on page 208 registers for debugging purposes only. A detected internal tamper event will either raise a TAMPC EVENT or trigger a system wide reset with reset reason `SECTAMPER` depending on the status of the [PROTECT.INTRESETEN.CTRL](#) on page 210 register. The internal tamper reset enable signal (`INTRESETEN`) is enabled from reset.

7.7.6.4 Signal protector

The device implements detectors to protect selected signals controlling critical device features.

The signal protector implements a detector per protected signal, detecting unintentional value changes in the protected signals. The detectors notifies TAMPC if a protected signal changes value caused by tampering. The detectors are enabled from reset, and can be disabled using the register for debugging purposes only. A detected unintentional value change in any of the protected signals leads to an internal tamper event. This will either raise a TAMPC EVENT or trigger a system wide reset with reset reason `SECTAMPER` depending on the status of the `PROTECT.INTRESETEN.CTRL` on page 210 register. The `INTRESETEN` is enabled from reset. The `PROTECT.nnn.STATUS` registers indicates which protected signal had an unintentional value change when the `INTENRESETEN` is disabled.

The signal protector implements a two stage write cycle to change the value of a protected signal, in addition to a required write key which must be included for all register writes. The two stages consist of:

1. an initial register write to clear the write protection
2. the register write to change the signal's value

Write `Clear` to the `WRITEPROTECTION` field in the `PROTECT.nnn.CTRL` register to clear the write protection in the first register write, and then write to the `VALUE` and `LOCK` fields in the subsequent register write operation.

Note: It is required to clear the `WRITEPROTECTION` field before any updates to the `VALUE` and `LOCK` fields are accepted by the register.

The write protection is automatically re-enabled after the subsequent write to change the `VALUE` field when the register write does not include the `Clear` value in the `WRITEPROTECTION` field.

The `LOCK` field controls a lock feature which prevents further updates to the `VALUE` and `LOCK` fields until a reset with required reset source for the specific signal is issued.

A `WRITEERROR` EVENT is raised in any of the following conditions:

- a register write does not have the correct write key
- the write protection is active and the write operation does not contain the value to clear the write protection.
- the lock is enabled

The sequence to change the `VALUE` or `LOCK` fields in the `PROTECT.nnn.CTRL` registers is as follows:

1. Write `Clear` to the `WRITEPROTECTION` field and `KEY` to the `KEY` field.
2. Write `Disabled` to the `WRITEPROTECTION` field, `KEY` to the `KEY` field, and the desired `LOCK` and `VALUE` fields.

7.7.6.4.1 Debugger signals

TAMPC provides protection for the following ARM Coresight debugger signals.

- **NIDEN**: Non-Invasive Debug Enable, i.e. ETM / ITM trace and other non-halting debug methods.
- **DBGEN**: Invasive Debug Enable, ie. the debugger may halt the CPU for debug purposes.
- **SPNIDEN**: Secure Privileged Non-Invasive Debug Enable, same as `NIDEN` with TrustZone security attribute secure.
- **SPIDEN**: Secure Privileged Debug Enable, same as `DBGEN` with Trustzone security attribute secure.

See the ARM Coresight Technical Reference manual for details on these signals.

More information about the usage of the protected debugger signals, see [Debug and trace](#) on page 749.

7.7.6.5 TAMPC reset behavior

TAMPC registers are reset by different sources.

Protected signals in TAMPC are divided into the following reset source categories.

- Category 1 – Brownout reset, power-on reset
- Category 2 – Pin reset and TAMPC reset
- Category 3 – Watchdog timer reset, CPU lockup reset, System reset request, and Wake-up from System OFF reset

For more information about reset sources, see [RESET — Reset control](#) on page 106.

Register	Function	Reset value	Reset source		
			Cat 1	Cat 2	Cat 3
PROTECT.DOMAIN[0].DBGEN	Allow invasive debugging in non-secure mode of Arm Cortex-M33.	0	x	x	
PROTECT.DOMAIN[0].NIDEN	Allow non-invasive debugging in non-secure mode of Arm Cortex-M33.	0	x	x	
PROTECT.DOMAIN[0].SPIDEN	Allow invasive debugging in secure mode of Arm Cortex-M33.	0	x	x	
PROTECT.DOMAIN[0].SPNIDEN	Allow non-invasive debugging in secure mode of Arm Cortex-M33.	0	x	x	
PROTECT.AP[0].DBGEN	Allow debugging of FLPR RISC-V CPU.	0	x	x	
PROTECT.ACTIVESHIELD	Enable active shield detector.	0	x	x	x
PROTECT.CRACENTAMP	Enable CRACEN tamper detector.	1	x	x	x
PROTECT.GLITCHSLOWDOMAIN	Enable slow domain glitch detector.	1	x	x	x
PROTECT.GLITCHFASTDOMAIN	Enable fast domain glitch detector.	1	x	x	x
PROTECT.EXTRESETEN	Enable automatic reset from external tamper detectors events.	0	x	x	x
PROTECT.INTRESETEN	Enable automatic reset from internal tamper detector events.	1	x	x	x
PROTECT.ERASEPROTECT	Allow device erase using CTRL-AP.	0	x	x	x

Table 30: TAMPC protected signals

7.7.6.6 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
TAMPC	GLOBAL	0x500DC000	HF	S	NA	No	Tamper controller TAMPC

Configuration

Instance	Domain	Configuration
TAMPC	GLOBAL	For the active shield function, use dedicated pins on P1 Reset value of field VALUE in register PROTECT.INTRESETE.N.CTRL: 1

Register overview

Register	Offset	TZ	Description
EVENTS_TAMPER	0x100		Tamper controller detected an error.
EVENTS_WRITEERROR	0x104		Attempt to write a VALUE in PROTECT registers without clearing the WRITEPROTECT.
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
INTPEND	0x30C		Pending interrupts
STATUS	0x400		The tamper controller status.
ACTIVESHIELD.CHEN	0x404		Active shield detector channel enable register.
PROTECT.DOMAIN[n].DBGEN.CTRL	0x500		Control register for invasive (halting) debug enable for the local debug components within domain n.
PROTECT.DOMAIN[n].DBGEN.STATUS	0x504		Status register for invasive (halting) debug enable for domain n.
PROTECT.DOMAIN[n].NIDEN.CTRL	0x508		Control register for non-invasive debug enable for the local debug components within domain n.
PROTECT.DOMAIN[n].NIDEN.STATUS	0x50C		Status register for non-invasive debug enable for domain n.
PROTECT.DOMAIN[n].SPIDEN.CTRL	0x510		Control register for secure privileged invasive (halting) debug enable for the local debug components within domain n.
PROTECT.DOMAIN[n].SPIDEN.STATUS	0x514		Status register for secure privileged invasive (halting) debug enable for domain n.
PROTECT.DOMAIN[n].SPNIDEN.CTRL	0x518		Control register for secure privileged non-invasive debug enable for the local debug components within domain n.
PROTECT.DOMAIN[n].SPNIDEN.STATUS	0x51C		Status register for secure privileged non-invasive debug enable for domain n.
PROTECT.AP[n].DBGEN.CTRL	0x700		Control register to enable invasive (halting) debug in domain n's access port.
PROTECT.AP[n].DBGEN.STATUS	0x704		Status register for invasive (halting) debug enable for domain n's access port.
PROTECT.ACTIVESHIELD.CTRL	0x900		Control register for active shield detector enable signal.
PROTECT.ACTIVESHIELD.STATUS	0x904		Status register for active shield detector enable signal.
PROTECT.CRACENTAMP.CTRL	0x938		Control register for CRACEN tamper detector enable signal.
PROTECT.CRACENTAMP.STATUS	0x93C		Status register for CRACEN tamper detector enable signal.
PROTECT.GLITCHSLOWDOMAIN.CTRL	0x940		Control register for slow domain glitch detectors enable signal.
PROTECT.GLITCHSLOWDOMAIN.STATUS	0x944		Status register for slow domain glitch detectors enable signal.
PROTECT.GLITCHFASTDOMAIN.CTRL	0x948		Control register for fast domain glitch detectors enable signal.
PROTECT.GLITCHFASTDOMAIN.STATUS	0x94C		Status register for fast domain glitch detectors enable signal.
PROTECT.EXTRESETE.N.CTRL	0x970		Control register for external tamper reset enable signal.
PROTECT.EXTRESETE.N.STATUS	0x974		Status register for external tamper reset enable signal.
PROTECT.INTRESETE.N.CTRL	0x978		Control register for internal tamper reset enable signal.
PROTECT.INTRESETE.N.STATUS	0x97C		Status register for internal tamper reset enable signal.
PROTECT.ERASEPROTECT.CTRL	0x980		Control register for erase protection.
PROTECT.ERASEPROTECT.STATUS	0x984		Status register for eraseprotect.

7.7.6.6.1 EVENTS_TAMPER

Address offset: 0x100

Tamper controller detected an error.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_TAMPER						Tamper controller detected an error.																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

7.7.6.6.2 EVENTS_WRITEERROR

Address offset: 0x104

Attempt to write a VALUE in PROTECT registers without clearing the WRITEPROTECT.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																																							A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value				Description																															
A	RW	EVENTS_WRITEERROR						Attempt to write a VALUE in PROTECT registers without clearing the WRITEPROTECT.																															
			NotGenerated	0				Event not generated																															
			Generated	1				Event generated																															

7.7.6.6.3 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	TAMPER			Enable or disable interrupt for event TAMPER																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
B	RW	WRITEERROR			Enable or disable interrupt for event WRITEERROR																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

7.7.6.6.4 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	TAMPER			Write '1' to enable interrupt for event TAMPER																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	WRITEERROR			Write '1' to enable interrupt for event WRITEERROR																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

7.7.6.6.5 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	TAMPER			Write '1' to disable interrupt for event TAMPER																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	WRITEERROR			Write '1' to disable interrupt for event WRITEERROR																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

7.7.6.6.6 INTPEND

Address offset: 0x30C

Pending interrupts

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	TAMPER			Read pending status of interrupt for event TAMPER																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
B	R	WRITEERROR			Read pending status of interrupt for event WRITEERROR																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														

7.7.6.6.7 STATUS

Address offset: 0x400

The tamper controller status.

Note: Unless cleared, the STATUS register will be cumulative. A field is cleared by writing '1' to it.

Note: The glitch detectors must be reset using their CTRL registers before the STATUS register bits for glitch detectors can be cleared. The glitch detector continuously drives its output status signal to the STATUS register, hence clearing only the STATUS register is not sufficient.

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			H G F E																D				C B				A							
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	ACTIVESHIELD W1C			Active shield detector detected an error.																													
			NotDetected	0	Not detected.																													
			Detected	1	Detected.																													
B	RW	PROTECT W1C			Error detected for the protected signals.																													
			NotDetected	0	Not detected.																													
			Detected	1	Detected.																													
C	RW	CRACENTAMP W1C			CRACEN detected an error.																													
			NotDetected	0	Not detected.																													
			Detected	1	Detected.																													
D	RW	GLITCHSLOWDOMAIN[i] (i=0..0) W1C			Slow domain glitch detector i detected an error.																													
			NotDetected	0	Not detected.																													
			Detected	1	Detected.																													
E-H	RW	GLITCHFASTDOMAIN[i] (i=0..3) W1C			Fast domain glitch detector i detected an error.																													
			NotDetected	0	Not detected.																													
			Detected	1	Detected.																													

7.7.6.6.8 ACTIVESHIELD.CHEN

Address offset: 0x404

Active shield detector channel enable register.

Pins reserved for the active shield channels must be configured before the channels can be used. Pins reserved for unused channels can be used as GPIO.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-D	RW	CH[i] (i=0..3)			Enable or disable active shield channel i.																														
			Disabled	0	Disable channel.																														
			Enabled	1	Enable channel.																														

7.7.6.6.9 PROTECT.DOMAIN[n].DBGEN.CTRL (n=0..0)

Address offset: 0x500 + (n × 0x20)

Control register for invasive (halting) debug enable for the local debug components within domain n.

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			D D D D D D D D D D D D D D D C C C C B A																															
Reset 0x00000010			0 1 0 0 0																															
ID	R/W	Field	Value ID	Value	Description																													
B	W1	LOCK W1S			Lock this register to prevent changes to the VALUE field until next reset.																													
			Disabled	0	Lock disabled.																													
			Enabled	1	Lock enabled.																													
C	RW	WRITEPROTECTION			The write protection must be cleared to allow updates to the VALUE field.																													
					The write protection is cleared by writing CLEAR in a separate write operation prior to updating the VALUE and LOCK fields.																													
					The write protection is automatically enabled after the corresponding change to the VALUE field.																													
			Disabled	0x0	Read: Write protection is disabled.																													
			Enabled	0x1	Read: Write protection is enabled.																													
			Clear	0xF	Write: Value to clear write protection.																													
D	W	KEY			Required write key for upper 16 bits. Must be included in all register write operations.																													
			KEY	0x50FA	Write key value.																													

7.7.6.6.12 PROTECT.DOMAIN[n].NIDEN.STATUS (n=0..0)

Address offset: 0x50C + (n × 0x20)

Status register for non-invasive debug enable for domain n.

Note: Unless cleared, the STATUS register will be cumulative. A field is cleared by writing '1' to it.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ERROR W1C			Error detection status.																														
			NoError	0	No error detected.																														
			Error	1	Error detected.																														

7.7.6.6.13 PROTECT.DOMAIN[n].SPIDEN.CTRL (n=0..0)

Address offset: 0x510 + (n × 0x20)

Control register for secure privileged invasive (halting) debug enable for the local debug components within domain n.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D										C	C	C	C		B	A	
Reset 0x00000010				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
ID	R/W	Field	Value ID	Value		Description																														
A	RW	VALUE				Set value of spiden signal.																														
			Low	0	Signal is logic 0.																															
			High	1	Signal is logic 1.																															
B	W1 W15	LOCK				Lock this register to prevent changes to the VALUE field until next reset.																														
			Disabled	0	Lock disabled.																															

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
Reset 0x00000010				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D	R/W	Field	Value ID	Value				Description																											
			Enabled	1				Lock enabled.																											
C	RW	WRITEPROTECTION				The write protection must be cleared to allow updates to the VALUE field.																													
				The write protection is cleared by writing CLEAR in a separate write operation prior to updating the VALUE and LOCK fields.																															
				The write protection is automatically enabled after the corresponding change to the VALUE field.																															
			Disabled	0x0				Read: Write protection is disabled.																											
			Enabled	0x1				Read: Write protection is enabled.																											
			Clear	0xF				Write: Value to clear write protection.																											
D	W	KEY				Required write key for upper 16 bits. Must be included in all register write operations.																													
			KEY	0x50FA				Write key value.																											

7.7.6.6.14 PROTECT.DOMAIN[n].SPIDEN.STATUS (n=0..0)

Address offset: 0x514 + (n × 0x20)

Status register for secure privileged invasive (halting) debug enable for domain n.

Note: Unless cleared, the STATUS register will be cumulative. A field is cleared by writing '1' to it.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ERROR						Error detection status.																											
		W1C																																	
			NoError	0				No error detected.																											
			Error	1				Error detected.																											

7.7.6.6.15 PROTECT.DOMAIN[n].SPNIDEN.CTRL (n=0..0)

Address offset: 0x518 + (n × 0x20)

Control register for secure privileged non-invasive debug enable for the local debug components within domain n.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D										C	C	C	C			B	A
Reset 0x00000010				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
ID	R/W	Field	Value ID	Value	Description																															
A	RW	VALUE			Set value of spniden signal.																															
			Low	0	Signal is logic 0.																															
			High	1	Signal is logic 1.																															
B	W1 W1S	LOCK			Lock this register to prevent changes to the VALUE field until next reset.																															
			Disabled	0	Lock disabled.																															
			Enabled	1	Lock enabled.																															

Bit number			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID			D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D										C	C	C	C			B	A
Reset 0x00000010			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
ID	R/W	Field	Value ID	Value	Description																														
C	RW	WRITEPROTECTION			The write protection must be cleared to allow updates to the VALUE field.																														
					The write protection is cleared by writing CLEAR in a separate write operation prior to updating the VALUE and LOCK fields.																														
					The write protection is automatically enabled after the corresponding change to the VALUE field.																														
			Disabled	0x0	Read: Write protection is disabled.																														
			Enabled	0x1	Read: Write protection is enabled.																														
			Clear	0xF	Write: Value to clear write protection.																														
D	W	KEY			Required write key for upper 16 bits. Must be included in all register write operations.																														
			KEY	0x50FA	Write key value.																														

7.7.6.6.18 PROTECT.AP[n].DBGEN.STATUS (n=0..0)

Address offset: 0x704 + (n × 0x10)

Status register for invasive (halting) debug enable for domain n's access port.

Note: Unless cleared, the STATUS register will be cumulative. A field is cleared by writing '1' to it.

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID										A																																			
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field		Value ID		Value		Description																																					
A	RW	ERROR						Error detection status.																																					
		W1C																																											
				NoError		0		No error detected.																																					
				Error		1		Error detected.																																					

7.7.6.6.19 PROTECT.ACTIVESHIELD

Enable active shield detector.

The active shield pins are dedicated GPIO pins that must be configured as inputs and outputs before use. Each active shield channel has one GPIO for output and one GPIO for input.

7.7.6.6.19.1 PROTECT.ACTIVESHIELD.CTRL

Address offset: 0x900

Control register for active shield detector enable signal.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	C	C	C	C			B	A
Reset 0x00000010				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
ID	R/W	Field	Value ID	Value	Description																														
A	RW	VALUE			Set value of active shield enable signal.																														
			Low	0	Signal is logic 0.																														
			High	1	Signal is logic 1.																														
B	W1	LOCK			Lock this register to prevent changes to the VALUE field until next reset.																														
	W1S																																		

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	C	C	C	C			B	A		
Reset 0x00000010				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
			Disabled	0		Lock disabled.																																
			Enabled	1		Lock enabled.																																
C	RW	WRITEPROTECTION				The write protection must be cleared to allow updates to the VALUE field.																																
				The write protection is cleared by writing CLEAR in a separate write operation prior to updating the VALUE and LOCK fields.																																		
				The write protection is automatically enabled after the corresponding change to the VALUE field.																																		
			Disabled	0x0		Read: Write protection is disabled.																																
			Enabled	0x1		Read: Write protection is enabled.																																
			Clear	0xF		Write: Value to clear write protection.																																
D	W	KEY				Required write key for upper 16 bits. Must be included in all register write operations.																																
			KEY	0x50FA		Write key value.																																

7.7.6.6.19.2 PROTECT.ACTIVESHIELD.STATUS

Address offset: 0x904

Status register for active shield detector enable signal.

Note: Unless cleared, the STATUS register will be cumulative. A field is cleared by writing '1' to it.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				A																																			
Reset 0x00000000				0 0																																			
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	ERROR				Error detection status.																																	
		W1C																																					
			NoError	0		No error detected.																																	
			Error	1		Error detected.																																	

7.7.6.6.20 PROTECT.CRACENTAMP

Enable tamper detector from CRACEN.

Note: Disabling this bit only disables the TAMPC handling of the CRACENTAMP event, it does not disable the CRACEN from generating the CRACENTAMP event.

7.7.6.6.20.1 PROTECT.CRACENTAMP.CTRL

Address offset: 0x938

Control register for CRACEN tamper detector enable signal.

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID		D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D									C	C	C	C		B	A	
Reset 0x00000011		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
ID	R/W	Field	Value ID	Value	Description																												
A	RW	VALUE			Set value of CRACEN tamper detector enable signal.																												
			Low	0	Signal is logic 0.																												

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D										C	C	C	C			B	A
Reset 0x00000011				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
ID	R/W	Field	Value ID	Value	Description																															
			High	1	Signal is logic 1.																															
B	W1 W1S	LOCK			Lock this register to prevent changes to the VALUE field until next reset.																															
			Disabled	0	Lock disabled.																															
			Enabled	1	Lock enabled.																															
C	RW	WRITEPROTECTION			The write protection must be cleared to allow updates to the VALUE field.																															
					The write protection is cleared by writing CLEAR in a separate write operation prior to updating the VALUE and LOCK fields.																															
					The write protection is automatically enabled after the corresponding change to the VALUE field.																															
			Disabled	0x0	Read: Write protection is disabled.																															
			Enabled	0x1	Read: Write protection is enabled.																															
		Clear	0xF	Write: Value to clear write protection.																																
D	W	KEY			Required write key for upper 16 bits. Must be included in all register write operations.																															
			KEY	0x50FA	Write key value.																															

7.7.6.6.20.2 PROTECT.CRACENTAMP.STATUS

Address offset: 0x93C

Status register for CRACEN tamper detector enable signal.

Note: Unless cleared, the STATUS register will be cumulative. A field is cleared by writing '1' to it.

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID										A																																		
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																							
A	RW	ERROR W1C			Error detection status.																																							
			NoError	0	No error detected.																																							
			Error	1	Error detected.																																							

7.7.6.6.21 PROTECT.GLITCHSLOWDOMAIN

Enable slow domain glitch detectors.

7.7.6.6.21.1 PROTECT.GLITCHSLOWDOMAIN.CTRL

Address offset: 0x940

Control register for slow domain glitch detectors enable signal.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	C	C	C	C			B	A	
Reset 0x00000011				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
ID	R/W	Field	Value ID	Value	Description																															
A	RW	VALUE			Set value of slow domain glitch detectors enable signal.																															
			Low	0	Signal is logic 0.																															
			High	1	Signal is logic 1.																															

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			D D D D D D D D D D D D D D D C C C C B A																															
Reset 0x00000011			0 1 0 0 0 1																															
ID	R/W	Field	Value ID	Value	Description																													
B	W1	LOCK W1S			Lock this register to prevent changes to the VALUE field until next reset.																													
			Disabled	0	Lock disabled.																													
			Enabled	1	Lock enabled.																													
C	RW	WRITEPROTECTION			The write protection must be cleared to allow updates to the VALUE field.																													
					The write protection is cleared by writing CLEAR in a separate write operation prior to updating the VALUE and LOCK fields.																													
					The write protection is automatically enabled after the corresponding change to the VALUE field.																													
			Disabled	0x0	Read: Write protection is disabled.																													
			Enabled	0x1	Read: Write protection is enabled.																													
			Clear	0xF	Write: Value to clear write protection.																													
D	W	KEY			Required write key for upper 16 bits. Must be included in all register write operations.																													
			KEY	0x50FA	Write key value.																													

7.7.6.6.21.2 PROTECT.GLITCHSLOWDOMAIN.STATUS

Address offset: 0x944

Status register for slow domain glitch detectors enable signal.

Note: Unless cleared, the STATUS register will be cumulative. A field is cleared by writing '1' to it.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				A																																	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value		Description																															
A	RW	ERROR W1C				Error detection status.																															
			NoError	0	No error detected.																																
			Error	1	Error detected.																																

7.7.6.6.22 PROTECT.GLITCHFASTDOMAIN

Enable fast domain glitch detectors.

7.7.6.6.22.1 PROTECT.GLITCHFASTDOMAIN.CTRL

Address offset: 0x948

Control register for fast domain glitch detectors enable signal.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D										C	C	C	C		B	A	
Reset 0x00000011				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
ID	R/W	Field	Value ID	Value	Description																															
A	RW	VALUE			Set value of fast domain glitch detector's enable signal.																															
			Low	0	Signal is logic 0.																															
			High	1	Signal is logic 1.																															

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			D D D D D D D D D D D D D D D C C C C B A																															
Reset 0x00000011			0 1 0 0 0 1																															
ID	R/W	Field	Value ID	Value	Description																													
B	W1	LOCK W1S			Lock this register to prevent changes to the VALUE field until next reset.																													
			Disabled	0	Lock disabled.																													
			Enabled	1	Lock enabled.																													
C	RW	WRITEPROTECTION			The write protection must be cleared to allow updates to the VALUE field.																													
					The write protection is cleared by writing CLEAR in a separate write operation prior to updating the VALUE and LOCK fields.																													
					The write protection is automatically enabled after the corresponding change to the VALUE field.																													
			Disabled	0x0	Read: Write protection is disabled.																													
			Enabled	0x1	Read: Write protection is enabled.																													
			Clear	0xF	Write: Value to clear write protection.																													
D	W	KEY			Required write key for upper 16 bits. Must be included in all register write operations.																													
			KEY	0x50FA	Write key value.																													

7.7.6.6.22.2 PROTECT.GLITCHFASTDOMAIN.STATUS

Address offset: 0x94C

Status register for fast domain glitch detectors enable signal.

Note: Unless cleared, the STATUS register will be cumulative. A field is cleared by writing '1' to it.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				A																																
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																														
A	RW	ERROR W1C				Error detection status.																														
			NoError	0	No error detected.																															
			Error	1	Error detected.																															

7.7.6.6.23 PROTECT.EXTRESETEN

Trigger a reset when tamper is detected by the external tamper detectors.

This reset gives reset reason *SECTAMPER*

7.7.6.6.23.1 PROTECT.EXTRESETEN.CTRL

Address offset: 0x970

Control register for external tamper reset enable signal.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D										C	C	C	C		B	A
Reset 0x00000010				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	VALUE						Set value of external tamper reset enable signal.																											
			Low	0				Signal is logic 0.																											

Bit number			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID			D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	C	C	C	C	B	A		
Reset 0x00000010			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
ID	R/W	Field	Value ID	Value	Description																													
B	W1	LOCK W1S	High	1	Signal is logic 1.																													
			Disabled	0	Lock disabled.																													
			Enabled	1	Lock enabled.																													
					Lock this register to prevent changes to the VALUE field until next reset.																													
C	RW	WRITEPROTECTION			The write protection must be cleared to allow updates to the VALUE field.																													
					The write protection is cleared by writing CLEAR in a separate write operation prior to updating the VALUE and LOCK fields.																													
					The write protection is automatically enabled after the corresponding change to the VALUE field.																													
			Disabled	0x0	Read: Write protection is disabled.																													
			Enabled	0x1	Read: Write protection is enabled.																													
			Clear	0xF	Write: Value to clear write protection.																													
D	W	KEY			Required write key for upper 16 bits. Must be included in all register write operations.																													
			KEY	0x50FA	Write key value.																													

7.7.6.6.23.2 PROTECT.EXTRESETEEN.STATUS

Address offset: 0x974

Status register for external tamper reset enable signal.

Note: Unless cleared, the STATUS register will be cumulative. A field is cleared by writing '1' to it.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ERROR						Error detection status.																											
		W1C																																	
			NoError	0				No error detected.																											
			Error	1				Error detected.																											

7.7.6.6.24 PROTECT.INTRESETEEN

Trigger a reset when tamper is detected by the glitch detectors, signal protector or CRACEN tamper detector.

This reset gives reset reason *SECTAMPER*

7.7.6.6.24.1 PROTECT.INTRESETEEN.CTRL

Address offset: 0x978

Control register for internal tamper reset enable signal.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D D D D D D D D D D D D D D D D																								C C C C				B A			
Reset 0x00000011				0 1 0 0 0 1																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	VALUE			Set value of internal tamper reset enable signal.																														
			Low	0	Signal is logic 0.																														
			High	1	Signal is logic 1.																														
B	W1 W1S	LOCK			Lock this register to prevent changes to the VALUE field until next reset.																														
			Disabled	0	Lock disabled.																														
			Enabled	1	Lock enabled.																														
C	RW	WRITEPROTECTION			The write protection must be cleared to allow updates to the VALUE field.																														
					The write protection is cleared by writing CLEAR in a separate write operation prior to updating the VALUE and LOCK fields.																														
					The write protection is automatically enabled after the corresponding change to the VALUE field.																														
			Disabled	0x0	Read: Write protection is disabled.																														
			Enabled	0x1	Read: Write protection is enabled.																														
	Clear	0xF	Write: Value to clear write protection.																																
D	W	KEY			Required write key for upper 16 bits. Must be included in all register write operations.																														
			KEY	0x50FA	Write key value.																														

7.7.6.6.24.2 PROTECT.INTRESETEN.STATUS

Address offset: 0x97C

Status register for internal tamper reset enable signal.

Note: Unless cleared, the STATUS register will be cumulative. A field is cleared by writing '1' to it.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ERROR W1C			Error detection status.																														
			NoError	0	No error detected.																														
			Error	1	Error detected.																														

7.7.6.6.25 PROTECT.ERASEPROTECT

Device erase protection.

7.7.6.6.25.1 PROTECT.ERASEPROTECT.CTRL

Address offset: 0x980

Control register for erase protection.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D D D D D D D D D D D D D D D D																C C C C B A															
Reset 0x00000010				0 1 0 0 0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	VALUE						Set value of eraseprotect signal.																											

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			D D D D D D D D D D D D D D D C C C C B A																															
Reset 0x00000010			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
B	W1	LOCK	Low	0	Signal is logic 0.																													
			High	1	Signal is logic 1.																													
			Disabled	0	Lock disabled.																													
			Enabled	1	Lock enabled.																													
C	RW	WRITEPROTECTION			The write protection must be cleared to allow updates to the VALUE field.																													
					The write protection is cleared by writing CLEAR in a separate write operation prior to updating the VALUE and LOCK fields.																													
					The write protection is automatically enabled after the corresponding change to the VALUE field.																													
			Disabled	0x0	Read: Write protection is disabled.																													
D	W	KEY	Enabled	0x1	Read: Write protection is enabled.																													
			Clear	0xF	Write: Value to clear write protection.																													
					Required write key for upper 16 bits. Must be included in all register write operations.																													
			KEY	0x50FA	Write key value.																													

7.7.6.6.25.2 PROTECT.ERASEPROTECT.STATUS

Address offset: 0x984

Status register for eraseprotect.

Note: Unless cleared, the STATUS register will be cumulative. A field is cleared by writing '1' to it.

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0					
ID										A																																				
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																									
A	RW	ERROR			Error detection status.																																									
		W1C																																												
			NoError	0	No error detected.																																									
			Error	1	Error detected.																																									

8 Peripherals

The device features a rich set of peripherals. The following sections describe the peripherals and how they are used.

8.1 Peripheral interface

Peripherals are controlled by the CPU through configuration, task, and event registers. Task registers are inputs, enabling the CPU and other peripherals to initiate a functionality. Event registers are outputs, enabling a peripheral to trigger tasks in other peripherals or the CPU by tying events to CPU interrupts.

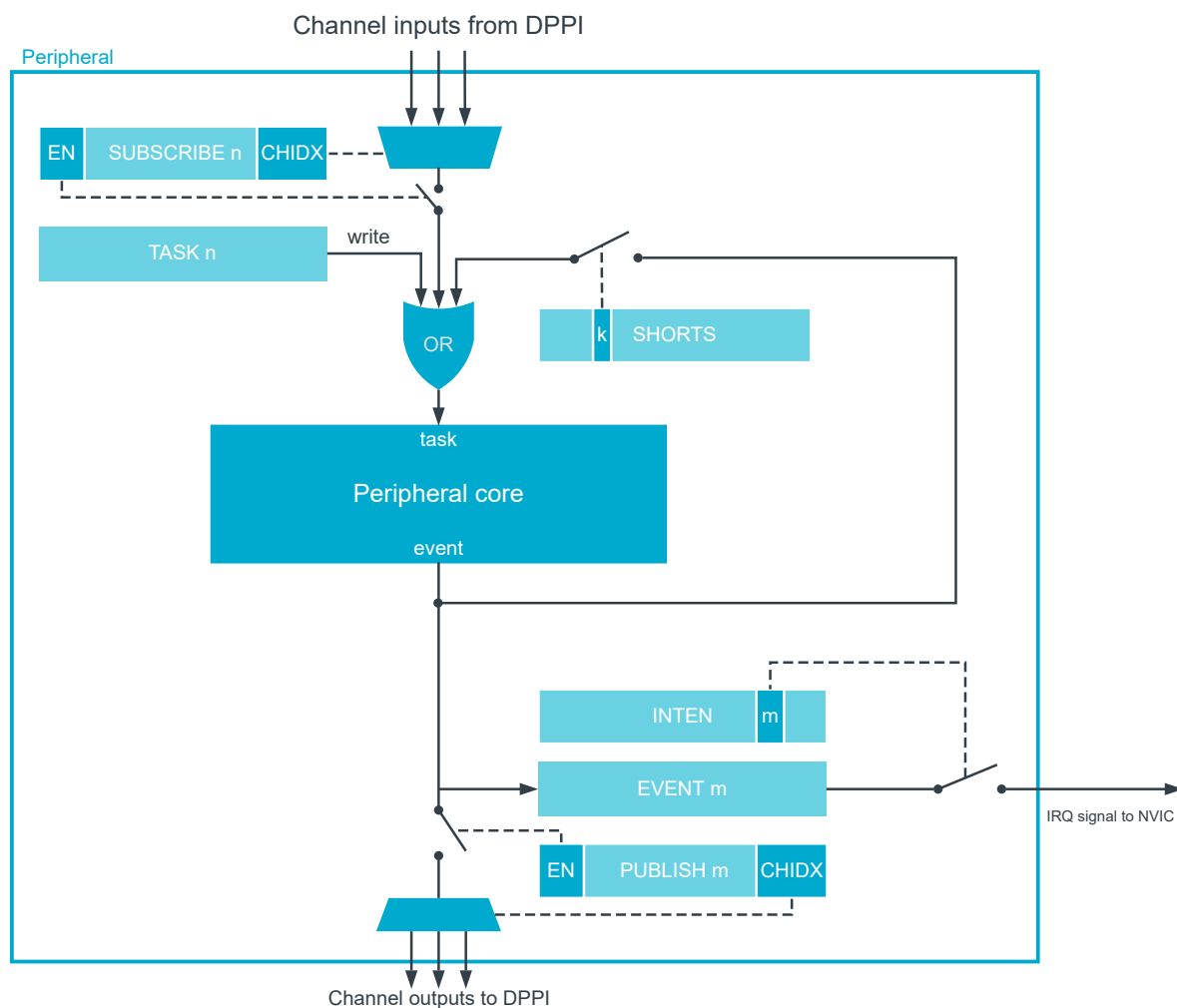


Figure 36: Peripheral interface

The distributed programmable peripheral interconnect (DPPI) feature enables peripherals to connect events to tasks without CPU intervention. For more information on DPPI and the DPPI channels, see [DPPI — Distributed programmable peripheral interconnect](#) on page 112.

8.1.2 Peripheral ID

Each peripheral is assigned a fixed block of address space that is minimum 4 KB in size and has at least 1024 registers of 32 bits.

For more information on available peripherals and their location in the address map, see [Instantiation](#) on page 216.

There is a direct relationship between peripheral ID and base address:

```
base_address = 0x40000000 + 0x1000 * ID
```

Example peripheral base addresses:

- 0x40000000 is assigned ID=0
- 0x40001000 is assigned ID=1
- 0x4001F000 is assigned ID=31

Peripherals can share the same ID, which has the following limitations:

- Shared registers or common resources
- Limited availability due to mutually exclusive operation; only one peripheral in use at a time
- Enforced peripheral behavior when switching between peripherals (disable the first peripheral before enabling the second)

8.1.3 Peripherals with shared ID

Peripherals sharing ID [1...n] and a base address may not be used simultaneously. Only one peripheral can be enabled at a given ID. Peripherals using ID 0 can be enabled simultaneously.

When switching between two peripherals sharing an ID, perform the following to prevent unwanted behavior.

1. Disable the previously used peripheral.
2. Disable any publish/subscribe connection to the DPPI system for the peripheral that is being disabled.
3. Clear all bits in the INTEN register (INTENCLR = 0xFFFFFFFF).
4. Configure the peripheral being enabled. Do not rely on the inherited configuration from the disabled peripheral.
5. Enable the peripheral.

For a list of peripherals that share an ID, see [Instantiation](#) on page 216.

8.1.4 Peripheral registers

Most peripherals have an ENABLE register. Unless otherwise specified, the peripheral registers must be configured before enabling the peripheral.

PSEL registers must be set before a peripheral is enabled or started. Updating PSEL registers while the peripheral is running can cause undefined behavior. To connect a peripheral to a different GPIO, the following must be performed:

1. Disable the peripheral.
2. Update the PSEL register.
3. Re-enable the peripheral.

Note: The peripheral must be enabled before tasks and events can be used.

Most of the register values are not retained during System OFF or when a reset is triggered. Some registers will retain their values in System OFF or for some specific reset sources. These registers are marked as

retained in the register description for a given peripheral. For more information on their behavior, see chapter [RESET — Reset control](#) on page 106.

8.1.5 Bit set and clear

Registers with multiple single-bit fields can implement the set-and-clear bit pattern. This bit pattern enables firmware to set and clear individual bits in a register without having to perform a read-modify-write operation to the main register.

This bit pattern is implemented using three consecutive addresses in the register map, where the main register is followed by dedicated SET and CLR registers (in that exact order).

In the main register, the SET register sets individual bits and the CLR register clears them. Writing 1 to a bit in the SET or CLR register will set or clear the same bit in the main register. Writing 0 to a bit in the SET or CLR register has no effect. Reading the SET or CLR register returns the value of the main register.

Note: The main register may not be visible, and therefore not directly accessible in all cases.

8.1.6 Tasks

Tasks trigger actions in a peripheral, such as to start a particular behavior. A peripheral can implement multiple tasks, with each task having a separate register in that peripheral's task register group.

A task is triggered when firmware writes 1 to the task register, or when the peripheral itself or another peripheral toggles the corresponding task signal. See the figure [Peripheral interface](#) on page 213.

8.1.7 Events

Events notify peripherals and the CPU about events that have happened, such as a state change in a peripheral. A peripheral may generate multiple events, where each event has a separate register in that peripheral's event register group.

An event is generated when the peripheral toggles the corresponding event signal and updates the event register to show an event has been generated, see figure [Peripheral interface](#) on page 213. An event register is cleared when firmware writes a 0 to that register. A peripheral can continually generate events when the event register is 1.

8.1.8 Publish and subscribe

Events and tasks from different peripherals can be connected together through the DPPI system using the PUBLISH and SUBSCRIBE registers in each peripheral. See [Peripheral interface](#) on page 213.

An event can be published onto a DPPI channel by configuring the event's PUBLISH register. Similarly, a task can subscribe to a DPPI channel by configuring the task's SUBSCRIBE register.

See [DPPI — Distributed programmable peripheral interconnect](#) on page 112 for details.

8.1.9 Shortcuts

A shortcut is a direct connection between an event and a task within the same peripheral. If a shortcut is enabled, the associated task is automatically triggered when its associated event is generated.

Using shortcuts is the same as connecting a task and event outside the peripheral through the DPPI. The propagation delay for a shortcut is usually shorter than the propagation delay through the DPPI.

Shortcuts are predefined, which means that their connections cannot be configured by firmware. Each shortcut can be individually enabled or disabled through the shortcut register, one bit per shortcut, giving a maximum of 32 shortcuts for each peripheral.

8.1.10 Interrupts

All peripherals support interrupts generated by events.

A peripheral can occupy single or multiple interrupts. For single interrupts, the interrupt number follows the peripheral ID. For example, the peripheral with ID=4 is connected to interrupt number 4 in the nested vectored interrupt controller (NVIC). In this case, only single INTEN registers are available.

Events generated by a peripheral can be configured to generate interrupts using registers INTEN, INTENSET, and INTENCLR. Multiple events can be enabled to generate interrupts simultaneously. Event registers in the peripheral register event group indicate the source.

Some peripherals implement only INTENSET and INTENCLR registers. The INTEN register is not available on those peripherals. See the individual peripheral chapters for details. In all cases, reading back the INTENSET or INTENCLR register returns the same information as INTEN.

The INTPEND register contains the interrupt pending status of events generated by a peripheral. This is a read-only register.

Peripherals implementing multiple interrupts have several INTEN registers that follow the convention of INTEN_n, where *n* is the interrupt number from the peripheral. This also applies to corresponding INTPEND, INTENSET, and INTENCLR registers. This feature enables any event to generate an interrupt from the peripheral.

Peripherals implementing more than 32 events have access to multiple INTEN registers that follow the convention of INTEN_n, where *n* is the event group number. The 32 lowest events in the peripheral make event group 0. The next 32 events in the peripheral make event group 1, and so on. This convention is also applicable for corresponding INTPEND, INTENSET, and INTENCLR registers.

Peripherals implementing both multiple interrupts and more than 32 events have multiple INTEN registers. In this case, registers follow the convention of INTEN_{nm}, where *n* is interrupt number from the peripheral and *m* is event group number. This convention is also applicable for corresponding INTPEND, INTENSET, and INTENCLR registers.

Each event implemented in the peripheral is associated with a specific bit position in the INTEN, INTENSET, and INTENCLR registers.

To ensure the lowest possible power consumption while in sleep, perform either of the following steps on any pending interrupts:

- Clear the pending interrupt by writing 0 to the corresponding EVENT register
- Disable the interrupt by using the INTEN or INTENCLR registers

This has to be done even if the peripheral is disabled in its ENABLE or POWER register.

The relationship between tasks, events, shortcuts, and interrupts is illustrated in [Peripheral interface](#) on page 213.

8.1.10.1 Interrupt clearing and disabling

Interrupts must be cleared by writing 0 to the corresponding EVENT register.

Interrupts are immediately re-triggered until cleared. Routines for software interrupt services continue to execute, even if a new event has not been received.

8.2 Instantiation

ID	Base address	Instance	TrustZone			Split access	Description
			Map	Att	DMA		
64	0x50040000	SPU00	HF	S	NA	No	System protection unit SPU00

ID	Base address	Instance	TrustZone			Split access	Description
			Map	Att	DMA		
65	0x50041000	MPC00	HF	S	NA	No	Memory privilege controller MPC00
66	0x50042000	DPPIC00 : S	US	S	NA	Yes	DPPI controller DPPIC00
	0x40042000	DPPIC00 : NS					
67	0x50043000	PPIB00 : S	US	S	NA	No	PPI bridge PPIB00
	0x40043000	PPIB00 : NS					
68	0x50044000	PPIB01 : S	US	S	NA	No	PPI bridge PPIB01
	0x40044000	PPIB01 : NS					
69	0x50045000	KMU	HF	S	NSA	No	Key management unit
70	0x50046000	AAR00 : S	US	S	SA	No	Accelerated address resolver 00
	0x40046000	AAR00 : NS					
70	0x50046000	CCM00 : S	US	S	SA	No	AES CCM mode encryption CCM00, running of HCLK128M
	0x40046000	CCM00 : NS					
71	0x50047000	ECB00 : S	US	S	SA	No	When configuring this peripheral's DMA security using SPU configuration (DMASEC field of SPU->PERIPH[apb_slave_index]), use apb_slave_index 6 (same as AAR00 and CCM00)
	0x40047000	ECB00 : NS					
72	0x50048000	CRACEN	HF	S	NSA	No	Crypto accelerator
74	0x5004A000	SPIM00 : S	US	S	SA	No	SPI controller SPIM00
	0x4004A000	SPIM00 : NS					
74	0x5004A000	SPIS00 : S	US	S	SA	No	SPI peripheral SPIS00
	0x4004A000	SPIS00 : NS					
74	0x5004A000	UARTE00 : S	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE00
	0x4004A000	UARTE00 : NS					
75	0x5004B000	GLITCHDET	HF	S	NA	No	Glitch detectors
75	0x5004B000	RRAMC	HF	S	NA	No	RRAM Non-Volatile Memory Controller
76	0x5004C000	VPR00 : S	US	NS	NSA	No	FLPR - VPR peripheral registers
	0x4004C000	VPR00 : NS					
80	0x50050400	P2 : S	US	S	NA	Yes	General purpose input and output, port P2 Does not support pin sense mechanism, and DETECTMODE register has no effect. Supports extra high drive (DRIVE0=E0, DRIVE1=E1).
	0x40050400	P2 : NS					
82	0x50052000	CTRLAP : S	US	S	NSA	No	Control access port CPU side
	0x40052000	CTRLAP : NS					
83	0x50053000	TAD : S	US	S	NA	No	Empty instance abstract
	0x40053000	TAD : NS					
85	0x50055000	TIMER00 : S	US	S	NA	No	Timer TIMER00
	0x40055000	TIMER00 : NS					
128	0x50080000	SPU10	HF	S	NA	No	System protection unit SPU10
130	0x50082000	DPPIC10 : S	US	S	NA	Yes	DPPI controller DPPIC10
	0x40082000	DPPIC10 : NS					
131	0x50083000	PPIB10 : S	US	S	NA	No	PPI bridge PPIB10
	0x40083000	PPIB10 : NS					
132	0x50084000	PPIB11 : S	US	S	NA	No	PPI bridge PPIB11
	0x40084000	PPIB11 : NS					
133	0x50085000	TIMER10 : S	US	S	NA	No	Timer TIMER10
	0x40085000	TIMER10 : NS					
135	0x50087000	EGU10 : S	US	S	NA	No	Event generator unit EGU10
	0x40087000	EGU10 : NS					
138	0x5008A000	RADIO : S	US	S	SA	No	2.4 GHz radio RADIO
	0x4008A000	RADIO : NS					
192	0x500C0000	SPU20	HF	S	NA	No	System protection unit SPU20

ID	Base address	Instance	TrustZone			Split access	Description
			Map	Att	DMA		
194	0x500C2000	DPPIC20 : S	US	S	NA	Yes	DPPI controller DPPIC20
	0x400C2000	DPPIC20 : NS					
195	0x500C3000	PPIB20 : S	US	S	NA	No	PPI bridge PPIB20
	0x400C3000	PPIB20 : NS					
196	0x500C4000	PPIB21 : S	US	S	NA	No	PPI bridge PPIB21
	0x400C4000	PPIB21 : NS					
197	0x500C5000	PPIB22 : S	US	S	NA	No	PPI bridge PPIB22
	0x400C5000	PPIB22 : NS					
198	0x500C6000	SPIM20 : S	US	S	SA	No	SPI controller SPIM20
	0x400C6000	SPIM20 : NS					
198	0x500C6000	SPIS20 : S	US	S	SA	No	SPI peripheral SPIS20
	0x400C6000	SPIS20 : NS					
198	0x500C6000	TWIM20 : S	US	S	SA	No	Two-wire interface controller TWIM20
	0x400C6000	TWIM20 : NS					
198	0x500C6000	TWIS20 : S	US	S	SA	No	Two-wire interface target TWIS20
	0x400C6000	TWIS20 : NS					
198	0x500C6000	UARTE20 : S	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE20
	0x400C6000	UARTE20 : NS					
199	0x500C7000	SPIM21 : S	US	S	SA	No	SPI controller SPIM21
	0x400C7000	SPIM21 : NS					
199	0x500C7000	SPIS21 : S	US	S	SA	No	SPI peripheral SPIS21
	0x400C7000	SPIS21 : NS					
199	0x500C7000	TWIM21 : S	US	S	SA	No	Two-wire interface controller TWIM21
	0x400C7000	TWIM21 : NS					
199	0x500C7000	TWIS21 : S	US	S	SA	No	Two-wire interface target TWIS21
	0x400C7000	TWIS21 : NS					
199	0x500C7000	UARTE21 : S	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE21
	0x400C7000	UARTE21 : NS					
200	0x500C8000	SPIM22 : S	US	S	SA	No	SPI controller SPIM22
	0x400C8000	SPIM22 : NS					
200	0x500C8000	SPIS22 : S	US	S	SA	No	SPI peripheral SPIS22
	0x400C8000	SPIS22 : NS					
200	0x500C8000	TWIM22 : S	US	S	SA	No	Two-wire interface controller TWIM22
	0x400C8000	TWIM22 : NS					
200	0x500C8000	TWIS22 : S	US	S	SA	No	Two-wire interface target TWIS22
	0x400C8000	TWIS22 : NS					
200	0x500C8000	UARTE22 : S	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE22
	0x400C8000	UARTE22 : NS					
201	0x500C9000	EGU20 : S	US	S	NA	No	Event generator unit EGU20
	0x400C9000	EGU20 : NS					
202	0x500CA000	TIMER20 : S	US	S	NA	No	Timer TIMER20
	0x400CA000	TIMER20 : NS					
203	0x500CB000	TIMER21 : S	US	S	NA	No	Timer TIMER21
	0x400CB000	TIMER21 : NS					
204	0x500CC000	TIMER22 : S	US	S	NA	No	Timer TIMER22
	0x400CC000	TIMER22 : NS					
205	0x500CD000	TIMER23 : S	US	S	NA	No	Timer TIMER23
	0x400CD000	TIMER23 : NS					
206	0x500CE000	TIMER24 : S	US	S	NA	No	Timer TIMER24
	0x400CE000	TIMER24 : NS					
207	0x500CF000	MEMCONF : S	US	S	NA	No	Memory Configuration MEMCONF
	0x400CF000	MEMCONF : NS					

ID	Base address	Instance	TrustZone			Split access	Description
			Map	Att	DMA		
208	0x500D0000	PDM20 : S	US	S	SA	No	Pulse density modulation (digital microphone) interface PDM20
	0x400D0000	PDM20 : NS					
209	0x500D1000	PDM21 : S	US	S	SA	No	Pulse density modulation (digital microphone) interface PDM21
	0x400D1000	PDM21 : NS					
210	0x500D2000	PWM20 : S	US	S	SA	No	Pulse width modulation unit PWM20
	0x400D2000	PWM20 : NS					
211	0x500D3000	PWM21 : S	US	S	SA	No	Pulse width modulation unit PWM21
	0x400D3000	PWM21 : NS					
212	0x500D4000	PWM22 : S	US	S	SA	No	Pulse width modulation unit PWM22
	0x400D4000	PWM22 : NS					
213	0x500D5000	SAADC : S	US	S	SA	No	Successive approximation analog-to-digital converter SAADC
	0x400D5000	SAADC : NS					
214	0x500D6000	NFCT : S	US	S	SA	No	Near field communication tag NFCT
	0x400D6000	NFCT : NS					
215	0x500D7000	TEMP : S	US	S	NA	No	Temperature sensor TEMP
	0x400D7000	TEMP : NS					
216	0x500D8200	P1 : S	US	S	NA	Yes	General purpose input and output, port P1
	0x400D8200	P1 : NS					
218	0x500DA000	GPIOTE20 : S	US	S	NA	Yes	8 channels and 2 interrupts for GPIO port P1 GPIO tasks and events GPIOTE20
	0x400DA000	GPIOTE20 : NS					
220	0x500DC000	TAMPC	HF	S	NA	No	Tamper controller TAMPC
221	0x500DD000	I2S20 : S	US	S	SA	No	Inter-IC sound interface I2S20
	0x400DD000	I2S20 : NS					
224	0x500E0000	QDEC20 : S	US	S	NA	No	Quadrature decoder QDEC20
	0x400E0000	QDEC20 : NS					
225	0x500E1000	QDEC21 : S	US	S	NA	No	Quadrature decoder QDEC21
	0x400E1000	QDEC21 : NS					
226	0x500E2000	GRTC : S	US	S	NA	Yes	Global RTC GRTC
	0x400E2000	GRTC : NS					
256	0x50100000	SPU30	HF	S	NA	No	System protection unit SPU30
258	0x50102000	DPPIC30 : S	US	S	NA	Yes	DPPI controller DPPIC30
	0x40102000	DPPIC30 : NS					
259	0x50103000	PPIB30 : S	US	S	NA	No	PPI bridge PPIB30
	0x40103000	PPIB30 : NS					
260	0x50104000	SPIM30 : S	US	S	SA	No	SPI controller SPIM30
	0x40104000	SPIM30 : NS					
260	0x50104000	SPIS30 : S	US	S	SA	No	SPI peripheral SPIS30
	0x40104000	SPIS30 : NS					
260	0x50104000	TWIM30 : S	US	S	SA	No	Two-wire interface controller TWIM30
	0x40104000	TWIM30 : NS					
260	0x50104000	TWIS30 : S	US	S	SA	No	Two-wire interface target TWIS30
	0x40104000	TWIS30 : NS					
260	0x50104000	UARTE30 : S	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE30
	0x40104000	UARTE30 : NS					
262	0x50106000	COMP : S	US	S	NA	No	Comparator COMP
	0x40106000	COMP : NS					
262	0x50106000	LPCOMP : S	US	S	NA	No	Low-power comparator LPCOMP
	0x40106000	LPCOMP : NS					
264	0x50108000	WDT30	HF	S	NA	No	Watchdog timer WDT30
265	0x50109000	WDT31 : S	US	S	NA	No	Watchdog timer WDT31
	0x40109000	WDT31 : NS					

ID	Base address	Instance	TrustZone			Split access	Description
			Map	Att	DMA		
266	0x5010A000	PO : S	US	S	NA	Yes	General purpose input and output, port P0
	0x4010A000	PO : NS					
268	0x5010C000	GPIOTE30 : S	US	S	NA	Yes	4 channels and 2 interrupts for GPIO port P0 GPIO tasks and events GPIOTE30
	0x4010C000	GPIOTE30 : NS					
270	0x5010E000	CLOCK : S	US	S	NA	No	Clock control
	0x4010E000	CLOCK : NS					
270	0x5010E000	POWER : S	US	S	NA	No	Power control
	0x4010E000	POWER : NS					
270	0x5010E000	RESET : S	US	S	NA	No	Reset status
	0x4010E000	RESET : NS					
288	0x50120000	OSCILLATORS : S	US	S	NA	No	Oscillator control
	0x40120000	OSCILLATORS : NS					
288	0x50120000	REGULATORS : S	US	S	NA	No	Regulator control
	0x40120000	REGULATORS : NS					
N/A	0x00FFC000	FICR	HF	NS	NA	No	Factory information configuration
N/A	0x00FFD000	UICR	HF	S	NA	No	User information configuration
N/A	0x00FFE000	SICR	HF	S	NA	No	Secure information configuration region
N/A	0x51800000	CRACENCORE	HF	S	NSA	No	CRACEN core

Table 31: Instantiation table

8.3 AAR — Accelerated address resolver

Accelerated address resolver is a cryptographic support function for implementing the Resolvable Private Address Resolution Procedure described in the *Bluetooth Core specification*.

The main features of AAR are:

- Memory-to-memory operations using Scatter/Gather DMA
- Real-time address resolution on incoming packets
- Multiple IRK resolution

The procedure allows two devices that share a secret key to generate and resolve a hash based on their device address. AAR enables real-time address resolution on incoming packets when configured as described in this chapter. This allows real-time packet filtering (whitelisting) using a list of known shared keys (Identity Resolving Keys (IRK) in *Bluetooth*).

The inputs and outputs of AAR are illustrated in the following figure.



Figure 37: AAR block diagram

8.3.1 Shared resources

AAR shares the same AES module as the ECB and CCM peripherals. ECB will always have the lowest priority. If there is a sharing conflict during encryption, ECB operation will be aborted and an ERRORECB event will be generated by ECB.

Additionally, AAR shares registers and other resources with the peripherals that have the same ID as AAR. See [Peripherals with shared ID](#) on page 214 for more information.

8.3.2 Resolving a resolvable address

As per *Bluetooth* specification, a private resolvable address is composed of six bytes.

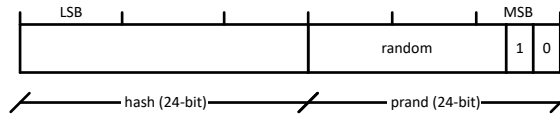


Figure 38: Resolvable address

To resolve an address, INPTR must point to a job list describing both the Hash and Prand parts of the private resolvable address (DEVICEADDR) field from the Bluetooth packet, as well as a number of Identity Resolving Keys (IRK). This is illustrated in the examples below. How many IRKs are used is determined by the number of IRKs in the job list. See [EasyDMA](#) on page 222 for an introduction to EasyDMA job lists.

The resolver is started by triggering the **START** task. A **RESOLVED** event is generated if AAR manages to resolve the address using one of the Identity Resolving Keys (IRK). AAR will generate a **NOTRESOLVED** event if it is not able to resolve the address using the specified list of IRKs. If there are no IRKs in the joblist, the **NOTRESOLVED** event is generated.

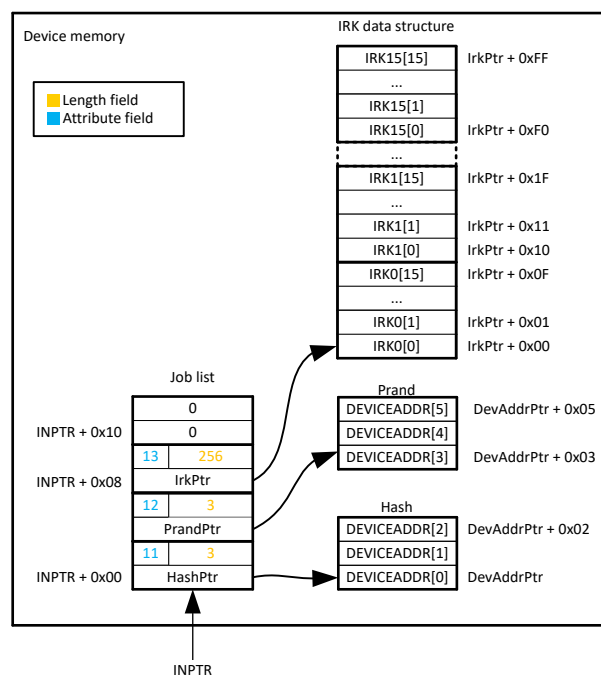


Figure 39: Address resolution with 16 IRKs and DEVICEADDR preloaded into RAM

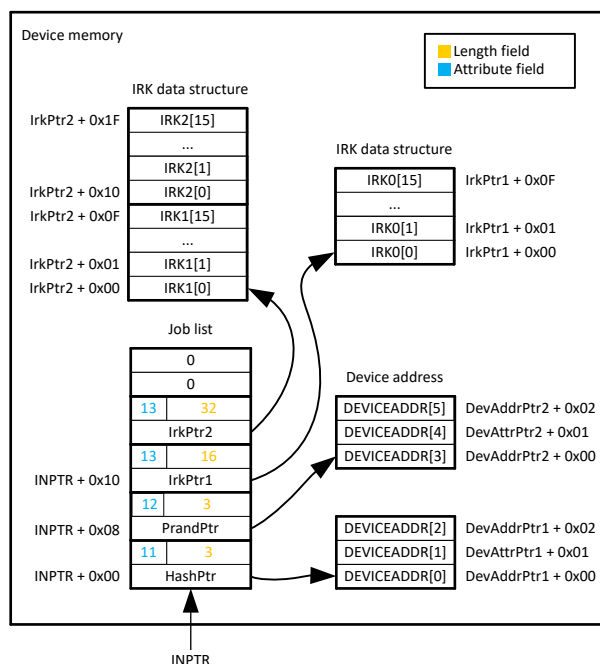


Figure 40: Address resolution with packet device address preloaded into multiple RAM locations, and three IRK keys

AAR will go through the list of available IRKs in the job list, and for each IRK try to resolve the address according to the Resolvable Private Address Resolution Procedure described in the *Bluetooth Specification*¹. The time it takes to resolve an address may vary depending on where in the list the resolvable address is located. The resolution time will also be affected by RAM accesses performed by other peripherals and the CPU. See the Electrical specifications for more information about resolution time.

AAR only compares the received address to those programmed in the module without checking the address type.

AAR will stop when it has managed to resolve the maximum number allowed, specified in the `MAXRESOLVED` register. Each time AAR resolves an IRK, the index of the corresponding IRK is written to memory through the output job list in `OUT.PTR`. For each IRK found, `OUT.AMOUNT` is updated accordingly.

At the end of the operation, AAR will generate the `END` event.

Triggering the `STOP` task will stop AAR. If AAR is stopped before the operation has completed, the `END`, `RESOLVED`, and `NOTRESOLVED` events are not generated. However, if `STOP` is triggered close to the end of the operation the events can be generated.

8.3.3 EasyDMA

This peripheral implements EasyDMA with scatter-gather functionality for reading from memory without CPU involvement.

The scatter-gather functionality allows EasyDMA to collect data from multiple memory regions, instead of one contiguous block. The memory regions are described by a job list. The job list consists of one or more job entries that consist of a 32-bit address field, 8-bit attribute field, and 24-bit length field. A job list ends with a zero filled job entry.

The job list must have separate entries for the following entries:

1. The three first bytes of the resolvable private address (the 24-bit hash)

¹ *Bluetooth Specification Version 4.0 [Vol 3] chapter 10.8.2.3.*

2. The three following bytes of the resolvable private address (the 24-bit prand)
3. The IRKs

The attribute field of each of these entries identify the job and must be set according to the following table.

Attribute	Value
Hash	11
Prand	12
Irk	13

Table 32: Attribute field

If the INPTR register or the entries in the job list are not pointing to memory connected to the DMA bus, an EasyDMA transfer may result in a HardFault or memory corruption. See [Memory](#) on page 19 for more information about the different memory regions and DMA connectivity.

The EasyDMA will have finished accessing the RAM when the [END](#), [RESOLVED](#), or [NOTRESOLVED](#) events are generated.

For instances supporting DMA error detection, the [ERRORSTATUS](#) register will report if a bus error has occurred during DMA access. To see if DMA error detection is supported, see the instance's configuration in [Instantiation](#) on page 216.

Example

The figure below shows an example of a job list with three job entries. Each of the entries point to a memory address, and the length field describes how many bytes of data is stored at that address. There are three blocks of memory in use

- Hash, an array of length 3
- Prand, an array of length 3
- Irk, an array of at least length 16

The data pointed to from the job list is what is fed into the module and processed according to the peripheral's operation. The entries of the job list comprises pointers to the individual arrays, as well as their sizes. Job entries with length greater than one are processed in little endian order.

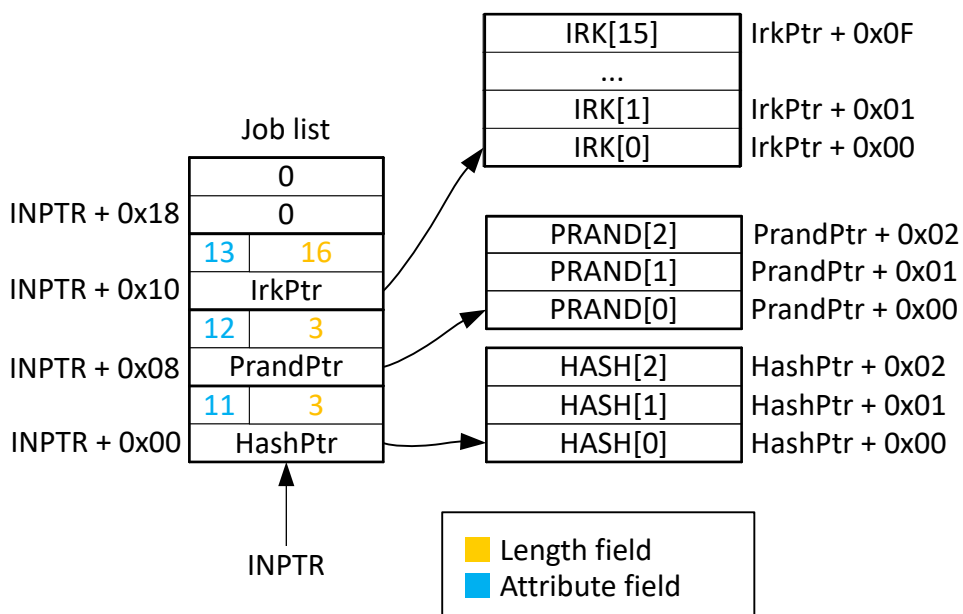


Figure 41: EasyDMA Scatter-Gather job list example

8.3.4 Use case example for chaining RADIO packet reception with address resolution using AAR

AAR may be started as soon as the 6 bytes required by AAR have been received by RADIO and stored in RAM. The Hash and Prand part of the job list must point to the part of the packet containing the device address.

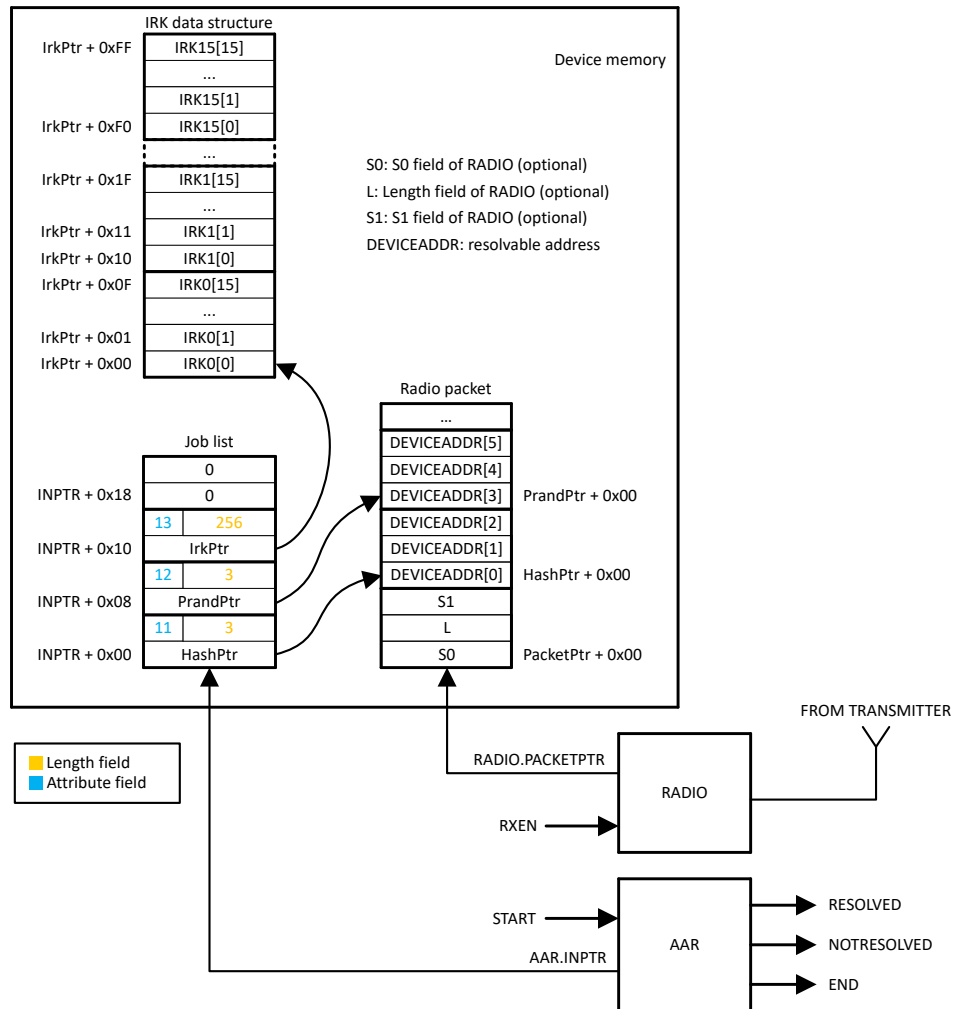


Figure 42: Address resolution with packet loaded into RAM by RADIO

8.3.5 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
AAR00 : S	GLOBAL	0x50046000	US	S	SA	No	Accelerated address resolver 00
AAR00 : NS		0x40046000					

Configuration

Instance	Domain	Configuration
AAR00 : S	GLOBAL	
AAR00 : NS		

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Start resolving addresses based on IRKs specified in the IRK data structure
TASKS_STOP	0x004		Stop resolving addresses
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
EVENTS_END	0x100		Address resolution procedure complete or ended due to an error
EVENTS_RESOLVED	0x104		Address resolved
EVENTS_NOTRESOLVED	0x108		Address not resolved
EVENTS_ERROR	0x10C		Operation aborted because of a STOP task or due to an error This event does not generate an interrupt
PUBLISH_END	0x180		Publish configuration for event END
PUBLISH_RESOLVED	0x184		Publish configuration for event RESOLVED
PUBLISH_NOTRESOLVED	0x188		Publish configuration for event NOTRESOLVED
PUBLISH_ERROR	0x18C		Publish configuration for event ERROR
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
ERRORSTATUS	0x404		Error status
ENABLE	0x500		Enable AAR
MAXRESOLVED	0x508		Maximum number of IRKs to resolve
IN.PTR	0x530		Input pointer
OUT.PTR	0x538		Output pointer
OUT.AMOUNT	0x53C		Number of bytes transferred in the last transaction

8.3.5.1 TASKS_START

Address offset: 0x000

Start resolving addresses based on IRKs specified in the IRK data structure

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																					A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																
A	W	TASKS_START			Start resolving addresses based on IRKs specified in the IRK data structure																																
			Trigger	1	Trigger task																																

8.3.5.2 TASKS_STOP

Address offset: 0x004

Stop resolving addresses

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	W	TASKS_STOP																		Stop resolving addresses															
			Trigger	1																Trigger task															

8.3.5.3 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task **START**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task START will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.3.5.4 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task **STOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task STOP will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.3.5.5 EVENTS_END

Address offset: 0x100

Address resolution procedure complete or ended due to an error

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_END			Address resolution procedure complete or ended due to an error																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.3.5.6 EVENTS_RESOLVED

Address offset: 0x104

Address resolved

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_RESOLVED			Address resolved																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.3.5.7 EVENTS_NOTRESOLVED

Address offset: 0x108

Address not resolved

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_NOTRESOLVED			Address not resolved																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.3.5.8 EVENTS_ERROR

Address offset: 0x10C

Operation aborted because of a STOP task or due to an error

This event does not generate an interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_ERROR			Operation aborted because of a STOP task or due to an error																														
					This event does not generate an interrupt																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.3.5.9 PUBLISH_END

Address offset: 0x180

Publish configuration for event [END](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
ID				B																								A A A A A A A A A A A A A A A A															
Reset 0x00000000				0 0																																							
ID	R/W	Field	Value ID	Value				Description																																			
A	RW	CHIDX		[0..255]				DPPI channel that event END will publish to																																			
B	RW	EN																																									
			Disabled	0				Disable publishing																																			
			Enabled	1				Enable publishing																																			

8.3.5.10 PUBLISH_RESOLVED

Address offset: 0x184

Publish configuration for event [RESOLVED](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0																																		
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that event RESOLVED will publish to																																
B	RW	EN																																				
			Disabled	0	Disable publishing																																	
			Enabled	1	Enable publishing																																	

8.3.5.11 PUBLISH_NOTRESOLVED

Address offset: 0x188

Publish configuration for event **NOTRESOLVED**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0																																		
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that event NOTRESOLVED will publish to																																
B	RW	EN																																				
			Disabled	0	Disable publishing																																	
			Enabled	1	Enable publishing																																	

8.3.5.12 PUBLISH_ERROR

Address offset: 0x18C

Publish configuration for event **ERROR**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				B																												A	A	A	A	A	A	A	A
Reset 0x00000000				0																																			
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	CHIDX		[0..255]		DPPI channel that event ERROR will publish to																																	
B	RW	EN																																					
			Disabled	0	Disable publishing																																		
			Enabled	1	Enable publishing																																		

8.3.5.13 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	END			Write '1' to enable interrupt for event END																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	RESOLVED			Write '1' to enable interrupt for event RESOLVED																														
			Set	1	Enable																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			C			RW	NOTRESOLVED	Write '1' to enable interrupt for event NOTRESOLVED																											
			Set		1			Enable																											
			Disabled		0			Read: Disabled																											
			Enabled		1			Read: Enabled																											
D	RW	ERROR	Write '1' to enable interrupt for event ERROR																																
			Set		1	Enable																													
			Disabled		0	Read: Disabled																													
			Enabled		1	Read: Enabled																													

8.3.5.14 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	END			Write '1' to disable interrupt for event END																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
B	RW	RESOLVED			Write '1' to disable interrupt for event RESOLVED																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
C	RW	NOTRESOLVED			Write '1' to disable interrupt for event NOTRESOLVED																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
D	RW	ERROR			Write '1' to disable interrupt for event ERROR																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

8.3.5.15 ERRORSTATUS

Address offset: 0x404

Error status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	ERRORSTATUS			Error status when the ERROR event is generated																														
			NoError	0	No errors have occurred																														
			PrematureInptrEnd	1	End of INPTR job list before data structure was read.																														
			DmaError	4	Bus error during DMA access.																														

8.3.5.16 ENABLE

Address offset: 0x500

Enable AAR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE			Enable or disable AAR																														
			Disabled	0	Disable																														
			Enabled	3	Enable																														

8.3.5.17 MAXRESOLVED

Address offset: 0x508

Maximum number of IRKs to resolve

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000001				0 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	MAXRESOLVED						The maximum number of IRKs to resolve																											
								After MAXRESOLVED number of IRKs have been resolved, AAR will stop processing and generate the END event																											

8.3.5.18 IN

IN EasyDMA channel

8.3.5.18.1 IN.PTR

Address offset: 0x530

Input pointer

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						Points to a job list containing AAR data structure																											

8.3.5.19 OUT

OUT EasyDMA channel

8.3.5.19.1 OUT.PTR

Address offset: 0x538

Output pointer

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						Output pointer																											

8.3.5.19.2 OUT.AMOUNT

Address offset: 0x53C

Number of bytes transferred in the last transaction

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0												
ID																												A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0										
ID	R/W	Field	Value ID	Value				Description																																							
A	R	AMOUNT		[1..255]				Number of bytes written to memory after triggering the START task. One IRK uses two bytes, so every two bytes means one resolved IRK index																																							

8.4 CCM — AES CCM mode encryption

Counter with cipher block chaining - message authentication code (CCM) mode is an authenticated encryption algorithm designed to provide both authentication and confidentiality (encryption/decryption) during data transfer.

The main features of CCM are:

- Memory-to-memory packet encryption and decryption operations using Scatter/Gather DMA
- Support for Bluetooth requirements and algorithm as defined in IETF [RFC3610](#)
- Support for IEEE 802.15.4
- Concurrent operation with RADIO

AES CCM combines counter (CTR) mode encryption and cipher block chaining - message authentication code (CBC-MAC) authentication. The CCM terminology message authentication code (MAC) is called message integrity check (MIC) in *Bluetooth* terminology, and also in this document.



Figure 43: CCM Overview

CCM generates an encrypted keystream that is applied to input data using the XOR operation and generates an M byte MAC field in one operation. CCM and RADIO can be configured to work synchronously. CCM will encrypt in time for transmission and decrypt after receiving bytes into memory from the radio. All operations can complete within the packet RX or TX time. CCM on this device is implemented to support the Bluetooth requirements and the algorithm as defined in IETF [RFC3610](#), and depends on the AES-128 block cipher. A description of the CCM algorithm can also be found in [NIST Special Publication 800-38C](#). The Bluetooth specification describes the configuration of counter mode blocks and encryption blocks to implement compliant encryption for Bluetooth Low Energy.

CCM uses EasyDMA to read/write additional authenticated data, plain text and cipher text.

Two operations are supported:

- Packet encryption
- Packet decryption

All operations are done in compliance with the *Bluetooth Core Specification*, as well as IEEE 802.15.4.

8.4.1 Shared resources

The CCM shares the same AES module as the AAR and ECB peripherals. The ECB will always have the lowest priority. If an operation is aborted due to a conflict among the shared resources, an ERROR event will be generated.

Additionally, the CCM shares registers and other resources with the peripherals that have the same ID as the CCM. See [Peripherals with shared ID](#) on page 214 for more information.

8.4.2 Encryption and decryption

CCM supports both packet encryption and decryption.

The following table shows the different CCM input/output and parameters supported by the CCM module for encryption and decryption:

Parameter	Valid input	Description
M	0, 4, 6, 8, 10, 12, 14, 16	Number of bytes in the authentication field
L	2 (fixed)	Number of bytes in the length field
l(a)	0-65279	Number of bytes in additional authenticated data
l(m)	0-(65535 - M)	Number of bytes in the message to authenticate and encrypt
l(c)	0-65535	Number of bytes in the encrypted message; l(m) + M bytes
a	l(a) number of bytes	Additional authenticated data
m	l(m) number of bytes	Message to authenticate and encrypt
c	l(c) number of bytes	Encrypted message

Table 33: CCM Parameters

In addition to the parameters listed above, the CCM requires two sets of data: a 128-bit key and a 128-bit nonce. These are supplied via dedicated register interfaces: [KEY.VALUE](#) registers for the 128-bit key, and [NONCE.VALUE](#) registers for the 128-bit nonce. The 128-bit key in the [KEY.VALUE](#) registers is stored in

reverse byte order relative to the payload. For example, using the sample session key from the Bluetooth Core Specification v5.4, Volume 6, Part C, chapter 1.2:

- Session Key (SK): 99AD1B5226A37E3E058E3B8E27C2C666

The **KEY.VALUE** registers are populated as follows:

- **KEY.VALUE[0]** = 0x27C2C666
- **KEY.VALUE[1]** = 0x058E3B8E
- **KEY.VALUE[2]** = 0x26A37E3E
- **KEY.VALUE[3]** = 0x99AD1B52

The same reverse byte order is used for the **NONCE.VALUE** registers. For the packet example "3. Data packet1" with the following values:

- IV: DEAFBABEBADCAB24
- Direction Bit: 1
- Packet Counter: 1

The **NONCE.VALUE** registers are populated as follows:

- **NONCE.VALUE[0]** = 0xBEBAAFDE
- **NONCE.VALUE[1]** = 0x24ABDCBA
- **NONCE.VALUE[2]** = 0x00000080
- **NONCE.VALUE[3]** = 0x00000001

Note: Although the NONCE in the example above is 13 bytes, it must be written as a 16-byte value with the first 3 bytes zero-padded.

Note: The KEY and NONCE byte order is reversed compared to the NRF52 and NRF53 series devices.

8.4.2.1 Encryption

During packet encryption, CCM will read the unencrypted packet located in memory at the address specified in register **IN.PTR**, encrypt the packet and append an M byte long message authentication code (MAC) field to the packet.

The message to authenticate and encrypt (m) and additional authenticated data (a) are included in the MAC generation. The first byte in the packet header can be masked by configuring the **ADATAMASK** register. This is useful for Bluetooth header masking. For protocols other than Bluetooth, the **ADATAMASK** register must be set to 0xFF for correct CCM operation; the reset value is configured to support Bluetooth.

Encryption is started by triggering the **START** task with the **MODE** register set to **Encryption**. The **END** event will be generated when packet encryption is completed.

The AES CCM will modify the l(c) output field of the packet to adjust for the appended MAC field, that is, add **MODE.MACLEN** bytes to l(m), and store the resulting packet back into memory at the locations specified in the **OUT.PTR** list, as illustrated in the following figure. The maximum length of l(m) plus **MODE.MACLEN** cannot exceed 65535 bytes.

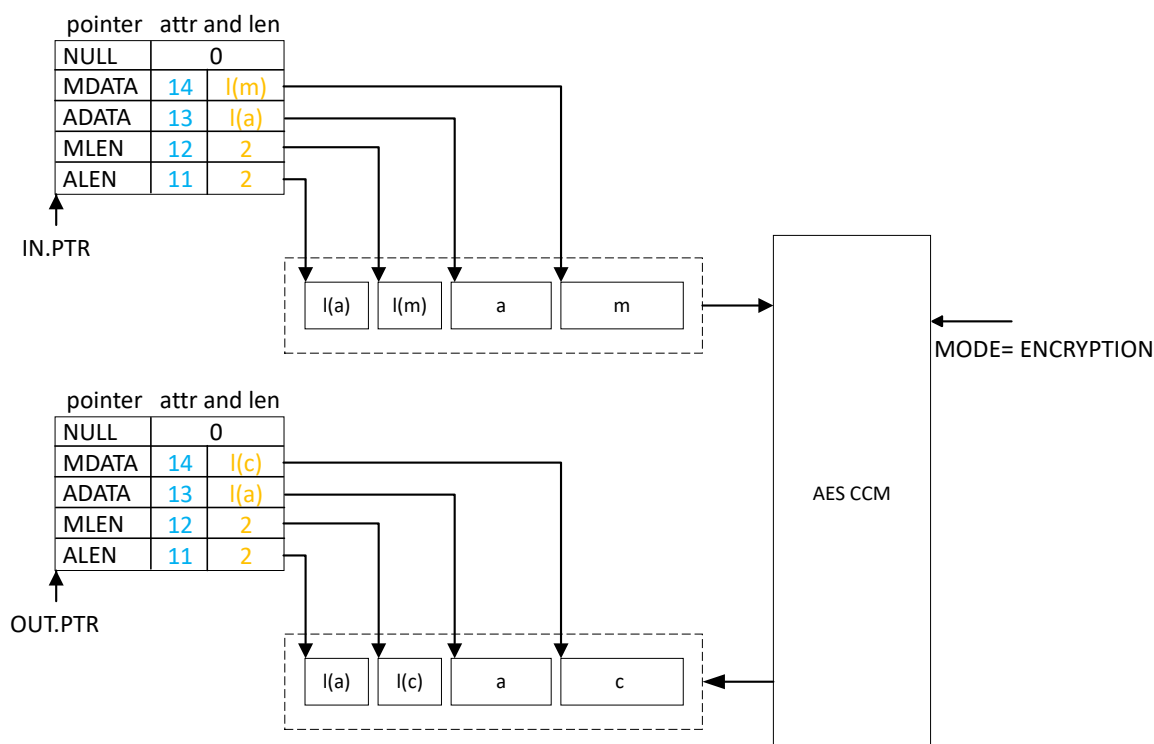


Figure 44: Encryption

If the following occurs, the **ERROR** event is generated, the CCM stops, and the **ERRORSTATUS** register will report the type of error that triggered the **ERROR** event:

- The **IN.PTR** job list ends before reading out the complete CCM data structure
- The **OUT.PTR** job list ends before writing out the complete encrypted CCM data structure
- The CCM is not able to operate fast enough to run concurrently with the RADIO as the RADIO transmits the encrypted packet.
- The EasyDMA engine encounters an error, see [EasyDMA and ERROR event](#) on page 239

Any values of $I(m)$ and $I(a)$ are allowed. If encrypting empty packets, i.e. $I(m) = I(a) = 0$, no encryption will take place; the **END** event is generated, and CCM operation is stopped.

For Bluetooth (**MODE.PROTOCOL=BLE**), valid packets with 0 payload ($I(a)$ is larger than 0 but $I(m)$ is 0) will not be authenticated but instead moved unmodified through the AES CCM peripheral, and thus no MAC will be generated.

For IEEE 802.15.4 (**MODE.PROTOCOL=IEEE802154**), valid packets with 0 payload ($I(a)$ is larger than 0 but $I(m)$ is 0) will be authenticated, and thus a MAC will be generated as part of the output data.

8.4.2.2 Decryption

During packet decryption, CCM will read the encrypted packet located in memory at the address specified in the **IN.PTR** pointer, decrypt the packet, authenticate the packet's MAC field and generate the appropriate MAC status.

The encrypted message in (c), is decrypted and authenticated together with additional authenticated data (a) and then matched against the decrypted MAC value. The decrypted MAC value is part of (c). Bits in the first byte of the data can be masked away before calculating the MAC value by configuring the **ADATAMASK** register. This is useful for Bluetooth header masking. For protocols other than Bluetooth, the **ADATAMASK** register must be set to 0xFF for correct CCM operation; the reset value is configured to support Bluetooth.

Decryption is started by triggering the **START** task with the **MODE** register set to **FastDecryption**.

CCM will write the $I(m)$ value of the decrypted packet to the location provided in **OUT.PTR**, and then store the decrypted packet into memory at the locations given by the **OUT.PTR** list as illustrated in the following figure.

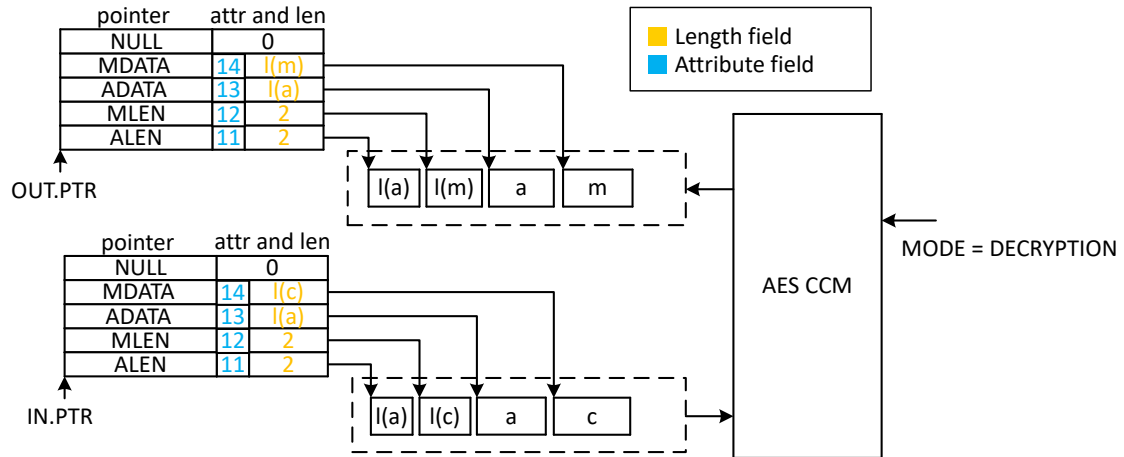


Figure 45: Decryption

For Bluetooth (**MODE.PROTOCOL=BLE**), CCM is only able to authenticate messages where $I(c)$ is at least **MACLEN+1** bytes long. If $I(c)$ is less than **MACLEN+1**, CCM will generate an **END** event and clear the **MACSTATUS** (indicating MAC check failure). Furthermore, empty packets ($I(c)=0$) will be moved unmodified through the AES CCM peripheral even though **ERROR** event shall be generated. In any other case that leads to a failed **MACSTATUS** or an **ERROR** event, the contents of the job addresses given in **OUT.PTR** are undefined.

For IEEE 802.15.4 (**MODE.PROTOCOL=IEEE802154**), CCM will also perform authentication on messages where only **ADATA** is present (i.e. $I(m)=0$ and $I(a)>0$). In this case **MACSTATUS** reflects the result of the authentication. If $I(c)<MACLEN$, then the **ERROR** event is generated, and the contents of the locations given in **OUT.PTR** are undefined.

If the following occurs, the **ERROR** event is generated, and CCM is stopped.

- The **IN.PTR** job list ends before reading out the complete CCM data structure
- The **OUT.PTR** job list ends before writing out the complete decrypted CCM data structure
- The EasyDMA engine encounters an error, see [EasyDMA and ERROR event](#) on page 239

If the **IN.PTR** or **OUT.PTR** job lists do not end before the complete encrypted/decrypted CCM data structures are read, the **END** event is generated and CCM operation is stopped.

8.4.3 Encrypting packets in radio transmit mode

When the AES CCM is encrypting a packet at the same time as the radio is transmitting it, the radio must read the encrypted packet from the same memory location as the AES CCM is writing to.

The **OUT.PTR** pointer in the AES CCM must therefore point to the same memory location as the **PACKETPTR** pointer in the radio, see [Example configuration of encryption during radio transmission](#) on page 237.

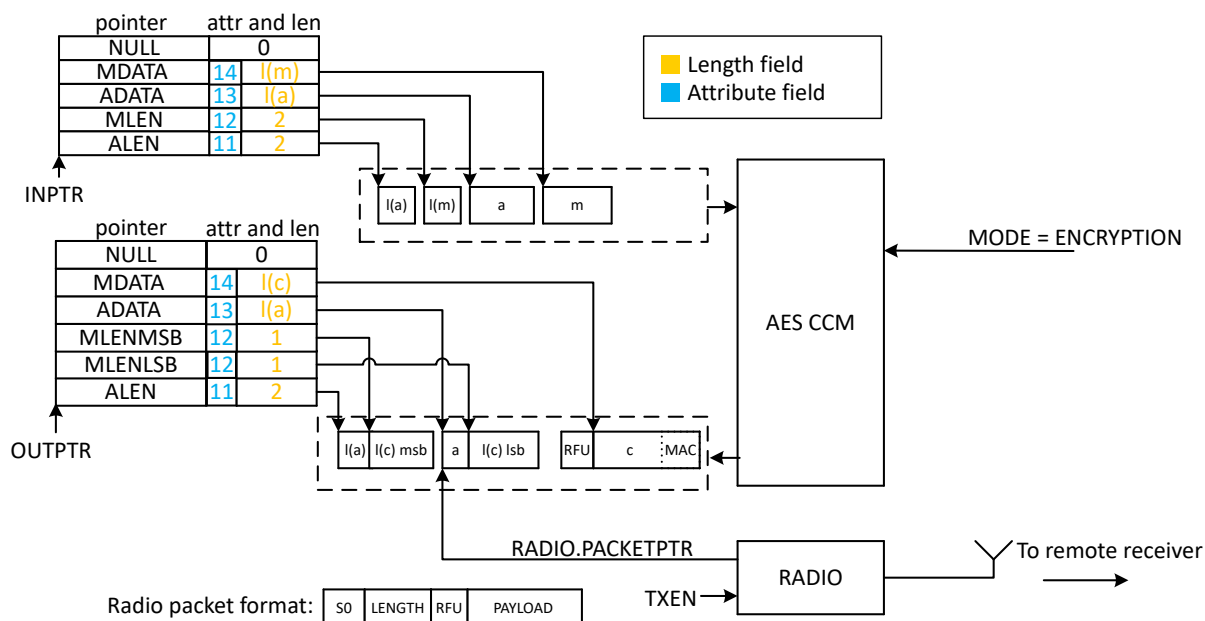


Figure 46: Example configuration of encryption during radio transmission

The **START** task must be triggered by **RADIO READY** event to ensure that the payload is encrypted in time for radio transmission. This is illustrated in the following figure, using a PPI connection between **RADIO.EVENTS_READY** and **CCM.TASKS_START**.

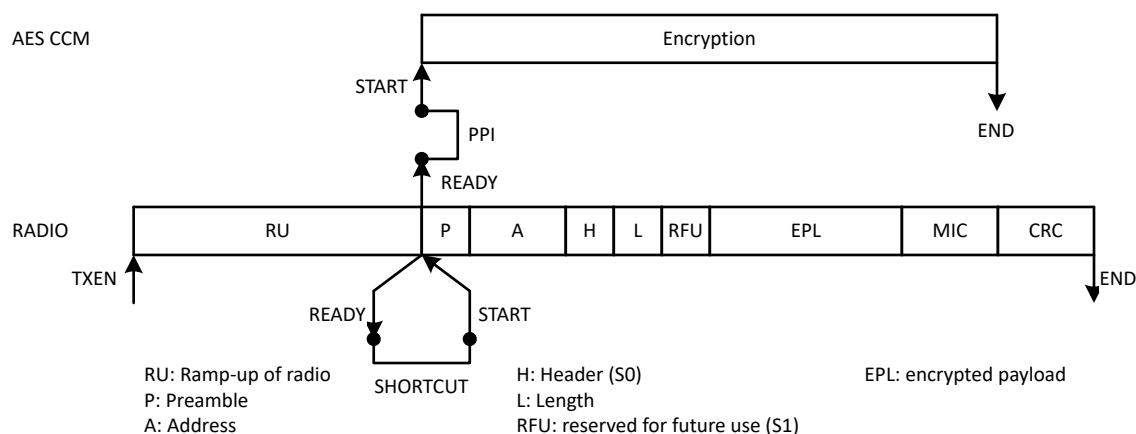


Figure 47: Radio transmission with encryption using a PPI connection

8.4.4 Decrypting packets received by the radio

To decrypt a packet received by the radio immediately upon its reception, CCM can be started when the **RADIO PAYLOAD** event is generated. The packet is decrypted when the **CCM.END** event is generated. Typically, CCM will decrypt the packet before or during the reception of the CRC. However, if the packet is large and the bitrate is high, CCM will not finish before the **PHYEND** event, but shortly afterward. After the **CCM.END** event is generated; the **MACSTATUS** can be checked.

AES CCM must therefore operate on the same memory location as RADIO, as illustrated in the following figure.

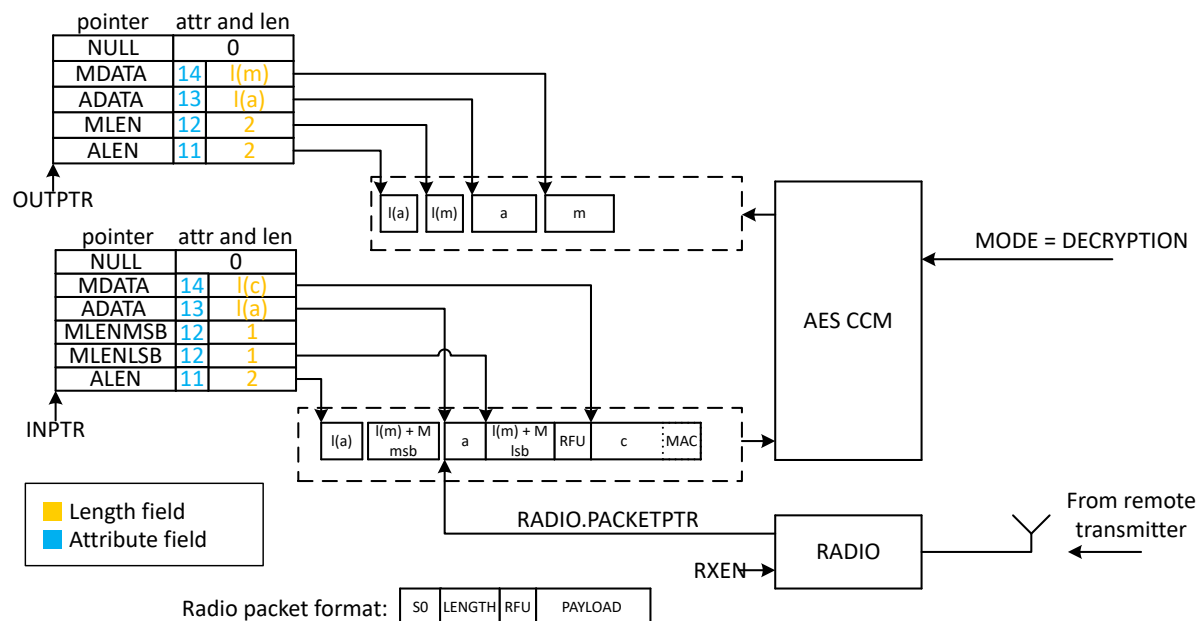


Figure 48: Example configuration of CCM for decrypting a packet as it is received by the RADIO

8.4.5 CCM data structure

The input and output data structures are located in memory specified by **IN.PTR** and **OUT.PTR** on page 246 respectively.

Both **IN.PTR** and **OUT.PTR** point to a scatter/gather job list. This job list must contain all the fields listed in the attribute field table. Each job list must be terminated with a 0 filled job entry. If either of the **IN.PTR** or **OUT.PTR** job list is not terminated, then the behavior of CCM is undefined.

The job list consists of one or more job entries each containing a 32-bit address field, an 8-bit attribute field, and a 24-bit length field. A job list ends with a zero-filled job entry. The EasyDMA job list example below illustrates a job list that points to three different memory sections with varying lengths. The data pointed to by the job list is fed into the module to be processed according to the CCM operation. Job entries with a length greater than one byte are processed in little endian order.

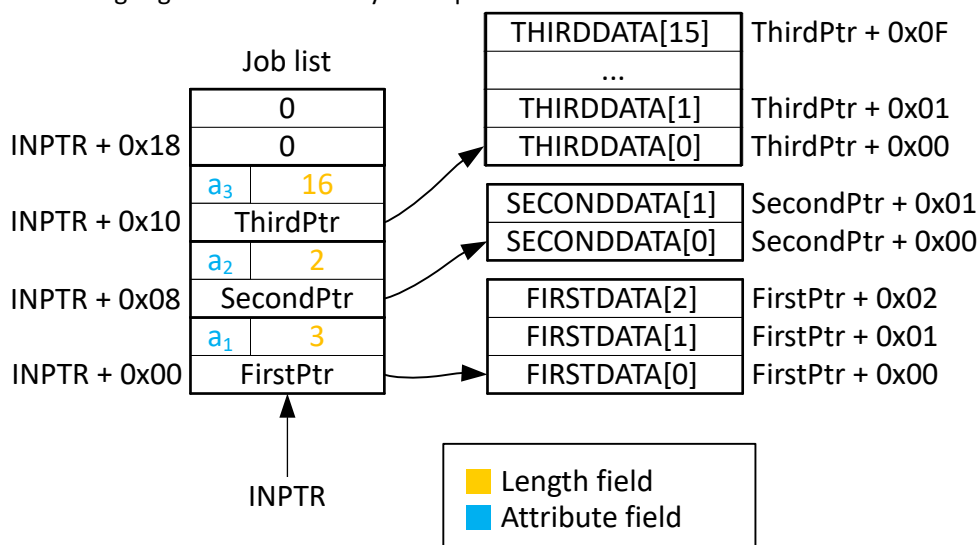


Figure 49: EasyDMA job list example

The attribute field identifies the job and must be set according to the following table.

Attribute	Value
Alen	11
Mlen	12
Adata	13
Mdata	14

Table 34: Attribute field

8.4.6 EasyDMA and ERROR event

The CCM implements an EasyDMA with scatter/gather mechanism for reading and writing to memory.

In cases where the CPU and other EasyDMA enabled peripherals are accessing the same RAM block at the same time, a high level of bus collisions may cause too slow operation for correct on the fly encryption. In this case the ERROR event will be generated.

EasyDMA will have finished accessing the memory when the **END** event is generated.

If the **IN.PTR** and the **OUT.PTR** are not pointing to memory with DMA connectivity, an EasyDMA transfer may result in a HardFault or memory corruption. See **Memory** on page 19 for more information about the different memory regions.

For instances supporting DMA error detection, the **ERRORSTATUS** register will report if a bus error has occurred during DMA access.

8.4.7 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
CCM00 : S	GLOBAL	0x50046000	US	S	SA	No	AES CCM mode encryption CCM00, running of HCLK128M
CCM00 : NS		0x40046000					

Configuration

Instance	Domain	Configuration
CCM00 : S	GLOBAL	Does not support on-the-fly decryption.
CCM00 : NS		

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Start encryption/decryption. This operation will stop by itself when completed.
TASKS_STOP	0x004		Stop encryption/decryption
TASKS_RATEOVERRIDE	0x008		Override DATARATE setting in MODE register with the contents of the RATEOVERRIDE register for any ongoing encryption/decryption
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
SUBSCRIBE_RATEOVERRIDE	0x088		Subscribe configuration for task RATEOVERRIDE
EVENTS_END	0x104		Encrypt/decrypt complete or ended because of an error

Register	Offset	TZ	Description
EVENTS_ERROR	0x108		CCM error event
PUBLISH_END	0x184		Publish configuration for event END
PUBLISH_ERROR	0x188		Publish configuration for event ERROR
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
MACSTATUS	0x400		MAC check result
ERRORSTATUS	0x404		Error status
ENABLE	0x500		Enable
MODE	0x504		Operation mode
KEY.VALUE[n]	0x510		128-bit AES key
NONCE.VALUE[n]	0x520		13-byte NONCE vector
			Only the lower 13 bytes are used
IN.PTR	0x530		Input pointer
			Points to a job list containing unencrypted CCM data structure in Encryption mode
			Points to a job list containing encrypted CCM data structure in Decryption mode
OUT.PTR	0x538		Output pointer
			Points to a job list containing encrypted CCM data structure in Encryption mode
			Points to a job list containing decrypted CCM data structure in Decryption mode
RATEOVERRIDE	0x544		Data rate override setting.
ADATAMASK	0x548		CCM adata mask.

8.4.7.1 TASKS_START

Address offset: 0x000

Start encryption/decryption. This operation will stop by itself when completed.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_START						Start encryption/decryption. This operation will stop by itself when completed.																											
			Trigger	1				Trigger task																											

8.4.7.2 TASKS_STOP

Address offset: 0x004

Stop encryption/decryption

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_STOP						Stop encryption/decryption																											
			Trigger	1				Trigger task																											

8.4.7.3 TASKS_RATEOVERRIDE

Address offset: 0x008

Override DATARATE setting in MODE register with the contents of the RATEOVERRIDE register for any ongoing encryption/decryption

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																							A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																		
A	W	TASKS_RATEOVERRIDE			Override DATARATE setting in MODE register with the contents of the RATEOVERRIDE register for any ongoing encryption/decryption																																		
			Trigger	1	Trigger task																																		

8.4.7.4 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task **START**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that task START will subscribe to																																																									
B	RW	EN																																																													
			Disabled	0	Disable subscription																																																										
			Enabled	1	Enable subscription																																																										

8.4.7.5 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task **STOP**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that task STOP will subscribe to																																																									
B	RW	EN																																																													
			Disabled	0	Disable subscription																																																										
			Enabled	1	Enable subscription																																																										

8.4.7.6 SUBSCRIBE_RATEOVERRIDE

Address offset: 0x088

Subscribe configuration for task **RATEOVERRIDE**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				B																								A				A	A	A	A	A	A
Reset 0x00000000				0																																	
ID	R/W	Field	Value ID	Value				Description																													
A	RW	CHIDX		[0..255]				DPPI channel that task RATEOVERRIDE will subscribe to																													
B	RW	EN																																			
			Disabled	0	Disable subscription																																
			Enabled	1	Enable subscription																																

8.4.7.7 EVENTS_END

Address offset: 0x104

Encrypt/decrypt complete or ended because of an error

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				A																																			
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	EVENTS_END				Encrypt/decrypt complete or ended because of an error																																	
			NotGenerated	0		Event not generated																																	
			Generated	1		Event generated																																	

8.4.7.8 EVENTS_ERROR

Address offset: 0x108

CCM error event

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				A																																			
Reset 0x00000000				0 0																																			
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	EVENTS_ERROR				CCM error event																																	
			NotGenerated	0		Event not generated																																	
			Generated	1		Event generated																																	

8.4.7.9 PUBLISH_END

Address offset: 0x184

Publish configuration for event [END](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0												
ID				B																								A				A				A				A				A			
Reset 0x00000000				0																																											
ID	R/W	Field	Value ID	Value		Description																																									
A	RW	CHIDX		[0..255]		DPPI channel that event END will publish to																																									
B	RW	EN																																													
			Disabled	0	Disable publishing																																										
			Enabled	1	Enable publishing																																										

8.4.7.10 PUBLISH_ERROR

Address offset: 0x188

Publish configuration for event **ERROR**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event ERROR will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.4.7.11 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	END			Write '1' to enable interrupt for event END																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	ERROR			Write '1' to enable interrupt for event ERROR																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.4.7.12 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	END			Write '1' to disable interrupt for event END																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	ERROR			Write '1' to disable interrupt for event ERROR																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.4.7.13 MACSTATUS

Address offset: 0x400

MAC check result

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				A																																			
Reset 0x00000000				0 0																																			
ID	R/W	Field	Value ID	Value		Description																																	
A	R	MACSTATUS				The result of the MAC check performed during the previous decryption operation																																	
			CheckFailed	0		MAC check failed																																	
			CheckPassed	1		MAC check passed																																	

8.4.7.14 ERRORSTATUS

Address offset: 0x404

Error status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	ERRORSTATUS			Error status when the ERROR event is generated																														
			NoError	0	No errors have occurred																														
			PrematureInptrEnd	1	End of INPTR job list before CCM data structure was read.																														
			PrematureOutptrEnd	2	End of OUTPTR job list before CCM data structure was read.																														
			EncryptionTooSlow	3	Encryption of the unencrypted CCM data structure did not complete in time.																														
			DmaError	4	Bus error during DMA access.																														

8.4.7.15 ENABLE

Address offset: 0x500

Enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE			Enable or disable CCM																														
			Disabled	0	Disable																														
			Enabled	2	Enable																														

8.4.7.16 MODE

Address offset: 0x504

Operation mode

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID														D D D						C C C						B B						A A			
Reset 0x00000001				0 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	MODE						The mode of operation to be used. The settings in this register apply when the CRYPT task is triggered.																											
			Encryption	0				AES CCM packet encryption mode																											

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				D			D	D	C			C	C	B			B	A			A																	
Reset 0x00000001				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1			
ID	R/W	Field	Value ID	Value	Description																																	
			Decryption	1	This mode will run CCM decryption in the speed of the DATARATE field.																																	
			FastDecryption	2	This enumerator is deprecated.																																	
					AES CCM decryption mode.																																	
				3																																		
B	RW	PROTOCOL			Protocol and packet format selection																																	
			Ble	0	Bluetooth Low Energy packet format																																	
			ieee802154	1	802.15.4 packet format																																	
				2																																		
				3																																		
C	RW	DATARATE			Radio data rate that the CCM shall run synchronous with																																	
			125Kbit	0	125 Kbps																																	
			250Kbit	1	250 Kbps																																	
			500Kbit	2	500 Kbps																																	
			1Mbit	3	1 Mbps																																	
			2Mbit	4	2 Mbps																																	
			4Mbit	5	4 Mbps																																	
D	RW	MACLEN			CCM MAC length (bytes)																																	
			M0	0	M = 0																																	
					This is a special case for CCM* where encryption is required but not authentication																																	
			M4	1	M = 4																																	
			M6	2	M = 6																																	
			M8	3	M = 8																																	
			M10	4	M = 10																																	
			M12	5	M = 12																																	
			M14	6	M = 14																																	
			M16	7	M = 16																																	

8.4.7.17 KEY.VALUE[n] (n=0..3)

Address offset: 0x510 + (n × 0x4)

128-bit AES key

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	W	VALUE						AES 128-bit key value, bits (32*(i+1))-1 : (32*i)																											

8.4.7.18 NONCE.VALUE[n] (n=0..3)

Address offset: 0x520 + (n × 0x4)

13-byte NONCE vector

Only the lower 13 bytes are used

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	VALUE						NONCE value, bits (32*(n+1))-1 : (32*n)																											

8.4.7.19 IN

IN EasyDMA channel

8.4.7.19.1 IN.PTR

Address offset: 0x530

Input pointer

Points to a job list containing unencrypted CCM data structure in Encryption mode

Points to a job list containing encrypted CCM data structure in Decryption mode

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						Input pointer																											

8.4.7.20 OUT

OUT EasyDMA channel

8.4.7.20.1 OUT.PTR

Address offset: 0x538

Output pointer

Points to a job list containing encrypted CCM data structure in Encryption mode

Points to a job list containing decrypted CCM data structure in Decryption mode

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						Output pointer																											

8.4.7.21 RATEOVERRIDE

Address offset: 0x544

Data rate override setting.

Override value to be used instead of the setting of MODE.DATARATE. This override value applies when the RATEOVERRIDE task is triggered.

Note: The override is only applied when operating in BLE Long Range mode.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000002				0 1 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	RATEOVERRIDE						Data rate override setting.																											
			125Kbit	0				125 Kbps																											
			500Kbit	2				500 Kbps																											
			1Mbit	3				1 Mbps																											
			2Mbit	4				2 Mbps																											
			4Mbit	5				4 Mbps																											

8.4.7.22 ADATAMASK

Address offset: 0x548

CCM adata mask.

Bitmask for the first adata byte. The masking is done before MAC generation/authentication.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																															A	A	A	A	A	A	A	A	A	A
Reset 0x000000E3					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1			
ID	R/W	Field	Value ID	Value	Description																																			
A	RW	ADATAMASK			CCM adata mask.																																			

8.5 COMP — Comparator

The comparator peripheral (COMP) compares one input voltage (VIN+) against another input voltage (VIN-). VIN+ can be derived from an analog input pin (**AIN0** to **AIN7**). VIN- can be derived from multiple sources depending on the operation mode of the comparator.

The main features of COMP are the following:

- Input range from 0 V to VDD
- Single-ended mode
 - Fully flexible hysteresis using a 64-level reference ladder
- Differential mode
 - Configurable hysteresis
- Reference inputs (VREF)
 - External reference from **AIN0** to **AIN7** (between 0 V and VDD)
 - Internal VDD reference
 - 1.2 V internal reference
- Two speed/power consumption modes, low-power and high-speed
- Single-pin capacitive sensor support
- Event generation on output changes
 - UP event on VIN- > VIN+
 - DOWN event on VIN- < VIN+
 - CROSS event on VIN+ and VIN- crossing
 - READY event on core and internal reference (if used) ready

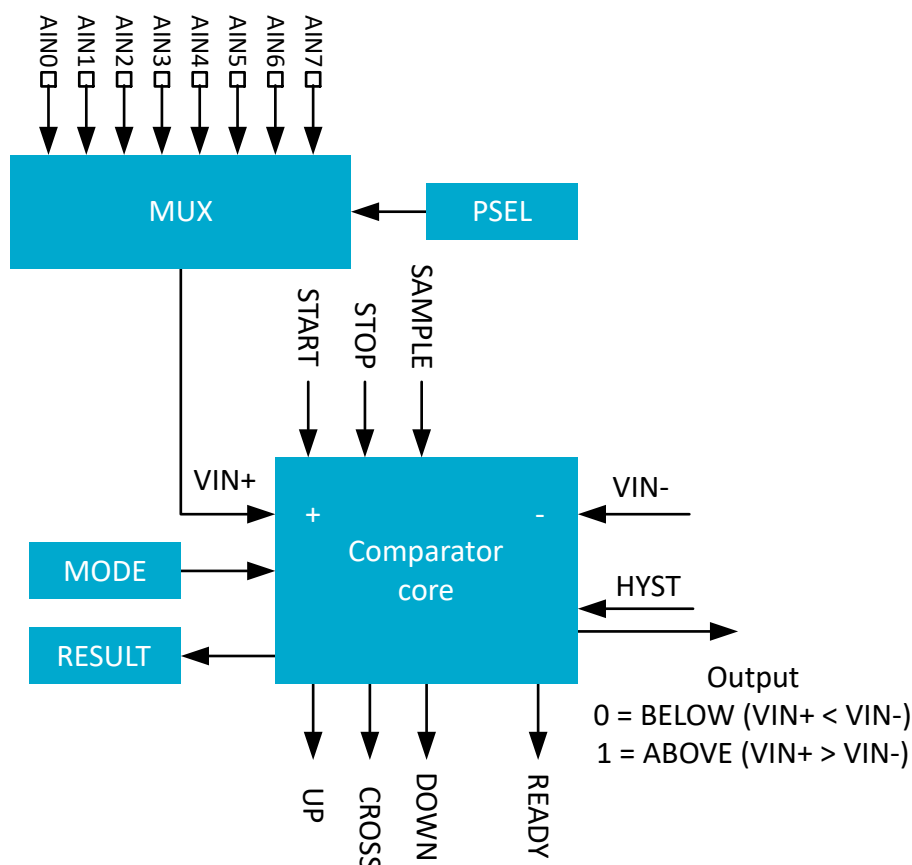


Figure 50: COMP overview

8.5.1 START and STOP tasks

Once enabled through register [ENABLE](#), COMP is started by triggering the START task and stopped by triggering the STOP task. COMP will generate a READY event to indicate when it is ready for use and the output is correct. The delay between START and READY is $t_{\text{INT_REF,START}}$ when an internal reference is selected, or $t_{\text{COMP,START}}$ if an external reference is used. When the COMP peripheral is started, events will be generated every time VIN+ crosses VIN-.

8.5.2 Operation modes

COMP can be configured to operate in the two main operation modes, differential mode and single-ended mode. See register [MODE](#) for more information. In both operation modes, COMP can operate in different speed and power consumption modes (low-power to high-speed). High-speed mode will consume more power compared to low-power mode. Low-power mode will result in a slower response time compared to high-speed mode.

Use register [PSEL](#) to select any of the **AIN[0 . . 7]** pins as the VIN+ input. The COMP operation mode does not matter. The source of VIN- depends on which operation mode is used.

- Differential mode – Derived directly from pins **AIN[0 . . 7]**.
- Single-ended mode – Derived from VREF. VREF can be derived from VDD, **AIN[0 . . 7]**, or internal 1.2 V reference.

The selected analog pins will be acquired by the comparator once it is enabled.

An optional hysteresis on VIN+ and VIN- can be enabled when the module is used in differential mode through the register [HYST](#). In single-ended mode, VUP and VDOWN thresholds can be set to implement

a hysteresis using the reference ladder (see [Comparator in single-ended mode](#) on page 251). This hysteresis is in the order of magnitude of $V_{DIFFHYST}$, and prevents noise on the signal to create unwanted events. See [Hysteresis example where VIN+ starts below VUP](#) on page 252 for an illustration of the effect of an active hysteresis on a noisy input signal.

An upward crossing will generate an UP event and a downward crossing will generate a DOWN event. The CROSS event will be generated every time there is a crossing, independent of direction.

The immediate value of the comparator can be sampled to register [RESULT](#) by triggering the SAMPLE task.

8.5.3 Current source on analog input

A selectable current can be applied through register [ISOURCE](#) on the selected AIN[n] line. Enabling the block creates a feedback path around the comparator, forming a relaxation oscillator. The circuit will sink current from VIN+ when the comparator output is high, and source current into VIN+ when the comparator output is low. The frequency of the oscillator is dependent on the capacitance at the analog input pin, the reference voltages, and the value of the current source. In this mode, only a capacitive sensor needs to be attached between the analog input pin and ground. With a selected current of 10 μ A, VUP-VDOWN equal to 1 V, and an external capacity of typically 10 pF, the resulting oscillation frequency is around 500 kHz.

The frequency of the oscillator can be calculated as follows:

$$f_{OSC} = I_{SOURCE} / (2C \cdot (VUP - VDOWN))$$

8.5.4 Shared resources

The COMP shares analog resources with other analog peripherals.

Additionally, COMP shares registers and other resources with other peripherals that have the same ID as the COMP. See [Peripherals with shared ID](#) on page 214 for more information.

The COMP peripheral shall not be disabled (by writing to the ENABLE register) before the peripheral has been stopped. Failing to do so may result in unpredictable behavior.

8.5.5 Differential mode

In differential mode, the reference input VIN- is derived directly from one of the AINx pins.

Before enabling the comparator via the [ENABLE](#) register, the following registers must be configured for the differential mode:

- [PSEL](#)
- [MODE](#)
- [EXTREFSEL](#)

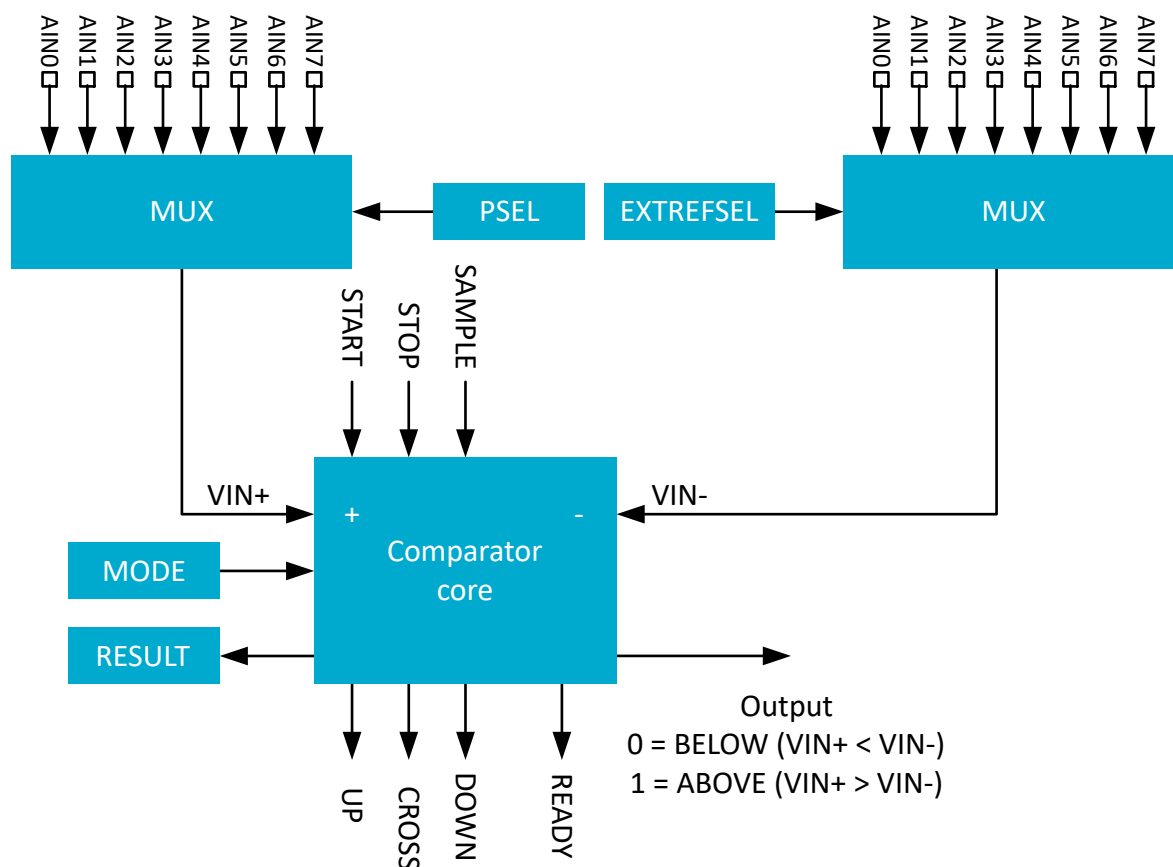


Figure 51: Comparator in differential mode

Note: Depending on the device, not all the analog inputs may be available for each MUX. See definitions for [PSEL](#) and [EXTREFSEL](#) for more information about which analog pins are available on a particular device.

When [HYST](#) register is turned on while in this mode, the output of the comparator (and associated events) will change from ABOVE to BELOW whenever V_{IN+} becomes lower than $V_{IN-} - (V_{DIFFHYST} / 2)$. It will also change from BELOW to ABOVE whenever V_{IN+} becomes higher than $V_{IN-} + (V_{DIFFHYST} / 2)$. This behavior is illustrated in [Hysteresis enabled in differential mode](#) on page 250.

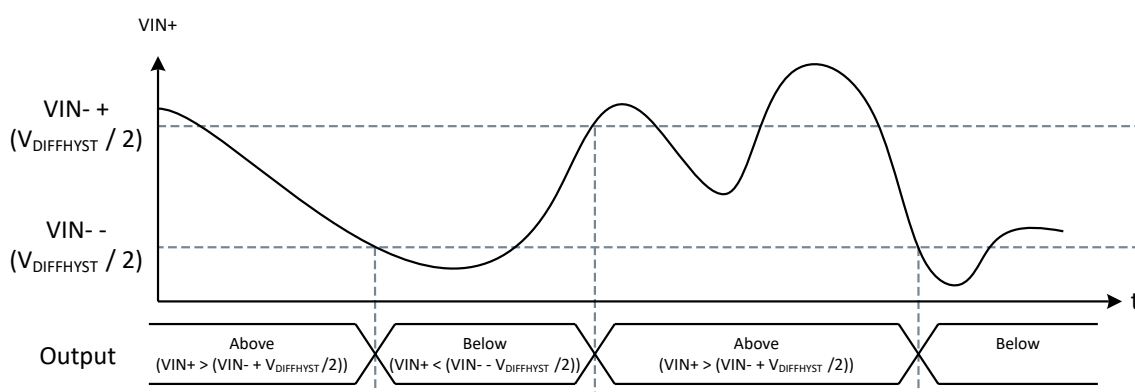


Figure 52: Hysteresis enabled in differential mode

8.5.6 Single-ended mode

In single-ended mode, VIN- is derived from the reference ladder.

Before enabling the comparator via the **ENABLE** register, the following registers must be configured for the single-ended mode:

- **PSEL**
- **MODE**
- **REFSEL**
- **EXTREFSEL**
- **TH**

The reference ladder uses the reference voltage (VREF) to derive two new voltage references, VUP and VDOWN. VUP and VDOWN are configured using THUP and THDOWN respectively in the **TH** register. VREF can be derived from any of the available reference sources, configured using the **EXTREFSEL** and **REFSEL** registers as illustrated in [Comparator in single-ended mode](#) on page 251. When AREF is selected in the **REFSEL** register, the **EXTREFSEL** register is used to select one of the AIN0-AIN7 analog input pins as reference input. The selected analog pins will be acquired by the comparator once it is enabled.

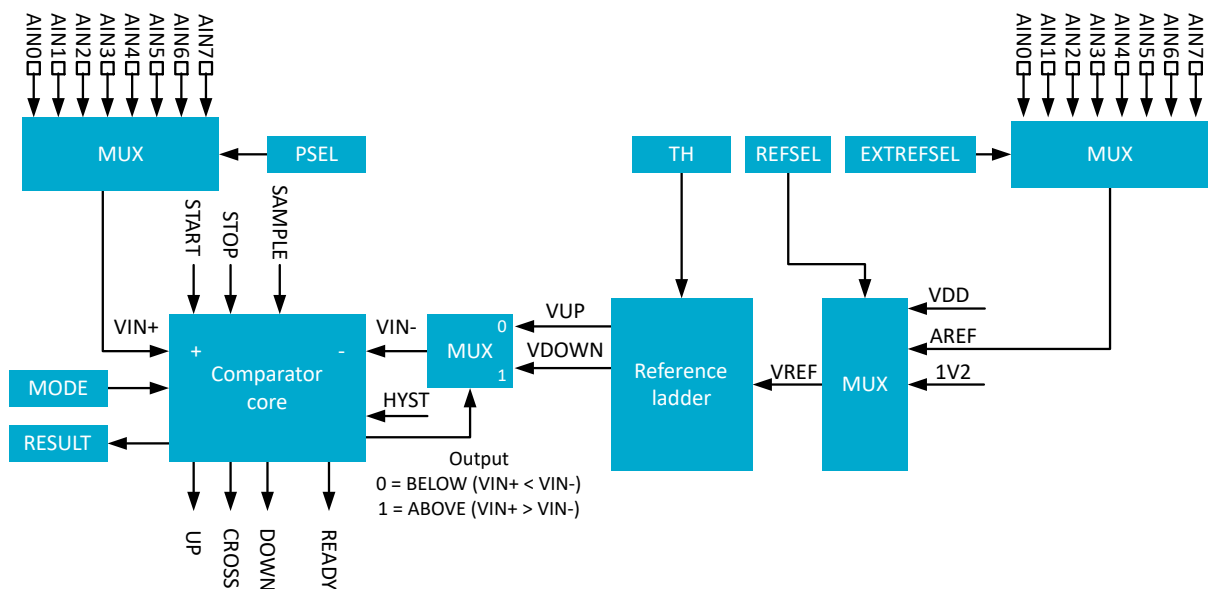


Figure 53: Comparator in single-ended mode

Note: Depending on the device, not all the analog inputs may be available for each MUX. See definitions for **PSEL** and **EXTREFSEL** for more information about which analog pins are available on a particular device.

When the comparator core detects that $VIN+ > VIN-$, i.e. ABOVE as per the **RESULT** register, **VIN-** will switch to **VDOWN**. When **VIN+** falls below **VIN-** again, **VIN-** will be switched back to **VUP**. By specifying **VUP** larger than **VDOWN**, a hysteresis can be generated as illustrated in [Hysteresis example where VIN+ starts below VUP](#) on page 252 and [Hysteresis example where VIN+ starts above VUP](#) on page 252.

Writing to **HYST** has no effect in single-ended mode, and the content of this register is ignored.

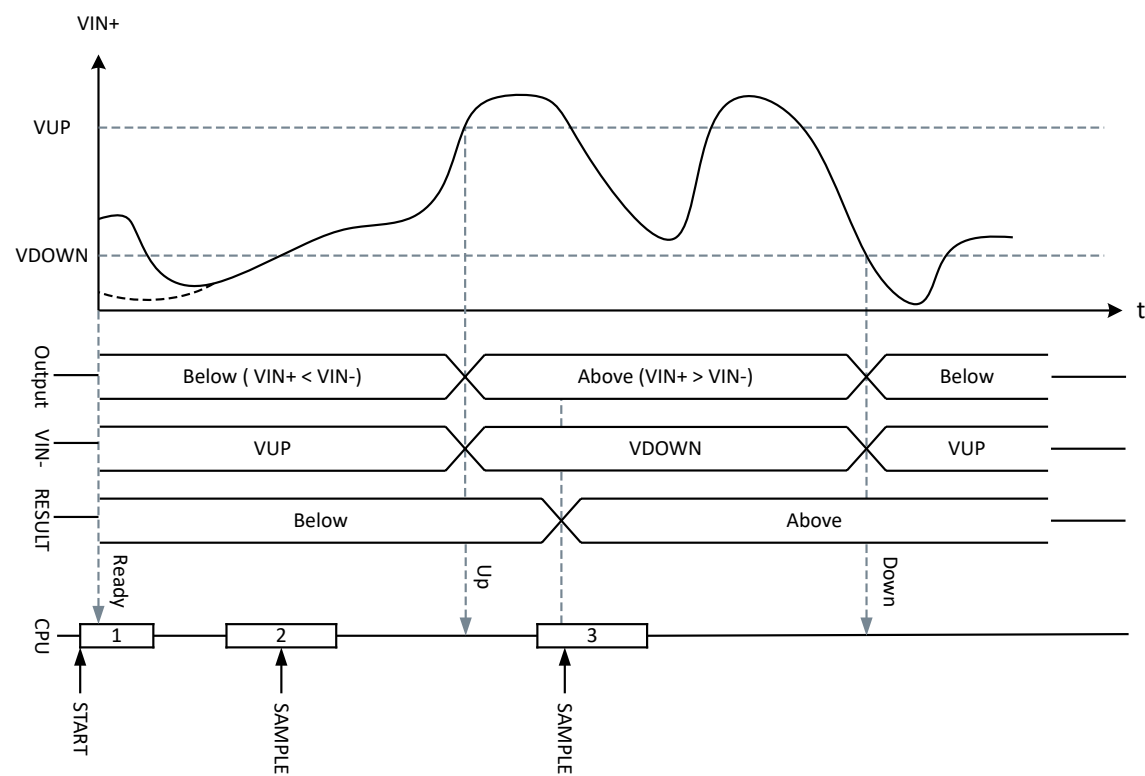


Figure 54: Hysteresis example where V_{IN+} starts below V_{UP}

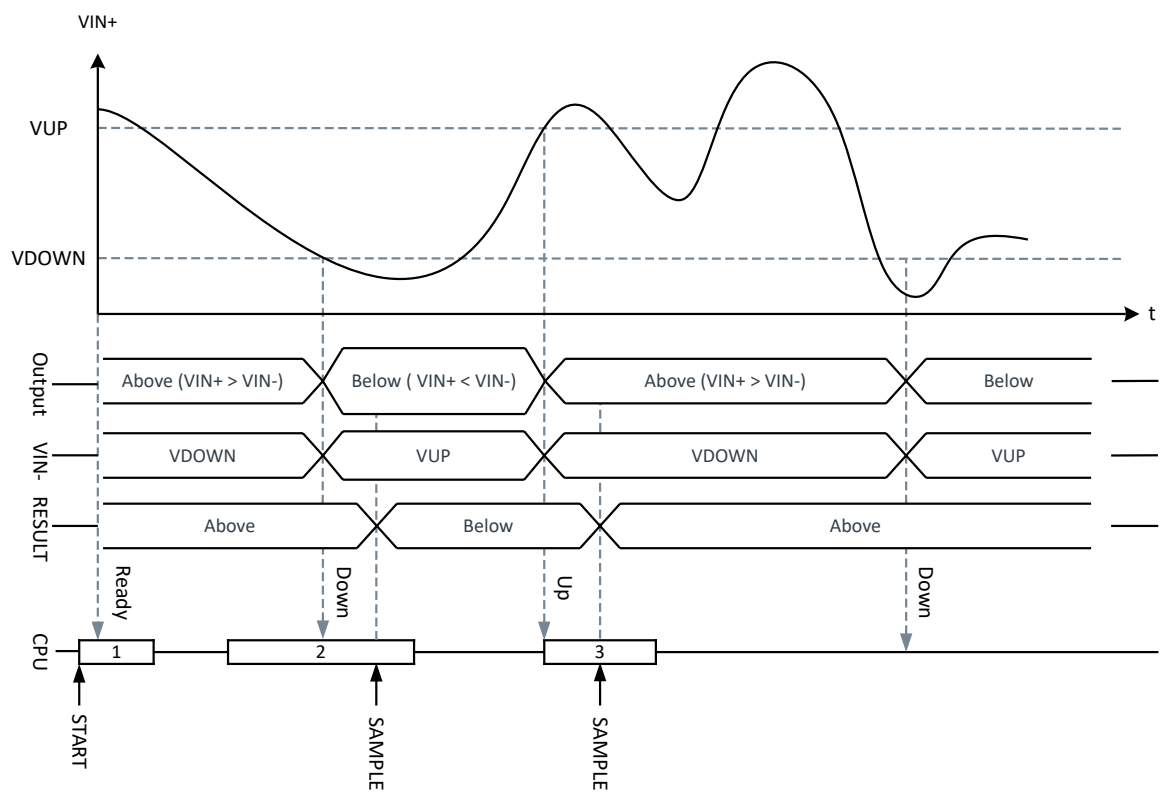


Figure 55: Hysteresis example where V_{IN+} starts above V_{UP}

8.5.7 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
COMP : S	GLOBAL	0x50106000	US	S	NA	No	Comparator COMP
COMP : NS		0x40106000					

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Start comparator
TASKS_STOP	0x004		Stop comparator
TASKS_SAMPLE	0x008		Sample comparator value. This task requires that COMP has been started by the START Task.
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
SUBSCRIBE_SAMPLE	0x088		Subscribe configuration for task SAMPLE
EVENTS_READY	0x100		COMP is ready and output is valid
EVENTS_DOWN	0x104		Downward crossing
EVENTS_UP	0x108		Upward crossing
EVENTS_CROSS	0x10C		Downward or upward crossing
PUBLISH_READY	0x180		Publish configuration for event READY
PUBLISH_DOWN	0x184		Publish configuration for event DOWN
PUBLISH_UP	0x188		Publish configuration for event UP
PUBLISH_CROSS	0x18C		Publish configuration for event CROSS
SHORTS	0x200		Shortcuts between local events and tasks
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
INTPEND	0x30C		Pending interrupts
RESULT	0x400		Compare result
ENABLE	0x500		COMP enable
PSEL	0x504		Pin select
REFSEL	0x508		Reference source select for single-ended mode
EXTREFSEL	0x50C		External reference select
TH	0x530		Threshold configuration for hysteresis unit
MODE	0x534		Mode configuration
HYST	0x538		Comparator hysteresis enable
ISOURCE	0x53C		Current source select on analog input

8.5.7.1 TASKS_START

Address offset: 0x000

Start comparator

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_START						Start comparator																											
			Trigger	1				Trigger task																											

8.5.7.2 TASKS_STOP

Address offset: 0x004

Stop comparator

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_STOP						Stop comparator																											
			Trigger	1				Trigger task																											

8.5.7.3 TASKS_SAMPLE

Address offset: 0x008

Sample comparator value. This task requires that COMP has been started by the START Task.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_SAMPLE						Sample comparator value. This task requires that COMP has been started by the START Task.																											
			Trigger	1				Trigger task																											

8.5.7.4 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task **START**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that task START will subscribe to																																
B	RW	EN																																				
			Disabled	0		Disable subscription																																
			Enabled	1		Enable subscription																																

8.5.7.5 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task **STOP**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																													A	A	A	A	A	A	A	A																							
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that task STOP will subscribe to																																																									
B	RW	EN																																																													
			Disabled	0	Disable subscription																																																										
			Enabled	1	Enable subscription																																																										

8.5.7.6 SUBSCRIBE_SAMPLE

Address offset: 0x088

Subscribe configuration for task **SAMPLE**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that task SAMPLE will subscribe to																																
B	RW	EN																																				
			Disabled	0	Disable subscription																																	
			Enabled	1	Enable subscription																																	

8.5.7.7 EVENTS_READY

Address offset: 0x100

COMP is ready and output is valid

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_READY			COMP is ready and output is valid																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.5.7.8 EVENTS_DOWN

Address offset: 0x104

Downward crossing

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_DOWN						Downward crossing																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.5.7.9 EVENTS_UP

Address offset: 0x108

Upward crossing

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	EVENTS_UP				Upward crossing																													
			NotGenerated	0		Event not generated																													
			Generated	1		Event generated																													

8.5.7.10 EVENTS_CROSS

Address offset: 0x10C

Downward or upward crossing

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_CROSS						Downward or upward crossing																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.5.7.11 PUBLISH_READY

Address offset: 0x180

Publish configuration for event [READY](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				B																								A				A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	CHIDX		[0..255]		DPPI channel that event READY will publish to																																	
B	RW	EN																																					
			Disabled	0		Disable publishing																																	
			Enabled	1		Enable publishing																																	

8.5.7.12 PUBLISH_DOWN

Address offset: 0x184

Publish configuration for event [DOWN](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A												A	A	A	A	A	A	A	A																
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																							
A	RW	CHIDX		[0..255]				DPPI channel that event DOWN will publish to																																																							
B	RW	EN																																																													
			Disabled	0				Disable publishing																																																							
			Enabled	1				Enable publishing																																																							

8.5.7.13 PUBLISH_UP

Address offset: 0x188

Publish configuration for event **UP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																																															
ID				B																																A				A				A				A				A															
Reset 0x00000000				0																																0				0				0				0				0				0				0				0			
ID	R/W	Field	Value ID	Value				Description																																																											
A	RW	CHIDX		[0..255]				DPPI channel that event UP will publish to																																																											
B	RW	EN																																																																	
			Disabled	0				Disable publishing																																																											
			Enabled	1				Enable publishing																																																											

8.5.7.14 PUBLISH_CROSS

Address offset: 0x18C

Publish configuration for event **CROSS**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event CROSS will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.5.7.15 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																																E D C B A			
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY_SAMPLE			Shortcut between event READY and task SAMPLE																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
B	RW	READY_STOP			Shortcut between event READY and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
C	RW	DOWN_STOP			Shortcut between event DOWN and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
D	RW	UP_STOP			Shortcut between event UP and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
E	RW	CROSS_STOP			Shortcut between event CROSS and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														

8.5.7.16 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Enable or disable interrupt for event READY																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
B	RW	DOWN			Enable or disable interrupt for event DOWN																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
C	RW	UP			Enable or disable interrupt for event UP																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
D	RW	CROSS			Enable or disable interrupt for event CROSS																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.5.7.17 INTENSET

Address offset: 0x304

Enable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	READY			Write '1' to enable interrupt for event READY																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
B	RW	DOWN			Write '1' to enable interrupt for event DOWN																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
C	RW	UP			Write '1' to enable interrupt for event UP																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
D	RW	CROSS			Write '1' to enable interrupt for event CROSS																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

8.5.7.18 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Write '1' to disable interrupt for event READY																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	DOWN			Write '1' to disable interrupt for event DOWN																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	UP			Write '1' to disable interrupt for event UP																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	CROSS			Write '1' to disable interrupt for event CROSS																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.5.7.19 INTPEND

Address offset: 0x30C

Pending interrupts

Bit number		31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																			
ID		D C B A																																			
Reset 0x00000000		0 0																																			
ID	R/W	Field	Value ID	Value	Description																																
A	R	READY			Read pending status of interrupt for event READY																																
			NotPending	0	Read: Not pending																																
			Pending	1	Read: Pending																																
B	R	DOWN			Read pending status of interrupt for event DOWN																																
			NotPending	0	Read: Not pending																																
			Pending	1	Read: Pending																																
C	R	UP			Read pending status of interrupt for event UP																																
			NotPending	0	Read: Not pending																																
			Pending	1	Read: Pending																																
D	R	CROSS			Read pending status of interrupt for event CROSS																																
			NotPending	0	Read: Not pending																																
			Pending	1	Read: Pending																																

8.5.7.20 RESULT

Address offset: 0x400

Compare result

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				A																																	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value		Description																															
A	R	RESULT				Result of last compare. Decision point SAMPLE task.																															
			Below	0	Input voltage is below the threshold (VIN+ < VIN-)																																
			Above	1	Input voltage is above the threshold (VIN+ > VIN-)																																

8.5.7.21 ENABLE

Address offset: 0x500

COMP enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ENABLE						Enable or disable COMP																											
			Disabled	0				Disable																											
			Enabled	2				Enable																											

8.5.7.22 PSEL

Address offset: 0x504

Pin select

The pin is selected based on PSEL.PORT

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PIN						Analog pin select																											
B	RW	PORT						GPIO Port selection																											

8.5.7.23 REFSEL

Address offset: 0x508

Reference source select for single-ended mode

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																				A	A	A
Reset 0x00000004				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	REFSEL			Reference select																																	
			Int1V2	0	VREF = internal 1.2 V reference																																	
			VDD	4	VREF = VDD																																	
			AREf	5	VREF = AREF																																	

8.5.7.24 EXTREFSEL

Address offset: 0x50C

External reference select

The external reference pin is selected based on EXTREFSEL.PORT

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																										B	B	B	B				A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																															
A	RW	PIN			External analog reference pin select																															
B	RW	PORT			GPIO Port selection																															

8.5.7.25 TH

Address offset: 0x530

Threshold configuration for hysteresis unit

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
ID																						B	B	B	B	B	B											A	A	A	A	A	A
Reset 0x00002020				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0							
ID	R/W	Field	Value ID	Value		Description																																					
A	RW	THDOWN		[63:0]		VDOWN = (THDOWN+1)/64*VREF																																					
B	RW	THUP		[63:0]		VUP = (THUP+1)/64*VREF																																					

8.5.7.26 MODE

Address offset: 0x534

Mode configuration

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																													B								A		A	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
ID	R/W	Field	Value ID	Value	Description																																			
A	RW	SP			Speed and power modes																																			
			Low	0	Low-power mode																																			
			Normal	1	Normal mode																																			
			High	2	High-speed mode																																			
B	RW	MAIN			Main operation modes																																			
			SE	0	Single-ended mode																																			
			Diff	1	Differential mode																																			

8.5.7.27 HYST

Address offset: 0x538

Comparator hysteresis enable

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				A																																		
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value				Description																														
A	RW	HYST						Comparator hysteresis																														
			NoHyst	0				Comparator hysteresis disabled																														
			Hyst40mV	1				Comparator hysteresis enabled																														

8.5.7.28 ISOURCE

Address offset: 0x53C

Current source select on analog input

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ISOURCE			Current source select on analog input																														
			Off	0	Current source disabled																														
			len2uA5	1	Current source enabled (+/- 2.5 uA)																														
			len5uA	2	Current source enabled (+/- 5 uA)																														
			len10uA	3	Current source enabled (+/- 10 uA)																														

8.6 ECB — AES electronic codebook mode encryption

The AES electronic codebook mode encryption (ECB) can be used for a range of cryptographic functions like hash generation, digital signatures, and keystream generation for data encryption/decryption. The ECB encryption block supports 128 bit AES encryption (encryption only, not decryption).

The main features of ECB are:

- 128-bit AES encryption
- Supports standard AES ECB block encryption
- Memory-to-memory operations using Scatter/Gather DMA

The inputs and outputs of the ECB are illustrated below.

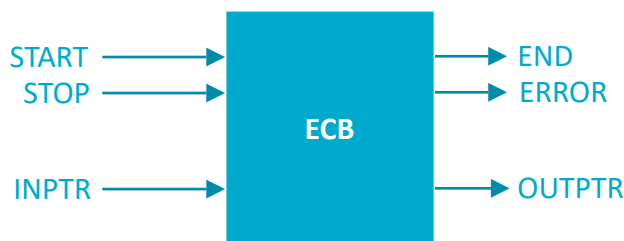


Figure 56: ECB block diagram

AES ECB uses EasyDMA with scatter-gather to access to memory for in-place operations on cleartext and ciphertext during encryption. ECB uses the same AES core as the CCM and AAR blocks and is an asynchronous operation which may not complete if the AES core is busy.

AES ECB performs a 128 bit AES block encrypt. At the **START** task, cleartext is loaded into the ECB from memory described by the scatter/gather job list pointed to by INPTR and the ciphertext is written into memory described by the job list pointed to by OUTPTR. When the last cleartext byte has been encrypted and written to OUTPTR, the **END** event is triggered.

The following figure illustrates how the input and output job lists can be configured. For more details of the joblists, see [EasyDMA](#) on page 264.

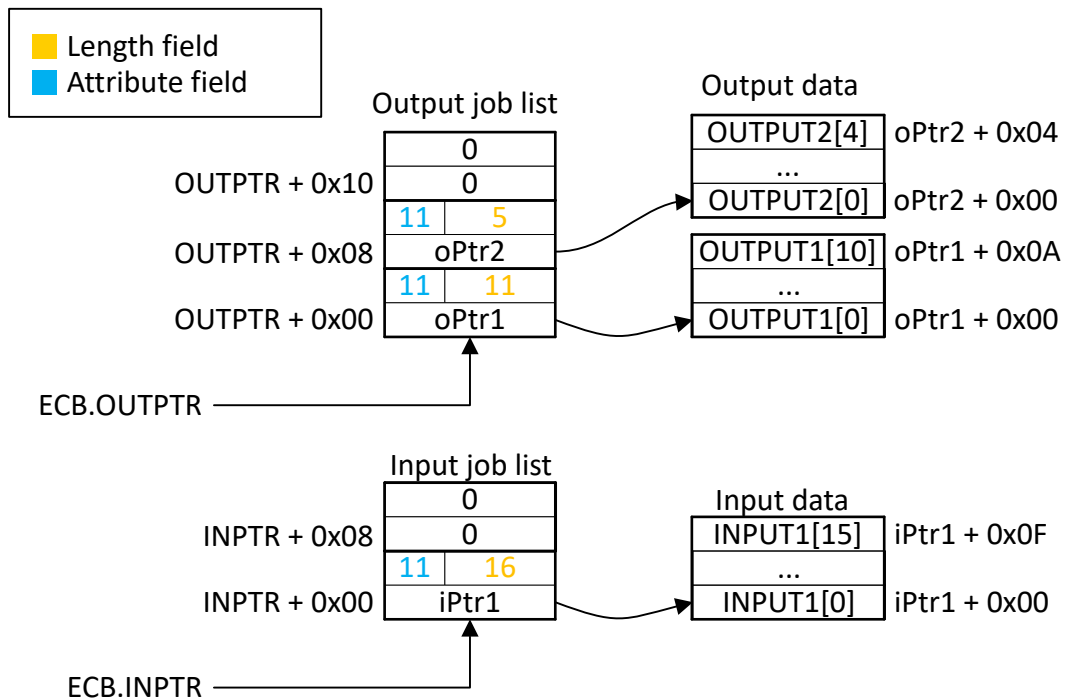


Figure 57: Example job lists for ECB operation

The AES key is set by writing the **KEY.VALUE** key registers. The same key can be used to encrypt multiple blocks by triggering the **START** task multiple times.

AES ECB can be stopped by triggering the **STOP** task.

ECB only supports a single 16-byte block. For different job list sizes the following rules apply:

- A premature end of the input job list is padded to 16 bytes with zeros.
- If more than 16 bytes is supplied as input, then only the first 16 bytes are used.
- For an output job, only the number of bytes specified in the job list are copied to memory.

The 128-bit key in the **KEY.VALUE** registers is stored in reverse byte order relative to the payload. For example, using the sample calculation from the Bluetooth Core Specification v5.4, Volume 6, Part C, chapter 1.1, with the following data:

- Key: 4C68384139F574D836BCF34E9DFB01BF
- Plaintext: 0213243546576879acbdcedfe0f10213
- Expected Encrypted Output: 99ad1b5226a37e3e058e3b8e27c2c666

The **KEY.VALUE** registers are populated as follows:

- **KEY.VALUE[0]** = 0x9DFB01BF
- **KEY.VALUE[1]** = 0x36BCF34E
- **KEY.VALUE[2]** = 0x39F574D8
- **KEY.VALUE[3]** = 0x4C683841

The **IN.PTR** points to a job that contains the following 16-byte input data array:

```
{0x02, 0x13, 0x24, 0x35, 0x46, 0x57, 0x68, 0x79, 0xAC, 0xBD, 0xCE, 0xDF, 0xE0,
0xF1, 0x02, 0x13}
```

Once the encryption is complete, the output buffer referenced by the output job will be filled with the following 16-byte array:

```
{0x99, 0xAD, 0x1B, 0x52, 0x26, 0xA3, 0x7E, 0x3E, 0x05, 0x8E, 0x3B, 0x8E, 0x27,
0xC2, 0xC6, 0x66}
```

Note: The KEY byte order is reversed compared to the NRF52 and NRF53 series devices.

8.6.1 Shared resources

The ECB shares the same AES module as the AAR and CCM peripherals. The ECB will always have lowest priority. If there is a sharing conflict during encryption, the ECB operation will be aborted and an **ERROR** event will be generated.

8.6.2 EasyDMA

This peripheral implements EasyDMA with scatter-gather functionality for reading from and writing to memory without CPU involvement.

The scatter-gather functionality allows EasyDMA to collect data from multiple memory regions, instead of one contiguous block. The memory regions are described by a job list. The job list consists of one or more job entries that consist of a 32-bit address field, 8-bit attribute field, and 24-bit length field. A job list ends with a zero filled job entry. The attribute field must be set to 11.

If INPTR or OUTPTR pointers or the entries in the job lists are not pointing to memory connected to the DMA bus, an EasyDMA transfer may result in a HardFault or memory corruption. See [Memory](#) on page 19 for more information about the different memory regions and DMA connectivity.

The EasyDMA will have finished accessing the RAM when the **END** or **ERROR** events are generated.

For instances supporting DMA error detection, the **ERRORSTATUS** register will report if a bus error has occurred during DMA access. To see if DMA error detection is supported, see the instance's configuration in [Instantiation](#) on page 216.

Example

The figure below shows an example of a job list with three job entries. Each of the entries point to a memory address, and the length field describes how many bytes of data is stored at that address. There are three blocks of memory in use

- FIRSTDATA, an array of length 3
- SECONDDATA, an array of length 2
- THIRDDATA, an array of length 11

The data pointed to from the job list is what is fed into the module and processed according to the peripheral's operation. The entries of the job list comprises pointers to the individual arrays, as well as their sizes. Job entries with length greater than one are processed in little endian order.

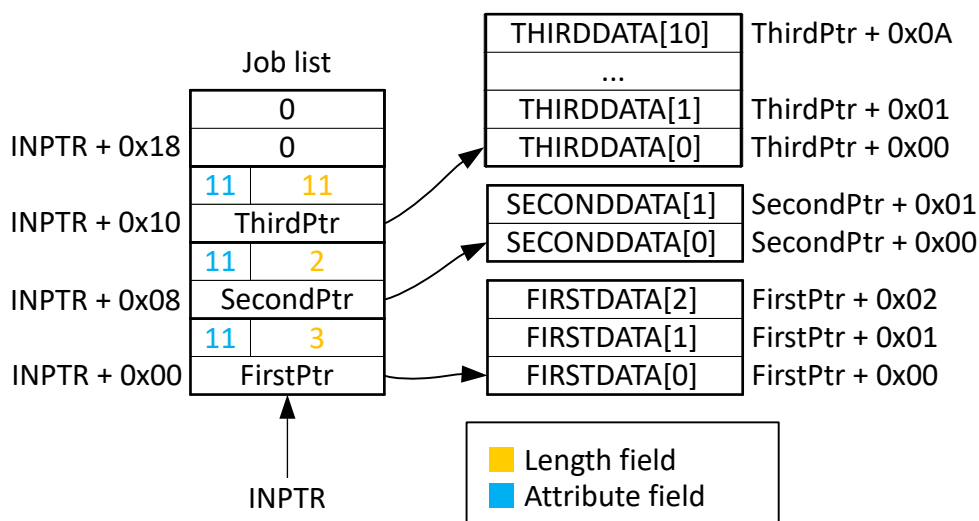


Figure 58: EasyDMA Scatter-Gather job list example

8.6.3 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
ECB00 : S	GLOBAL	0x50047000	US	S	SA	No	AES ECB mode encryption 00
ECB00 : NS		0x40047000					When configuring this peripheral's DMA security using SPU configuration (DMASEC field of SPU->PERIPH[apb_slave_index]), use apb_slave_index 6 (same as AAR00 and CCM00)

Configuration

Instance	Domain	Configuration
ECB00 : S	GLOBAL	
ECB00 : NS		

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Start ECB block encrypt
TASKS_STOP	0x004		Abort a possible executing ECB operation
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
EVENTS_END	0x100		ECB block encrypt complete
EVENTS_ERROR	0x104		ECB block encrypt aborted because of a STOP task or due to an error
PUBLISH_END	0x180		Publish configuration for event END
PUBLISH_ERROR	0x184		Publish configuration for event ERROR
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt

Register	Offset	TZ	Description
ERRORSTATUS	0x400		Error status
KEY.VALUE[n]	0x510		128-bit AES key
IN.PTR	0x530		Input pointer
OUT.PTR	0x538		Output pointer
			Points to a job list containing encrypted ECB data structure

8.6.3.1 TASKS_START

Address offset: 0x000

Start ECB block encrypt

If a crypto operation is already running in the AES core, the START task will not start a new encryption and an ERROR event will be triggered

Bit number		31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
ID			
Reset 0x00000000		0 0	
ID	R/W	Field	Description
A	W	TASKS_START	Start ECB block encrypt
			If a crypto operation is already running in the AES core, the START task will not start a new encryption and an ERROR event will be triggered
		Trigger	1
			Trigger task

8.6.3.2 TASKS_STOP

Address offset: 0x004

Abort a possible executing ECB operation

If a running ECB operation is aborted by STOP, the ERROR event is triggered.

Bit number		31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
ID			
Reset 0x00000000		0 0	
ID	R/W	Field	Description
A	W	TASKS_STOP	Abort a possible executing ECB operation
			If a running ECB operation is aborted by STOP, the ERROR event is triggered.
		Trigger	1
			Trigger task

8.6.3.3 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task [START](#)

If a crypto operation is already running in the AES core, the START task will not start a new encryption and an ERROR event will be triggered

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task START will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.6.3.4 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task **STOP**

If a running ECB operation is aborted by **STOP**, the **ERROR** event is triggered.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task STOP will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.6.3.5 EVENTS_END

Address offset: 0x100

ECB block encrypt complete

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_END			ECB block encrypt complete																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.6.3.6 EVENTS_ERROR

Address offset: 0x104

ECB block encrypt aborted because of a **STOP** task or due to an error

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_ERROR			ECB block encrypt aborted because of a STOP task or due to an error																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.6.3.7 PUBLISH_END

Address offset: 0x180

Publish configuration for event [END](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event END will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.6.3.8 PUBLISH_ERROR

Address offset: 0x184

Publish configuration for event [ERROR](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CHIDX		[0..255]	DPPI channel that event ERROR will publish to																														
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.6.3.9 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	END			Write '1' to enable interrupt for event END																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	ERROR			Write '1' to enable interrupt for event ERROR																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.6.3.10 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	END			Write '1' to disable interrupt for event END																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	ERROR			Write '1' to disable interrupt for event ERROR																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.6.3.11 ERRORSTATUS

Address offset: 0x400

Error status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	ERRORSTATUS			Error status when the ERROR event is generated																														
			NoError	0	No errors have occurred																														
			PrematureInptrEnd	1	End of INPTR job list before data structure was read.																														
			PrematureOutptrEnd	2	End of OUTPTR job list before data structure was read.																														
			EncryptionTooSlow	3	Encryption aborted due to higher priority peripheral requesting or using the AES module.																														
					This enumerator is deprecated.																														
			Aborted	3	Encryption aborted due to higher priority peripheral requesting or using the AES module.																														
			DmaError	4	Bus error during DMA access.																														

8.6.3.12 KEY.VALUE[n] (n=0..3)

Address offset: 0x510 + (n × 0x4)

128-bit AES key

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

8.6.3.13 IN

IN EasyDMA channel

8.6.3.13.1 IN.PTR

Address offset: 0x530

Input pointer

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																								
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A																								
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																								
ID	R/W	Field	Value ID	Value																												Description																											
A	RW	PTR																														Points to a job list containing unencrypted ECB data structure																											

8.6.3.14 OUT

OUT EasyDMA channel

8.6.3.14.1 OUT.PTR

Address offset: 0x538

Output pointer

Points to a job list containing encrypted ECB data structure

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW	PTR										Output pointer																							

8.7 EGU — Event generator unit

Event generator unit (EGU) provides support for interlayer signaling. This means providing support for atomic triggering of both CPU execution and hardware tasks, from both firmware (by CPU) and hardware (by PPI). This feature can be used for triggering CPU execution at a lower priority execution from a higher priority execution, or to handle a peripheral's interrupt service routine (ISR) execution at a lower priority for some of its events. However, triggering any priority from any priority is possible.

Listed here are the main EGU features:

- Software-enabled interrupt triggering
- Separate interrupt vectors for every EGU instance
- Up to 16 separate event flags per interrupt for multiplexing

Each instance of EGU implements a set of tasks which can individually be triggered to generate the corresponding event. For example, the corresponding event for TASKS_TRIGGER[n] is EVENTS_TRIGGERED[n]. See [Instances](#) on page 270 for a list of EGU instances.

8.7.1 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
EGU10 : S	GLOBAL	0x50087000	US	S	NA	No	Event generator unit EGU10
EGU10 : NS		0x40087000					
EGU20 : S	GLOBAL	0x500C9000	US	S	NA	No	Event generator unit EGU20
EGU20 : NS		0x400C9000					

Configuration

Instance	Domain	Configuration
EGU10 : S	GLOBAL	16 events
EGU10 : NS		
EGU20 : S	GLOBAL	6 events
EGU20 : NS		

Register overview

Register	Offset	TZ	Description
TASKS_TRIGGER[n]	0x000		Trigger n for triggering the corresponding TRIGGERED[n] event
SUBSCRIBE_TRIGGER[n]	0x080		Subscribe configuration for task TRIGGER[n]
EVENTS_TRIGGERED[n]	0x100		Event number n generated by triggering the corresponding TRIGGER[n] task
PUBLISH_TRIGGERED[n]	0x180		Publish configuration for event TRIGGERED[n]
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt

8.7.1.1 TASKS_TRIGGER[n] (n=0..15)

Address offset: $0x000 + (n \times 0x4)$

Trigger n for triggering the corresponding TRIGGERED[n] event

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1																															
ID																																			
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	W	TASKS_TRIGGER			Trigger n for triggering the corresponding TRIGGERED[n] event																														
			Trigger	1	Trigger task																														

8.7.1.2 SUBSCRIBE_TRIGGER[n] (n=0..15)

Address offset: $0x080 + (n \times 0x4)$

Subscribe configuration for task TRIGGER[n]

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1																												
ID				B																								A				A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																								
A	RW	CHIDX		[0..255]		DPPI channel that task TRIGGER[n] will subscribe to																																																								
B	RW	EN																																																												
			Disabled	0		Disable subscription																																																								
			Enabled	1		Enable subscription																																																								

8.7.1.3 EVENTS_TRIGGERED[n] (n=0..15)

Address offset: $0x100 + (n \times 0x4)$

Event number n generated by triggering the corresponding TRIGGER[n] task

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_TRIGGERED			Event number n generated by triggering the corresponding TRIGGER[n] task																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.7.1.4 PUBLISH_TRIGGERED[n] (n=0..15)

Address offset: $0x180 + (n \times 0x4)$

Publish configuration for event TRIGGERED[n]

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0																																		
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that event TRIGGERED[n] will publish to																																
B	RW	EN																																				
			Disabled	0		Disable publishing																																
			Enabled	1		Enable publishing																																

8.7.1.5 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A-P	RW	TRIGGERED[i] (i=0..15)						Enable or disable interrupt for event TRIGGERED[i]																											
			Disabled	0				Disable																											
			Enabled	1				Enable																											

8.7.1.6 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-P	RW	TRIGGERED[i] (i=0..15)			Write '1' to enable interrupt for event TRIGGERED[i]																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.7.1.7 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																				P O N M L K J I H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-P	RW	TRIGGERED[i] (i=0..15)			Write '1' to disable interrupt for event TRIGGERED[i]																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.8 GPIO — General purpose input/output

The general purpose input/output (GPIO) pins are grouped as one or more ports, with each port having up to 32 GPIO pins.

The number of ports and GPIO pins per port varies with product variant and package. Refer to [Registers](#) on page 278 and [Pin assignments](#) on page 793 for more information about the number of GPIO pins that are supported.

GPIO has the following user-configurable features:

- Configurable output drive strength
- Internal pull-up and pull-down resistors
- Wake-up from high or low level triggers on all pins in PERI and LP power domains
- Trigger interrupt on state changes on any pin
- All pins can be used by the PPI task/event system
- One or more GPIO outputs can be controlled through PPI and GPIOTE channels
- All pins can be individually mapped to interface blocks for layout flexibility
- GPIO state changes captured on SENSE signal can be stored by LATCH register
- Support for secure and non-secure attributes for pins in conjunction with the system protection unit ([SPU — System protection unit](#) on page 180)

The following figure illustrates the GPIO port containing 32 individual pins, where PIN0 is illustrated in more detail as a reference. All signals on the left side in the illustration are used by other peripherals in the system and therefore not directly available to the CPU.

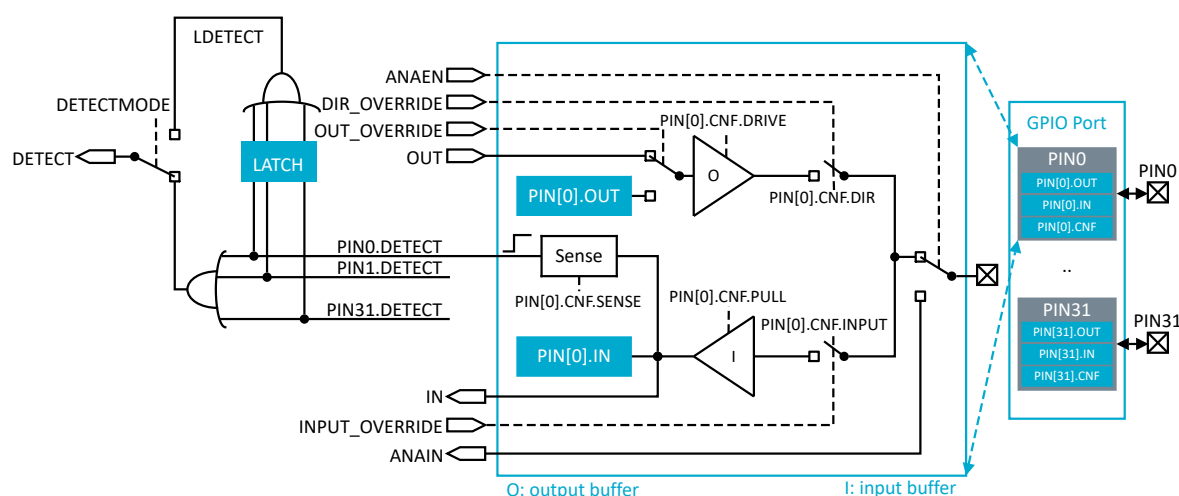


Figure 59: GPIO port and the GPIO pin details

8.8.1 Pin configuration

The GPIO port peripheral implements up to 32 pins, $PIN[n]$ ($n = 0..31$), that can be individually configured in the $PIN_CNF[n]$ registers ($n=0..31$).

The following parameters can be enabled or configured in these registers:

- Direction
- Drive strength
- Pull-up and pull-down resistors
- Pin sensing
- Input buffer disconnect
- Analog input (for selected pins)

All write-capable registers are retained registers. See [POWER — Power control](#) on page 97 for more information.

When not used as an input, disconnect the input buffer of the GPIO pin to save power. An input must be connected to get a valid value in the [IN](#) register and for the sense mechanism to have access to the pin.

Other peripherals in the system can connect to GPIO pins to override their output value, override their configuration, or read their analog or digital input value.

Selected pins also support analog input signals (ANAIN). The assignment of the analog pins can be found in [Pin assignments](#) on page 793.

GPIO drive strength is configured using the `DRIVE0` and `DRIVE1` fields of register [PIN_CNF\[n\] \(n=0..31\) \(Retained\)](#) on page 281. Some pins may not support every drive configuration, see [Pin assignments](#) on page 793 for more information.

When a pin is configured as digital input, it is important to minimize increased current consumption when the input voltage is between V_{IL} and V_{IH} . It is a good practice to ensure that the external circuitry does not drive the pin to levels between V_{IL} and V_{IH} for a long period of time.

For more information on pin assignment and the corresponding effect of read and write operations of GPIO registers, see [Peripheral and subsystem assignment](#) on page 276.

Note: NFCT uses two pins to connect to the antenna, which are shared with GPIOs. NFC pins are enabled from reset. To use them as GPIO pins, NFC use must be disabled using register [PADCONFIG](#) on page 388. For more details, see [NFCT — Near field communication tag](#) on page 350.

8.8.2 Pin sense mechanism

Pin sensitivity can be individually configured through the `SENSE` field in the $PIN_CNF[n]$ register to detect a high level or a low level on their input. When the correct level is detected, the sense mechanism will set the `DETECT` signal `high`. Each pin has a separate `DETECT` signal.

The default behavior for the `DETECT` signal is defined by the register [DETECTMODE](#). By default, the `DETECT` signals from all pins in the GPIO port are combined into one common `DETECT` signal that is routed throughout the system, and can be utilized by other peripherals. This mechanism is functional in both System ON and System OFF modes. The `DETECTMODE` applies to both secure and non-secure pins.

Pins must be in a level that cannot trigger the sense mechanism before enabling it. When the sense mechanism is enabled, the `DETECT` signal will immediately go high if the `SENSE` condition configured in the PIN_CNF registers is met. This will trigger a `PORT` event if the `DETECT` signal was low before enabling the sense mechanism.

The `DETECT` signal is used by the power and clock management system to exit from System OFF mode, and by the `GPIOE` peripheral to allow pins to generate events and interrupts.

When a pin's PINx.DETECT signal goes high, a flag will be set in the register **LATCH**. For example, when the PIN0.DETECT signal goes high, bit 0 in the register **LATCH** will be set to 1. If the CPU performs a clear operation on a bit in the register **LATCH** when the associated PINx.DETECT signal is high, the bit in the register **LATCH** will not be cleared. The register **LATCH** will only be cleared if the CPU explicitly clears it by writing a 1 to the bit to be cleared. This means the register **LATCH** will not be affected by a PINx.DETECT signal being set low.

The LATCH register has split security. Non-secure code can only read the state of the non-secure pins, while the secure pins read as 0. Secure code is able to read the state of all pins.

The LDETECT signal will be set high when one or more bits in the register **LATCH** are 1. The LDETECT signal will be set low when all bits in the register **LATCH** are successfully cleared to 0.

If one or more bits in the register **LATCH** are 1 after the CPU has performed a clear operation, a rising edge will be generated on the LDETECT signal. This is illustrated in **DETECT signal behavior** on page 275.

Note: The CPU can read the register **LATCH** at any time to check if a SENSE condition has been met on one or more of the GPIO pins. This is true even if that condition is no longer met at the time the CPU queries the register **LATCH**. This mechanism will work even if the LDETECT signal is not used as the DETECT signal.

LDETECT is enabled using the DETECTMODE register. See **GPIO port and the GPIO pin details** on page 273.

The following figure illustrates the DETECT signal behavior for these two alternatives.

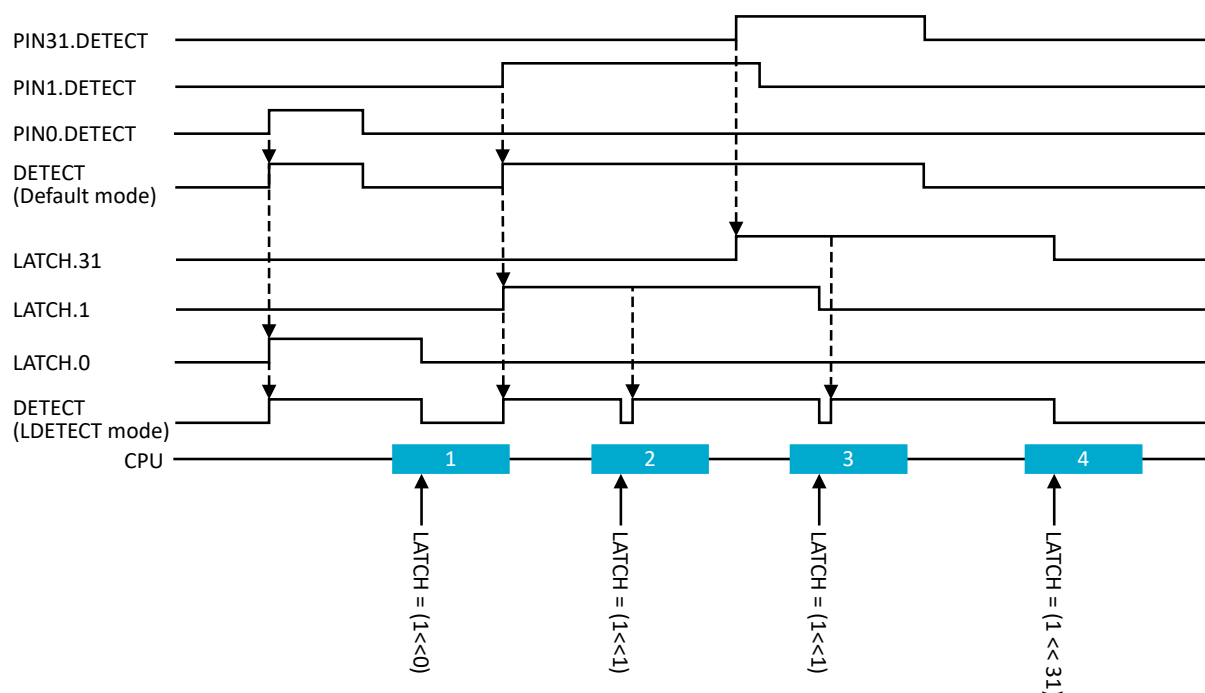


Figure 60: DETECT signal behavior

8.8.3 Port capabilities

The device power domains have their own GPIO ports with different capabilities.

The following is a list of all GPIO ports (P[n]) in the system.

- **P0 low-power domain** – These I/O pins can wake the system up from System OFF or System ON sleep, and can be accessed by all peripherals in the low-power domain.

- P1 peripheral domain – These I/O pins can wake the system up from System OFF or System ON sleep, and can be accessed by all peripherals in the peripheral domain.
- P2 MCU domain – These I/O pins are faster and can be used for high-speed signals such as trace or fast serial peripheral communication. GPIO P2 cannot wake the system from sleep. P2 does not have the GPIO SENSE or DETECT mechanism, or GPIOTE.

Peripherals must use pins in their own domain. However, some P2 pins can be used for select serial interfaces in the peripheral domain. This is not the most power-efficient way of connecting these serial interfaces, but adds flexibility when designing a circuit board. These pins must be configured and used only for the function listed in the pin assignments table, see [Pin assignments](#) on page 793. When setting up the peripheral's PSEL registers, it must be connected to the corresponding function listed in the pin assignments table, e.g. a UARTE TXD pin must be configured in a PSEL.TXD register.

Peripherals cannot mix pins from different ports. All pins must be on the same port.

The following table lists the port special functions and characteristics.

Port	Wakeup source	Extra drive strength (E0E1)	Pin sense/detect	GPIOTE	Maximum speed [MHz]
P0	Yes	No	Yes	Yes	8
P1	Yes	No	Yes	Yes	8
P2	No	Yes	No	No	64

Table 35: Port capabilities

In addition to the capabilities of the port, some specific pins have additional functions. These are listed in [Pin assignments](#) on page 793.

8.8.4 Peripheral and subsystem assignment

System GPIO pins can be allocated to peripherals with dedicated pins or subsystems such as trace and debug.

The pins of the system are listed in [Pin assignments](#) on page 793.

A pin can be assigned to any of the following:

- GPIO or peripheral with PSEL registers
- Peripheral with dedicated pins (VPR and GRTC)
- Trace and debug (TND) subsystem

By default, all pins are assigned to GPIO or peripherals with PSEL registers. This is the default value of the CTRLSEL value in register [PIN_CNF\[n\] \(n=0..31\) \(Retained\)](#) on page 281.

To allocate a pin to a peripheral or subsystem with dedicated pins, such as GRTC or TND, change the CTRLSEL value in register [PIN_CNF\[n\] \(n=0..31\) \(Retained\)](#) on page 281 for that pin. This will connect the pin to the subsystem or peripheral.

Only the peripheral or subsystem where the pin was allocated can observe and control that pin's state. Reading a pin that is not allocated to the current subsystem will return zero, and writes will be ignored. If a pin is allocated to a subsystem that cannot access it, the pin stays under control of the GPIO peripheral.

When CTRLSEL is used to allocate a peripheral or subsystem, reading the GPIO peripheral registers will not reveal the state of the pins.

The following figure illustrates how to assign a pin to a peripheral that has dedicated pins, or a subsystem such as trace and debug.

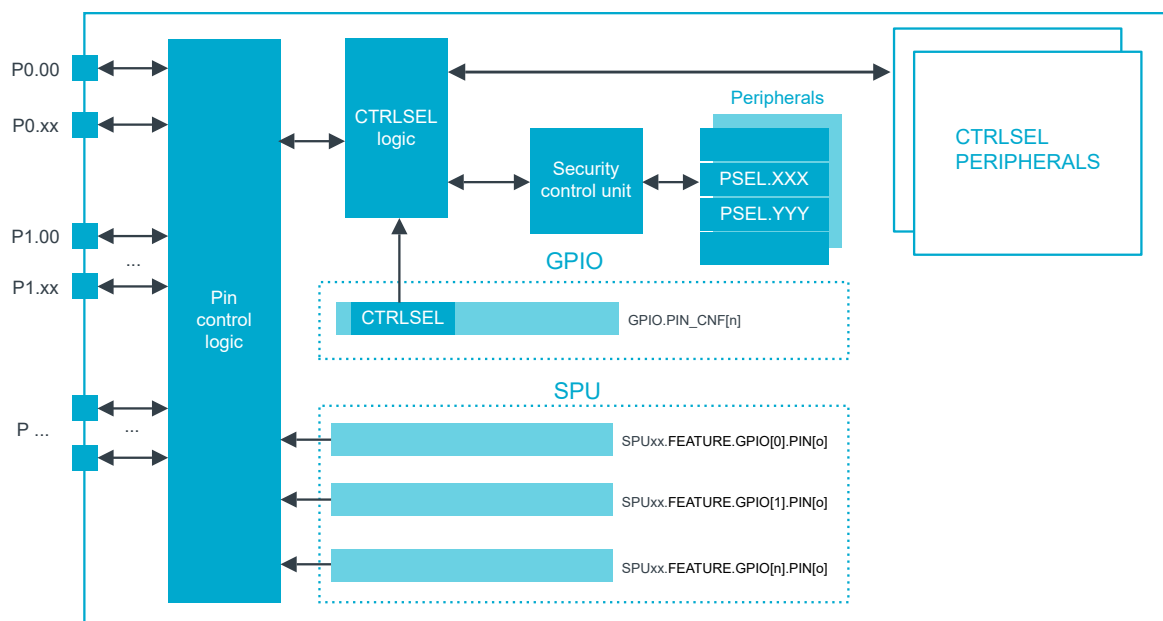


Figure 61: Pin access using CTRLSEL

For details on pin security, see [Security](#) on page 126.

Note: CTRLSEL must be configured before any pins are used, otherwise glitches on the GPIO pins of the corresponding port can occur.

8.8.5 Clock pins

The device has dedicated clock pins. Some peripherals, such as SPI, TWI, and TRACE, have clock signals.

The dedicated clock pins are optimized to ensure correct timing between the clock and data signals for these peripherals. All peripherals that have clock signals must use these pins. See [Pin assignments](#) on page 793 for the full list.

The data signal associated with the peripheral must use pins close to the clock pin. This ensures that the internal paths from the peripheral to the pin have the same delay, so that the data and clock signals reach the pins at the same time.

For high-speed signals, the printed circuit board (PCB) layout must use short PCB traces of identical length. This reduces delays and ensures the same delay on the clock and data path.

Note: TWIM and TWIS must use clock pins for both SDA and SCL.

8.8.6 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
P2 : S P2 : NS	GLOBAL	0x50050400 0x40050400	US	S	NA	Yes	General purpose input and output, port P2 Does not support pin sense mechanism, and DETECTMODE register has no effect. Supports extra high drive (DRIVE0=E0, DRIVE1=E1).
P1 : S P1 : NS		0x500D8200 0x400D8200					General purpose input and output, port P1
P0 : S P0 : NS	GLOBAL	0x5010A000 0x4010A000	US	S	NA	Yes	General purpose input and output, port P0

Configuration

Instance	Domain	Configuration
P2 : S P2 : NS	GLOBAL	I/O pins on this port does not have pin sense mechanism P2 has 11 pins, P2.00 through P2.10.
P1 : S P1 : NS		I/O pins on this port have pin sense mechanism P1 has 17 pins, P1.00 through P1.16.
P0 : S P0 : NS	GLOBAL	I/O pins on this port have pin sense mechanism P0 has 7 pins, P0.00 through P0.06.

Register overview

Register	Offset	TZ	Description
OUT	0x000		Write GPIO port This register is retained.
OUTSET	0x004		Set individual bits in GPIO port
OUTCLR	0x008		Clear individual bits in GPIO port
IN	0x00C		Read GPIO port
DIR	0x010		Direction of GPIO pins This register is retained.
DIRSET	0x014		DIR set register
DIRCLR	0x018		DIR clear register
LATCH	0x020		Latch register indicating what GPIO pins that have met the criteria set in the PIN_CNF[n].SENSE registers This register is retained.
DETECTMODE	0x024	S	Select between default DETECT signal behavior and LDETECT mode This register is retained.
PIN_CNF[n]	0x080		Pin n configuration of GPIO pin This register is retained.

8.8.6.1 OUT (Retained)

Address offset: 0x000

Write GPIO port

This register is retained.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				f e d c b a Z Y X W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A-f	RW	PIN[i] (i=0..31)				Pin i																													
			Low	0	Pin driver is low																														
			High	1	Pin driver is high																														

8.8.6.2 OUTSET

Address offset: 0x004

Set individual bits in GPIO port

Note: Read: reads value of OUT register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				f e d c b a Z Y X W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A-f	RW	PIN[i] (i=0..31) W1S				Pin i																													
			Low	0	Read: pin driver is low																														
			High	1	Read: pin driver is high																														
			Set	1	Write: writing a '1' sets the pin high; writing a '0' has no effect																														

8.8.6.3 OUTCLR

Address offset: 0x008

Clear individual bits in GPIO port

Note: Read: reads value of OUT register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				f e d c b a Z Y X W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-f	RW	PIN[i] (i=0..31) W1C			Pin i																														
			Low	0	Read: pin driver is low																														
			High	1	Read: pin driver is high																														
			Clear	1	Write: writing a '1' sets the pin low; writing a '0' has no effect																														

8.8.6.4 IN

Address offset: 0x00C

Read GPIO port

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				f e d c b a Z Y X W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A-f	R	PIN[i] (i=0..31)						Pin i																											
			Low	0				Pin input is low																											
			High	1				Pin input is high																											

8.8.6.5 DIR (Retained)

Address offset: 0x010

Direction of GPIO pins

This register is retained.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				f e d c b a Z Y X W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A-f	RW	PIN[i] (i=0..31)						Pin i																											
			Input	0				Pin set as input																											
			Output	1				Pin set as output																											

8.8.6.6 DIRSET

Address offset: 0x014

DIR set register

Note: Read: reads value of DIR register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				f e d c b a Z Y X W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A-f	RW	PIN[i] (i=0..31)						Set as output pin i																											
		W1S																																	
			Input	0				Read: pin set as input																											
			Output	1				Read: pin set as output																											
			Set	1				Write: writing a '1' sets pin to output; writing a '0' has no effect																											

8.8.6.7 DIRCLR

Address offset: 0x018

DIR clear register

Note: Read: reads value of DIR register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				f	e	d	c	b	a	Z	Y	X	W	V	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																													
A-f	RW	PIN[i] (i=0..31)	W1C			Set as input pin i																													
				Input	0	Read: pin set as input																													
				Output	1	Read: pin set as output																													
				Clear	1	Write: writing a '1' sets pin to input; writing a '0' has no effect																													

8.8.6.8 LATCH (Retained)

Address offset: 0x020

Latch register indicating what GPIO pins that have met the criteria set in the PIN_CNF[n].SENSE registers

This register is retained.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				f	e	d	c	b	a	Z	Y	X	W	V	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A			
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A-f	RW	PIN[i] (i=0..31)				Status on whether PINi has met criteria set in PIN_CNF[i].SENSE register.																																
						Write '1' to clear.																																
			NotLatched	0	Criteria has not been met																																	
			Latched	1	Criteria has been met																																	

8.8.6.9 DETECTMODE (Retained)

Address offset: 0x024

Select between default DETECT signal behavior and LDETECT mode

DETECTMODE applies to both secure (DETECT_SEC) and non-secure pins (DETECT_NONSEC)

This register is retained.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	DETECTMODE			Select between default DETECT signal behavior and LDETECT mode																															
			Default	0	DETECT directly connected to PIN DETECT signals																															
			LDETECT	1	Use the latched LDETECT behavior																															

8.8.6.10 PIN_CNF[n] (n=0..31) (Retained)

Address offset: 0x080 + (n × 0x4)

Pin n configuration of GPIO pin

This register is retained.

8.9 GPIOTE — GPIO tasks and events

The main features of GPIOTE are:

- A GPIOTE block enables GPIOs to generate events on pin state change which can be used to carry out tasks through the PPI system. A GPIO can also be driven to change state on system events using the PPI system.

Tasks and events are briefly introduced in [Peripheral interface](#) on page 213, and GPIO is described in more detail in [GPIO — General purpose input/output](#) on page 273.

Low power detection of pin state changes is possible when in System ON or System OFF.

GPIOE supports split-security. Each channel is assigned a security state (S/NS).

Up to three tasks can be used in each GPIOE channel for performing write operations to a pin. Two tasks are fixed (SET and CLR), and one (OUT) is configurable to perform following operations:

- Set
- Clear
- Toggle

An event can be generated in each GPIOE channel from one of the following input conditions:

- Rising edge
- Falling edge
- Any change

8.9.1 Pin events and tasks

The GPIOE module has a number of tasks and events that can be configured to operate on individual GPIO pins.

The tasks SET[n], CLR[n], and OUT[n] can write to individual pins, and events IN[n] can be generated from input changes of individual pins.

The SET task will set the pin selected in [CONFIG\[n\].PSEL](#) to high. The CLR task will set the pin low.

The effect of the OUT task on the pin is configurable in [CONFIG\[n\].POLARITY](#). It can set the pin high, set it low, or toggle it.

Tasks and events are configured using the [CONFIG\[n\]](#) registers. One [CONFIG\[n\]](#) register is associated with a set of SET[n], CLR[n], and OUT[n] tasks and IN[n] events.

As long as a SET[n], CLR[n], and OUT[n] task or an IN[n] event is configured to control pin *n*, the pin's output value will only be updated by the GPIOE module. The pin's output value, as specified in the GPIO, will therefore be ignored as long as the pin is controlled by GPIOE. Attempting to write to the pin as a normal GPIO pin will have no effect. When the GPIOE is disconnected from a pin, the associated pin gets the output and configuration values specified in the GPIO module, see *MODE* field in [CONFIG\[n\]](#) register.

When conflicting tasks are triggered simultaneously (i.e. during the same clock cycle) in one channel, the priority of the tasks is as described in the following table.

Priority	Task
1	OUT
2	CLR
3	SET

Table 36: Task priorities

When setting the [CONFIG\[n\]](#) registers, *MODE*=Disabled does not have the same effect as *MODE*=Task and *POLARITY*=None. In the latter case, a CLR or SET task occurring at the exact same time as OUT will end up with no change on the pin, based on the priorities described in the table above.

When a GPIOE channel is configured to operate on a pin as a task, the initial value of that pin is configured in the *OUTINIT* field of [CONFIG\[n\]](#).

8.9.2 Port event

PORT is an event that can be generated from multiple input pins using the GPIO DETECT signal.

The event will be generated on the rising edge of the DETECT signal. See [GPIO — General purpose input/output](#) on page 273 for more information about the DETECT signal.

There are two DETECT signals that come from the [GPIO](#) peripheral, the secure DETECT_SEC for pins marked as secure and the non-secure DETECT_NONSEC. Each signal has a corresponding port event, [EVENTS_PORT\[n\].SECURE](#) and [EVENTS_PORT\[n\].NONSECURE](#). Secure events are not accessible from the non-secure side.

The GPIO DETECT signal will not wake the system up again if the system is put into System ON IDLE while the DETECT signal is high. Make sure to clear all DETECT sources before entering sleep. If the LATCH register is used as a source, a new rising edge will be generated on DETECT if any bit in LATCH is still high after clearing all or part of the register. This could occur if one of the PINx.DETECT signals is still high, for example. See [Pin sense mechanism](#) on page 274 for more information.

Setting the system to System OFF while DETECT is high will cause a wakeup from System OFF reset.

This feature can be used to wake up the CPU from a WFI or WFE type sleep in System ON when all peripherals and the CPU are idle, meaning the lowest power consumption in System ON mode.

In order to prevent spurious interrupts from the PORT event while configuring the sources, the following must be performed:

1. Disable interrupts on the PORT event (through [INTENCLR.PORT](#)).
2. Configure the sources ([PIN_CNF\[n\].SENSE](#) in [GPIO](#)).
3. Clear any potential event that could have occurred during configuration (write '0' to [EVENTS_PORT](#)).
4. Enable interrupts (through [INTENSET.PORT](#)).

8.9.3 Tasks and events pin configuration

Each GPIOTE channel is associated with one physical GPIO pin through the [CONFIG.PSEL](#) field.

When Event mode is selected in [CONFIG.MODE](#), the pin specified by [CONFIG.PSEL](#) will be configured as an input, overriding the DIR setting in GPIO. Similarly, when Task mode is selected in [CONFIG.MODE](#), the pin specified by [CONFIG.PSEL](#) will be configured as an output overriding the DIR setting and OUT value in GPIO. When Disabled is selected in [CONFIG.MODE](#), the pin specified by [CONFIG.PSEL](#) will use its configuration from the [PIN\[n\].CNF](#) registers in GPIO.

For the range of possible [CONFIG.PSEL](#) and [CONFIG.PORT](#) values in the product, see GPIO mapping. Writing other values may lead to undefined behavior.

Note: A pin can only be assigned to one GPIOTE channel at a time. Failing to do so may result in unpredictable behavior.

8.9.4 Split security attribute

Individual GPIOTE channels and interrupts can have independent security attributes.

GPIOTE is implemented with split security, meaning it handles accesses from both secure and non-secure code. GPIOTE channels and interrupts can be defined as secure or non-secure.

For more information on GPIOTE security attributes, see [GPIOTE](#) on page 134.

Security attribute of GPIOTE interrupts are fixed and cannot be modified. For information on how to configure security attributes of GPIOTE channels and interrupts, see UICR.

8.9.5 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
GPIOTE20 : S	GLOBAL	0x500DA000	US	S	NA	Yes	8 channels and 2 interrupts for GPIO port P1
GPIOTE20 : NS		0x400DA000					GPIO tasks and events GPIOTE20
GPIOTE30 : S	GLOBAL	0x5010C000	US	S	NA	Yes	4 channels and 2 interrupts for GPIO port P0
GPIOTE30 : NS		0x4010C000					GPIO tasks and events GPIOTE30

Configuration

Instance	Domain	Configuration
GPIOTE20 : S	GLOBAL	Number of GPIOTE channels: 0..7
GPIOTE20 : NS		Number of GPIOTE port events: 0..0
		Number of GPIOTE interrupts: 0..1
GPIOTE30 : S	GLOBAL	Number of GPIOTE channels: 0..3
GPIOTE30 : NS		Number of GPIOTE port events: 0..0
		Number of GPIOTE interrupts: 0..1

Register overview

Register	Offset	TZ	Description
TASKS_OUT[n]	0x000		Task for writing to pin specified in CONFIG[n].PSEL. Action on pin is configured in CONFIG[n].POLARITY.
TASKS_SET[n]	0x030		Task for writing to pin specified in CONFIG[n].PSEL. Action on pin is to set it high.
TASKS_CLR[n]	0x060		Task for writing to pin specified in CONFIG[n].PSEL. Action on pin is to set it low.
SUBSCRIBE_OUT[n]	0x080		Subscribe configuration for task OUT[n]
SUBSCRIBE_SET[n]	0x0B0		Subscribe configuration for task SET[n]
SUBSCRIBE_CLR[n]	0x0E0		Subscribe configuration for task CLR[n]
EVENTS_IN[n]	0x100		Event from pin specified in CONFIG[n].PSEL
EVENTS_PORT[n].NONSECURE	0x140	NS	Non-secure port event from owner n
EVENTS_PORT[n].SECURE	0x144	S	Secure port event from owner n
PUBLISH_IN[n]	0x180		Publish configuration for event IN[n]
PUBLISH_PORT[n].NONSECURE	0x1C0	NS	Publish configuration for event PORT[n].NONSECURE
PUBLISH_PORT[n].SECURE	0x1C4	S	Publish configuration for event PORT[n].SECURE
INTENSET0	0x304		Enable interrupt
INTENCLR0	0x308		Disable interrupt
INTENSET1	0x314		Enable interrupt
INTENCLR1	0x318		Disable interrupt
CONFIG[n]	0x510		Configuration for OUT[n], SET[n], and CLR[n] tasks and IN[n] event

8.9.5.1 TASKS_OUT[n] (n=0..7)

Address offset: 0x000 + (n × 0x4)

Task for writing to pin specified in CONFIG[n].PSEL. Action on pin is configured in CONFIG[n].POLARITY.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	W	TASKS_OUT			Task for writing to pin specified in CONFIG[n].PSEL. Action on pin is configured in CONFIG[n].POLARITY.																														
			Trigger	1	Trigger task																														

8.9.5.2 TASKS_SET[n] (n=0..7)

Address offset: 0x030 + (n × 0x4)

Task for writing to pin specified in CONFIG[n].PSEL. Action on pin is to set it high.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	W	TASKS_SET			Task for writing to pin specified in CONFIG[n].PSEL. Action on pin is to set it high.																														
			Trigger	1	Trigger task																														

8.9.5.3 TASKS_CLR[n] (n=0..7)

Address offset: 0x060 + (n × 0x4)

Task for writing to pin specified in CONFIG[n].PSEL. Action on pin is to set it low.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	TASKS_CLR			Task for writing to pin specified in CONFIG[n].PSEL. Action on pin is to set it low.																															
			Trigger	1	Trigger task																															

8.9.5.4 SUBSCRIBE_OUT[n] (n=0..7)

Address offset: 0x080 + (n × 0x4)

Subscribe configuration for task OUT[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CHIDX		[0..255]	DPPI channel that task OUT[n] will subscribe to																														
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.9.5.5 SUBSCRIBE_SET[n] (n=0..7)

Address offset: 0x0B0 + (n × 0x4)

Subscribe configuration for task SET[n]

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															

8.9.5.6 SUBSCRIBE_CLR[n] (n=0..7)

Address offset: 0x0E0 + (n × 0x4)

Subscribe configuration for task CLR[n]

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that task CLR[n] will subscribe to																																																									
B	RW	EN																																																													
			Disabled	0	Disable subscription																																																										
			Enabled	1	Enable subscription																																																										

8.9.5.7 EVENTS_IN[n] (n=0..7)

Address offset: 0x100 + (n × 0x4)

Event from pin specified in CONFIG[n].PSEL

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_IN			Event from pin specified in CONFIG[n].PSEL																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.9.5.8 EVENTS_PORT[n] (n=0..0)

Peripheral events.

8.9.5.8.1 EVENTS_PORT[n].NONSECURE (n=0..0)

Address offset: 0x140 + (n × 0x8)

Non-secure port event from owner n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	TZ	Field	Value ID	Value	Description																													
A	RW	NS	NONSECURE			Non-secure port event from owner n																													
				NotGenerated	0	Event not generated																													
				Generated	1	Event generated																													

8.9.5.8.2 EVENTS_PORT[n].SECURE (n=0..0)

Address offset: $0x144 + (n \times 0x8)$

Secure port event from owner n

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	TZ	Field	Value ID	Value	Description																																
A	RW	S	SECURE			Secure port event from owner n																																
				NotGenerated	0	Event not generated																																
				Generated	1	Event generated																																

8.9.5.9 PUBLISH_IN[n] (n=0..7)

Address offset: $0x180 + (n \times 0x4)$

Publish configuration for event IN[n]

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value				Description																														
A	RW	CHIDX		[0..255]				DPPI channel that event IN[n] will publish to																														
B	RW	EN																																				
			Disabled	0				Disable publishing																														
			Enabled	1				Enable publishing																														

8.9.5.10 PUBLISH_PORT[n] (n=0..0)

Publish configuration for events

8.9.5.10.1 PUBLISH_PORT[n].NONSECURE (n=0..0)

Address offset: $0x1C0 + (n \times 0x8)$

Publish configuration for event PORT[n].NONSECURE

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																							
A	RW	CHIDX		[0..255]				DPPI channel that event PORT[n].NONSECURE will publish to																																																							
B	RW	EN																																																													
			Disabled	0				Disable publishing																																																							
			Enabled	1				Enable publishing																																																							

8.9.5.10.2 PUBLISH_PORT[n].SECURE (n=0..0)

Address offset: $0x1C4 + (n \times 0x8)$

Publish configuration for event PORT[n].SECURE

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event PORT[n].SECURE will publish to																													
B	RW	EN																																	
			Disabled	0		Disable publishing																													
			Enabled	1		Enable publishing																													

8.9.5.11 INTENSET0

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				J I																H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	TZ	Field	Value ID	Value	Description																													
A-H	RW		IN[i] (i=0..7)			Write '1' to enable interrupt for event IN[i]																													
				Set	1	Enable																													
				Disabled	0	Read: Disabled																													
				Enabled	1	Read: Enabled																													
I	RW	NS	PORTONONSECURE			Write '1' to enable interrupt for event PORTONONSECURE																													
				Set	1	Enable																													
				Disabled	0	Read: Disabled																													
				Enabled	1	Read: Enabled																													
J	RW	S	PORTOSECURE			Write '1' to enable interrupt for event PORTOSECURE																													
				Set	1	Enable																													
				Disabled	0	Read: Disabled																													
				Enabled	1	Read: Enabled																													

8.9.5.12 INTENCLR0

Address offset: 0x308

Disable interrupt

 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|--|--|--|--|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--
--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																				
Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
ID					J																I																H																G																F																E																D																C																B																A																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																													
Reset 0x00000000					0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																													
																										0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0								
																							0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0											
																				0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0														
																	0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																	
														0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																				

8.9.5.13 INTENSET1

Address offset: 0x314

Enable interrupt

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					J I																H G F E D C B A															
Reset 0x00000000					0 0																															
D	R/W	TZ	Field	Value ID	Value	Description																														
A-H	RW		IN[i] (i=0..7)			Write '1' to enable interrupt for event IN[i]																														
				Set	1	Enable																														
				Disabled	0	Read: Disabled																														
				Enabled	1	Read: Enabled																														
I	RW	NS	PORTONONSECURE			Write '1' to enable interrupt for event PORTONONSECURE																														
				Set	1	Enable																														
				Disabled	0	Read: Disabled																														
				Enabled	1	Read: Enabled																														
J	RW	S	PORTOSECURE			Write '1' to enable interrupt for event PORTOSECURE																														
				Set	1	Enable																														
				Disabled	0	Read: Disabled																														
				Enabled	1	Read: Enabled																														

8.9.5.14 INTENCLR1

Address offset: 0x318

Disable interrupt

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					J I																H G F E D C B A															
Reset 0x00000000					0 0																															
D	R/W	TZ	Field	Value ID	Value	Description																														
A-H	RW		IN[i] (i=0..7)			Write '1' to disable interrupt for event IN[i]																														
				Clear	1	Disable																														
				Disabled	0	Read: Disabled																														
				Enabled	1	Read: Enabled																														
I	RW	NS	PORTONONSECURE			Write '1' to disable interrupt for event PORTONONSECURE																														
				Clear	1	Disable																														
				Disabled	0	Read: Disabled																														
				Enabled	1	Read: Enabled																														
J	RW	S	PORTOSECURE			Write '1' to disable interrupt for event PORTOSECURE																														
				Clear	1	Disable																														
				Disabled	0	Read: Disabled																														
				Enabled	1	Read: Enabled																														

8.9.5.15 CONFIG[n] (n=0..7)

Address offset: 0x510 + (n × 0x4)

Configuration for OUT[n], SET[n], and CLR[n] tasks and IN[n] event

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																	
ID			E										D		D		C		C		C		C		B		B		B		B		A		A	
Reset 0x00000000			0 0																																	
ID	R/W	Field	Value ID	Value	Description																															
A	RW	MODE			Mode																															
			Disabled	0	Disabled. Pin specified by PSEL will not be acquired by the GPIOTE module.																															
			Event	1	Event mode																															
					The pin specified by PSEL will be configured as an input and the IN[n] event will be generated if operation specified in POLARITY occurs on the pin.																															
			Task	3	Task mode																															
					The GPIO specified by PSEL will be configured as an output and triggering the SET[n], CLR[n] or OUT[n] task will perform the operation specified by POLARITY on the pin. When enabled as a task the GPIOTE module will acquire the pin and the pin can no longer be written as a regular output pin from the GPIO module.																															
B	RW	PSEL		[0..31]	GPIO number associated with SET[n], CLR[n], and OUT[n] tasks and IN[n] event																															
C	RW	PORT		[0..15]	Port number																															
D	RW	POLARITY			When In task mode: Operation to be performed on output when OUT[n] task is triggered. When In event mode: Operation on input that shall trigger IN[n] event.																															
			None	0	Task mode: No effect on pin from OUT[n] task. Event mode: no IN[n] event generated on pin activity.																															
			LoToHi	1	Task mode: Set pin from OUT[n] task. Event mode: Generate IN[n] event when rising edge on pin.																															
			HiToLo	2	Task mode: Clear pin from OUT[n] task. Event mode: Generate IN[n] event when falling edge on pin.																															
			Toggle	3	Task mode: Toggle pin from OUT[n]. Event mode: Generate IN[n] when any change on pin.																															
E	RW	OUTINIT			When in task mode: Initial value of the output when the GPIOTE channel is configured. When in event mode: No effect.																															
			Low	0	Task mode: Initial value of pin before task triggering is low																															
			High	1	Task mode: Initial value of pin before task triggering is high																															

8.10 GRTC — Global real-time counter

The global real-time counter peripheral (GRTC) is an ultra-low power shared system timer. GRTC implements a high-resolution system timer that is available in all power modes, including System OFF.

The system timer has a 1 μ s resolution and is 52 bits wide. This provides a run time of 142 years after initial power-on until the counter wraps around. It uses the 16 MHz clock when the high-speed clock is active, but automatically switches to 32.768 kHz in the other power modes. It will continue to be updated in all power modes. Due to the combination of clock sources, it has a 1 μ s resolution and an accuracy equal to the 32.768 kHz clock.

The main features of GRTC are the following:

- System timer – SYSCOUNTER
 - 1 μ s resolution
 - Runs on a fast 16 MHz clock
 - Automatic synchronization of SYSCOUNTER with the internal low frequency timer for ultra-low power operation
 - Internal low frequency timer runs on LFCLK (32768 Hz clock)

- Internal low frequency timer can run while the device is in System OFF mode
- Multiple compare/capture channels on SYSCOUNTER
- PPI connection only for compare events and capture tasks
- Periodic interval generation for one compare event
- Wake up from System OFF mode
- Supports split security for GRTC features
- Pulse Width Modulation (PWM)
 - Single channel PWM
 - Operates in System OFF mode
- Clock output on pin
 - LFCLK
 - Configurable divided fast clock output

The system timer is a high resolution timer which can be accessed by all processors in the system. It allows the same system counter to be shared among all users. GRTC can operate in System OFF mode.

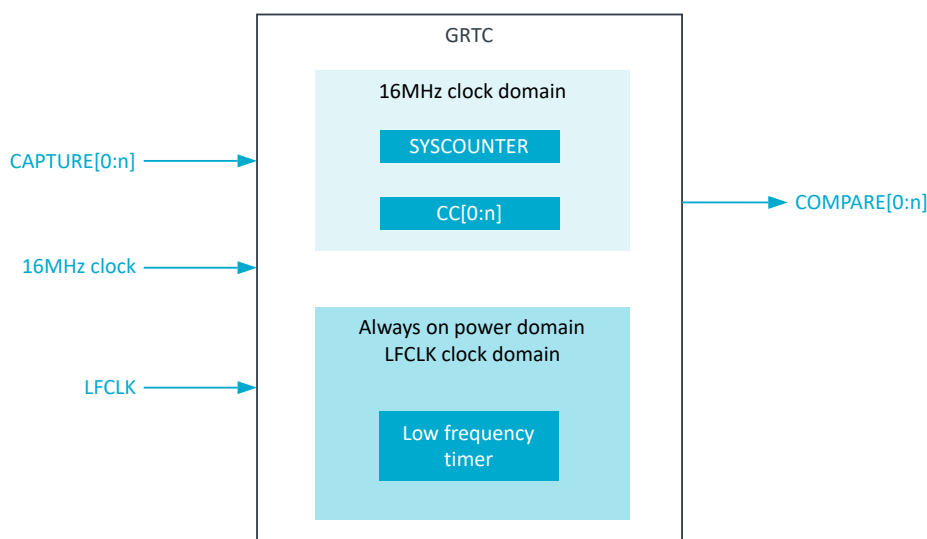


Figure 62: GRTC block diagram

The internal low-frequency timer can run while in System OFF mode.

8.10.1 GRTC clock sources

The low frequency timer in GRTC will run off the LFCLK.

When the low frequency timer is started, the GRTC peripheral will automatically request the LFCLK source if the LFCLK is not already running.

The GRTC low frequency timer clock source can be selected using register [CLKCFG.CLKSEL](#) between LFXO and LFLPRC. The clock source cannot be changed after GRTC is started.

All GRTC registers are reset during wakeup from System OFF mode, but the clock source selection at GRTC is retained internally.

The SYSCOUNTER runs off the fast clock (16MHz clock), but uses only LFCLK while SYSCOUNTER is in sleep state.

See [CLOCK](#) for more information about clock sources.

8.10.2 SYSCOUNTER

The internal counter at SYSCOUNTER increments every 1 μ s.

SYSCOUNTER is a 52-bit counter and is enabled using [MODE.SYSCOUNTEREN](#) register. The internal low frequency timer must be started for the proper operation of the SYSCOUNTER while SYSCOUNTER goes into sleep mode. It can be started by the [TASKS_START](#) task and can be stopped by the [TASKS_STOP](#) task when SYSCOUNTER is no more in use.

There are [m] registers [SYSCOUNTER\[m\]](#) providing access to SYSCOUNTER for each security attribute. The current value of SYSCOUNTER can be read using corresponding [SYSCOUNTER\[m\].SYSCOUNTERL](#) and [SYSCOUNTER\[m\].SYSCOUNTERH](#). But, the [SYSCOUNTER\[m\].SYSCOUNTERL](#) must be read before corresponding [SYSCOUNTER\[m\].SYSCOUNTERH](#). The [SYSCOUNTER\[m\].SYSCOUNTERH.OVERFLOW](#) indicates if the [SYSCOUNTER\[m\].SYSCOUNTERL](#) is overflown after reading it.

Sample code for reading the SYSCOUNTER value:

```
uint32_t syscounterl_value, syscounterh_value, syscounterh;
uint64_t syscounter;

do
{
    syscounterl_value = GRTC.SYSCOUNTER[m].SYSCOUNTERL;
    syscounterh = GRTC.SYSCOUNTER[m].SYSCOUNTERH;

    syscounterh_value = ((syscounterh & GRTC_SYSCOUNTER_SYSCOUNTERH_VALUE_Msk) >>
GRTC_SYSCOUNTER_SYSCOUNTERH_VALUE_Pos);

    if (((syscounterh & GRTC_SYSCOUNTER_SYSCOUNTERH_OVERFLOW_Msk) >>
GRTC_SYSCOUNTER_SYSCOUNTERH_OVERFLOW_Pos) ==
GRTC_SYSCOUNTER_SYSCOUNTERH_OVERFLOW_Overflow)
    {
        syscounterh = syscounterh_value - 1;
    }

} while (((syscounterh & GRTC_SYSCOUNTER_SYSCOUNTERH_BUSY_Msk) >>
GRTC_SYSCOUNTER_SYSCOUNTERH_BUSY_Pos) != GRTC_SYSCOUNTER_SYSCOUNTERH_BUSY_Ready);

syscounter = (syscounterh_value << 32) + syscounterl_value;
```

Compare and Capture (CC)

The [CC\[n\]](#) is group of registers interfacing the compare and capture channels of SYSCOUNTER, where *n* is the number of compare and capture channels specified in the GRTC instance configuration table below. Each [CC\[n\]](#) has an associated [TASKS_CAPTURE\[n\]](#) task and [EVENTS_COMPARE\[n\]](#) event. The [CC\[n\].CCEN.ACTIVE](#) must be enabled in order to use the corresponding SYSCOUNTER compare and capture channel. Each compare and capture channel operate in one-shot mode, so the channel is disabled automatically following the first compare event or capture task in the corresponding channel.

The [EVENTS_COMPARE\[n\]](#) event can be generated by writing the compare values to the corresponding [CC\[n\].CCL](#) and [CC\[n\].CCH](#) registers. Write to [CC\[n\].CCL](#) disables the corresponding compare channel and write to [CC\[n\].CCH](#) enables the the corresponding compare channel. So, the [CC\[n\].CCL](#) must be written first for a compare channel.

A compare channel is automatically disabled when the corresponding [EVENTS_COMPARE\[n\]](#) is generated or triggering [TASKS_CAPTURE\[n\]](#) task.

The `EVENTS_COMPARE[n]` event is generated immediately if the configured compare value at `CC[n]` is less than the current SYSCOUNTER value.

Every time the `TASKS_CAPTURE[n]` task is triggered, the current SYSCOUNTER is copied into the corresponding `CC[n].CCL` and `CC[n].CCH` registers. The `CC[n].CCL` and `CC[n].CCH` registers can be read in any order. The `TASKS_CAPTURE[n]` task will not generate `EVENTS_COMPARE[n]` event.

The `TASKS_CAPTURE[n]` tasks and `EVENTS_COMPARE[n]` events can be connected with the PPI. However, the `TASKS_CAPTURE[n]` is functional only when the SYSCOUNTER is in active state. The GRTC can be forced into active state by setting any `SYSCOUNTER[n].ACTIVE` register.

The compare value for `CC[n]` can also be updated by adding a fixed value provided at `CC[n].CCADD.VALUE`. Based on the `CC[n].CCADD.REFERENCE` configuration, either the current CC value or the current SYSCOUNTER value is added to the `CC[n].CCADD.VALUE` to configure the new compare value. If the `CC[n]` overflows after writing to `CC[n].CCADD.VALUE`, then `EVENTS_COMPARE[n]` is generated immediately. Writing to `CC[n].CCADD` enables the the corresponding compare channel.

Writes to `CC[n].CCADD` are ignored when the SYSCOUNTER is in sleep state.

Periodic interval

In addition to one-shot mode, the `CC[0]` can produce periodic `EVENTS_COMPARE[0]` event without any software interaction. The interval between these events can be programmed using `INTERVAL` register and non-zero interval enables this periodic interval feature. On every `EVENTS_COMPARE[0]` event, `CC[0]` becomes `CC[0] + INTERVAL`.

SYSCOUNTER sleep mode

SYSCOUNTER supports the following power modes:

- SYSCOUNTER is in active state
- SYSCOUNTER is in sleep state - This is the GRTC ultra-low power sleep mode

To save power, SYSCOUNTER automatically goes into sleep state when there is no activity.

Before SYSCOUNTER goes into sleep state, the GRTC configures the internal low frequency timer compare match based on the next expected SYSCOUNTER compare match using `CC[n]` configuration. An internal event on low frequency timer is generated when the compare match happens.

The internal counter at SYSCOUNTER is not ticking when SYSCOUNTER is in sleep state, the internal low frequency timer is configured as described above.

SYSCOUNTER returns to active state when any one of the following condition is met,

- Any of `SYSCOUNTER[n].ACTIVE` register is set to Active
- SYSCOUNTER counter value is read
- Any of the following registers are written:
 - `MODE`
 - `CC[n]`
 - `INTENn/INTENSETn/INTENCLRn/INTENPENDn`
 - Write to tasks `TASKS_START`, `TASKS_STOP` and `TASKS_CLEAR`
 - When internal low frequency timer compare match happens
- Any CPU is not sleeping, if `MODE.AUTOEN` is set

SYSCOUNTER goes back into sleep state when none of the above conditions met. However, the SYSCOUNTER active state can be extended by configuring the number of LFCLK cycles at `TIMEOUT` register.

On wake up to active state, SYSCOUNTER is updated based on the internal low frequency timer compare match. The status `SYSCOUNTER[m].SYSCOUNTERH.BUSY` indicates SYSCOUNTER is synchronized and valid after the SYSCOUNTER is woken up.

Before handling next scheduled `EVENTS_COMPARE[n]` event, GRTC must wake up from the low power state. The `WAKETIME` register configures the number of LFCLK cycles that GRTC will wake up before the compare event. This duration allows the device sufficient time to power up, initialize, and activate the necessary clocks to accurately generate and handle events from the GRTC SYSCOUNTER. A longer system wake-up time requires a larger `WAKETIME` value to ensure reliable event processing. When the device is in System OFF, the `WAKETIME` must cover the system wakeup time from System OFF mode and in addition the required time the system uses to configure GRTC and enable the event.

All GRTC registers must be restored at wakeup from system OFF before the next scheduled `COMPARE[n]` event is generated. The `TIMEOUT` register must be configured to a value higher than `WAKETIME` (`TIMEOUT > WAKETIME + guard_time`). This makes sure that GRTC is not entering sleep again if the next event is nearer than `TIMEOUT` LFCLK cycles. The minimum guard time is 1 LFCLK cycle.

Recommendation on reading SYSCOUNTER

The following steps are recommended while reading SYSCOUNTER:

1. Set the corresponding `SYSCOUNTER[m].ACTIVE` to Active
2. Wait until the corresponding status `SYSCOUNTER[m].SYSCOUNTERH.BUSY` is cleared
3. Read the corresponding `SYSCOUNTER[m].SYSCOUNTERL/H` values
4. Clear the `SYSCOUNTER[m].ACTIVE` set above

Entering System OFF mode

The following steps are recommended before entering System OFF mode:

1. Set the SYSCOUNTER in active state, either
 - Set `MODE.AUTOEN`
 - Set corresponding `SYSCOUNTER[m].ACTIVE` to Active
2. If GRTC is wakeup source, then set the corresponding `CC[n]` value to expected wakeup time
3. Set `WAKETIME` for the boot latency
4. Set the SYSCOUNTER in sleep state, by clearing the configuration set at step 1 above
5. Wait for either of `EVENTS_RTCOMPARESYNC` or any `EVENTS_COMPARE[n]`
 - If any `EVENTS_COMPARE[n]` triggered,
 - a. Allow CPUs to wakeup on interrupts
 - b. Do not enter System OFF mode
 - Else,
 - a. Enter System OFF by using the `SYSTEMOFF` register

8.10.3 Pulse Width Modulation (PWM)

The GRTC peripheral has a built-in PWM that can drive one output pin as an 8-bit non-inverted pulse-width modulated output.

The PWM is based on the internal low frequency timer of the GRTC and the PWM has a period of 256 LFCLK clock cycles, resulting frequency is 128Hz.

The PWM can be started by the `TASKS_PWMSTART` task and can be stopped by the `TASKS_PWMSTOP` task. The PWM starts/stops on next time when the lower 8 bits of internal low frequency timer becomes to zero. It takes up to 256 LFCLK clock cycles to take these tasks to go into effect.

The PWM compare value is configured using `PWMCONFIG` and the copied to the internal PWM compare register when the lower 8 bits of internal low frequency timer is 0.

The PWM output goes high when the the lower 8 bits of internal low frequency timer goes to zero and the PWM output goes low when the lower 8 bits of internal low frequency timer matches the PWM compare value. The [EVENTS_PWMPERIODEND](#) event is generated on the rising edge of the PWM output.

To optimize the GRTC power consumption, the [EVENTS_PWMPERIODEND](#) can be disabled using [EVTEN/EVTENSET/EVTENCLR](#) registers to prevent clock from being requested when those events are triggered.

The PWM is operating even while the device is in system OFF.

For the PWM output pin mapping, see [Pin assignments](#) on page 793. GRTC uses standard output drive strength for PWM output.

8.10.4 Clock output

The GRTC can be configured to output the clock on a pin.

The following clocks can be configured to be output on pins:

- LFCLK (32 KHz clock) output
- Divided 16M Hz clock (fast clock) output

The clock outputs can be enabled or disabled using [CLKOUT](#).

The 16 MHz clock is divided before it is output on a pin. The divisor can be configured in [CLKCFG](#), where clock output is (fast clock) / ([CLKCFG.CLKFASTDIV](#) * 2). The [CLKCFG.CLKFASTDIV](#) should be changed only when [CLKOUT.CLKOUTFAST](#) is disabled.

The LFCLK clock is output also when the device is in System OFF mode.

For the clock output pin mapping, see [Pin assignments](#) on page 793. GRTC uses standard output drive strength for clock output.

8.10.5 Split Security

The GRTC peripheral supports split security, where the split security features can have different security attributes than the GRTC peripheral.

The following GRTC features have split security attributes:

- Each compare/capture channel at [CC\[n\]](#) register group - The same security attribute applies to the corresponding channels for,
 - [TASKS_CAPTURE\[n\]](#)
 - [SUBSCRIBE_CAPTURE\[n\]](#)
 - [EVENTS_COMPARE\[n\]](#)
 - [PUBLISH_COMPARE\[n\]](#)
- Each [INTENm/INTENSETm/INTENCLRm/INTENPENDm](#) - The same security attribute applies to the following registers
 - [SYSCOUNTER\[m\]](#)
 - [PWMCONFIG](#)
 - [CLKOUT](#) and [CLKCFG](#)

[INTERVAL](#) has same security attribute as the [CC\[0\]](#) register group.

For more information on GRTC split ownership and security attributes, see [Split Security](#) on page 296.

Interrupts

GRTC provides multiple interrupts. Each interrupt is associated with its own set of [INTENm/INTENSETm/INTENCLRm/INTENPENDm](#) register. All events are routed to each of [INTENm/INTENSETm/INTENCLRm/](#)

INTENPENDm registers, however only those events matching the security attributes can be accessible using the INTENm/INTENSETm/INTENCLRm/INTENPENDm registers.

8.10.6 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
GRTC : S	GLOBAL	0x500E2000	US	S	NA	Yes	Global RTC GRTC
GRTC : NS		0x400E2000					

Configuration

Instance	Domain	Configuration
GRTC : S GRTC : NS	GLOBAL	Local CPUs connected to MODE.AUTO.EN.CpuActive : Application core Arm Cortex-M33.
		CLKSEL settings XO and LFLPRC must be used for lowest possible power consumption in sleep modes.
		Width of the RTCOUNTERH, RTCOMPAREH and RTCOMPARESYNCH registers : 0..14
		Number of compare/capture registers : 0..11
		Width of the TIMEOUT register : 0..15
		Number of GRTC interrupts : 0..3
		The PWM registers are available.
		The CLKOUT register is available.
		The CLKCFG.CLKSEL register is available.
		The CLKCFG.CLKSEL register supports LFLPRC.
		The CC[n].CCADD register has read/write access.
		The ready status and events are not available.
		SYSCOUNTER[n].SYSCOUNTERH.LOADED status is not available
		CC[n].CCEN.PASTCC status is not available
		4 interrupts with interrupt remapping
		12 capture compare channels implemented

Register overview

Register	Offset	TZ	Description
TASKS_CAPTURE[n]	0x000		Capture the counter value to CC[n] register
TASKS_START	0x060		Start the counter
TASKS_STOP	0x064		Stop the counter
TASKS_CLEAR	0x068		Clear the counter
TASKS_PWMSTART	0x06C		Start the PWM
TASKS_PWMSTOP	0x070		Stop the PWM
SUBSCRIBE_CAPTURE[n]	0x080		Subscribe configuration for task CAPTURE[n]
EVENTS_COMPARE[n]	0x100		Compare event on CC[n] match

Register	Offset	TZ	Description
EVENTS_RTCOMPARESYNC	0x164		The GRTC low frequency timer is synchronized with the SYSCOUNTER
EVENTS_PWMPERIODEND	0x16C		Event on end of each PWM period
PUBLISH_COMPARE[n]	0x180		Publish configuration for event COMPARE[n]
SHORTS	0x200		Shortcuts between local events and tasks
INTEN0	0x300		Enable or disable interrupt
INTENSET0	0x304		Enable interrupt
INTENCLR0	0x308		Disable interrupt
INTPEND0	0x30C		Pending interrupts
INTEN1	0x310		Enable or disable interrupt
INTENSET1	0x314		Enable interrupt
INTENCLR1	0x318		Disable interrupt
INTPEND1	0x31C		Pending interrupts
INTEN2	0x320		Enable or disable interrupt
INTENSET2	0x324		Enable interrupt
INTENCLR2	0x328		Disable interrupt
INTPEND2	0x32C		Pending interrupts
INTEN3	0x330		Enable or disable interrupt
INTENSET3	0x334		Enable interrupt
INTENCLR3	0x338		Disable interrupt
INTPEND3	0x33C		Pending interrupts
EVTEN	0x400		Enable or disable event routing
EVTENSET	0x404		Enable event routing
EVTENCLR	0x408		Disable event routing
MODE	0x510		Counter mode selection
CC[n].CCL	0x520		The lower 32-bits of Capture/Compare register CC[n]
CC[n].CCH	0x524		The higher 32-bits of Capture/Compare register CC[n]
CC[n].CCADD	0x528		Count to add to CC[n] when this register is written.
CC[n].CCEN	0x52C		Configure Capture/Compare register CC[n]
TIMEOUT	0x6A4		Timeout after all CPUs gone into sleep state to stop the SYSCOUNTER
INTERVAL	0x6A8		Count to add to CC[0] when the event EVENTS_COMPARE[0] triggers.
WAKETIME	0x6AC		GRTC wake up time.
PWMCONFIG	0x710		PWM configuration.
CLKOUT	0x714		Configuration of clock output
CLKCFG	0x718		Clock Configuration
SYSCOUNTER[n].SYSCOUNTERL	0x720		The lower 32-bits of the SYSCOUNTER for index [n]
SYSCOUNTER[n].SYSCOUNTERH	0x724		The higher 20-bits of the SYSCOUNTER for index [n]
SYSCOUNTER[n].ACTIVE	0x728		Request to keep the SYSCOUNTER in the active state and prevent going to sleep for index [n]

8.10.6.1 TASKS_CAPTURE[n] (n=0..11)

Address offset: $0x000 + (n \times 0x4)$

Capture the counter value to CC[n] register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_CAPTURE						Capture the counter value to CC[n] register																											
			Trigger	1				Trigger task																											

8.10.6.2 TASKS_START

Address offset: 0x060

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																								A			
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																						
A	W	TASKS_PWMSTOP			Stop the PWM																																						
			Trigger	1	Trigger task																																						

8.10.6.7 SUBSCRIBE_CAPTURE[n] (n=0..11)

Address offset: 0x080 + (n × 0x4)

Subscribe configuration for task CAPTURE[n]

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that task CAPTURE[n] will subscribe to																																
B	RW	EN																																				
			Disabled	0		Disable subscription																																
			Enabled	1		Enable subscription																																

8.10.6.8 EVENTS_COMPARE[n] (n=0..11)

Address offset: 0x100 + (n × 0x4)

Compare event on CC[n] match

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	EVENTS_COMPARE			Compare event on CC[n] match																																	
			NotGenerated	0	Event not generated																																	
			Generated	1	Event generated																																	

8.10.6.9 EVENTS_RTCOMPARESYNC

Address offset: 0x164

The GRTC low frequency timer is synchronized with the SYSCOUNTER

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_RTCOMPARESYNC			The GRTC low frequency timer is synchronized with the SYSCOUNTER																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.10.6.10 EVENTS_PWMPERIODEND

Address offset: 0x16C

Event on end of each PWM period

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				A																																	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	EVENTS_PWMPERIODEND				Event on end of each PWM period																															
			NotGenerated	0	Event not generated																																
			Generated	1	Event generated																																

8.10.6.11 PUBLISH_COMPARE[n] (n=0..11)

Address offset: $0x180 + (n \times 0x4)$

Publish configuration for event COMPARE[n]

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																											
ID				B																								A				A	A	A	A	A	A	A																								
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																						
A	RW	CHIDX		[0..255]				DPPI channel that event COMPARE[n] will publish to																																																						
B	RW	EN																																																												
			Disabled	0				Disable publishing																																																						
			Enabled	1				Enable publishing																																																						

8.10.6.12 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	RTCOMPARE_CLEAR						Shortcut between event RTCOMPARE and task CLEAR																											
			Disabled	0				Disable shortcut																											
			Enabled	1				Enable shortcut																											

8.10.6.13 INTENO

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M																L K J I H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-L	RW	COMPARE[i] (i=0..11)			Enable or disable interrupt for event COMPARE[i]																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
M	RW	RTCOMPARESYNC			Enable or disable interrupt for event RTCOMPARESYNC																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
N	RW	PWMPERIODEND			Enable or disable interrupt for event PWMPERIODEND																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.10.6.14 INTENSET0

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N										M										L K J I H G F E D C B A											
Reset 0x00000000				0 0																															

8.10.6.15 INTENCLR0

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N										M										L K J I H G F E D C B A											
Reset 0x00000000				0 0																															

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																															
ID				N																M																L K J I H G F E D C B A															
Reset 0x00000000				0 0																																															
ID	R/W	Field	Value ID	Value																Description																															
N	RW	PWMPERIODEND																		Write '1' to disable interrupt for event PWMPERIODEND																															
			Clear	1																Disable																															
			Disabled	0																Read: Disabled																															
			Enabled	1																Read: Enabled																															

8.10.6.16 INTPENDO

Address offset: 0x30C

Pending interrupts

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0									
ID				N										M										L										K	J	I	H	G	F	E	D	C	B	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								
ID	R/W	Field	Value ID	Value		Description																																						
A-L	R	COMPARE[i] (i=0..11)				Read pending status of interrupt for event COMPARE[i]																																						
			NotPending	0		Read: Not pending																																						
			Pending	1		Read: Pending																																						
M	R	RTCOMPARESYNC				Read pending status of interrupt for event RTCOMPARESYNC																																						
			NotPending	0		Read: Not pending																																						
			Pending	1		Read: Pending																																						
N	R	PWMPERIODEND				Read pending status of interrupt for event PWMPERIODEND																																						
			NotPending	0		Read: Not pending																																						
			Pending	1		Read: Pending																																						

8.10.6.17 INTEN1

Address offset: 0x310

Enable or disable interrupt

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0									
ID				N										M										L										K	J	I	H	G	F	E	D	C	B	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								
ID	R/W	Field	Value ID	Value	Description																																							
A-L	RW	COMPARE[i] (i=0..11)			Enable or disable interrupt for event COMPARE[i]																																							
			Disabled	0	Disable																																							
			Enabled	1	Enable																																							
M	RW	RTCOMPARESYNC			Enable or disable interrupt for event RTCOMPARESYNC																																							
			Disabled	0	Disable																																							
			Enabled	1	Enable																																							
N	RW	PWMPERIODEND			Enable or disable interrupt for event PWMPERIODEND																																							
			Disabled	0	Disable																																							
			Enabled	1	Enable																																							

8.10.6.18 INTENSET1

Address offset: 0x314

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																															
ID				N																M																L K J I H G F E D C B A															
Reset 0x00000000				0 0																																															
ID	R/W	Field	Value ID	Value	Description																																														
A-L	RW	COMPARE[i] (i=0..11)			Write '1' to enable interrupt for event COMPARE[i]																																														
			Set	1	Enable																																														
			Disabled	0	Read: Disabled																																														
			Enabled	1	Read: Enabled																																														
M	RW	RTCOMPARESYNC			Write '1' to enable interrupt for event RTCOMPARESYNC																																														
			Set	1	Enable																																														
			Disabled	0	Read: Disabled																																														
			Enabled	1	Read: Enabled																																														
N	RW	PWMPERIODEND			Write '1' to enable interrupt for event PWMPERIODEND																																														
			Set	1	Enable																																														
			Disabled	0	Read: Disabled																																														
			Enabled	1	Read: Enabled																																														

8.10.6.19 INTENCLR1

Address offset: 0x318

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																															
ID				N																M																L K J I H G F E D C B A															
Reset 0x00000000				0 0																																															
ID	R/W	Field	Value ID	Value	Description																																														
A-L	RW	COMPARE[i] (i=0..11)			Write '1' to disable interrupt for event COMPARE[i]																																														
			Clear	1	Disable																																														
			Disabled	0	Read: Disabled																																														
			Enabled	1	Read: Enabled																																														
M	RW	RTCOMPARESYNC			Write '1' to disable interrupt for event RTCOMPARESYNC																																														
			Clear	1	Disable																																														
			Disabled	0	Read: Disabled																																														
			Enabled	1	Read: Enabled																																														
N	RW	PWMPERIODEND			Write '1' to disable interrupt for event PWMPERIODEND																																														
			Clear	1	Disable																																														
			Disabled	0	Read: Disabled																																														
			Enabled	1	Read: Enabled																																														

8.10.6.20 INTPEND1

Address offset: 0x31C

Pending interrupts

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M																L K J I H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-L	R	COMPARE[i] (i=0..11)			Read pending status of interrupt for event COMPARE[i]																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
M	R	RTCOMPARESYNC			Read pending status of interrupt for event RTCOMPARESYNC																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
N	R	PWMPERIODEND			Read pending status of interrupt for event PWMPERIODEND																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														

8.10.6.21 INTEN2

Address offset: 0x320

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M																L K J I H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-L	RW	COMPARE[i] (i=0..11)			Enable or disable interrupt for event COMPARE[i]																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
M	RW	RTCOMPARESYNC			Enable or disable interrupt for event RTCOMPARESYNC																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
N	RW	PWMPERIODEND			Enable or disable interrupt for event PWMPERIODEND																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.10.6.22 INTENSET2

Address offset: 0x324

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M																L K J I H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-L	RW	COMPARE[i] (i=0..11)			Write '1' to enable interrupt for event COMPARE[i]																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
M	RW	RTCOMPARESYNC			Write '1' to enable interrupt for event RTCOMPARESYNC																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
N	RW	PWMPERIODEND			Write '1' to enable interrupt for event PWMPERIODEND																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														

8.10.6.23 INTENCLR2

Disable interrupt

8.10.6.24 INTPEND2

Pending interrupts

8.10.6.25 INTEN3

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M																L K J I H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-L	RW	COMPARE[i] (i=0..11)			Enable or disable interrupt for event COMPARE[i]																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
M	RW	RTCOMPARESYNC			Enable or disable interrupt for event RTCOMPARESYNC																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
N	RW	PWMPERIODEND			Enable or disable interrupt for event PWMPERIODEND																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.10.6.26 INTENSET3

Address offset: 0x334

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M																L K J I H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-L	RW	COMPARE[i] (i=0..11)			Write '1' to enable interrupt for event COMPARE[i]																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
M	RW	RTCOMPARESYNC			Write '1' to enable interrupt for event RTCOMPARESYNC																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
N	RW	PWMPERIODEND			Write '1' to enable interrupt for event PWMPERIODEND																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.10.6.27 INTENCLR3

Address offset: 0x338

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M																L K J I H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-L	RW	COMPARE[i] (i=0..11)			Write '1' to disable interrupt for event COMPARE[i]																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
M	RW	RTCOMPARESYNC			Write '1' to disable interrupt for event RTCOMPARESYNC																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																															
ID				N																M																L K J I H G F E D C B A															
Reset 0x00000000				0 0																																															
ID	R/W	Field	Value ID	Value				Description																																											
N	RW	PWMPERIODEND						Write '1' to disable interrupt for event PWMPERIODEND																																											
			Clear	1				Disable																																											
			Disabled	0				Read: Disabled																																											
			Enabled	1				Read: Enabled																																											

8.10.6.28 INTPEND3

Address offset: 0x33C

Pending interrupts

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																															
ID				N																M																L K J I H G F E D C B A															
Reset 0x00000000				0 0																																															

8.10.6.29 EVTEN

Address offset: 0x400

Enable or disable event routing

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PWMPERIODEND						Enable or disable event routing for event PWMPERIODEND																											
			Disabled	0				Disable																											
			Enabled	1				Enable																											

8.10.6.30 EVTENSET

Address offset: 0x404

Enable event routing

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	PWMPERIODEND			Write '1' to enable event routing for event PWMPERIODEND																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			Set	1	Enable																														

8.10.6.31 EVTENCLR

Address offset: 0x408

Disable event routing

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	PWMPERIODEND			Write '1' to disable event routing for event PWMPERIODEND																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			Clear	1	Disable																														

8.10.6.32 MODE

Address offset: 0x510

Counter mode selection

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	AUTOEN			Automatic enable to keep the SYSCOUNTER active.																														
			Default	0	Default configuration to keep the SYSCOUNTER active.																														
			CpuActive	1	In addition to the above mode, any local CPU that is not sleeping keep the SYSCOUNTER active.																														
B	RW	SYSCOUNTEREN			Enable the SYSCOUNTER																														
			Disabled	0	SYSCOUNTER disabled																														
			Enabled	1	SYSCOUNTER enabled																														

8.10.6.33 CC[n].CCL (n=0..11)

Address offset: 0x520 + (n × 0x10)

The lower 32-bits of Capture/Compare register CC[n]

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A		
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	CCL			Capture/Compare low value in 1 μs																																	

8.10.6.34 CC[n].CCH (n=0..11)

Address offset: 0x524 + (n × 0x10)

The higher 32-bits of Capture/Compare register CC[n]

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID	A A																															
Reset 0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																											
A	RW	CCH			Capture/Compare high value in 1 µs																											

8.10.6.35 CC[n].CCADD (n=0..11)

Address offset: 0x528 + (n × 0x10)

Count to add to CC[n] when this register is written.

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID	B	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																											
A	RW	VALUE			Count to add to CC[n]																											
B	RW	REFERENCE			Configure the Capture/Compare register																											
			SYSCOUNTER	0	Adds SYSCOUNTER value.																											
			CC	1	Adds CC value.																											
					CC[n] becomes CCADD[n].VALUE + SYSCOUNTER when this register is written.																											
					CC[n] becomes CCADD[n].VALUE + CC[n] when this register is written.																											

8.10.6.36 CC[n].CCEN (n=0..11)

Address offset: 0x52C + (n × 0x10)

Configure Capture/Compare register CC[n]

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID	A																															
Reset 0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																											
A	RW	ACTIVE			Configure the Capture/Compare register																											
			Disable	0	Capture/Compare register CC[n] Disabled.																											
			Enable	1	Capture/Compare register CC[n] enabled.																											

8.10.6.37 TIMEOUT

Address offset: 0x6A4

Timeout after all CPUs gone into sleep state to stop the SYSCOUNTER

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value																Description																	
A	RW	VALUE																		Number of 32Ki cycles																	

8.10.6.38 INTERVAL

Address offset: 0x6A8

Count to add to CC[0] when the event EVENTS_COMPARE[0] triggers.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															

8.10.6.39 WAKETIME

Address offset: 0x6AC

GRTC wake up time.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																												A	A	A	A	A	A	A	A	A	A
Reset 0x00000001				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1		
ID	R/W	Field	Value ID	Value																Description																	
A	RW	VALUE																		Number of LFCLK clock cycles to wake up before the next scheduled EVENTS_COMPARE event																	

8.10.6.40 PWMCONFIG

Address offset: 0x710

PWM configuration.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

8.10.6.41 CLKOUT

Address offset: 0x714

Configuration of clock output

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																				B	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value	Description																																
A	RW	CLKOUT32K			Enable 32Ki clock output on pin																																
			Disabled	0	Disabled																																
			Enabled	1	Enabled																																
B	RW	CLKOUTFAST			Enable fast clock output on pin																																
			Disabled	0	Disabled																																
			Enabled	1	Enabled																																

8.10.6.42 CLKCFG

Address offset: 0x718

Clock Configuration

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0						
ID																					B	B													A	A	A	A	A	A	A
Reset 0x00010001				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1					
ID	R/W	Field	Value ID	Value	Description																																				
A	RW	CLKFASTDIV		1..255	Fast clock divisor value of clock output																																				
					Fast clock divisor value 0 behaves same as 1.																																				
B	W	CLKSEL			GRTC LFCLK clock source selection																																				
			LFXO	0	GRTC LFCLK clock source is LFXO																																				
			SystemLFCLK	1	GRTC LFCLK clock source is system LFCLK																																				
			LFLPRC	2	GRTC LFCLK clock source is LFLPRC																																				

8.10.6.43 SYSCOUNTER[n].SYSCOUNTERL (n=0..3)

Address offset: 0x720 + (n × 0x10)

The lower 32-bits of the SYSCOUNTER for index [n]

The SYSCOUNTERL must be read before SYSCOUNTERH.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	VALUE						The lower 32-bits of the SYSCOUNTER value.																											

8.10.6.44 SYSCOUNTER[n].SYSCOUNTERH (n=0..3)

Address offset: 0x724 + (n × 0x10)

The higher 20-bits of the SYSCOUNTER for index [n]

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				C	B															A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	
Reset 0x40000000				0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	R	VALUE			The higher 20-bits of the SYSCOUNTER value.																																	
B	R	BUSY			SYSCOUNTER busy status																																	
			Ready	0	SYSCOUNTER is ready for read																																	
			Busy	1	SYSCOUNTER is busy, so not ready for read (value returned in the VALUE field of this register is not valid)																																	
C	R	OVERFLOW			The SYSCOUNTERL overflow indication after reading it.																																	
			NoOverflow	0	SYSCOUNTERL is not overflown																																	
			Overflow	1	SYSCOUNTERL overflown																																	

8.10.6.45 SYSCOUNTER[n].ACTIVE (n=0..3)

Address offset: 0x728 + (n × 0x10)

Request to keep the SYSCOUNTER in the active state and prevent going to sleep for index [n]

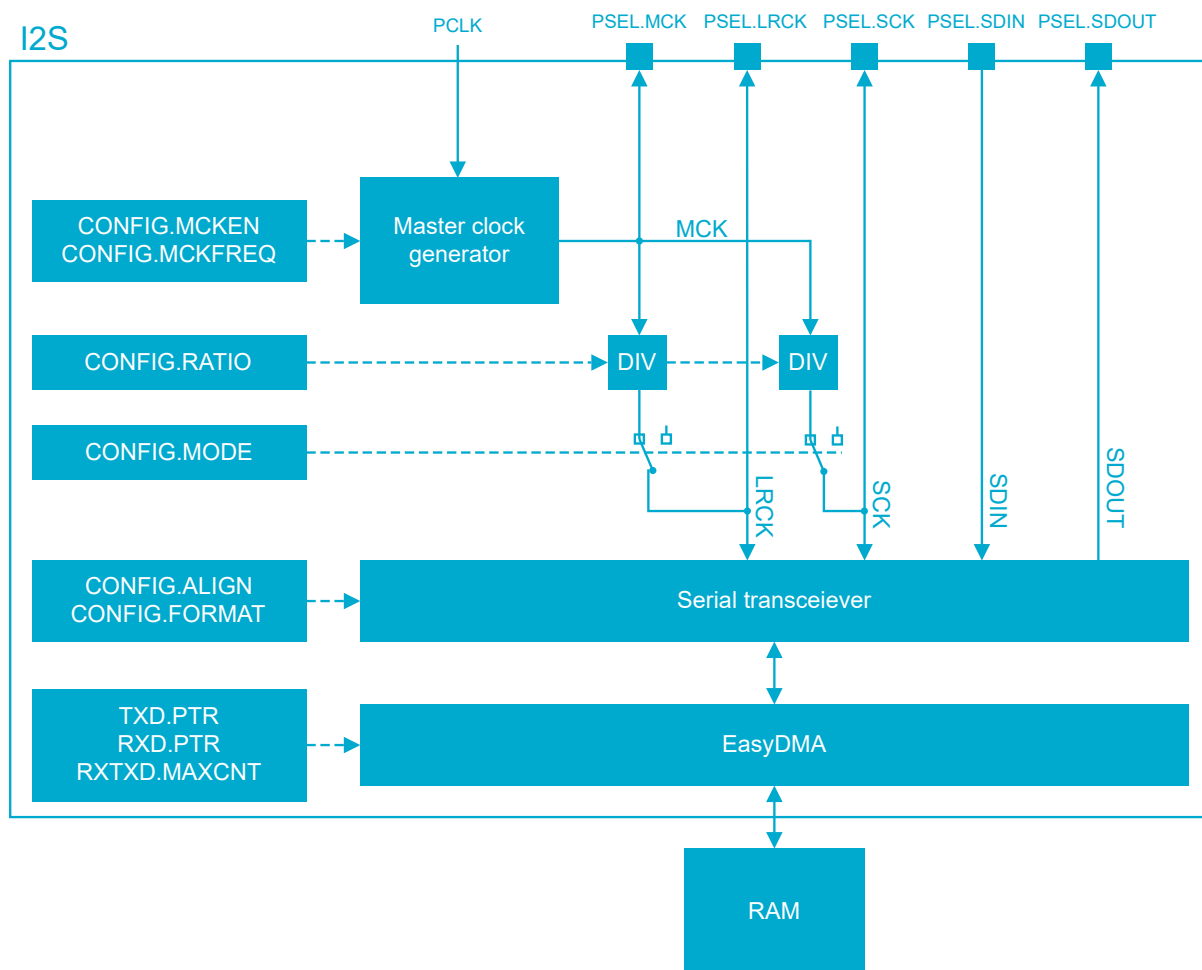
Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ACTIVE			Keep SYSCOUNTER in active state																														
			NotActive	0	Allow SYSCOUNTER to go to sleep																														
			Active	1	Keep SYSCOUNTER active																														

8.11 I²S — Inter-IC sound interface

The I²S (Inter-IC Sound) module, supports the original two-channel I²S format, and left- or right-aligned formats. It implements EasyDMA for sample transfer directly to and from RAM without CPU intervention.

The I²S peripheral has the following main features:

- Master and Slave mode
- Simultaneous bidirectional (TX and RX) audio streaming
- Original I²S and left- or right-aligned format
- 32, 24, 16 and 8-bit sample widths
- Separate sample and word widths
- Low-jitter master clock generator
- Various sample rates

Figure 63: I²S master

8.11.1 Mode

The I²S protocol specification defines two modes of operation, Master and Slave.

The I²S mode decides which of the two sides (master or slave) shall provide the clock signals LRCK and SCK, and these signals are always supplied by the master to the slave.

8.11.2 Transmitting and receiving

The I²S module supports both transmission (TX) and reception (RX) of serial data. In both cases the serial data is shifted synchronously to the clock signals SCK and LRCK.

TX data is written to the SDOUT pin on the falling edge of SCK, and RX data is read from the SDIN pin on the rising edge of SCK. The most significant bit (MSB) is always transmitted first.

Note: When starting a transmission in master mode, the first frame is filled with zeros.

TX and RX are available in both Master and Slave modes and can be enabled/disabled independently in the [CONFIG.TXEN](#) on page 333 and [CONFIG.RXEN](#) on page 333.

Transmission and/or reception is started by triggering the [START](#) task. With transmission enabled in [CONFIG.TXEN](#), the [TXPTRUPD](#) event will be generated for every number of transmitted data words given by [RXTXD.MAXCNT](#) on page 337. Each data word contains one or more samples. The [TXPTRUPD](#) event is generated just before [MAXCNT](#) number of data words have been transmitted. Similarly, with reception enabled in [CONFIG.RXEN](#), the [RXPTRUPD](#) event will be generated for every received data word given by

RXTXD.MAXCNT on page 337. The **RXPTRUPD** event is generated just after **MAXCNT** number of data words have been received.

The **FRAMESTART** event is generated synchronously to the active LRCK edge at the beginning of a frame after transmitting **RXTXD.MAXCNT** data words. The initial **FRAMESTART** event is generated at the first active edge of LRCK after the **START** task has been triggered. The **FRAMESTART** event is only defined for transmitting full left and right sample pairs. If **MAXCNT** is configured so that the frame ends between the left and right sample pairs, the **FRAMESTART** event is not generated. This occurs for the following combinations of **WIDTH** and **MAXCNT**:

WIDTH	MAXCNT restriction
24Bit	Only even numbers (2,4,6, etc)
32	Only even numbers (2, 4, 6, etc)
24BitIn32	Only even numbers (2, 4, 6, etc)

Table 37: Restrictions on combinations of **WIDTH** and **MAXCNT** for correct **FRAMESTART**

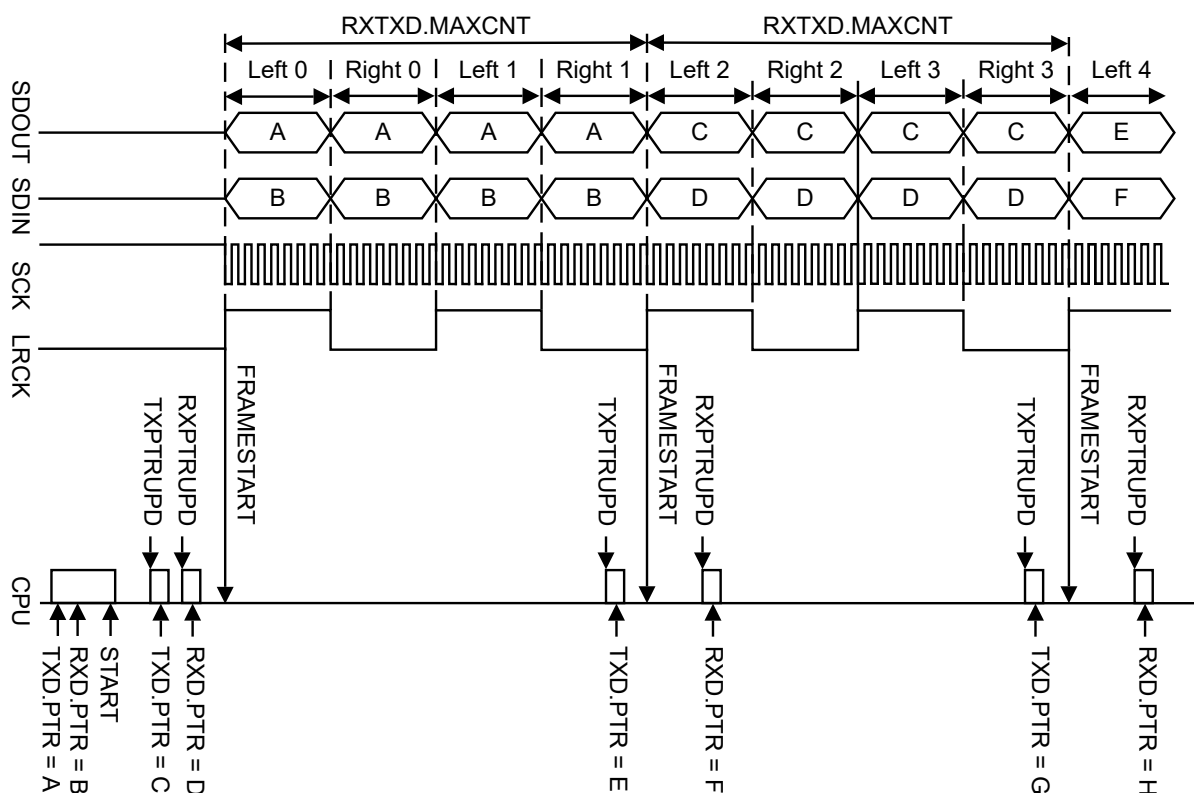


Figure 64: Transmitting and receiving. **CONFIG.FORMAT** = Aligned, **CONFIG.SWIDTH** = 8Bit, **CONFIG.CHANNELS** = Stereo, **RXTXD.MAXCNT** = 1

8.11.3 Left right clock (LRCK)

The left right clock (LRCK), often referred to as word clock, sample clock, or word select in I²S context, is the clock defining the frames in serial bitstreams sent and received on SDOUT and SDIN, respectively.

In I2S format, each frame contains one left and/or right sample pair. The left sample is transferred during the low half period of LRCK, followed by the right sample being transferred during the high half period of LRCK.

In Aligned format, each frame contains one left and/or right sample pair. The left sample is transferred during the high half period of LRCK, followed by the right sample being transferred during the low half period of LRCK.

For mono, the frame will contain only zeros for the unused half period of LRCK.

Consequently, the LRCK frequency is equivalent to the audio sample rate.

When operating in Master mode, the LRCK is generated from the MCK, and the frequency of LRCK is then given as:

$$\text{LRCK} = \text{MCK} / \text{CONFIG.RATIO}$$

LRCK always toggles around the falling edge of the serial clock SCK.

8.11.4 Serial clock (SCK)

The serial clock (SCK), often referred to as the serial bit clock, pulses once for each data bit being transferred on the serial data lines SDIN and SDOUT.

When operating in Master mode, the SCK is generated from the MCK, and the frequency of SCK is then given as:

$$\text{SCK} = 2 * \text{LRCK} * \text{CONFIG.SWIDTH}$$

The falling edge of the SCK falls on the toggling edge of LRCK.

When operating in Slave mode, SCK is provided by the external I²S master.

8.11.5 Master clock (MCK)

The master clock (MCK) is the clock from which LRCK and SCK are derived when operating in Master mode.

The master clock generator always needs to be enabled when in Master mode, but the generator can also be enabled when in Slave mode. Enabling the generator when in Slave mode can be useful in the case where the external master is not able to generate its own master clock.

MCK is generated from the [CONFIG.MCKFREQ](#) registers.

The following equation can be used to calculate the value of [CONFIG.MCKFREQ](#) for given MCK and clock source frequency:

$$\text{MCKFREQ} = 4096 \cdot \left\lfloor \frac{f_{\text{MCK}} \cdot 1048576}{f_{\text{source}} + \frac{f_{\text{MCK}}}{2}} \right\rfloor$$

Figure 65: MCK clock frequency equation

The parameter f_{MCK} is the requested MCK clock frequency in Hz, and f_{source} is the frequency of the selected clock source in Hz. Because of rounding errors, an accurate MCK clock may not be achievable. The equation does not take into account the maximum register value of [CONFIG.MCKFREQ](#) on page 334.

The actual MCK frequency can be calculated using the equation below.

$$f_{\text{actual}} = \frac{f_{\text{source}}}{\left\lfloor \frac{1048576 \cdot 4096}{\text{MCKFREQ}} \right\rfloor}$$

Figure 66: Actual MCK clock frequency

The clock error can be calculated using the equation below. The error e is the percentage difference from the requested f_{MCK} frequency.

$$e = 100 \cdot \frac{f_{actual} - f_{MCK}}{f_{MCK}} = 100 \cdot \frac{\left[\frac{f_{source}}{MCKFREQ} \right] - f_{MCK}}{f_{MCK}}$$

Figure 67: MCK frequency error equation

The master clock generator does not add any jitter to the clock source chosen.

The master clock generator is enabled/disabled using [CONFIG.MCKEN](#) on page 333, and the generator is started or stopped by the [START](#) or [STOP](#) tasks respectively.

The MCK frequency can be adjusted on-the-fly by using [MCKFREQ](#).

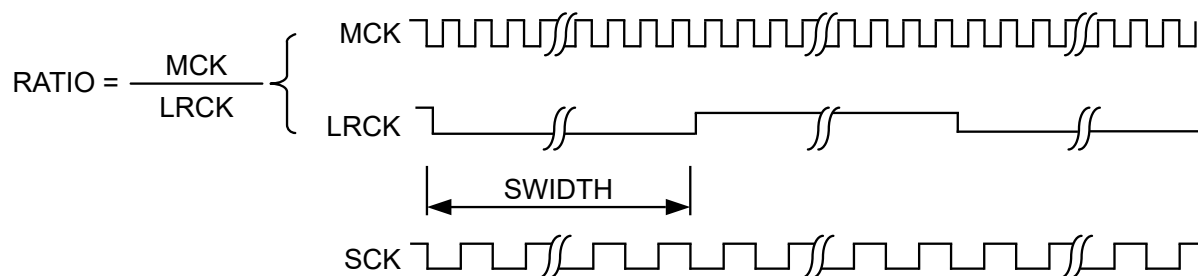
In Master mode, the LRCK and the SCK frequencies are closely related as both are derived from MCK and set indirectly through [CONFIG.RATIO](#) on page 335 and [CONFIG.SWIDTH](#) on page 335.

When configuring these registers, the user is responsible for fulfilling the following requirements:

1. The SCK frequency can never exceed the MCK frequency.
2. The MCK/LRCK ratio shall be a multiple of $2 * \text{CONFIG.SWIDTH}$.

The MCK signal can be routed to an output pin (specified in [PSEL.MCK](#)) to supply external I²S devices that require the MCK to be supplied from the outside.

When operating in Slave mode, the I²S module does not use the MCK and the MCK generator does not need to be enabled.

Figure 68: Relation between *RATIO*, MCK and LRCK

8.11.5.1 Configuration examples

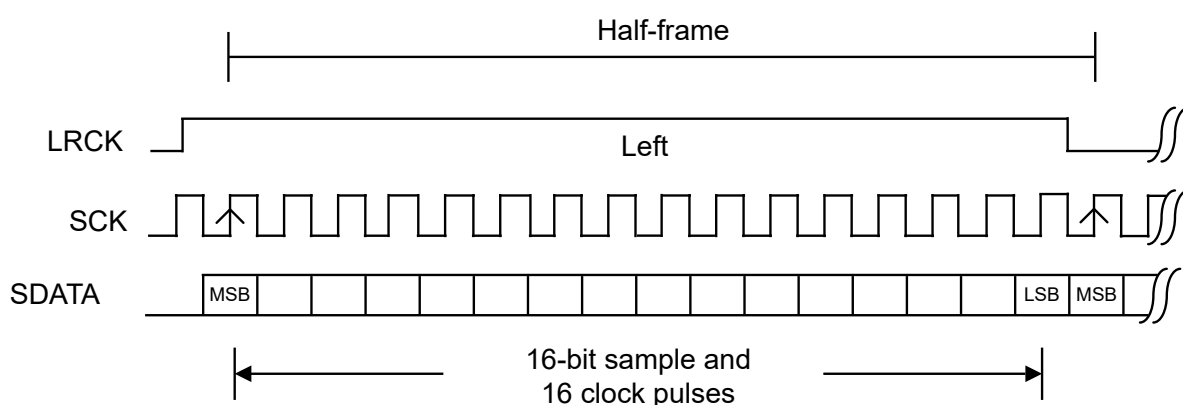
The following are example configurations for popular sample rates.

Source frequency [Hz]	Requested LRCK [Hz]	RATIO	Requested MCK [Hz]	MCKFREQ	MCK [Hz]	LRCK [Hz]	LRCK error [%]
32000000	16000	32	512000	68173824	507936	15873	-0.8
32000000	16000	64	1024000	135274496	1032258	16129	0.8
32000000	16000	256	4096000	516685824	4000000	15625	-2.3
32000000	32000	32	1024000	135274496	1032258	32258	0.8
32000000	32000	64	2048000	266350592	2000000	31250	-2.3
32000000	32000	256	8192000	974741504	8000000	31250	-2.3
32000000	44100	32	1411200	185319424	1391304	43478	-1.4
32000000	44100	64	2822400	362815488	2909090	45455	3.1
32000000	48000	32	1536000	201326592	1523809	47619	-0.8
32000000	48000	64	3072000	393428992	3200000	50000	4.2
32000000	96000	32	3072000	393428992	3200000	100000	4.2
32000000	96000	64	6144000	752402432	6400000	100000	4.2

Table 38: Configuration examples for 32 MHz PCLK

8.11.6 Width, alignment and format

The register `CONFIG.SWIDTH` on page 335 defines the sample width of the data read and written to memory, as well as the number of SCK clock cycles per half-frame. Figure [Aligned format, with `CONFIG.SWIDTH` configured to 16 bit samples in a 16 bit half-frame](#) on page 318 illustrates a configuration with identical sample and half-frame widths. The number of SCK pulses matches the number of sample bits. [Aligned format, with `CONFIG.SWIDTH` configured to 16-bit samples in a 24-bit half-frame](#) on page 319 illustrates a configuration with greater half-frame width than sample width. The number of SCK pulses are greater than the number of sample bits, with the sample being left-aligned in the half-frame.

Figure 69: Aligned format, with `CONFIG.SWIDTH` configured to 16 bit samples in a 16 bit half-frame

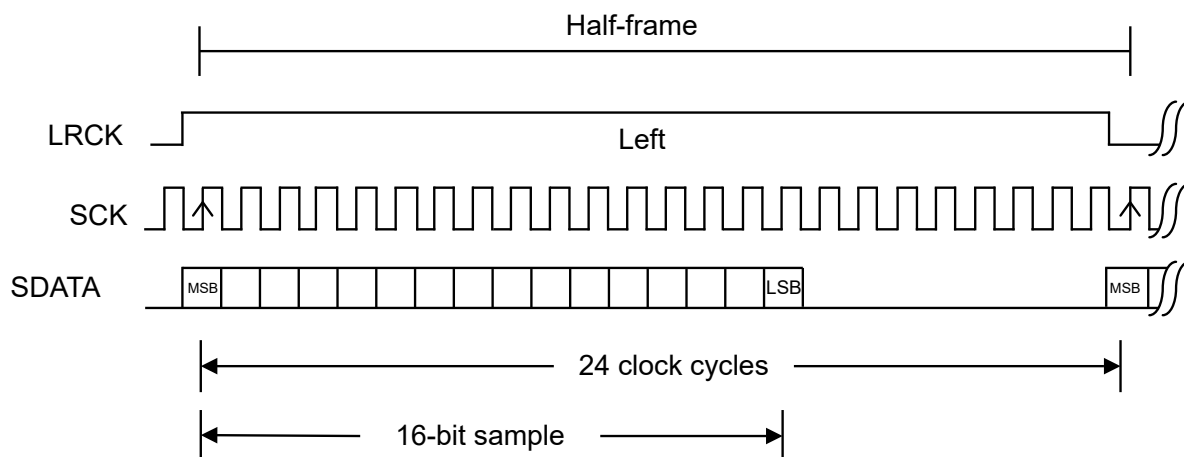


Figure 70: Aligned format, with `CONFIG.SWIDTH` configured to 16-bit samples in a 24-bit half-frame

The register `CONFIG.FORMAT` on page 336 is used to choose whether a word shall be aligned on the LRCK edge, or be delayed one bit period after this edge:

- When using Aligned format, the first bit in a half-frame gets sampled on the first rising edge of SCK following a LRCK edge, as illustrated in [Aligned format. Identical sample width and half-frame width. Left sample on high level of LRCK](#) on page 319. The left sample is transferred during the high half period of LRCK.
- When using I^2S format, the first bit in a half-frame (containing one left or right sample) gets sampled on the second rising edge of the SCK after a LRCK edge, as illustrated in [I²S format. Identical sample width and half-frame width. Left sample on low level of LRCK](#) on page 319. The left sample is transferred during the low half period of LRCK.

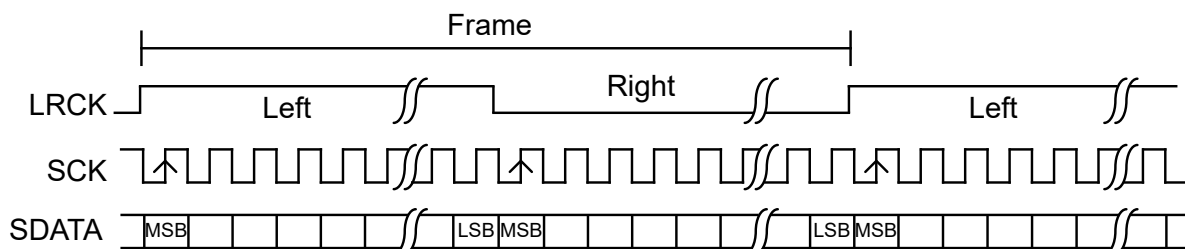


Figure 71: Aligned format. Identical sample width and half-frame width. Left sample on high level of LRCK

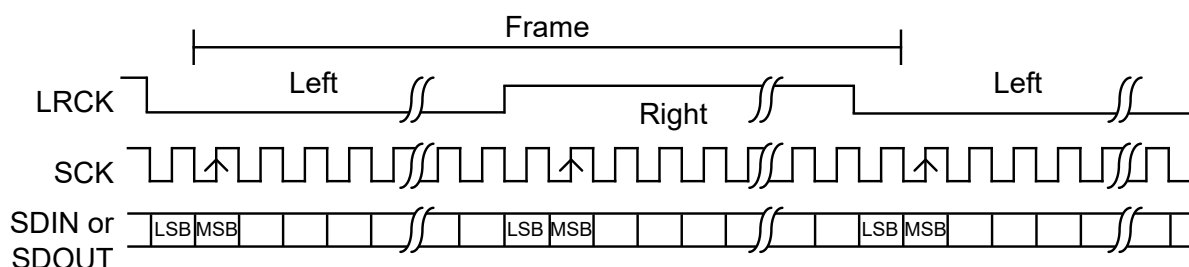


Figure 72: I^2S format. Identical sample width and half-frame width. Left sample on low level of LRCK

If the half-frame width differs from the sample width, the sample value can be either right or left-aligned inside a half-frame, as specified in `CONFIG.ALIGN` on page 335

- When using left-alignment, each half-frame starts with the MSB of the sample value, as illustrated by [CONFIG.ALIGN set to left justified](#) on page 320.
- When using right-alignment, each half-frame ends with the LSB of the sample value. This is illustrated in [CONFIG.ALIGN set to right justified](#) on page 320.

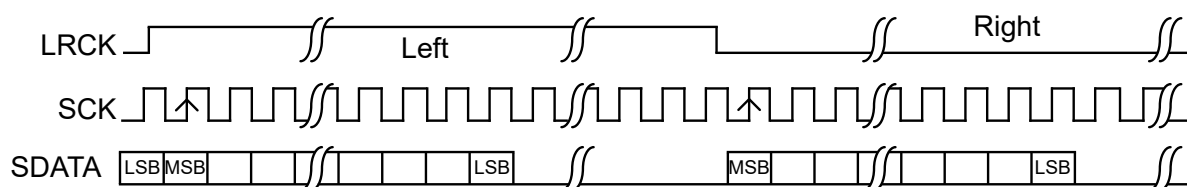


Figure 73: CONFIG.ALIGN set to left justified

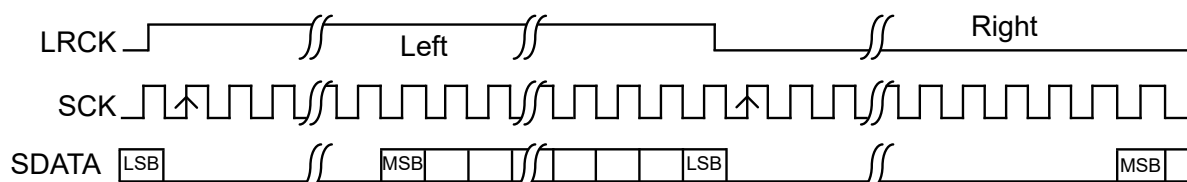


Figure 74: CONFIG.ALIGN set to right justified

Slave mode considerations

In Slave mode, the sample width does not need to equal the half-frame width, or even frame size. This means that there can be extra or fewer SCK pulses per half-frame than what the sample and half-frame widths specified in [CONFIG.SWIDTH](#) on page 335 require.

In cases where **left-alignment** is used, and the number of SCK pulses per half-frame is **higher** than the configured width, the following will apply:

- For data received on SDIN, all bits after the least significant bit (LSB) of the word value will be discarded.
- For data sent on SDOUT, all bits after the LSB of the word value will be 0.

In cases where **left-alignment** is used, and the number of SCK pulses per frame is **lower** than the word width, the following will apply:

- Data sent and received on SDOUT and SDIN will be truncated with the LSBs being removed first.

In cases where **right-alignment** is used, and the number of SCK pulses per frame is **higher** than the configured width, the following will apply:

- For data received on SDIN, all bits before the MSB of the word value will be discarded.
- For data sent on SDOUT, all bits after the LSB of the word value will be 0 (same behavior as for left-alignment).

In cases where **right-alignment** is used, and the number of SCK pulses per frame is **lower** than the configured width, the following will apply:

- Data received on SDIN will be sign-extended to the same number of bits as the sample width before being written to memory.
- Data sent on SDOUT will be truncated with the LSBs being removed first (same behavior as for left-alignment).

8.11.7 EasyDMA

The I²S module implements EasyDMA for accessing internal Data RAM without CPU intervention.

The source and destination pointers for the TX and RX data are configured in [TXD.PTR](#) on page 337 and [RXD.PTR](#) on page 336. The memory pointed to by these pointers will only be read or written when TX or RX are enabled in [CONFIG.TXEN](#) on page 333, and [CONFIG.RXEN](#) on page 333.

The addresses written to the pointer registers [TXD.PTR](#) on page 337 and [RXD.PTR](#) on page 336 are double-buffered in hardware. These double buffers are updated for every number of transmitted data

words given by [RXTXD.MAXCNT](#) on page 337 read from/written to memory. The events TXPTRUPD and RXPTRUPD are generated whenever the TXD.PTR and RXD.PTR are transferred to these double buffers.

If [TXD.PTR](#) on page 337 is not pointing to the Data RAM region when transmission is enabled, or [RXD.PTR](#) on page 336 is not pointing to the Data RAM region when reception is enabled, an EasyDMA transfer may result in a HardFault and/or memory corruption. See [Memory](#) on page 19 for more information about the different memory regions.

Due to the nature of I²S, where the number of transmitted samples always equals the number of received samples (at least when both TX and RX are enabled), one common register [RXTXD.MAXCNT](#) on page 337 is used for specifying the sizes of these two memory buffers. The size of the buffers is specified in a number of 32-bit words. Such a 32-bit memory word can either contain one 32-bit sample, one right-aligned 24-bit sample sign extended to 32-bit, two 16-bit samples or four 8-bit samples.

In Stereo mode ([CONFIG.CHANNELS](#) on page 336=Stereo), the samples are stored as left and right sample pairs in memory. [Memory mapping for 8-bit stereo. CONFIG.SWIDTH = 8Bit, CONFIG.CHANNELS = Stereo.](#) on page 321, [Memory mapping for 16-bit stereo. CONFIG.SWIDTH = 16Bit, CONFIG.CHANNELS = Stereo.](#) on page 322 and [Memory mapping for 24-bit stereo. CONFIG.SWIDTH = 24Bit, CONFIG.CHANNELS = Stereo.](#) on page 322 show how the samples are mapped to memory in this mode. The mapping is valid for both RX and TX.

In Mono mode ([CONFIG.CHANNELS](#) on page 336=Left or Right), RX sample from only one channel in the frame is stored in memory, the other channel sample is ignored. [Memory mapping for 8-bit mono. CONFIG.SWIDTH = 8Bit, CONFIG.CHANNELS = Left.](#) on page 321, [Memory mapping for 16-bit mono, left channel only. CONFIG.SWIDTH = 16Bit, CONFIG.CHANNELS = Left.](#) on page 322 and [Memory mapping for 24-bit mono, left channel only. CONFIG.SWIDTH = 24Bit, CONFIG.CHANNELS = Left.](#) on page 323 show how RX samples are mapped to memory in this mode. For TX, the same outgoing sample read from memory is transmitted on both left and right in a frame, resulting in a mono output stream.

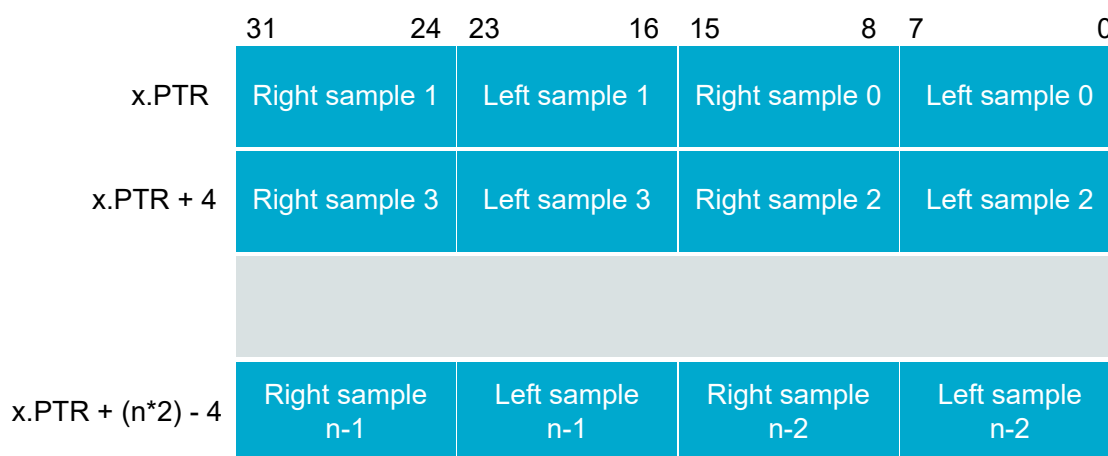


Figure 75: Memory mapping for 8-bit stereo. [CONFIG.SWIDTH = 8Bit, CONFIG.CHANNELS = Stereo.](#)

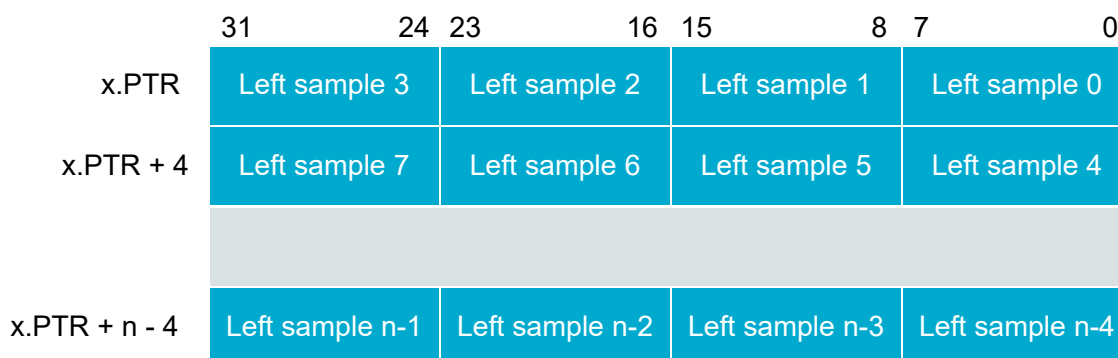


Figure 76: Memory mapping for 8-bit mono. [CONFIG.SWIDTH = 8Bit, CONFIG.CHANNELS = Left.](#)

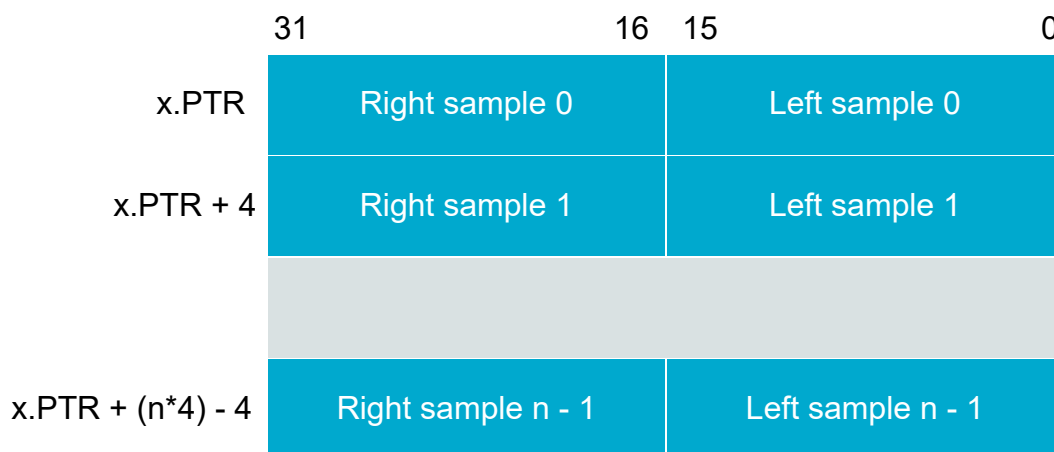


Figure 77: Memory mapping for 16-bit stereo. CONFIG.SWIDTH = 16Bit, CONFIG.CHANNELS = Stereo.

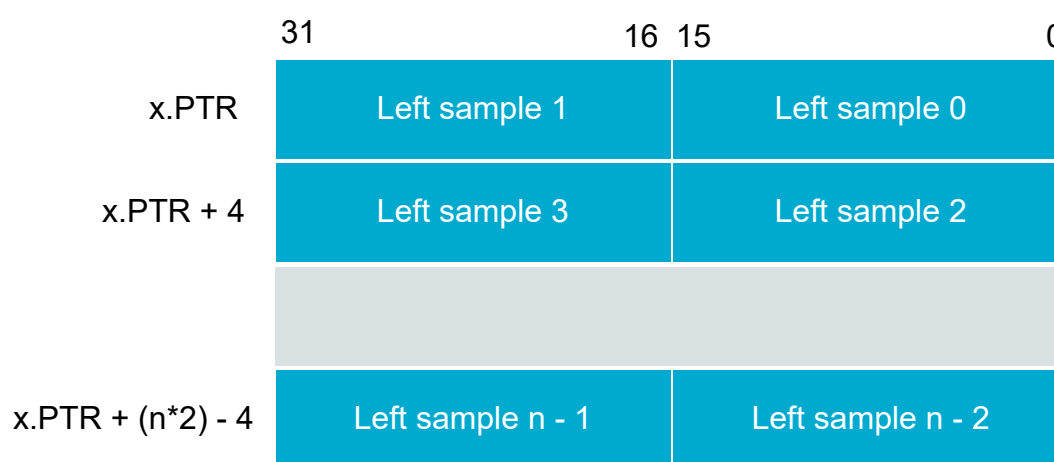


Figure 78: Memory mapping for 16-bit mono, left channel only. CONFIG.SWIDTH = 16Bit, CONFIG.CHANNELS = Left.

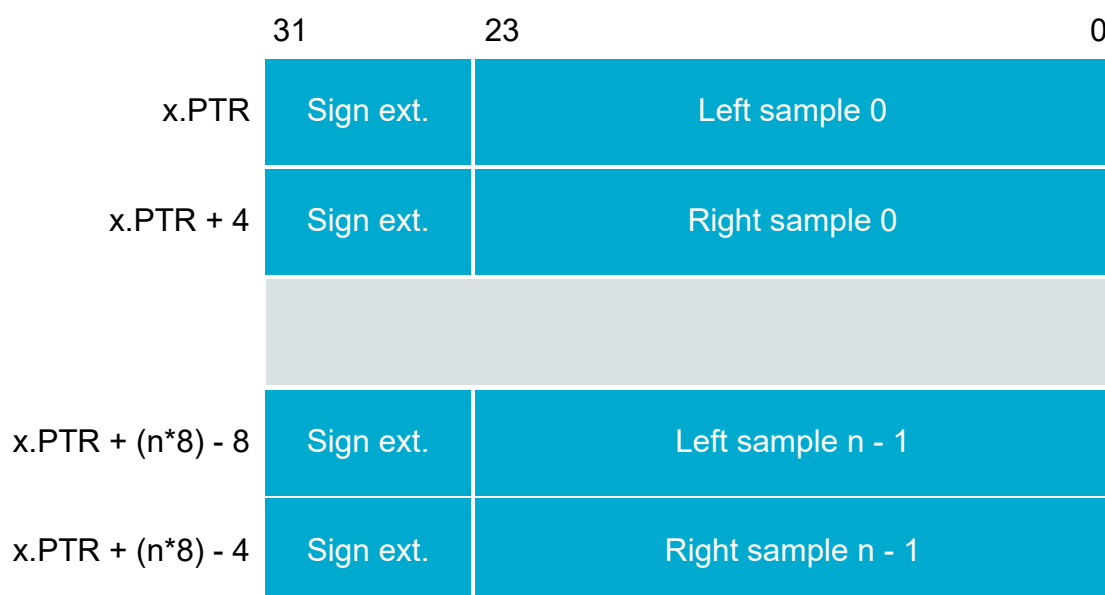


Figure 79: Memory mapping for 24-bit stereo. CONFIG.SWIDTH = 24Bit, CONFIG.CHANNELS = Stereo.

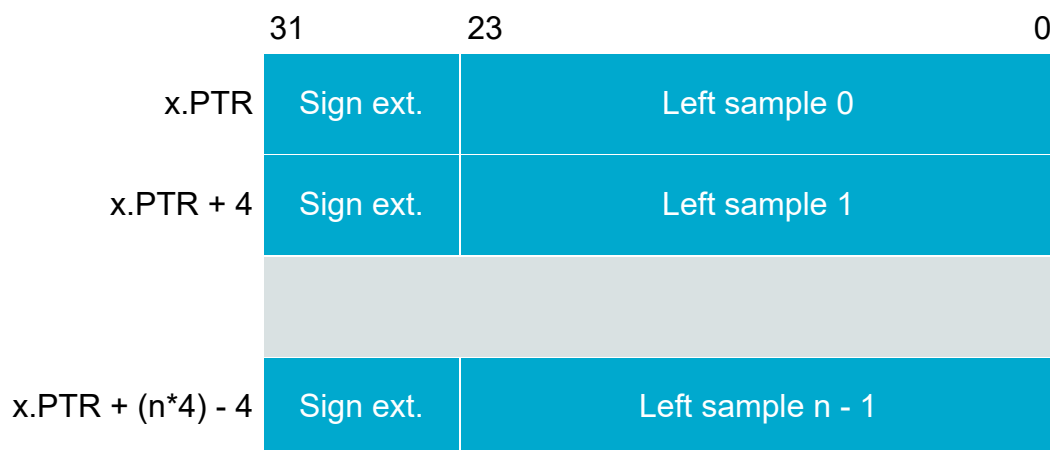


Figure 80: Memory mapping for 24-bit mono, left channel only. CONFIG.SWIDTH = 24Bit, CONFIG.CHANNELS = Left.

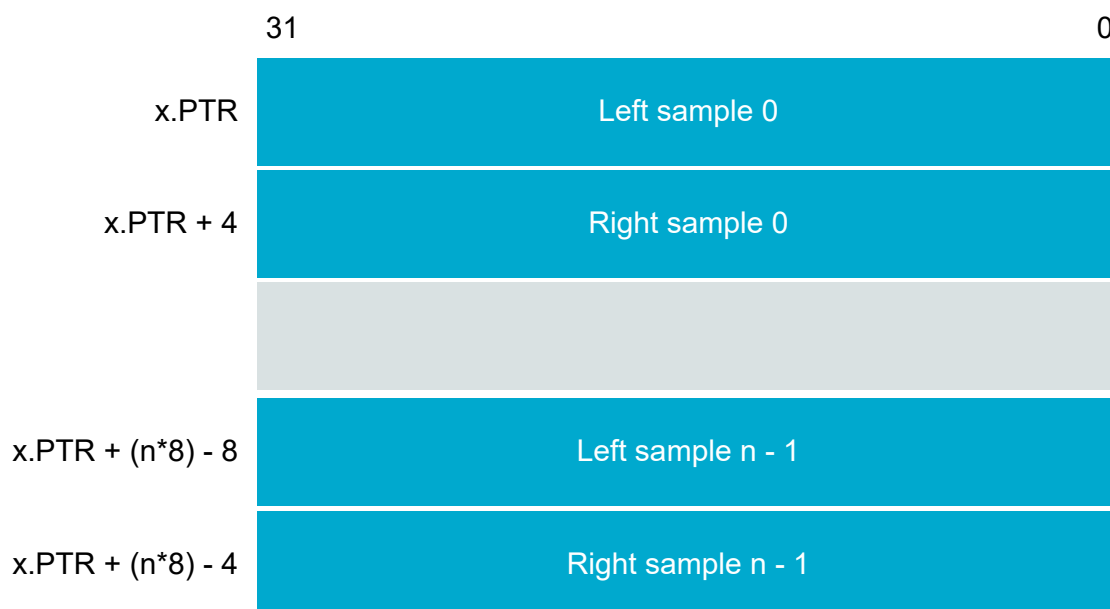


Figure 81: Memory mapping for 32-bit stereo. CONFIG.SWIDTH = 32Bit, CONFIG.CHANNELS = Stereo.

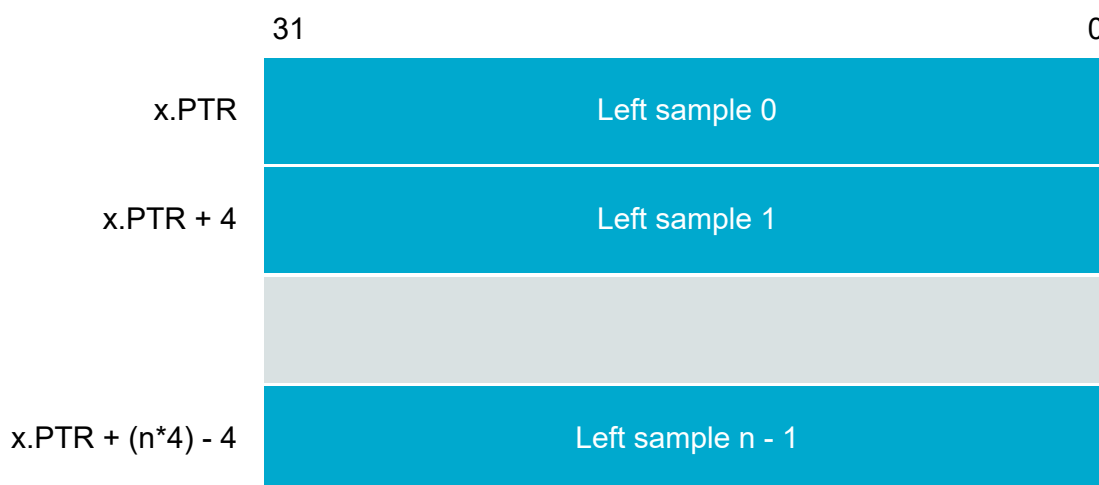


Figure 82: Memory mapping for 32-bit mono, left channel only. CONFIG.SWIDTH = 32Bit, CONFIG.CHANNELS = Left.

8.11.8 Module operation

Described here is a typical operating procedure for the I²S module.

1. Configure the I²S module using the CONFIG registers

```
// Enable reception
NRF_I2S->CONFIG.RXEN = (I2S_CONFIG_RXEN_RXEN_Enabled <<
                        I2S_CONFIG_RXEN_RXEN_Pos);

// Enable transmission
NRF_I2S->CONFIG.TXEN = (I2S_CONFIG_TXEN_TXEN_Enabled <<
                        I2S_CONFIG_TXEN_TXEN_Pos);

// Enable MCK generator
NRF_I2S->CONFIG.MCKEN = (I2S_CONFIG_MCKEN_MCKEN_Enabled <<
                        I2S_CONFIG_MCKEN_MCKEN_Pos);

// MCKFREQ = 4 MHz
NRF_I2S->CONFIG.MCKFREQ = I2S_CONFIG_MCKFREQ_MCKFREQ_32MDIV8 <<
                        I2S_CONFIG_MCKFREQ_MCKFREQ_Pos;

// Ratio = 256
NRF_I2S->CONFIG.RATIO = I2S_CONFIG_RATIO_RATIO_256X <<
                        I2S_CONFIG_RATIO_RATIO_Pos;
// MCKFREQ = 4 MHz and Ratio = 256 gives sample rate = 15.625 ks/s
// Sample width = 16 bit
NRF_I2S->CONFIG.SWIDTH = I2S_CONFIG_SWIDTH_SWIDTH_16Bit <<
                        I2S_CONFIG_SWIDTH_SWIDTH_Pos;

// Alignment = Left
NRF_I2S->CONFIG.ALIGN = I2S_CONFIG_ALIGN_ALIGN_Left <<
                        I2S_CONFIG_ALIGN_ALIGN_Pos;

// Format = I2S
NRF_I2S->CONFIG.FORMAT = I2S_CONFIG_FORMAT_FORMAT_I2S <<
                        I2S_CONFIG_FORMAT_FORMAT_Pos;

// Use stereo
NRF_I2S->CONFIG.CHANNELS = I2S_CONFIG_CHANNELS_CHANNELS_Stereo <<
                        I2S_CONFIG_CHANNELS_CHANNELS_Pos;
```


2. Map IO pins using the PINSEL registers

```
// MCK routed to pin 0
NRF_I2S->PSEL.MCK = (0 << I2S_PSEL_MCK_PIN_Pos) |
                    (I2S_PSEL_MCK_CONNECT_Connected <<
                     I2S_PSEL_MCK_CONNECT_Pos);

// SCK routed to pin 1
NRF_I2S->PSEL.SCK = (1 << I2S_PSEL_SCK_PIN_Pos) |
                    (I2S_PSEL_SCK_CONNECT_Connected <<
                     I2S_PSEL_SCK_CONNECT_Pos);

// LRCK routed to pin 2
NRF_I2S->PSEL.LRCK = (2 << I2S_PSEL_LRCK_PIN_Pos) |
                     (I2S_PSEL_LRCK_CONNECT_Connected <<
                      I2S_PSEL_LRCK_CONNECT_Pos);

// SDOUT routed to pin 3
NRF_I2S->PSEL.SDOUT = (3 << I2S_PSEL_SDOUT_PIN_Pos) |
                      (I2S_PSEL_SDOUT_CONNECT_Connected <<
                       I2S_PSEL_SDOUT_CONNECT_Pos);

// SDIN routed on pin 4
NRF_I2S->PSEL.SDIN = (4 << I2S_PSEL_SDIN_PIN_Pos) |
                     (I2S_PSEL_SDIN_CONNECT_Connected <<
                      I2S_PSEL_SDIN_CONNECT_Pos);
```

3. Configure TX and RX data pointers using the TXD, RXD and RXTXD registers

```
NRF_I2S->TXD.PTR = my_tx_buf;
NRF_I2S->RXD.PTR = my_rx_buf;
NRF_I2S->TXD.MAXCNT = MY_BUF_SIZE;
```

4. Enable the I²S module using the ENABLE register

```
NRF_I2S->ENABLE = 1;
```

5. Start audio streaming using the START task

```
NRF_I2S->TASKS_START = 1;
```

6. Handle received and transmitted data when receiving the TXPTRUPD and RXPTRUPD events

```
if (NRF_I2S->EVENTS_TXPTRUPD != 0)
{
    NRF_I2S->TXD.PTR = my_next_tx_buf;
    NRF_I2S->EVENTS_TXPTRUPD = 0;
}

if (NRF_I2S->EVENTS_RXPTRUPD != 0)
{
    NRF_I2S->RXD.PTR = my_next_rx_buf;
    NRF_I2S->EVENTS_RXPTRUPD = 0;
}
```

8.11.9 Pin configuration

The MCK, SCK, LRCK, SDIN and SDOUT signals associated with the I²S module are mapped to physical pins according to the pin numbers specified in the PSEL.x registers.

These pins are acquired whenever the I²S module is enabled through the register [ENABLE](#) on page 332.

When a pin is acquired by the I²S module, the direction of the pin (input or output) will be configured automatically, and any pin direction setting done in the GPIO module will be overridden. The directions for the various I²S pins are shown below in [GPIO configuration before enabling peripheral \(Master mode\)](#) on page 326 and [GPIO configuration before enabling peripheral \(Slave mode\)](#) on page 326.

To secure correct signal levels on the pins in System OFF mode, and when the I²S module is disabled, these pins must be configured in the GPIO peripheral directly.

I ² S signal	I ² S pin	Direction	Output value	Comment
MCK	As specified in PSEL.MCK	Output	0	
LRCK	As specified in PSEL.LRCK	Output	0	
SCK	As specified in PSEL.SCK	Output	0	
SDIN	As specified in PSEL.SDIN	Input	Not applicable	
SDOUT	As specified in PSEL.SDOUT	Output	0	

Table 39: GPIO configuration before enabling peripheral (Master mode)

I ² S signal	I ² S pin	Direction	Output value	Comment
MCK	As specified in PSEL.MCK	Output	0	
LRCK	As specified in PSEL.LRCK	Input	Not applicable	
SCK	As specified in PSEL.SCK	Input	Not applicable	
SDIN	As specified in PSEL.SDIN	Input	Not applicable	
SDOUT	As specified in PSEL.SDOUT	Output	0	

Table 40: GPIO configuration before enabling peripheral (Slave mode)

8.11.10 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
I2S20 : S	GLOBAL	0x500DD000	US	S	SA	No	Inter-IC sound interface I2S20
I2S20 : NS		0x400DD000					

Configuration

Instance	Domain	Configuration
I2S20 : S	GLOBAL	Available GPIO port: P1
I2S20 : NS		

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Starts continuous I2S transfer. Also starts MCK generator when this is enabled

Register	Offset	TZ	Description
TASKS_STOP	0x004		Stops I2S transfer and MCK generator. Triggering this task will cause the event STOPPED to be generated.
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
EVENTS_RXPTRUPD	0x104		The RXD.PTR register has been copied to internal double-buffers. When the I2S module is started and RX is enabled, this event will be generated for every RXTXD.MAXCNT words received on the SDIN pin.
EVENTS_STOPPED	0x108		I2S transfer stopped.
EVENTS_TXPTRUPD	0x114		The TDX.PTR register has been copied to internal double-buffers. When the I2S module is started and TX is enabled, this event will be generated for every RXTXD.MAXCNT words that are sent on the SDOUT pin.
EVENTS_FRAMESTART	0x11C		Frame start event, generated on the active edge of LRCK
PUBLISH_RXPTRUPD	0x184		Publish configuration for event RXPTRUPD
PUBLISH_STOPPED	0x188		Publish configuration for event STOPPED
PUBLISH_TXPTRUPD	0x194		Publish configuration for event TXPTRUPD
PUBLISH_FRAMESTART	0x19C		Publish configuration for event FRAMESTART
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
ENABLE	0x500		Enable I2S module
CONFIG.MODE	0x504		I2S mode
CONFIG.RXEN	0x508		Reception (RX) enable
CONFIG.TXEN	0x50C		Transmission (TX) enable
CONFIG.MCKEN	0x510		Master clock generator enable
CONFIG.MCKFREQ	0x514		I2S clock generator control
CONFIG.RATIO	0x518		MCK / LRCK ratio
CONFIG.SWIDTH	0x51C		Sample width
CONFIG.ALIGN	0x520		Alignment of sample within a frame
CONFIG.FORMAT	0x524		Frame format
CONFIG.CHANNELS	0x528		Enable channels
RXD.PTR	0x538		Receive buffer RAM start address.
TXD.PTR	0x540		Transmit buffer RAM start address
RXTXD.MAXCNT	0x550		Size of RXD and TXD buffers
PSEL.MCK	0x560		Pin select for MCK signal
PSEL.SCK	0x564		Pin select for SCK signal
PSEL.LRCK	0x568		Pin select for LRCK signal
PSEL.SDIN	0x56C		Pin select for SDIN signal
PSEL.SDOUT	0x570		Pin select for SDOUT signal
CHANNEL[n].TERMINATEONBUSERROR	0x580		Terminate the transaction if a BUSERROR event is detected.
CHANNEL[n].BUSERRORADDRESS	0x584		Address of transaction that generated the last BUSERROR event.

8.11.10.1 TASKS_START

Address offset: 0x000

Starts continuous I2S transfer. Also starts MCK generator when this is enabled

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_START						Starts continuous I2S transfer. Also starts MCK generator when this is enabled																											
		Trigger		1				Trigger task																											

8.11.10.2 TASKS_STOP

Address offset: 0x004

Stops I2S transfer and MCK generator. Triggering this task will cause the event **STOPPED** to be generated.

Bit number		31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
ID		A	
Reset 0x00000000		0 0	
ID	R/W	Field	Description
A	W	TASKS_STOP	Stops I2S transfer and MCK generator. Triggering this task will cause the event STOPPED to be generated.
		Trigger	1 Trigger task

8.11.10.3 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task **START**

Bit number		31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
ID		B	
Reset 0x00000000		0 0	
ID	R/W	Field	Description
A	RW	CHIDX	DPPI channel that task START will subscribe to
B	RW	EN	Disabled
			Enabled

8.11.10.4 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task **STOP**

Bit number		31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	
ID		B	
Reset 0x00000000		0 0	
ID	R/W	Field	Description
A	RW	CHIDX	DPPI channel that task STOP will subscribe to
B	RW	EN	Disabled
			Enabled

8.11.10.5 EVENTS_RXPTRUPD

Address offset: 0x104

The RXD.PTR register has been copied to internal double-buffers. When the I2S module is started and RX is enabled, this event will be generated for every RXTXD.MAXCNT words received on the SDIN pin.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_RXPTRUPD			The RXD.PTR register has been copied to internal double-buffers. When the I2S module is started and RX is enabled, this event will be generated for every RXTXD.MAXCNT words received on the SDIN pin.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.11.10.6 EVENTS_STOPPED

Address offset: 0x108

I2S transfer stopped.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	EVENTS_STOPPED				I2S transfer stopped.																													
			NotGenerated	0		Event not generated																													
			Generated	1		Event generated																													

8.11.10.7 EVENTS_TXPTRUPD

Address offset: 0x114

The TDX.PTR register has been copied to internal double-buffers. When the I2S module is started and TX is enabled, this event will be generated for every RXTXD.MAXCNT words that are sent on the SDOUT pin.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_TXPTRUPD			The TDX.PTR register has been copied to internal double-buffers. When the I2S module is started and TX is enabled, this event will be generated for every RXTXD.MAXCNT words that are sent on the SDOUT pin.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.11.10.8 EVENTS_FRAMESTART

Address offset: 0x11C

Frame start event, generated on the active edge of LRCK

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_FRAMESTART			Frame start event, generated on the active edge of LRCK																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.11.10.9 PUBLISH_RXPTRUPD

Address offset: 0x184

Publish configuration for event **RXPTRUPD**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event RXPTRUPD will publish to																													
B	RW	EN																																	
			Disabled	0		Disable publishing																													
			Enabled	1		Enable publishing																													

8.11.10.10 PUBLISH_STOPPED

Address offset: 0x188

Publish configuration for event **STOPPED**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event STOPPED will publish to																													
B	RW	EN																																	
			Disabled	0		Disable publishing																													
			Enabled	1		Enable publishing																													

8.11.10.11 PUBLISH_TXPTRUPD

Address offset: 0x194

Publish configuration for event **TXPTRUPD**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event TXPTRUPD will publish to																													
B	RW	EN																																	
			Disabled	0		Disable publishing																													
			Enabled	1		Enable publishing																													

8.11.10.12 PUBLISH_FRAMESTART

Address offset: 0x19C

Publish configuration for event **FRAMESTART**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																														
ID				B																								A				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A																
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																									
A	RW	CHIDX		[0..255]				DPPI channel that event FRAMESTART will publish to																																																									
B	RW	EN																																																															
			Disabled	0				Disable publishing																																																									
			Enabled	1				Enable publishing																																																									

8.11.10.13 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																												D		C				B		A		
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	RXPTRUPD			Enable or disable interrupt for event RXPTRUPD																																	
			Disabled	0	Disable																																	
			Enabled	1	Enable																																	
B	RW	STOPPED			Enable or disable interrupt for event STOPPED																																	
			Disabled	0	Disable																																	
			Enabled	1	Enable																																	
C	RW	TXPTRUPD			Enable or disable interrupt for event TXPTRUPD																																	
			Disabled	0	Disable																																	
			Enabled	1	Enable																																	
D	RW	FRAMESTART			Enable or disable interrupt for event FRAMESTART																																	
			Disabled	0	Disable																																	
			Enabled	1	Enable																																	

8.11.10.14 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																												D	C					B	A			
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
D	R/W	Field	Value ID	Value	Description																																	
A	RW	RXPTRUPD			Write '1' to enable interrupt for event RXPTRUPD																																	
			Set	1	Enable																																	
			Disabled	0	Read: Disabled																																	
			Enabled	1	Read: Enabled																																	
B	RW	STOPPED			Write '1' to enable interrupt for event STOPPED																																	
			Set	1	Enable																																	
			Disabled	0	Read: Disabled																																	
			Enabled	1	Read: Enabled																																	
C	RW	TXPTRUPD			Write '1' to enable interrupt for event TXPTRUPD																																	
			Set	1	Enable																																	
			Disabled	0	Read: Disabled																																	
			Enabled	1	Read: Enabled																																	
D	RW	FRAMESTART			Write '1' to enable interrupt for event FRAMESTART																																	

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
			Set	1		Enable																													
			Disabled	0		Read: Disabled																													
			Enabled	1		Read: Enabled																													

8.11.10.15 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																												D	C					B	A
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	RXPTRUPD			Write '1' to disable interrupt for event RXPTRUPD																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	STOPPED			Write '1' to disable interrupt for event STOPPED																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	TXPTRUPD			Write '1' to disable interrupt for event TXPTRUPD																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	FRAMESTART			Write '1' to disable interrupt for event FRAMESTART																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.11.10.16 ENABLE

Address offset: 0x500

Enable I2S module

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																				A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value		Description																															
A	RW	ENABLE				Enable I2S module																															
			Disabled	0		Disable																															
			Enabled	1		Enable																															

8.11.10.17 CONFIG.MODE

Address offset: 0x504

I2S mode

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	MODE			I2S mode																														
			Master	0	Master mode. SCK and LRCK generated from internal master clcok (MCK) and output on pins defined by PSEL.xxx.																														
			Slave	1	Slave mode. SCK and LRCK generated by external master and received on pins defined by PSEL.xxx																														

8.11.10.18 CONFIG.RXEN

Address offset: 0x508

Reception (RX) enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	RXEN						Reception (RX) enable																											
			Disabled	0				Reception disabled and now data will be written to the RXD.PTR address.																											
			Enabled	1				Reception enabled.																											

8.11.10.19 CONFIG.TXEN

Address offset: 0x50C

Transmission (TX) enable

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																							A
Reset 0x00000001					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1		
ID	R/W	Field	Value ID	Value	Description																																		
A	RW	TXEN			Transmission (TX) enable																																		
			Disabled	0	Transmission disabled and now data will be read from the RXD.TXD address.																																		
			Enabled	1	Transmission enabled.																																		

8.11.10.20 CONFIG.MCKEN

Address offset: 0x510

Master clock generator enable

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000001				0 1																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	MCKEN			Master clock generator enable																														
			Disabled	0	Master clock generator disabled and PSEL.MCK not connected(available as GPIO).																														
			Enabled	1	Master clock generator running and MCK output on PSEL.MCK.																														

8.11.10.21 CONFIG.MCKFREQ

Address offset: 0x514

I2S clock generator control

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			A A																															
Reset 0x20000000			0 0 1 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	MCKFREQ			I2S MCK frequency configuration																													
					NOTE: Enumerations are deprecated, use MCKFREQ equation.																													
					NOTE: The 12 least significant bits of the register are ignored and shall be set to zero.																													
			32MDIV2	0x80000000	32 MHz / 2 = 16.0 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV3	0x50000000	32 MHz / 3 = 10.6666667 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV4	0x40000000	32 MHz / 4 = 8.0 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV5	0x30000000	32 MHz / 5 = 6.4 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV6	0x28000000	32 MHz / 6 = 5.3333333 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV8	0x20000000	32 MHz / 8 = 4.0 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV10	0x18000000	32 MHz / 10 = 3.2 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV11	0x16000000	32 MHz / 11 = 2.9090909 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV15	0x11000000	32 MHz / 15 = 2.1333333 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV16	0x10000000	32 MHz / 16 = 2.0 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV21	0x0C000000	32 MHz / 21 = 1.5238095 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV23	0x0B000000	32 MHz / 23 = 1.3913043 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV30	0x08800000	32 MHz / 30 = 1.0666667 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV31	0x08400000	32 MHz / 31 = 1.0322581 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV32	0x08000000	32 MHz / 32 = 1.0 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV42	0x06000000	32 MHz / 42 = 0.7619048 MHz																													
					Deprecated, use MCKFREQ equation.																													
			32MDIV63	0x04100000	32 MHz / 63 = 0.5079365 MHz																													
					Deprecated, use MCKFREQ equation.																													

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x20000000				0 0 1 0																															
ID	R/W	Field	Value ID	Value								Description																							

32MDIV125

0x020C0000

32 MHz / 125 = 0.256 MHz

Deprecated, use MCKFREQ equation.

8.11.10.22 CONFIG.RATIO

Address offset: 0x518

MCK / LRCK ratio

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																																			
Reset 0x00000006				0 0																															

A RW RATIO

MCK / LRCK ratio

32X

0

LRCK = MCK / 32

48X

1

LRCK = MCK / 48

64X

2

LRCK = MCK / 64

96X

3

LRCK = MCK / 96

128X

4

LRCK = MCK / 128

192X

5

LRCK = MCK / 192

256X

6

LRCK = MCK / 256

384X

7

LRCK = MCK / 384

512X

8

LRCK = MCK / 512

8.11.10.23 CONFIG.SWIDTH

Address offset: 0x51C

Sample width

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000001				0 0																															

A RW SWIDTH

Sample and half-frame width

8Bit

0

8 bit sample.

16Bit

1

16 bit sample.

24Bit

2

24 bit sample.

32Bit

3

32 bit sample.

8BitIn16

4

8 bit sample in a 16-bit half-frame.

8BitIn32

5

8 bit sample in a 32-bit half-frame.

16BitIn32

6

16 bit sample in a 32-bit half-frame.

24BitIn32

7

24 bit sample in a 32-bit half-frame.

8.11.10.24 CONFIG.ALIGN

Address offset: 0x520

Alignment of sample within a frame

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	ALIGN				Alignment of sample within a frame																													
			Left	0		Left-aligned.																													
			Right	1		Right-aligned.																													

8.11.10.25 CONFIG.FORMAT

Address offset: 0x524

Frame format

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	FORMAT			Frame format																														
			I2S	0	Original I2S format.																														
			Aligned	1	Alternate (left- or right-aligned) format.																														

8.11.10.26 CONFIG.CHANNELS

Address offset: 0x528

Enable channels

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CHANNELS			Enable channels																														
			Stereo	0	Stereo.																														
			Left	1	Left only.																														
			Right	2	Right only.																														

8.11.10.27 RXD.PTR

Address offset: 0x538

Receive buffer RAM start address.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.11.10.28 TXD.PTR

Address offset: 0x540

Transmit buffer RAM start address

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x20000000				0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						Transmit buffer Data RAM start address. When transmitting, words containing samples will be fetched from this address. This address is a word aligned Data RAM address.																											

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.11.10.29 RXTXD.MAXCNT

Address offset: 0x550

Size of RXD and TXD buffers

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	MAXCNT						Size of RXD and TXD buffers in number of 32 bit words																											

8.11.10.30 PSEL.MCK

Address offset: 0x560

Pin select for MCK signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				C																								B				B	B	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	PIN		[0..31]		Pin number																															
B	RW	PORT		[0..7]		Port number																															
C	RW	CONNECT				Connection																															
			Disconnected	1		Disconnect																															
			Connected	0		Connect																															

8.11.10.31 PSEL.SCK

Address offset: 0x564

Pin select for SCK signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				C																								B				B	B	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
ID	R/W	Field	Value ID	Value				Description																													
A	RW	PIN		[0..31]				Pin number																													
B	RW	PORT		[0..7]				Port number																													
C	RW	CONNECT						Connection																													
			Disconnected	1				Disconnect																													
			Connected	0				Connect																													

8.11.10.32 PSEL.LRCK

Address offset: 0x568

Pin select for LRCK signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				C																								B				B	B	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			
ID	R/W	Field	Value ID	Value				Description																														
A	RW	PIN		[0..31]				Pin number																														
B	RW	PORT		[0..7]				Port number																														
C	RW	CONNECT						Connection																														
			Disconnected	1				Disconnect																														
			Connected	0				Connect																														

8.11.10.33 PSEL.SDIN

Address offset: 0x56C

Pin select for SDIN signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				C																								B				B	B	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			
ID	R/W	Field	Value ID	Value				Description																														
A	RW	PIN		[0..31]				Pin number																														
B	RW	PORT		[0..7]				Port number																														
C	RW	CONNECT						Connection																														
			Disconnected	1				Disconnect																														
			Connected	0				Connect																														

8.11.10.34 PSEL.SDOUT

Address offset: 0x570

Pin select for SDOUT signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				C																								B				B	B	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			
ID	R/W	Field	Value ID	Value				Description																														
A	RW	PIN		[0..31]				Pin number																														
B	RW	PORT		[0..7]				Port number																														
C	RW	CONNECT						Connection																														
			Disconnected	1				Disconnect																														
			Connected	0				Connect																														

8.11.10.35 CHANNEL[n].TERMINATEONBUSERROR (n=0..1)

Address offset: 0x580 + (n × 0x8)

Terminate the transaction if a BUSERROR event is detected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ENABLE																																	
			Disabled	0				Disable																											
			Enabled	1				Enable																											

8.11.10.36 CHANNEL[n].BUSERRORADDRESS (n=0..1)

Address offset: 0x584 + (n × 0x8)

Address of transaction that generated the last BUSERROR event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	ADDRESS																																	

8.12 LPCOMP — Low-power comparator

The low-power comparator (LPCOMP) peripheral compares an input voltage against a reference voltage.

The main features of LPCOMP are:

- Input range of 0 to VDD
- Ultra-low power
- Eight input options (AIN0 to AIN7)
- Two reference voltage options:
 - Two external analog reference inputs
 - 15-level internal reference ladder (VDD/16)
- Optional hysteresis enable on input
- Wakeup source from System OFF or System ON sleep

In System ON, LPCOMP can generate separate events on rising and falling edges of a signal, or sample the current state of the pin to determine if it is above or below the selected reference. The block is configurable to use any of the analog inputs on the device. Additionally, LPCOMP can be used as an analog

wakeup source from System ON idle or System OFF. The comparator threshold is programmable to a range of supply voltage fractions.

Note: LPCOMP cannot be used (STARTed) at the same time as COMP. Only one comparator can be used at a time.

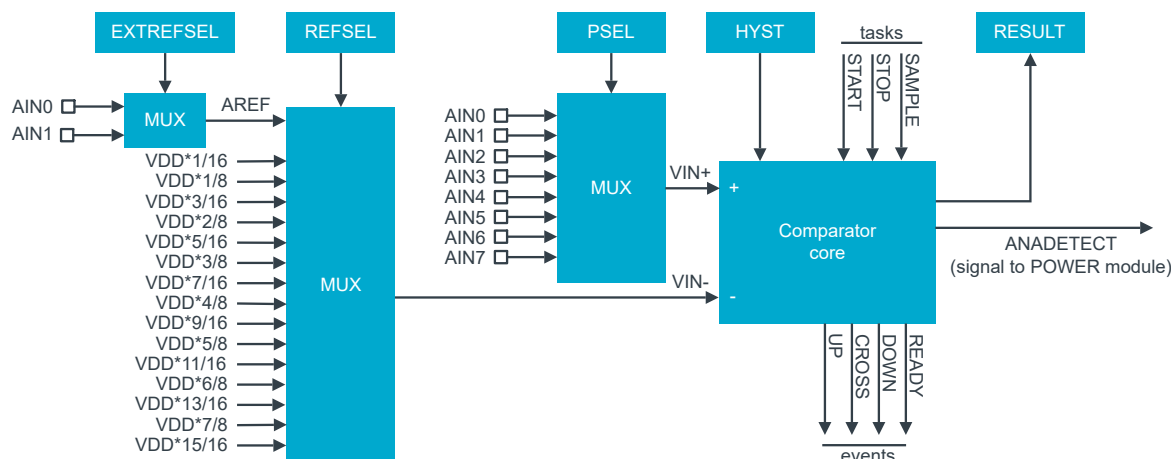


Figure 83: LPCOMP block diagram

The LPCOMP peripheral compares an input voltage (VIN+) from an analog input pin selected via the PSEL register, against a reference voltage (VIN-) selected via registers REFSEL on page 349 and EXTREFSEL.

The PSEL, REFSEL, and EXTREFSEL registers must be configured before LPCOMP is enabled through the ENABLE register.

The HYST register allows enabling an optional hysteresis in the comparator core. This hysteresis prevents noise on the signal, which would create unwanted events. The following figure illustrates the effect of an active hysteresis on a noisy input signal. It is disabled by default, and must be configured before enabling LPCOMP.

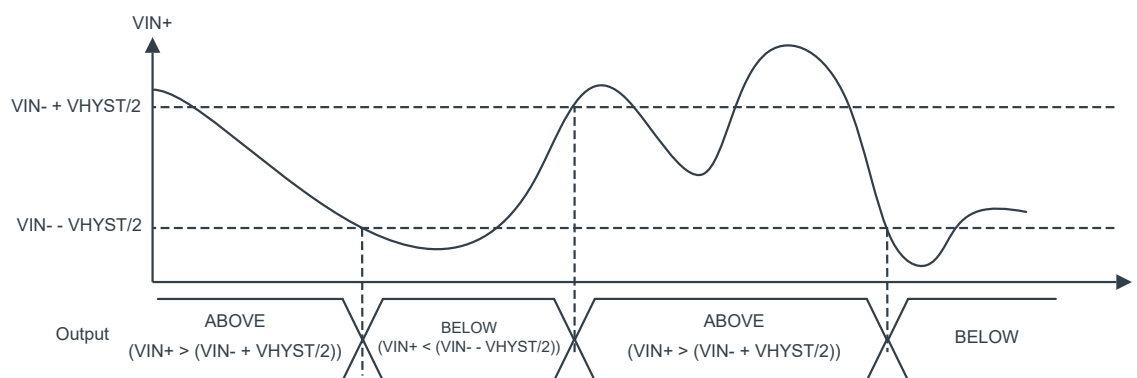


Figure 84: Effect of hysteresis on a noisy input signal

LPCOMP is started by triggering the START task. After a startup time of $t_{LPCOMP,STARTUP}$, LPCOMP generates a READY event to indicate that the comparator is ready to use and the output of LPCOMP is correct. LPCOMP generates events every time VIN+ crosses VIN-. More specifically, every time VIN+ rises above VIN- (upward crossing), an UP event and CROSS event are generated. Every time VIN+ falls below VIN- (downward crossing), a DOWN event and CROSS event are generated. When hysteresis is enabled, the upward crossing level becomes $VIN- + VHYST/2$, and the downward crossing level becomes $VIN- - VHYST/2$.

LPCOMP is stopped by triggering the STOP task.

LPCOMP is operational in both System ON and System OFF mode when enabled through the **ENABLE** register. See [POWER — Power control](#) on page 97 for more information about power modes. Entering System OFF is not allowed when a READY event is pending to be generated.

All LPCOMP registers, including **ENABLE**, are classified as retained registers when the LPCOMP is enabled. However, when the device wakes up from System OFF, all LPCOMP registers are reset.

LPCOMP can wake up the system from System OFF by asserting the ANADETECT signal. The ANADETECT signal can be derived from any of the event sources that generate UP, DOWN, and CROSS events. If wakeup from System OFF occurs, only the ANADETECT signal is generated. See the ANADETECT register ([ANADETECT](#) on page 350) for more information on configuring the ANADETECT signal.

The immediate value of the LPCOMP can be sampled to **RESULT** on page 348 by triggering the **SAMPLE** task.

See [RESETREAS](#) on page 109 for more information on how to detect a wakeup from LPCOMP.

8.12.1 Shared resources

LPCOMP shares analog resources with SAADC. While it is possible to use the SAADC at the same time as the LPCOMP, selecting the same analog input pin for both modules is not supported.

Additionally, LPCOMP shares registers and other resources with other peripherals that have the same ID as the LPCOMP. See [Peripherals with shared ID](#) on page 214 for more information.

The LPCOMP peripheral should not be disabled (by writing to the **ENABLE** register) before the peripheral has stopped. Failing to do so may result in unpredictable behavior.

8.12.2 Pin configuration

The LPCOMP.PSEL register is used to select an analog input pin for LPCOMP. The pins available are **AIN0** through **AIN7**.

See [GPIO — General purpose input/output](#) on page 273 for more information about the pins. Similarly, you can use [EXTREFSEL](#) on page 349 to select one of the analog reference input pins, **AIN0** and **AIN1**, as input for AREF if it is selected in [EXTREFSEL](#) on page 349. The selected analog pins are acquired by LPCOMP when it is enabled through [ENABLE](#) on page 348.

8.12.3 Registers

Instances

Instance	Domain	Base address	TrustZone			Split	Description
			Map	Att	DMA		
LPCOMP : S	GLOBAL	0x50106000	US	S	NA	No	Low-power comparator LPCOMP
LPCOMP : NS		0x40106000					

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Start comparator
TASKS_STOP	0x004		Stop comparator
TASKS_SAMPLE	0x008		Sample comparator value. This task requires that LPCOMP has been started by the START task.
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
SUBSCRIBE_SAMPLE	0x088		Subscribe configuration for task SAMPLE

Register	Offset	TZ	Description
EVENTS_READY	0x100		LPCOMP is ready and output is valid
EVENTS_DOWN	0x104		Downward crossing
EVENTS_UP	0x108		Upward crossing
EVENTS_CROSS	0x10C		Downward or upward crossing
PUBLISH_READY	0x180		Publish configuration for event READY
PUBLISH_DOWN	0x184		Publish configuration for event DOWN
PUBLISH_UP	0x188		Publish configuration for event UP
PUBLISH_CROSS	0x18C		Publish configuration for event CROSS
SHORTS	0x200		Shortcuts between local events and tasks
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
INTPEND	0x30C		Pending interrupts
RESULT	0x400		Compare result
ENABLE	0x500		Enable LPCOMP
PSEL	0x504		Input pin select
REFSEL	0x508		Reference select
EXTREFSEL	0x50C		External reference select
ANADETECT	0x520		Analog detect configuration
HYST	0x538		Comparator hysteresis enable

8.12.3.1 TASKS_START

Address offset: 0x000

Start comparator

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_START						Start comparator																											
			Trigger	1				Trigger task																											

8.12.3.2 TASKS_STOP

Address offset: 0x004

Stop comparator

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_STOP						Stop comparator																											
			Trigger	1				Trigger task																											

8.12.3.3 TASKS_SAMPLE

Address offset: 0x008

Sample comparator value. This task requires that LPCOMP has been started by the START task.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_SAMPLE						Sample comparator value. This task requires that LPCOMP has been started by the START task.																											
			Trigger	1				Trigger task																											

8.12.3.4 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task **START**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task START will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.12.3.5 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task **STOP**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0 0																																		
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that task STOP will subscribe to																																
B	RW	EN																																				
			Disabled	0		Disable subscription																																
			Enabled	1		Enable subscription																																

8.12.3.6 SUBSCRIBE_SAMPLE

Address offset: 0x088

Subscribe configuration for task **SAMPLE**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that task SAMPLE will subscribe to																																																									
B	RW	EN																																																													
			Disabled	0		Disable subscription																																																									
			Enabled	1		Enable subscription																																																									

8.12.3.7 EVENTS_READY

Address offset: 0x100

LPCOMP is ready and output is valid

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	EVENTS_READY			LPCOMP is ready and output is valid																															
			NotGenerated	0	Event not generated																															
			Generated	1	Event generated																															

8.12.3.8 EVENTS_DOWN

Address offset: 0x104

Downward crossing

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	EVENTS_DOWN			Downward crossing																															
			NotGenerated	0	Event not generated																															
			Generated	1	Event generated																															

8.12.3.9 EVENTS_UP

Address offset: 0x108

Upward crossing

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	EVENTS_UP			Upward crossing																															
			NotGenerated	0	Event not generated																															
			Generated	1	Event generated																															

8.12.3.10 EVENTS_CROSS

Address offset: 0x10C

Downward or upward crossing

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	EVENTS_CROSS			Downward or upward crossing																															
			NotGenerated	0	Event not generated																															
			Generated	1	Event generated																															

Publish configuration for event **READY**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
ID				B																								A				A				A				A			
Reset 0x00000000				0																																							
ID	R/W	Field	Value ID	Value		Description																																					
A	RW	CHIDX		[0..255]		DPPI channel that event READY will publish to																																					
B	RW	EN																																									
			Disabled	0		Disable publishing																																					
			Enabled	1		Enable publishing																																					

Publish configuration for event **DOWN**

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID					B																								A				A	A	A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value	ID	Value				Description																														
A	RW	CHIDX			[0..255]				DPPI channel that event DOWN will publish to																														
B	RW	EN																																					
			Disabled		0				Disable publishing																														
			Enabled		1				Enable publishing																														

Publish configuration for event UP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															

Publish configuration for event [CROSS](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event CROSS will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.12.3.15 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY_SAMPLE			Shortcut between event READY and task SAMPLE																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
B	RW	READY_STOP			Shortcut between event READY and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
C	RW	DOWN_STOP			Shortcut between event DOWN and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
D	RW	UP_STOP			Shortcut between event UP and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
E	RW	CROSS_STOP			Shortcut between event CROSS and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														

8.12.3.16 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	READY			Enable or disable interrupt for event READY																													
			Disabled	0	Disable																													
			Enabled	1	Enable																													
B	RW	DOWN			Enable or disable interrupt for event DOWN																													
			Disabled	0	Disable																													
			Enabled	1	Enable																													
C	RW	UP			Enable or disable interrupt for event UP																													
			Disabled	0	Disable																													
			Enabled	1	Enable																													
D	RW	CROSS			Enable or disable interrupt for event CROSS																													

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
			Disabled	0				Disable																											
			Enabled	1				Enable																											

8.12.3.17 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Write '1' to enable interrupt for event READY																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	DOWN			Write '1' to enable interrupt for event DOWN																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	UP			Write '1' to enable interrupt for event UP																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	CROSS			Write '1' to enable interrupt for event CROSS																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.12.3.18 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Write '1' to disable interrupt for event READY																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	DOWN			Write '1' to disable interrupt for event DOWN																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	UP			Write '1' to disable interrupt for event UP																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
D	RW	CROSS	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event CROSS																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.12.3.19 INTPEND

Address offset: 0x30C

Pending interrupts

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	READY			Read pending status of interrupt for event READY																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
B	R	DOWN			Read pending status of interrupt for event DOWN																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
C	R	UP			Read pending status of interrupt for event UP																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
D	R	CROSS			Read pending status of interrupt for event CROSS																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														

8.12.3.20 RESULT

Address offset: 0x400

Compare result

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				A																																
Reset 0x00000000				0 0																																
ID	R/W	Field	Value ID	Value	Description																															
A	R	RESULT			Result of last compare. Decision point SAMPLE task.																															
			Below	0	Input voltage is below the reference threshold (VIN+ < VIN-)																															
			Above	1	Input voltage is above the reference threshold (VIN+ > VIN-)																															

8.12.3.21 ENABLE

Address offset: 0x500

Enable LPCOMP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE			Enable or disable LPCOMP																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.12.3.22 PSEL

Address offset: 0x504

Input pin select

The pin is selected based on PSEL.PORT

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0					
ID																													B	B	B	B					A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
ID	R/W	Field	Value ID	Value	Description																																				
A	RW	PIN			Analog pin select																																				
B	RW	PORT			GPIO Port selection																																				

8.12.3.23 REFSEL

Address offset: 0x508

Reference select

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000004				0 1 0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	REFSEL			Reference select																														
			Ref1_8Vdd	0	VDD * 1/8 selected as reference																														
			Ref2_8Vdd	1	VDD * 2/8 selected as reference																														
			Ref3_8Vdd	2	VDD * 3/8 selected as reference																														
			Ref4_8Vdd	3	VDD * 4/8 selected as reference																														
			Ref5_8Vdd	4	VDD * 5/8 selected as reference																														
			Ref6_8Vdd	5	VDD * 6/8 selected as reference																														
			Ref7_8Vdd	6	VDD * 7/8 selected as reference																														
			ARef	7	External analog reference selected																														
			Ref1_16Vdd	8	VDD * 1/16 selected as reference																														
			Ref3_16Vdd	9	VDD * 3/16 selected as reference																														
			Ref5_16Vdd	10	VDD * 5/16 selected as reference																														
			Ref7_16Vdd	11	VDD * 7/16 selected as reference																														
			Ref9_16Vdd	12	VDD * 9/16 selected as reference																														
			Ref11_16Vdd	13	VDD * 11/16 selected as reference																														
			Ref13_16Vdd	14	VDD * 13/16 selected as reference																														
			Ref15_16Vdd	15	VDD * 15/16 selected as reference																														

8.12.3.24 EXTREFSEL

Address offset: 0x50C

External reference select

8.12.3.25 ANADETECT

Analog detect configuration

8.12.3.26 HYST

Comparator hysteresis enable

8.13 NFCT — Near field communication tag

The main features of NFCT are:

- NFC-A listen mode operation:
 - 13.56 MHz input frequency
 - Bit rate 106 kbps
- Wake-on-field low power field detection (SENSE) mode
- Frame assemble and disassemble for the NFC-A frames specified by the NFC Forum
- Programmable frame timing controller
- Integrated automatic collision resolution, cyclic redundancy check (CRC), and parity functions

With appropriate software, the NFCT peripheral can be used as the listening device NFC-A as specified by the [NFC Forum](#).

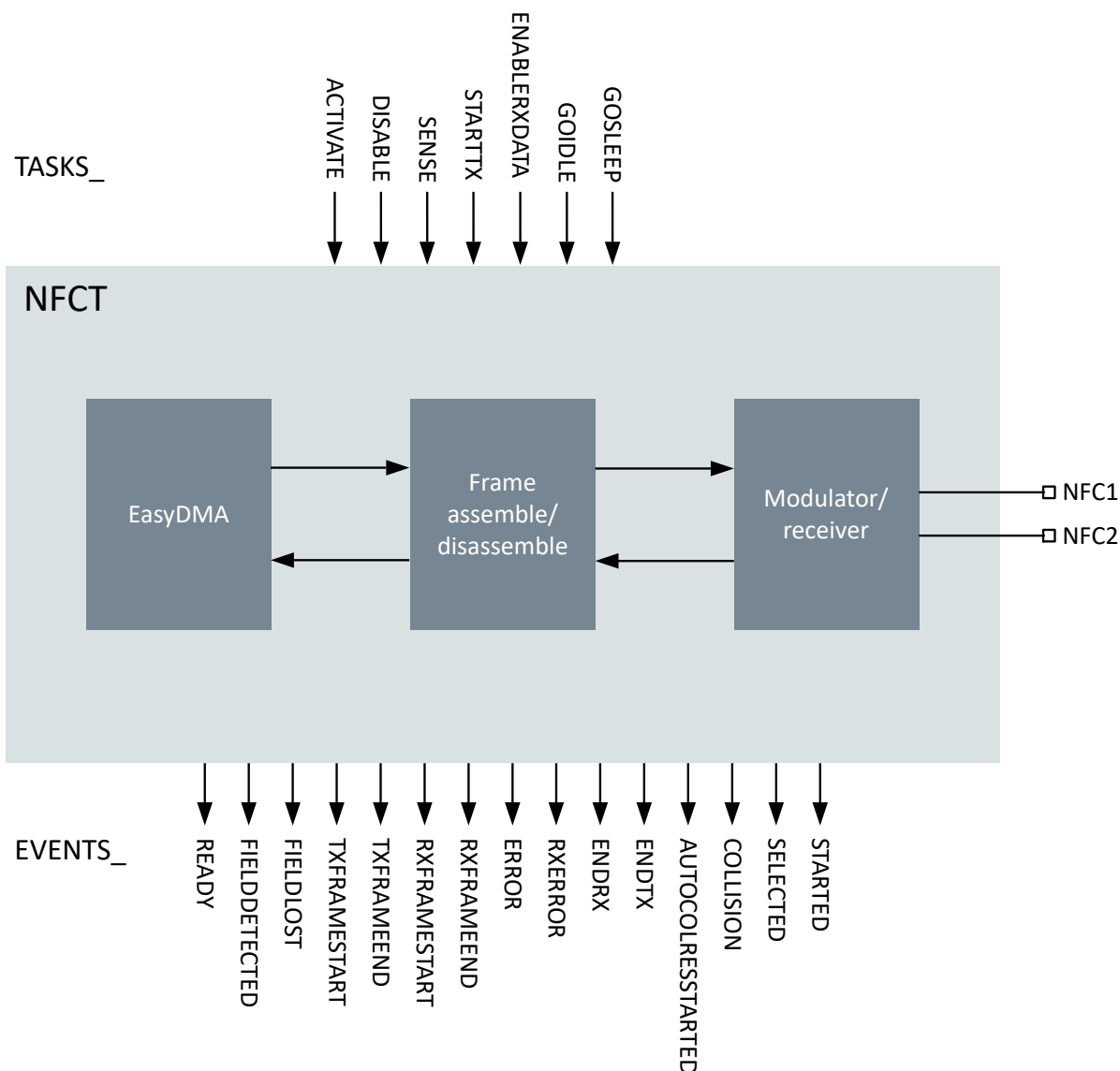


Figure 85: NFCT block diagram

8.13.1 Overview

The NFCT peripheral contains a 13.56 MHz AM receiver and a 13.56 MHz load modulator with 106 kbps data rate as defined by the NFC Forum.

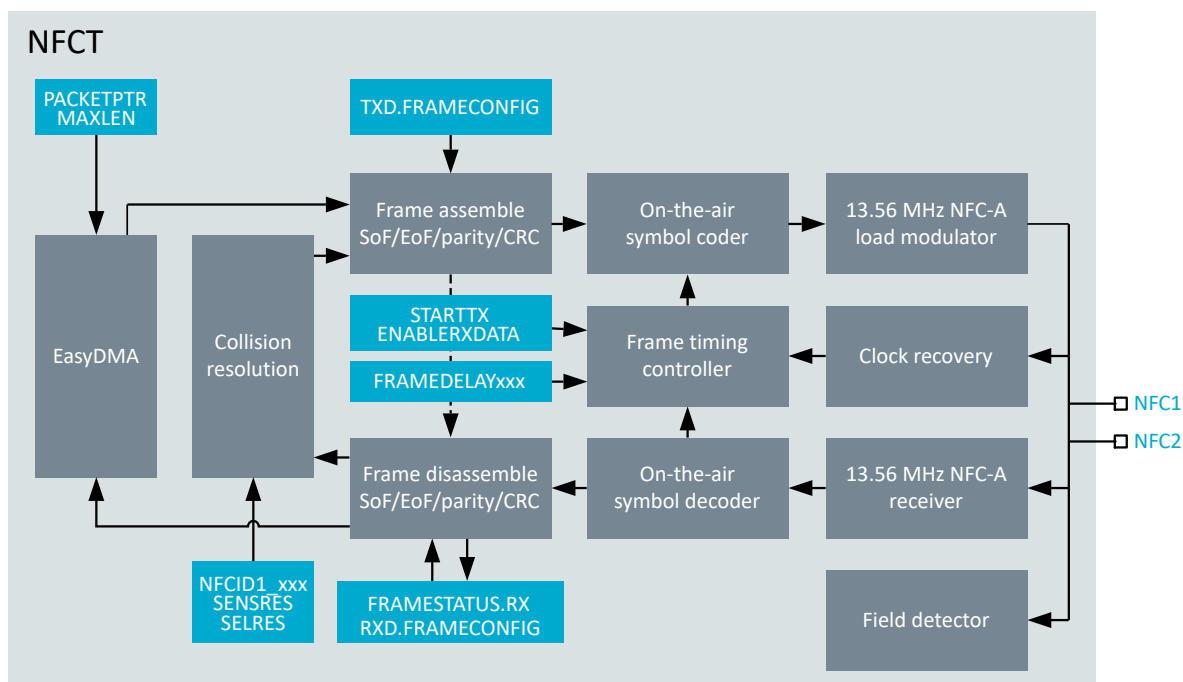


Figure 86: NFCT overview

When transmitting, the frame data will be transferred directly from RAM and transmitted with configurable frame type and delay timing. The system will be notified by an event whenever a complete frame is received or sent. The received frames will be automatically disassembled and the data part of the frame transferred to RAM.

The NFCT peripheral also supports the collision detection and resolution ("anticollision") as defined by the NFC Forum.

Wake-on-field is supported in SENSE mode while the device is either in System OFF or System ON mode. When the antenna enters an NFC field, an event will be triggered notifying the system to activate the NFCT functionality for incoming frames. In System ON, if the energy detected at the antenna increases beyond a threshold value, the module will generate a **FIELDDETECTED** event. When the strength of the field no longer supports NFC communication, the module will generate a **FIELDLOST** event. For the Low Power Field Detect threshold values, refer to **NFCT Electrical Specification** on page 819.

In System OFF, the NFCT Low Power Field Detect function can wake the system up through a reset. See **RESETREAS** on page 109 for more information on how to detect a wakeup from NFCT.

If the system is put into System OFF mode while a field is already present, the NFCT Low Power Field Detect function will wake the system up right away and generate a reset.

Note: As a consequence of a reset, NFCT is disabled, and therefore the reset handler will have to activate NFCT again and set it up properly.

The HFXO must be running before the NFCT peripheral goes into ACTIVATED state. Note that the NFCT peripheral calibration is automatically done on **ACTIVATE** task. The HFXO can be turned off when the NFCT peripheral goes into SENSE mode. The shortcut **FIELDDETECTED_ACTIVATE** can be used when the HFXO is already running while in SENSE mode.

Outgoing data will be collected from RAM with the EasyDMA function and assembled according to the **TXD.FRAMECONFIG** register. Incoming data will be disassembled according to the **RXD.FRAMECONFIG** register and the data section in the frame will be written to RAM via the EasyDMA function.

The NFCT peripheral includes a frame timing controller that can be used to accurately control the inter-frame delay between the incoming frame and a corresponding outgoing frame. It also includes optional CRC functionality.

8.13.2 Operating states

Tasks and events are used to control the operating state of the peripheral. The module can change state by triggering a task, or when specific operations are finalized. Events and tasks allow software to keep track of and change the current state.

See [NFCT block diagram](#) on page 351 and [NFCT state diagram, automatic collision resolution enabled](#) on page 353 for more information. See *NFC Forum, NFC Activity Technical Specification* for description on NFCT operating states.

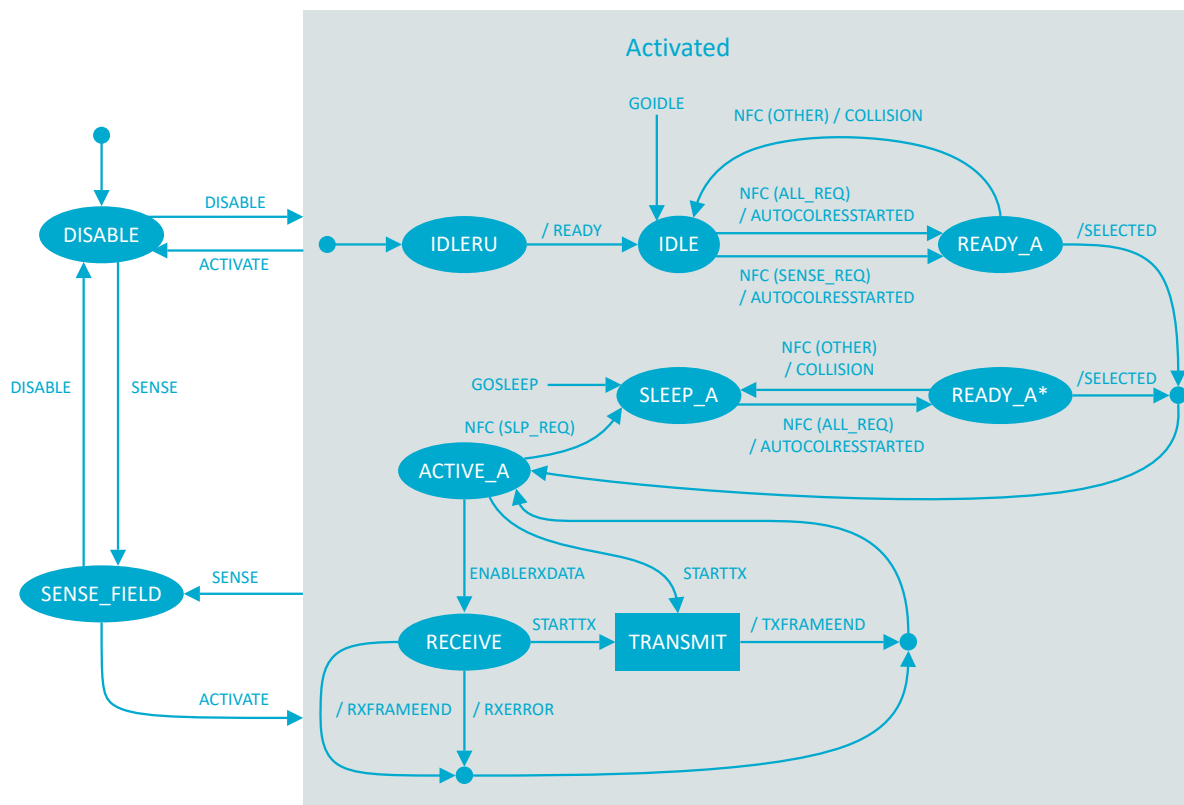


Figure 87: NFCT state diagram, automatic collision resolution enabled

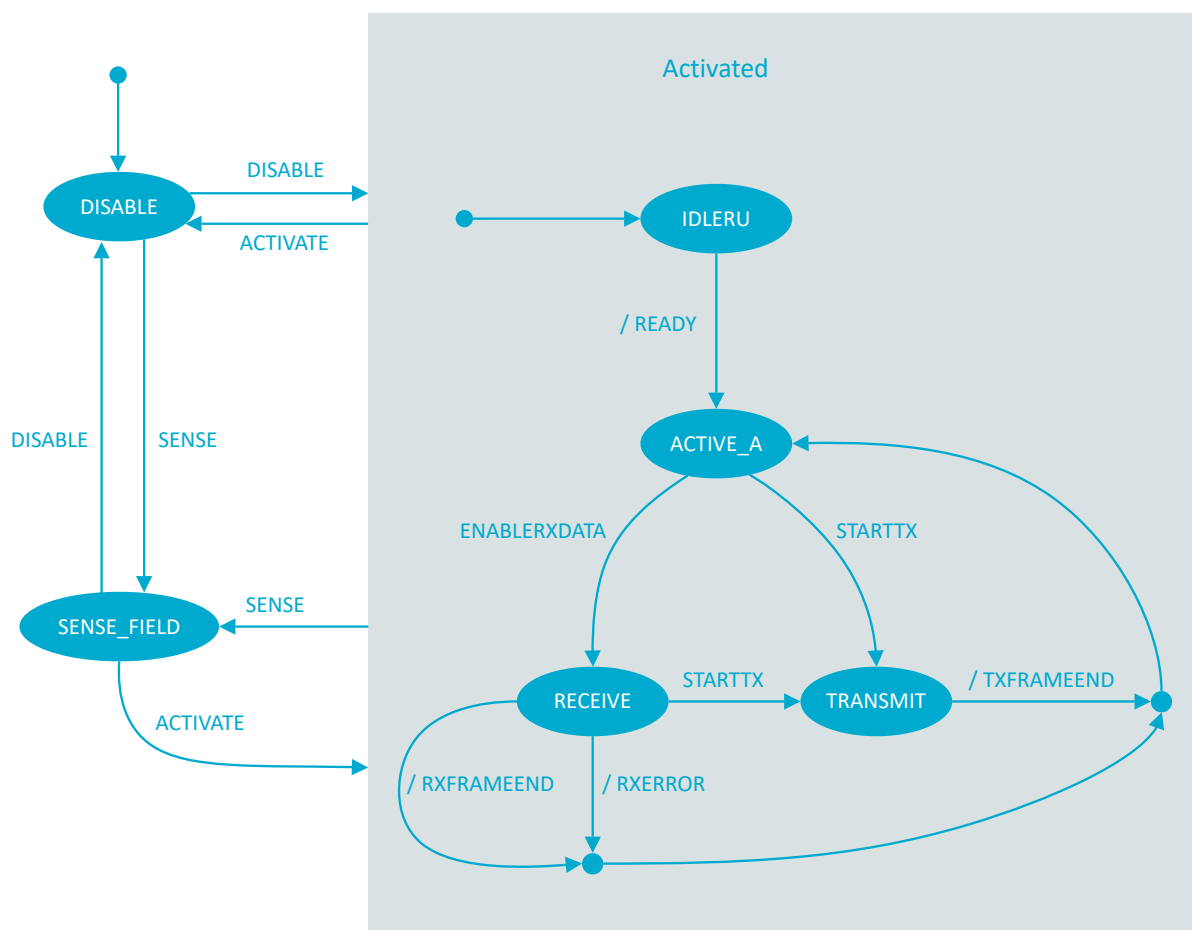


Figure 88: NFCT state diagram, automatic collision resolution disabled

Important:

- FIELDLOST event is not generated in SENSE mode.
- Sending SENSE task while field is still present does not generate **FIELDDETECTED** event.
- If the FIELDDETECTED event is cleared before sending the **ACTIVATE** task, then the FIELDDETECTED event shows up again after sending the ACTIVATE task. The shortcut FIELDDETECTED_ACTIVATE can be used to avoid this condition.

8.13.3 Pin configuration

NFCT uses two pins to connect the antenna and these pins are shared with GPIOs.

The ENABLE field in register **PADCONFIG** on page 388 defines the usage of these pins and their protection level against excessive voltages. See **Pin assignments** on page 793 for the pins used by the NFCT peripheral.

When **PADCONFIG.ENABLE=Enabled**, a protection circuit will be enabled on the dedicated pins, preventing the chip from being damaged in the presence of a strong NFC field. The protection circuit will short the two pins together if the voltage difference exceeds approximately 2V. The GPIO function on those pins will also be disabled.

When **PADCONFIG.ENABLE=Disabled**, the device will not be protected against strong NFC field damages caught by a connected NFCT antenna, and the NFCT peripheral will not operate as expected, as it will never leave the DISABLE state.

The pins dedicated to the NFCT antenna function will have some limitation when the pins are configured for normal GPIO operation. The pin capacitance will be higher on those pins (refer to C_{PAD_NFC} in the

Electrical Specification of [GPIO — General purpose input/output](#) on page 273), and some increased leakage current between the two pins is to be expected if they are used in GPIO mode, and are driven to different logical values. To save power, the two pins should always be set to the same logical value whenever entering one of the device power saving modes. For details, refer to I_{NFC_LEAK} in the Electrical Specification of [GPIO — General purpose input/output](#) on page 273.

8.13.4 EasyDMA

The NFCT peripheral implements EasyDMA for reading and writing of data packets from and to the Data RAM.

The NFCT EasyDMA utilizes a pointer called [PACKETPTR](#) on page 383 for receiving and transmitting packets.

The NFCT peripheral uses EasyDMA to read or write RAM, but not both at the same time. The event [RXFRAMESTART](#) indicates that the EasyDMA has started writing to the RAM for a receive frame and the event [RXFRAMEEND](#) indicates that the EasyDMA has completed writing to the RAM. Similarly, the event [TXFRAMESTART](#) indicates that the EasyDMA has started reading from the RAM for a transmit frame and the event [TXFRAMEEND](#) indicates that the EasyDMA has completed reading from the RAM. If a transmit and a receive operation is issued at the same time, the transmit operation would be prioritized.

Starting a transmit operation while the EasyDMA is writing a receive frame to the RAM will result in unpredictable behavior. Starting an EasyDMA operation when there is an ongoing EasyDMA operation may result in unpredictable behavior. It is recommended to wait for the [TXFRAMEEND](#) or [RXFRAMEEND](#) event for the ongoing transmit or receive before starting a new receive or transmit operation.

The [MAXLEN](#) on page 383 register determines the maximum number of bytes that can be read from or written to the RAM. This feature can be used to ensure that the NFCT peripheral does not overwrite, or read beyond, the RAM assigned to a packet. Note that if the [RXD.AMOUNT](#) or [TXD.AMOUNT](#) register indicates longer data packets than set in MAXLEN, the frames sent to or received from the physical layer will be incomplete. If that situation occurs in RX mode, the [OVERRUN](#) bit in the [FRAMESTATUS.RX](#) register will be set and an [RXERROR](#) event will be triggered.

Important: The [RXD.AMOUNT](#) and [TXD.AMOUNT](#) define a frame length in bytes and bits excluding start of frame (SoF), end of frame (EoF), and parity, but including CRC for [RXD.AMOUNT](#) only. Make sure to take potential additional bits into account when setting MAXLEN.

Only sending task [ENABLERXDATA](#) ensures that a new value in [PACKETPTR](#) pointing to the RX buffer in Data RAM is taken into account.

If [PACKETPTR](#) is not pointing to the Data RAM region, an EasyDMA transfer may result in a hard fault or RAM corruption. For more information about the different memory regions, see Chapter [Memory](#) on page 19.

The NFCT peripherals normally do alternative receive and transmit frames. Therefore, to prepare for the next frame, the [PACKETPTR](#), [MAXLEN](#), [TXD.FRAMECONFIG](#) and [TXD.AMOUNT](#) can be updated while the receive is in progress, and, similarly, the [PACKETPTR](#), [MAXLEN](#) and [RXD.FRAMECONFIG](#) can be updated while the transmit is in progress. They can be updated and prepared for the next NFC frame immediately after the [STARTED](#) event of the current frame has been received. Updating the [TXD.FRAMECONFIG](#) and [TXD.AMOUNT](#) during the current transmit frame or updating [RXD.FRAMECONFIG](#) during current receive frame may cause unpredictable behaviour.

In accordance with *NFC Forum, NFC Digital Protocol Technical Specification*, the least significant bit (LSB) from the least significant byte (LSByte) is sent on air first. The bytes are stored in increasing order, starting at the lowest address in the EasyDMA buffer in RAM.

8.13.5 Frame assembler

The NFCT peripheral implements a frame assembler in hardware.

When the NFCT peripheral is in the ACTIVE_A state, the software can decide to enter RX or TX mode. For RX mode, see [Frame disassembler](#) on page 357. For TX mode, the software must indicate the address of the source buffer in Data RAM and its size through programming the [PACKETPTR](#) and MAXLEN registers respectively, then issuing a STARTTX task.

MAXLEN must be set so that it matches the size of the frame to be sent.

The [STARTED](#) event indicates that the PACKETPTR and MAXLEN registers have been captured by the frame assembler EasyDMA.

When asserting the [STARTTX](#) task, the frame assembler module will start reading TXD.AMOUNT.TXDATABYTES bytes (plus one additional byte if TXD.AMOUNT.TXDATABITS > 0) from the RAM position set by the PACKETPTR.

The NFCT peripheral transmits the data as read from RAM, adding framing and the CRC calculated on the fly if set in TXD.FRAMECONFIG. The NFCT peripheral will take $(8 * \text{TXD.AMOUNT.TXDATABYTES} + \text{TXD.AMOUNT.TXDATABITS})$ bits and assemble a frame according to the settings in [TXD.FRAMECONFIG](#). Both short frames, standard frames, and bit-oriented SDD frames as specified in the *NFC Forum, NFC Digital Protocol Technical Specification* can be assembled by the correct setting of the TXD.FRAMECONFIG register.

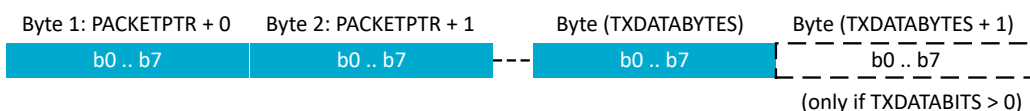
The bytes will be transmitted on air in the same order as they are read from RAM with a rising bit order within each byte, least significant bit (LSB) first. That is, the least significant bit (b0) will be transmitted on air before the second bit (b1), and so on. The bits read from RAM will be coded into symbols as defined in the *NFC Forum, NFC Digital Protocol Technical Specification*.

Note: Some NFC Forum documents, such as *NFC Forum, NFC Digital Protocol Technical Specification*, define bit numbering in a byte from b1 (LSB) to b8 (most significant bit (MSB)), while most other technical documents from the NFC Forum, and also the Nordic Semiconductor documentation, traditionally number them from b0 to b7. The present document uses the b0–b7 numbering scheme. Be aware of this when comparing the *NFC Forum, NFC Digital Protocol Technical Specification* to others.

The frame assembler can be configured in TXD.FRAMECONFIG to add SoF symbol, calculate and add parity bits, and calculate and add CRC to the data read from RAM when assembling the frame. The total frame will then be longer than what is defined by TXD.AMOUNT.TXDATABYTES. TXDATABITS. DISCARDMODE will select if the first bits in the first byte read from RAM or the last bits in the last byte read from RAM will be discarded if TXD.AMOUNT.TXDATABITS are not equal to zero. Note that if TXD.FRAMECONFIG.PARITY = Parity and TXD.FRAMECONFIG.DISCARDMODE=DiscardStart, a parity bit will be included after the non-complete first byte. No parity will be added after a non-complete last byte.

The frame assemble operation for different settings in TXD.FRAMECONFIG is illustrated in the following table. All shaded bit fields are added by the frame assembler. Some of these bits are optional and appearances are configured in TXD.FRAMECONFIG. Note that the frames illustrated do not necessarily comply with the NFC specification. The figure only illustrates the behavior of the NFCT peripheral.

Data from RAM

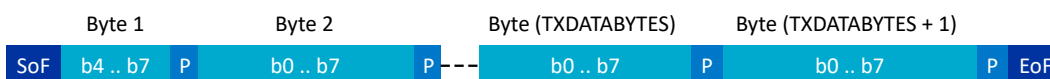


Frame on air

PARITY = Parity
TXDATABITS = 0
CRCMODETX = CRC16TX



PARITY = Parity
TXDATABITS = 4
CRCMODETX = NoCRCTX
DISCARDMODE = DiscardStart



PARITY = Parity
TXDATABITS = 0
CRCMODETX = NoCRCTX



Figure 89: Frame assemble illustration

The accurate timing for transmitting the frame on air is set using the frame timing controller settings.

8.13.6 Frame disassembler

The NFCT peripheral implements a frame disassembler in hardware.

When the NFCT peripheral is in the `ACTIVE_A` state, the software can decide to enter RX or TX mode. For TX mode, see [Frame assembler](#) on page 355. For RX mode, the software must indicate the address and size of the destination buffer in Data RAM through programming the `PACKETPTR` and `MAXLEN` registers before issuing an `ENABLERXDATA` task.

The `STARTED` event indicates that the `PACKETPTR` and `MAXLEN` registers have been captured by the frame disassembler EasyDMA.

When an incoming frame starts, the `RXFRAMESTART` event will get issued and data will be written to the buffer in Data RAM. The frame disassembler will verify and remove any parity bits, start of frame (SoF) and end of frame (EoF) symbols on the fly based on `RXD.FRAMECONFIG` register configuration. It will, however, verify and transfer the CRC bytes into RAM, if the CRC is enabled through `RXD.FRAMECONFIG`.

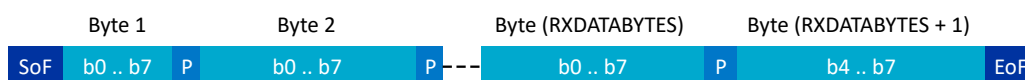
When an EoF symbol is detected, the NFCT peripheral will assert the `RXFRAMEEND` event and write the `RXD.AMOUNT` register to indicate numbers of received bytes and bits in the data packet. The module does not interpret the content of the data received from the remote NFC device, except for SoF, EoF, parity, and CRC checking, as described above. The frame disassemble operation is illustrated in the following figure.

Frame on air

PARITY = Parity
 RXDATABITS = 0
 CRCMODERX = CRC16RX



PARITY = Parity
 CRCMODERX = NoCRCTR
 RXDATABITS = 4



PARITY = NoParity
 CRCMODERX = NoCRCRX
 RXDATABITS = 0



Data to RAM

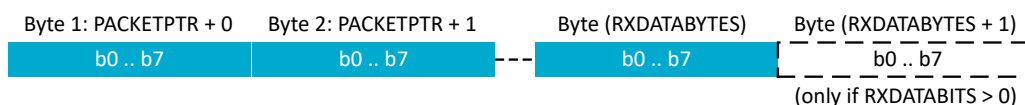


Figure 90: Frame disassemble illustration

Per NFC specification, the time between EoF to the next SoF can be as short as 86 μ s, and therefore care must be taken that PACKETPTR and MAXLEN are ready and ENABLERXDATA is issued on time after the end of previous frame. The use of a PPI shortcut from **TXFRAMEEND** to ENABLERXDATA is recommended.

8.13.7 Frame timing controller

The NFCT peripheral includes a frame timing controller that continuously keeps track of the number of the 13.56 MHz RF carrier clock periods since the end of the EoF of the last received frame.

The NFCT peripheral can be programmed to send a responding frame within a time window or at an exact count of RF carrier periods. In case of **FRAMEDELAYMODE** = Window, a **STARTTX** task triggered before the frame timing controller counter is equal to **FRAMEDELAYMIN** will force the transmission to halt until the counter is equal to **FRAMEDELAYMIN**. If the counter is within **FRAMEDELAYMIN** and **FRAMEDELAYMAX** when the **STARTTX** task is triggered, the NFCT peripheral will start the transmission straight away. In case of **FRAMEDELAYMODE** = ExactVal, a **STARTTX** task triggered before the frame delay counter is equal to **FRAMEDELAYMAX** will halt the actual transmission start until the counter is equal to **FRAMEDELAYMAX**.

In case of **FRAMEDELAYMODE** = WindowGrid, the behaviour is similar to the **FRAMEDELAYMODE** = Window, but the actual transmission between **FRAMEDELAYMIN** and **FRAMEDELAYMAX** starts on a bit grid as defined for NFC-A Listen frames (slot duration of 128 RF carrier periods).

An **ERROR** event (with **FRAMEDElayTIMEOUT** cause in **ERRORSTATUS**) will be asserted if the frame timing controller counter reaches **FRAMEDELAYMAX** without any **STARTTX** task triggered. This may happen even when the response is not required as per *NFC Forum, NFC Digital Protocol Technical Specification*. Any commands handled by the automatic collision resolution that don't involve a response being

generated may also result in an ERROR event (with FRAMEDELAYTIMEOUT cause in ERRORSTATUS). The FRAMEDELAYMIN and FRAMEDELAYMAX values shall only be updated before the STARTTX task is triggered. Failing to do so may cause unpredictable behaviour.

The frame timing controller operation is illustrated in the following figure. The frame timing controller automatically adjusts the frame timing counter based on the last received data bit according to NFC-A technology in the *NFC Forum, NFC Digital Protocol Technical Specification*.

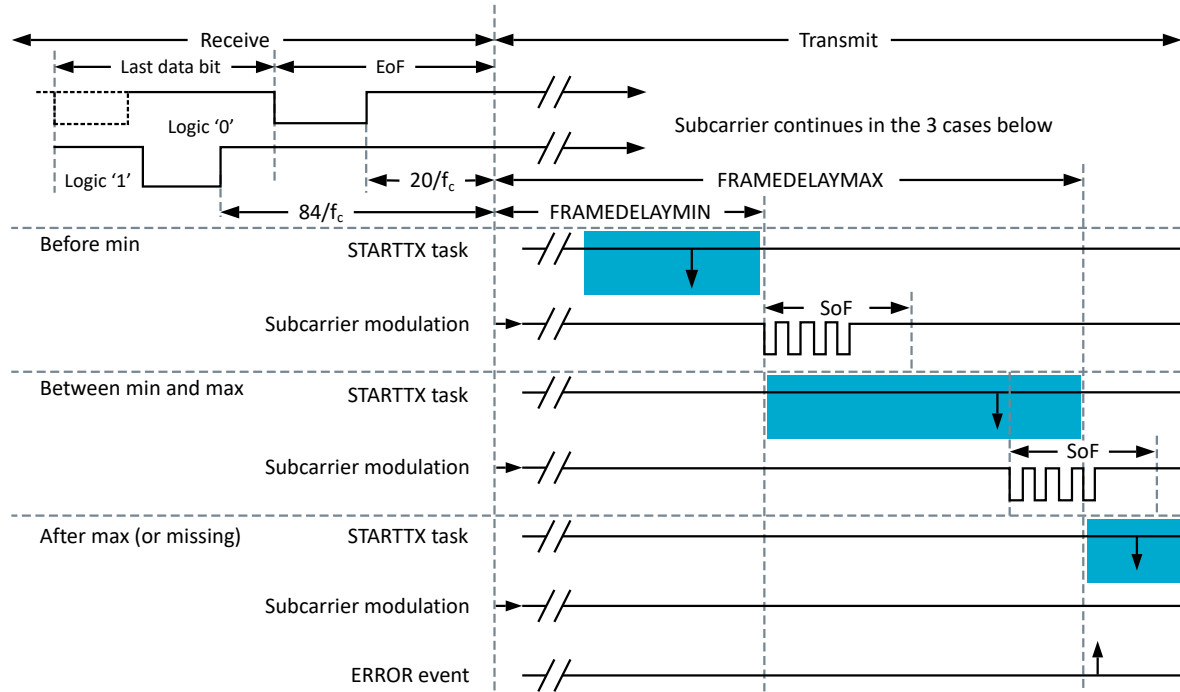


Figure 91: Frame timing controller (FRAMEDELAYMODE=Window)

8.13.8 Collision resolution

The NFCT peripheral implements an automatic collision resolution function as defined by the NFC Forum.

Automatic collision resolution is enabled by default, and it is recommended that the feature is used since it is power efficient and reduces the complexity of software handling the collision resolution sequence. This feature can be disabled through the MODE field in the [AUTOCOLRESCONFIG](#) register. When the automatic collision resolution is disabled, all commands will be sent over EasyDMA as defined in frame disassembler.

The [SENSRES](#) and [SELRES](#) registers need to be programmed upfront in order for the collision resolution to behave correctly. Depending on the NFCIDSIZE field in SENSRES, the following registers also need to be programmed upfront:

- NFCID1_LAST if NFCID1SIZE=NFCID1Single (ID = 4 bytes);
- NFCID1_2ND_LAST and NFCID1_LAST if NFCID1SIZE=NFCID1Double (ID = 7 bytes);
- NFCID1_3RD_LAST, NFCID1_2ND_LAST and NFCID1_LAST if NFCID1SIZE=NFCID1Triple (ID = 10 bytes);

A pre-defined set of registers, NFC.TAGHEADER0..3, containing a valid NFCID1 value, is available in FICR and can be used by software to populate the NFCID1_3RD_LAST, NFCID1_2ND_LAST, and NFCID1_LAST registers.

[NFCID1 byte allocation \(top sent first on air\)](#) on page 360 explains the position of the ID bytes in NFCID1_3RD_LAST, NFCID1_2ND_LAST, and NFCID1_LAST, depending on the ID size, and as compared to the definition used in the *NFC Forum, NFC Digital Protocol Technical Specification*.

	ID = 4 bytes	ID = 7 bytes	ID = 10 bytes
NFCID1.Q			nfcid1 ₀
NFCID1.R			nfcid1 ₁
NFCID1.S			nfcid1 ₂
NFCID1.T		nfcid1 ₀	nfcid1 ₃
NFCID1.U		nfcid1 ₁	nfcid1 ₄
NFCID1.V		nfcid1 ₂	nfcid1 ₅
NFCID1.W	nfcid1 ₀	nfcid1 ₃	nfcid1 ₆
NFCID1.X	nfcid1 ₁	nfcid1 ₄	nfcid1 ₇
NFCID1.Y	nfcid1 ₂	nfcid1 ₅	nfcid1 ₈
NFCID1.Z	nfcid1 ₃	nfcid1 ₆	nfcid1 ₉

Table 41: NFCID1 byte allocation (top sent first on air)

The hardware implementation can handle the states from IDLE to ACTIVE_A automatically as defined in the *NFC Forum, NFC Activity Technical Specification*, and the other states are to be handled by software. The software keeps track of the state through events. The collision resolution will trigger an **AUTOCOLRESSTARTED** event when it has started. Reaching the ACTIVE_A state is indicated by the **SELECTED** event.

If collision resolution fails, a **COLLISION** event is triggered. Note that errors occurring during automatic collision resolution may also cause **ERROR** and/or **RXERROR** events to be generated. Other events may also get generated. It is recommended that the software ignores any event except COLLISION, SELECTED and FIELDLOST during automatic collision resolution. Software shall also make sure that any unwanted SHORT or PPI shortcut is disabled during automatic collision resolution.

The automatic collision resolution will be restarted, if the packets are received with CRC or parity errors while in ACTIVE_A state. The automatic collision resolution feature can be disabled while in ACTIVE_A state to avoid this.

The SLP_REQ is automatically handled by the NFCT peripheral when the automatic collision resolution is enabled. However, this results in an ERROR event (with FRAMEDELAYTIMEOUT cause in ERRORSTATUS) since the SLP_REQ has no response. This error must be ignored until the SELECTED event is triggered and this error should be cleared by the software when the SELECTED event is triggered.

8.13.9 Antenna interface

In ACTIVATED state, an amplitude regulator will adjust the voltage swing on the antenna pins to a value that is within the V_{swing} limit.

Refer to [NFCT Electrical Specification](#) on page 819.

8.13.10 NFCT antenna recommendations

The NFCT antenna coil must be connected differential between NFC1 and NFC2 pins of the device.

Two external capacitors should be used to tune the resonance of the antenna circuit to 13.56 MHz.

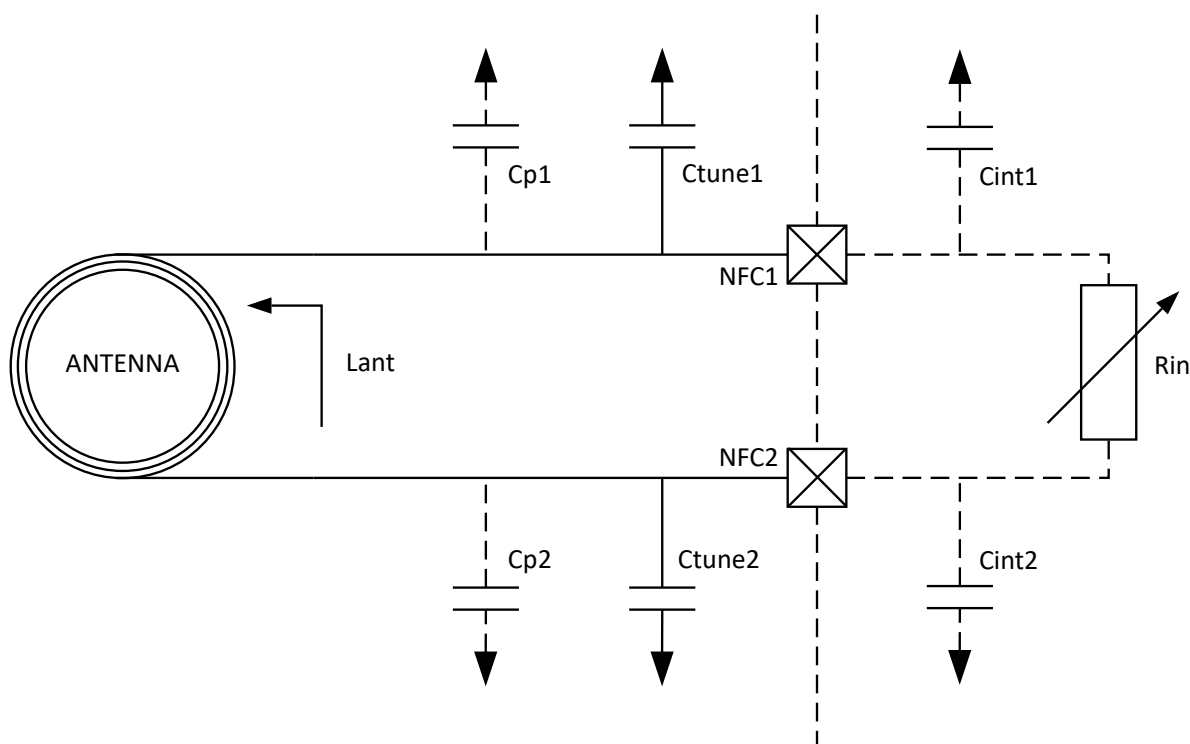


Figure 92: NFCT antenna recommendations

The required tuning capacitor value is given by the below equations:

$$C'_{tune} = \frac{1}{(2\pi \cdot 13.56 \text{ MHz})^2 \cdot L_{ant}} \quad \text{where } C'_{tune} = \frac{1}{2} \cdot (C_p + C_{int} + C_{tune})$$

$$\text{and } C_{tune1} = C_{tune2} = C_{tune} \quad C_{p1} = C_{p2} = C_p \quad C_{int1} = C_{int2} = C_{int}$$

$$C_{tune} = \frac{2}{(2\pi \cdot 13.56 \text{ MHz})^2 \cdot L_{ant}} - C_p - C_{int}$$

An antenna inductance of $L_{ant} = 2 \mu\text{H}$ will give tuning capacitors in the range of 130 pF on each pin. The total capacitance on **NFC1** and **NFC2** must be matched.

8.13.11 Battery protection

If the antenna is exposed to a strong NFC field, current may flow in the opposite direction on the supply due to parasitic diodes and ESD structures.

If the battery used does not tolerate return current, a series diode must be placed between the battery and the device in order to protect the battery.

8.13.12 Digital Modulation Signal

Support for external analog frontends or antenna architectures is possible by optionally outputting the digital modulation signal to a GPIO.

The NFCT peripheral is designed to connect directly to a loop antenna, receive a modulated signal from an NFC Reader with its internal analog frontend and transmit data back by changing the input resistance that is then seen as modulated load by the NFC Reader.

In addition, the peripheral has an option to output the digital modulation signal to a GPIO. Reception still occurs through the internal analog frontend, whereas transmission can be done by one of the following:

- The internal analog frontend through the loop antenna (default)
- An external frontend using the digital modulation signal
- The combination of both above

There are two registers that allow configuration of the modulation signal (i.e. of the response from NFCT to the NFC Reader), [MODULATIONCTRL](#) and [MODULATIONPSEL](#). The registers need to be programmed before NFCT sends a response to a request from a reader. Ideally, this configuration is performed during startup and whenever the NFCT peripheral is powered up.

The selected GPIO needs to be configured as output in the corresponding GPIO configuration register. It is recommended to set an output value in the corresponding GPIO.OUT register – this value will be driven whenever the NFCT peripheral is disabled.

NFCT drives the pin low when there is no modulation, and drives it with On-Off Keying (OOK) modulation of an 847 kHz subcarrier (derived from the carrier frequency) when it responds to commands from an NFC Reader.

8.13.13 References

NFC Forum, NFC Analog Specification version 2.1, www.nfc-forum.org

NFC Forum, NFC Digital Protocol Technical Specification version 2.2, www.nfc-forum.org

NFC Forum, NFC Activity Technical Specification version 2.1, www.nfc-forum.org

8.13.14 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
NFCT : S	GLOBAL	0x500D6000	US	S	SA	No	Near field communication tag NFCT
NFCT : NS		0x400D6000					

Configuration

Instance	Domain	Configuration
NFCT : S	GLOBAL	Reset value of register NFCTFIELDDETCFG: 1
NFCT : NS		

Register overview

Register	Offset	TZ	Description
TASKS_ACTIVATE	0x000		Activate NFCT peripheral for incoming and outgoing frames, change state to activated
TASKS_DISABLE	0x004		Disable NFCT peripheral
TASKS_SENSE	0x008		Enable NFC sense field mode, change state to sense mode
TASKS_STARTTX	0x00C		Start transmission of an outgoing frame, change state to transmit
TASKS_STOPTX	0x010		Stops an issued transmission of a frame
TASKS_ENABLERXDATA	0x01C		Initializes the EasyDMA for receive.
TASKS_GOIDLE	0x024		Force state machine to IDLE state
TASKS_GOSLEEP	0x028		Force state machine to SLEEP_A state

Register	Offset	TZ	Description
SUBSCRIBE_ACTIVATE	0x080		Subscribe configuration for task ACTIVATE
SUBSCRIBE_DISABLE	0x084		Subscribe configuration for task DISABLE
SUBSCRIBE_SENSE	0x088		Subscribe configuration for task SENSE
SUBSCRIBE_STARTTX	0x08C		Subscribe configuration for task STARTTX
SUBSCRIBE_STOPTX	0x090		Subscribe configuration for task STOPTX
SUBSCRIBE_ENABLERXDATA	0x09C		Subscribe configuration for task ENABLERXDATA
SUBSCRIBE_GOIDLE	0x0A4		Subscribe configuration for task GOIDLE
SUBSCRIBE_GOSLEEP	0x0A8		Subscribe configuration for task GOSLEEP
EVENTS_READY	0x100		The NFCT peripheral is ready to receive and send frames
EVENTS_FIELDDETECTED	0x104		Remote NFC field detected
EVENTS_FIELDLOST	0x108		Remote NFC field lost
EVENTS_TXFRAMESTART	0x10C		Marks the start of the first symbol of a transmitted frame
EVENTS_TXFRAMEEND	0x110		Marks the end of the last transmitted on-air symbol of a frame
EVENTS_RXFRAMESTART	0x114		Marks the end of the first symbol of a received frame
EVENTS_RXFRAMEEND	0x118		Received data has been checked (CRC, parity) and transferred to RAM, and EasyDMA has ended accessing the RX buffer
EVENTS_ERROR	0x11C		NFC error reported. The ERRORSTATUS register contains details on the source of the error.
EVENTS_RXERROR	0x128		NFC RX frame error reported. The FRAMESTATUS.RX register contains details on the source of the error.
EVENTS_ENDRX	0x12C		RX buffer (as defined by PACKETPTR and MAXLEN) in Data RAM full.
EVENTS_ENDTX	0x130		Transmission of data in RAM has ended, and EasyDMA has ended accessing the TX buffer
EVENTS_AUTOCOLRESSTARTED	0x138		Auto collision resolution process has started
EVENTS_COLLISION	0x148		NFC auto collision resolution error reported.
EVENTS_SELECTED	0x14C		NFC auto collision resolution successfully completed
EVENTS_STARTED	0x150		EasyDMA is ready to receive or send frames.
PUBLISH_READY	0x180		Publish configuration for event READY
PUBLISH_FIELDDETECTED	0x184		Publish configuration for event FIELDDETECTED
PUBLISH_FIELDLOST	0x188		Publish configuration for event FIELDLOST
PUBLISH_TXFRAMESTART	0x18C		Publish configuration for event TXFRAMESTART
PUBLISH_TXFRAMEEND	0x190		Publish configuration for event TXFRAMEEND
PUBLISH_RXFRAMESTART	0x194		Publish configuration for event RXFRAMESTART
PUBLISH_RXFRAMEEND	0x198		Publish configuration for event RXFRAMEEND
PUBLISH_ERROR	0x19C		Publish configuration for event ERROR
PUBLISH_RXERROR	0x1A8		Publish configuration for event RXERROR
PUBLISH_ENDRX	0x1AC		Publish configuration for event ENDRX
PUBLISH_ENDTX	0x1B0		Publish configuration for event ENDTX
PUBLISH_AUTOCOLRESSTARTED	0x1B8		Publish configuration for event AUTOCOLRESSTARTED
PUBLISH_COLLISION	0x1C8		Publish configuration for event COLLISION
PUBLISH_SELECTED	0x1CC		Publish configuration for event SELECTED
PUBLISH_STARTED	0x1D0		Publish configuration for event STARTED
SHORTS	0x200		Shortcuts between local events and tasks
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
ERRORSTATUS	0x404		NFC Error Status register
FRAMESTATUS.RX	0x40C		Result of last incoming frame
NFCTAGSTATE	0x410		Current operating state of NFC tag
SLEEPSTATE	0x420		Sleep state during automatic collision resolution
FIELDPRESENT	0x43C		Indicates the presence or not of a valid field
FRAMEDELAYMIN	0x504		Minimum frame delay
FRAMEDELAYMAX	0x508		Maximum frame delay
FRAMEDELAYMODE	0x50C		Configuration register for the Frame Delay Timer
PACKETPTR	0x510		Packet pointer for TXD and RXD data storage in Data RAM

Register	Offset	TZ	Description
MAXLEN	0x514		Size of the RAM buffer allocated to TXD and RXD data storage each
TXD.FRAMECONFIG	0x518		Configuration of outgoing frames
TXD.AMOUNT	0x51C		Size of outgoing frame
RXD.FRAMECONFIG	0x520		Configuration of incoming frames
RXD.AMOUNT	0x524		Size of last incoming frame
MODULATIONCTRL	0x52C		Enables the modulation output to a GPIO pin which can be connected to a second external antenna.
MODULATIONPSEL	0x538		Pin select for Modulation control
MODE	0x550		Configure EasyDMA mode
NFCID1.LAST	0x590		Last NFCID1 part (4, 7 or 10 bytes ID)
NFCID1.SECONDLAST	0x594		Second last NFCID1 part (7 or 10 bytes ID)
NFCID1.THIRDLAST	0x598		Third last NFCID1 part (10 bytes ID)
AUTOCOLRESCONFIG	0x59C		Controls the auto collision resolution function. This setting must be done before the NFCT peripheral is activated.
SENSRES	0x5A0		NFC-A SENS_RES auto-response settings
SELRES	0x5A4		NFC-A SEL_RES auto-response settings
PADCONFIG	0x6D4		NFC pad configuration

8.13.14.1 TASKS_ACTIVATE

Address offset: 0x000

Activate NFCT peripheral for incoming and outgoing frames, change state to activated

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_ACTIVATE						Activate NFCT peripheral for incoming and outgoing frames, change state to activated																											
			Trigger	1				Trigger task																											

8.13.14.2 TASKS_DISABLE

Address offset: 0x004

Disable NFCT peripheral

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_DISABLE						Disable NFCT peripheral																											
			Trigger	1				Trigger task																											

8.13.14.3 TASKS_SENSE

Address offset: 0x008

Enable NFC sense field mode, change state to sense mode

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_SENSE						Enable NFC sense field mode, change state to sense mode																											
			Trigger	1				Trigger task																											

8.13.14.4 TASKS_STARTTX

Address offset: 0x00C

Start transmission of an outgoing frame, change state to transmit

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																							A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value	Description																																		
A	W	TASKS_STARTTX			Start transmission of an outgoing frame, change state to transmit																																		
			Trigger	1	Trigger task																																		

8.13.14.5 TASKS_STOPTX

Address offset: 0x010

Stops an issued transmission of a frame

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	TASKS_STOPTX			Stops an issued transmission of a frame																															
			Trigger	1	Trigger task																															

8.13.14.6 TASKS_ENABLERXDATA

Address offset: 0x01C

Initializes the EasyDMA for receive.

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																								A			
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																						
A	W	TASKS_ENABLERXDATA			Initializes the EasyDMA for receive.																																						
			Trigger	1	Trigger task																																						

8.13.14.7 TASKS_GOIDLE

Address offset: 0x024

Force state machine to IDLE state

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_GOIDLE						Force state machine to IDLE state																											
			Trigger	1				Trigger task																											

8.13.14.8 TASKS_GOSLEEP

Address offset: 0x028

Force state machine to SLEEP_A state

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_GOSLEEP						Force state machine to SLEEP_A state																											
			Trigger	1				Trigger task																											

8.13.14.9 SUBSCRIBE_ACTIVATE

Address offset: 0x080

Subscribe configuration for task [ACTIVATE](#)

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID					B																								A					A	A	A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
ID	R/W	Field	Value ID	Value	Description																																			
A	RW	CHIDX		[0..255]	DPPI channel that task ACTIVATE will subscribe to																																			
B	RW	EN																																						
			Disabled	0	Disable subscription																																			
			Enabled	1	Enable subscription																																			

8.13.14.10 SUBSCRIBE_DISABLE

Address offset: 0x084

Subscribe configuration for task [DISABLE](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value				Description																														
A	RW	CHIDX		[0..255]				DPPI channel that task DISABLE will subscribe to																														
B	RW	EN																																				
			Disabled	0				Disable subscription																														
			Enabled	1				Enable subscription																														

8.13.14.11 SUBSCRIBE_SENSE

Address offset: 0x088

Subscribe configuration for task [SENSE](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task SENSE will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.13.14.12 SUBSCRIBE_STARTTX

Address offset: 0x08C

Subscribe configuration for task **STARTTX**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task STARTTX will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.13.14.13 SUBSCRIBE_STOPTX

Address offset: 0x090

Subscribe configuration for task **STOPTX**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task STOPTX will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.13.14.14 SUBSCRIBE_ENABLERXDATA

Address offset: 0x09C

Subscribe configuration for task **ENABLERXDATA**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task ENABLERXDATA will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.13.14.15 SUBSCRIBE_GOIDLE

Address offset: 0x0A4

Subscribe configuration for task **GOIDLE**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task GOIDLE will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.13.14.16 SUBSCRIBE_GOSLEEP

Address offset: 0x0A8

Subscribe configuration for task **GOSLEEP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task GOSLEEP will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.13.14.17 EVENTS_READY

Address offset: 0x100

The NFCT peripheral is ready to receive and send frames

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_READY			The NFCT peripheral is ready to receive and send frames																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.13.14.18 EVENTS_FIELDDETECTED

Address offset: 0x104

Remote NFC field detected

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	EVENTS_FIELDDETECTED				Remote NFC field detected																													
			NotGenerated	0		Event not generated																													
			Generated	1		Event generated																													

8.13.14.19 EVENTS_FIELDLOST

Address offset: 0x108

Remote NFC field lost

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_FIELDLOST			Remote NFC field lost																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.13.14.20 EVENTS_TXFRAMESTART

Address offset: 0x10C

Marks the start of the first symbol of a transmitted frame

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_TXFRAMESTART			Marks the start of the first symbol of a transmitted frame																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.13.14.21 EVENTS_TXFRAMEEND

Address offset: 0x110

Marks the end of the last transmitted on-air symbol of a frame

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																				A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value		Description																														
A	RW	EVENTS_TXFRAMEEND				Marks the end of the last transmitted on-air symbol of a frame																														
			NotGenerated	0	Event not generated																															
			Generated	1	Event generated																															

8.13.14.22 EVENTS_RXFRAMESTART

Address offset: 0x114

Marks the end of the first symbol of a received frame

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_RXFRAMESTART			Marks the end of the first symbol of a received frame																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.13.14.23 EVENTS_RXFRAMEEND

Address offset: 0x118

Received data has been checked (CRC, parity) and transferred to RAM, and EasyDMA has ended accessing the RX buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_RXFRAMEEND			Received data has been checked (CRC, parity) and transferred to RAM, and EasyDMA has ended accessing the RX buffer																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.13.14.24 EVENTS_ERROR

Address offset: 0x11C

NFC error reported. The ERRORSTATUS register contains details on the source of the error.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_ERROR			NFC error reported. The ERRORSTATUS register contains details on the source of the error.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.13.14.25 EVENTS_RXERROR

Address offset: 0x128

NFC RX frame error reported. The FRAMESTATUS.RX register contains details on the source of the error.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_RXERROR			NFC RX frame error reported. The FRAMESTATUS.RX register contains details on the source of the error.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.13.14.26 EVENTS_ENDRX

Address offset: 0x12C

RX buffer (as defined by PACKETPTR and MAXLEN) in Data RAM full.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	EVENTS_ENDRX			RX buffer (as defined by PACKETPTR and MAXLEN) in Data RAM full.																																	
			NotGenerated	0	Event not generated																																	
			Generated	1	Event generated																																	

8.13.14.27 EVENTS_ENDTX

Address offset: 0x130

Transmission of data in RAM has ended, and EasyDMA has ended accessing the TX buffer

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	EVENTS_ENDTX			Transmission of data in RAM has ended, and EasyDMA has ended accessing the TX buffer																																	
			NotGenerated	0	Event not generated																																	
			Generated	1	Event generated																																	

8.13.14.28 EVENTS_AUTOCOLRESSTARTED

Address offset: 0x138

Auto collision resolution process has started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_AUTOCOLRESSTARTED						Auto collision resolution process has started																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.13.14.29 EVENTS_COLLISION

Address offset: 0x148

NFC auto collision resolution error reported.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_COLLISION						NFC auto collision resolution error reported.																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.13.14.30 EVENTS_SELECTED

Address offset: 0x14C

NFC auto collision resolution successfully completed

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_SELECTED			NFC auto collision resolution successfully completed																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.13.14.31 EVENTS_STARTED

Address offset: 0x150

EasyDMA is ready to receive or send frames.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_STARTED			EasyDMA is ready to receive or send frames.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.13.14.32 PUBLISH_READY

Address offset: 0x180

Publish configuration for event **READY**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that event READY will publish to																																
B	RW	EN																																				
			Disabled	0		Disable publishing																																
			Enabled	1		Enable publishing																																

8.13.14.33 PUBLISH_FIELDDETECTED

Address offset: 0x184

Publish configuration for event **FIELDDETECTED**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event FIELDDETECTED will publish to																													
B	RW	EN																																	
			Disabled	0		Disable publishing																													
			Enabled	1		Enable publishing																													

8.13.14.34 PUBLISH_FIELDLOST

Address offset: 0x188

Publish configuration for event **FIELDLOST**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event FIELDLOST will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.13.14.35 PUBLISH_TXFRAMESTART

Address offset: 0x18C

Publish configuration for event **TXFRAMESTART**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event TXFRAMESTART will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.13.14.36 PUBLISH_TXFRAMEEND

Address offset: 0x190

Publish configuration for event **TXFRAMEEND**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event TXFRAMEEND will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.13.14.37 PUBLISH_RXFRAMESTART

Address offset: 0x194

Publish configuration for event [RXFRAMESTART](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event RXFRAMESTART will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.13.14.38 PUBLISH_RXFRAMEEND

Address offset: 0x198

Publish configuration for event [RXFRAMEEND](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															

8.13.14.39 PUBLISH_ERROR

Address offset: 0x19C

Publish configuration for event [ERROR](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															

8.13.14.40 PUBLISH_RXERROR

Address offset: 0x1A8

Publish configuration for event [RXERROR](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0									
ID				B																								A					A		A		A		A		A		A	
Reset 0x00000000				0																																								
ID	R/W	Field	Value ID	Value		Description																																						
A	RW	CHIDX		[0..255]		DPPI channel that event RXERROR will publish to																																						
B	RW	EN																																										
			Disabled	0	Disable publishing																																							
			Enabled	1	Enable publishing																																							

8.13.14.41 PUBLISH_ENDRX

Address offset: 0x1AC

Publish configuration for event **ENDRX**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0									
ID				B																								A					A		A		A		A		A		A	
Reset 0x00000000				0																																								
ID	R/W	Field	Value ID	Value		Description																																						
A	RW	CHIDX		[0..255]		DPPI channel that event ENDRX will publish to																																						
B	RW	EN																																										
			Disabled	0		Disable publishing																																						
			Enabled	1		Enable publishing																																						

8.13.14.42 PUBLISH_ENDTX

Address offset: 0x1B0

Publish configuration for event **ENDTX**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0							
ID				B																								A					A		A		A		A		A	
Reset 0x00000000				0																																						
ID	R/W	Field	Value ID	Value		Description																																				
A	RW	CHIDX		[0..255]		DPPI channel that event ENDTX will publish to																																				
B	RW	EN																																								
			Disabled	0	Disable publishing																																					
			Enabled	1	Enable publishing																																					

8.13.14.43 PUBLISH_AUTOCOLRESSTARTED

Address offset: 0x1B8

Publish configuration for event **AUTOCOLRESSTARTED**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0									
ID				B																								A					A		A		A		A		A		A	
Reset 0x00000000				0																																								
ID	R/W	Field	Value ID	Value		Description																																						
A	RW	CHIDX		[0..255]		DPPI channel that event AUTOCOLRESSTARTED will publish to																																						
B	RW	EN																																										
			Disabled	0	Disable publishing																																							
			Enabled	1	Enable publishing																																							

8.13.14.44 PUBLISH_COLLISION

Address offset: 0x1C8

Publish configuration for event **PUBLISH_COLLISION**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event COLLISION will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.13.14.45 PUBLISH_SELECTED

Address offset: 0x1CC

Publish configuration for event **PUBLISH_SELECTED**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																												A A A A A A A A			
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event SELECTED will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.13.14.46 PUBLISH_STARTED

Address offset: 0x1D0

Publish configuration for event **PUBLISH_STARTED**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CHIDX		[0..255]	DPPI channel that event PUBLISH_STARTED will publish to																														
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.13.14.47 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	FIELDDETECTED_ACTIVATE			Shortcut between event FIELDDETECTED and task ACTIVATE																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
B	RW	FIELDLOST_SENSE			Shortcut between event FIELDLOST and task SENSE																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
C	RW	TXFRAMEEND_ENABLERXDATA			Shortcut between event TXFRAMEEND and task ENABLERXDATA																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														

8.13.14.48 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				O N M																L			K J I			H G F E D C B A									
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Enable or disable interrupt for event READY																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
B	RW	FIELDDETECTED			Enable or disable interrupt for event FIELDDETECTED																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
C	RW	FIELDLOST			Enable or disable interrupt for event FIELDLOST																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
D	RW	TXFRAMESTART			Enable or disable interrupt for event TXFRAMESTART																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
E	RW	TXFRAMEEND			Enable or disable interrupt for event TXFRAMEEND																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
F	RW	RXFRAMESTART			Enable or disable interrupt for event RXFRAMESTART																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
G	RW	RXFRAMEEND			Enable or disable interrupt for event RXFRAMEEND																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
H	RW	ERROR			Enable or disable interrupt for event ERROR																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
I	RW	RXERROR			Enable or disable interrupt for event RXERROR																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
J	RW	ENDRX			Enable or disable interrupt for event ENDRX																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
K	RW	ENDTX			Enable or disable interrupt for event ENDTX																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
L	RW	AUTOCOLRESSTARTED			Enable or disable interrupt for event AUTOCOLRESSTARTED																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
M	RW	COLLISION			Enable or disable interrupt for event COLLISION																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
N	RW	SELECTED			Enable or disable interrupt for event SELECTED																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
O	RW	STARTED			Enable or disable interrupt for event STARTED																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.13.14.49 INTENSET

Address offset: 0x304

Enable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			O N M L K J I H G F E D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	READY			Write '1' to enable interrupt for event READY																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
B	RW	FIELDDETECTED			Write '1' to enable interrupt for event FIELDDETECTED																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
C	RW	FIELDLOST			Write '1' to enable interrupt for event FIELDLOST																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
D	RW	TXFRAMESTART			Write '1' to enable interrupt for event TXFRAMESTART																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
E	RW	TXFRAMEEND			Write '1' to enable interrupt for event TXFRAMEEND																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
F	RW	RXFRAMESTART			Write '1' to enable interrupt for event RXFRAMESTART																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

8.13.14.50 INTENCLR

Disable interrupt

4503 018 v0.8

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
B	RW	FIELDDETECTED	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event FIELDDETECTED																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
C	RW	FIELDLOST	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event FIELDLOST																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
D	RW	TXFRAMESTART	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event TXFRAMESTART																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
E	RW	TXFRAMEEND	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event TXFRAMEEND																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
F	RW	RXFRAMESTART	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event RXFRAMESTART																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
G	RW	RXFRAMEEND	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event RXFRAMEEND																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
H	RW	ERROR	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event ERROR																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
I	RW	RXERROR	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event RXERROR																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
J	RW	ENDRX	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event ENDRX																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
K	RW	ENDTX	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event ENDTX																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
L	RW	AUTOCOLRESSTARTED	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event AUTOCOLRESSTARTED																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
M	RW	COLLISION	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event COLLISION																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
N	RW	SELECTED			Write '1' to disable interrupt for event SELECTED																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
O	RW	STARTED			Write '1' to disable interrupt for event STARTED																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.13.14.51 ERRORSTATUS

Address offset: 0x404

NFC Error Status register

Note: Write a bit to 1 to clear it. Writing 0 has no effect.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	RW	FRAMEDELAYTIMEOUT																		No STARTTX task triggered before expiration of the time set in															
		W1C																		FRAMEDELAYMAX															

8.13.14.52 FRAMESTATUS.RX

Address offset: 0x40C

Result of last incoming frame

Note: Write a bit to 1 to clear it. Writing 0 has no effect.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																				C	B	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	CRCERROR W1C			No valid end of frame (EoF) detected																																	
			CRCCorrect	0	Valid CRC detected																																	
			CRCError	1	CRC received does not match local check																																	
B	RW	PARITYSTATUS W1C			Parity status of received frame																																	
			ParityOK	0	Frame received with parity OK																																	
			ParityError	1	Frame received with parity error																																	
C	RW	OVERRUN W1C			Overrun detected																																	
			NoOverrun	0	No overrun detected																																	
			Overrun	1	Overrun error																																	

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000480					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	FRAMEDELAYMIN			Minimum frame delay in number of 13.56 MHz clock cycles																																	

8.13.14.57 FRAMEDELAYMAX

Address offset: 0x508

Maximum frame delay

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00001000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																		
A	RW	FRAMEDELAYMAX			Maximum frame delay in number of 13.56 MHz clock cycles																																		

8.13.14.58 FRAMEDELAYMODE

Address offset: 0x50C

Configuration register for the Frame Delay Timer

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																														
ID				A																																			A																										
Reset 0x00000001				0																																			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
ID	R/W	Field	Value ID	Value	Description																																																												
A	RW	FRAMEDELAYMODE			Configuration register for the Frame Delay Timer																																																												
			FreeRun	0	Transmission is independent of frame timer and will start when the STARTTX task is triggered. No timeout.																																																												
			Window	1	Frame is transmitted between FRAMEDELAYMIN and FRAMEDELAYMAX																																																												
			ExactVal	2	Frame is transmitted exactly at FRAMEDELAYMAX																																																												
			WindowGrid	3	Frame is transmitted on a bit grid between FRAMEDELAYMIN and FRAMEDELAYMAX																																																												

8.13.14.59 PACKETPTR

Address offset: 0x510

Packet pointer for TXD and RXD data storage in Data RAM

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						Packet pointer for TXD and RXD data storage in Data RAM. This address is a byte-aligned RAM address.																											

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.13.14.60 MAXLEN

Address offset: 0x514

Size of the RAM buffer allocated to TXD and RXD data storage each

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																												A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	MAXLEN		[0..257]		Size of the RAM buffer allocated to TXD and RXD data storage each																															

8.13.14.61 TXD.FRAMECONFIG

Address offset: 0x518

Configuration of outgoing frames

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																																D	C	B	A
Reset 0x00000017				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1		
ID	R/W	Field	Value ID	Value	Description																														
A	RW	PARITY			Indicates if parity is added to the frame																														
			NoParity	0	Parity is not added to TX frames																														
			Parity	1	Parity is added to TX frames																														
B	RW	DISCARDMODE			Discarding unused bits at start or end of a frame																														
			DiscardEnd	0	Unused bits are discarded at end of frame (EoF)																														
			DiscardStart	1	Unused bits are discarded at start of frame (SoF)																														
C	RW	SOF			Adding SoF or not in TX frames																														
			NoSoF	0	SoF symbol not added																														
			SoF	1	SoF symbol added																														
D	RW	CRCMODETX			CRC mode for outgoing frames																														
			NoCRCTX	0	CRC is not added to the frame																														
			CRC16TX	1	16 bit CRC added to the frame based on all the data read from RAM that is used in the frame																														

8.13.14.62 TXD.AMOUNT

Address offset: 0x51C

Size of outgoing frame

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B B B B B B B A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	TXDATABITS		[0..7]	<p>Number of bits in the last or first byte read from RAM that shall be included in the frame (excluding parity bit).</p> <p>The DISCARDMODE field in FRAMECONFIG.TX selects if unused bits is discarded at the start or at the end of a frame. A value of 0 data bytes and 0 data bits is invalid.</p>																														
B	RW	TXDATABYTES		[0..257]	<p>Number of complete bytes that shall be included in the frame, excluding CRC, parity, and framing.</p>																														

8.13.14.63 RXD.FRAMECONFIG

Address offset: 0x520

Configuration of incoming frames

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C B A																															
Reset 0x00000015				0 0																															

8.13.14.64 RXD.AMOUNT

Address offset: 0x524

Size of last incoming frame

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B B B B B A A A																															
Reset 0x00000000				0 0																															

8.13.14.65 MODULATIONCTRL

Address offset: 0x52C

Enables the modulation output to a GPIO pin which can be connected to a second external antenna.

See [MODULATIONPSEL](#) for GPIO configuration.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000001				0 0																															

8.13.14.66 MODULATIONPSEL

Address offset: 0x538

Pin select for Modulation control

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				C																								B				A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PIN		[0..31]				Pin number																											
B	RW	PORT		[0..3]				Port number																											
C	RW	CONNECT						Connection																											
			Disconnected	1				Disconnect																											
			Connected	0				Connect																											

8.13.14.67 MODE

Address offset: 0x550

Configure EasyDMA mode

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																					A	A
Reset 0x00000001					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	LPOP			Enable low-power operation, or use low-latency																																	
			LowLat	0	Low-latency operation																																	
			LowPower	1	Low-power operation																																	
			FullLowPower	3	Full Low-power operation																																	

8.13.14.68 NFCID1.LAST

Address offset: 0x590

Last NFCID1 part (4, 7 or 10 bytes ID)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D	D	D	D	D	D	D	D	C	C	C	C	C	C	C	B	B	B	B	B	B	B	B	B	A	A	A	A	A	A	A	A
Reset 0x00006363				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	1	0	1	1	0	0	0	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	RW	Z						NFCID1 byte Z (very last byte sent)																											
B	RW	Y						NFCID1 byte Y																											
C	RW	X						NFCID1 byte X																											
D	RW	W						NFCID1 byte W																											

8.13.14.69 NFCID1.SECONDLAST

Address offset: 0x594

Second last NFCID1 part (7 or 10 bytes ID)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																C	C	C	C	C	C	C	B	B	B	B	B	B	B	A	A	A	A	A	A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value												Description																				
A	RW	V														NFCID1 byte V																				
B	RW	U														NFCID1 byte U																				
C	RW	T														NFCID1 byte T																				

8.13.14.70 NFCID1.THIRDLAST

Address offset: 0x598

Third last NFCID1 part (10 bytes ID)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C C C C C C C C B B B B B B B A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	S						NFCID1 byte S																											
B	RW	R						NFCID1 byte R																											
C	RW	Q						NFCID1 byte Q																											

8.13.14.71 AUTOCOLRESCONFIG

Address offset: 0x59C

Controls the auto collision resolution function. This setting must be done before the NFCT peripheral is activated.

Note: When modifying this register, bit 1 must be written to 1.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																				A		
Reset 0x00000002				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0		
ID	R/W	Field	Value ID	Value		Description																																
A	RW	MODE				Enables/disables auto collision resolution																																
			Enabled	0		Auto collision resolution enabled																																
			Disabled	1		Auto collision resolution disabled																																

8.13.14.72 SENSRES

Address offset: 0x5A0

NFC-A SENS_RES auto-response settings

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				E E E E D D D C C B A A A A A																															
Reset 0x00000001				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
ID	R/W	Field	Value ID	Value		Description																													
A	RW	BITFRAMESDD				Bit frame SDD as defined by the b5:b1 of byte 1 in SENS_RES response in the NFC Forum, NFC Digital Protocol Technical Specification																													
			SDD00000	0	SDD pattern 00000																														
			SDD00001	1	SDD pattern 00001																														
			SDD00010	2	SDD pattern 00010																														
			SDD00100	4	SDD pattern 00100																														
			SDD01000	8	SDD pattern 01000																														
			SDD10000	16	SDD pattern 10000																														
B	RW	RFU5				Reserved for future use. Shall be 0.																													
C	RW	NFCIDSIZE				NFCID1 size. This value is used by the auto collision resolution engine.																													
			NFCID1Single	0	NFCID1 size: single (4 bytes)																														
			NFCID1Double	1	NFCID1 size: double (7 bytes)																														
			NFCID1Triple	2	NFCID1 size: triple (10 bytes)																														

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0											
ID																													E	E	E	E	D	D	D	D	C	C	B	A	A	A	A	A	A	A	A
Reset 0x00000001					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1									
ID	R/W	Field	Value ID	Value	Description																																										
D	RW	PLATFCONFIG			Tag platform configuration as defined by the b4:b1 of byte 2 in SENS_RES response in the NFC Forum, NFC Digital Protocol Technical Specification																																										
E	RW	RFU74			Reserved for future use. Shall be 0.																																										

8.13.14.73 SELRES

Address offset: 0x5A4

NFC-A SEL_RES auto-response settings

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				E D D C C B A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	RFU10						Reserved for future use. Shall be 0.																											
B	RW	CASCADE						Cascade as defined by the b3 of SEL_RES response in the NFC Forum, NFC Digital Protocol Technical Specification (controlled by hardware, shall be 0)																											
C	RW	RFU43						Reserved for future use. Shall be 0.																											
D	RW	PROTOCOL						Protocol as defined by the b7:b6 of SEL_RES response in the NFC Forum, NFC Digital Protocol Technical Specification																											
E	RW	RFU7						Reserved for future use. Shall be 0.																											

8.13.14.74 PADCONFIG

Address offset: 0x6D4

NFC pad configuration

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000001				0 1																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE			Enable NFC pads																														
					See GPIO port mapping section to find NFC pads.																														
			Disabled	0	NFC pads are used as GPIO pins																														
			Enabled	1	The NFC pads are configured as NFC antenna pins																														
					Also enables the protection for NFC pads.																														

8.14 PDM — Pulse density modulation interface

The pulse density modulation (PDM) module enables input of pulse density modulated signals from external audio frontends, for example, digital microphones. The PDM module generates the PDM clock and supports single-channel or dual-channel (left and right) data input. Data is transferred directly to RAM buffers using EasyDMA.

The main features of PDM are:

- Up to two PDM microphones configured as a left/right pair using the same data input
- 8 kHz, 16 kHz, 32 kHz, or 48k Hz output sample rate, 16-bit samples

- Supports digital microphone clocks at 768 kHz, 800 kHz, 1.024 MHz, 1.536 MHz, 2.048 MHz, 3.072 MHz, 1.28 MHz, and 2.56 MHz
- Selectable ratio of 32, 48, 50, 64, 80, 96, 100, or 128 between PDM_CLK and output sample rate
- HW decimation filters
- EasyDMA support for sample buffering

The PDM module illustrated below is interfacing up to two digital microphones with the PDM interface. EasyDMA is implemented to relieve the real-time requirements associated with controlling of the PDM slave from a low priority CPU execution context. It also includes all the necessary digital filter elements to produce pulse code modulation (PCM) samples. The PDM module allows continuous audio streaming.

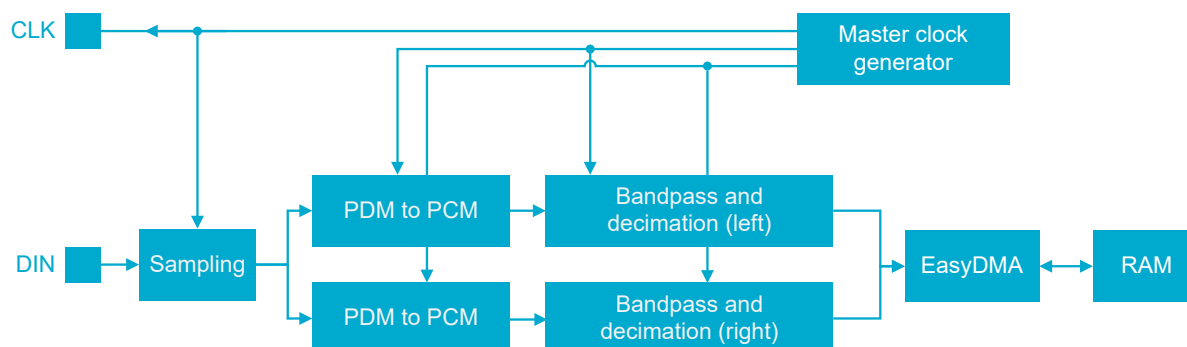


Figure 93: PDM module

8.14.1 Master clock generator

The master clock generator's PRESCALER register allows adjusting the PDM clock's frequency.

The master clock generator does not add any jitter to the HFCLK source chosen. It is recommended (but not mandatory) to use the Xtal as HFCLK source.

The PDM frequency can be adjusted by using PRESCALER, even while the clock generator is running. The PDM clock frequency $CLK = PCLK32M / (2 * PRESCALER)$

Requested PDM frequency f_{pdm} [Hz]	f_{source} [Hz]	RATIO	PRESCALER	Actual PDM frequency f_{actual} [Hz]	Sample frequency [Hz]	Error [%]
1024000	32000000 (PCLK32M)	64	31	1032258	16129	0.81
1280000	32000000 (PCLK32M)	80	25	1280000	16000	0.0
800000	32000000 (PCLK32M)	50	40	800000	16000	0.0

Table 42: Configuration examples

8.14.2 Module operation

By default, bits from the left PDM microphone are sampled on PDM_CLK falling edge, and bits for the right are sampled on the rising edge of PDM_CLK, resulting in two bitstreams. Each bitstream is fed into a digital filter which converts the PDM stream into 16-bit PCM samples, then filters and down-samples them to reach the appropriate sample rate.

The EDGE field in the MODE register allows swapping left and right, so that left will be sampled on rising edge, and right on falling.

The PDM module uses EasyDMA to store the samples coming out from the filters into one buffer in RAM. Depending on the mode chosen in the OPERATION field in the MODE register, memory either contains alternating left and right 16-bit samples (Stereo), or only left (or only right, depending on the value of the EDGE field) 16-bit samples (Mono). To ensure continuous PDM sampling, it is up to the application to update the EasyDMA destination address pointer as the previous buffer is filled.

The continuous transfer can be started or stopped by sending the START and STOP tasks. STOP becomes effective after the current frame has finished transferring, which will generate the STOPPED event. The STOPPED event indicates that all activity in the module is finished, and that the data is available in RAM (EasyDMA has finished transferring as well). Attempting to restart before receiving the STOPPED event may result in unpredictable behavior.

8.14.3 Decimation filter

In order to convert the incoming data stream into PCM audio samples, a decimation filter is included in the PDM interface module.

The input of the filter is the two-channel PDM serial stream (with left channel on clock high, right channel on clock low). Its output is 2×16 -bit PCM samples at a sample rate lower than the PDM clock rate at a ratio depending on the RATIO register.

The filter stage of each channel is followed by a digital volume control, to attenuate or amplify the output samples in a range of -20 dB to +20 dB around the default (reset) setting, defined by $G_{\text{PDM,default}}$. The gain is controlled by the GAINL and GAINR registers.

As an example, if the goal is to achieve 2500 RMS output samples (16-bit) with a 1 kHz 90 dBA signal into a -26 dBFS sensitivity PDM microphone, do the following:

- Sum the PDM module's default gain ($G_{\text{PDM,default}}$) and the gain introduced by the microphone and acoustic path of his implementation (an attenuation would translate into a negative gain)
- Adjust GAINL and GAINR by the above summed amount. Assuming that only the PDM module influences the gain, GAINL and GAINR must be set to $-G_{\text{PDM,default}}$ dB to achieve the requirement.

With $G_{\text{PDM,default}} = 3.2$ dB, and as GAINL and GAINR are expressed in 0.5 dB steps, the closest value to program would be 3.0 dB, which can be calculated as:

$$\text{GAINL} = \text{GAINR} = (\text{DefaultGain} - (2 * 3))$$

Remember to check that the resulting values programmed into GAINL and GAINR fall within MinGain and MaxGain.

8.14.4 EasyDMA

Samples will be written directly to RAM, and EasyDMA must be configured accordingly.

The address pointer for the EasyDMA channel is set in SAMPLE.PTR register. If the destination address set in SAMPLE.PTR is not pointing to the Data RAM region, an EasyDMA transfer may result in a HardFault or RAM corruption. See [Memory](#) on page 19 for more information about the different memory regions.

Note: The value programmed in SAMPLE.PTR register should be 32bit aligned

The DMA transfer supports Stereo (left and right 16-bit samples) and Mono (left only) data transfer as configured in the OPERATION field of the MODE register. The samples are stored little endian.

MODE.OPERATION	Bits per sample	Result stored per RAM word	Physical RAM allocated (32-bit words)	Result boundary indexes in RAM	Note
Stereo	32 (2x16)	L+R	$\text{ceil}(\text{SAMPLE.MAXCNT}/2)$	$R0=[31:16]; L0=[15:0]$	Default
Mono	16	2xL	$\text{ceil}(\text{SAMPLE.MAXCNT}/2)$	$L1=[31:16]; L0=[15:0]$	

Table 43: DMA sample storage

The destination buffer in RAM consists of one block, the size of which is set in SAMPLE.MAXCNT register. Format is number of bytes used by the samples that will be stored in the memory. The physical RAM allocated is always:

```
(RAM allocation, in bytes) = SAMPLE.MAXCNT * 2;
```

(but the mapping of the samples depends on MODE.OPERATION.

If OPERATION=Stereo, RAM will contain a succession of left and right samples.

If OPERATION=Mono, RAM will contain a succession of left only samples.

For a given value of SAMPLE.MAXCNT, the buffer in RAM can contain half the stereo sampling time as compared to the mono sampling time.

The PDM acquisition can be started by the START task, after the SAMPLE.PTR and SAMPLE.MAXCNT registers have been written. When starting the module, it will take some time for the filters to start outputting valid data. Transients from the PDM microphone itself may also occur. The first few samples (typically around 50) might hence contain invalid values or transients. It is therefore advised to discard the first few samples after a PDM start.

As soon as the STARTED event is received, the firmware can write the next SAMPLE.PTR value (this register is double-buffered), to ensure continuous operation.

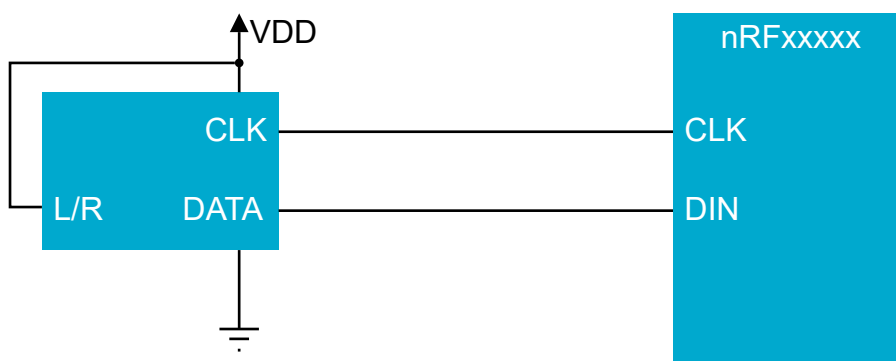
When the buffer in RAM is filled with samples, an END event is triggered. The firmware can start processing the data in the buffer. Meanwhile, the PDM module starts acquiring data into the new buffer pointed to by SAMPLE.PTR, and sends a new STARTED event, so that the firmware can update SAMPLE.PTR to the next buffer address.

8.14.5 Hardware example

PDM can be configured with a single microphone (mono), or with two microphones.

When a single microphone is used, connect the microphone clock to CLK, and data to DIN.

The following figures show a single PDM microphone, wired as left.



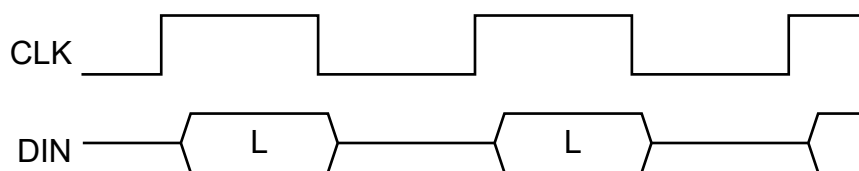


Figure 94: Left wired microphone

The following figures show a single PDM microphone, wired as right.

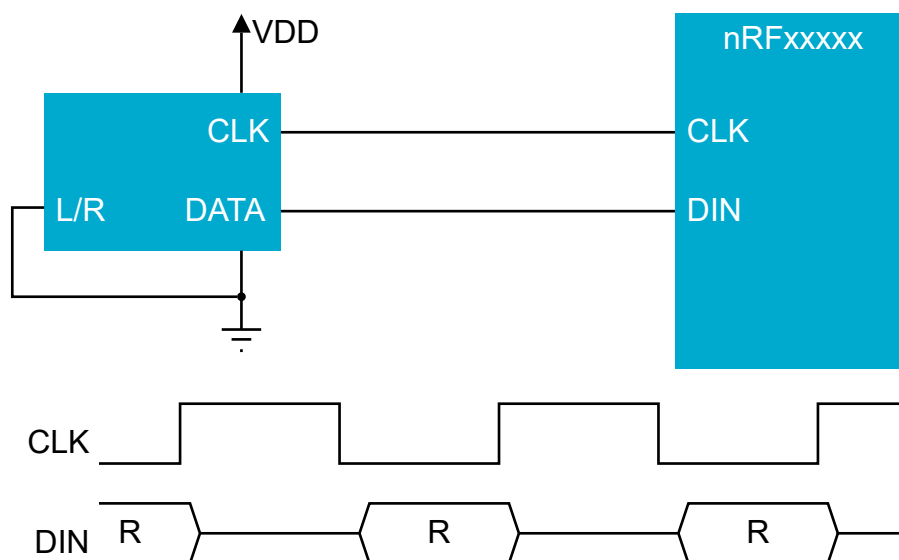


Figure 95: Right wired microphone

Note that in a single microphone (mono) configuration, depending on the microphone's implementation, either the left or the right channel (sampled at falling or rising CLK edge respectively) will contain reliable data.

If two microphones are used, one of them has to be set as left, the other as right (L/R pin tied high or to GND on the respective microphone). It is strongly recommended to use two microphones of exactly the same brand and type so that their timings in left and right operation match.

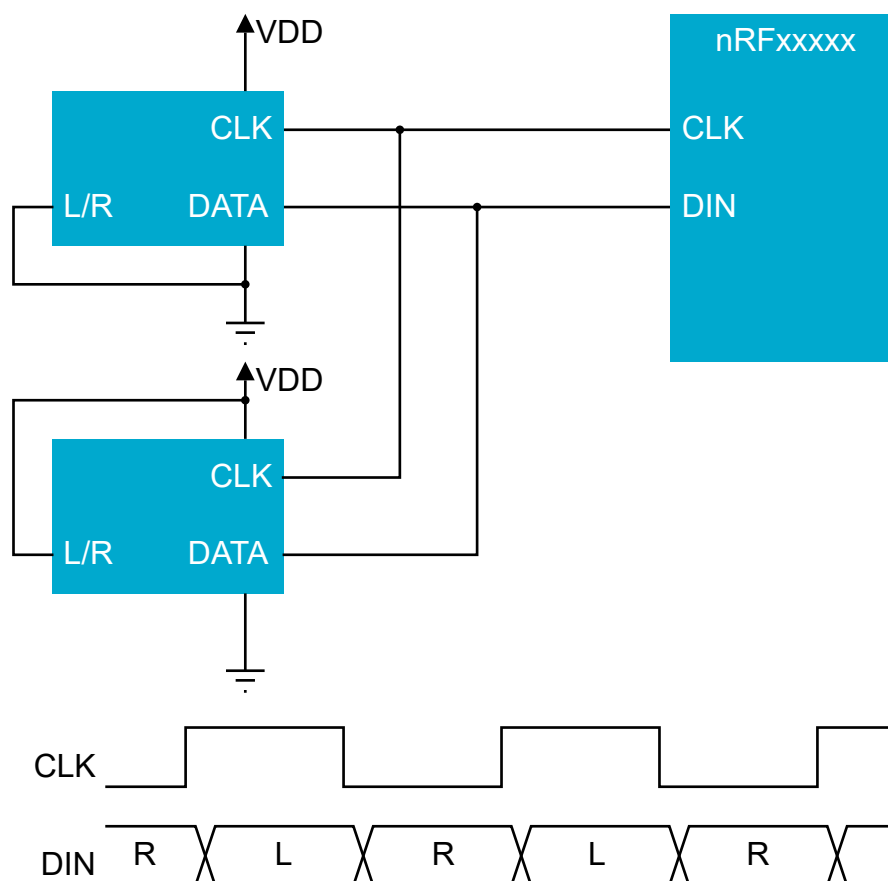


Figure 96: Example of two PDM microphones

8.14.6 Pin configuration

The CLK and DIN signals associated to the PDM module are mapped to physical pins according to the configuration specified in the PSEL.CLK and PSEL.DIN registers respectively. If the CONNECT field in any PSEL register is set to Disconnected, the associated PDM module signal will not be connected to the required physical pins, and will not operate properly.

The PSEL.CLK and PSEL.DIN registers and their configurations are only used as long as the PDM module is enabled, and retained only as long as the device is in System ON mode. See [POWER — Power control](#) on page 97 for more information about power modes. When the peripheral is disabled, the pins will behave as regular GPIOs, and use the configuration in their respective OUT bit field and PIN_CNF[n] register.

To ensure correct behavior in the PDM module, the pins used by the PDM module must be configured in the GPIO peripheral as described in [GPIO configuration before enabling peripheral](#) on page 394 before enabling the PDM module. This is to ensure that the pins used by the PDM module are driven correctly if the PDM module itself is temporarily disabled or the device temporarily enters System OFF. This configuration must be retained in the GPIO for the selected I/Os as long as the PDM module is supposed to be connected to an external PDM circuit.

Only one peripheral can be assigned to drive a particular GPIO pin at a time. Failing to do so may result in unpredictable behavior.

PDM signal	PDM pin	Direction	Output value	Comment
CLK	As specified in PSEL.CLK	Output	0	
DIN	As specified in PSEL.DIN	Input	Not applicable	

Table 44: GPIO configuration before enabling peripheral

8.14.7 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
PDM20 : S	GLOBAL	0x500D0000	US	S	SA	No	Pulse density modulation (digital microphone) interface PDM20
PDM20 : NS		0x400D0000					
PDM21 : S	GLOBAL	0x500D1000	US	S	SA	No	Pulse density modulation (digital microphone) interface PDM21
PDM21 : NS		0x400D1000					

Configuration

Instance	Domain	Configuration
PDM20 : S	GLOBAL	Use GPIO port P1
PDM20 : NS		Supports 8, 16, 32, 48 kHz sample rate.
		CURRENTAMOUNT register included.
PDM21 : S	GLOBAL	Use GPIO port P1
PDM21 : NS		Supports 8, 16, 32, 48 kHz sample rate.
		CURRENTAMOUNT register included.

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Starts continuous PDM transfer
TASKS_STOP	0x004		Stops PDM transfer
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
EVENTS_STARTED	0x100		PDM transfer has started
EVENTS_STOPPED	0x104		PDM transfer has finished
EVENTS_END	0x108		The PDM has written the last sample specified by SAMPLE.MAXCNT (or the last sample after a STOP task has been received) to Data RAM
EVENTS_DMA.BUSERROR	0x110		This event is generated if an error occurs during the bus transfer.
PUBLISH_STARTED	0x180		Publish configuration for event STARTED
PUBLISH_STOPPED	0x184		Publish configuration for event STOPPED
PUBLISH_END	0x188		Publish configuration for event END
PUBLISH_DMA.BUSERROR	0x190		Publish configuration for event DMA.BUSERROR
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt

Register	Offset	TZ	Description
INTPEND	0x30C		Pending interrupts
ENABLE	0x500		PDM module enable register
MODE	0x508		Defines the routing of the connected PDM microphone signals
GAINL	0x518		Left output gain adjustment
GAINR	0x51C		Right output gain adjustment
RATIO	0x520		Selects the decimation ratio between PDM_CLK and output sample rate. Change PRESCALER accordingly.
PSEL.CLK	0x540		Pin number configuration for PDM CLK signal
PSEL.DIN	0x544		Pin number configuration for PDM DIN signal
SAMPLE.PTR	0x560		RAM address pointer to write samples to with EasyDMA
SAMPLE.MAXCNT	0x564		Number of bytes to allocate memory for in EasyDMA mode
PRESCALER	0x580		The prescaler is used to set the PDM frequency
DMA.TERMINATEONBUSERROR	0x700		Terminate the transaction if a BUSERROR event is detected.
DMA.BUSERRORADDRESS	0x704		Address of transaction that generated the last BUSERROR event.

8.14.7.1 TASKS_START

Address offset: 0x000

Starts continuous PDM transfer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_START						Starts continuous PDM transfer																											
			Trigger	1				Trigger task																											

8.14.7.2 TASKS_STOP

Address offset: 0x004

Stops PDM transfer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_STOP						Stops PDM transfer																											
			Trigger	1				Trigger task																											

8.14.7.3 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task [START](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task START will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.14.7.4 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task **STOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task STOP will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.14.7.5 EVENTS_STARTED

Address offset: 0x100

PDM transfer has started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_STARTED			PDM transfer has started																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.14.7.6 EVENTS_STOPPED

Address offset: 0x104

PDM transfer has finished

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_STOPPED			PDM transfer has finished																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.14.7.7 EVENTS_END

Address offset: 0x108

The PDM has written the last sample specified by `SAMPLE.MAXCNT` (or the last sample after a `STOP` task has been received) to Data RAM

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																													
A	RW	EVENTS_END				The PDM has written the last sample specified by SAMPLE.MAXCNT (or the last sample after a STOP task has been received) to Data RAM																													
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.14.7.8 EVENTS_DMA

Peripheral events.

8.14.7.8.1 EVENTS_DMA.BUSERROR

Address offset: 0x110

This event is generated if an error occurs during the bus transfer.

When this event is generated, the address which caused the error can be read from the `BUSERRORADDRESS` register.

Errors occurring while the `EVENTS_BUSERROR` register is set are ignored.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	BUSERROR			This event is generated if an error occurs during the bus transfer.																															
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																															
					Errors occurring while the EVENTS_BUSERROR register is set are ignored.																															
			NotGenerated	0	Event not generated																															
			Generated	1	Event generated																															

8.14.7.9 PUBLISH_STARTED

Address offset: 0x180

Publish configuration for event `STARTED`

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that event STARTED will publish to																																
B	RW	EN																																				
			Disabled	0		Disable publishing																																
			Enabled	1		Enable publishing																																

8.14.7.10 PUBLISH_STOPPED

Address offset: 0x184

Publish configuration for event **STOPPED**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event STOPPED will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.14.7.11 PUBLISH_END

Address offset: 0x188

Publish configuration for event **END**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event END will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.14.7.12 PUBLISH_DMA

Publish configuration for events

8.14.7.12.1 PUBLISH_DMA.BUSERROR

Address offset: 0x190

Publish configuration for event **DMA.BUSERROR**

When this event is generated, the address which caused the error can be read from the **BUSERRORADDRESS** register.

Errors occurring while the **EVENTS_BUSERROR** register is set are ignored.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event DMA.BUSERROR will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.14.7.13 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	STARTED			Enable or disable interrupt for event STARTED																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
B	RW	STOPPED			Enable or disable interrupt for event STOPPED																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
C	RW	END			Enable or disable interrupt for event END																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
D	RW	DMABUSERERROR			Enable or disable interrupt for event DMABUSERERROR																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
					Errors occurring while the EVENTS_BUSERERROR register is set are ignored.																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.14.7.14 INTENSET

Address offset: 0x304

Enable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	STARTED			Write '1' to enable interrupt for event STARTED																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
B	RW	STOPPED			Write '1' to enable interrupt for event STOPPED																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
C	RW	END			Write '1' to enable interrupt for event END																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
D	RW	DMABUSERERROR			Write '1' to enable interrupt for event DMABUSERERROR																													
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																													
					Errors occurring while the EVENTS_BUSERERROR register is set are ignored.																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

8.14.7.15 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																															D	C	B	A
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	STARTED			Write '1' to disable interrupt for event STARTED																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
B	RW	STOPPED			Write '1' to disable interrupt for event STOPPED																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
C	RW	END			Write '1' to disable interrupt for event END																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
D	RW	DMABUSERROR			Write '1' to disable interrupt for event DMABUSERROR																													
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																													
					Errors occurring while the EVENTS_BUSERROR register is set are ignored.																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

8.14.7.16 INTPEND

Address offset: 0x30C

Pending interrupts

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																	
ID																																	D	C	B	A
Reset 0x00000000			0 0																																	
ID	R/W	Field	Value ID	Value	Description																															
A	R	STARTED			Read pending status of interrupt for event STARTED																															
			NotPending	0	Read: Not pending																															
			Pending	1	Read: Pending																															
B	R	STOPPED			Read pending status of interrupt for event STOPPED																															
			NotPending	0	Read: Not pending																															
			Pending	1	Read: Pending																															
C	R	END			Read pending status of interrupt for event END																															
			NotPending	0	Read: Not pending																															
			Pending	1	Read: Pending																															
D	R	DMABUSERERROR			Read pending status of interrupt for event DMABUSERERROR																															
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																															
					Errors occurring while the EVENTS_BUSERROR register is set are ignored.																															
			NotPending	0	Read: Not pending																															
			Pending	1	Read: Pending																															

8.14.7.17 ENABLE

Address offset: 0x500

PDM module enable register

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	ENABLE			Enable or disable PDM module																															
			Disabled	0	Disable																															
			Enabled	1	Enable																															

8.14.7.18 MODE

Address offset: 0x508

Defines the routing of the connected PDM microphone signals

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	OPERATION			Mono or stereo operation																														
			Stereo	0	Sample and store one pair (left + right) of 16-bit samples per RAM word R=[31:16]; L=[15:0]																														
			Mono	1	Sample and store two successive left samples (16 bits each) per RAM word L1=[31:16]; L0=[15:0]																														
B	RW	EDGE			Defines on which PDM_CLK edge left (or mono) is sampled.																														
					The right channel is sampled on the opposite edge of the left channel.																														
					When EDGE is set to 1 (LeftRising) and stereo input is used the right and left channels are swapped relative to EDGE set to 0 (LeftFalling).																														
			LeftFalling	1	Left (or mono) is sampled on falling edge of PDM_CLK																														
			LeftRising	0	Left (or mono) is sampled on rising edge of PDM_CLK																														

8.14.7.19 GAINL

Address offset: 0x518

Left output gain adjustment

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			A A A A A A A																															
Reset 0x00000028			0 1 0 1 0 0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	GAINL			Left output gain adjustment, in 0.5 dB steps, around the default module gain (see electrical parameters)																													
					0x00 -20 dB gain adjust																													
					0x01 -19.5 dB gain adjust																													
					(...)																													
					0x27 -0.5 dB gain adjust																													
					0x28 0 dB gain adjust																													
					0x29 +0.5 dB gain adjust																													
					(...)																													
					0x4F +19.5 dB gain adjust																													
					0x50 +20 dB gain adjust																													
			MinGain	0x00	-20 dB gain adjustment (minimum)																													
			DefaultGain	0x28	0 dB gain adjustment																													
			MaxGain	0x50	+20 dB gain adjustment (maximum)																													

8.14.7.20 GAINR

Address offset: 0x51C

Right output gain adjustment

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																														A	A	A	A	A	A
Reset 0x00000028		0 1 0 1 0 0 0																																	
ID	R/W	Field	Value ID	Value	Description																														
A	RW	GAINR			Right output gain adjustment, in 0.5 dB steps, around the default module gain (see electrical parameters)																														
			MinGain	0x00	-20 dB gain adjustment (minimum)																														
			DefaultGain	0x28	0 dB gain adjustment																														
			MaxGain	0x50	+20 dB gain adjustment (maximum)																														

8.14.7.21 RATIO

Address offset: 0x520

Selects the decimation ratio between PDM_CLK and output sample rate.

Change PRESCALER accordingly.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000002				0 1 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	RATIO						Selects the decimation ratio between PDM_CLK and output sample rate																											
			Ratio32	0				Ratio of 32																											
			Ratio48	1				Ratio of 48																											
			Ratio50	2				Ratio of 50																											
			Ratio64	3				Ratio of 64																											
			Ratio80	4				Ratio of 80																											
			Ratio96	5				Ratio of 96																											
			Ratio100	6				Ratio of 100																											
			Ratio128	7				Ratio of 128																											

8.14.7.22 PSEL.CLK

Address offset: 0x540

Pin number configuration for PDM CLK signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				C																								B				B	B	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
ID	R/W	Field	Value ID	Value				Description																													
A	RW	PIN		[0..31]				Pin number																													
B	RW	PORT		[0..7]				Port number																													
C	RW	CONNECT						Connection																													
			Disconnected	1				Disconnect																													
			Connected	0				Connect																													

8.14.7.23 PSEL.DIN

Address offset: 0x544

Pin number configuration for PDM DIN signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				C																								B				B	B	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
ID	R/W	Field	Value ID	Value				Description																													
A	RW	PIN		[0..31]				Pin number																													
B	RW	PORT		[0..7]				Port number																													
C	RW	CONNECT						Connection																													
			Disconnected	1				Disconnect																													
			Connected	0				Connect																													

8.14.7.24 SAMPLE.PTR

Address offset: 0x560

RAM address pointer to write samples to with EasyDMA

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	SAMPLEPTR						Address to write PCM samples to over DMA																											

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.14.7.25 SAMPLE.MAXCNT

Address offset: 0x564

Number of bytes to allocate memory for in EasyDMA mode

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value				Description																													
A	RW	BUFFSIZE		[0..32767]				Length of DMA RAM allocation in number of bytes																													

8.14.7.26 PRESCALER

Address offset: 0x580

The prescaler is used to set the PDM frequency

The prescaler divides the clock by DIVISOR to make the PDM clock. The resulting frequency is given by the frequency of the 'core clock' divided by DIVISOR.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																													A	A	A	A	A	A	A	
Reset 0x00000004				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
ID	R/W	Field	Value ID	Value				Description																												
A	RW	DIVISOR		4..126				Core clock to PDM divisor																												

8.14.7.27 DMA.TERMINATEONBUSERROR

Address offset: 0x700

Terminate the transaction if a BUSERROR event is detected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ENABLE																																	
			Disabled	0				Disable																											
			Enabled	1				Enable																											

8.14.7.28 DMA.BUSERRORADDRESS

Address offset: 0x704

Address of transaction that generated the last BUSERROR event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	ADDRESS																																	

8.15 PWM — Pulse width modulation

The pulse width modulation peripheral (PWM) enables the generation of pulse width modulated signals on GPIO. The peripheral implements a counter with up-count mode and up-and-down-count mode, consisting of four PWM channels that can drive assigned GPIO pins.

The main features of PWM are the following:

- Programmable PWM frequency
- Up to four PWM channels with individual polarity and duty cycle values
- Edge or center-aligned pulses across PWM channels
- Multiple duty cycle arrays (sequences) defined in RAM
- Autonomous and glitch-free update of duty cycle values directly from memory through EasyDMA (no CPU involvement)
- Change of polarity, duty cycle, and base frequency on every PWM period
- RAM sequences can be repeated or connected into loops

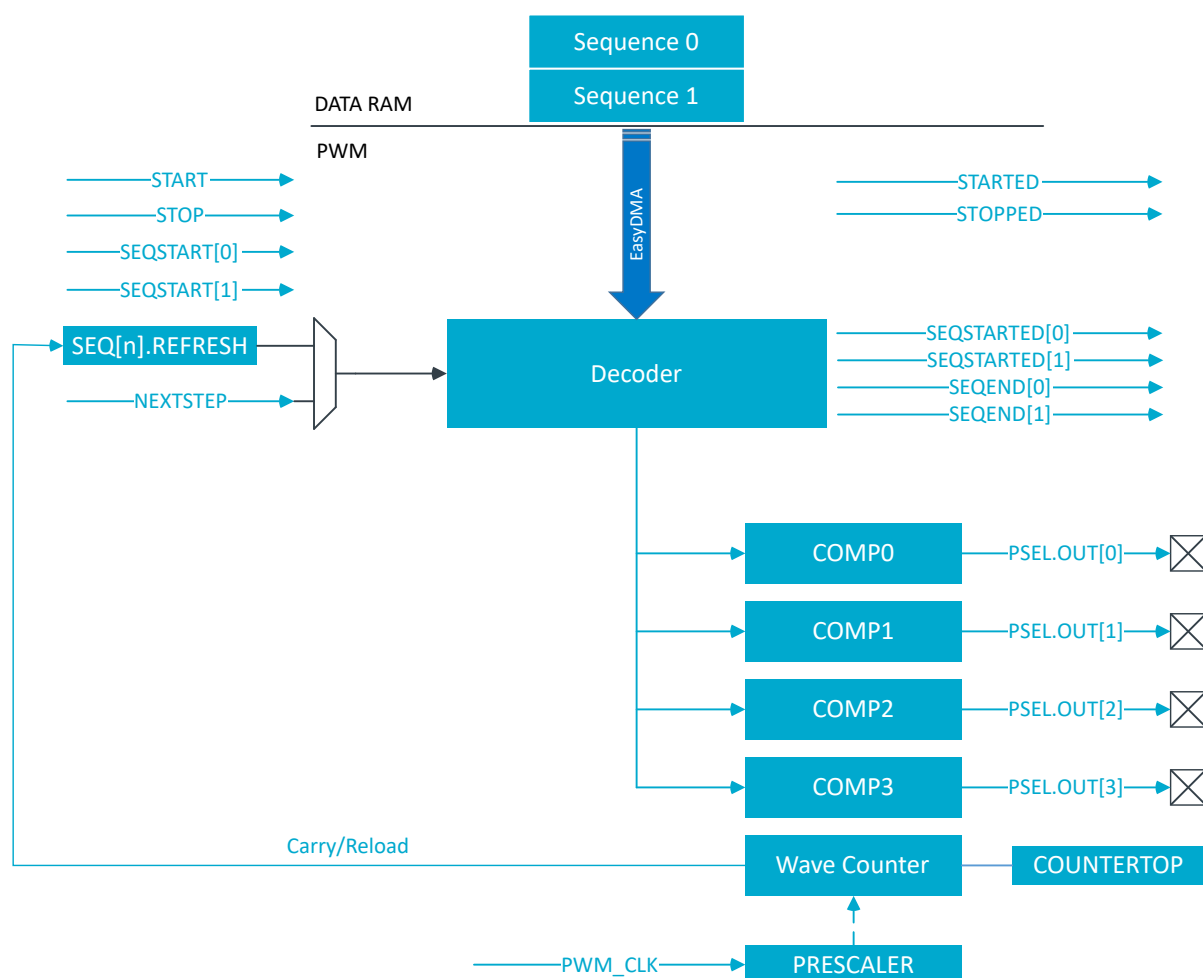


Figure 97: PWM module

8.15.1 Wave counter

The wave counter is responsible for generating the pulses at a duty cycle that depends on the compare values, and at a frequency that depends on COUNTERTOP.

There is one common 15-bit counter with four compare channels. Thus, all four channels will share the same period (PWM frequency), but can have individual duty cycle and polarity. The polarity is set by the most significant bit (MSB) of the value read from RAM (see figure [Decoder memory access modes](#) on page 409). When the MSB bit is high (FallingEdge polarity), OUT[n] starts high to become low during the given PWM cycle, whereas the inverse occurs for RisingEdge polarity. Whether the counter counts up, or up and down, is controlled by the MODE register.

The timer top value is controlled by the COUNTERTOP register. This register value, in conjunction with the selected PRESCALER of the PWM_CLK, will result in a given PWM period. A COUNTERTOP value smaller than the compare setting will result in a state where no PWM edges are generated. OUT[n] is held high, given that the polarity is set to FallingEdge. All compare registers are internal and can only be configured through decoder presented later. COUNTERTOP can be safely written at any time.

Sampling follows the START task. If DECODER.LOAD=WaveForm, the register value is ignored and taken from RAM instead (see section [Decoder with EasyDMA](#) on page 409 for more details). If DECODER.LOAD is anything else than the WaveForm, it is sampled following a STARTSEQ[n] task and when loading a new value from RAM during a sequence playback.

The following figure shows the counter operating in up mode (MODE=PWM_MODE_Up), with two PWM channels with the same frequency but different duty cycle:

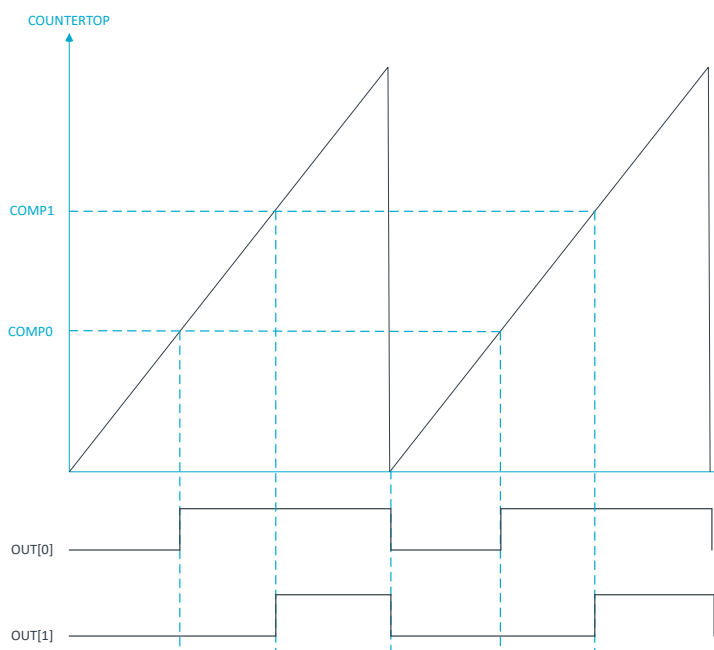


Figure 98: PWM counter in up mode example - RisingEdge polarity

The counter is automatically reset to zero when COUNTERTOP is reached and OUT[n] will invert. OUT[n] is held low if the compare value is 0 and held high if set to COUNTERTOP, given that the polarity is set to

FallingEdge. Counter running in up mode results in pulse widths that are edge-aligned. The following is the code for the counter in up mode example:

```
uint16_t pwm_seq[4] = {PWM_CH0_DUTY, PWM_CH1_DUTY, PWM_CH2_DUTY, PWM_CH3_DUTY};
NRF_PWM0->PSEL.OUT[0] = (first_port << PWM_PSEL_OUT_PORT_Pos) |
                        (first_pin << PWM_PSEL_OUT_PIN_Pos) |
                        (PWM_PSEL_OUT_CONNECT_Connected <<
                         PWM_PSEL_OUT_CONNECT_Pos);
NRF_PWM0->PSEL.OUT[1] = (second_port << PWM_PSEL_OUT_PORT_Pos) |
                        (second_pin << PWM_PSEL_OUT_PIN_Pos) |
                        (PWM_PSEL_OUT_CONNECT_Connected <<
                         PWM_PSEL_OUT_CONNECT_Pos);
NRF_PWM0->ENABLE      = (PWM_ENABLE_ENABLE_Enabled << PWM_ENABLE_ENABLE_Pos);
NRF_PWM0->MODE        = (PWM_MODE_UPDOWN_Up << PWM_MODE_UPDOWN_Pos);
NRF_PWM0->PRESCALER   = (PWM_PRESCALER_PRESCALER_DIV_1 <<
                         PWM_PRESCALER_PRESCALER_Pos);
NRF_PWM0->COUNTERTOP = (16000 << PWM_COUNTERTOP_COUNTERTOP_Pos); //1 msec
NRF_PWM0->LOOP         = (PWM_LOOP_CNT_Disabled << PWM_LOOP_CNT_Pos);
NRF_PWM0->DECODER      = (PWM_DECODER_LOAD_Individual << PWM_DECODER_LOAD_Pos) |
                        (PWM_DECODER_MODE_RefreshCount << PWM_DECODER_MODE_Pos);
NRF_PWM0->DMA.SEQ[0].PTR = ((uint32_t) (pwm_seq) << PWM_DMA_SEQ_PTR_PTR_Pos);
NRF_PWM0->DMA.SEQ[0].MAXCNT = (sizeof(pwm_seq) << PWM_DMA_SEQ_MAXCNT_MAXCNT_Pos);
NRF_PWM0->SEQ[0].REFRESH = 0;
NRF_PWM0->SEQ[0].ENDDELAY = 0;
NRF_PWM0->TASKS_DMA.SEQ[0].START = 1;
```

When the counter is running in up mode, the following formula can be used to compute the PWM period and the step size:

PWM period: $T_{\text{PWM(Up)}} = T_{\text{PWM_CLK}} * \text{COUNTERTOP}$

Step width/Resolution: $T_{\text{steps}} = T_{\text{PWM_CLK}}$

The following figure shows the counter operating in up-and-down mode (MODE=PWM_MODE_UpAndDown), with two PWM channels with the same frequency but different duty cycle and output polarity:

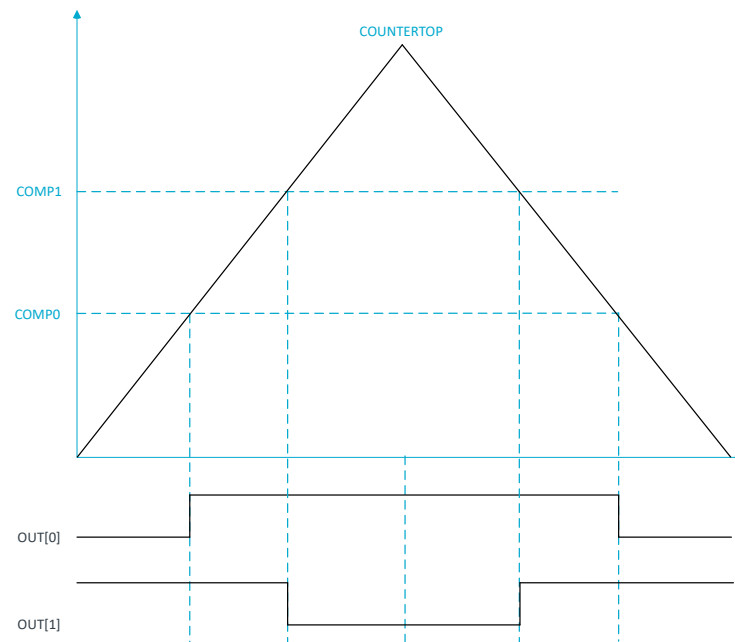


Figure 99: PWM counter in up-and-down mode example

The counter starts decrementing to zero when COUNTERTOP is reached and will invert the OUT[n] when compare value is hit for the second time. This results in a set of pulses that are center-aligned. The following is the code for the counter in up-and-down mode example:

```
uint16_t pwm_seq[4] = {PWM_CH0_DUTY, PWM_CH1_DUTY, PWM_CH2_DUTY, PWM_CH3_DUTY};
NRF_PWM0->PSEL.OUT[0] = (first_port << PWM_PSEL_OUT_PORT_Pos) |
    (first_pin << PWM_PSEL_OUT_PIN_Pos) |
    (PWM_PSEL_OUT_CONNECT_Connected <<
        PWM_PSEL_OUT_CONNECT_Pos);
NRF_PWM0->PSEL.OUT[1] = (second_pin << PWM_PSEL_OUT_PIN_Pos) |
    (PWM_PSEL_OUT_CONNECT_Connected <<
        PWM_PSEL_OUT_CONNECT_Pos);
NRF_PWM0->ENABLE = (PWM_ENABLE_ENABLE_Enabled << PWM_ENABLE_ENABLE_Pos);
NRF_PWM0->MODE = (PWM_MODE_UPDOWN_UpAndDown << PWM_MODE_UPDOWN_Pos);
NRF_PWM0->PRESCALER = (PWM_PRESCALER_PRESCALER_DIV_1 <<
    PWM_PRESCALER_PRESCALER_Pos);
NRF_PWM0->COUNTERTOP = (16000 << PWM_COUNTERTOP_COUNTERTOP_Pos); //1 msec
NRF_PWM0->LOOP = (PWM_LOOP_CNT_Disabled << PWM_LOOP_CNT_Pos);
NRF_PWM0->DECODER = (PWM_DECODER_LOAD_Individual << PWM_DECODER_LOAD_Pos) |
    (PWM_DECODER_MODE_RefreshCount << PWM_DECODER_MODE_Pos);
NRF_PWM0->DMA.SEQ[0].PTR = ((uint32_t) (pwm_seq) << PWM_DMA_SEQ_PTR_PTR_Pos);
NRF_PWM0->DMA.SEQ[0].MAXCNT = (sizeof(pwm_seq) << PWM_DMA_SEQ_MAXCNT_MAXCNT_Pos);
NRF_PWM0->SEQ[0].REFRESH = 0;
NRF_PWM0->SEQ[0].ENDDelay = 0;
NRF_PWM0->TASKS_DMA.SEQ[0].START = 1;
```

When the counter is running in up-and-down mode, the following formula can be used to compute the PWM period and the step size:

$$T_{\text{PWM (Up And Down)}} = T_{\text{PWM_CLK}} * 2 * \text{COUNTERTOP}$$

$$\text{Step width/Resolution: } T_{\text{steps}} = T_{\text{PWM_CLK}} * 2$$

8.15.2 Decoder with EasyDMA

The decoder uses EasyDMA to take PWM parameters stored in RAM and update the internal compare registers of the wave counter, based on the mode of operation.

PWM parameters are organized into a sequence containing at least one half word (16 bit). Its most significant bit[15] denotes the polarity of the OUT[n] while bit[14:0] is the 15-bit compare value.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
Id																					B	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
Id	RW	Field	Value Id	Value		Description																																
A	RW	COMPARE				Duty cycle setting - value loaded to internal compare register																																
B	RW	POLARITY				Edge polarity of GPIO.																																
			RisingEdge	0	First edge within the PWM period is rising																																	
			FallingEdge	1	First edge within the PWM period is falling																																	

The DECODER register controls how the RAM content is interpreted and loaded into the internal compare registers. The LOAD field controls if the RAM values are loaded to all compare channels, or to update a group or all channels with individual values. The following figure illustrates how parameters stored in RAM are organized and routed to various compare channels in different modes:

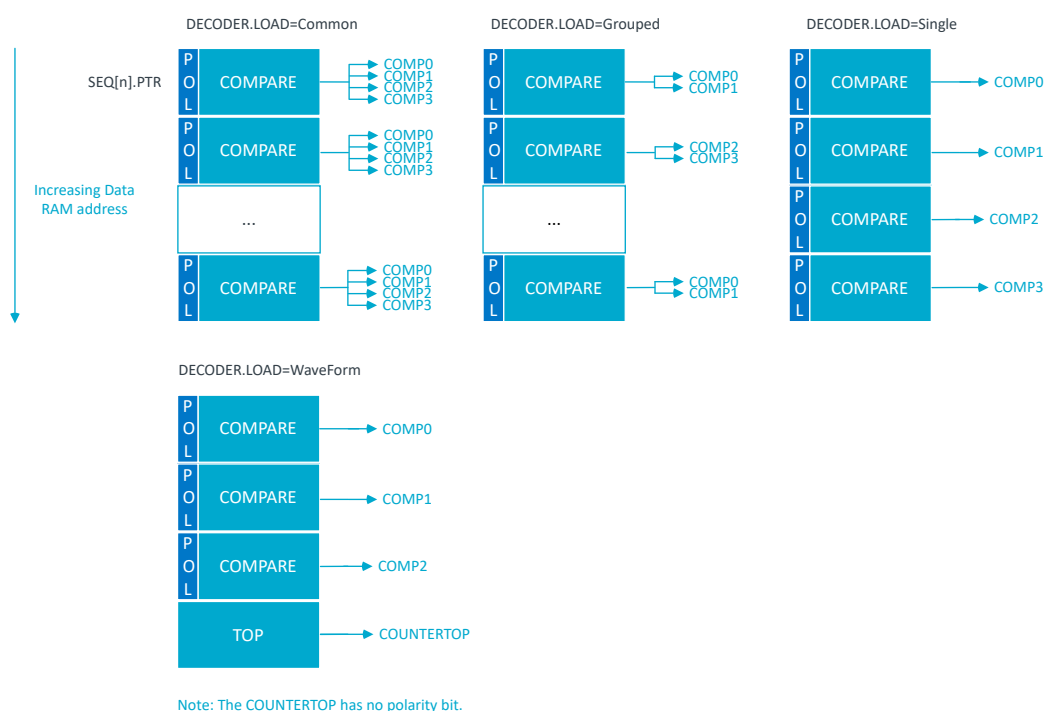


Figure 100: Decoder memory access modes

A special mode of operation is available when DECODER.LOAD is set to WaveForm. In WaveForm mode, up to three PWM channels can be enabled - OUT[0] to OUT[2]. In RAM, four values are loaded at a time: the first, second and third location are used to load the values, and the fourth RAM location is used to load the COUNTERTOP register. This way one can have up to three PWM channels with a frequency base that changes on a per PWM period basis. This mode of operation is useful for arbitrary wave form generation in applications, such as LED lighting.

The register `SEQ[n].REFRESH=N` (one per sequence $n=0$ or 1) will instruct a new RAM stored pulse width value on every $(N+1)^{\text{th}}$ PWM period. Setting the register to zero will result in a new duty cycle update every PWM period, as long as the minimum PWM period is observed.

Note that registers `SEQ[n].REFRESH` and `SEQ[n].ENDDELAY` are ignored when `DECODER.MODE=NextStep`. The next value is loaded upon every received `NEXTSTEP` task.

`SEQ[n].PTR` is the pointer used to fetch `COMPARE` values from RAM. If the `SEQ[n].PTR` is not pointing to a RAM region, an EasyDMA transfer may result in a `HardFault` or RAM corruption. See [Memory](#) on page 19 for more information about the different memory regions. After the `SEQ[n].PTR` is set to the desired RAM location, the `SEQ[n].MAXCNT` register must be set to the number of bytes in the sequence. It is important to observe that the Grouped mode requires one half word per group, while the Single mode requires one half word per channel, thus increasing the RAM size occupation. If PWM generation is not running when the `DMA.SEQ[n].START` task is triggered, the task will load the first value from RAM and then start the PWM generation. A `SEQSTARTED[n]` event is generated as soon as the EasyDMA has read the first PWM parameter from RAM and the wave counter has started executing it. When `LOOP.MAXCNT=0`, sequence $n=0$ or 1 is played back once. After the last value in the sequence has been loaded and started executing, a `SEQEND[n]` event is generated. The PWM generation will then continue with the output defined in the `IDLEOUT` register. The following figure illustrates an example of a simple playback.

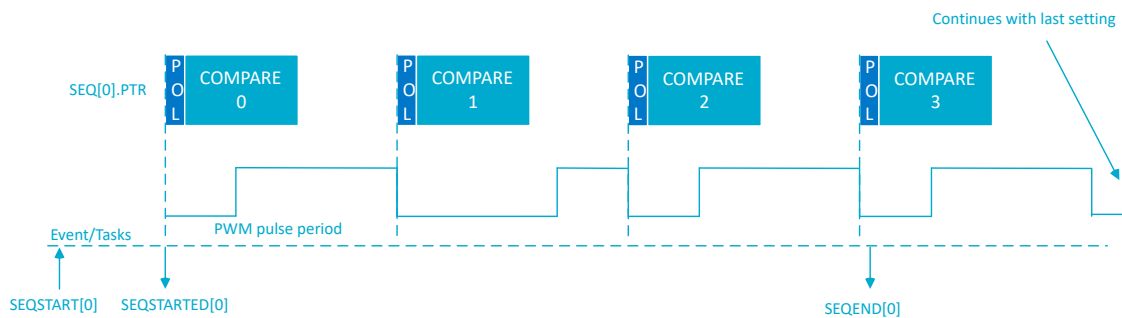


Figure 101: Simple sequence example

The following source code is used for configuration and timing details in a sequence where only sequence 0 is used and only run once with a new PWM duty cycle for each period.

```
NRF_PWM0->PSEL.OUT[0] = (first_port << PWM_PSEL_OUT_PORT_Pos) |
                        (first_pin << PWM_PSEL_OUT_PIN_Pos) |
                        (PWM_PSEL_OUT_CONNECT_Connected <<
                          PWM_PSEL_OUT_CONNECT_Pos);
NRF_PWM0->ENABLE      = (PWM_ENABLE_ENABLE_Enabled << PWM_ENABLE_ENABLE_Pos);
NRF_PWM0->MODE        = (PWM_MODE_UPDOWN_Up << PWM_MODE_UPDOWN_Pos);
NRF_PWM0->PRESCALER    = (PWM_PRESCALER_PRESCALER_DIV_1 <<
                          PWM_PRESCALER_PRESCALER_Pos);
NRF_PWM0->COUNTERTOP   = (16000 << PWM_COUNTERTOP_COUNTERTOP_Pos); //1 msec
NRF_PWM0->LOOP         = (PWM_LOOP_CNT_Disabled << PWM_LOOP_CNT_Pos);
NRF_PWM0->DECODER      = (PWM_DECODER_LOAD_Common << PWM_DECODER_LOAD_Pos) |
                          (PWM_DECODER_MODE_RefreshCount << PWM_DECODER_MODE_Pos);
NRF_PWM0->DMA.SEQ[0].PTR = ((uint32_t)(seq0_ram) << PWM_DMA_SEQ_PTR_PTR_Pos);
NRF_PWM0->DMA.SEQ[0].MAXCNT = (sizeof(seq0_ram) << PWM_DMA_SEQ_MAXCNT_MAXCNT_Pos);
NRF_PWM0->SEQ[0].REFRESH = 0;
NRF_PWM0->SEQ[0].ENDDELAY = 0;
NRF_PWM0->TASKS_DMA.SEQ[0].START = 1;
```

To completely stop the PWM generation and force the associated pins to a defined state, a STOP task can be triggered at any time. A STOPPED event is generated when the PWM generation has stopped at the end of the currently running PWM period, and the pins go into their idle state as defined by the IDLEOUT register. PWM generation can then only be restarted through a DMA.SEQ[n].START task. DMA.SEQ[n].START will resume PWM generation after having loaded the first value from the RAM buffer defined in the SEQ[n].PTR register.

The following table indicates when specific registers get sampled by the hardware. Care should be taken when updating these registers to avoid that values are applied earlier than expected.

Register	Taken into account by hardware	Recommended (safe) update
SEQ[n].PTR	When sending the DMA.SEQ[n].START task	After having received the SEQSTARTED[n] event
SEQ[n].MAXCNT	When sending the DMA.SEQ[n].START task	After having received the SEQSTARTED[n] event
SEQ[0].ENDELAY	When sending the SEQSTART[0] task	Before starting sequence [0] through a SEQSTART[0] task
	Every time a new value from sequence [0] has been loaded from RAM and gets applied to the Wave Counter (indicated by the PWMPERIODEND event)	When no more value from sequence [0] gets loaded from RAM (indicated by the SEQEND[0] event) At any time during sequence [1] (which starts when the SEQSTARTED[1] event is generated)
SEQ[1].ENDELAY	When sending the SEQSTART[1] task	Before starting sequence [1] through a SEQSTART[1] task
	Every time a new value from sequence [1] has been loaded from RAM and gets applied to the Wave Counter (indicated by the PWMPERIODEND event)	When no more value from sequence [1] gets loaded from RAM (indicated by the SEQEND[1] event) At any time during sequence [0] (which starts when the SEQSTARTED[0] event is generated)
SEQ[0].REFRESH	When sending the SEQSTART[0] task	Before starting sequence [0] through a SEQSTART[0] task
	Every time a new value from sequence [0] has been loaded from RAM and gets applied to the Wave Counter (indicated by the PWMPERIODEND event)	At any time during sequence [1] (which starts when the SEQSTARTED[1] event is generated)
SEQ[1].REFRESH	When sending the SEQSTART[1] task	Before starting sequence [1] through a SEQSTART[1] task
	Every time a new value from sequence [1] has been loaded from RAM and gets applied to the Wave Counter (indicated by the PWMPERIODEND event)	At any time during sequence [0] (which starts when the SEQSTARTED[0] event is generated)
COUNTERTOP	In DECODER.LOAD=WaveForm: this register is ignored. In all other LOAD modes: at the end of current PWM period (indicated by the PWMPERIODEND event)	Before starting PWM generation through a DMA.SEQ[n].START task After a STOP task has been triggered, and the STOPPED event has been received.
MODE	Immediately	Before starting PWM generation through a DMA.SEQ[n].START task After a STOP task has been triggered, and the STOPPED event has been received.
DECODER	Immediately	Before starting PWM generation through a DMA.SEQ[n].START task After a STOP task has been triggered, and the STOPPED event has been received.
PRESCALER	Immediately	Before starting PWM generation through a DMA.SEQ[n].START task After a STOP task has been triggered, and the STOPPED event has been received.
LOOP	Immediately	Before starting PWM generation through a DMA.SEQ[n].START task After a STOP task has been triggered, and the STOPPED event has been received.
PSEL.OUT[n]	Immediately	Before enabling the PWM instance through the ENABLE register

Table 45: When to safely update PWM registers

Note: SEQ[n].REFRESH and SEQ[n].ENDDELAY are ignored at the end of a complex sequence, indicated by a LOOPSDONE event. The reason for this is that the last value loaded from RAM is maintained until further action from software (restarting a new sequence, or stopping PWM generation).

The following figure shows a more complex example using the register `LOOP` on page 434.

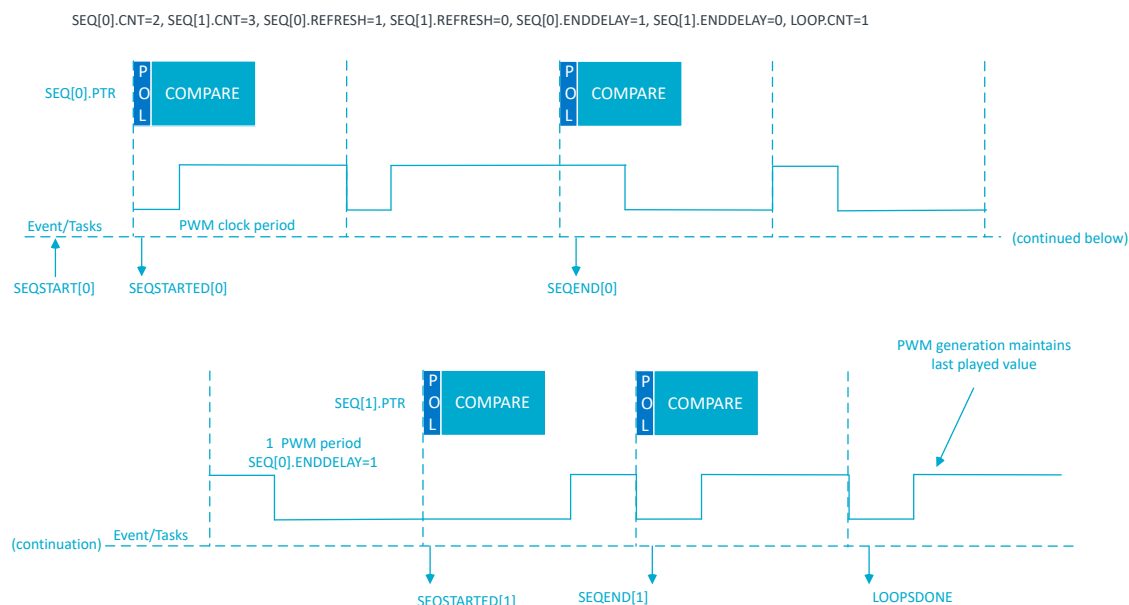


Figure 102: Example using two sequences

In this case, an automated playback takes place, consisting of SEQ[0], delay 0, SEQ[1], delay 1, then again SEQ[0], etc. The user can choose to start a complex playback with SEQ[0] or SEQ[1] through sending the SEQSTART[0] or SEQSTART[1] task. The complex playback always ends with delay 1.

The two sequences 0 and 1 are defined by the addresses of value tables in RAM (pointed to by SEQ[n].PTR) and the buffer size (SEQ[n].MAXCNT). The rate at which a new value is loaded is defined individually for each sequence by SEQ[n].REFRESH. The chaining of sequence 1 following the sequence 0 is implicit, the LOOP.CNT register allows the chaining of sequence 1 to sequence 0 for a determined number of times. In other words, it allows to repeat a complex sequence a number of times in a fully automated way.

In the following code example, sequence 0 is defined with SEQ[0].REFRESH set to 1, meaning that a new PWM duty cycle is pushed every second PWM period. This complex sequence is started with the SEQSTART[0] task, so SEQ[0] is played first. Since SEQ[0].ENDDELAY=1 there will be one PWM period delay between last period on sequence 0 and the first period on sequence 1. Since SEQ[1].ENDDELAY=0 there is no delay 1, so SEQ[0] would be started immediately after the end of SEQ[1]. However, as LOOP.CNT is

1, the playback stops after having played SEQ[1] only once, and both SEQEND[1] and LOOPSDONE are generated (their order is not guaranteed in this case).

```

NRF_PWM0->PSEL.OUT[0] = (first_port << PWM_PSEL_OUT_PORT_Pos) |
                        (first_pin << PWM_PSEL_OUT_PIN_Pos) |
                        (PWM_PSEL_OUT_CONNECT_Connected <<
                         PWM_PSEL_OUT_CONNECT_Pos);
NRF_PWM0->ENABLE      = (PWM_ENABLE_ENABLE_Enabled << PWM_ENABLE_ENABLE_Pos);
NRF_PWM0->MODE        = (PWM_MODE_UPDOWN_Up << PWM_MODE_UPDOWN_Pos);
NRF_PWM0->PRESCALER   = (PWM_PRESCALER_PRESCALER_DIV_1 <<
                         PWM_PRESCALER_PRESCALER_Pos);
NRF_PWM0->COUNTERTOP  = (16000 << PWM_COUNTERTOP_COUNTERTOP_Pos); //1 msec
NRF_PWM0->LOOP        = (1 << PWM_LOOP_CNT_Pos);
NRF_PWM0->DECODER     = (PWM_DECODER_LOAD_Common << PWM_DECODER_LOAD_Pos) |
                        (PWM_DECODER_MODE_RefreshCount << PWM_DECODER_MODE_Pos);
NRF_PWM0->DMA.SEQ[0].PTR   = ((uint32_t)(seq0_ram) << PWM_DMA_SEQ_PTR_PTR_Pos);
NRF_PWM0->DMA.SEQ[0].MAXCNT = (sizeof(seq0_ram) << PWM_DMA_SEQ_MAXCNT_MAXCNT_Pos);
NRF_PWM0->SEQ[0].REFRESH  = 1;
NRF_PWM0->SEQ[0].ENDDELAY = 1;
NRF_PWM0->DMA.SEQ[1].PTR   = ((uint32_t)(seq1_ram) << PWM_DMA_SEQ_PTR_PTR_Pos);
NRF_PWM0->DMA.SEQ[1].MAXCNT = (sizeof(seq1_ram) << PWM_DMA_SEQ_MAXCNT_MAXCNT_Pos);
NRF_PWM0->SEQ[1].REFRESH  = 0;
NRF_PWM0->SEQ[1].ENDDELAY = 0;
NRF_PWM0->TASKS_DMA.SEQ[0].START = 1;

```

The decoder can also be configured to asynchronously load new PWM duty cycle. If the DECODER.MODE register is set to NextStep, then the NEXTSTEP task will cause an update of internal compare registers on the next PWM period.

The following figures provide an overview of each part of an arbitrary sequence, in various modes (LOOP.CNT=0 and LOOP.CNT>0). In particular, the following are represented:

- Initial and final duty cycle on the PWM output(s)
- Chaining of SEQ[0] and SEQ[1] if LOOP.CNT>0
- Influence of registers on the sequence
- Events generated during a sequence
- DMA activity (loading of next value and applying it to the output(s))

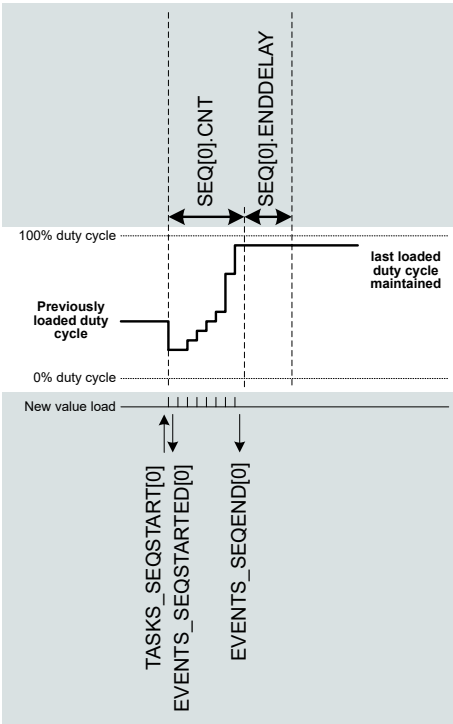


Figure 103: Single shot (LOOP.CNT=0)

Note: The single-shot example also applies to SEQ[1]. Only SEQ[0] is represented for simplicity.

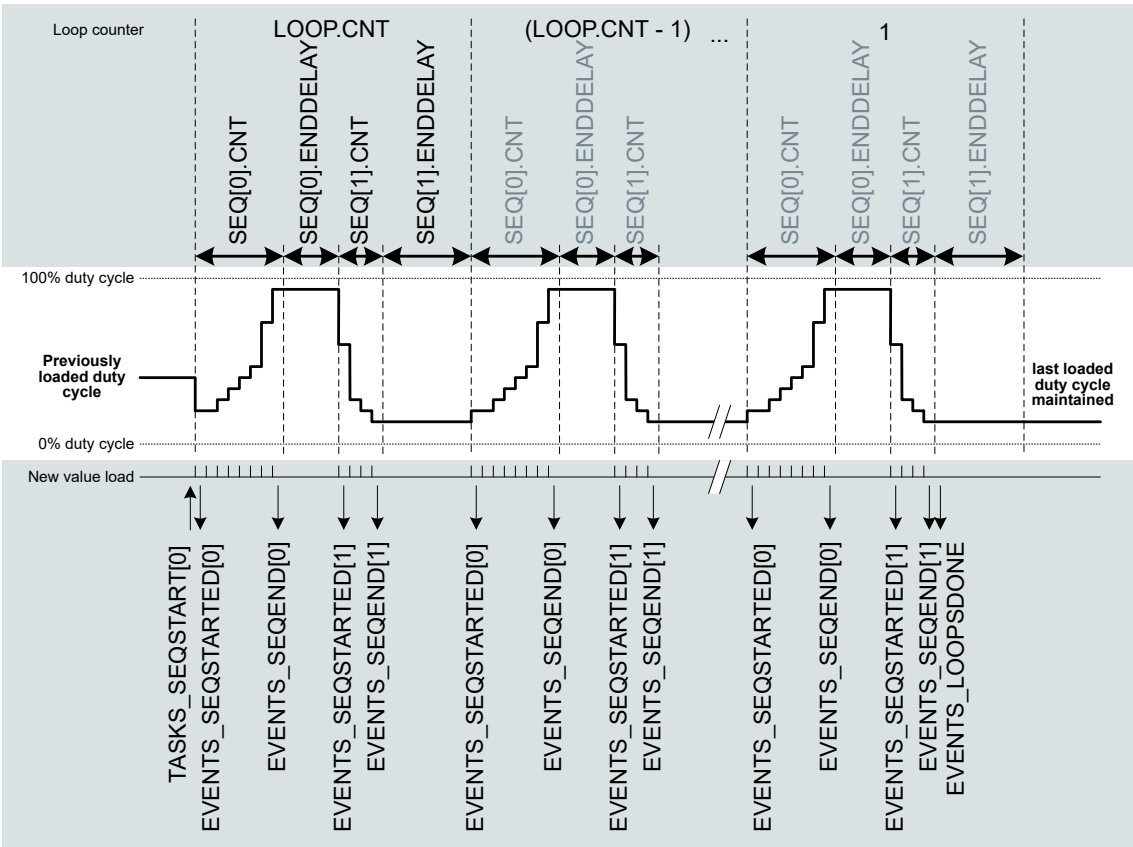


Figure 104: Complex sequence (LOOP.CNT>0) starting with SEQ[0]

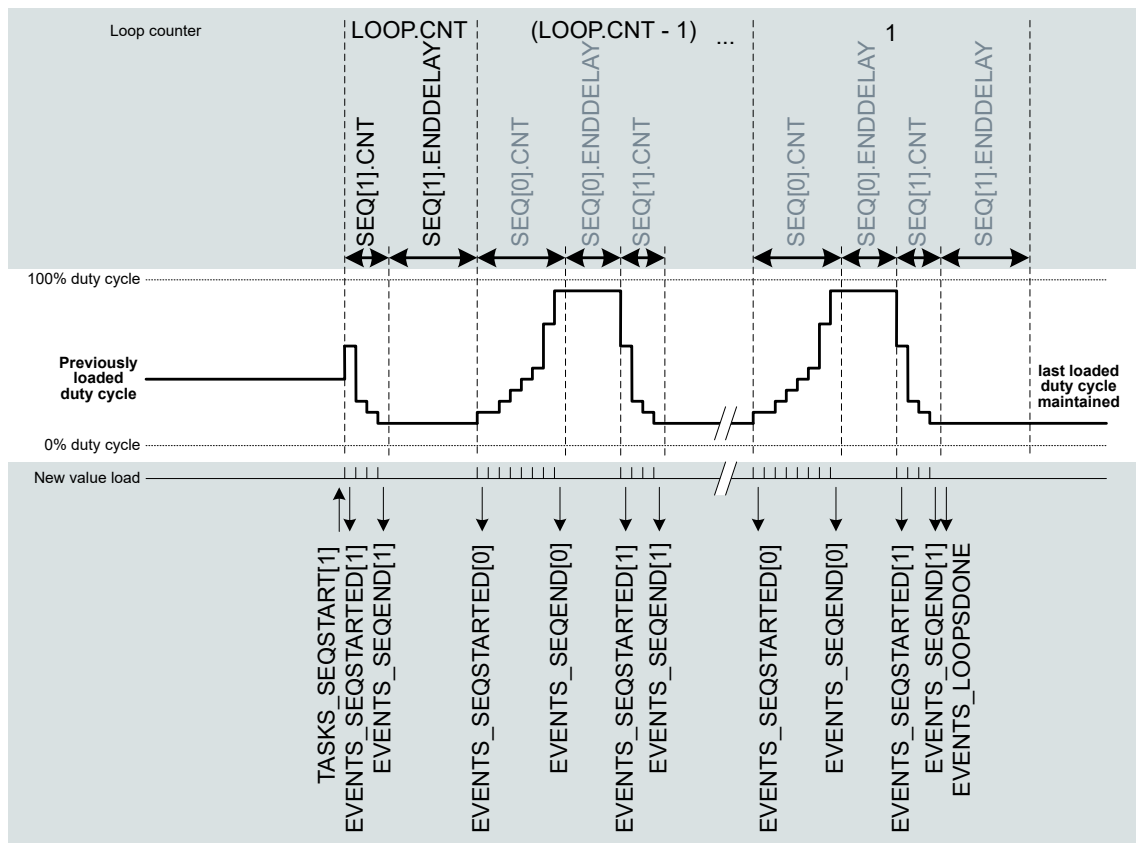


Figure 105: Complex sequence ($LOOP.CNT > 0$) starting with $SEQ[1]$

Note: If a sequence is in use in a simple or complex sequence, it must have a length of $SEQ[n].MAXCNT > 0$.

This example shows how the PWM module can be configured to repeat a single sequence until stopped.

```
NRF_PWM0->PSEL.OUT[0] = (first_port << PWM_PSEL_OUT_PORT_Pos) |
                          (first_pin << PWM_PSEL_OUT_PIN_Pos) |
                          (PWM_PSEL_OUT_CONNECT_Connected <<
                           PWM_PSEL_OUT_CONNECT_Pos);

NRF_PWM0->ENABLE        = (PWM_ENABLE_ENABLE_Enabled << PWM_ENABLE_ENABLE_Pos);
NRF_PWM0->MODE          = (PWM_MODE_UPDOWN_Up << PWM_MODE_UPDOWN_Pos);
NRF_PWM0->PRESCALER     = (PWM_PRESCALER_PRESCALER_DIV_1 <<
                           PWM_PRESCALER_PRESCALER_Pos);

NRF_PWM0->COUNTERTOP    = (16000 << PWM_COUNTERTOP_COUNTERTOP_Pos); //1 msec
// Enable the shortcut from LOOPSDONE event to DMA.SEQ1.START task for infinite loop
NRF_PWM0->SHORTS        = (PWM_SHORTS_LOOPSDONE_DMA_SEQ1_START_Enabled <<
                           PWM_SHORTS_LOOPSDONE_DMA_SEQ1_START_Pos);

// LOOP_CNT must be greater than 0 for the LOOPSDONE event to trigger and enable looping
NRF_PWM0->LOOP          = (1 << PWM_LOOP_CNT_Pos);
NRF_PWM0->DECODER       = (PWM_DECODER_LOAD_Common << PWM_DECODER_LOAD_Pos) |
                          (PWM_DECODER_MODE_RefreshCount << PWM_DECODER_MODE_Pos);

// To repeat a single sequence until stopped, it must be configured in SEQ[1]
NRF_PWM0->DMA.SEQ[1].PTR = ((uint32_t)(seq0_ram)) << PWM_DMA_SEQ_PTR_PTR_Pos;
NRF_PWM0->DMA.SEQ[1].MAXCNT = (sizeof(seq0_ram) << PWM_DMA_SEQ_MAXCNT_MAXCNT_Pos);
NRF_PWM0->SEQ[1].REFRESH = 0;
NRF_PWM0->SEQ[1].ENDDelay = 0;
NRF_PWM0->TASKS_DMA.SEQ[1].START = 1;
```

8.15.3 Limitations

The previous compare value is repeated if the PWM period is shorter than the time it takes for the EasyDMA to retrieve from RAM and update the internal compare registers. This is to ensure a glitch-free operation even for very short PWM periods.

Only SEQ[1] can trigger the **LOOPSDONE** event upon completion, not SEQ[0]. This requires looping to be enabled (**LOOP** > 0) and SEQ[1].MAXCNT > 0 when sequence playback starts.

8.15.4 Pin configuration

The OUT[n] (n=0..3) signals associated with each PWM channel are mapped to physical pins according to the configuration of PSEL.OUT[n] registers. If PSEL.OUT[n].CONNECT is set to Disconnected, the associated PWM module signal will not be connected to any physical pins.

Once PWM has been enabled, the PSEL.OUT[n] registers take effect and PWM generation starts from the IDLEOUT register. PWM can then be started and sequences generated.

To ensure correct behavior in the PWM module, the pins that are used must be configured in the GPIO peripheral in the following way before the PWM module is enabled:

PWM signal	PWM pin	Direction	Output value	Comment
OUT[n]	As specified in PSEL.OUT[n] (n=0..3)	Output	0	Idle state defined in GPIO OUT register and the IDLEOUT register

Table 46: Recommended GPIO configuration before starting PWM generation

The idle state of a pin is defined by the OUT register in the GPIO module and the IDLEOUT register, to ensure that the pins used by the PWM module are driven correctly. Both OUT register in the GPIO module and the IDLEOUT register should be set with same value for each PWM channel before enabling the PWM module. When PWM is disabled using the ENABLE register the PWM module stops controlling the GPIO pins, and the corresponding pins are then controlled by the GPIO peripheral.

Only one peripheral can be assigned to drive a particular GPIO pin at a time. Failing to do so may result in unpredictable behavior.

8.15.5 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
PWM20 : S	GLOBAL	0x500D2000	US	S	SA	No	Pulse width modulation unit PWM20
PWM20 : NS		0x400D2000					
PWM21 : S	GLOBAL	0x500D3000	US	S	SA	No	Pulse width modulation unit PWM21
PWM21 : NS		0x400D3000					
PWM22 : S	GLOBAL	0x500D4000	US	S	SA	No	Pulse width modulation unit PWM22
PWM22 : NS		0x400D4000					

Configuration

Instance	Domain	Configuration
PWM20 : S PWM20 : NS	GLOBAL	Use GPIO port P1 IDLEOUT register is available. EVENTS_COMPAREMATCH events are available. CURRENTAMOUNT register included.
PWM21 : S PWM21 : NS	GLOBAL	Use GPIO port P1 IDLEOUT register is available. EVENTS_COMPAREMATCH events are available. CURRENTAMOUNT register included.
PWM22 : S PWM22 : NS	GLOBAL	Use GPIO port P1 IDLEOUT register is available. EVENTS_COMPAREMATCH events are available. CURRENTAMOUNT register included.

Register overview

Register	Offset	TZ	Description
TASKS_STOP	0x004		Stops PWM pulse generation on all channels at the end of current PWM period, and stops sequence playback
TASKS_NEXTSTEP	0x008		Steps by one value in the current sequence on all enabled channels if DECODER.MODE=NextStep. Does not cause PWM generation to start if not running.
TASKS_DMA.SEQ[n].START	0x010		Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.
TASKS_DMA.SEQ[n].STOP	0x014		Stops operation using easyDMA. This does not trigger an END event.
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
SUBSCRIBE_NEXTSTEP	0x088		Subscribe configuration for task NEXTSTEP
SUBSCRIBE_DMA.SEQ[n].START	0x090		Subscribe configuration for task START
SUBSCRIBE_DMA.SEQ[n].STOP	0x094		Subscribe configuration for task STOP
EVENTS_STOPPED	0x104		Response to STOP task, emitted when PWM pulses are no longer generated
EVENTS_SEQSTARTED[n]	0x108		First PWM period started on sequence n
EVENTS_SEQEND[n]	0x110		Emitted at end of every sequence n, when last value from RAM has been applied to wave counter
EVENTS_PWMPERIODEND	0x118		Emitted at the end of each PWM period
EVENTS_LOOPSDONE	0x11C		Concatenated sequences have been played the amount of times defined in LOOP.CNT
EVENTS_RAMUNDERFLOW	0x120		Emitted when retrieving from RAM does not complete in time for the PWM module
EVENTS_DMA.SEQ[n].END	0x124		Generated after all MAXCNT bytes have been transferred
EVENTS_DMA.SEQ[n].READY	0x128		Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.
EVENTS_DMA.SEQ[n].BUSERROR	0x12C		An error occurred during the bus transfer.
EVENTS_COMPAREMATCH[n]	0x13C		This event is generated when the compare matches for the compare channel [n].
PUBLISH_STOPPED	0x184		Publish configuration for event STOPPED
PUBLISH_SEQSTARTED[n]	0x188		Publish configuration for event SEQSTARTED[n]
PUBLISH_SEQEND[n]	0x190		Publish configuration for event SEQEND[n]
PUBLISH_PWMPERIODEND	0x198		Publish configuration for event PWMPERIODEND
PUBLISH_LOOPSDONE	0x19C		Publish configuration for event LOOPSDONE
PUBLISH_RAMUNDERFLOW	0x1A0		Publish configuration for event RAMUNDERFLOW
PUBLISH_DMA.SEQ[n].END	0x1A4		Publish configuration for event END
PUBLISH_DMA.SEQ[n].READY	0x1A8		Publish configuration for event READY

Register	Offset	TZ	Description
PUBLISH_DMA.SEQ[n].BUSERROR	0x1AC		Publish configuration for event BUSERROR
PUBLISH_COMPAREMATCH[n]	0x1BC		Publish configuration for event COMPAREMATCH[n]
SHORTS	0x200		Shortcuts between local events and tasks
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
INTPEND	0x30C		Pending interrupts
ENABLE	0x500		PWM module enable register
MODE	0x504		Selects operating mode of the wave counter
COUNTERTOP	0x508		Value up to which the pulse generator counter counts
PRESCALER	0x50C		Configuration for PWM_CLK
DECODER	0x510		Configuration of the decoder
LOOP	0x514		Number of playbacks of a loop
IDLEOUT	0x518		Configure the output value on the PWM channel during idle
SEQ[n].REFRESH	0x528		Number of additional PWM periods between samples loaded into compare register
SEQ[n].ENDDELAY	0x52C		Time added after the sequence
PSEL.OUT[n]	0x560		Output pin select for PWM channel n
DMA.SEQ[n].PTR	0x704		RAM buffer start address
DMA.SEQ[n].MAXCNT	0x708		Maximum number of bytes in channel buffer
DMA.SEQ[n].AMOUNT	0x70C		Number of bytes transferred in the last transaction, updated after the END event.
DMA.SEQ[n].CURRENTAMOUNT	0x710		Number of bytes transferred in the current transaction
DMA.SEQ[n].TERMINATEONBUSERROR	0x71C		Terminate the transaction if a BUSERROR event is detected.
DMA.SEQ[n].BUSERRORADDRESS	0x720		Address of transaction that generated the last BUSERROR event.

8.15.5.1 TASKS_STOP

Address offset: 0x004

Stops PWM pulse generation on all channels at the end of current PWM period, and stops sequence playback

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A																															
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value	ID	Value	Description																														
A	W	TASKS_STOP				Stops PWM pulse generation on all channels at the end of current PWM period, and stops sequence playback																														
			Trigger	1		Trigger task																														

8.15.5.2 TASKS_NEXTSTEP

Address offset: 0x008

Steps by one value in the current sequence on all enabled channels if DECODER.MODE=NextStep. Does not cause PWM generation to start if not running.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value	ID	Value	Description																													
A	W	TASKS_NEXTSTEP				Steps by one value in the current sequence on all enabled channels if DECODER.MODE=NextStep. Does not cause PWM generation to start if not running.																													
			Trigger	1		Trigger task																													

8.15.5.3 TASKS_DMA

Peripheral tasks.

8.15.5.3.1 TASKS_DMA.SEQ[n] (n=0..1)

Peripheral tasks.

8.15.5.3.1.1 TASKS_DMA.SEQ[n].START (n=0..1)

Address offset: $0x010 + (n \times 0x8)$

Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	W	START			Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.																																	
		Trigger		1	Trigger task																																	

8.15.5.3.1.2 TASKS_DMA.SEQ[n].STOP (n=0..1)

Address offset: $0x014 + (n \times 0x8)$

Stops operation using easyDMA. This does not trigger an END event.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	STOP			Stops operation using easyDMA. This does not trigger an END event.																															
		Trigger		1	Trigger task																															

8.15.5.4 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task STOP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																			
ID				B																																A A A A A A A A			
Reset 0x00000000				0 0																																			
ID	R/W	Field	Value ID	Value				Description																															
A	RW	CHIDX		[0..255]				DPPI channel that task STOP will subscribe to																															
B	RW	EN																																					
			Disabled	0				Disable subscription																															
			Enabled	1				Enable subscription																															

8.15.5.5 SUBSCRIBE_NEXTSTEP

Address offset: 0x088

Subscribe configuration for task NEXTSTEP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task NEXTSTEP will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.15.5.6 SUBSCRIBE_DMA

Subscribe configuration for tasks

8.15.5.6.1 SUBSCRIBE_DMA.SEQ[n] (n=0..1)

Subscribe configuration for tasks

8.15.5.6.1.1 SUBSCRIBE_DMA.SEQ[n].START (n=0..1)

Address offset: 0x090 + (n × 0x8)

Subscribe configuration for task **START**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task START will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.15.5.6.1.2 SUBSCRIBE_DMA.SEQ[n].STOP (n=0..1)

Address offset: 0x094 + (n × 0x8)

Subscribe configuration for task **STOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																												A A A A A A A A			
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task STOP will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.15.5.7 EVENTS_STOPPED

Address offset: 0x104

Response to STOP task, emitted when PWM pulses are no longer generated

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_STOPPED			Response to STOP task, emitted when PWM pulses are no longer generated																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.15.5.8 EVENTS_SEQSTARTED[n] (n=0..1)

Address offset: 0x108 + (n × 0x4)

First PWM period started on sequence n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_SEQSTARTED						First PWM period started on sequence n																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.15.5.9 EVENTS_SEQEND[n] (n=0..1)

Address offset: 0x110 + (n × 0x4)

Emitted at end of every sequence n, when last value from RAM has been applied to wave counter

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_SEQEND			Emitted at end of every sequence n, when last value from RAM has been applied to wave counter																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.15.5.10 EVENTS_PWMPERIODEND

Address offset: 0x118

Emitted at the end of each PWM period

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_PWMPERIODEND						Emitted at the end of each PWM period																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.15.5.11 EVENTS_LOOPSDONE

Address offset: 0x11C

Concatenated sequences have been played the amount of times defined in LOOP.CNT

This event triggers after the last SEQ[1] completion of the loop, and only if looping was enabled (LOOP > 0) when the sequence playback was started.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																				A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	EVENTS_LOOPS	DONE			Concatenated sequences have been played the amount of times defined in LOOP.CNT																															
						This event triggers after the last SEQ[1] completion of the loop, and only if looping was enabled (LOOP > 0) when the sequence playback was started.																															
				NotGenerated	0	Event not generated																															
				Generated	1	Event generated																															

8.15.5.12 EVENTS_RAMUNDERFLOW

Address offset: 0x120

Emitted when retrieving from RAM does not complete in time for the PWM module

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID										A																																		
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																							
A	RW	EVENTS_RAMUNDERFLOW			Emitted when retrieving from RAM does not complete in time for the PWM module																																							
			NotGenerated	0	Event not generated																																							
			Generated	1	Event generated																																							

8.15.5.13 EVENTS_DMA

Peripheral events.

8.15.5.13.1 EVENTS_DMA.SEQ[n] (n=0..1)

Peripheral events.

8.15.5.13.1.1 EVENTS_DMA.SEQ[n].END (n=0..1)

Address offset: 0x124 + (n × 0xC)

Generated after all MAXCNT bytes have been transferred

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	END						Generated after all MAXCNT bytes have been transferred																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.15.5.13.1.2 EVENTS_DMA.SEQ[n].READY (n=0..1)

Address offset: 0x128 + (n × 0xC)

Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.15.5.13.1.3 EVENTS_DMA.SEQ[n].BUSERROR (n=0..1)

Address offset: 0x12C + (n × 0xC)

An error occurred during the bus transfer.

When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	BUSERROR			An error occurred during the bus transfer.																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.15.5.14 EVENTS_COMPAREMATCH[n] (n=0..3)

Address offset: 0x13C + (n × 0x4)

This event is generated when the compare matches for the compare channel [n].

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_COMPAREMATCH			This event is generated when the compare matches for the compare channel [n].																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.15.5.15 PUBLISH_STOPPED

Address offset: 0x184

Publish configuration for event STOPPED

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event STOPPED will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.15.5.16 PUBLISH_SEQSTARTED[n] (n=0..1)

Address offset: 0x188 + (n × 0x4)

Publish configuration for event **SEQSTARTED[n]**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																													
ID				B																								A												A	A	A	A	A	A	A																		
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																										
A	RW	CHIDX		[0..255]		DPPI channel that event SEQSTARTED[n] will publish to																																																										
B	RW	EN																																																														
			Disabled	0	Disable publishing																																																											
			Enabled	1	Enable publishing																																																											

8.15.5.17 PUBLISH_SEQEND[n] (n=0..1)

Address offset: 0x190 + (n × 0x4)

Publish configuration for event **SEQEND[n]**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																													
ID				B																								A												A	A	A	A	A	A	A																		
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																										
A	RW	CHIDX		[0..255]		DPPI channel that event SEQEND[n] will publish to																																																										
B	RW	EN																																																														
			Disabled	0	Disable publishing																																																											
			Enabled	1	Enable publishing																																																											

8.15.5.18 PUBLISH_PWMPERIODEND

Address offset: 0x198

Publish configuration for event **PWMPERIODEND**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that event PWMPERIODEND will publish to																																
B	RW	EN																																				
			Disabled	0	Disable publishing																																	
			Enabled	1	Enable publishing																																	

8.15.5.19 PUBLISH_LOOPSDONE

Address offset: 0x19C

Publish configuration for event **LOOPSDONE**

This event triggers after the last SEQ[1] completion of the loop, and only if looping was enabled (LOOP > 0) when the sequence playback was started.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				B																								A					A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
ID	R/W	Field	Value ID	Value	Description																																		
A	RW	CHIDX		[0..255]	DPPI channel that event LOOPSDONE will publish to																																		
B	RW	EN																																					
			Disabled	0	Disable publishing																																		
			Enabled	1	Enable publishing																																		

8.15.5.20 PUBLISH_RAMUNDERFLOW

Address offset: 0x1A0

Publish configuration for event **RAMUNDERFLOW**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0							
ID				B																								A					A		A		A		A		A	
Reset 0x00000000				0																																						
ID	R/W	Field	Value ID	Value		Description																																				
A	RW	CHIDX		[0..255]		DPPI channel that event RAMUNDERFLOW will publish to																																				
B	RW	EN																																								
			Disabled	0	Disable publishing																																					
			Enabled	1	Enable publishing																																					

8.15.5.21 PUBLISH_DMA

Publish configuration for events

8.15.5.21.1 PUBLISH_DMA.SEQ[n] (n=0..1)

Publish configuration for events

8.15.5.21.1.1 PUBLISH_DMA.SEQ[n].END (n=0..1)

Address offset: 0x1A4 + (n × 0xC)

Publish configuration for event **END**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event END will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.15.5.21.1.2 PUBLISH_DMA.SEQ[n].READY (n=0..1)Address offset: $0x1A8 + (n \times 0xC)$ Publish configuration for event **READY**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				B																								A				A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	CHIDX		[0..255]		DPPI channel that event READY will publish to																															
B	RW	EN																																			
			Disabled	0	Disable publishing																																
			Enabled	1	Enable publishing																																

8.15.5.21.1.3 PUBLISH_DMA.SEQ[n].BUSERROR (n=0..1)Address offset: $0x1AC + (n \times 0xC)$ Publish configuration for event **BUSERROR**

When this event is generated, the address which caused the error can be read from the **BUSERRORADDRESS** register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0												
ID				B																								A				A				A				A				A			
Reset 0x00000000				0 0																																											
ID	R/W	Field	Value ID	Value		Description																																									
A	RW	CHIDX		[0..255]		DPPI channel that event BUSERROR will publish to																																									
B	RW	EN																																													
			Disabled	0	Disable publishing																																										
			Enabled	1	Enable publishing																																										

8.15.5.22 PUBLISH_COMPAREMATCH[n] (n=0..3)Address offset: $0x1BC + (n \times 0x4)$ Publish configuration for event **COMPAREMATCH[n]**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																								
ID				B																								A				A				A				A				A																															
Reset 0x00000000				0																																0				0				0				0				0				0				0				0				0				0			
ID	R/W	Field	Value ID	Value				Description																																																																			
A	RW	CHIDX		[0..255]				DPPI channel that event COMPAREMATCH[n] will publish to																																																																			
B	RW	EN																																																																									
			Disabled	0				Disable publishing																																																																			
			Enabled	1				Enable publishing																																																																			

8.15.5.23 SHORTSAddress offset: $0x200$

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-B	RW	SEQEND[i]_STOP (i=0..1)			Shortcut between event SEQEND[n] and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
C-D	RW	LOOPSDONE_DMA_SEQ[i]_START (i=0..1)			Shortcut between event LOOPSDONE and task DMA.SEQ[n].START																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
E	RW	LOOPSDONE_STOP			Shortcut between event LOOPSDONE and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
F	RW	RAMUNDERFLOW_STOP			Shortcut between event RAMUNDERFLOW and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
G-H	RW	DMA_SEQ[i]_BUSERROR_STOP (i=0..1)			Shortcut between event DMA.SEQ[n].BUSERROR and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														

8.15.5.24 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	STOPPED			Enable or disable interrupt for event STOPPED																													
			Disabled	0	Disable																													
			Enabled	1	Enable																													
B-C	RW	SEQSTARTED[i] (i=0..1)			Enable or disable interrupt for event SEQSTARTED[i]																													
			Disabled	0	Disable																													
			Enabled	1	Enable																													
D-E	RW	SEQEND[i] (i=0..1)			Enable or disable interrupt for event SEQEND[i]																													
			Disabled	0	Disable																													
			Enabled	1	Enable																													
F	RW	PWMPERIODEND			Enable or disable interrupt for event PWMPERIODEND																													
			Disabled	0	Disable																													
			Enabled	1	Enable																													
G	RW	LOOPSDONE			Enable or disable interrupt for event LOOPSDONE																													
					This event triggers after the last SEQ[1] completion of the loop, and only if looping was enabled (LOOP > 0) when the sequence playback was started.																													
			Disabled	0	Disable																													
			Enabled	1	Enable																													
			H	RW	RAMUNDERFLOW			Enable or disable interrupt for event RAMUNDERFLOW																										
						Disabled	0	Disable																										
Enabled	1	Enable																																
I	RW	DMASEQ0END			Enable or disable interrupt for event DMASEQ0END																													
			Disabled	0	Disable																													
			Enabled	1	Enable																													
J	RW	DMASEQ0READY			Enable or disable interrupt for event DMASEQ0READY																													

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
						Enable or disable interrupt for event DMASEQ0BUSERROR																													
						When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																													
			Disabled	0	Disable																														
			Enabled	1	Enable																														
L	RW	DMASEQ1END				Enable or disable interrupt for event DMASEQ1END																													
			Disabled	0	Disable																														
			Enabled	1	Enable																														
M	RW	DMASEQ1READY				Enable or disable interrupt for event DMASEQ1READY																													
			Disabled	0	Disable																														
			Enabled	1	Enable																														
N	RW	DMASEQ1BUSERROR				Enable or disable interrupt for event DMASEQ1BUSERROR																													
						When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																													
			Disabled	0	Disable																														
			Enabled	1	Enable																														
O-R	RW	COMPAREMATCH[i] (i=0..3)				Enable or disable interrupt for event COMPAREMATCH[i]																													
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.15.5.25 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	STOPPED			Write '1' to enable interrupt for event STOPPED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B-C	RW	SEQSTARTED[i] (i=0..1)			Write '1' to enable interrupt for event SEQSTARTED[i]																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D-E	RW	SEQEND[i] (i=0..1)			Write '1' to enable interrupt for event SEQEND[i]																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
F	RW	PWMPERIODEND			Write '1' to enable interrupt for event PWMPERIODEND																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
G	RW	LOOPSDONE			Write '1' to enable interrupt for event LOOPSDONE																													
					This event triggers after the last SEQ[1] completion of the loop, and only if looping was enabled (LOOP > 0) when the sequence playback was started.																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
		Enabled	1	Read: Enabled																														
H	RW	RAMUNDERFLOW			Write '1' to enable interrupt for event RAMUNDERFLOW																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
I	RW	DMASEQ0END			Write '1' to enable interrupt for event DMASEQ0END																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
J	RW	DMASEQ0READY			Write '1' to enable interrupt for event DMASEQ0READY																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
K	RW	DMASEQ0BUSERROR			Write '1' to enable interrupt for event DMASEQ0BUSERROR																													
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
		Enabled	1	Read: Enabled																														
L	RW	DMASEQ1END			Write '1' to enable interrupt for event DMASEQ1END																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
M	RW	DMASEQ1READY			Write '1' to enable interrupt for event DMASEQ1READY																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
N	RW	DMASEQ1BUSERROR			Write '1' to enable interrupt for event DMASEQ1BUSERROR																													
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
		Enabled	1	Read: Enabled																														
O-R	RW	COMPAREMATCH[i] (i=0..3)			Write '1' to enable interrupt for event COMPAREMATCH[i]																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

8.15.5.26 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	STOPPED			Write '1' to disable interrupt for event STOPPED																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B-C	RW	SEQSTARTED[i] (i=0..1)			Write '1' to disable interrupt for event SEQSTARTED[i]																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D-E	RW	SEQEND[i] (i=0..1)			Write '1' to disable interrupt for event SEQEND[i]																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
F	RW	PWMPERIODEND			Write '1' to disable interrupt for event PWMPERIODEND																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
G	RW	LOOPSDONE			Write '1' to disable interrupt for event LOOPSDONE																														
					This event triggers after the last SEQ[1] completion of the loop, and only if looping was enabled (LOOP > 0) when the sequence playback was started.																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
H	RW	RAMUNDERFLOW			Write '1' to disable interrupt for event RAMUNDERFLOW																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
I	RW	DMASEQ0END			Write '1' to disable interrupt for event DMASEQ0END																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
J	RW	DMASEQ0READY			Write '1' to disable interrupt for event DMASEQ0READY																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
K	RW	DMASEQ0BUSERROR			Write '1' to disable interrupt for event DMASEQ0BUSERROR																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
L	RW	DMASEQ1END			Write '1' to disable interrupt for event DMASEQ1END																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
M	RW	DMASEQ1READY			Write '1' to disable interrupt for event DMASEQ1READY																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
N	RW	DMASEQ1BUSERROR			Write '1' to disable interrupt for event DMASEQ1BUSERROR																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
O-R	RW	COMPAREMATCH[i] (i=0..3)			Write '1' to disable interrupt for event COMPAREMATCH[i]																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.15.5.27 INTPEND

Address offset: 0x30C

Pending interrupts

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	STOPPED			Read pending status of interrupt for event STOPPED																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
B-C	R	SEQSTARTED[i] (i=0..1)			Read pending status of interrupt for event SEQSTARTED[i]																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
D-E	R	SEQEND[i] (i=0..1)			Read pending status of interrupt for event SEQEND[i]																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
F	R	PWMPERIODEND			Read pending status of interrupt for event PWMPERIODEND																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
G	R	LOOPSDONE			Read pending status of interrupt for event LOOPSDONE																														
					This event triggers after the last SEQ[1] completion of the loop, and only if looping was enabled (LOOP > 0) when the sequence playback was started.																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
H	R	RAMUNDERFLOW			Read pending status of interrupt for event RAMUNDERFLOW																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
I	R	DMASEQ0END			Read pending status of interrupt for event DMASEQ0END																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
J	R	DMASEQ0READY			Read pending status of interrupt for event DMASEQ0READY																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
K	R	DMASEQ0BUSERROR			Read pending status of interrupt for event DMASEQ0BUSERROR																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
L	R	DMASEQ1END			Read pending status of interrupt for event DMASEQ1END																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
M	R	DMASEQ1READY			Read pending status of interrupt for event DMASEQ1READY																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
N	R	DMASEQ1BUSERROR			Read pending status of interrupt for event DMASEQ1BUSERROR																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
O-R	R	COMPAREMATCH[i] (i=0..3)			Read pending status of interrupt for event COMPAREMATCH[i]																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														

8.15.5.28 ENABLE

Address offset: 0x500

PWM module enable register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE			Enable or disable PWM module																														
			Disabled	0	Disabled																														
			Enabled	1	Enable																														

8.15.5.29 MODE

Address offset: 0x504

Selects operating mode of the wave counter

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	UPDOWN			Selects up mode or up-and-down mode for the counter																														
			Up	0	Up counter, edge-aligned PWM duty cycle																														
			UpAndDown	1	Up and down counter, center-aligned PWM duty cycle																														

8.15.5.30 COUNTERTOP

Address offset: 0x508

Value up to which the pulse generator counter counts

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																	A A A A A A A A A A A A A A A A															
Reset 0x000003FF	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															
ID	R/W	Field	Value ID	Value	Description																											
A	RW	COUNTERTOP		[3..32767]	Value up to which the pulse generator counter counts. This register is ignored when DECODER.MODE=WaveForm and only values from RAM are used.																											

8.15.5.31 PRESCALER

Address offset: 0x50C

Configuration for PWM_CLK

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	RW	PRESCALER																		Prescaler of PWM_CLK															
			DIV_1	0																Divide by 1 (16 MHz)															
			DIV_2	1																Divide by 2 (8 MHz)															
			DIV_4	2																Divide by 4 (4 MHz)															
			DIV_8	3																Divide by 8 (2 MHz)															
			DIV_16	4																Divide by 16 (1 MHz)															
			DIV_32	5																Divide by 32 (500 kHz)															
			DIV_64	6																Divide by 64 (250 kHz)															
			DIV_128	7																Divide by 128 (125 kHz)															

8.15.5.32 DECODER

Address offset: 0x510

Configuration of the decoder

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	LOAD			How a sequence is read from RAM and spread to the compare register																														
			Common	0	1st half word (16-bit) used in all PWM channels 0..3																														
			Grouped	1	1st half word (16-bit) used in channel 0..1; 2nd word in channel 2..3																														
			Individual	2	1st half word (16-bit) in ch.0; 2nd in ch.1; ...; 4th in ch.3																														
			WaveForm	3	1st half word (16-bit) in ch.0; 2nd in ch.1; ...; 4th in COUNTERTOP																														
B	RW	MODE			Selects source for advancing the active sequence																														
			RefreshCount	0	SEQ[n].REFRESH is used to determine loading internal compare registers																														
			NextStep	1	NEXTSTEP task causes a new value to be loaded to internal compare registers																														

8.15.5.33 LOOP

Address offset: 0x514

Number of playbacks of a loop

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	CNT			Number of playbacks of pattern cycles																															
			Disabled	0	Looping disabled (stop at the end of the sequence)																															

8.15.5.34 IDLEOUT

Address offset: 0x518

Configure the output value on the PWM channel during idle

Writes to this register are ignored when the PWM is enabled.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																																					D	C	B	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value	Description																																			
A-D	RW	VAL[i] (i=0..3)			Idle output value for PWM channel [i]																																			

8.15.5.35 SEQ[n].REFRESH (n=0..1)

Address offset: 0x528 + (n × 0x20)

Number of additional PWM periods between samples loaded into compare register

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A																															
Reset 0x00000001					0 1																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	CNT			Number of additional PWM periods between samples loaded into compare register (load every REFRESH.CNT+1 PWM periods)																															
			Continuous	0	Update every PWM period																															

8.15.5.36 SEQ[n].ENDDELAY (n=0..1)

Address offset: 0x52C + (n × 0x20)

Time added after the sequence

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	CNT			Time added after the sequence in PWM periods																															

8.15.5.37 PSEL.OUT[n] (n=0..3)

Address offset: 0x560 + (n × 0x4)

Output pin select for PWM channel n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C																								B B B A A A A A							
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PIN		[0..31]				Pin number																											
B	RW	PORT		[0..7]				Port number																											
C	RW	CONNECT						Connection																											
			Disconnected	1				Disconnect																											
			Connected	0				Connect																											

8.15.5.38 DMA.SEQ[n].PTR (n=0..1)

Address offset: $0x704 + (n \times 0x24)$

RAM buffer start address

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	PTR			RAM buffer start address for this EasyDMA channel. This address is a word aligned Data RAM address.																															

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.15.5.39 DMA.SEQ[n].MAXCNT (n=0..1)

Address offset: $0x708 + (n \times 0x24)$

Maximum number of bytes in channel buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																																							
ID																																A				A				A				A				A				A				A			
Reset 0x00000000				0 0																																																							
ID	R/W	Field	Value ID	Value								Description																																															
A	RW	MAXCNT		[1..0x7fff]								Maximum number of bytes in channel buffer																																															

8.15.5.40 DMA.SEQ[n].AMOUNT (n=0..1)

Address offset: $0x70C + (n \times 0x24)$

Number of bytes transferred in the last transaction, updated after the END event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																				A A A A A A A A A A A A A A A A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	AMOUNT		[1..0x7fff]				Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.																											

8.15.5.41 DMA.SEQ[n].CURRENTAMOUNT (n=0..1)

Address offset: $0x710 + (n \times 0x24)$

Number of bytes transferred in the current transaction

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value		Description																															
A	R	AMOUNT		[1..0x7fff]		Number of bytes transferred in the current transaction. Continuously updated.																															

8.15.5.42 DMA.SEQ[n].TERMINATEONBUSERROR (n=0..1)

Address offset: $0x71C + (n \times 0x24)$

Terminate the transaction if a BUSERROR event is detected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ENABLE																																	
			Disabled	0				Disable																											
			Enabled	1				Enable																											

8.15.5.43 DMA.SEQ[n].BUSERRORADDRESS (n=0..1)

Address offset: $0x720 + (n \times 0x24)$

Address of transaction that generated the last BUSERROR event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	ADDRESS																																	

8.16 QDEC — Quadrature decoder

The Quadrature decoder (QDEC) provides buffered decoding of quadrature-encoded sensor signals. It is suitable for mechanical and optical sensors.

The main features of QDEC are:

- Digital waveform decoding from off-chip quadrature encoder.
- Sample accumulation eliminating hard real-time requirements to be enforced on application.
- Configurable sample period and accumulation to match application requirements.
- Optional input de-bounce filters.
- Optional LED output signal for optical encoders.

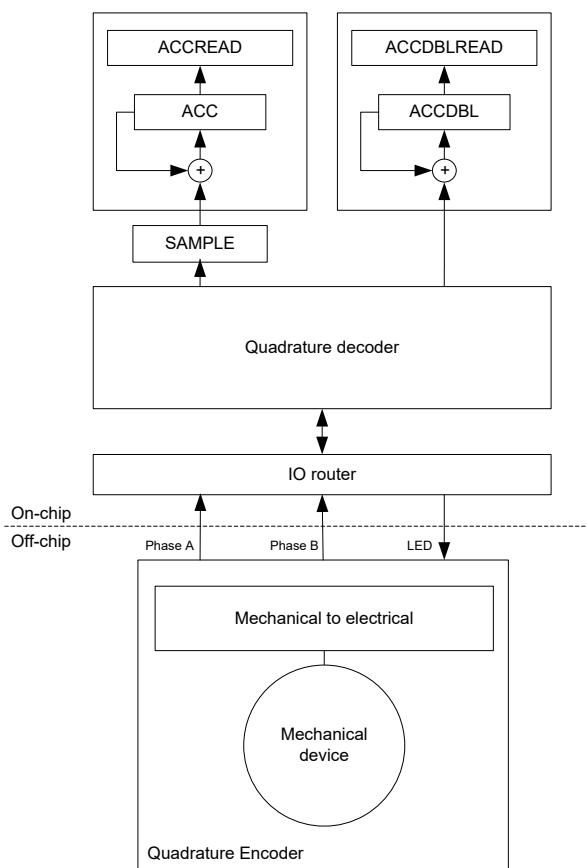


Figure 106: Quadrature decoder configuration

8.16.1 Sampling and decoding

The QDEC decodes the output from an incremental motion encoder by sampling the QDEC phase input pins (A and B).

The off-chip quadrature encoder is an incremental motion encoder outputting two waveforms, phase A and phase B. The two output waveforms are always 90 degrees out of phase, meaning that one always changes level before the other. The direction of movement is indicated by the waveform that changes level first. Invalid transitions may occur, meaning the two waveforms simultaneously switch. This may occur if the wheel rotates too fast relative to the sample rate set for the decoder.

The QDEC decodes the output from the off-chip encoder by sampling the QDEC phase input pins (A and B) at a fixed rate as specified in the SAMPLEPER register.

If the SAMPLEPER value needs to be changed, the QDEC shall be stopped using the STOP task. SAMPLEPER can be then changed upon receiving the STOPPED event, and QDEC can be restarted using the START task. Failing to do so may result in unpredictable behavior.

It is good practice to only change registers LEDPOL, REPORTPER, DBFEN, and LEDPRE when the QDEC is stopped.

When started, the decoder continuously samples the two input waveforms and decodes these by comparing the current sample pair (n) with the previous sample pair (n-1).

The decoding of the sample pairs is described in the table below.

Previous sample pair(n - 1)		Current samples pair(n)		SAMPLE register	ACC operation	ACCDBL operation	Description
A	B	A	B				
0	0	0	0	0	No change	No change	No movement
0	0	0	1	1	Increment	No change	Movement in positive direction
0	0	1	0	-1	Decrement	No change	Movement in negative direction
0	0	1	1	2	No change	Increment	Error: Double transition
0	1	0	0	-1	Decrement	No change	Movement in negative direction
0	1	0	1	0	No change	No change	No movement
0	1	1	0	2	No change	Increment	Error: Double transition
0	1	1	1	1	Increment	No change	Movement in positive direction
1	0	0	0	1	Increment	No change	Movement in positive direction
1	0	0	1	2	No change	Increment	Error: Double transition
1	0	1	0	0	No change	No change	No movement
1	0	1	1	-1	Decrement	No change	Movement in negative direction
1	1	0	0	2	No change	Increment	Error: Double transition
1	1	0	1	-1	Decrement	No change	Movement in negative direction
1	1	1	0	1	Increment	No change	Movement in positive direction
1	1	1	1	0	No change	No change	No movement

Table 47: Sampled value encoding

8.16.2 LED output

The LED output follows the sample period. The LED is switched on for a set period before sampling and then switched off immediately after. The period the LED is switched on before sampling is given in the LEDPRE register.

The LED output pin polarity is specified in the LEDPOL register.

When using off-chip mechanical encoders not requiring an LED, the LED output can be disabled by writing value 'Disconnected' to the CONNECT field of the PSEL.LED register. In this case, the QDEC will not acquire access to a pin for the LED output.

8.16.3 Debounce filters

Each of the two-phase inputs have digital debounce filters.

When enabled through the DBFEN register, the filter inputs are sampled at a fixed 1 MHz frequency during the entire sample period (which is specified in the SAMPLEPER register). The filters require all of the samples within this sample period to equal before the input signal is accepted and transferred to the output of the filter.

As a result, only input signal with a steady state longer than twice the period specified in SAMPLEPER are guaranteed to pass through the filter. Any signal with a steady state shorter than SAMPLEPER will always be suppressed by the filter. It is assumed that the frequency during the debounce period never exceeds 500 kHz (as required by the Nyquist theorem when using a 1 MHz sample frequency).

The LED will always be ON when the debounce filters are enabled, as the inputs in this case will be sampled continuously.

When the debounce filters are enabled, displacements reported by the QDEC peripheral are delayed by one SAMPLEPER period.

8.16.4 Accumulators

The quadrature decoder contains two accumulator registers, ACC and ACCDBL. These registers accumulate valid motion sample values and the number of detected invalid samples (double transitions), respectively.

The ACC register accumulates all valid values (1/-1) written to the SAMPLE register. This can be useful for preventing hard real-time requirements from being enforced on the application. When using the ACC register, the application can fetch data when necessary instead of reading all SAMPLE register output. The ACC register holds the relative movement of the external mechanical device from the previous clearing of the ACC register. Sample values indicating a double transition (2) will not be accumulated in the ACC register.

An ACCOF event is generated if the ACC receives a SAMPLE value that would cause the register to overflow or underflow. Any SAMPLE value that would cause an ACC overflow or underflow will be discarded, but any samples that do not cause the ACC to overflow or underflow will still be accepted.

The accumulator ACCDBL accumulates the number of detected double transitions since the previous clearing of the ACCDBL register.

The ACC and ACCDBL registers can be cleared by the READCLRACC and subsequently read using the ACCREAD and ACCDBLREAD registers.

The ACC register can be separately cleared by the RDCLRACC and subsequently read using the ACCREAD registers.

The ACCDBL register can be separately cleared by the RDCLRDBL and subsequently read using the ACCDBLREAD registers.

The REPORTPER register allows automated capture of multiple samples before sending an event. When a non-null displacement is captured and accumulated, a REPORTRDY event is sent. When one or more double-displacements are captured and accumulated, a DBLRDY event is sent. The REPORTPER field in this register determines how many samples must be accumulated before the contents are evaluated and a REPORTRDY or DBLRDY event is sent.

Using the RDCLRACC task (manually sent upon receiving the event, or using the DBLRDY_RDCLRACC shortcut), ACCREAD can then be read.

When a double transition has been captured and accumulated, a DBLRDY event is sent. Using the RDCLRDBL task (manually sent upon receiving the event, or using the DBLRDY_RDCLRDBL shortcut), ACCDBLREAD can then be read.

8.16.5 Output/input pins

The QDEC uses a three-pin interface to the off-chip quadrature encoder.

These pins are acquired when the QDEC is enabled in the ENABLE register. The pins acquired by the QDEC cannot be written by the CPU, but they can still be read by the CPU.

The pin numbers used for the QDEC are selected using the PSEL.n registers.

8.16.6 Pin configuration

The Phase A, Phase B, and LED signals are mapped to physical pins according to the configuration specified in the PSEL.A, PSEL.B, and PSEL.LED registers respectively.

If the CONNECT field value 'Disconnected' is specified in any of these registers, the associated signal will not be connected to any physical pin. The PSEL.A, PSEL.B, and PSEL.LED registers and their configurations are only used as long as the QDEC is enabled, and retained only as long as the device is in ON mode. When the peripheral is disabled, the pins will behave as regular GPIOs, and use the configuration in their respective OUT bit field and PIN_CNF[n] register.

To secure correct behavior in the QDEC, the pins used by the QDEC must be configured in the GPIO peripheral as described in [GPIO configuration before enabling peripheral](#) on page 440 before enabling the QDEC. This configuration must be retained in the GPIO for the selected I/Os as long as the QDEC is enabled.

Only one peripheral can be assigned to drive a particular GPIO pin at a time. Failing to do so may result in unpredictable behavior.

QDEC signal	QDEC pin	Direction	Output value	Comment
Phase A	As specified in PSEL.A	Input	Not applicable	
Phase B	As specified in PSEL.B	Input	Not applicable	
LED	As specified in PSEL.LED	Input	Not applicable	

Table 48: GPIO configuration before enabling peripheral

8.16.7 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
QDEC20 : S	GLOBAL	0x500E0000	US	S	NA	No	Quadrature decoder QDEC20
QDEC20 : NS		0x400E0000					
QDEC21 : S	GLOBAL	0x500E1000	US	S	NA	No	Quadrature decoder QDEC21
QDEC21 : NS		0x400E1000					

Configuration

Instance	Domain	Configuration
QDEC20 : S	GLOBAL	Use GPIO port P1
QDEC20 : NS		
QDEC21 : S	GLOBAL	Use GPIO port P1
QDEC21 : NS		

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Task starting the quadrature decoder
TASKS_STOP	0x004		Task stopping the quadrature decoder
TASKS_READCLRACC	0x008		Read and clear ACC and ACCDBL
TASKS_RDCLRACC	0x00C		Read and clear ACC
TASKS_RDCLRDBL	0x010		Read and clear ACCDBL
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
SUBSCRIBE_READCLRACC	0x088		Subscribe configuration for task READCLRACC
SUBSCRIBE_RDCLRACC	0x08C		Subscribe configuration for task RDCLRACC
SUBSCRIBE_RDCLRDBL	0x090		Subscribe configuration for task RDCLRDBL
EVENTS_SAMPLERDY	0x100		Event being generated for every new sample value written to the SAMPLE register
EVENTS_REPORTRDY	0x104		Non-null report ready
EVENTS_ACCOF	0x108		ACC or ACCDBL register overflow
EVENTS_DBLRDY	0x10C		Double displacement(s) detected
EVENTS_STOPPED	0x110		QDEC has been stopped
PUBLISH_SAMPLERDY	0x180		Publish configuration for event SAMPLERDY
PUBLISH_REPORTRDY	0x184		Publish configuration for event REPORTRDY
PUBLISH_ACCOF	0x188		Publish configuration for event ACCOF
PUBLISH_DBLRDY	0x18C		Publish configuration for event DBLRDY

Register	Offset	TZ	Description
PUBLISH_STOPPED	0x190		Publish configuration for event STOPPED
SHORTS	0x200		Shortcuts between local events and tasks
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
ENABLE	0x500		Enable the quadrature decoder
LEDPOL	0x504		LED output pin polarity
SAMPLEPER	0x508		Sample period
SAMPLE	0x50C		Motion sample value
REPORTPER	0x510		Number of samples to be taken before REPORTRDY and DBLRDY events can be generated
ACC	0x514		Register accumulating the valid transitions
ACCREAD	0x518		Snapshot of the ACC register, updated by the READCLRACC or RDCLRACC task
PSEL.LED	0x51C		Pin select for LED signal
PSEL.A	0x520		Pin select for A signal
PSEL.B	0x524		Pin select for B signal
DBFEN	0x528		Enable input debounce filters
LEDPRE	0x540		Time period the LED is switched ON prior to sampling
ACCDL	0x544		Register accumulating the number of detected double transitions
ACCDLREAD	0x548		Snapshot of the ACCDL, updated by the READCLRACC or RDCLRDL task

8.16.7.1 TASKS_START

Address offset: 0x000

Task starting the quadrature decoder

When started, the SAMPLE register will be continuously updated at the rate given in the SAMPLEPER register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_START						Task starting the quadrature decoder																											
								When started, the SAMPLE register will be continuously updated at the rate given in the SAMPLEPER register.																											
		Trigger		1				Trigger task																											

8.16.7.2 TASKS_STOP

Address offset: 0x004

Task stopping the quadrature decoder

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_STOP						Task stopping the quadrature decoder																											
		Trigger		1				Trigger task																											

8.16.7.3 TASKS_READCLRACC

Address offset: 0x008

Read and clear ACC and ACCDL

Task transferring the content of ACC to ACCREAD and the content of ACCDBL to ACCDBLREAD, and then clearing the ACC and ACCDBL registers. These read-and-clear operations will be done atomically.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_READCLRACC						Read and clear ACC and ACCDBL																											
				Task transferring the content of ACC to ACCREAD and the content of ACCDBL																															
				to ACCDBLREAD, and then clearing the ACC and ACCDBL registers. These																															
				read-and-clear operations will be done atomically.																															
				Trigger				1				Trigger task																							

8.16.7.4 TASKS_RDCLRACC

Address offset: 0x00C

Read and clear ACC

Task transferring the content of ACC to ACCREAD, and then clearing the ACC register. This read-and-clear operation will be done atomically.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_RDCLRACC						Read and clear ACC																											
								Task transferring the content of ACC to ACCREAD, and then clearing the ACC register. This read-and-clear operation will be done atomically.																											
			Trigger	1				Trigger task																											

8.16.7.5 TASKS_RDCLRDBL

Address offset: 0x010

Read and clear ACCDBL

Task transferring the content of ACCDBL to ACCDBLREAD, and then clearing the ACCDBL register. This read-and-clear operation will be done atomically.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				A																																
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																												
A	W	TASKS_RDCLRDBL						Read and clear ACCDBL																												
								Task transferring the content of ACCDBL to ACCDBLREAD, and then clearing the ACCDBL register. This read-and-clear operation will be done atomically.																												
			Trigger	1				Trigger task																												

8.16.7.6 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task **START**

When started, the SAMPLE register will be continuously updated at the rate given in the SAMPLEPER register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task START will subscribe to																													
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.16.7.7 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task **STOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task STOP will subscribe to																													
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.16.7.8 SUBSCRIBE_READCLRACC

Address offset: 0x088

Subscribe configuration for task **READCLRACC**

Task transferring the content of ACC to ACCREAD and the content of ACCDBL to ACCDBLREAD, and then clearing the ACC and ACCDBL registers. These read-and-clear operations will be done atomically.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task READCLRACC will subscribe to																													
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.16.7.9 SUBSCRIBE_RDCLRACC

Address offset: 0x08C

Subscribe configuration for task **RDCLRACC**

Task transferring the content of ACC to ACCREAD, and then clearing the ACC register. This read-and-clear operation will be done atomically.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task RDCLRACC will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.16.7.10 SUBSCRIBE_RDCLRDBL

Address offset: 0x090

Subscribe configuration for task **RDCLRDBL**

Task transferring the content of ACCDBL to ACCDBLREAD, and then clearing the ACCDBL register. This read-and-clear operation will be done atomically.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task RDCLRDBL will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.16.7.11 EVENTS_SAMPLERDY

Address offset: 0x100

Event being generated for every new sample value written to the SAMPLE register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_SAMPLERDY			Event being generated for every new sample value written to the SAMPLE register																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.16.7.12 EVENTS_REPORTRDY

Address offset: 0x104

Non-null report ready

Event generated when REPORTPER number of samples has been accumulated in the ACC register and the content of the ACC register is not equal to 0. (Thus, this event is only generated if a motion is detected since the previous clearing of the ACC register).

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	EVENTS_REPORTRDY				Non-null report ready																													
						Event generated when REPORTPER number of samples has been accumulated in the ACC register and the content of the ACC register is not equal to 0. (Thus, this event is only generated if a motion is detected since the previous clearing of the ACC register).																													
			NotGenerated	0		Event not generated																													
			Generated	1		Event generated																													

8.16.7.13 EVENTS_ACCOF

Address offset: 0x108

ACC or ACCDBL register overflow

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_ACCOF						ACC or ACCDBL register overflow																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.16.7.14 EVENTS_DBLRDY

Address offset: 0x10C

Double displacement(s) detected

Event generated when REPORTPER number of samples has been accumulated and the content of the ACCDBL register is not equal to 0. (Thus, this event is only generated if a double transition is detected since the previous clearing of the ACCDBL register).

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_DBLRDY			Double displacement(s) detected																														
					Event generated when REPORTPER number of samples has been accumulated and the content of the ACCDBL register is not equal to 0. (Thus, this event is only generated if a double transition is detected since the previous clearing of the ACCDBL register).																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.16.7.15 EVENTS_STOPPED

Address offset: 0x110

QDEC has been stopped

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	EVENTS_STOPPED			QDEC has been stopped																																	
			NotGenerated	0	Event not generated																																	
			Generated	1	Event generated																																	

8.16.7.16 PUBLISH_SAMPLERDY

Address offset: 0x180

Publish configuration for event [SAMPLERDY](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																													
ID				B																								A												A	A	A	A	A	A	A																		
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																										
A	RW	CHIDX		[0..255]		DPPI channel that event SAMPLERDY will publish to																																																										
B	RW	EN																																																														
			Disabled	0		Disable publishing																																																										
			Enabled	1		Enable publishing																																																										

8.16.7.17 PUBLISH_REPORTRDY

Address offset: 0x184

Publish configuration for event [REPORTRDY](#)

Event generated when REPORTPER number of samples has been accumulated in the ACC register and the content of the ACC register is not equal to 0. (Thus, this event is only generated if a motion is detected since the previous clearing of the ACC register).

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																													
ID				B																								A												A	A	A	A	A	A	A																		
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																								
A	RW	CHIDX		[0..255]				DPPI channel that event REPORTRDY will publish to																																																								
B	RW	EN																																																														
			Disabled	0				Disable publishing																																																								
			Enabled	1				Enable publishing																																																								

8.16.7.18 PUBLISH_ACCOF

Address offset: 0x188

Publish configuration for event [ACCOF](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event ACCOF will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.16.7.19 PUBLISH_DBLRDY

Address offset: 0x18C

Publish configuration for event **DBLRDY**

Event generated when REPORTPER number of samples has been accumulated and the content of the ACCDBL register is not equal to 0. (Thus, this event is only generated if a double transition is detected since the previous clearing of the ACCDBL register).

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event DBLRDY will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.16.7.20 PUBLISH_STOPPED

Address offset: 0x190

Publish configuration for event **STOPPED**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event STOPPED will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.16.7.21 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	REPORTRDY_READCLRACC						Shortcut between event REPORTRDY and task READCLRACC																											
			Disabled	0				Disable shortcut																											

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
B	RW	SAMPLERDY_STOP	Enabled	1	Enable shortcut																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
C	RW	REPORTRDY_RDCLRACC			Shortcut between event REPORTRDY and task RDCLRACC																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
D	RW	REPORTRDY_STOP			Shortcut between event REPORTRDY and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
E	RW	DBLRDY_RDCLRDBL			Shortcut between event DBLRDY and task RDCLRDBL																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
F	RW	DBLRDY_STOP			Shortcut between event DBLRDY and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
G	RW	SAMPLERDY_READCLRACC			Shortcut between event SAMPLERDY and task READCLRACC																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														

8.16.7.22 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	SAMPLERDY			Write '1' to enable interrupt for event SAMPLERDY																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	REPORTRDY			Write '1' to enable interrupt for event REPORTRDY																														
					Event generated when REPORTPER number of samples has been accumulated in the ACC register and the content of the ACC register is not equal to 0. (Thus, this event is only generated if a motion is detected since the previous clearing of the ACC register).																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
		Enabled	1	Read: Enabled																															
C	RW	ACCOF			Write '1' to enable interrupt for event ACCOF																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
D	RW	DBLRDY			Write '1' to enable interrupt for event DBLRDY																														
					Event generated when REPORTPER number of samples has been accumulated and the content of the ACCDBL register is not equal to 0. (Thus, this event is only generated if a double transition is detected since the previous clearing of the ACCDBL register).																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
E	RW	STOPPED			Write '1' to enable interrupt for event STOPPED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.16.7.23 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			E D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	SAMPLERDY			Write '1' to disable interrupt for event SAMPLERDY																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
B	RW	REPORTRDY			Write '1' to disable interrupt for event REPORTRDY																													
					Event generated when REPORTPER number of samples has been accumulated in the ACC register and the content of the ACC register is not equal to 0. (Thus, this event is only generated if a motion is detected since the previous clearing of the ACC register).																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
C	RW	ACCOF			Write '1' to disable interrupt for event ACCOF																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
D	RW	DBLRDY			Write '1' to disable interrupt for event DBLRDY																													
					Event generated when REPORTPER number of samples has been accumulated and the content of the ACCDBL register is not equal to 0. (Thus, this event is only generated if a double transition is detected since the previous clearing of the ACCDBL register).																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
E	RW	STOPPED			Write '1' to disable interrupt for event STOPPED																													
			Clear	1	Disable																													

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																
ID				E D C B A																																
Reset 0x00000000				0 0																																
ID	R/W	Field	Value ID	Value	Description																															
			Disabled	0	Read: Disabled																															
			Enabled	1	Read: Enabled																															

8.16.7.24 ENABLE

Address offset: 0x500

Enable the quadrature decoder

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-	RW	ENABLE			Enable or disable the quadrature decoder																														
					When enabled the decoder pins will be active. When disabled the quadrature decoder pins are not active and can be used as GPIO .																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.16.7.25 LEDPOL

Address offset: 0x504

LED output pin polarity

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	LEDPOL				LED output pin polarity																													
			ActiveLow	0	Led active on output pin low																														
			ActiveHigh	1	Led active on output pin high																														

8.16.7.26 SAMPLEPER

Address offset: 0x508

Sample period

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
			8192us	6		8192 μs																													
			16384us	7		16384 μs																													
			32ms	8		32768 μs																													
			65ms	9		65536 μs																													
			131ms	10		131072 μs																													

8.16.7.27 SAMPLE

Address offset: 0x50C

Motion sample value

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	SAMPLE		[-1..2]								Last motion sample																							

The value is a 2's complement value, and the sign gives the direction of the motion. The value '2' indicates a double transition.

8.16.7.28 REPORTPER

Address offset: 0x510

Number of samples to be taken before REPORTRDY and DBLRDY events can be generated

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	REPORTPER			Specifies the number of samples to be accumulated in the ACC register																														

The report period in [μ s] is given as: $RPUS = SP * RP$, where RPUS is the report period in [μ s/report], SP is the sample period in [μ s/sample] specified in SAMPLEPER, and RP is the report period in [samples/report] specified in REPORTPER.

10Smpl	0	10 samples/report
40Smpl	1	40 samples/report
80Smpl	2	80 samples/report
120Smpl	3	120 samples/report
160Smpl	4	160 samples/report
200Smpl	5	200 samples/report
240Smpl	6	240 samples/report
280Smpl	7	280 samples/report
1Smpl	8	1 sample/report

8.16.7.29 ACC

Address offset: 0x514

Register accumulating the valid transitions

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	ACC		[-1024..1023]				Register accumulating all valid samples (not double transition) read from the SAMPLE register. Double transitions (SAMPLE = 2) will not be accumulated in this register. The value is a 32 bit 2's complement value. If a sample that would cause this register to overflow or underflow is received, the sample will be ignored and an overflow event (ACCOF) will be generated. The ACC register is cleared by triggering the READCLRACC or the RDCLRACC task.																											

8.16.7.30 ACCREAD

Address offset: 0x518

Snapshot of the ACC register, updated by the READCLRACC or RDCLRACC task

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																													
A	R	ACCREAD		[-1024..1023]		Snapshot of the ACC register.																													
						The ACCREAD register is updated when the READCLRACC or RDCLRACC task is triggered.																													

8.16.7.31 PSEL.LED

Address offset: 0x51C

Pin select for LED signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				C																								B				B	B	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	PIN		[0..31]		Pin number																															
B	RW	PORT		[0..7]		Port number																															
C	RW	CONNECT				Connection																															
			Disconnected	1		Disconnect																															
			Connected	0		Connect																															

8.16.7.32 PSEL.A

Address offset: 0x520

Pin select for A signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				C																								B				B	B	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
ID	R/W	Field	Value ID	Value				Description																													
A	RW	PIN		[0..31]				Pin number																													
B	RW	PORT		[0..7]				Port number																													
C	RW	CONNECT						Connection																													
			Disconnected	1				Disconnect																													
			Connected	0				Connect																													

8.16.7.33 PSEL.B

Address offset: 0x524

Pin select for B signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				C																								B				B	B	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
ID	R/W	Field	Value ID	Value				Description																													
A	RW	PIN		[0..31]				Pin number																													
B	RW	PORT		[0..7]				Port number																													
C	RW	CONNECT						Connection																													
			Disconnected	1				Disconnect																													
			Connected	0				Connect																													

8.16.7.34 DBFEN

Address offset: 0x528

Enable input debounce filters

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																				A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	DBFEN				Enable input debounce filters																															
			Disabled	0		Debounce input filters disabled																															
			Enabled	1		Debounce input filters enabled																															

8.16.7.35 LEDPRE

Address offset: 0x540

Time period the LED is switched ON prior to sampling

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

8.16.7.36 ACCDBL

Address offset: 0x544

Register accumulating the number of detected double transitions

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	ACCDBL		[0..15]	Register accumulating the number of detected double or illegal transitions. (SAMPLE = 2). When this register has reached its maximum value, the accumulation of double/illegal transitions will stop. An overflow event (ACCOF) will be generated if any double or illegal transitions are detected after the maximum value was reached. This field is cleared by triggering the READCLRACC or RDCLRDBL task.																														

8.16.7.37 ACCDBLREAD

Address offset: 0x548

Snapshot of the ACCDBL, updated by the READCLRACC or RDCLRDBL task

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	ACCDBLREAD		[0..15]				Snapshot of the ACCDBL register. This field is updated when the READCLRACC or RDCLRDBL task is triggered.																											

8.17 RADIO — 2.4 GHz radio

The 2.4 GHz radio transceiver is compatible with multiple radio standards such as Bluetooth Low Energy, IEEE 802.15.4, and Nordic's proprietary modes.

The main features of RADIO are:

- Multidomain 2.4 GHz radio transceiver, with
 - Bluetooth Low Energy 1 Mbps and 2 Mbps modes
 - Bluetooth Low Energy Long Range (125 kbps and 500 kbps) modes
 - IEEE 802.15.4 250 kbps mode
 - 1 Mbps, 2 Mbps and 4 Mbps Nordic proprietary modes
- Best in class link budget and low power operation
- Efficient data interface with EasyDMA support
- Automatic address filtering and pattern matching
- Automated packet assembler/disassembler
- Automated CRC generator and checker

EasyDMA, in combination with an automated packet assembler, packet disassembler, automated CRC generator and CRC checker, makes it easy to configure and use RADIO. See the following figure for details.

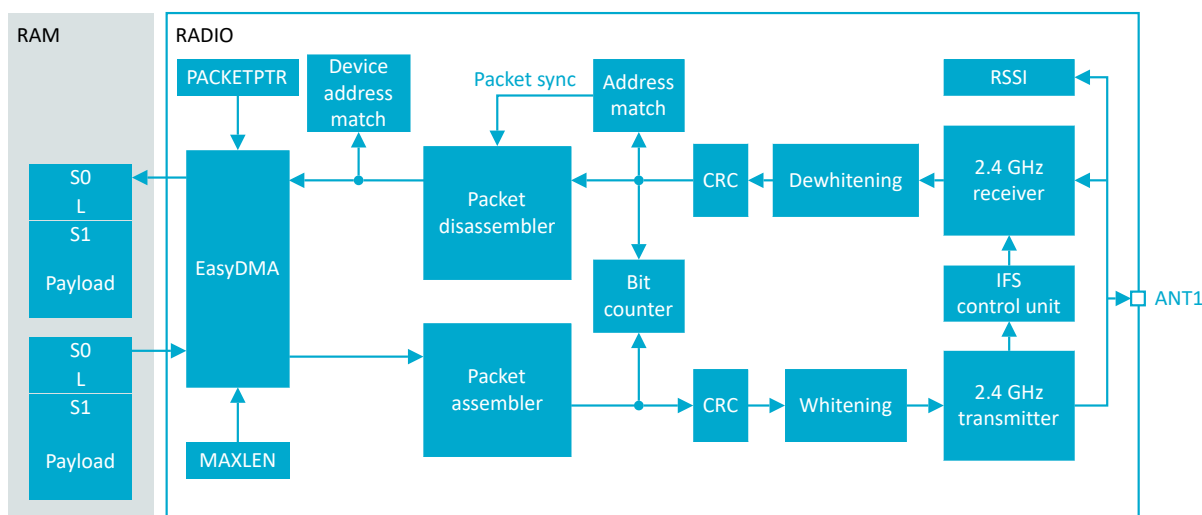


Figure 107: RADIO block diagram

RADIO includes a device address match unit and an interframe spacing control unit that can be utilized to simplify device filtering and interframe spacing respectively in Bluetooth Low Energy and similar applications.

RADIO also includes a received signal strength indicator (RSSI) and a bit counter. The bit counter generates events when a preconfigured number of bits are sent or received by RADIO.

8.17.1 Packet configuration

A RADIO packet contains the fields PREAMBLE, ADDRESS, S0, LENGTH, S1, PAYLOAD, and CRC. For Long Range (125 kbps and 500 kbps) Bluetooth Low Energy modes, fields CI, TERM1, and TERM2 are also included.

The content of a RADIO packet is illustrated in the following figures. RADIO sends the fields in the packet according to the sequence shown in the figures, starting on the left.

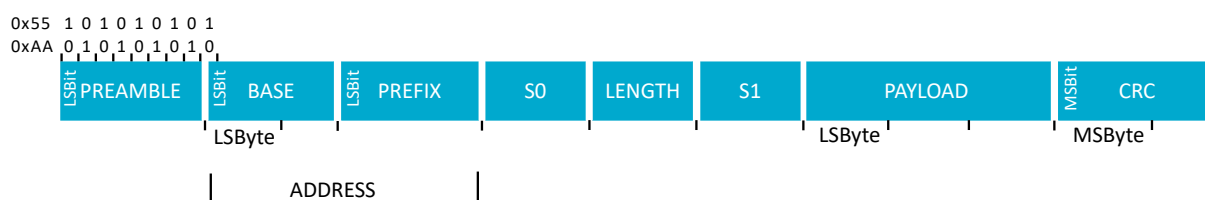


Figure 108: On-air packet layout



Figure 109: On-air packet layout for Long Range (125 kbps and 500 kbps) Bluetooth Low Energy modes

Not shown in the figures is the static payload add-on (the length of which is defined in `PCNF1.STATLEN`, and which is 0 bytes in a standard BLE packet). The static payload add-on is sent between PAYLOAD and CRC fields. RADIO sends the different fields in the packet in the order they are illustrated above, from left to right.

PREAMBLE is sent with least significant bit first on air. The size of the PREAMBLE depends on the mode selected in the `MODE` register:

- The PREAMBLE is one byte for **MODE** = Ble_1Mbit as well as all Nordic proprietary operating modes (**MODE** = Nrf_1Mbit and **MODE** = Nrf_2Mbit), and **PCNF0.PLEN** has to be set accordingly. If the first bit of the ADDRESS is 0, the preamble will be set to 0xAA. Otherwise the PREAMBLE will be set to 0x55.
- For **MODE** = Ble_2Mbit, the PREAMBLE must be set to 2 bytes through **PCNF0.PLEN**. If the first bit of the ADDRESS is 0, the preamble will be set to 0xAAAA. Otherwise the PREAMBLE will be set to 0x5555.
- For **MODE** = Ble_LR125Kbit and **MODE** = Ble_LR500Kbit, the PREAMBLE is 10 repetitions of 0x3C.
- For **MODE** = leee802154_250Kbit, the PREAMBLE is 4 bytes and set to all zeros.

Radio packets are stored in memory inside instances of a RADIO packet data structure as illustrated below. The PREAMBLE, ADDRESS, CI, TERM1, TERM2, and CRC fields are omitted in this data structure. Fields S0, LENGTH, and S1 are optional.



Figure 110: Representation of a RADIO packet in RAM

The byte ordering on air is always least significant byte first for the ADDRESS and PAYLOAD fields, and most significant byte first for the CRC field. The ADDRESS fields are always transmitted and received least significant bit first. The CRC field is always transmitted and received most significant bit first. The endianness, i.e. the order in which the bits are sent and received, of the S0, LENGTH, S1, and PAYLOAD fields can be configured via **PCNF1.ENDIAN**.

The sizes of the S0, LENGTH, and S1 fields can be individually configured via **S0LEN**, **LFLen**, and **S1LEN** in **PCNF0** respectively. If any of these fields are configured to be less than 8 bits, the least significant bits of the fields are used.

If S0, LENGTH, or S1 are specified with zero length, their fields will be omitted in memory. Otherwise each field will be represented as a separate byte, regardless of the number of bits in their on-air counterpart.

Independent of the configuration of **PCNF1.MAXLEN**, the combined length of S0, LENGTH, S1, and PAYLOAD cannot exceed 258 bytes.

8.17.2 Address configuration

The on-air radio ADDRESS field is composed of two parts, the base address field and the address prefix field.

The size of the base address field is configurable via **PCNF1.BALEN**. The base address is truncated from the least significant byte if the **PCNF1.BALEN** is less than 4. See [Definition of logical addresses](#) on page 457.

The on-air addresses are defined in the **BASE0/BASE1** and **PREFIX0/PREFIX1** registers. It is only when writing these registers that the user must relate to the actual on-air addresses. For other radio address registers, such as the **TXADDRESS**, **RXADDRESSES**, and **RXMATCH** registers, logical radio addresses ranging from 0 to 7 are being used. The relationship between the on-air radio addresses and the logical addresses is described in the following table.

Logical address	Base address	Prefix byte
0	BASE0	PREFIX0.AP0
1	BASE1	PREFIX0.AP1
2	BASE1	PREFIX0.AP2
3	BASE1	PREFIX0.AP3
4	BASE1	PREFIX1.AP4
5	BASE1	PREFIX1.AP5
6	BASE1	PREFIX1.AP6
7	BASE1	PREFIX1.AP7

Table 49: Definition of logical addresses

8.17.3 Data whitening

RADIO can do packet whitening and de-whitening which is enabled in [PCNF1.WHITEEN](#). When enabled, whitening and de-whitening will be handled by RADIO automatically as packets are sent and received.

The data whitening is done by means of a configurable linear feedback shift register in a one-to-many topology, as illustrated in the following figure. The data packet that is to be whitened or de-whitened is XORed with bit 0. The linear feedback shift register is configured and initialized using the [DATAWHITE](#) register. The reset value for the [DATAWHITE.POLY](#) field is compatible with Bluetooth Low Energy. The initial vector is configured in [DATAWHITE.IV](#).

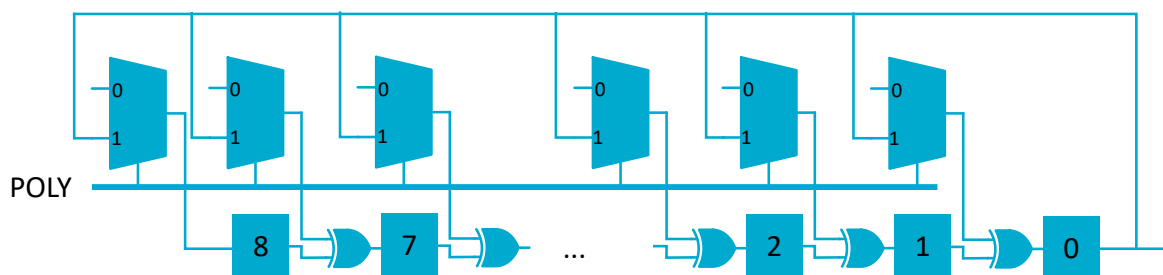


Figure 111: Data whitening and de-whitening

Whitening and de-whitening will be performed over the whole packet except for the preamble and the address fields.

Including the address field in CRC check ([CRCCNF.SKIPADDR](#)=Include) is not supported for whitened packets.

8.17.4 CRC

The CRC generator in RADIO calculates the CRC over the whole packet excluding the preamble.

If useful, the address field can be excluded from the CRC calculation as well. See the [CRCCNF](#) register for more information.

The CRC polynomial is configurable as illustrated in the following figure, where bit 0 in the [CRCPOLY](#) register corresponds to X^0 and bit 1 corresponds to X^1 etc. See [CRCPOLY](#) on page 522 for more information.

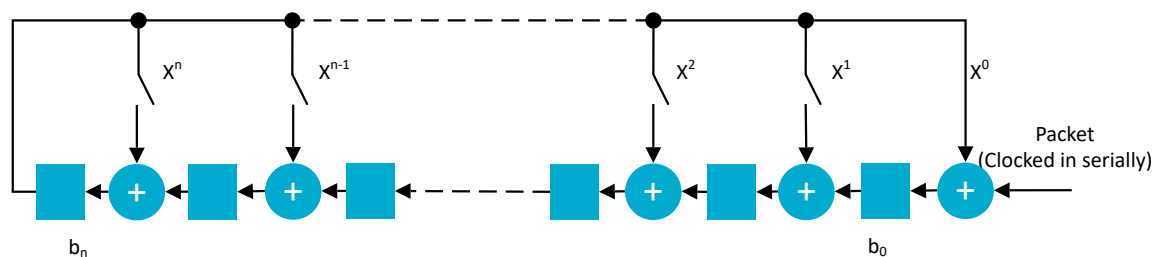


Figure 112: CRC generation of an n bit CRC

The figure shows that the CRC is calculated by feeding the packet serially through the CRC generator. Before the packet is clocked through the CRC generator, the CRC generator's latches b_0 through b_n will be initialized with a predefined value specified in the [CRCINIT](#) register. After the whole packet has been clocked through the CRC generator, b_0 through b_n will hold the resulting CRC. This value will be used by RADIO during both transmission and reception. Latches b_0 through b_n are not available to be read by the CPU at any time. However, a received CRC can be read by the CPU via the [RXCRC](#) register.

The length (n) of the CRC is configurable, see [CRCCNF](#) for more information.

Once the entire packet, including the CRC, has been received and no errors were detected, RADIO generates a [CRCOK](#) event. If CRC errors were detected, a [CRCERROR](#) event is generated.

The status of the CRC check can be read from the [CRCSTATUS](#) register after a packet has been received.

8.17.5 Radio states

Tasks and events are used to control the operating state of RADIO.

RADIO can enter the states described in the following table.

State	Description
DISABLED	No operations are going on inside RADIO and the power consumption is at a minimum
RXRU	RADIO is ramping up and preparing for reception
RXIDLE	RADIO is ready for reception to start
RX	Reception has been started and the addresses enabled in the RXADDRESSES register are being monitored
TXRU	RADIO is ramping up and preparing for transmission
TXIDLE	RADIO is ready for transmission to start
TX	RADIO is transmitting a packet
RXDISABLE	RADIO is disabling the receiver
TXDISABLE	RADIO is disabling the transmitter

Table 50: RADIO state diagram

A state diagram showing an overview of RADIO is shown in the following figure.

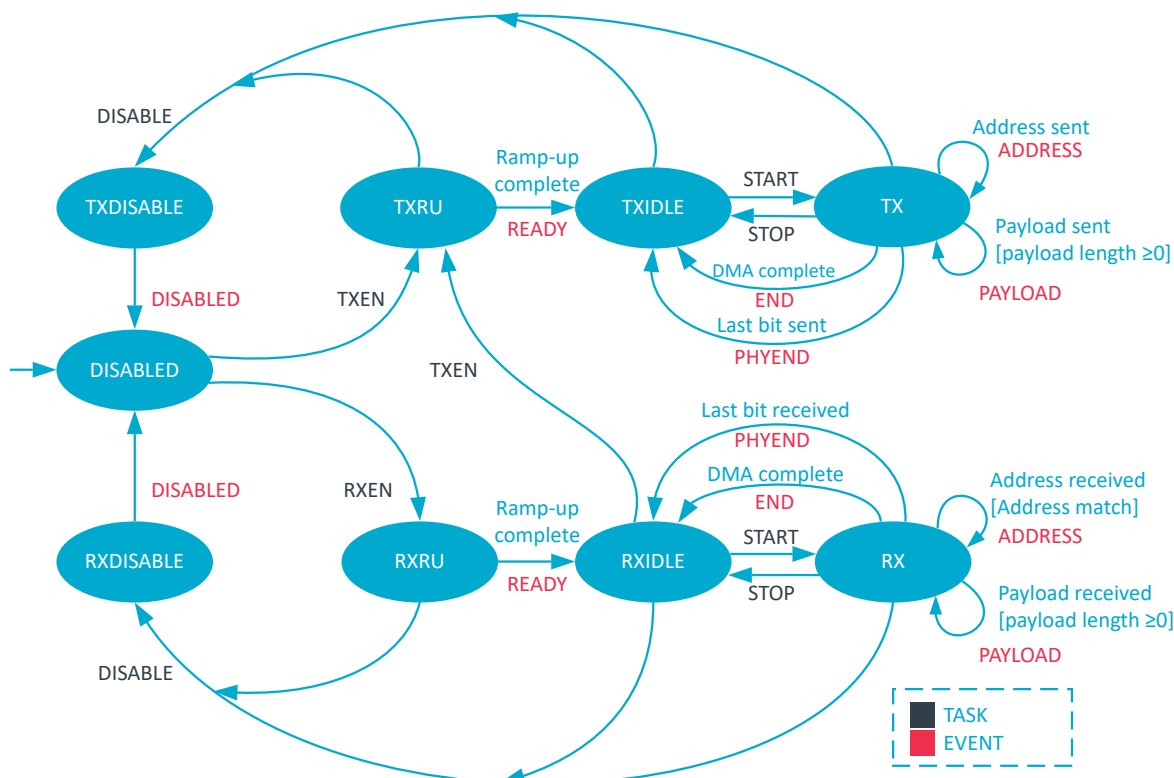


Figure 113: Radio states

This figure shows how the tasks and events relate to RADIO's operation. RADIO does not prevent a task from being triggered from the wrong state. If a task is triggered from the wrong state, for example if the **RXEN** task is triggered from the **RXDISABLE** state, this may lead to incorrect behavior. The **PAYLOAD** event is always generated even if the payload is zero.

The **END** to **START** shortcut should not be used with IEEE 802.15.4 250 kbps mode. Use the **PHYEND** to **START** shortcut instead.

The **END** to **START** shortcut should not be used with Long Range (125 kbps and 500 kbps) Bluetooth Low Energy modes. Use the **PHYEND** to **START** shortcut instead.

8.17.6 Transmit sequence

Before RADIO can transmit a packet, it must first ramp-up in TX mode. See **TXRU** in [Radio states](#) on page 459 and [Transmit sequence](#) on page 460. A **TXRU** ramp-up sequence is initiated when the **TXEN** task is triggered. After RADIO has successfully ramped up it will generate the **READY** event indicating that a packet transmission can be initiated. A packet transmission is initiated by triggering the **START** task. The **START** task can first be triggered after RADIO has entered into the **TXIDLE** state.

The following figure illustrates a single packet transmission where the CPU manually triggers the different tasks needed to control the flow of RADIO, i.e. no shortcuts are used. If shortcuts are not used, a certain amount of delay caused by CPU execution is expected between **READY** and **START**, and between **PHYEND** and **DISABLE**. As illustrated in the following figure, RADIO will by default transmit an unmodulated carrier between **READY** and **START**, and between **PHYEND** and **DISABLE**.

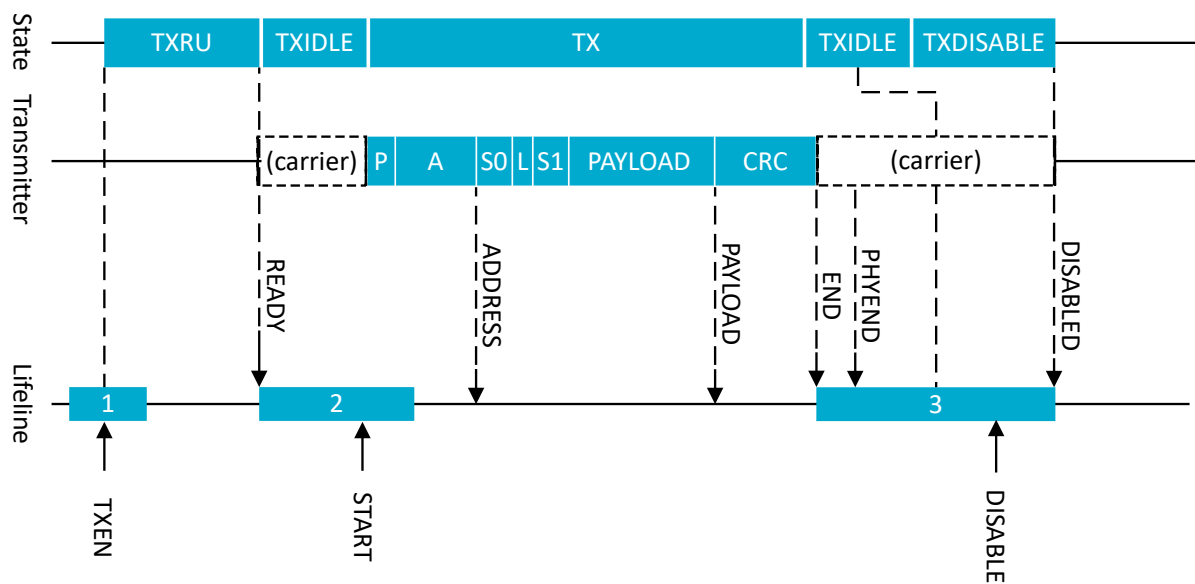


Figure 114: Transmit sequence

The following figure shows a slightly modified version of the transmit sequence where RADIO is configured to use shortcuts between **READY** and **START**, and between **PHYEND** and **DISABLE**, which means that no delay is introduced.

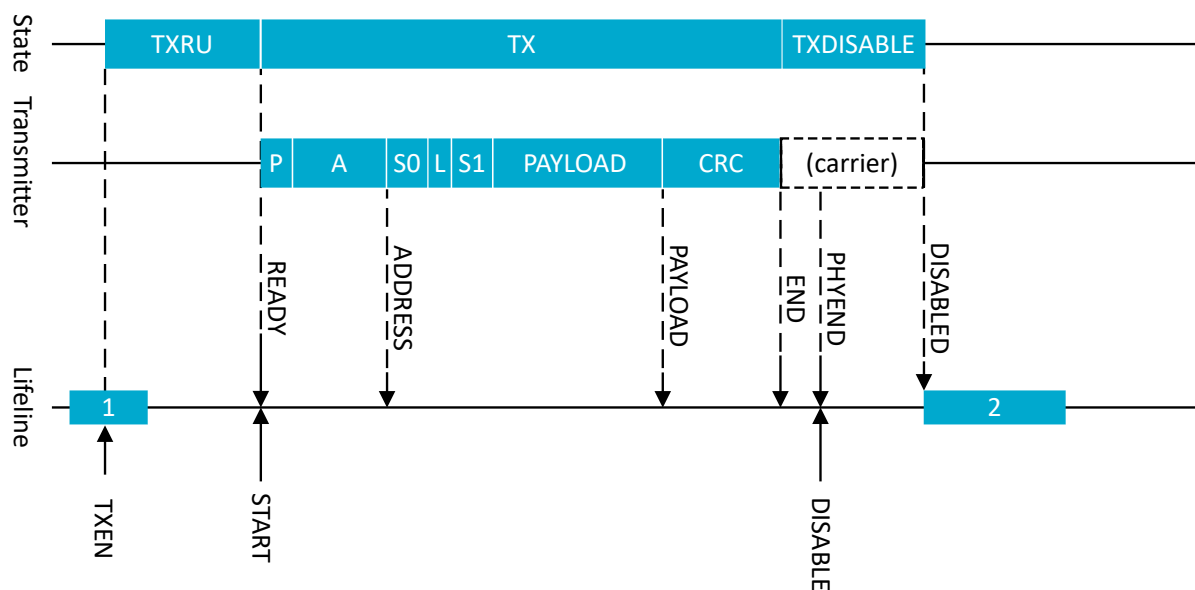


Figure 115: Transmit sequence using shortcuts to avoid delays

RADIO is able to send multiple packets one after the other without having to disable and re-enable RADIO between packets, as illustrated in the following figure.

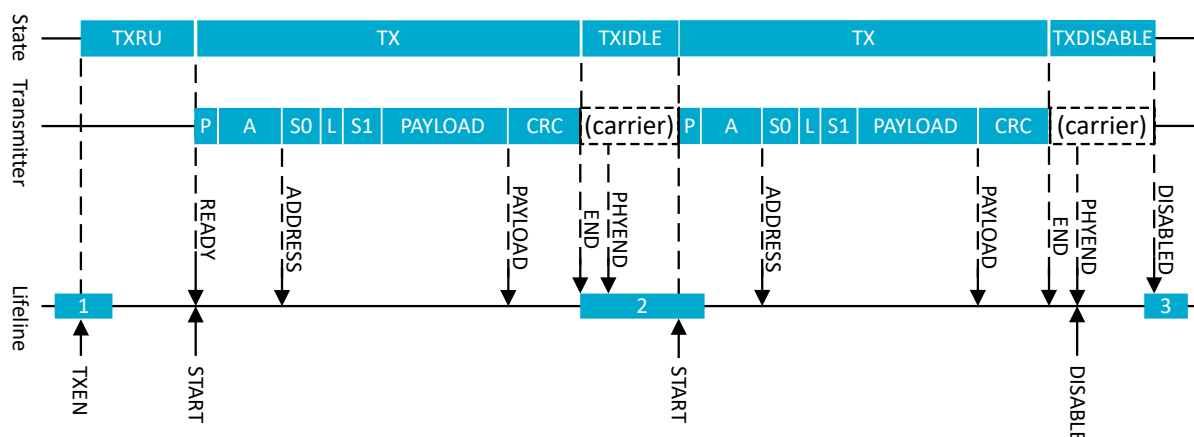


Figure 116: Transmission of multiple packets

8.17.7 Receive sequence

Before RADIO is able to receive a packet, it must first ramp up in RX mode. See RXRU in [Radio states](#) on page 459 and [Receive sequence](#) on page 461 for more information.

An RXRU ramp up sequence is initiated when the [RXEN](#) task is triggered. After RADIO has successfully ramped up it will generate the [READY](#) event indicating that a packet reception can be initiated. A packet reception is initiated by triggering the [START](#) task. As illustrated in [Radio states](#) on page 459, the [START](#) task can first be triggered after RADIO has entered into the RXIDLE state.

The following figure shows a single packet reception where the CPU manually triggers the different tasks needed to control the flow of RADIO, i.e. no shortcuts are used. If shortcuts are not used, a certain amount of delay caused by CPU execution is expected between [READY](#) and [START](#), and between [PHYEND](#) and [DISABLE](#). RADIO will be listening and possibly receiving undefined data, represented with an 'X', from [START](#) and until a packet with valid preamble (P) is received.

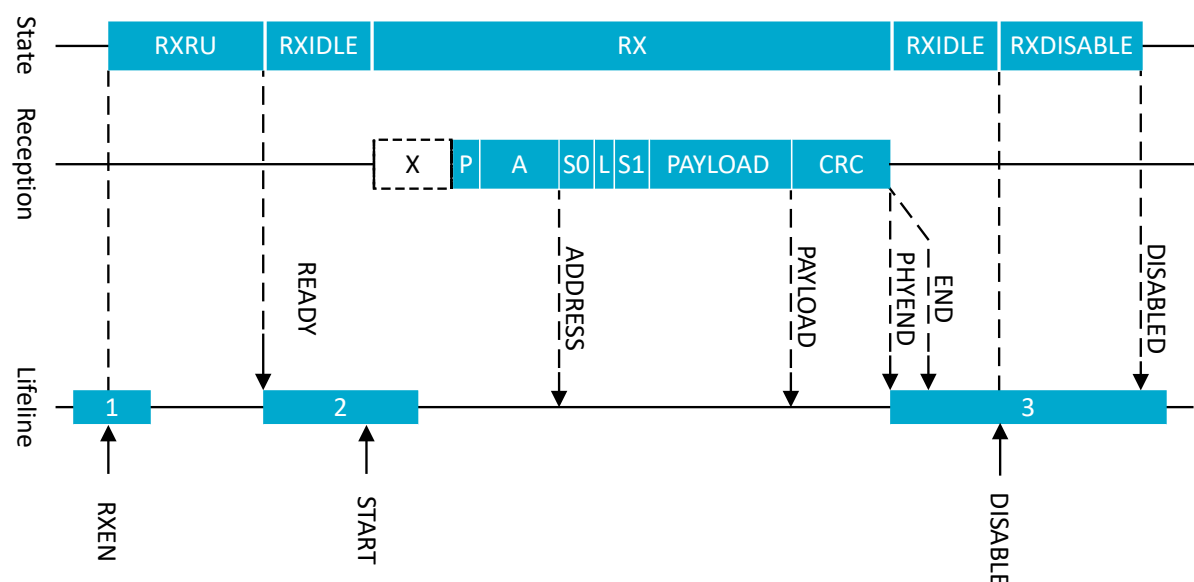


Figure 117: Receive sequence

The following figure shows a modified version of the receive sequence, where RADIO is configured to use shortcuts between [READY](#) and [START](#), and between [END](#) and [DISABLE](#), which means that no delay is introduced.

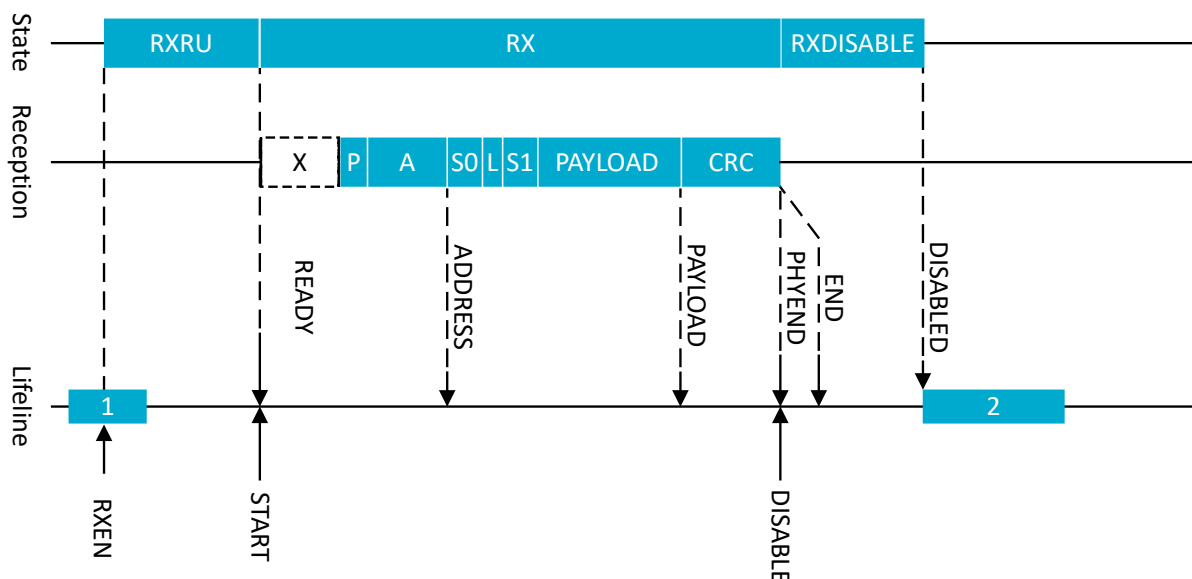


Figure 118: Receive sequence using shortcuts to avoid delays

RADIO can receive consecutive packets without having to disable and re-enable RADIO between packets, as illustrated in the following figure.

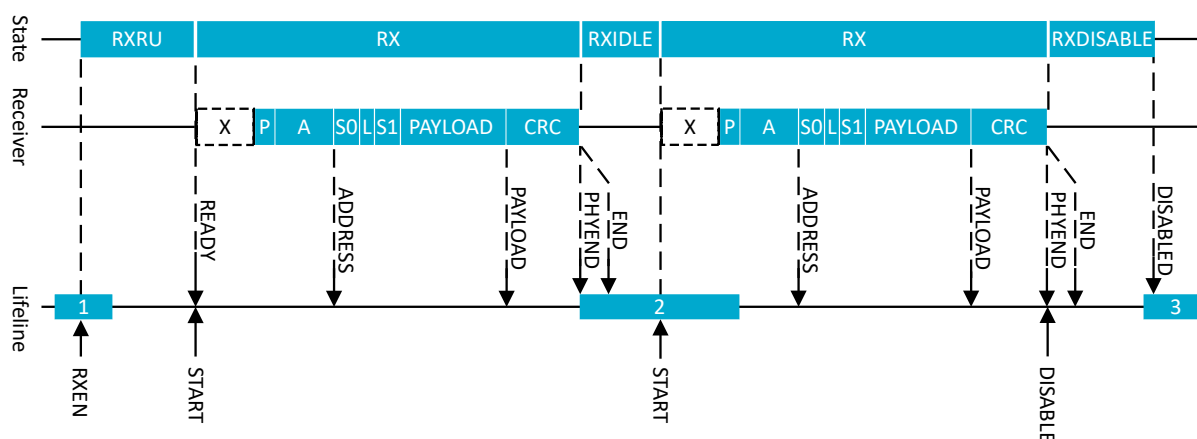


Figure 119: Reception of multiple packets

8.17.8 Received signal strength indicator (RSSI)

RADIO implements a mechanism for measuring the power in the received signal. This feature is called received signal strength indicator (RSSI).

The RSSI is measured continuously and the value filtered using a single-pole IIR filter. After a signal level change, the RSSI will settle after approximately $RSSI_{SETTLE}$.

Sampling of the received signal strength is started by using the `RSSISTART` task. The sample can be read from the `RSSISAMPLE` register.

The sample period of the RSSI is defined by $RSSI_{PERIOD}$. The `RSSISAMPLE` will hold the filtered received signal strength after this sample period.

For the RSSI sample to be valid, RADIO has to be enabled in RX mode (`RXEN` task) and the reception has to be started (`READY` event followed by `START` task).

8.17.9 Interframe spacing (IFS)

Interframe spacing (IFS) is defined as the time, in microseconds, between two consecutive packets, starting from when the end of the last bit of the previous packet is received, to the beginning of the first bit of the subsequent packet that is transmitted.

RADIO can enforce this interval, as specified in the **TIFS** register, as long as the **TIFS** register is not specified to be shorter than RADIO's turnaround time (i.e. the time needed to switch off the receiver, and then switch the transmitter back on). The **TIFS** register can be written any time before the last bit on air is received.

This timing is illustrated in the following figure.

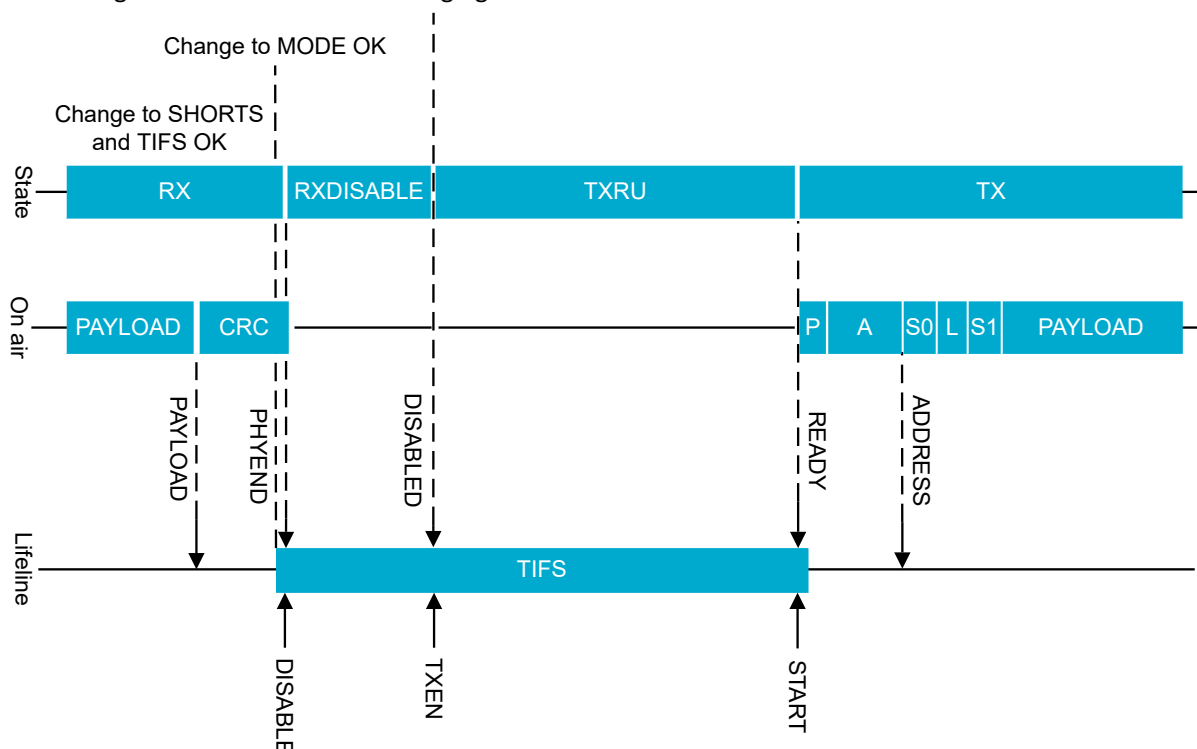


Figure 120: IFS timing detail

The TIFS duration starts after the last bit on air (at the **PHYEND** event), and elapses with the first bit being transmitted on air (just after **READY** event).

TIFS is only enforced if the shortcuts **PHYEND** to **DISABLE** and **DISABLED** to **TXEN** or **PHYEND** to **DISABLE** and **DISABLED** to **RXEN** are enabled. The short **READY** to **START** must also be enabled. In these configurations, **TXEN** or **RXEN** is automatically delayed to achieve the configured interframe spacing.

TIFS is qualified for use in IEEE 802.15.4 250kbps mode, Bluetooth Low Energy Long Range (125 kbps and 500 kbps) modes, and Bluetooth Low Energy 1 Mbps and 2 Mbps modes, using the default ramp-up mode.

SHORTS and **TIFS** registers are not double-buffered, and can be updated at any point before the last bit on air is received. The **MODE** register is double-buffered and sampled at the **TXEN** or **RXEN** task.

8.17.10 Device address match

The device address match feature is tailored for device filtering in Bluetooth Low Energy and similar implementations.

This feature enables on-the-fly device address matching while receiving a packet on air. This feature only works in receive mode and when RADIO is configured for little endian, see **PCNF1.ENDIAN** for more information.

The device address match unit assumes that the first 48 bits of the payload are the device address and that bit number 6 in S0 is the TxAdd bit. See the *Bluetooth Core Specification* for more information about device addresses, TxAdd, and device filtering procedure.

RADIO is able to listen for eight different device addresses at the same time. These addresses are specified in a DAB/DAP register pair, one pair per address, in addition to a TxAdd bit configured in the DACNF register. The DAB register specifies the 32 least significant bits of the device address, while the DAP register specifies the 16 most significant bits of the device address.

Each of the device addresses can be individually included or excluded from the matching mechanism. This is configured in the DACNF register.

8.17.11 Bit counter

RADIO implements a simple counter that can be configured to generate an event after a specific number of bits have been transmitted or received.

By using shortcuts, this counter can be started from different events generated by RADIO and count relative to these.

The bit counter is started by triggering the BCSTART task, and stopped by triggering the BCSTOP task. A BCMATCH event will be generated when the bit counter has counted the number of bits specified in the BCC register. The bit counter will continue to count bits until the DISABLED event is generated or until the BCSTOP task is triggered. After a BCMATCH event, the CPU can reconfigure the BCC value for new BCMATCH events within the same packet.

The bit counter can only be started after RADIO has received the ADDRESS event.

The bit counter will stop and reset on either the BCSTOP, STOP, or DISABLE task, or the END event.

The following figure shows how the bit counter can be used to generate a BCMATCH event in the beginning of the packet payload, and again generate a second BCMATCH event after sending 2 bytes (16 bits) of the payload.

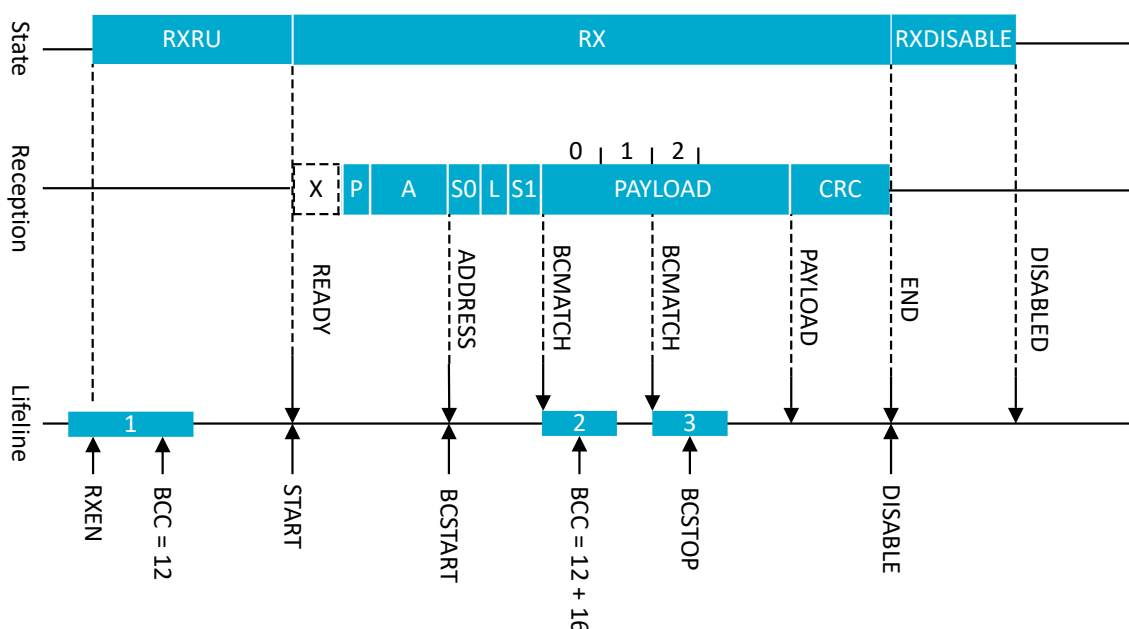


Figure 121: Bit counter example

RXRU assumes that the total combined length of S0, L, and S1 is 12 bits.

8.17.12 IEEE 802.15.4 operation

With the `MODE=ieee802154_250kbit`, RADIO will comply with the IEEE 802.15.4 standard implementing its 250 kbps, 2450 MHz, O-QPSK PHY.

IEEE standard 802.15.4 differs from Nordic's proprietary and Bluetooth Low Energy modes. Notable differences include modulation scheme, channel structure, packet structure, security, and medium access control.

The main features of the IEEE 802.15.4 mode are:

- Ultra-low power 250 kbps, 2450 MHz, IEEE 802.15.4 compliant link
- Clear channel assessment (CCA)
- Energy detection (ED) scan
- CRC generation

8.17.12.1 Packet structure

IEEE 802.15.4 defines an on-the-air frame/packet that is different from what is used in Bluetooth Low Energy.

The following figure provides an overview of the physical frame structure and its timing.

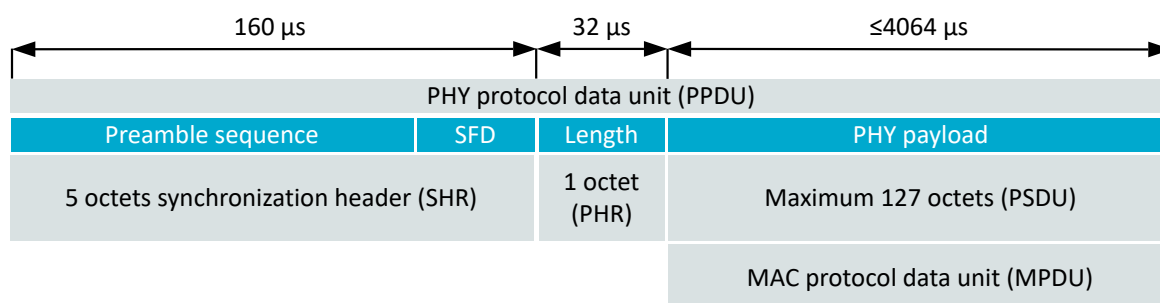


Figure 122: IEEE 802.15.4 frame format (PPDU)

The standard uses the term *octet* for an 8-bit storage unit within the PPDU. For timing, the value *symbol* is used, and it has a duration of 16 μ s.

The total usable payload (PSDU) is 127 octets, but when CRC is in use, this is reduced to 125 octets of usable payload.

The preamble sequence consists of four octets that are all zero, and are used for synchronizing RADIO's receiver. Following the preamble is the single octet *start of frame delimiter (SFD)*, with a fixed value of 0xA7. An alternate SFD can be programmed through the `SFD` register, providing an initial level of frame filtering for those who choose non-standard compliance. It is a valuable feature when operating in a congested or private network. The preamble sequence and the SFD are generated by RADIO, and are not programmed by the user into the frame buffer.

Following the five octet *synchronization header (SHR)* is the single octet *phy header (PHR)*. The least significant seven bits of PHR denote the frame length of the following PSDU. The most significant bit is reserved and is set to 0 for frames that are standard compliant. RADIO reports all eight bits which can be used to carry additional information. The PHR is the first byte written to the frame data memory pointed to by `PACKETPTR`. Frames with zero length are discarded, and the `FRAMESTART` event is not generated in this case.

The next N octets carry the data of the PHY packet, where N equals the value of the PHR. For an implementation also using the IEEE 802.15.4 medium access control (MAC) layer, the PHY data is a MAC frame of N-2 octets, since two octets occupy a CRC field.

An IEEE 802.15.4 MAC layer frame consists of the following:

- A header:
 - The frame control field (FCF)

- The sequence number
- Addressing fields
- A payload
- The 16-bit frame control sequence (FCS)

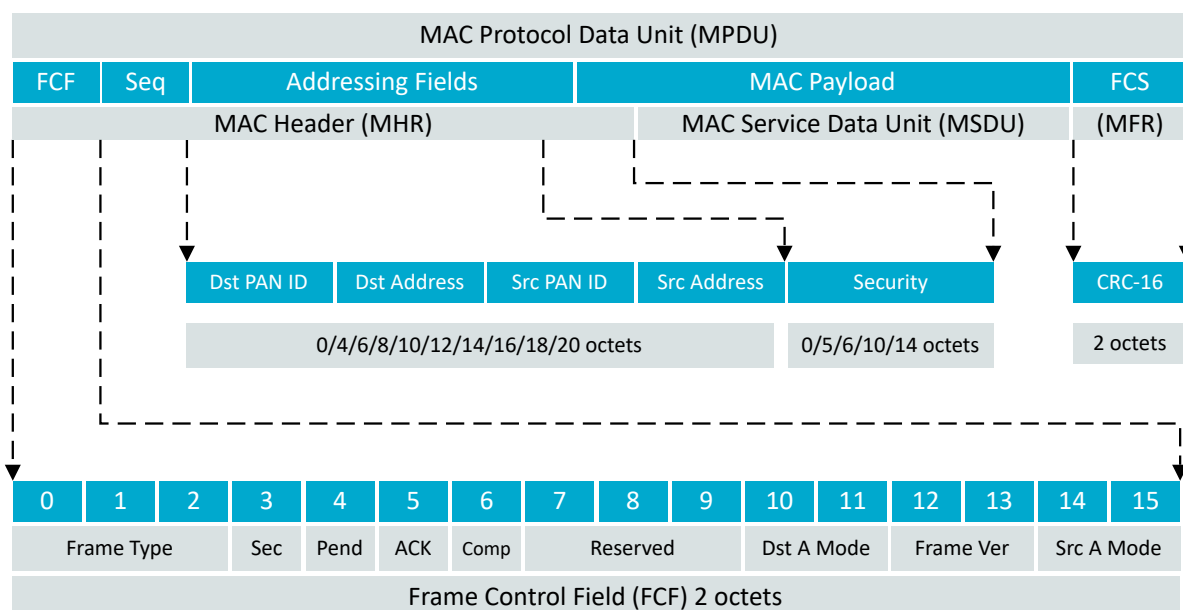


Figure 123: IEEE 802.15.4 frame format (MPDU)

The two FCF octets contain information about the frame type, addressing, and other control flags. This field is decoded when using the assisted operating modes offered by RADIO.

The sequence number is a single octet in size and is unique for a frame. It is used in the associated acknowledgement frame sent upon successful frame reception.

The addressing field can be zero (acknowledgement frame) or up to 20 octets in size. The field is used to direct packets to the correct recipient and denote its origin. IEEE 802.15.4 bases its addressing on networks being organized in PANs with 16-bit identifier and nodes having a 16-bit or 64-bit address. In the assisted receive mode, these parameters are analyzed for address matching and acknowledgement.

The MAC payload carries the data of the next higher layer, or in the case of a MAC command frame, information used by the MAC layer itself.

The two last octets contain the 16-bit ITU-T CRC. The FCS is calculated over the MAC header (MHR) and MAC payload (MSDU) parts of the frame. This field is calculated automatically when sending a frame, or indicated in the [CRCSTATUS](#) register when a frame is received. If configured, this feature is maintained autonomously by the CRC module.

8.17.12.2 Operating frequencies

IEEE 802.15.4 defines 16 channels in the 2450 MHz frequency band. The channels are numbered from 11 to 26, and each channel is 5 MHz wide.

To choose the correct channel center frequency, the [FREQUENCY](#) register must be programmed according to the following table.

IEEE 802.15.4 channel	Center frequency (MHz)	FREQUENCY setting
Channel 11	2405	5
Channel 12	2410	10
Channel 13	2415	15
Channel 14	2420	20
Channel 15	2425	25
Channel 16	2430	30
Channel 17	2435	35
Channel 18	2440	40
Channel 19	2445	45
Channel 20	2450	50
Channel 21	2455	55
Channel 22	2460	60
Channel 23	2465	65
Channel 24	2470	70
Channel 25	2475	75
Channel 26	2480	80

Table 51: IEEE 802.15.4 center frequency definition

8.17.12.3 Energy detection (ED)

As required by IEEE 802.15.4, it must be possible to sample the received signal power within the bandwidth of a channel, for the purpose of determining presence of activity.

To prevent the channel signal from being decoded, the shortcut between the **READY** event and the **START** task should be disabled before putting RADIO in receive mode. The energy detection (ED) measurement time, where RSSI samples are averaged, is 8 symbol periods, corresponding to 128 μ s. The standard further specifies the measurement to be a number between 0 and 255, where 0 shall indicate received power less than 10 dB above the selected receiver sensitivity. The power range of the ED values must be at least a 40 dB linear mapping with accuracy of ± 6 dB. See section 6.9.7 *Receiver ED* in IEEE 802.15.4 for further details.

The following example shows how to perform a single energy detection measurement and convert to IEEE 802.15.4 scale.

IEEE 802.15.4 ED measurement example

```
#define ED_RSSISCALE 4 // From electrical specifications
uint8_t sample_ed(void)
{
    int val;
    NRF_RADIO->TASKS_EDSTART = 1; // Start
    while (NRF_RADIO->EVENTS_EDEND != 1) {
        // CPU can sleep here or do something else
        // Use of interrupts are encouraged
    }
    val = NRF_RADIO->EDSAMPLE * ED_RSSISCALE; // Read level
    return (uint8_t)(val>255 ? 255 : val); // Convert to IEEE 802.15.4 scale
}
```

For scaling between hardware value and dBm, see [Clear channel assessment \(CCA\)](#) on page 468.

The `mlme-scan.req` primitive of the MAC layer uses the ED measurement to detect channels where there might be wireless activity. To assist this primitive, a tailored mode of operation is available where the ED measurement runs for a defined number of iterations keeping track of the maximum ED level. This is engaged by writing the `EDCNT` field of the `EDCTRL` register to a value different from 0, where it will run the specified number of iterations and report the maximum energy measurement in the `EDSAMPLE` register. The scan is started with `EDSTART` task and its end indicated with the `EDEND` event. This significantly reduces the interrupt frequency and therefore power consumption. The following figure shows how the ED measurement will operate depending on the `EDCNT` and `EDPERIOD` fields of the `EDCTRL` register.

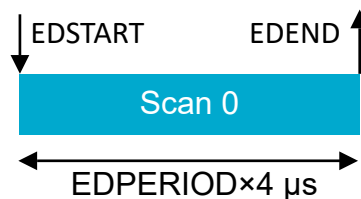


Figure 124: Energy detection measurement for a single iteration (`EDCNT=0`)

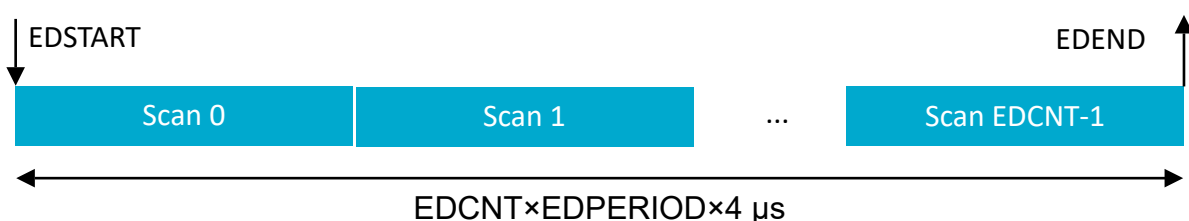


Figure 125: Energy detection measurement example with multiple iterations

The scan is stopped by writing the `EDSTOP` task. It is followed by the `EDSTOPPED` event when the module has terminated.

8.17.12.4 Clear channel assessment (CCA)

IEEE 802.15.4 implements a listen-before-talk channel access method to avoid collisions when transmitting, known as *carrier sense multiple access with collision avoidance (CSMA-CA)*. The key part of this is measuring if the wireless medium is busy or not.

The following clear channel assesment modes are supported:

- **CCA Mode 1** (energy above threshold) - The medium is reported busy upon detecting any energy above the ED threshold.

- **CCA Mode 2** (carrier sense only) - The medium is reported busy upon detection of a signal compliant with IEEE 802.15.4 with the same modulation and spreading characteristics.
- **CCA Mode 3** (carrier sense with energy above threshold) - The medium is reported busy using a logical combination (AND/OR) between the results from CCA Mode 1 and CCA Mode 2.

The clear channel assessment should survey a period equal to 8 symbols or 128 μ s.

RADIO must be in RX mode and be able to receive correct packets when performing the CCA. The shortcut between **READY** and **START** must be disabled if baseband processing is not to be performed while the measurement is running.

Register **EDSAMPLE** on page 510 is updated at the end of the clear channel assessment and can be used to read the energy level measured during the procedure. For **CCACTRL.CCAMODE**=EdModeEdModeTest1, **EDSAMPLE** holds the first ED measurement. For the other CCA modes, **EDSAMPLE** holds the average ED value.

CCA Mode 1

CCA Mode 1 is enabled by first configuring the field **CCACTRL.CCAMODE**=EdMode and writing the **CCACTRL.CCAEDTHRES** field to a chosen value. Once the **CCASTART** task is written, RADIO will perform an ED measurement for 8 symbols and compare the measured level with that found in the **CCACTRL.CCAEDTHRES** field. If the measured value is higher than or equal to this threshold, the **CCABUSY** event is generated. If the measured level is less than the threshold, the **CCAIDLE** event is generated.

CCA Mode 2

CCA Mode 2 is enabled by configuring **CCACTRL.CCAMODE**=CarrierMode. RADIO will sample to see if a valid SFD is found during the 8 symbols. If a valid SFD is detected, the **CCABUSY** event is generated and the device should not send any data. The **CCABUSY** event is also generated if the scan was performed during an ongoing frame reception. In the case where the measurement period completes with no SFD detection, the **CCAIDLE** event is generated. When **CCACTRL.CCACORRCNT** is not zero, the algorithm will look at the correlator output in addition to the SFD detection signal. If a SFD is reported during the scan period, it will terminate immediately indicating busy medium. Similarly, if the number of peaks above **CCACTRL.CCACORRTHRES** crosses the **CCACTRL.CCACORRCNT**, the **CCACTRL.CCABUSY** event is generated. If less than **CCACORRCOUNT** crossings are found and no SFD is reported, the **CCAIDLE** event will be generated and the device can send data.

CCA Mode 3

CCA Mode 3 is enabled by configuring **CCACTRL.CCAMODE**=CarrierAndEdMode or **CCACTRL.CCAMODE**=CarrierOrEdMode, performing the required logical combination of the result from CCA Mode 1 and 2. The **CCABUSY** or **CCAIDLE** events are generated by ANDing or ORing the *energy above threshold* and *carrier detection* scans.

Shortcuts

An ongoing CCA can always be stopped by issuing the **CCASTOP** task. This will trigger the associated **CCASTOPPED** event.

For CCA mode automation, the following shortcuts are available:

- To automatically switch between RX mode (when performing the CCA) and to TX mode where the packet is sent, the shortcut between **CCAIDLE** and **TXEN**, in conjunction with the short between **CCAIDLE** and **STOP** must be used.
- To automatically disable RADIO whenever the CCA reports a busy medium, the shortcut between **CCABUSY** and **DISABLE** can be used.

- To immediately start a CCA after ramping up into RX mode, the shortcut between [RXREADY](#) and [CCASTART](#) can be used.

Conversion

The conversion from a CCAEDTHRES, LQI, or EDSAMPLE value to dBm can be done with the following equation, where $VAL_{HARDWARE}$ is either CCAEDTHRES, LQI, or EDSAMPLE. LQI and EDSAMPLE are hardware-reported values, while CCAEDTHRES is set by software. Constants ED_RSSISCALE and ED_RSSIOFFS are from electrical specifications.

$$P_{RF}[dBm] = ED_RSSIOFFS + VAL_{HARDWARE}$$

The ED_RSSISCALE constant is used to calculate power in 802.15.4 units (0-255), using the following formula:

$$P_{RF}[802.15.4 \text{ units}] = \text{MIN}(ED_RSSISCALE \times VAL_{HARDWARE}, 255)$$

8.17.12.5 Cyclic redundancy check (CRC)

IEEE 802.15.4 uses a 16-bit ITU-T cyclic redundancy check (CRC) calculated over the MAC header (MHR) and MAC service data unit (MSDU).

The standard defines the following generator polynomial:

$$G(x) = x^{16} + x^{12} + x^5 + 1$$

In RX mode, RADIO will trigger the CRC module when the first octet after the frame length (PHR) is received. The CRC will then update on each consecutive octet received. When a complete frame is received the [CRCSTATUS](#) register will be updated accordingly and the [CRCOK](#) or [CRCERROR](#) events generated. When the CRC module is enabled it will not write the two last octets (CRC) to the frame Data RAM. When transmitting, the CRC will be computed on the fly, starting with the first octet after PHR, and inserted as the two last octets in the frame. The EasyDMA will fetch frame length minus 2 octets from RAM and insert the CRC octets at their correct positions in the frame.

The following code shows how to configure the CRC module for correct operation when in IEEE 802.15.4 mode. The [CRCCNF](#) is written to 16-bit CRC and the [CRCPOLY](#) is written to 0x11021. The start value used by IEEE 802.15.4 is 0 and [CRCINIT](#) is configured to reflect this.

```
/* 16-bit CRC with ITU-T polynomial with 0 as start condition*/
NRF_RADIO->CRCCNF = ((RADIO_CRCCNF_SKIPADDR_Ieee802154 << RADIO_CRCCNF_SKIPADDR_Pos) |
                    (RADIO_CRCCNF_LEN_Two << RADIO_CRCCNF_LEN_Pos));
NRF_RADIO->CRCPOLY = 0x11021;
NRF_RADIO->CRCINIT = 0;
```

The ENDIANESS subregister must be set to little-endian since the FCS field is transmitted from left bit to right.

8.17.12.6 Transmit sequence

The transmission is started by first putting RADIO in RX mode and triggering the [RXEN](#) task.

An outline of the IEEE 802.15.4 transmission is illustrated in the following figure.

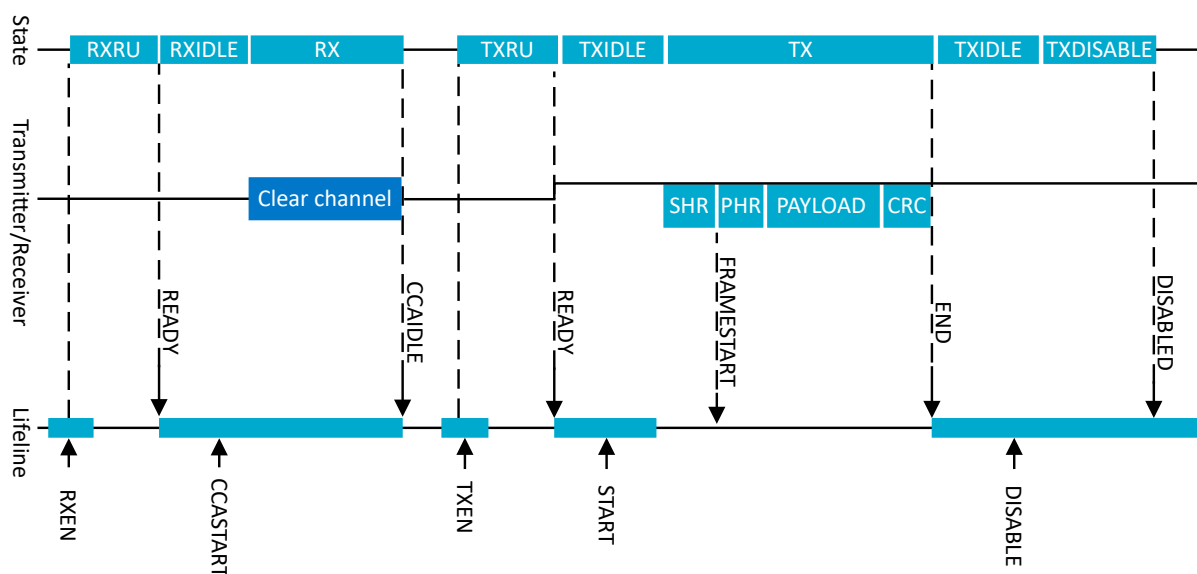


Figure 126: IEEE 802.15.4 transmit sequence

The receiver will ramp up and enter the RXIDLE state where the **READY** event is generated. Upon receiving the ready event, the CCA is started by triggering the **CCASTART** task. The chosen mode of assessment (**CCACTRL.CCAMODE** register) will be performed and signal the **CCAIDLE** or **CCABUSY** event 128 μ s later. If the **CCABUSY** event is received, RADIO will have to retry the CCA after a specific back-off period. This is outlined in the *IEEE 802.15.4 standard, Figure 69 in section 7.5.1.4 The CSMA-CA algorithm*.

If the **CCAIDLE** event is generated, a write to the **TXEN** task register enters RADIO in TXRU state. The **READY** event will be generated when RADIO is in TXIDLE state and ready to transmit. With the **PACKETPTR** pointing to the length (PHR) field of the frame, the **START** task can be written. RADIO will send the four octet preamble sequence followed by the start of frame delimiter (**SFD** register). The first byte read from the Data RAM is the length field (PHR) followed by the transmission of the number of bytes indicated as the frame length. If the CRC module is configured it will run for PHR-2 octets. The last two octets will be substituted with the results from running the CRC. The necessary CRC parameters are sampled on the **START** task. The FCS field of the frame is little endian.

In addition to the already available shortcuts, one is provided between the **READY** event and the **CCASTART** task so that a CCA can automatically start when the receiver is ready. A second shortcut has been added between the **CCAIDLE** event and the **TXEN** task, so that upon detecting a clear channel RADIO can immediately enter TX mode.

8.17.12.7 Receive sequence

The reception is started by first putting RADIO in receive mode. After writing to the **RXEN** task, RADIO will start ramping up and enter the RXRU state.

When the **READY** event is generated, RADIO enters the RXIDLE mode. For the baseband processing to be enabled, the **START** task must be written. An outline of the IEEE 802.15.4 reception can be found in the following figure.

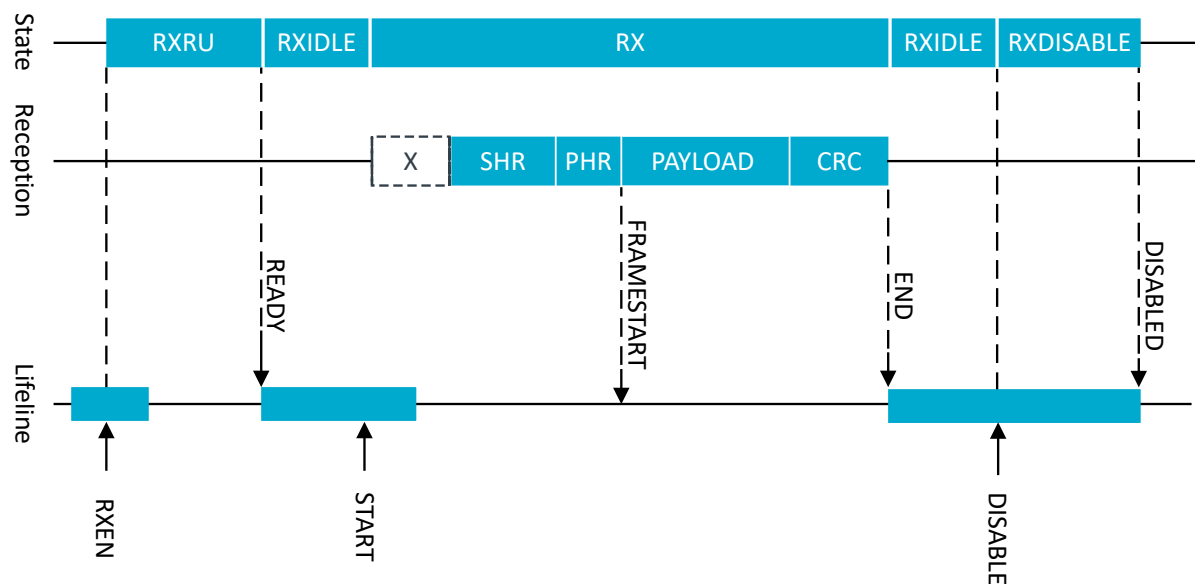


Figure 127: IEEE 802.15.4 receive sequence

When a valid SHR is received, RADIO will start storing future octets (starting with PHR) to the data memory pointed to by [PACKETPTR](#). After the SFD octet is received, the [FRAMESTART](#) event is generated. If the CRC module is enabled it will start updating with the second byte received (first byte in payload) and run for the full frame length. The two last bytes in the frame are not written to RAM when CRC is configured. However, if the result of the CRC after running the full frame is zero, the [CRCOK](#) event will be generated. The [END](#) event is generated when the last octet has been received and is available in data memory.

When a packet is received, a link quality indicator (LQI) is also generated and appended immediately after the last received octet. When using an IEEE 802.15.4 compliant frame, this will be just after the MSDU since the FCS is not reported. In the case of a non-compliant frame, it will be appended after the full frame. The LQI reported by the hardware must be converted to the IEEE 802.15.4 range by an 8-bit saturating multiplication of 4, as shown in [IEEE 802.15.4 ED measurement example](#) on page 468. The LQI is only valid for frames equal to or longer than three octets. When receiving a frame, the RSSI (reported as negative dB) will be measured at three points during the reception. These three values will be sorted and the middle one selected (median 3) to be remapped within the LQI range. The following figure illustrates the LQI measurement and how the data is arranged in data memory.

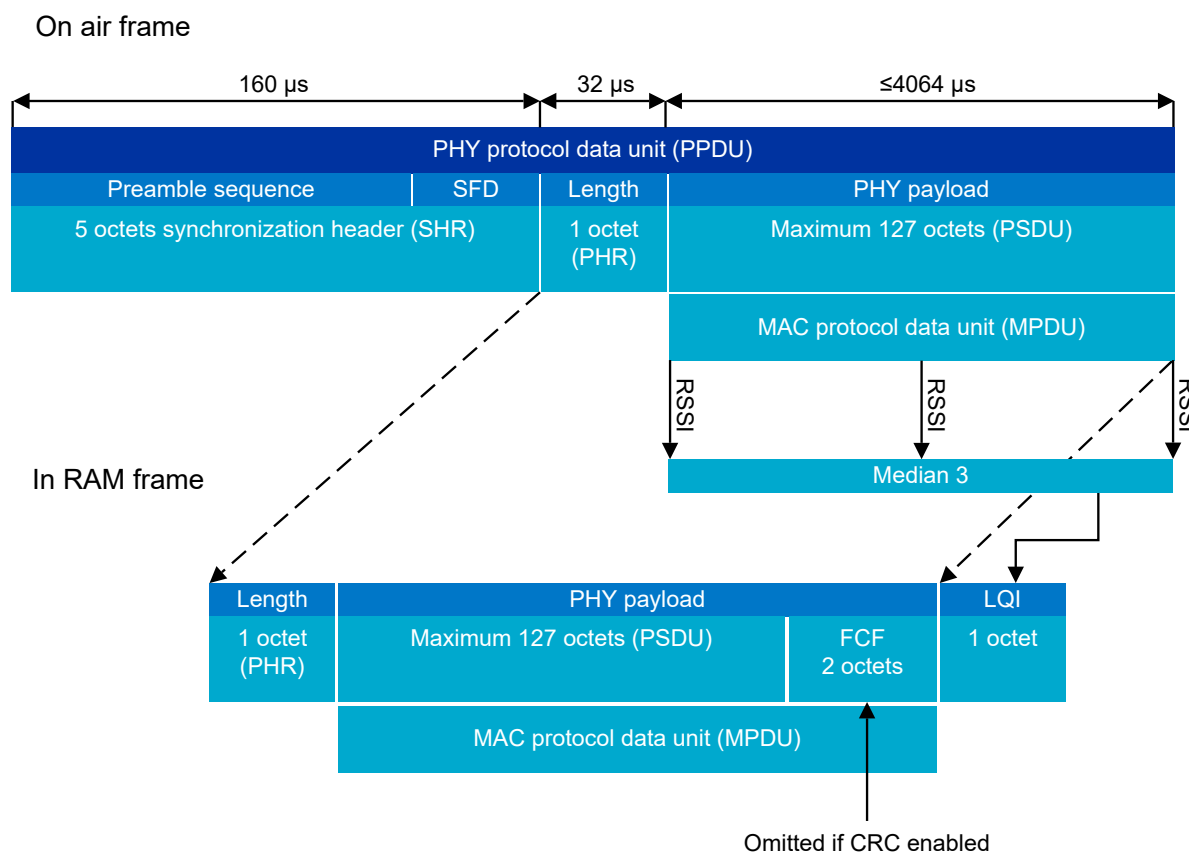


Figure 128: IEEE 802.15.4 frame in data memory

A shortcut has been added between the **FRAMESTART** event and the **BCSTART** task. This can be used to trigger a **BCMATCH** event after N bits, such as when inspecting the MAC addressing fields.

8.17.12.8 Interframe spacing (IFS)

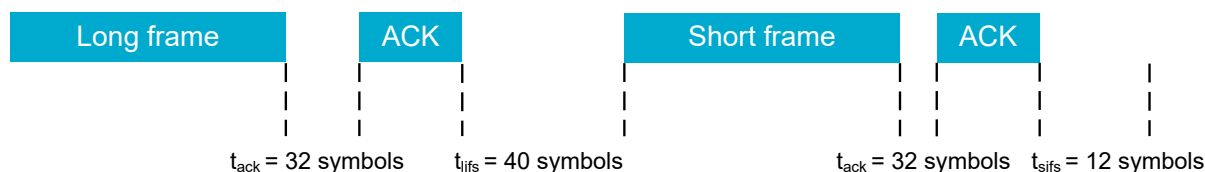
IEEE 802.15.4 defines a specific time that is allotted for the MAC sublayer to process received data. The interframe spacing (IFS) is used to prevent two frames from being transmitted too close together. If the transmission is requesting an acknowledgement, the space before the second frame shall be at least one IFS period.

IFS is determined to be one of the following:

- IFS equals `macMinSIFSPeriod` (12 symbols) if the MPDU is less than or equal to `aMaxSIFSFrameSize` (18 octets) octets
- IFS equals `macMinLIFSPeriod` (40 symbols) if the MPDU is larger than `aMaxSIFSFrameSize`

Using the efficient assisted modes in RADIO, the **TIFS** will be programmed with the correct value based on the frame being transmitted. If the assisted modes are not in use, the **TIFS** register must be updated manually. The following figure provides details on what IFS period is valid in both acknowledged and unacknowledged transmissions.

Acknowledged transmission



Unacknowledged transmission

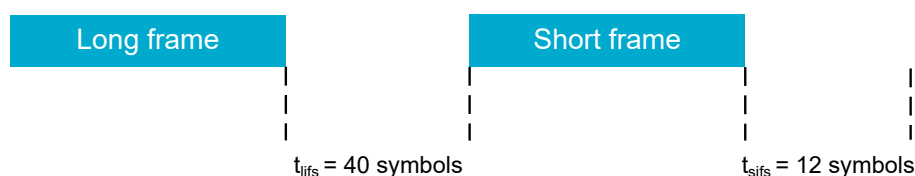


Figure 129: Interframe spacing examples

8.17.13 EasyDMA

RADIO uses EasyDMA to read and write packets to RAM without CPU involvement.

As illustrated in [RADIO block diagram](#) on page 455, RADIO's EasyDMA utilizes the same [PACKETPTR](#) for receiving and transmitting packets. This pointer should be reconfigured by the CPU each time before RADIO is started by the [START](#) task. The [PACKETPTR](#) register is double-buffered, meaning that it can be updated and prepared for the next transmission.

The [END](#) event indicates that the last bit has been processed by RADIO. The [DISABLED](#) event is issued to acknowledge that a [DISABLE](#) task is done.

The structure of a packet is described in detail in [Packet configuration](#) on page 455. The data that is stored in Data RAM and transported by EasyDMA consists of the following fields:

- S0
- LENGTH
- S1
- PAYLOAD

In addition, a static add-on is sent immediately after the payload.

The size of each of the above fields in the frame is configurable (see [Packet configuration](#) on page 455), and the space occupied in RAM depends on these settings. The size of the field can be zero, as long as the resulting frame complies with the chosen RF protocol.

All fields are extended in size to align with a byte boundary in RAM. For instance, a 3-bit long field on air will occupy 1 byte in RAM while a 9-bit long field will be extended to 2 bytes.

The packet's elements can be configured as follows:

- CI, TERM1, and TERM2 fields are only present in Bluetooth Low Energy Long Range mode
- S0 is configured through the [PCNF0.SOLEN](#) field
- LENGTH is configured through the [PCNF0.LFLEN](#) field
- S1 is configured through the [PCNF0.S1LEN](#) field
- Payload size is configured through the value in RAM corresponding to the LENGTH field
- Static add-on size is configured through the [PCNF1.STATLEN](#) field

The [PCNF1.MAXLEN](#) field configures the maximum packet payload plus add-on size in number of bytes that can be transmitted or received by RADIO. This feature can be used to ensure that RADIO does not overwrite, or read beyond, the RAM assigned to the packet payload. This means that if the LENGTH

field of the packet payload exceeds `PCNF1.STATLEN`, and the `LENGTH` field in the packet specifies a packet larger than configured in `PCNF1.MAXLEN`, the payload will be truncated to the length specified in `PCNF1.MAXLEN`.

Note: The `PCNF1.MAXLEN` field includes the payload and the add-on, but excludes the size occupied by the `S0`, `LENGTH`, and `S1` fields. This has to be taken into account when allocating RAM.

If the payload and add-on length is specified larger than `PCNF1.MAXLEN`, RADIO will still transmit or receive in the same way as before, except the payload is now truncated to `PCNF1.MAXLEN`. The packet's `LENGTH` field will not be altered when the payload is truncated. RADIO will calculate CRC as if the packet length is equal to `PCNF1.MAXLEN`.

Note: If `PACKETPTR` is not pointing to the Data RAM region, an EasyDMA transfer may result in a HardFault or RAM corruption. See [Memory](#) on page 19 for more information about the different memory regions.

The `END` event indicates that the last bit has been processed by RADIO. The `DISABLED` event is issued to acknowledge that an `DISABLE` task is done.

8.17.14 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
RADIO : S	GLOBAL	0x5008A000	US	S	SA	No	2.4 GHz radio RADIO
RADIO : NS		0x4008A000					

Configuration

Instance	Domain	Configuration
RADIO : S	GLOBAL	For the PSEL registers, use only dedicated pins on port P1
RADIO : NS		No internal instantiation of <code>DmaChannelPeripheral</code>

Register overview

Register	Offset	TZ	Description
<code>TASKS_TXEN</code>	0x000		Enable RADIO in TX mode
<code>TASKS_RXEN</code>	0x004		Enable RADIO in RX mode
<code>TASKS_START</code>	0x008		Start RADIO
<code>TASKS_STOP</code>	0x00C		Stop RADIO
<code>TASKS_DISABLE</code>	0x010		Disable RADIO
<code>TASKS_RSSISTART</code>	0x014		Start the RSSI and take one single sample of the receive signal strength
<code>TASKS_BCSTART</code>	0x018		Start the bit counter
<code>TASKS_BCSTOP</code>	0x01C		Stop the bit counter
<code>TASKS_EDSTART</code>	0x020		Start the energy detect measurement used in IEEE 802.15.4 mode
<code>TASKS_EDSTOP</code>	0x024		Stop the energy detect measurement
<code>TASKS_CCASTART</code>	0x028		Start the clear channel assessment used in IEEE 802.15.4 mode
<code>TASKS_CCACSTOP</code>	0x02C		Stop the clear channel assessment
<code>TASKS_SOFTRESET</code>	0x0A4		Reset all public registers, but with these exceptions: DMA registers and <code>EVENT/INTEN/SUBSCRIBE/PUBLISH</code> registers. Only to be used in <code>DISABLED</code> state.

Register	Offset	TZ	Description
SUBSCRIBE_TXEN	0x100		Subscribe configuration for task TXEN
SUBSCRIBE_RXEN	0x104		Subscribe configuration for task RXEN
SUBSCRIBE_START	0x108		Subscribe configuration for task START
SUBSCRIBE_STOP	0x10C		Subscribe configuration for task STOP
SUBSCRIBE_DISABLE	0x110		Subscribe configuration for task DISABLE
SUBSCRIBE_RSSISTART	0x114		Subscribe configuration for task RSSISTART
SUBSCRIBE_BCSTART	0x118		Subscribe configuration for task BCSTART
SUBSCRIBE_BCSTOP	0x11C		Subscribe configuration for task BCSTOP
SUBSCRIBE_EDSTART	0x120		Subscribe configuration for task EDSTART
SUBSCRIBE_EDSTOP	0x124		Subscribe configuration for task EDSTOP
SUBSCRIBE_CCASTART	0x128		Subscribe configuration for task CCASTART
SUBSCRIBE_CCASTOP	0x12C		Subscribe configuration for task CCASTOP
SUBSCRIBE_SOFTRESET	0x1A4		Subscribe configuration for task SOFTRESET
EVENTS_READY	0x200		RADIO has ramped up and is ready to be started
EVENTS_TXREADY	0x204		RADIO has ramped up and is ready to be started TX path
EVENTS_RXREADY	0x208		RADIO has ramped up and is ready to be started RX path
EVENTS_ADDRESS	0x20C		Address sent or received
EVENTS_FRAMESTART	0x210		IEEE 802.15.4 length field received
EVENTS_PAYLOAD	0x214		Packet payload sent or received
EVENTS_END	0x218		Memory access for packet data has been completed
EVENTS_PHYEND	0x21C		The last bit is sent on air or last bit is received
EVENTS_DISABLED	0x220		RADIO has been disabled
EVENTS_DEVMATCH	0x224		A device address match occurred on the last received packet
EVENTS_DEVMISS	0x228		No device address match occurred on the last received packet
EVENTS_CRCOK	0x22C		Packet received with CRC ok
EVENTS_CRCERROR	0x230		Packet received with CRC error
EVENTS_BCMATCH	0x238		Bit counter reached bit count value
EVENTS_EDEND	0x23C		Sampling of energy detection complete (a new ED sample is ready for readout from the RADIO.EDSAMPLE register)
EVENTS_EDSTOPPED	0x240		The sampling of energy detection has stopped
EVENTS_CCAIDLE	0x244		Wireless medium in idle - clear to send
EVENTS_CCABUSY	0x248		Wireless medium busy - do not send
EVENTS_CCASTOPPED	0x24C		The CCA has stopped
EVENTS_RATEBOOST	0x250		Ble_LR CI field received, receive mode is changed from Ble_LR125Kbit to Ble_LR500Kbit
EVENTS_MHRMATCH	0x254		MAC header match found
EVENTS_SYNC	0x258		Initial sync detected
EVENTS_CTEPRESENT	0x25C		CTEInfo byte is received
PUBLISH_READY	0x300		Publish configuration for event READY
PUBLISH_TXREADY	0x304		Publish configuration for event TXREADY
PUBLISH_RXREADY	0x308		Publish configuration for event RXREADY
PUBLISH_ADDRESS	0x30C		Publish configuration for event ADDRESS
PUBLISH_FRAMESTART	0x310		Publish configuration for event FRAMESTART
PUBLISH_PAYLOAD	0x314		Publish configuration for event PAYLOAD
PUBLISH_END	0x318		Publish configuration for event END
PUBLISH_PHYEND	0x31C		Publish configuration for event PHYEND
PUBLISH_DISABLED	0x320		Publish configuration for event DISABLED
PUBLISH_DEVMATCH	0x324		Publish configuration for event DEVMATCH
PUBLISH_DEVMISS	0x328		Publish configuration for event DEVMISS
PUBLISH_CRCOK	0x32C		Publish configuration for event CRCOK
PUBLISH_CRCERROR	0x330		Publish configuration for event CRCERROR
PUBLISH_BCMATCH	0x338		Publish configuration for event BCMATCH
PUBLISH_EDEND	0x33C		Publish configuration for event EDEND
PUBLISH_EDSTOPPED	0x340		Publish configuration for event EDSTOPPED

Register	Offset	TZ	Description
PUBLISH_CCAIDLE	0x344		Publish configuration for event CCAIDLE
PUBLISH_CCABUSY	0x348		Publish configuration for event CCABUSY
PUBLISH_CCASTOPPED	0x34C		Publish configuration for event CCASTOPPED
PUBLISH_RATEBOOST	0x350		Publish configuration for event RATEBOOST
PUBLISH_MHRMATCH	0x354		Publish configuration for event MHRMATCH
PUBLISH_SYNC	0x358		Publish configuration for event SYNC
PUBLISH_CTEPRESENT	0x35C		Publish configuration for event CTEPRESENT
SHORTS	0x400		Shortcuts between local events and tasks
INTENSET00	0x488		Enable interrupt
INTENCLR00	0x490		Disable interrupt
INTENSET10	0x4A8		Enable interrupt
INTENCLR10	0x4B0		Disable interrupt
MODE	0x500		Data rate and modulation
STATE	0x520		Current radio state
EDCTRL	0x530		IEEE 802.15.4 energy detect control
EDSAMPLE	0x534		IEEE 802.15.4 energy detect level
CCACTRL	0x538		IEEE 802.15.4 clear channel assessment control
DATAWHITE	0x540		Data whitening configuration
TIMING	0x704		Timing
FREQUENCY	0x708		Frequency
TXPOWER	0x710		Output power
TIFS	0x714		Interframe spacing in μ s
RSSISAMPLE	0x718		RSSI sample
FECONFIG	0x908		Config register
DFEMODE	0xD00		Whether to use Angle-of-Arrival (AOA) or Angle-of-Departure (AOD)
DFESTATUS	0xD04		DFE status information
DFECTRL1	0xD10		Various configuration for Direction finding
DFECTRL2	0xD14		Start offset for Direction finding
SWITCHPATTERN	0xD28		GPIO patterns to be used for each antenna
CLEARPATTERN	0xD2C		Clear the GPIO pattern array for antenna control
PSEL.DFEGPIO[n]	0xD30		Pin select for DFE pin n
DFEPACKET.PTR	0xD50		Data pointer
DFEPACKET.MAXCNT	0xD54		Maximum number of bytes to transfer
DFEPACKET.AMOUNT	0xD58		Number of bytes transferred in the last transaction
DFEPACKET.CURRENTAMOUNT	0xD5C		Number of bytes transferred in the current transaction
CRCSTATUS	0xE0C		CRC status
RXMATCH	0xE10		Received address
RXCRC	0xE14		CRC field of previously received packet
DAI	0xE18		Device address match index
PDUSTAT	0xE1C		Payload status
PCNF0	0xE20		Packet configuration register 0
PCNF1	0xE28		Packet configuration register 1
BASE0	0xE2C		Base address 0
BASE1	0xE30		Base address 1
PREFIX0	0xE34		Prefixes bytes for logical addresses 0-3
PREFIX1	0xE38		Prefixes bytes for logical addresses 4-7
TXADDRESS	0xE3C		Transmit address select
RXADDRESSES	0xE40		Receive address select
CRCCNF	0xE44		CRC configuration
CRCPOLY	0xE48		CRC polynomial
CRCINIT	0xE4C		CRC initial value
DAB[n]	0xE50		Device address base segment n
DAP[n]	0xE70		Device address prefix n

Register	Offset	TZ	Description
DACNF	0xE90		Device address match configuration
BCC	0xE94		Bit counter compare
CTESTATUS	0xEA4		CTEInfo parsed from received packet
MHRMATCHCONF	0xEB4		Search pattern configuration
MHRMATCHMASK	0xEB8		Pattern mask
SFD	0xEBC		IEEE 802.15.4 start of frame delimiter
CTEINLINECONF	0xEC0		Configuration for CTE inline mode
PACKETPTR	0xED0		Packet pointer
CSTONES.MODE	0x1000		Selects the mode(s) that are activated on the start signal
CSTONES.NUMSAMPLES	0x1004		Number of input samples at 2MHz sample rate
CSTONES.NEXTFREQUENCY	0x1008		The value of FREQUENCY that will be used in the next step
CSTONES.FFOIN	0x100C		Override value of FFO (Fractional Frequency Offset) if not to be based on the frequency estimate derived from CnAcc (autocorrelation of the scaled input signal) value
CSTONES.FFOSOURCE	0x1010		Source of FFO
CSTONES.FAEPEER	0x1014		FAEPEER (Frequency Actuation Error) of peer if known. Used during Mode 0 steps.
CSTONES.PHASESHIFT	0x1018		Parameter used in TPM, provided by software
CSTONES.NUMSAMPLESCOEFF	0x101C		Parameter used in TPM, provided by software
CSTONES.PCT16	0x1020		Mean magnitude and mean phase converted to IQ
CSTONES.MAGPHASEMEAN	0x1024		Mean magnitude and phase of the signal before it is converted to PCT16
CSTONES.IQRAWMEAN	0x1028		Mean of IQ values
CSTONES.MAGSTD	0x102C		Magnitude standard deviation approximation
CSTONES.CNACC	0x1030		Output of the autocorrelation of the accumulated IQ signal
CSTONES.FFOEST	0x1034		FFO estimate
CSTONES.DOWNSAMPLE	0x1038		Turn on/off down sample of input IQ-signals
CSTONES.FINETUNENEXT	0x103C		Number of full ADPLL finetune steps
CSTONES.CFOPHASE	0x1040		Cordic output of CnAcc
CSTONES.FREQOFFSET	0x1044		Frequency offset estimate
CSTONES.PCT11	0x1048		Mean magnitude and mean phase converted to IQ. IQ values limited to [-1024,1023].
CSTONES.LFAENEXT	0x104C		Quantization error between ADPLL frequency and the desired value of FFO * RF Frequency. Values limited to [-64,63] with units 7.6294 Hz.
RTT.CONFIG	0x1050		RTT Config.
RTT.SEGMENT01	0x1054		RTT segments 0 and 1
RTT.SEGMENT23	0x1058		RTT segments 2 and 3
RTT.SEGMENT45	0x105C		RTT segments 4 and 5
RTT.SEGMENT67	0x1060		RTT segments 6 and 7

8.17.14.1 TASKS_TXEN

Address offset: 0x000

Enable RADIO in TX mode

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_TXEN						Enable RADIO in TX mode																											
			Trigger	1				Trigger task																											

8.17.14.2 TASKS_RXEN

Address offset: 0x004

Enable RADIO in RX mode

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_RXEN						Enable RADIO in RX mode																											
			Trigger	1				Trigger task																											

8.17.14.3 TASKS_START

Address offset: 0x008

Start RADIO

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_START						Start RADIO																											
			Trigger	1				Trigger task																											

8.17.14.4 TASKS_STOP

Address offset: 0x00C

Stop RADIO

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	TASKS_STOP			Stop RADIO																															
			Trigger	1	Trigger task																															

8.17.14.5 TASKS_DISABLE

Address offset: 0x010

Disable RADIO

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																								A			
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID		Value					Description																																	
A	W	TASKS_DISABLE								Disable RADIO																																	
			Trigger	1						Trigger task																																	

8.17.14.6 TASKS_RSSISTART

Address offset: 0x014

Start the RSSI and take one single sample of the receive signal strength

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																					A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value	Description																																
A	W	TASKS_RSSISTART			Start the RSSI and take one single sample of the receive signal strength																																
			Trigger	1	Trigger task																																

8.17.14.7 TASKS_BCSTART

Address offset: 0x018

Start the bit counter

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																							A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value	Description																																		
A	W	TASKS_BCSTART			Start the bit counter																																		
			Trigger	1	Trigger task																																		

8.17.14.8 TASKS_BCSTOP

Address offset: 0x01C

Stop the bit counter

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_BCSTOP						Stop the bit counter																											
			Trigger	1				Trigger task																											

8.17.14.9 TASKS_EDSTART

Address offset: 0x020

Start the energy detect measurement used in IEEE 802.15.4 mode

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																					A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																
A	W	TASKS_EDSTART			Start the energy detect measurement used in IEEE 802.15.4 mode																																
			Trigger	1	Trigger task																																

8.17.14.10 TASKS_EDSTOP

Address offset: 0x024

Stop the energy detect measurement

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_EDSTOP						Stop the energy detect measurement																											
			Trigger	1				Trigger task																											

8.17.14.11 TASKS_CCSTART

Address offset: 0x028

Start the clear channel assessment used in IEEE 802.15.4 mode

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																					A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value	Description																																
A	W	TASKS_CCSTART			Start the clear channel assessment used in IEEE 802.15.4 mode																																
			Trigger	1	Trigger task																																

8.17.14.12 TASKS_CCSTOP

Address offset: 0x02C

Stop the clear channel assessment

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	TASKS_CCSTOP			Stop the clear channel assessment																															
			Trigger	1	Trigger task																															

8.17.14.13 TASKS_SOFTRESET

Address offset: 0x0A4

Reset all public registers, but with these exceptions: DMA registers and EVENT/INTEN/SUBSCRIBE/PUBLISH registers. Only to be used in DISABLED state.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	W	TASKS_SOFTRESET			Reset all public registers, but with these exceptions: DMA registers and EVENT/INTEN/SUBSCRIBE/PUBLISH registers. Only to be used in DISABLED state.																														
			Trigger	1	Trigger task																														

8.17.14.14 SUBSCRIBE_TXEN

Address offset: 0x100

Subscribe configuration for task [TXEN](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task TXEN will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.17.14.15 SUBSCRIBE_RXEN

Address offset: 0x104

Subscribe configuration for task **RXEN**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task RXEN will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.17.14.16 SUBSCRIBE_START

Address offset: 0x108

Subscribe configuration for task **START**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task START will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.17.14.17 SUBSCRIBE_STOP

Address offset: 0x10C

Subscribe configuration for task **STOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task STOP will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.17.14.18 SUBSCRIBE_DISABLE

Address offset: 0x110

Subscribe configuration for task **DISABLE**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task DISABLE will subscribe to																													
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.17.14.19 SUBSCRIBE_RSSISTART

Address offset: 0x114

Subscribe configuration for task **RSSISTART**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task RSSISTART will subscribe to																													
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.17.14.20 SUBSCRIBE_BCSTART

Address offset: 0x118

Subscribe configuration for task **BCSTART**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task BCSTART will subscribe to																													
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.17.14.21 SUBSCRIBE_BCSTOP

Address offset: 0x11C

Subscribe configuration for task **BCSTOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task BCSTOP will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.17.14.22 SUBSCRIBE_EDSTART

Address offset: 0x120

Subscribe configuration for task **EDSTART**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CHIDX		[0..255]	DPPI channel that task EDSTART will subscribe to																														
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.17.14.23 SUBSCRIBE_EDSTOP

Address offset: 0x124

Subscribe configuration for task **EDSTOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task EDSTOP will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.17.14.24 SUBSCRIBE_CCASTART

Address offset: 0x128

Subscribe configuration for task **CCASTART**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task CCASTART will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.17.14.25 SUBSCRIBE_CCASTOP

Address offset: 0x12C

Subscribe configuration for task CCASTOP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task CCASTOP will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.17.14.26 SUBSCRIBE_SOFTRESET

Address offset: 0x1A4

Subscribe configuration for task SOFTRESET

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task SOFTRESET will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.17.14.27 EVENTS_READY

Address offset: 0x200

RADIO has ramped up and is ready to be started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_READY			RADIO has ramped up and is ready to be started																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.28 EVENTS_TXREADY

Address offset: 0x204

RADIO has ramped up and is ready to be started TX path

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_TXREADY			RADIO has ramped up and is ready to be started TX path																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.29 EVENTS_RXREADY

Address offset: 0x208

RADIO has ramped up and is ready to be started RX path

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_RXREADY			RADIO has ramped up and is ready to be started RX path																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.30 EVENTS_ADDRESS

Address offset: 0x20C

Address sent or received

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_ADDRESS			Address sent or received																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.31 EVENTS_FRAMESTART

Address offset: 0x210

IEEE 802.15.4 length field received

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_FRAMESTART			IEEE 802.15.4 length field received																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.32 EVENTS_PAYLOAD

Address offset: 0x214

Packet payload sent or received

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_PAYLOAD			Packet payload sent or received																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.33 EVENTS_END

Address offset: 0x218

Memory access for packet data has been completed

In TX: Last byte to be transmitted has been fetched from RAM

In RX: Last byte received on air has been stored to RAM

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_END			Memory access for packet data has been completed																														
					In TX: Last byte to be transmitted has been fetched from RAM																														
					In RX: Last byte received on air has been stored to RAM																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.34 EVENTS_PHYEND

Address offset: 0x21C

The last bit is sent on air or last bit is received

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_PHYEND			The last bit is sent on air or last bit is received																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.35 EVENTS_DISABLED

Address offset: 0x220

RADIO has been disabled

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_DISABLED						RADIO has been disabled																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.17.14.36 EVENTS_DEVMATCH

Address offset: 0x224

A device address match occurred on the last received packet

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0 0																																	
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	EVENTS_DEVMATCH			A device address match occurred on the last received packet																																	
			NotGenerated	0	Event not generated																																	
			Generated	1	Event generated																																	

8.17.14.37 EVENTS_DEVMISS

Address offset: 0x228

No device address match occurred on the last received packet

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	EVENTS_DEVMISS			No device address match occurred on the last received packet																																	
			NotGenerated	0	Event not generated																																	
			Generated	1	Event generated																																	

8.17.14.38 EVENTS_CRCOK

Address offset: 0x22C

Packet received with CRC ok

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_CRCOK			Packet received with CRC ok																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.39 EVENTS_CRCERROR

Address offset: 0x230

Packet received with CRC error

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_CRCERROR						Packet received with CRC error																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.17.14.40 EVENTS_BCMATCH

Address offset: 0x238

Bit counter reached bit count value

Bit counter value is specified in the RADIO.BCC register

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_BCMATCH						Bit counter reached bit count value																											
								Bit counter value is specified in the RADIO.BCC register																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.17.14.41 EVENTS_EDEND

Address offset: 0x23C

Sampling of energy detection complete (a new ED sample is ready for readout from the RADIO.EDSAMPLE register)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_EDEND			Sampling of energy detection complete (a new ED sample is ready for readout from the RADIO.EDSAMPLE register)																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.42 EVENTS_EDSTOPPED

Address offset: 0x240

The sampling of energy detection has stopped

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_EDSTOPPED						The sampling of energy detection has stopped																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.17.14.43 EVENTS_CCAIDLE

Address offset: 0x244

Wireless medium in idle - clear to send

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				A																																		
Reset 0x00000000				0 0																																		
ID	R/W	Field	Value ID	Value		Description																																
A	RW	EVENTS_CCAIDLE				Wireless medium in idle - clear to send																																
			NotGenerated	0	Event not generated																																	
			Generated	1	Event generated																																	

8.17.14.44 EVENTS_CCABUSY

Address offset: 0x248

Wireless medium busy - do not send

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																													
A	RW	EVENTS_CCABUSY				Wireless medium busy - do not send																													
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.45 EVENTS_CCASTOPPED

Address offset: 0x24C

The CCA has stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_CCASTOPPED			The CCA has stopped																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.46 EVENTS_RATEBOOST

Address offset: 0x250

Ble_LR CI field received, receive mode is changed from Ble_LR125Kbit to Ble_LR500Kbit

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																																							A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	EVENTS_RATEBOOST				Ble_LR CI field received, receive mode is changed from Ble_LR125Kbit to Ble_LR500Kbit																																	
			NotGenerated	0		Event not generated																																	
			Generated	1		Event generated																																	

8.17.14.47 EVENTS_MHRMATCH

Address offset: 0x254

MAC header match found

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_MHRMATCH			MAC header match found																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.17.14.48 EVENTS_SYNC

Address offset: 0x258

Initial sync detected

MODE=Ble_LR125Kbit, Ble_LR500Kbit, or leee802154_250Kbit: A possible preamble has been received. However, due to the sporadic reception of noise, this event can be falsely triggered.

For MODE=Nrf_1Mbit, Nrf_2Mbit, Ble_1Mbit, or Ble_2Mbit: A possible preamble and the first two bytes of the address field has been received. The event can be generated falsely also in this mode.

It is also possible that the event is not generated, or not generated before the ADDRESS event.

Bit number			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID			A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	EVENTS_SYNC			Initial sync detected																													
					MODE=Ble_LR125Kbit, Ble_LR500Kbit, or leee802154_250Kbit: A possible preamble has been received. However, due to the sporadic reception of noise, this event can be falsely triggered.																													
					For MODE=Nrf_1Mbit, Nrf_2Mbit, Ble_1Mbit, or Ble_2Mbit: A possible preamble and the first two bytes of the address field has been received. The event can be generated falsely also in this mode.																													
					It is also possible that the event is not generated, or not generated before the ADDRESS event.																													
			NotGenerated	0	Event not generated																													
			Generated	1	Event generated																													

8.17.14.49 EVENTS_CTEPRESENT

Address offset: 0x25C

CTEInfo byte is received

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_CTEPRESENT						CTEInfo byte is received																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.17.14.50 PUBLISH_READY

Address offset: 0x300

Publish configuration for event **READY**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event READY will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.17.14.51 PUBLISH_TXREADY

Address offset: 0x304

Publish configuration for event **TXREADY**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event TXREADY will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.17.14.52 PUBLISH_RXREADY

Address offset: 0x308

Publish configuration for event **RXREADY**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event RXREADY will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.17.14.53 PUBLISH_ADDRESS

Address offset: 0x30C

Publish configuration for event **ADDRESS**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				B																								A					A	A	A	A	A	A	A
Reset 0x00000000				0																																			
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	CHIDX		[0..255]		DPPI channel that event ADDRESS will publish to																																	
B	RW	EN																																					
			Disabled	0	Disable publishing																																		
			Enabled	1	Enable publishing																																		

8.17.14.54 PUBLISH_FRAMESTART

Address offset: 0x310

Publish configuration for event [FRAMESTART](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																													A	A	A	A	A	A	A	A																							
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																							
A	RW	CHIDX		[0..255]				DPPI channel that event FRAMESTART will publish to																																																							
B	RW	EN																																																													
			Disabled	0				Disable publishing																																																							
			Enabled	1				Enable publishing																																																							

8.17.14.55 PUBLISH_PAYLOAD

Address offset: 0x314

Publish configuration for event [PAYLOAD](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																										
ID				B																													A	A	A	A	A	A	A	A																					
Reset 0x00000000				0																																					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																							
A	RW	CHIDX		[0..255]		DPPI channel that event PAYLOAD will publish to																																																							
B	RW	EN																																																											
			Disabled	0	Disable publishing																																																								
			Enabled	1	Enable publishing																																																								

8.17.14.56 PUBLISH_END

Address offset: 0x318

Publish configuration for event [END](#)

In TX: Last byte to be transmitted has been fetched from RAM

In RX: Last byte received on air has been stored to RAM

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event END will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.17.14.57 PUBLISH_PHYEND

Address offset: 0x31C

Publish configuration for event **PHYEND**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event PHYEND will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.17.14.58 PUBLISH_DISABLED

Address offset: 0x320

Publish configuration for event **DISABLED**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event DISABLED will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.17.14.59 PUBLISH_DEVMATCH

Address offset: 0x324

Publish configuration for event **DEVMATCH**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event DEVMATCH will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.17.14.60 PUBLISH_DEVMISS

Address offset: 0x328

Publish configuration for event [DEVMISS](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event DEVMISS will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.17.14.61 PUBLISH_CRCOK

Address offset: 0x32C

Publish configuration for event [CRCOK](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event CRCOK will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.17.14.62 PUBLISH_CRCERROR

Address offset: 0x330

Publish configuration for event [CRCERROR](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event CRCERROR will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.17.14.63 PUBLISH BCMATCH

Address offset: 0x338

Publish configuration for event [BCMATCH](#)

Bit counter value is specified in the RADIO.BCC register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event BCMATCH will publish to																													
B	RW	EN																																	
			Disabled	0		Disable publishing																													
			Enabled	1		Enable publishing																													

8.17.14.64 PUBLISH_EDEND

Address offset: 0x33C

Publish configuration for event **EDEND**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event EDEND will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.17.14.65 PUBLISH_EDSTOPPED

Address offset: 0x340

Publish configuration for event **EDSTOPPED**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event EDSTOPPED will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.17.14.66 PUBLISH_CCAIDLE

Address offset: 0x344

Publish configuration for event **CCAIDLE**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event CCAIDLE will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.17.14.67 PUBLISH_CCABUSY

Address offset: 0x348

Publish configuration for event [CCABUSY](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event CCABUSY will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.17.14.68 PUBLISH_CCASTOPPED

Address offset: 0x34C

Publish configuration for event [CCASTOPPED](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event CCASTOPPED will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.17.14.69 PUBLISH_RATEBOOST

Address offset: 0x350

Publish configuration for event [RATEBOOST](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event RATEBOOST will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.17.14.70 PUBLISH_MHRMATCH

Address offset: 0x354

Publish configuration for event [MHRMATCH](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event MHRMATCH will publish to																													
B	RW	EN																																	
			Disabled	0		Disable publishing																													
			Enabled	1		Enable publishing																													

8.17.14.71 PUBLISH_SYNC

Address offset: 0x358

Publish configuration for event **SYNC**

MODE=Ble_LR125Kbit, Ble_LR500Kbit, or leee802154_250Kbit: A possible preamble has been received. However, due to the sporadic reception of noise, this event can be falsely triggered.

For MODE=Nrf_1Mbit, Nrf_2Mbit, Ble_1Mbit, or Ble_2Mbit: A possible preamble and the first two bytes of the address field has been received. The event can be generated falsely also in this mode.

It is also possible that the event is not generated, or not generated before the ADDRESS event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event SYNC will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.17.14.72 PUBLISH_CTEPRESENT

Address offset: 0x35C

Publish configuration for event **CTEPRESENT**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event CTEPRESENT will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.17.14.73 SHORTS

Address offset: 0x400

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				Q P O N M L K J I H G																F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY_START			Shortcut between event READY and task START																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
B	RW	DISABLED_TXEN			Shortcut between event DISABLED and task TXEN																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
C	RW	DISABLED_RXEN			Shortcut between event DISABLED and task RXEN																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
D	RW	ADDRESS_RSISTART			Shortcut between event ADDRESS and task RSSISTART																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
E	RW	END_START			Shortcut between event END and task START																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
F	RW	ADDRESS_BCSTART			Shortcut between event ADDRESS and task BCSTART																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
G	RW	RXREADY_CCSTART			Shortcut between event RXREADY and task CCSTART																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
H	RW	CCAIDLE_TXEN			Shortcut between event CCAIDLE and task TXEN																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
I	RW	CCABUSY_DISABLE			Shortcut between event CCABUSY and task DISABLE																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
J	RW	FRAMESTART_BCSTART			Shortcut between event FRAMESTART and task BCSTART																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
K	RW	READY_EDSTART			Shortcut between event READY and task EDSTART																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
L	RW	EDEND_DISABLE			Shortcut between event EDEND and task DISABLE																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
M	RW	CCAIDLE_STOP			Shortcut between event CCAIDLE and task STOP																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
N	RW	TXREADY_START			Shortcut between event TXREADY and task START																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
O	RW	RXREADY_START			Shortcut between event RXREADY and task START																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
P	RW	PHYEND_DISABLE			Shortcut between event PHYEND and task DISABLE																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
Q	RW	PHYEND_START			Shortcut between event PHYEND and task START																														
			Disabled	0	Disable shortcut																														

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				Q P O N M L K J I H G																F E D C B A															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
			Enabled	1				Enable shortcut																											

8.17.14.74 INTENSET00

Address offset: 0x488

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				W V U T S R Q P O N																M L K J I H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Write '1' to enable interrupt for event READY																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	TXREADY			Write '1' to enable interrupt for event TXREADY																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	RXREADY			Write '1' to enable interrupt for event RXREADY																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	ADDRESS			Write '1' to enable interrupt for event ADDRESS																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
E	RW	FRAMESTART			Write '1' to enable interrupt for event FRAMESTART																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
F	RW	PAYLOAD			Write '1' to enable interrupt for event PAYLOAD																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
G	RW	END			Write '1' to enable interrupt for event END																														
					In TX: Last byte to be transmitted has been fetched from RAM																														
					In RX: Last byte received on air has been stored to RAM																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			H	RW	PHYEND			Write '1' to enable interrupt for event PHYEND																											
						Set	1	Enable																											
Disabled	0	Read: Disabled																																	
Enabled	1	Read: Enabled																																	
I	RW	DISABLED			Write '1' to enable interrupt for event DISABLED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

4503_018 v0.8

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				W V U T S R Q P O N																M L K J I H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
V	RW	SYNC			Write '1' to enable interrupt for event SYNC																														
					MODE=Ble_LR125Kbit, Ble_LR500Kbit, or leee802154_250Kbit: A possible preamble has been received. However, due to the sporadic reception of noise, this event can be falsely triggered.																														
					For MODE=Nrf_1Mbit, Nrf_2Mbit, Ble_1Mbit, or Ble_2Mbit: A possible preamble and the first two bytes of the address field has been received. The event can be generated falsely also in this mode.																														
					It is also possible that the event is not generated, or not generated before the ADDRESS event.																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
W	RW	CTEPRESENT			Write '1' to enable interrupt for event CTEPRESENT																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.17.14.75 INTENCLR00

Address offset: 0x490

Disable interrupt

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				W V U T S R Q P O N																M L K J I H G F E D C B A																	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value	ID	Value	Description																															
A	RW	READY				Write '1' to disable interrupt for event READY																															
			Clear	1	Disable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
B	RW	TXREADY				Write '1' to disable interrupt for event TXREADY																															
			Clear	1	Disable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
C	RW	RXREADY				Write '1' to disable interrupt for event RXREADY																															
			Clear	1	Disable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
D	RW	ADDRESS				Write '1' to disable interrupt for event ADDRESS																															
			Clear	1	Disable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
E	RW	FRAMESTART				Write '1' to disable interrupt for event FRAMESTART																															
			Clear	1	Disable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
F	RW	PAYLOAD				Write '1' to disable interrupt for event PAYLOAD																															
			Clear	1	Disable																																

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
			G RW END			Write '1' to disable interrupt for event END																												
						In TX: Last byte to be transmitted has been fetched from RAM																												
					In RX: Last byte received on air has been stored to RAM																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
H	RW	PHYEND	Write '1' to disable interrupt for event PHYEND																															
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
I	RW	DISABLED	Write '1' to disable interrupt for event DISABLED																															
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
J	RW	DEVMATCH	Write '1' to disable interrupt for event DEVMATCH																															
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
K	RW	DEVMISS	Write '1' to disable interrupt for event DEVMISS																															
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
L	RW	CRCOK	Write '1' to disable interrupt for event CRCOK																															
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
M	RW	CRCERROR	Write '1' to disable interrupt for event CRCERROR																															
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
N	RW	BCMATCH	Write '1' to disable interrupt for event BCMATCH																															
			Bit counter value is specified in the RADIO.BCC register																															
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
			O	RW	EDEND	Write '1' to disable interrupt for event EDEND																												
Clear	1	Disable																																
Disabled	0	Read: Disabled																																
Enabled	1	Read: Enabled																																
P	RW	EDSTOPPED	Write '1' to disable interrupt for event EDSTOPPED																															
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
Q	RW	CCAIDLE	Write '1' to disable interrupt for event CCAIDLE																															
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
R	RW	CCABUSY			Write '1' to disable interrupt for event CCABUSY																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
S	RW	CCASTOPPED			Write '1' to disable interrupt for event CCASTOPPED																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
T	RW	RATEBOOST			Write '1' to disable interrupt for event RATEBOOST																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
U	RW	MHRMATCH			Write '1' to disable interrupt for event MHRMATCH																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
V	RW	SYNC			Write '1' to disable interrupt for event SYNC																													
					MODE=Ble_LR125Kbit, Ble_LR500Kbit, or leee802154_250Kbit: A possible preamble has been received. However, due to the sporadic reception of noise, this event can be falsely triggered.																													
					For MODE=Nrf_1Mbit, Nrf_2Mbit, Ble_1Mbit, or Ble_2Mbit: A possible preamble and the first two bytes of the address field has been received. The event can be generated falsely also in this mode.																													
					It is also possible that the event is not generated, or not generated before the ADDRESS event.																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
W	RW	CTEPRESENT			Write '1' to disable interrupt for event CTEPRESENT																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

8.17.14.76 INTENSET10

Address offset: 0x4A8

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Write '1' to enable interrupt for event READY																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	TXREADY			Write '1' to enable interrupt for event TXREADY																														
			Set	1	Enable																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			C	RW	RXREADY		Write '1' to enable interrupt for event RXREADY																												
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			D	RW	ADDRESS		Write '1' to enable interrupt for event ADDRESS																												
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			E	RW	FRAMESTART		Write '1' to enable interrupt for event FRAMESTART																												
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			F	RW	PAYLOAD		Write '1' to enable interrupt for event PAYLOAD																												
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			G	RW	END		Write '1' to enable interrupt for event END																												
							In TX: Last byte to be transmitted has been fetched from RAM																												
					In RX: Last byte received on air has been stored to RAM																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			H	RW	PHYEND		Write '1' to enable interrupt for event PHYEND																												
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			I	RW	DISABLED		Write '1' to enable interrupt for event DISABLED																												
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			J	RW	DEVMATCH		Write '1' to enable interrupt for event DEVMATCH																												
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			K	RW	DEVMISS		Write '1' to enable interrupt for event DEVMISS																												
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			L	RW	CRCOK		Write '1' to enable interrupt for event CRCOK																												
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			M	RW	CRCERROR		Write '1' to enable interrupt for event CRCERROR																												
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
N	RW	BCMATCH			Write '1' to enable interrupt for event BCMATCH																														
					Bit counter value is specified in the RADIO.BCC register																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
		Enabled	1	Read: Enabled																															
O	RW	EDEND			Write '1' to enable interrupt for event EDEND																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
P	RW	EDSTOPPED			Write '1' to enable interrupt for event EDSTOPPED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
Q	RW	CCAIDLE			Write '1' to enable interrupt for event CCAIDLE																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
R	RW	CCABUSY			Write '1' to enable interrupt for event CCABUSY																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
S	RW	CCASTOPPED			Write '1' to enable interrupt for event CCASTOPPED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
T	RW	RATEBOOST			Write '1' to enable interrupt for event RATEBOOST																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
U	RW	MHRMATCH			Write '1' to enable interrupt for event MHRMATCH																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
V	RW	SYNC			Write '1' to enable interrupt for event SYNC																														
					MODE=Ble_LR125Kbit, Ble_LR500Kbit, or leee802154_250Kbit: A possible preamble has been received. However, due to the sporadic reception of noise, this event can be falsely triggered.																														
					For MODE=Nrf_1Mbit, Nrf_2Mbit, Ble_1Mbit, or Ble_2Mbit: A possible preamble and the first two bytes of the address field has been received. The event can be generated falsely also in this mode.																														
					It is also possible that the event is not generated, or not generated before the ADDRESS event.																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			W	RW	CTEPRESENT			Write '1' to enable interrupt for event CTEPRESENT																											
Set	1	Enable																																	
Disabled	0	Read: Disabled																																	
Enabled	1	Read: Enabled																																	

Address offset: 0x4B0

4503_018 v0.8

507



NORDIC

 SEMICONDUCTOR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
L	RW	CRCOK	Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			Write '1' to disable interrupt for event CRCOK																																
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
M	RW	CRCERROR	Enabled	1	Read: Enabled																														
			Write '1' to disable interrupt for event CRCERROR																																
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
N	RW	BCMATCH	Write '1' to disable interrupt for event BCMATCH																																
			Bit counter value is specified in the RADIO.BCC register																																
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
O	RW	EDEND	Write '1' to disable interrupt for event EDEND																																
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			P	RW	EDSTOPPED	Write '1' to disable interrupt for event EDSTOPPED																													
Clear	1	Disable																																	
Disabled	0	Read: Disabled																																	
Enabled	1	Read: Enabled																																	
Q	RW	CCAIDLE				Write '1' to disable interrupt for event CCAIDLE																													
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			R	RW	CCABUSY	Write '1' to disable interrupt for event CCABUSY																													
Clear	1	Disable																																	
Disabled	0	Read: Disabled																																	
Enabled	1	Read: Enabled																																	
S	RW	CCASTOPPED				Write '1' to disable interrupt for event CCASTOPPED																													
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			T	RW	RATEBOOST	Write '1' to disable interrupt for event RATEBOOST																													
Clear	1	Disable																																	
Disabled	0	Read: Disabled																																	
Enabled	1	Read: Enabled																																	
U	RW	MHRMATCH				Write '1' to disable interrupt for event MHRMATCH																													
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			W V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
V	RW	SYNC			Write '1' to disable interrupt for event SYNC																													
					MODE=Ble_LR125Kbit, Ble_LR500Kbit, or leee802154_250Kbit: A possible preamble has been received. However, due to the sporadic reception of noise, this event can be falsely triggered.																													
					For MODE=Nrf_1Mbit, Nrf_2Mbit, Ble_1Mbit, or Ble_2Mbit: A possible preamble and the first two bytes of the address field has been received. The event can be generated falsely also in this mode.																													
					It is also possible that the event is not generated, or not generated before the ADDRESS event.																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
W	RW	CTEPRESENT			Write '1' to disable interrupt for event CTEPRESENT																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

8.17.14.78 MODE

Address offset: 0x500

Data rate and modulation

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	MODE			Radio data rate and modulation setting. The radio supports frequency-shift keying (FSK) modulation.																														
			Nrf_1Mbit	0	1 Mbps Nordic proprietary radio mode																														
			Nrf_2Mbit	1	2 Mbps Nordic proprietary radio mode																														
			Ble_1Mbit	3	1 Mbps BLE																														
			Ble_2Mbit	4	2 Mbps BLE																														
			Ble_LR125Kbit	5	Long range 125 kbps TX, 125 kbps and 500 kbps RX																														
			Ble_LR500Kbit	6	Long range 500 kbps TX, 125 kbps and 500 kbps RX																														
			Nrf_4Mbit_OBT6	9	4 Mbps Nordic proprietary radio mode (BT=0.6/h=0.5)																														
			Nrf_4Mbit_OBT4	10	4 Mbps Nordic proprietary radio mode (BT=0.4/h=0.5)																														
			Ieee802154_250Kbit	15	IEEE 802.15.4-2006 250 kbps																														

8.17.14.79 STATE

Address offset: 0x520

Current radio state

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	STATE			Current radio state																														
			Disabled	0	RADIO is in the DISABLED state																														
			RxRu	1	RADIO is in the RXRU state																														
			RxIdle	2	RADIO is in the RXIDLE state																														
			Rx	3	RADIO is in the RX state																														
			RxDisable	4	RADIO is in the RXDISABLE state																														
			TxRu	9	RADIO is in the TXRU state																														
			TxIdle	10	RADIO is in the TXIDLE state																														
			Tx	11	RADIO is in the TX state																														
			TxDisable	12	RADIO is in the TXDISABLE state																														

8.17.14.80 EDCTRL

Address offset: 0x530

IEEE 802.15.4 energy detect control

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0							
ID				B								B	B	B	B	B	A								A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x20000000				0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
ID	R/W	Field	Value ID	Value				Description																																		
A	RW	EDCNT						IEEE 802.15.4 energy detect loop count																																		
								Number of iterations to perform an ED scan. If set to 0 one scan is performed, otherwise the specified number + 1 of ED scans will be performed and the max ED value tracked in EDSAMPLE.																																		
B	RW	EDPERIOD						IEEE 802.15.4 energy detect period, 4us resolution, no averaging except the IEEE 802.15.4 ED range 128us (32)																																		
			Default	32				EDPERIOD value other than Default is not supported.																																		

8.17.14.81 EDSAMPLE

Address offset: 0x534

IEEE 802.15.4 energy detect level

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	EDLVL		[0..127]				IEEE 802.15.4 energy detect level																											
								Register value must be converted to IEEE 802.15.4 range by an 8-bit saturating multiplication by factor ED_RSSISCALE, as shown in the code example for ED sampling																											

8.17.14.82 CCACTRL

Address offset: 0x538

IEEE 802.15.4 clear channel assessment control

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D D D D D D D D C C C C C C C C B B B B B B B B A A A																															
Reset 0x052D0000				0 0 0 0 0 1 0 1 0 0 1 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CCAMODE			CCA mode of operation																														
			EdMode	0	Energy above threshold																														
					Will report busy whenever energy is detected above CCAEDTHRES																														
			CarrierMode	1	Carrier seen																														
					Will report busy whenever compliant IEEE 802.15.4 signal is seen																														
			CarrierAndEdMode	2	Energy above threshold AND carrier seen																														
		CarrierOrEdMode	3	Energy above threshold OR carrier seen																															
		EdModeTest1	4	Energy above threshold test mode that will abort when first ED measurement over threshold is seen. No averaging.																															
B	RW	CCAEDTHRES			CCA energy busy threshold. Used in all the CCA modes except CarrierMode. Must be converted from IEEE 802.15.4 range by dividing by factor ED_RSSISCALE - similar to EDSAMPLE register																														
C	RW	CCACORRTHRES			CCA correlator busy threshold. Only relevant to CarrierMode, CarrierAndEdMode, and CarrierOrEdMode.																														
D	RW	CCACORRCNT			Limit for occurrences above CCACORRTHRES. When not equal to zero the corrolator based signal detect is enabled.																														

8.17.14.83 DATAWHITE

Address offset: 0x540

Data whitening configuration

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				B B B B B B B B B B B B																A A A A A A A A A A A A																
Reset 0x00890040				0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																												
A	RW	IV						Whitening initial value																												
								Data whitening initial value.																												
B	RW	POLY						Whitening polynomial																												
								Data whitening polynomial. Bit 0 is always interpreted as 1.																												

8.17.14.84 TIMING

Address offset: 0x704

Timing

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000001				0 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	RU						Ramp-up time																											
			Legacy	0				Legacy ramp-up time																											
			Fast	1				Fast ramp-up (default)																											

8.17.14.85 FREQUENCY

Address offset: 0x708

Frequency

Bit number		31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																								
ID																																			B	A	A	A	A	A	A	
Reset 0x00000002		0 1 0																																								
ID	R/W	Field	Value ID	Value	Description																																					
A	RW	FREQUENCY			Radio channel frequency. Frequency = 2400 + FREQUENCY (MHz).																																					
B	RW	MAP			Channel map selection. 0: Channel map between 2400 MHz to 2500 MHz, Frequency = 2400 + FREQUENCY (MHz). 1: Channel map between 2360 MHz to 2460 MHz, Frequency = 2360 + FREQUENCY (MHz).																																					

8.17.14.86 TXPOWER

Address offset: 0x710

Output power

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			A A																															

8.17.14.87 TIFS

Address offset: 0x714

Interframe spacing in μ s

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	TIFS						Interframe spacing in us. Interframe space is the time interval between two consecutive packets. It is defined as the time, in microseconds, from the end of the last bit of the previous packet to the start of the first bit of the subsequent packet.																											

8.17.14.88 RSSISAMPLE

Address offset: 0x718

RSSI sample

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A																															
Reset 0x0000007F				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	R	RSSISAMPLE																		RSSI sample result. The value of this register is read as a positive value while the actual received signal strength is a negative value. Actual received signal strength is therefore as follows: received signal strength = -A dBm.															

8.17.14.89 FECONFIG

Address offset: 0x908

Config register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x10800005				0 0 0 1 0 0 0 0 1 0 1 0 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	SCALERMODE						Mode for narrow scaling output.																											
			Disabled	0				Classic log based scaling mode.																											
			Enabled	1				LUT based scaling mode.																											

8.17.14.90 DFEMODE

Address offset: 0xD00

Whether to use Angle-of-Arrival (AOA) or Angle-of-Departure (AOD)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	DFEOPMODE			Direction finding operation mode																														
			Disabled	0	Direction finding mode disabled																														
			AoD	2	Direction finding mode set to AoD																														
			AoA	3	Direction finding mode set to AoA																														

8.17.14.91 DFESTATUS

Address offset: 0xD04

DfE status information

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																																A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	R	SWITCHINGSTATE				Internal state of switching state machine																																
			Idle	0	Switching state Idle																																	
			Offset	1	Switching state Offset																																	
			Guard	2	Switching state Guard																																	
			Ref	3	Switching state Ref																																	
			Switching	4	Switching state Switching																																	
			Ending	5	Switching state Ending																																	
B	R	SAMPLINGSTATE				Internal state of sampling state machine																																
			Idle	0	Sampling state Idle																																	
			Sampling	1	Sampling state Sampling																																	

8.17.14.92 DFCTRL1

Address offset: 0xD10

Various configuration for Direction finding

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				H H H H G G G G F F F E D D D C C C B A A A A A A																															
Reset 0x00023282				0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	1	0	1	0	0	0	0	0	1	0
ID	R/W	Field	Value ID	Value	Description																														
A	RW	NUMBEROF8US			Length of the AoA/AoD procedure in number of 8 us units																														
					Always used in TX mode, but in RX mode only when CTEINLINECTRLLEN is 0																														
B	RW	DFEINEXTENSION			Add CTE extension and do antenna switching/sampling in this extension																														
			CRC	1	AoA/AoD procedure triggered at end of CRC																														
			Payload	0	Antenna switching/sampling is done in the packet payload																														
C	RW	TSWITCHSPACING			Interval between every time the antenna is changed in the SWITCHING state																														
			4us	1	4us																														
			2us	2	2us																														
			1us	3	1us																														
D	RW	TSAMPLESPACINGREF			Interval between samples in the REFERENCE period																														
			4us	1	4us																														
			2us	2	2us																														
			1us	3	1us																														
			500ns	4	0.5us																														

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				H H H H G G G G F F F E D D D C C C B A A A A A A																															
Reset 0x00023282				0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 0 0 1 0 1 0 0 0 0 0 1 0																															
ID	R/W	Field	Value ID	Value	Description																														
			250ns	5	0.25us																														
			125ns	6	0.125us																														
			Whether to sample I/Q or magnitude/phase																																
E	RW	SAMPLETYPE	IQ	0	Complex samples in I and Q																														
			MagPhase	1	Complex samples as magnitude and phase																														
F	RW	TSAMPLESPACING	Interval between samples in the SWITCHING period when CTEINLINECTRLLEN is 0																																
Note: Not used when CTEINLINECTRLLEN is set. Then either CTEINLINERXMODE1US or CTEINLINERXMODE2US are used.																																			
			4us	1	4us																														
			2us	2	2us																														
			1us	3	1us																														
			500ns	4	0.5us																														
			250ns	5	0.25us																														
			125ns	6	0.125us																														
G	RW	REPEATPATTERN	Repeat every antenna pattern N times.																																
			NoRepeat	0	Do not repeat (1 time in total)																														
H	RW	AGCBACKOFFGAIN	Gain will be lowered by the specified number of gain steps at the start of CTE																																
Note: First LNAGAIN gain drops, then MIXGAIN, then AAFGAIN																																			

8.17.14.93 DFCTRL2

Address offset: 0xD14

Start offset for Direction finding

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B B B B B B B B B B B B B																A A A A A A A A A A A A A A A A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	TSWITCHOFFSET				Signed value offset after the end of the CRC before starting switching in number of 16M cycles																													
B	RW	TSAMPLEOFFSET				Signed value offset before starting sampling in number of 16M cycles relative to the beginning of the REFERENCE state - 12 us after switching start																													

8.17.14.94 SWITCHPATTERN

Address offset: 0xD28

GPIO patterns to be used for each antenna

Maximum 8 GPIOs can be controlled. To secure correct signal levels on the pins, the pins must be configured in the GPIO peripheral as described in Pin configuration.

If the total number of antenna slots is bigger than the number of patterns, we loop back to the pattern used after the reference pattern.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A				
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
ID	R/W	Field	Value ID	Value				Description																															
A	RW	PTR						Data pointer																															
																								See the memory chapter for details about which memories are available for EasyDMA.															

8.17.14.97.2 DFEPACKET.MAXCNT

Address offset: 0xD54

Maximum number of bytes to transfer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00004000				0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	RW	MAXCNT																		Maximum number of bytes to transfer															

8.17.14.97.3 DFEPACKET.AMOUNT

Address offset: 0xD58

Number of bytes transferred in the last transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	AMOUNT						Number of bytes transferred in the last transaction																											

8.17.14.97.4 DFEPACKET.CURRENTAMOUNT

Address offset: 0xD5C

Number of bytes transferred in the current transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	AMOUNT						Number of bytes transferred in the current transaction. Continuously updated.																											

8.17.14.98 CRCSTATUS

Address offset: 0xE0C

CRC status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	CRCSTATUS			CRC status of packet received																														
			CRCError	0	Packet received with CRC error																														
			CRCOk	1	Packet received with CRC ok																														

8.17.14.99 RXMATCH

Address offset: 0xE10

Received address

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	RXMATCH						Received address																											
								Logical address of which previous packet was received																											

8.17.14.100 RXCRC

Address offset: 0xE14

CRC field of previously received packet

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	RXCRC						CRC field of previously received packet																											
								CRC field of previously received packet																											

8.17.14.101 DAI

Address offset: 0xE18

Device address match index

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	DAI			Device address match index																														
					Index (n) of device address, see DAB[n] and DAP[n], that got an address match																														

8.17.14.102 PDUSTAT

Address offset: 0xE1C

Payload status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	PDUSTAT			Status on payload length vs. PCNF1.MAXLEN																														
			LessThan	0	Payload less than PCNF1.MAXLEN																														
			GreaterThan	1	Payload greater than PCNF1.MAXLEN																														
B	R	CISTAT			Status on what rate packet is received with in Long Range																														
			LR125kbit	0	Frame is received at 125 kbps																														
			LR500kbit	1	Frame is received at 500 kbps																														

8.17.14.103 PCNF0

Address offset: 0xE20

Packet configuration register 0

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				H H		G F F E E D D C C C C																B								A A A A					
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	LFLEN			Length on air of LENGTH field in number of bits.																														
B	RW	SOLEN			Length on air of S0 field in number of bytes.																														
C	RW	S1LEN			Length on air of S1 field in number of bits.																														
D	RW	S1INCL			Include or exclude S1 field in RAM																														
			Automatic	0	Include S1 field in RAM only if S1LEN > 0																														
			Include	1	Always include S1 field in RAM independent of S1LEN																														
E	RW	CILEN			Length of code indicator - long range																														
F	RW	PLEN			Length of preamble on air. Decision point: TASKS_START task																														
			8bit	0	8-bit preamble																														
			16bit	1	16-bit preamble																														
			32bitZero	2	32-bit zero preamble - used for IEEE 802.15.4																														
			LongRange	3	Preamble - used for BLE long range																														
G	RW	CRCINC			Indicates if LENGTH field contains CRC or not																														
			Exclude	0	LENGTH does not contain CRC																														
			Include	1	LENGTH includes CRC																														
H	RW	TERMLEN			Length of TERM field in Long Range operation																														

8.17.14.104 PCNF1

Address offset: 0xE28

Packet configuration register 1

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				F E D																C C C B B B B B B B A A A A A A A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	MAXLEN		[0..255]				Maximum length of packet payload. If the packet payload is larger than MAXLEN, the radio will truncate the payload to MAXLEN.																											

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				F E D																C C C B B B B B B B A A A A A A A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
B	RW	STATLEN		[0..255]	Static length in number of bytes																														
					The static length parameter is added to the total length of the payload when sending and receiving packets, e.g. if the static length is set to N the radio will receive or send N bytes more than what is defined in the LENGTH field of the packet.																														
C	RW	BALEN		[2..4]	Base address length in number of bytes																														
					The address field is composed of the base address and the one byte long address prefix, e.g. set BALEN=2 to get a total address of 3 bytes.																														
D	RW	ENDIAN			On-air endianness of packet, this applies to the S0, LENGTH, S1, and the PAYLOAD fields.																														
			Little	0	Least significant bit on air first																														
			Big	1	Most significant bit on air first																														
E	RW	WHITEEN			Enable or disable packet whitening																														
					Including the address field to CRC check is not supported for whitened packets.																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
F	RW	WHITEOFFSET			If whitening is enabled S0 can be configured to be excluded from whitening																														
			Include	0	S0 included in whitening																														
			Exclude	1	S0 excluded from whitening																														

8.17.14.105 BASE0

Address offset: 0xE2C

Base address 0

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW	BASE0										Base address 0																							

8.17.14.106 BASE1

Address offset: 0xE30

Base address 1

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	RW	BASE1		Base address 1																															

8.17.14.107 PREFIX0

Address offset: 0xE34

Prefixes bytes for logical addresses 0-3

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D	D	D	D	D	D	D	D	C	C	C	C	C	C	C	B	B	B	B	B	B	B	B	A	A	A	A	A	A	A	A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A-D	RW	AP[i] (i=0..3)						Address prefix i																											

8.17.14.108 PREFIX1

Address offset: 0xE38

Prefixes bytes for logical addresses 4-7

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D	D	D	D	D	D	D	D	C	C	C	C	C	C	C	C	B	B	B	B	B	B	B	B	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A-D	RW	AP[i] (i=4..7)						Address prefix i																											

8.17.14.109 TXADDRESS

Address offset: 0xE3C

Transmit address select

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	TXADDRESS						Transmit address select																											
				Logical address to be used when transmitting a packet																															

8.17.14.110 RXADDRESSES

Address offset: 0xE40

Receive address select

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																												H	G	F	E	D	C	B	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																													
A-H	RW	ADDR[i] (i=0..7)				Enable or disable reception on logical address i																													
			Disabled	0		Disable																													
			Enabled	1		Enable																													

8.17.14.111 CRCCNF

Address offset: 0xE44

CRC configuration

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	LEN			CRC length in number of bytes.																														
					Note: For MODE Ble_LR125Kbit and Ble_LR500Kbit, only LEN set to 3 is supported																														
			Disabled	0	CRC length is zero and CRC calculation is disabled																														
			One	1	CRC length is one byte and CRC calculation is enabled																														
			Two	2	CRC length is two bytes and CRC calculation is enabled																														
			Three	3	CRC length is three bytes and CRC calculation is enabled																														
B	RW	SKIPADDR			Control whether CRC calculation skips the address field. Other fields can also be skipped.																														
			Include	0	CRC calculation includes address field																														
			Skip	1	CRC calculation starting at first byte after address field.																														
			ieee802154	2	CRC calculation starting at first byte after length field (as per 802.15.4 standard).																														
			SkipS0	3	CRC calculation starting at first byte after S0 field.																														
			SkipS1	4	CRC calculation starting at first byte after S1 field.																														

8.17.14.112 CRCPOLY

Address offset: 0xE48

CRC polynomial

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

8.17.14.113 CRCINIT

Address offset: 0xE4C

CRC initial value

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CRCINIT						CRC initial value																											
								Initial value for CRC calculation																											

8.17.14.114 DAB[n] (n=0..7)

Address offset: 0xE50 + (n × 0x4)

Device address base segment n

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	DAB						Device address base segment n																											

8.17.14.115 DAP[n] (n=0..7)

Address offset: 0xE70 + (n × 0x4)

Device address prefix n

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	DAP						Device address prefix n																											

8.17.14.116 DACNF

Address offset: 0xE90

Device address match configuration

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				P O N M L K J I H G F E D C B A																																		
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A-H	RW	ENA[i] (i=0..7)				Enable or disable device address matching using device address i																																
			Disabled	0		Disabled																																
			Enabled	1		Enabled																																
I-P	RW	TXADD[i] (i=0..7)				TxAdd for device address i																																

8.17.14.117 BCC

Address offset: 0xE94

Bit counter compare

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	BCC						Bit counter compare																											
								Bit counter compare register																											

8.17.14.118 CTESTATUS

Address offset: 0xEA4

CTEInfo parsed from received packet

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C C B A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	CTETIME						CTETime parsed from packet																											
B	R	RFU						RFU parsed from packet																											
C	R	CTETYPE						CTEType parsed from packet																											

8.17.14.119 MHRMATCHCONF

Address offset: 0xEB4

Search pattern configuration

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A		
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	MHRMATCHCONF			Search pattern configuration																																	

8.17.14.120 MHRMATCHMASK

Address offset: 0xEB8

Pattern mask

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A		
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	MHRMATCHMASK			Pattern mask																																	

8.17.14.121 SFD

Address offset: 0xEBC

IEEE 802.15.4 start of frame delimiter

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A A A A A A A																															
Reset 0x000000A7					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	SFD			IEEE 802.15.4 start of frame delimiter. Note: the least significant 4 bits of the SFD cannot all be zeros.																															

8.17.14.122 CTEINLINECONF

Address offset: 0xEC0

Configuration for CTE inline mode

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			H H H H H H H H G G G G G G G F F F E E E D D C B A																															
Reset 0x00002800			0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	CTEINLINECTRLLEN			Enable parsing of CTEInfo from received packet in BLE modes																													
			Enabled	1	Parsing of CTEInfo is enabled																													
			Disabled	0	Parsing of CTEInfo is disabled																													
B	RW	CTEINFOINS1			CTEInfo is S1 byte or not																													
			InS1	1	CTEInfo is in S1 byte (data PDU)																													
			NotInS1	0	CTEInfo is NOT in S1 byte (advertising PDU)																													
C	RW	CTEERRORHANDLING			Sampling/switching if CRC is not OK																													
			Yes	1	Sampling and antenna switching also when CRC is not OK																													
			No	0	No sampling and antenna switching when CRC is not OK																													
D	RW	CTETIMEVALIDRANGE			Max range of CTETime																													
					Note: Valid range is 2-20 in BLE core spec. If larger than 20, it can be an indication of an error in the received packet.																													
			20	0	20 in 8us unit (default)																													
					Set to 20 if parsed CTETime is larger han 20																													
			31	1	31 in 8us unit																													
		63	2	63 in 8us unit																														
E	RW	CTEINLINERXMODE1US			Spacing between samples for the samples in the SWITCHING period when CTEINLINEMODE is set																													
					When the device is in AoD mode, this is used when the received CTEType is "AoD 1 us". When in AoA mode, this is used when TSWITCHSPACING is 2 us.																													
			4us	1	4us																													
			2us	2	2us																													
			1us	3	1us																													
			500ns	4	0.5us																													
			250ns	5	0.25us																													
			125ns	6	0.125us																													
F	RW	CTEINLINERXMODE2US			Spacing between samples for the samples in the SWITCHING period when CTEINLINEMODE is set																													
					When the device is in AoD mode, this is used when the received CTEType is "AoD 2 us". When in AoA mode, this is used when TSWITCHSPACING is 4 us.																													
			4us	1	4us																													
			2us	2	2us																													
			1us	3	1us																													
			500ns	4	0.5us																													
			250ns	5	0.25us																													
			125ns	6	0.125us																													
G	RW	S0CONF			S0 bit pattern to match																													
					The least significant bit always corresponds to the first bit of S0 received																													
H	RW	S0MASK			S0 bit mask to set which bit to match																													
					The least significant bit always corresponds to the first bit of S0 received																													

8.17.14.123 PACKETPTR

Address offset: 0xED0

Packet pointer

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						Data pointer																											
																				See the memory chapter for details about which memories are available for EasyDMA.															

8.17.14.124 CSTONES.MODE

Address offset: 0x1000

Selects the mode(s) that are activated on the start signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000003				0 1 1																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	TPM				Enable or disable TPM																													
			Disabled	0	TPM is disabled																														
			Enabled	1	TPM is enabled																														
B	RW	TFM				Enable or disable TFM																													
			Disabled	0	TFM is disabled																														
			Enabled	1	TFM is enabled																														

8.17.14.125 CSTONES.NUMSAMPLES

Address offset: 0x1004

Number of input samples at 2MHz sample rate

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																												A	A	A	A	A	A	A	A	A	A
Reset 0x000000A0				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	
ID	R/W	Field	Value ID	Value				Description																													
A	RW	NUMSAMPLES						Maximum value supported is 160																													

8.17.14.126 CSTONES.NEXTFREQUENCY

Address offset: 0x1008

The value of FREQUENCY that will be used in the next step

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	RW	NEXTFREQUENCY		Frequency = 2400 + FREQUENCY (MHz)																															

8.17.14.127 CSTONES.FFOIN

Address offset: 0x100C

Override value of FFO (Fractional Frequency Offset) if not to be based on the frequency estimate derived from CnAcc (autocorrelation of the scaled input signal) value

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0									
ID																											A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0								
ID	R/W	Field		Value ID	Value				Description																																			
A	RW	FFFIN							Units 62.5 ppb. Max range +/-100 ppm plus margin.																																			

8.17.14.128 CSTONES.FFOSOURCE

Address offset: 0x1010

Source of FFO

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000001				0 1																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	FFOSOURCE			Use external or internal FFOSOURCE																														
			External	0	Use FFOIN																														
			Internal	1	Calc FFO from CnAcc																														

8.17.14.129 CSTONES.FAEPEER

Address offset: 0x1014

FAEPEER (Frequency Actuation Error) of peer if known. Used during Mode 0 steps.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW	FAEPEER										Units 31.25 ppb.																							

8.17.14.130 CSTONES.PHASESHIFT

Address offset: 0x1018

Parameter used in TPM, provided by software

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value																Description																	
A	RW	PHASESHIFT																		Phase shift used in TPM calculation																	

8.17.14.131 CSTONES.NUMSAMPLESCOEFF

Address offset: 0x101C

Parameter used in TPM, provided by software

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x0000199A				0 1 1 0 0 1 1 0 0 1 1 0 1 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW	NUMSAMPLESCOEFF										Coefficient 2**16/(numSamples/16) in Q1.15 format (Default numsamples value is 160)																							

8.17.14.132 CSTONES.PCT16

Address offset: 0x1020

Mean magnitude and mean phase converted to IQ

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	PCT16I						Inphase																											
B	R	PCT16Q						Quadrature																											

8.17.14.133 CSTONES.MAGPHASEMEAN

Address offset: 0x1024

Mean magnitude and phase of the signal before it is converted to PCT16

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B B B B B B B B B B B B A																															

8.17.14.134 CSTONES.IQRAWMEAN

Address offset: 0x1028

Mean of IQ values

Bit number	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																														
ID	B B B B B B B B B B B B B B B A A A A A A A A A A A A A A A A																														
Reset 0x00000000	0 0																														
ID	R/W	Field	Value ID	Value	Description																										
A	R	IQRAWMEANI			Inphase																										
B	R	IQRAWMEANQ			Quadrature																										

8.17.14.135 CSTONES.MAGSTD

Address offset: 0x102C

Magnitude standard deviation approximation

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	MAGSTD						Magnitude standard deviation approximation																											

8.17.14.136 CSTONES.CNACC

Address offset: 0x1030

Output of the autocorrelation of the accumulated IQ signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B B B B B B B B B B B B B A A A A A A A A A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	CNACCI																																	
B	R	CNACCQ																																	

8.17.14.137 CSTONES.FFOEST

Address offset: 0x1034

FFO estimate

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	FFOEST						Units 62.5 ppb. Max range +/-100 ppm plus margin.																											

8.17.14.138 CSTONES.DOWNSAMPLE

Address offset: 0x1038

Turn on/off down sample of input IQ-signals

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLEFILTER			Turn on/off down sample of input IQ-signals																														
			OFF	0	Disable filter																														
			ON	1	Enable filter																														
B	RW	RATE			Indicating if BLE1M or BLE2M is used																														
			BLE1M	0	Radio mode BLE1M is used																														
			BLE2m	1	Radio mode BLE2M is used																														

8.17.14.139 CSTONES.FINETUNENEXT

Address offset: 0x103C

Number of full ADPLL finetune steps

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																					A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																											
A	R	FINETUNENEXT			Units of 488.28125 Hz																											

8.17.14.140 CSTONES.CFOPHASE

Address offset: 0x1040

Cordic output of CnAcc

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																					A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																											
A	R	CFOPHASE																														

8.17.14.141 CSTONES.FREQOFFSET

Address offset: 0x1044

Frequency offset estimate

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																					A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																											
A	R	FREQOFFSET																														

8.17.14.142 CSTONES.PCT11

Address offset: 0x1048

Mean magnitude and mean phase converted to IQ. IQ values limited to [-1024,1023].

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B B B B B B B B B B A A A A A A A A A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	PCT11I						Inphase																											
B	R	PCT11Q						Quadrature																											

8.17.14.143 CSTONES.LFAENEXT

Address offset: 0x104C

Quantization error between ADPLL frequency and the desired value of FFO * RF Frequency. Values limited to [-64,63] with units 7.6294 Hz.

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																
Reset 0x00000000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																											
A	R	LFAENEXT			Inphase																											

8.17.14.144 RTT.CONFIG

Address offset: 0x1050

RTT Config.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																				E E E E E E E E D D D D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EN			Enable RTT Functionality. Only valid for BLE 1MBPS and 2MBPS mode																														
			Disabled	0	Disable RTT Block																														
			Enabled	1	Enable RTT Block																														
B	RW	ENFULLAA			Enabling/Disable ping over the entire access address.																														
			Disabled	0	Disable ping over the entire access address, i.e., enable only over the first 16-bit access address																														
			Enabled	1	Enable ping over the entire access address																														
C	RW	ROLE			Role as a Initiator or Reflector.																														
			Initiator	0	Initiator																														
			Reflector	1	Reflector																														
D	RW	NUMSEGMENTS			Number of 16bit payload segments available for ToA detection. Allowed values are 0, 2, 4, 6 and 8.																														
E	RW	EFSDELAY			Early Frame Sync Delay, i.e., number of cycles to wait for access address to anchor correctly. For 2MBPSBLE mode, the EFSDELAY value is 64 (2us) and for 1MBPSBLE mode, it can be 256 (8us).																														

8.17.14.145 RTT.SEGMENT01

Address offset: 0x1054

RTT segments 0 and 1

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	RW	DATA		Data Bits 31 - 0																															

8.17.14.146 RTT.SEGMENT23

Address offset: 0x1058

RTT segments 2 and 3

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW	DATA										Data Bits 63 - 32																							

8.17.14.147 RTT.SEGMENT45

Address offset: 0x105C

RTT segments 4 and 5

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	DATA						Data Bits 95 - 64																											

8.17.14.148 RTT.SEGMENT67

Address offset: 0x1060

RTT segments 6 and 7

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																															
A	RW	DATA			Data Bits 127 - 96																															

8.18 SAADC — Successive approximation analog-to-digital converter

The SAADC peripheral is a differential successive approximation register (SAR) analog-to-digital converter.

The main features of SAADC are the following:

- Three operation modes
 - 10-bit mode with a maximum sample rate of 2 Msps
 - 12-bit mode with a sample rate of 250 ksps
 - 14-bit mode with a sample rate of 31.25 ksps
- 8/10/12-bit resolution, 14-bit resolution with oversampling
- Multiple analog inputs
 - GPIO pins with analog function (input range 0 to VDD)
 - VDD (divided down to a valid range)
- Up to eight input channels
 - One channel per single-ended input and two channels per differential input
 - Scan mode can be configured with both single-ended channels and differential channels
 - Each channel can be configured to select any of the above analog inputs
- Sampling triggered by a task from software or a DPPI channel for full flexibility on sample frequency source from low-power 32.768 kHz RTC or more accurate 1/16 MHz timers
- One-shot conversion mode to sample a single channel
- Scan mode to sample a series of channels in sequence with configurable sample delay
- Support for direct sample transfer to RAM using EasyDMA
- Interrupts on single sample and full buffer events
- Samples stored as 16-bit two's complement values for differential and single-ended sampling
- Continuous sampling without the need of an external timer
- Internal resistor string
- On-the-fly limit checking

8.18.1 Shared resources

The ADC can coexist with COMP and other peripherals using one of AIN0–AIN7, provided these are assigned to different pins.

It is not recommended to select the same analog input pin for both modules.

8.18.2 Overview

The ADC supports up to eight external analog input channels. It can be operated in One-shot mode with sampling under software control, or Continuous mode with a programmable sampling rate.

The analog inputs can be configured as eight single-ended inputs, four differential inputs or a combination of these. Each channel can be configured to select:

- GPIO pins with analog input function, see [Pin assignments](#) on page 793, also marked with name **AIN**. Input range is 0 to VDD.
- **VDD** (divided down to a valid range)
- **DVDD**
- **AVDD**

Channels can be sampled individually in one-shot or continuous sampling modes, or, using scan mode, multiple channels can be sampled in sequence. Channels can also be oversampled to improve noise performance.

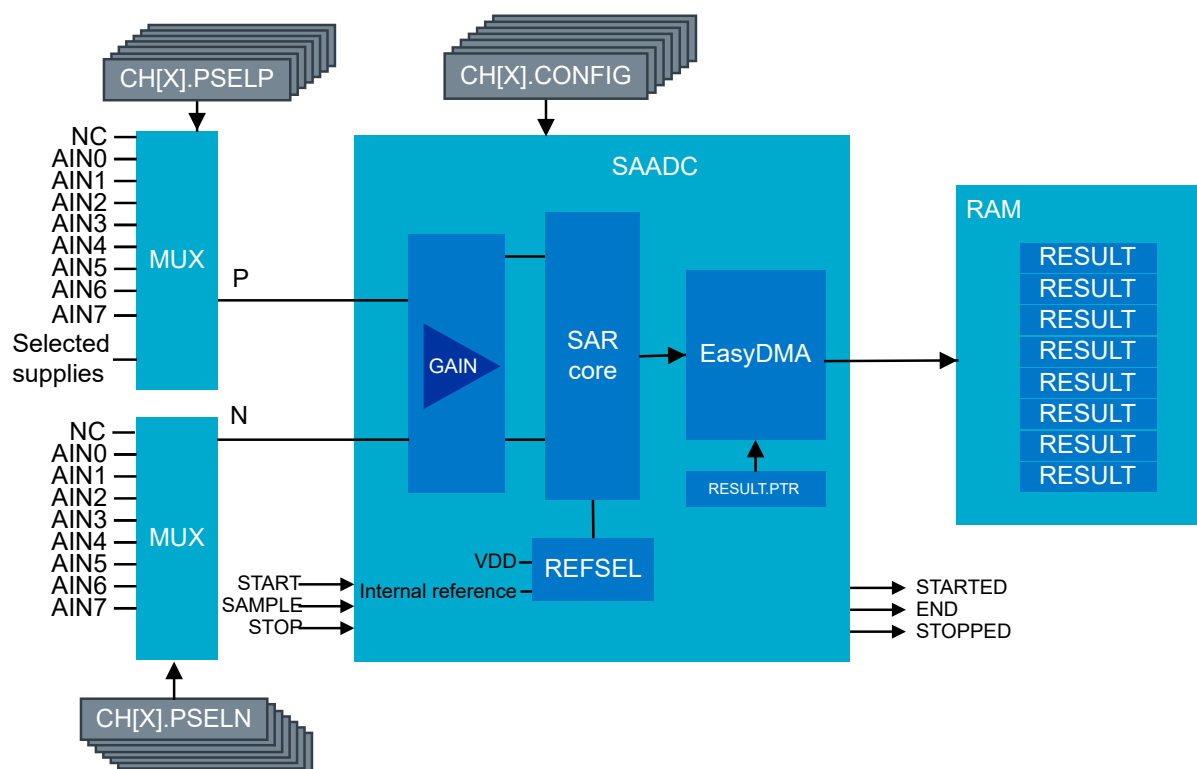


Figure 130: Simplified ADC block diagram

Internally, the ADC is always a differential analog-to-digital converter, but by default it is configured with single-ended input in the MODE field of the CH[n].CONFIG register. In single-ended mode, the negative input will be shorted to ground internally.

The assumption in single-ended mode is that the internal ground of the ADC is the same as the external ground that the measured voltage is referred to. The ADC is thus sensitive to ground bounce on the PCB in single-ended mode. If this is a concern we recommend using differential measurement.

8.18.3 Digital output

The output result of the ADC depends on the settings in the CH[n].CONFIG and RESOLUTION registers as follows:

$$\text{RESULT} = [V(P) - V(N)] * \text{GAIN/REFERENCE} * 2^{(\text{RESOLUTION} - m)}$$

where

V(P)

is the voltage at input P

V(N)

is the voltage at input N

GAIN

is the selected gain setting

m

is the mode setting. Use m=0 if CONFIG.MODE=SE, or m=1 if CONFIG.MODE=Diff

REFERENCE

is the selected reference voltage

The result generated by the ADC will deviate from the expected due to DC errors like offset, gain, differential non-linearity (DNL), and integral non-linearity (INL). See [Electrical specification](#) for details on these parameters. The result can also vary due to AC errors like non-linearities in the GAIN block, settling errors due to high source impedance and sampling jitter. For battery measurement, the DC errors are most noticeable.

The ADC has a wide selection of gains controlled in the GAIN field of the CH[n].CONFIG register. If CH[n].CONFIG.REFSEL=0, the input range of the ADC core is nominally ± 0.6 V differential and the input must be scaled accordingly.

The ADC has a temperature dependent offset. If the ADC is to operate over a large temperature range, we recommend running CALIBRATEOFFSET at regular intervals. The CALIBRATEDONE event will be fired when the calibration has been completed. Note that the DONE and RESULTDONE events will also be generated.

8.18.4 Analog inputs and channels

Up to eight analog input channels, CH[n](n=0..7), can be configured.

See [Shared resources](#) on page 533 for shared input with comparators.

Any one of the available channels can be enabled for the ADC to operate in one-shot mode. If more than one CH[n] is configured, the ADC enters scan mode.

An analog input is selected as a positive converter input if CH[n].PSELP is set, setting CH[n].PSELP also enables the particular channel.

An analog input is selected as a negative converter input if CH[n].PSELN is set. The CH[n].PSELN register will have no effect unless differential mode is enabled and CH[n].PSELP is set, see MODE field in CH[n].CONFIG register.

Important: Channels selected for COMP cannot be used at the same time for ADC sampling, though channels not selected for use by these blocks can be used by the ADC.

8.18.5 Operation modes

The ADC input configuration supports three modes: one-shot (with optional oversampling), continuous and scan .

Note: Scan mode and oversampling should not be combined.

The ADC must be enabled and started via the [ENABLE](#) on page 553, [TASKS_START](#) on page 541 registers, and at least one channel must be enabled (via the CH[n] registers) before sampling the ADC. Otherwise, sampling the ADC (by writing 1 to [TASKS_SAMPLE](#) on page 541) will have no effect.

The ADC indicates a single ongoing conversion via the register [STATUS](#) on page 553. During scan mode, oversampling, or continuous modes, more than a single conversion takes place in the ADC. As consequence, the value reflected in STATUS register will toggle at the end of each single conversion.

8.18.5.1 One-shot mode

One-shot operation is configured by enabling only one of the available channels defined by CH[n].PSELP, CH[n].PSELN, and CH[n].CONFIG registers.

Upon a SAMPLE task, the ADC starts to sample the input voltage. The CH[n].CONFIG.TACQ controls the acquisition time.

A DONE event signals that one sample has been taken.

In this mode, the RESULTDONE event has the same meaning as DONE when no oversampling takes place. Note that both events may occur before the actual value has been transferred into RAM by EasyDMA. For more information, see [EasyDMA](#) on page 536.

8.18.5.2 Continuous mode

Continuous sampling can be achieved by using the internal timer in the ADC, or triggering the SAMPLE task from one of the general purpose timers through the PPI system.

Care shall be taken to ensure that the sample rate fulfils the following criteria, depending on how many channels are active:

$$f_{\text{SAMPLE}} < 1 / (t_{\text{ACQ}} + t_{\text{conv}})$$

The SAMPLERATE register can be used as a local timer instead of triggering individual SAMPLE tasks. When SAMPLERATE.MODE is set to Timers, it is sufficient to trigger SAMPLE task only once in order to start the SAADC and triggering the STOP task will stop sampling. The SAMPLERATE.CC field controls the sample rate.

The SAMPLERATE timer should not be combined with SCAN mode; only one channel should be enabled when using the internal timer.

A DONE event signals that one sample has been taken.

In this mode, the RESULTDONE event has the same meaning as DONE when no oversampling takes place. Note that both events may occur before the actual value has been transferred into RAM by EasyDMA.

8.18.5.3 Oversampling

Depending on the SAADC mode, different oversampling methods are used. In general, oversampling improves the signal-to-noise ratio (SNR). Oversampling, however, does not improve the integral non-linearity (INL), or differential non-linearity (DNL).

Oversampling and scan should not be combined, since oversampling and scan will average over input channels.

Oversampling has two modes:

- **Noise shaping with decimation and filtering:** Noise shaping is implemented by using the SAR-ADC in a first-order delta-sigma loop and the filtering is done with the use of FIR filters. The sampling rate in these modes is 1MS/s, and only 12 and 14 bit resolution are supported. To enable noise shaping use the [NOISESHAPE](#) on page 558
- **Accumulation and averaging:** This is the default mode for oversampling.

The accumulator is controlled in the OVERSAMPLE register. The SAMPLE task must be set $2^{\text{OVERSAMPLE}}$ number of times before the result is written to RAM. This can be achieved by:

- Configuring a fixed sampling rate using the local timer or a general purpose timer and the PPI system to trigger a SAMPLE task
- Triggering SAMPLE $2^{\text{OVERSAMPLE}}$ times from software
- Enabling BURST mode

CH[n].CONFIG.BURST can be enabled to avoid setting SAMPLE task $2^{\text{OVERSAMPLE}}$ times. With BURST = 1 the ADC will sample the input $2^{\text{OVERSAMPLE}}$ times as fast as it can (actual timing: $<(t_{\text{ACQ}}+t_{\text{CONV}}) \times 2^{\text{OVERSAMPLE}})$). Thus, for the user it will just appear like the conversion took a bit longer time, but other than that, it is similar to one-shot mode.

A DONE event signals that one sample has been taken.

In this mode, the RESULTDONE event signals that enough conversions have taken place for an oversampled result to get transferred into RAM. Note that both events may occur before the actual value has been transferred into RAM by EasyDMA.

8.18.5.4 Scan mode

A channel is considered enabled if CH[n].PSEL is set. If more than one channel, CH[n], is enabled, the ADC enters scan mode.

In scan mode, one SAMPLE task will trigger one conversion per enabled channel. The time it takes to sample all channels is:

```
Total time < Sum(CH[x].tACQ+tCONV), x=0..enabled channels
```

A DONE event signals that one channel has been sampled.

In this mode, the RESULTDONE event signals that all enabled channels have been sampled.

8.18.6 EasyDMA

After configuring RESULT.PTR and RESULT.MAXCNT, the ADC resources are started by triggering the START task. The ADC is using EasyDMA to store results in a Result buffer in RAM.

The Result buffer is located at the address specified in the RESULT.PTR register. The RESULT.PTR register is double-buffered and it can be updated and prepared for the next START task immediately after the STARTED event is generated. The size of the Result buffer (in bytes) is specified in the RESULT.MAXCNT register and the ADC will generate an END event when it has filled up the Result buffer, see [ADC](#) on page 537. Results are stored in little-endian byte order in Data RAM. Every sample will be sign extended to 16 bit before stored in the Result buffer.

The ADC is stopped by triggering the STOP task. The STOP task will terminate an ongoing sampling. The ADC will generate a STOPPED event when it has stopped. If the ADC is already stopped when the STOP task is triggered, the STOPPED event will still be generated.

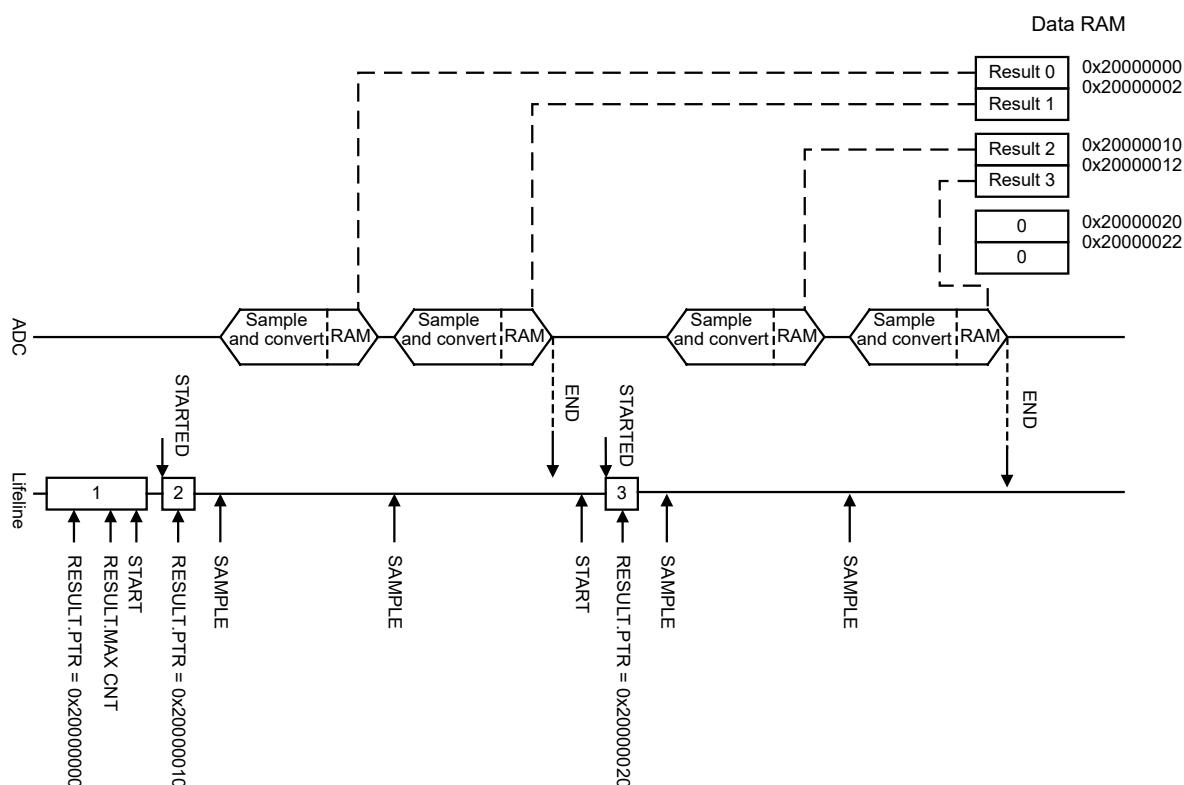


Figure 131: ADC

If the RESULT.PTR is not pointing to a RAM region accessible from the peripheral, an EasyDMA transfer may result in a HardFault and/or memory corruption. See [Memory](#) on page 19 for more information about the different memory regions.

The EasyDMA will have finished accessing the RAM when the END or STOPPED event has been generated.

The RESULT.AMOUNT register can be read following an END event or a STOPPED event to see how many results have been transferred to the Result buffer in RAM since the START task was triggered.

In scan mode, SAMPLE tasks can be triggered once the START task is triggered. The END event is generated when the number of samples transferred to memory reaches the value specified by RESULT.MAXCNT. After an END event, the START task needs to be triggered again before new samples can be taken. For more information about the scan mode, see [Scan mode](#) on page 536.

Note: The user must make sure that the size of the Result buffer is large enough to have space for at least one result from each of the enabled channels, by specifying $\text{RESULT.MAXCNT} \geq 2 * \text{number of channels enabled}$, failing to do so leads to undefined behaviour.

8.18.7 Reference

The ADC can use different reference voltages VREF, controlled in the REFSEL field of the CH[n].CONFIG register.

These are:

- Internal reference, VREF = 0.9 V
- External reference, VREF provided by the EXTREF pin

Note: The external reference voltage should be close to the internal reference voltage. Preferably no more than 5% deviation from the internal reference voltage, VREF. Using a reference voltage >1.2V will lead to increased leakage, and can lead to undefined behaviour.

The SAADC is preceded by a gain stage which has a programmable GAIN. The voltage range seen at the input of the gain stage is:

$$V_{\text{RangeDifferential}} = \pm V_{\text{REF}}/\text{GAIN}$$

$$V_{\text{RangeSingleEnded}} = \pm 0.5 \cdot V_{\text{REF}}/\text{GAIN}$$

The AIN0-AIN7 inputs cannot exceed VDD, or be lower than VSS. The input ranges are also limited by the REFERENCE and GAIN used. The condition

$$[V(P) - V(N)] \cdot \text{GAIN}/\text{REFERENCE} \leq 1$$

must always hold true for valid measurements, otherwise the ADC will saturate and report the max value determined by the RESOLUTION.

8.18.8 Acquisition time

To sample the input voltage, the ADC connects a capacitor to the input.

For illustration, see [Simplified ADC sample network](#) on page 538. The acquisition time indicates how long the capacitor is connected, see TACQ field in CH[n].CONFIG register. The required acquisition time depends on the source (R_{source}) resistance. For high source resistance the acquisition time should be increased, see [Acquisition time](#) on page 538.

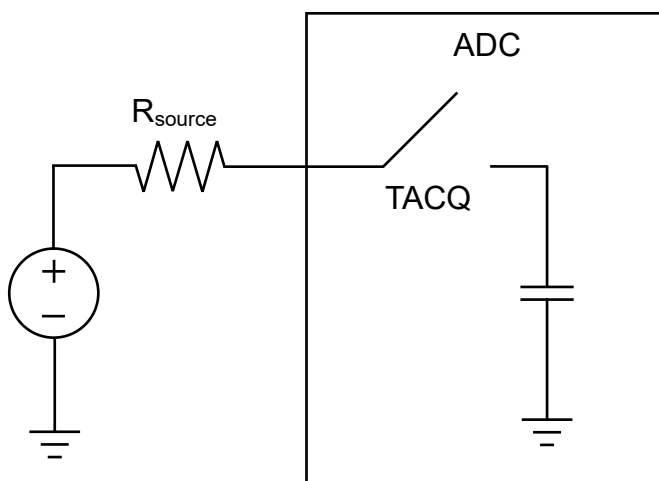


Figure 132: Simplified ADC sample network

TACQ [μs]	Maximum source resistance [kOhm]
3	10
5	40
10	100
15	200
20	400
40	800

Table 52: Acquisition time

8.18.9 Limits event monitoring

A channel can be event monitored by configuring limit register CH[n].LIMIT.

If the conversion result is higher than the defined high limit, or lower than the defined low limit, the appropriate event will get fired.

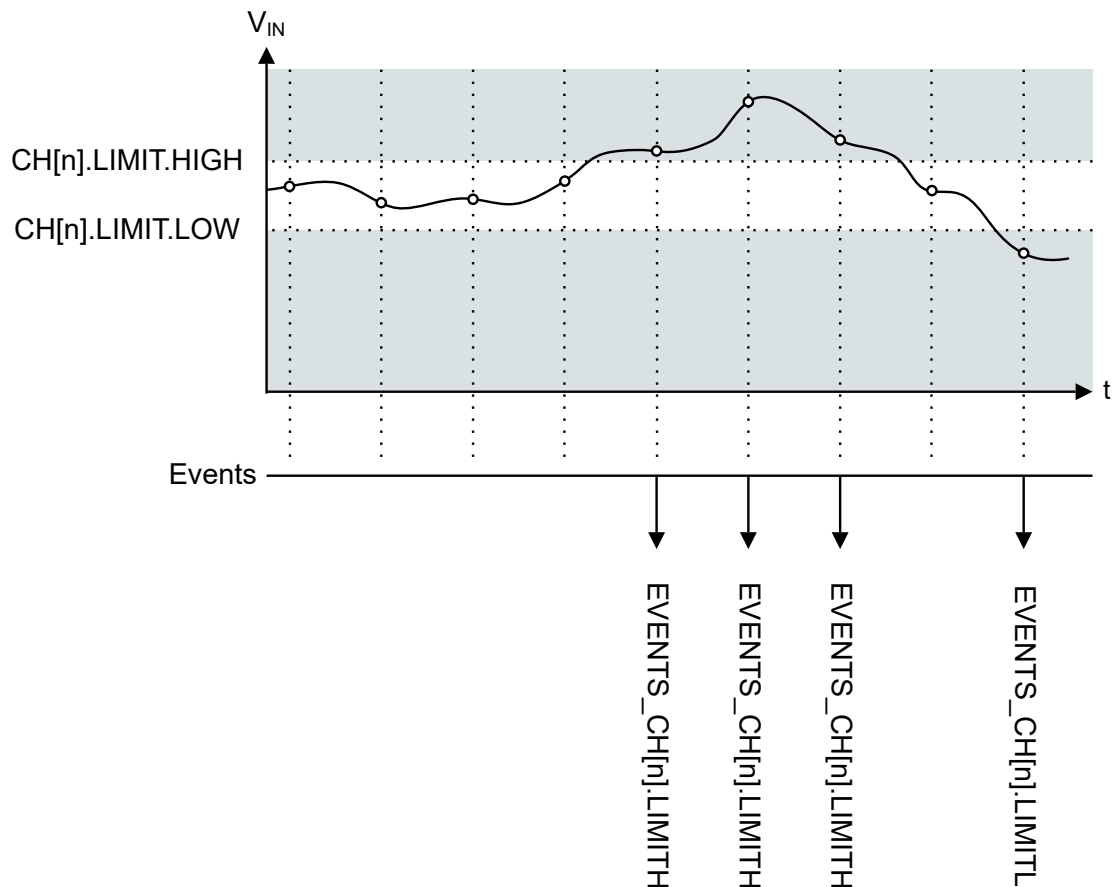


Figure 133: Example of limits monitoring on channel 'n'

Note that when setting the limits, `CH[n].LIMIT.HIGH` shall always be higher than or equal to `CH[n].LIMIT.LOW`. In other words, an event can be fired only when the input signal has been sampled outside of the defined limits. It is not possible to fire an event when the input signal is inside a defined range by swapping high and low limits.

The comparison to limits always takes place, there is no need to enable it. If comparison is not required on a channel, the software shall simply ignore the related events. In that situation, the value of the limits registers is irrelevant, so it does not matter if `CH[n].LIMIT.LOW` is lower than `CH[n].LIMIT.HIGH` or not.

8.18.10 Performance factors

Clock jitter, affecting sample timing accuracy, and circuit noise can affect ADC performance.

Jitter can be between START tasks or from START task to acquisition. START timer accuracy and startup times of regulators and references will contribute to variability. Sources of circuit noise may include CPU activity and the DC/DC regulator. Best ADC performance is achieved using START timing based on the TIMER module, HFXO clock source, and Constant Latency mode.

8.18.11 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
SAADC : S	GLOBAL	0x500D5000	US	S	SA	No	Successive approximation analog-to-digital converter SAADC
SAADC : NS		0x400D5000					

Configuration

Instance	Domain	Configuration
SAADC : S	GLOBAL	CURRENTAMOUNT register not included.
SAADC : NS		

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Start the ADC and prepare the result buffer in RAM
TASKS_SAMPLE	0x004		Take one ADC sample, if scan is enabled all channels are sampled. This task requires that SAADC has started, i.e. EVENTS_STARTED was set and EVENTS_STOPPED was not.
TASKS_STOP	0x008		Stop the ADC and terminate any on-going conversion
TASKS_CALIBRATEOFFSET	0x00C		Starts offset auto-calibration
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_SAMPLE	0x084		Subscribe configuration for task SAMPLE
SUBSCRIBE_STOP	0x088		Subscribe configuration for task STOP
SUBSCRIBE_CALIBRATEOFFSET	0x08C		Subscribe configuration for task CALIBRATEOFFSET
EVENTS_STARTED	0x100		The ADC has started
EVENTS_END	0x104		The ADC has filled up the Result buffer
EVENTS_DONE	0x108		A conversion task has been completed. Depending on the mode, multiple conversions might be needed for a result to be transferred to RAM.
EVENTS_RESULTDONE	0x10C		A result is ready to get transferred to RAM.
EVENTS_CALIBRATEDONE	0x110		Calibration is complete
EVENTS_STOPPED	0x114		The ADC has stopped
EVENTS_CH[n].LIMITH	0x118		Last results is equal or above CH[n].LIMIT.HIGH
EVENTS_CH[n].LIMITL	0x11C		Last results is equal or below CH[n].LIMIT.LOW
PUBLISH_STARTED	0x180		Publish configuration for event STARTED
PUBLISH_END	0x184		Publish configuration for event END
PUBLISH_DONE	0x188		Publish configuration for event DONE
PUBLISH_RESULTDONE	0x18C		Publish configuration for event RESULTDONE
PUBLISH_CALIBRATEDONE	0x190		Publish configuration for event CALIBRATEDONE
PUBLISH_STOPPED	0x194		Publish configuration for event STOPPED
PUBLISH_CH[n].LIMITH	0x198		Publish configuration for event CH[n].LIMITH
PUBLISH_CH[n].LIMITL	0x19C		Publish configuration for event CH[n].LIMITL
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
STATUS	0x400		Status
TRIM.LINCALCOEFF[n]	0x440		Linearity calibration coefficient
ENABLE	0x500		Enable or disable ADC
CH[n].PSEL	0x510		Input positive pin selection for CH[n]

Register	Offset	TZ	Description
CH[n].PSELN	0x514		Input negative pin selection for CH[n]
CH[n].CONFIG	0x518		Input configuration for CH[n]
CH[n].LIMIT	0x51C		High/low limits for event monitoring a channel
RESOLUTION	0x5F0		Resolution configuration
OVERSAMPLE	0x5F4		Oversampling configuration. OVERSAMPLE should not be combined with SCAN. The RESOLUTION is applied before averaging, thus for high OVERSAMPLE a higher RESOLUTION should be used.
SAMPLERATE	0x5F8		Controls normal or continuous sample rate
RESULT.PTR	0x62C		Data pointer
RESULT.MAXCNT	0x630		Maximum number of buffer bytes to transfer
RESULT.AMOUNT	0x634		Number of buffer bytes transferred since last START, updated after the END or STOPPED events
RESULT.CURRENTAMOUNT	0x638		Number of buffer bytes transferred since last START, continuously updated
NOISESHAPE	0x654		Enable noise shaping

8.18.11.1 TASKS_START

Address offset: 0x000

Start the ADC and prepare the result buffer in RAM

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	W	TASKS_START			Start the ADC and prepare the result buffer in RAM																														
			Trigger	1	Trigger task																														

8.18.11.2 TASKS_SAMPLE

Address offset: 0x004

Take one ADC sample, if scan is enabled all channels are sampled. This task requires that SAADC has started, i.e. EVENTS_STARTED was set and EVENTS_STOPPED was not.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	W	TASKS_SAMPLE			Take one ADC sample, if scan is enabled all channels are sampled. This task requires that SAADC has started, i.e. EVENTS_STARTED was set and EVENTS_STOPPED was not.																														
			Trigger	1	Trigger task																														

8.18.11.3 TASKS_STOP

Address offset: 0x008

Stop the ADC and terminate any on-going conversion

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_STOP						Stop the ADC and terminate any on-going conversion																											
			Trigger	1				Trigger task																											

8.18.11.4 TASKS_CALIBRATEOFFSET

Address offset: 0x00C

Starts offset auto-calibration

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_CALIBRATEOFFSET				Starts offset auto-calibration																													
			Trigger	1				Trigger task																											

8.18.11.5 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task [START](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task START will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.18.11.6 SUBSCRIBE_SAMPLE

Address offset: 0x084

Subscribe configuration for task [SAMPLE](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that task SAMPLE will subscribe to																																
B	RW	EN																																				
			Disabled	0	Disable subscription																																	
			Enabled	1	Enable subscription																																	

8.18.11.7 SUBSCRIBE_STOP

Address offset: 0x088

Subscribe configuration for task [STOP](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task STOP will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.18.11.8 SUBSCRIBE_CALIBRATEOFFSET

Address offset: 0x08C

Subscribe configuration for task **CALIBRATEOFFSET**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task CALIBRATEOFFSET will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.18.11.9 EVENTS_STARTED

Address offset: 0x100

The ADC has started

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_STARTED			The ADC has started																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.18.11.10 EVENTS_END

Address offset: 0x104

The ADC has filled up the Result buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_END			The ADC has filled up the Result buffer																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.18.11.11 EVENTS_DONE

Address offset: 0x108

A conversion task has been completed. Depending on the mode, multiple conversions might be needed for a result to be transferred to RAM.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_DONE			A conversion task has been completed. Depending on the mode, multiple conversions might be needed for a result to be transferred to RAM.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.18.11.12 EVENTS_RESULTDONE

Address offset: 0x10C

A result is ready to get transferred to RAM.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_RESULTDONE			A result is ready to get transferred to RAM.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.18.11.13 EVENTS_CALIBRATEDONE

Address offset: 0x110

Calibration is complete

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_CALIBRATEDONE						Calibration is complete																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.18.11.14 EVENTS_STOPPED

Address offset: 0x114

The ADC has stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_STOPPED			The ADC has stopped																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.18.11.15 EVENTS_CH[n] (n=0..7)

Peripheral events.

8.18.11.15.1 EVENTS_CH[n].LIMITH (n=0..7)

Address offset: 0x118 + (n × 0x8)

Last results is equal or above CH[n].LIMIT.HIGH

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	LIMITH			Last results is equal or above CH[n].LIMIT.HIGH																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.18.11.15.2 EVENTS_CH[n].LIMITL (n=0..7)

Address offset: 0x11C + (n × 0x8)

Last results is equal or below CH[n].LIMIT.LOW

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	LIMITL			Last results is equal or below CH[n].LIMIT.LOW																																	
			NotGenerated	0	Event not generated																																	
			Generated	1	Event generated																																	

8.18.11.16 PUBLISH_STARTED

Address offset: 0x180

Publish configuration for event [STARTED](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event STARTED will publish to																													
B	RW	EN																																	
			Disabled	0		Disable publishing																													
			Enabled	1		Enable publishing																													

8.18.11.17 PUBLISH_END

Address offset: 0x184

Publish configuration for event [END](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event END will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.18.11.18 PUBLISH_DONE

Address offset: 0x188

Publish configuration for event **DONE**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A	A																								
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event DONE will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.18.11.19 PUBLISH_RESULTDONE

Address offset: 0x18C

Publish configuration for event **RESULTDONE**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0														
ID				B																								A				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0												
ID	R/W	Field	Value ID	Value				Description																																									
A	RW	CHIDX		[0..255]				DPPI channel that event RESULTDONE will publish to																																									
B	RW	EN																																															
			Disabled	0				Disable publishing																																									
			Enabled	1				Enable publishing																																									

8.18.11.20 PUBLISH_CALIBRATEDONE

Address offset: 0x190

Publish configuration for event **CALIBRATEDONE**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that event CALIBRATEDONE will publish to																																
B	RW	EN																																				
			Disabled	0	Disable publishing																																	
			Enabled	1	Enable publishing																																	

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				V U T S R Q P O N M L K J I H G F E D C B A																																	
Reset 0x00000000				0 0																																	
ID	R/W	Field	Value	ID	Value	Description																															
A	RW	STARTED				Enable or disable interrupt for event STARTED																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
B	RW	END				Enable or disable interrupt for event END																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
C	RW	DONE				Enable or disable interrupt for event DONE																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
D	RW	RESULTDONE				Enable or disable interrupt for event RESULTDONE																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
E	RW	CALIBRATEDONE				Enable or disable interrupt for event CALIBRATEDONE																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
F	RW	STOPPED				Enable or disable interrupt for event STOPPED																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
G	RW	CH0LIMITH				Enable or disable interrupt for event CH0LIMITH																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
H	RW	CH0LIMITL				Enable or disable interrupt for event CH0LIMITL																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
I	RW	CH1LIMITH				Enable or disable interrupt for event CH1LIMITH																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
J	RW	CH1LIMITL				Enable or disable interrupt for event CH1LIMITL																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
K	RW	CH2LIMITH				Enable or disable interrupt for event CH2LIMITH																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
L	RW	CH2LIMITL				Enable or disable interrupt for event CH2LIMITL																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
M	RW	CH3LIMITH				Enable or disable interrupt for event CH3LIMITH																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
N	RW	CH3LIMITL				Enable or disable interrupt for event CH3LIMITL																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
O	RW	CH4LIMITH				Enable or disable interrupt for event CH4LIMITH																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
P	RW	CH4LIMITL				Enable or disable interrupt for event CH4LIMITL																															
			Disabled	0	Disable																																
			Enabled	1	Enable																																
Q	RW	CH5LIMITH				Enable or disable interrupt for event CH5LIMITH																															
			Disabled	0	Disable																																

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
R	RW	CH5LIMITL	Enabled	1	Enable																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
S	RW	CH6LIMITH			Enable or disable interrupt for event CH6LIMITH																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
T	RW	CH6LIMITL			Enable or disable interrupt for event CH6LIMITL																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
U	RW	CH7LIMITH			Enable or disable interrupt for event CH7LIMITH																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
V	RW	CH7LIMITL			Enable or disable interrupt for event CH7LIMITL																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.18.11.24 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	STARTED			Write '1' to enable interrupt for event STARTED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	END			Write '1' to enable interrupt for event END																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	DONE			Write '1' to enable interrupt for event DONE																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	RESULTDONE			Write '1' to enable interrupt for event RESULTDONE																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
E	RW	CALIBRATEDONE			Write '1' to enable interrupt for event CALIBRATEDONE																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
F	RW	STOPPED			Write '1' to enable interrupt for event STOPPED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
			Enabled	1	Read: Enabled																													
G	RW	CH0LIMITH			Write '1' to enable interrupt for event CH0LIMITH																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
H	RW	CH0LIMITL			Write '1' to enable interrupt for event CH0LIMITL																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
I	RW	CH1LIMITH			Write '1' to enable interrupt for event CH1LIMITH																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
J	RW	CH1LIMITL			Write '1' to enable interrupt for event CH1LIMITL																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
K	RW	CH2LIMITH			Write '1' to enable interrupt for event CH2LIMITH																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
L	RW	CH2LIMITL			Write '1' to enable interrupt for event CH2LIMITL																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
M	RW	CH3LIMITH			Write '1' to enable interrupt for event CH3LIMITH																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
N	RW	CH3LIMITL			Write '1' to enable interrupt for event CH3LIMITL																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
O	RW	CH4LIMITH			Write '1' to enable interrupt for event CH4LIMITH																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
P	RW	CH4LIMITL			Write '1' to enable interrupt for event CH4LIMITL																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
Q	RW	CH5LIMITH			Write '1' to enable interrupt for event CH5LIMITH																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
R	RW	CH5LIMITL			Write '1' to enable interrupt for event CH5LIMITL																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
S	RW	CH6LIMITH			Write '1' to enable interrupt for event CH6LIMITH																													

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
T	RW	CH6LIMITL			Write '1' to enable interrupt for event CH6LIMITL																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
U	RW	CH7LIMITH			Write '1' to enable interrupt for event CH7LIMITH																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
V	RW	CH7LIMITL			Write '1' to enable interrupt for event CH7LIMITL																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.18.11.25 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																														
ID			V U T S R Q P O N M L K J I H G F E D C B A																														
Reset 0x00000000			0 0																														
ID	R/W	Field	Value ID	Value	Description																												
A	RW	STARTED			Write '1' to disable interrupt for event STARTED																												
			Clear	1	Disable																												
			Disabled	0	Read: Disabled																												
			Enabled	1	Read: Enabled																												
B	RW	END			Write '1' to disable interrupt for event END																												
			Clear	1	Disable																												
			Disabled	0	Read: Disabled																												
			Enabled	1	Read: Enabled																												
C	RW	DONE			Write '1' to disable interrupt for event DONE																												
			Clear	1	Disable																												
			Disabled	0	Read: Disabled																												
			Enabled	1	Read: Enabled																												
D	RW	RESULTDONE			Write '1' to disable interrupt for event RESULTDONE																												
			Clear	1	Disable																												
			Disabled	0	Read: Disabled																												
			Enabled	1	Read: Enabled																												
E	RW	CALIBRATEDONE			Write '1' to disable interrupt for event CALIBRATEDONE																												
			Clear	1	Disable																												
			Disabled	0	Read: Disabled																												
			Enabled	1	Read: Enabled																												
F	RW	STOPPED			Write '1' to disable interrupt for event STOPPED																												
			Clear	1	Disable																												
			Disabled	0	Read: Disabled																												
			Enabled	1	Read: Enabled																												

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
G	RW	CH0LIMITH			Write '1' to disable interrupt for event CH0LIMITH																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
H	RW	CH0LIMITL			Write '1' to disable interrupt for event CH0LIMITL																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
I	RW	CH1LIMITH			Write '1' to disable interrupt for event CH1LIMITH																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
J	RW	CH1LIMITL			Write '1' to disable interrupt for event CH1LIMITL																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
K	RW	CH2LIMITH			Write '1' to disable interrupt for event CH2LIMITH																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
L	RW	CH2LIMITL			Write '1' to disable interrupt for event CH2LIMITL																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
M	RW	CH3LIMITH			Write '1' to disable interrupt for event CH3LIMITH																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
N	RW	CH3LIMITL			Write '1' to disable interrupt for event CH3LIMITL																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
O	RW	CH4LIMITH			Write '1' to disable interrupt for event CH4LIMITH																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
P	RW	CH4LIMITL			Write '1' to disable interrupt for event CH4LIMITL																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
Q	RW	CH5LIMITH			Write '1' to disable interrupt for event CH5LIMITH																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
R	RW	CH5LIMITL			Write '1' to disable interrupt for event CH5LIMITL																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
S	RW	CH6LIMITH			Write '1' to disable interrupt for event CH6LIMITH																														
			Clear	1	Disable																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				V U T S R Q P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event CH6LIMITL																														
			Disabled	0	Read: Disabled																														
T	RW	CH6LIMITL	Enabled	1	Read: Enabled																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
U	RW	CH7LIMITH	Clear	1	Write '1' to disable interrupt for event CH7LIMITH																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event CH7LIMITH																														
V	RW	CH7LIMITL	Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.18.11.26 STATUS

Address offset: 0x400

Status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	R	STATUS				Status																													
			Ready	0		ADC is ready. No on-going conversion.																													
			Busy	1		ADC is busy. Single conversion in progress.																													

8.18.11.27 TRIM.LINCALCOEFF[n] (n=0..5)

Address offset: 0x440 + (n × 0x4)

Linearity calibration coefficient

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	RW	VAL		0..65535																value															

8.18.11.28 ENABLE

Address offset: 0x500

Enable or disable ADC

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID					A																																
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value	ID	Value	Description																															
A	RW	ENABLE				Enable or disable ADC																															
			Disabled	0	Disable ADC																																
			Enabled	1	Enable ADC																																
			When enabled, the ADC will acquire access to the GPIO pins specified in the CH[n].PSEL and CH[n].PSELN registers.																																		

8.18.11.29 CH[n].PSEL (n=0..7)

Address offset: 0x510 + (n × 0x10)

Input positive pin selection for CH[n]

The analog input is selected based on CH[n].PSEL.PORT and CH[n].PSEL.PIN

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																			
ID				D D																C C B B B B																A A A A			
Reset 0x00000000				0 0																																			
ID	R/W	Field	Value	ID	Value	Description																																	
A	RW	PIN				GPIO pin selection.																																	
B	RW	PORT				GPIO port selection																																	
C	RW	INTERNAL				Internal input selection for analog positive input when CH[n].PSELP.CONNECT = Internal																																	
			Avdd	0	Connected to the internal 0.9V analog supply rail																																		
			Dvdd	1	Connected to the internal 0.9V digital supply rail																																		
			Vdd	2	Connected to VDD																																		
D	RW	CONNECT				Connection																																	
			NC	0	Not connected																																		
			AnalogInput	1	Select analog input																																		
					The analog input is connected based on CH[n].PSELP.PIN and CH[n].PSELP.PORT																																		
			Internal	2	Selects internal inputs.																																		
							The analog input is connected based on CH[n].PSELP.INTERNAL																																

8.18.11.30 CH[n].PSELN (n=0..7)

Address offset: 0x514 + (n × 0x10)

Input negative pin selection for CH[n]

The analog input is selected based on CH[n].PSELN.PORT and CH[n].PSELN.PIN

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C C B B B B A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	PIN			GPIO pin selection.																														
B	RW	PORT			GPIO Port selection																														
C	RW	CONNECT			Connection																														
			NC	0	Not connected																														
			AnalogInput	1	Select analog input																														
					The analog input is connected based on CH[n].PSELN.PIN and CH[n].PSELN.PORT																														

8.18.11.31 CH[n].CONFIG (n=0..7)

Address offset: 0x518 + (n × 0x10)

Input configuration for CH[n]

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				F	F	F				E	E	E	E	E	E	E	E	D				C	B	A	A	A												
Reset 0x00020000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	GAIN			Gain control																																	
			Gain2	0	2																																	
			Gain1	1	1																																	
			Gain2_3	2	2/3																																	
			Gain2_4	3	2/4																																	
			Gain2_5	4	2/5																																	
			Gain2_6	5	2/6																																	
			Gain2_7	6	2/7																																	
		Gain2_8	7	2/8																																		
B	RW	BURST			Enable burst mode																																	
			Disabled	0	Burst mode is disabled (normal operation)																																	
			Enabled	1	Burst mode is enabled. SAADC takes 2^OVERSAMPLE number of samples as fast as it can, and sends the average to Data RAM.																																	
C-	RW	REFSEL			Reference control																																	
			Internal	0	Internal reference (0.9 V)																																	
			External	1	External reference given at PADC_EXT_REF_1V2																																	
D	RW	MODE			Enable differential mode																																	
			SE	0	Single ended, PSELN will be ignored, negative input to ADC shorted to GND																																	
			Diff	1	Differential																																	
E	RW	TACQ		[1..319]	Acquisition time, the time the ADC uses to sample the input voltage. Resulting acquisition time is ((TACQ+1) x 125 ns)																																	
F	RW	TCONV		[1..7]	Conversion time. Resulting conversion time is ((TCONV+1) x 250 ns)																																	

8.18.11.32 CH[n].LIMIT (n=0..7)

Address offset: 0x51C + (n × 0x10)

High/low limits for event monitoring a channel

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x7FFF8000				0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	LOW		[-32768 to +32767]				Low level limit																											
B	RW	HIGH		[-32768 to +32767]				High level limit																											

8.18.11.33 RESOLUTION

Address offset: 0x5F0

Resolution configuration

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																				A	A	A
Reset 0x00000001				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1		
ID	R/W	Field	Value ID	Value				Description																														
A	RW	VAL						Set the resolution																														
			8bit	0				8 bit																														
			10bit	1				10 bit																														
			12bit	2				12 bit																														
			14bit	3				14 bit																														

8.18.11.34 OVERSAMPLE

Address offset: 0x5F4

Oversampling configuration. OVERSAMPLE should not be combined with SCAN. The RESOLUTION is applied before averaging, thus for high OVERSAMPLE a higher RESOLUTION should be used.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	OVERSAMPLE			Oversample control																														
			Bypass	0	Bypass oversampling																														
			Over2x	1	Oversample 2x																														
			Over4x	2	Oversample 4x																														
			Over8x	3	Oversample 8x																														
			Over16x	4	Oversample 16x																														
			Over32x	5	Oversample 32x																														
			Over64x	6	Oversample 64x																														
			Over128x	7	Oversample 128x																														
			Over256x	8	Oversample 256x																														

8.18.11.35 SAMPLERATE

Address offset: 0x5F8

Controls normal or continuous sample rate

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0											
ID																												B	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0										
ID	R/W	Field	Value ID	Value		Description																																								
A	RW	CC		[8..2047]		Capture and compare value. Sample rate is 16 MHz/CC																																								
B	RW	MODE				Select mode for sample rate control																																								
			Task	0	Rate is controlled from SAMPLE task																																									
			Timers	1	Rate is controlled from local timer (use CC to control the rate)																																									

8.18.11.36 RESULT

RESULT EasyDMA channel

8.18.11.36.1 RESULT.PTR

Address offset: 0x62C

Data pointer

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						Data pointer																											

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.18.11.36.2 RESULT.MAXCNT

Address offset: 0x630

Maximum number of buffer bytes to transfer

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0												
ID																												A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0											
ID	R/W	Field	Value ID	Value												Description																															
A	RW	MAXCNT														Maximum number of buffer bytes to transfer																															

8.18.11.36.3 RESULT.AMOUNT

Address offset: 0x634

Number of buffer bytes transferred since last START, updated after the END or STOPPED events

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

8.18.11.36.4 RESULT.CURRENTAMOUNT

Address offset: 0x638

Number of buffer bytes transferred since last START, continuously updated

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																
ID																				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A																														
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																															
ID	R/W	Field	Value ID	Value																																Description																															
A	R	AMOUNT																																		Number of buffer bytes transferred since last START, continuously updated.																															

8.18.11.37 NOISESHAPE

Address offset: 0x654

Enable noise shaping

Note: The first EVENT_RESULTREADY will take longer upon arrival when using noise shaping, as the filters are first filled with valid data

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				A																																A																											
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																																										
A	RW	NOISESHAPE			Enable noise shaping																																																										
			Disable	0	Disable noiseshaping. Oversampling based on accumulate and average.																																																										
			Audio	1	Noiseshaping and decimating. Larger passband. Provides a 50kS/s cut off frequency, 8x the oversampling ratio. See design description for more information																																																										
			Accuracy	2	Noiseshaping and decimating. Smaller passband. Recommended resolution setting is 14 bits. Provides a 5kS/s cut off frequency, 32x the oversampling ratio. See design description for more information																																																										

8.19 SPIM — Serial peripheral interface controller with EasyDMA

The SPI controller peripheral (SPIM) with EasyDMA provides a full duplex, 4-wire synchronous serial communication interface.

The main features of SPIM are the following:

- EasyDMA direct transfer to and from RAM
- SPI mode [0..3]
- Individual selection of I/O pins

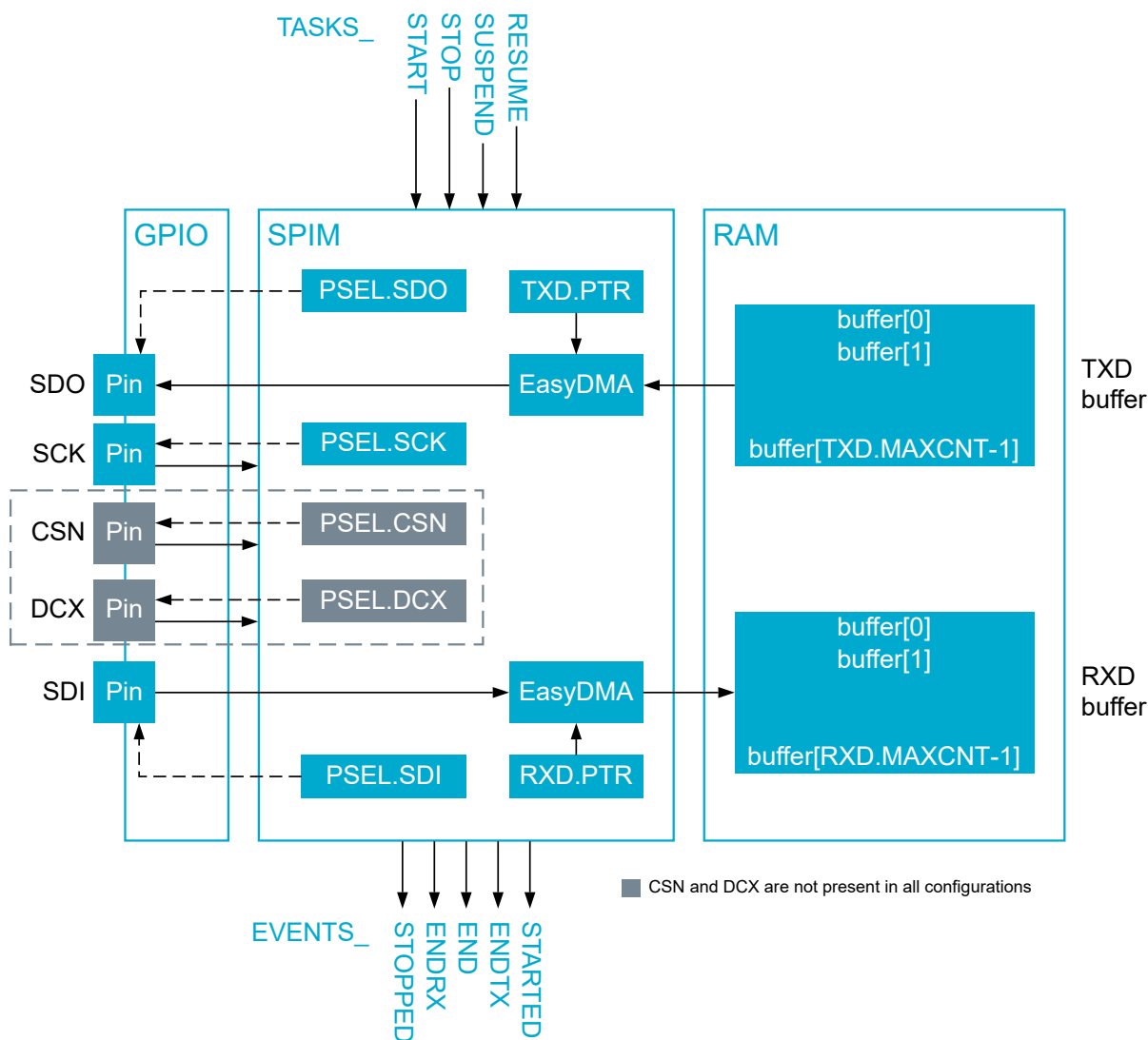


Figure 134: SPIM with EasyDMA

8.19.1 SPIM transaction sequence

An SPIM transaction is started by triggering the START task. This initiates a number of bytes to be transmitted/received on SDO/SDI.

The following figure illustrates an SPIM transaction.

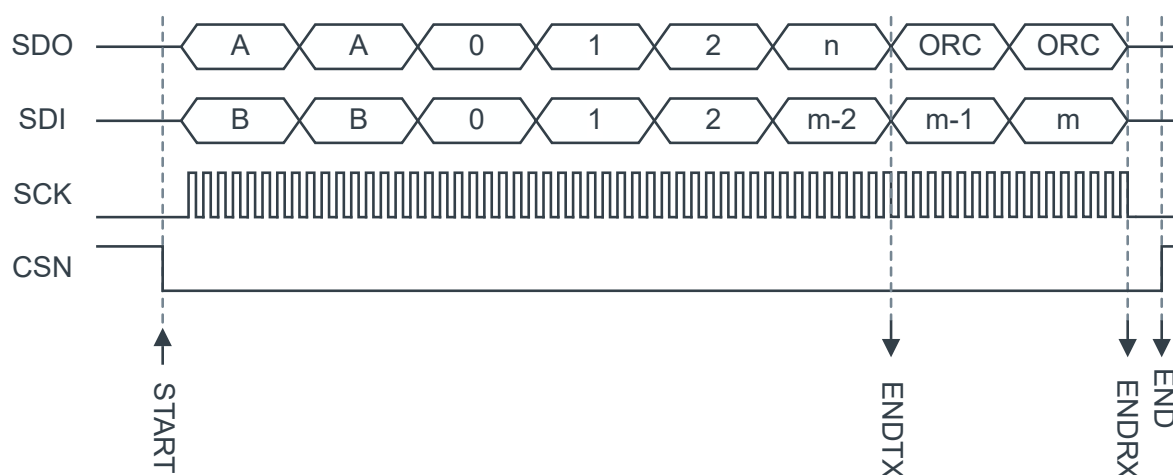


Figure 135: SPIM transaction

The ENDTX event is generated when all bytes in buffer [DMA.TX.PTR](#) on page 583 are transmitted. The number of bytes in the transmit buffer is specified in register [DMA.TX.MAXCNT](#) on page 584. The ENDRX event is generated when buffer [DMA.RX.PTR](#) on page 581 is full; that is when the number of bytes specified in register [DMA.RX.MAXCNT](#) on page 581 have been received. The transaction stops automatically after all bytes are transmitted or received. When the maximum number of bytes in the receive buffer is larger than the number of bytes in the transmit buffer, the contents of register [ORC](#) on page 579 will be transmitted after the last byte in the transmit buffer has been transmitted.

The END event is generated after both the ENDRX and ENDTX events have been generated.

SPIM is stopped by triggering the STOP task. A STOPPED event is generated when the SPIM has stopped. If the STOP task is triggered in the middle of a transaction, SPIM completes the process for the current byte before stopping. The STOPPED event is generated even if the STOP task is triggered while there is no ongoing transaction.

If the ENDTX event has not been generated when the SPIM peripheral stops, the ENDTX event will be generated, even if all bytes in the buffer have not been transmitted.

If the ENDRX event has not been generated when the SPIM stops, the ENDRX event will be generated even if the buffer [DMA.RX.PTR](#) on page 581 is not full.

A transaction can be suspended and resumed using the SUSPEND and RESUME tasks, respectively. When the SUSPEND task is triggered, SPIM completes transmitting and receiving the current byte before it is suspended.

8.19.2 Pin configuration

To configure pins for SPIM use, see the corresponding [PSEL.n](#) registers.

The contents of registers [PSEL.SCK](#), [PSEL.MOSI](#), and [PSEL.MISO](#) are only used when SPIM is enabled, and retained while the device is in System ON mode. The [PSEL.n](#) registers can be configured only when SPIM is disabled in register [ENABLE](#) on page 577.

To ensure correct behavior, the pins used by SPIM must be configured in the GPIO peripheral as described in [GPIO configuration](#) on page 561 before SPIM is enabled.

Only one peripheral can be assigned to drive a GPIO pin at a time. If more than one peripheral is assigned to a GPIO pin, it could result in unpredictable behavior.

SPIM signal	SPIM pin	Direction	Output value
SCK	As specified in PSEL.SCK on page 580	Output	Same as CONFIG.CPOL
SDO	As specified in PSEL.MOSI on page 580	Output	0
SDI	As specified in PSEL.MISO on page 580	Input	Not applicable

Table 53: GPIO configuration

SPIM supports SPI modes [0..3]. The clock polarity (CPOL) and the clock phase (CPHA) are configured in register [CONFIG](#) on page 578.

Mode	Clock polarity (CPOL)	Clock phase (CPHA)
SPI_MODE0	0 (Active High)	0 (Leading)
SPI_MODE1	0 (Active High)	1 (Trailing)
SPI_MODE2	1 (Active Low)	0 (Leading)
SPI_MODE3	1 (Active Low)	1 (Trailing)

Table 54: SPI modes

8.19.3 Shared resources

The SPIM peripheral shares registers and other resources with peripherals that have the same ID as SPIM. Before SPIM can be configured and used, all peripherals that have the same ID as SPIM must be disabled.

Disabling a peripheral with the same ID as SPIM will not reset any shared SPIM registers. Configure all SPIM registers to ensure they operate correctly.

See the Instantiation table in [Instantiation](#) on page 216 for details on peripherals and their IDs.

8.19.4 EasyDMA

SPIM uses EasyDMA to fetch data to transmit from RAM or store received data in RAM.

SPIM implements the following EasyDMA channels.

Channel	Type	Register Cluster
TXD	READER	DMA.TX.PTR on page 583
RXD	WRITER	DMA.RX.PTR on page 581

Table 55: SPIM EasyDMA channels

The .PTR and .MAXCNT registers are double-buffered. After receiving the STARTED event, the registers can be written to before the next transmission.

SPIM automatically stops transmitting after TXD.MAXCNT bytes have been transmitted and RXD.MAXCNT bytes have been received. If RXD.MAXCNT is larger than TXD.MAXCNT, the remaining transmitted bytes will contain the value defined in the ORC register. If TXD.MAXCNT is larger than RXD.MAXCNT, the additional received bytes will be discarded.

The RX.END and TX.END events indicate that EasyDMA has finished accessing buffers in RAM. Both RX and TX must be finished before the END event is generated.

If several AHB bus masters try to access the same AHB slave at the same time, AHB bus congestion can occur. In this case, the EasyDMA channel behavior will depend on the SPIM instance. Refer to [Instances](#) on page 563 for information about what behavior is expected in each instance.

See [EasyDMA](#) for more detailed information.

8.19.5 Low power

When the peripheral is not needed, stop and disable SPIM to ensure lowest possible power consumption.

When the STOP task is sent, the software must wait until the STOPPED event is received before disabling the peripheral through the ENABLE register. If the peripheral is already stopped, the STOP task is not needed to ensure data is not lost.

8.19.6 Timing diagrams

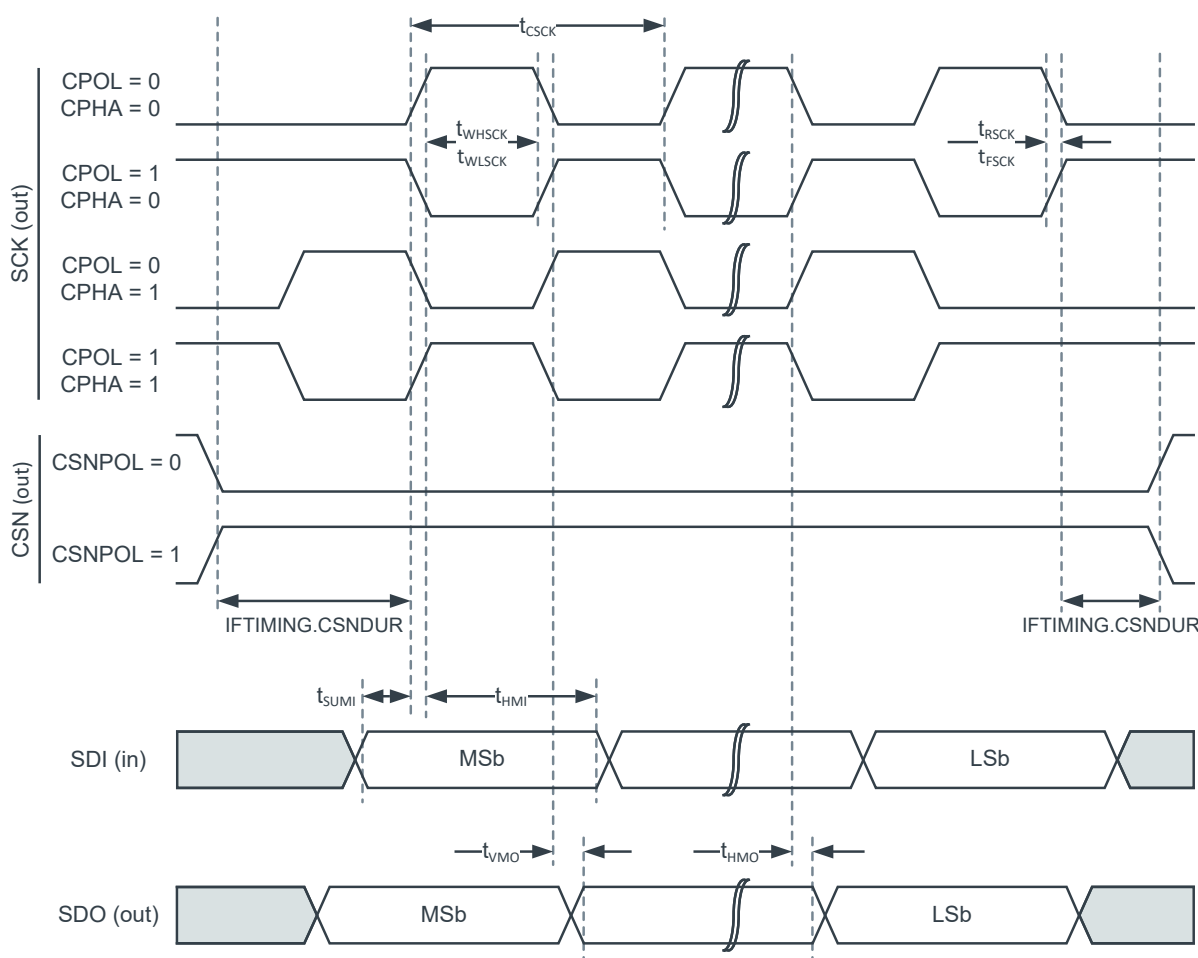


Figure 136: SPIM timing diagram

8.19.7 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
SPIM00 : S	GLOBAL	0x5004A000	US	S	SA	No	SPI controller SPIM00
SPIM00 : NS		0x4004A000					
SPIM20 : S	GLOBAL	0x500C6000	US	S	SA	No	SPI controller SPIM20
SPIM20 : NS		0x400C6000					
SPIM21 : S	GLOBAL	0x500C7000	US	S	SA	No	SPI controller SPIM21
SPIM21 : NS		0x400C7000					
SPIM22 : S	GLOBAL	0x500C8000	US	S	SA	No	SPI controller SPIM22
SPIM22 : NS		0x400C8000					
SPIM30 : S	GLOBAL	0x50104000	US	S	SA	No	SPI controller SPIM30
SPIM30 : NS		0x40104000					

Configuration

Instance	Domain	Configuration
SPIM00 : S	GLOBAL	Use GPIO port P2
SPIM00 : NS		Peripheral core frequency is 128 MHz. Prescaler divisor range is 4..126
SPIM20 : S	GLOBAL	Use GPIO port P1, or dedicated pins on P2
SPIM20 : NS		Peripheral core frequency is 16 MHz. Prescaler divisor range is 2..126
SPIM21 : S	GLOBAL	Use GPIO port P1, or dedicated pins on P2
SPIM21 : NS		Peripheral core frequency is 16 MHz. Prescaler divisor range is 2..126
SPIM22 : S	GLOBAL	Use GPIO port P1, or dedicated pins on P2
SPIM22 : NS		Peripheral core frequency is 16 MHz. Prescaler divisor range is 2..126
SPIM30 : S	GLOBAL	Use GPIO port P0
SPIM30 : NS		Peripheral core frequency is 16 MHz. Prescaler divisor range is 2..126

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Start SPI transaction
TASKS_STOP	0x004		Stop SPI transaction
TASKS_SUSPEND	0x00C		Suspend SPI transaction
TASKS_RESUME	0x010		Resume SPI transaction
TASKS_DMA.RX.ENABLEMATCH[n]	0x030		Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.
TASKS_DMA.RX.DISABLEMATCH[n]	0x040		Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
SUBSCRIBE_SUSPEND	0x08C		Subscribe configuration for task SUSPEND

Register	Offset	TZ	Description
SUBSCRIBE_RESUME	0x090		Subscribe configuration for task RESUME
SUBSCRIBE_DMA.RX.ENABLEMATCH[n]	0x0B0		Subscribe configuration for task ENABLEMATCH[n]
SUBSCRIBE_DMA.RX.DISABLEMATCH[n]	0x0C0		Subscribe configuration for task DISABLEMATCH[n]
EVENTS_STARTED	0x100		SPI transaction has started
EVENTS_STOPPED	0x104		SPI transaction has stopped
EVENTS_END	0x108		End of RXD buffer and TXD buffer reached
EVENTS_DMA.RX.END	0x14C		Generated after all MAXCNT bytes have been transferred
EVENTS_DMA.RX.READY	0x150		Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.
EVENTS_DMA.RX.BUSERROR	0x154		An error occurred during the bus transfer.
EVENTS_DMA.RX.MATCH[n]	0x158		Pattern match is detected on the DMA data bus.
EVENTS_DMA.TX.END	0x168		Generated after all MAXCNT bytes have been transferred
EVENTS_DMA.TX.READY	0x16C		Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.
EVENTS_DMA.TX.BUSERROR	0x170		An error occurred during the bus transfer.
PUBLISH_STARTED	0x180		Publish configuration for event STARTED
PUBLISH_STOPPED	0x184		Publish configuration for event STOPPED
PUBLISH_END	0x188		Publish configuration for event END
PUBLISH_DMA.RX.END	0x1CC		Publish configuration for event END
PUBLISH_DMA.RX.READY	0x1D0		Publish configuration for event READY
PUBLISH_DMA.RX.BUSERROR	0x1D4		Publish configuration for event BUSERROR
PUBLISH_DMA.RX.MATCH[n]	0x1D8		Publish configuration for event MATCH[n]
PUBLISH_DMA.TX.END	0x1E8		Publish configuration for event END
PUBLISH_DMA.TX.READY	0x1EC		Publish configuration for event READY
PUBLISH_DMA.TX.BUSERROR	0x1F0		Publish configuration for event BUSERROR
SHORTS	0x200		Shortcuts between local events and tasks
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
ENABLE	0x500		Enable SPIM
PRESCALER	0x52C		The prescaler is used to set the SPI frequency.
CONFIG	0x554		Configuration register
IFTIMING.RXDELAY	0x5AC		Sample delay for input serial data on SDI
IFTIMING.CSNDUR	0x5B0		Minimum duration between edge of CSN and edge of SCK. When SHORTS.END_START is used, this is also the minimum duration CSN must stay high between transactions.
DCXCNT	0x5B4		DCX configuration
CSNPOL	0x5B8		Polarity of CSN output
ORC	0x5C0		Byte transmitted after TXD.MAXCNT bytes have been transmitted in the case when RXD.MAXCNT is greater than TXD.MAXCNT
PSEL.SCK	0x600		Pin select for SCK
PSEL.MOSI	0x604		Pin select for SDO signal
PSEL.MISO	0x608		Pin select for SDI signal
PSEL.DCX	0x60C		Pin select for DCX signal
PSEL.CSN	0x610		Pin select for CSN
DMA.RX.PTR	0x704		RAM buffer start address
DMA.RX.MAXCNT	0x708		Maximum number of bytes in channel buffer
DMA.RX.AMOUNT	0x70C		Number of bytes transferred in the last transaction, updated after the END event. Also updated after each MATCH event.
DMA.RX.LIST	0x714		EasyDMA list type
DMA.RX.TERMINATEONBUSERROR	0x71C		Terminate the transaction if a BUSERROR event is detected.
DMA.RX.BUSERRORADDRESS	0x720		Address of transaction that generated the last BUSERROR event.
DMA.RX.MATCH.CONFIG	0x724		Configure individual match events
DMA.RX.MATCH.CANDIDATE[n]	0x728		The data to look for - any match will trigger the MATCH[n] event, if enabled.
DMA.TX.PTR	0x73C		RAM buffer start address

Register	Offset	TZ	Description
DMA.TX.MAXCNT	0x740		Maximum number of bytes in channel buffer
DMA.TX.AMOUNT	0x744		Number of bytes transferred in the last transaction, updated after the END event. Also updated after each MATCH event.
DMA.TX.LIST	0x74C		EasyDMA list type
DMA.TX.TERMINATEONBUSERROR	0x754		Terminate the transaction if a BUSERROR event is detected.
DMA.TX.BUSERRORADDRESS	0x758		Address of transaction that generated the last BUSERROR event.

8.19.7.1 TASKS_START

Address offset: 0x000

Start SPI transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_START						Start SPI transaction																											
			Trigger	1				Trigger task																											

8.19.7.2 TASKS_STOP

Address offset: 0x004

Stop SPI transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_STOP						Stop SPI transaction																											
			Trigger	1				Trigger task																											

8.19.7.3 TASKS_SUSPEND

Address offset: 0x00C

Suspend SPI transaction

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_SUSPEND						Suspend SPI transaction																											
			Trigger	1				Trigger task																											

8.19.7.4 TASKS_RESUME

Address offset: 0x010

Resume SPI transaction

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																						A				
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																					
A	W	TASKS_RESUME			Resume SPI transaction																																					
			Trigger	1	Trigger task																																					

8.19.7.5 TASKS_DMA

Peripheral tasks.

8.19.7.5.1 TASKS_DMA.RX

Peripheral tasks.

8.19.7.5.1.1 TASKS_DMA.RX.ENABLEMATCH[n] (n=0..3)

Address offset: 0x030 + (n × 0x4)

Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																				A		
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	W	ENABLEMATCH				Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.																																
			Trigger	1		Trigger task																																

8.19.7.5.1.2 TASKS_DMA.RX.DISABLEMATCH[n] (n=0..3)

Address offset: 0x040 + (n × 0x4)

Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																				A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value				Description																												
A	W	DISABLEMATCH						Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.																												
			Trigger	1				Trigger task																												

8.19.7.6 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task **START**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
ID				B																								A				A				A				A			
Reset 0x00000000				0 0																																							
ID	R/W	Field	Value ID	Value		Description																																					
A	RW	CHIDX		[0..255]		DPPI channel that task START will subscribe to																																					
B	RW	EN																																									
			Disabled	0	Disable subscription																																						
			Enabled	1	Enable subscription																																						

Subscribe configuration for task **ENABLEMATCH[n]**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task ENABLEMATCH[n] will subscribe to																													
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.19.7.10.1.2 SUBSCRIBE_DMA.RX.DISABLEMATCH[n] (n=0..3)

Address offset: 0x0C0 + (n × 0x4)

Subscribe configuration for task **DISABLEMATCH[n]**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task DISABLEMATCH[n] will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.19.7.11 EVENTS_STARTED

Address offset: 0x100

SPI transaction has started

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	EVENTS_STARTED			SPI transaction has started																															
			NotGenerated	0	Event not generated																															
			Generated	1	Event generated																															

8.19.7.12 EVENTS_STOPPED

Address offset: 0x104

SPI transaction has stopped

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	EVENTS_STOPPED			SPI transaction has stopped																															
			NotGenerated	0	Event not generated																															
			Generated	1	Event generated																															

8.19.7.13 EVENTS_END

Address offset: 0x108

End of RXD buffer and TXD buffer reached

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_END			End of RXD buffer and TXD buffer reached																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.19.7.14 EVENTS_DMA

Peripheral events.

8.19.7.14.1 EVENTS_DMA.RX

Peripheral events.

8.19.7.14.1.1 EVENTS_DMA.RX.END

Address offset: 0x14C

Generated after all MAXCNT bytes have been transferred

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	END			Generated after all MAXCNT bytes have been transferred																															
			NotGenerated	0	Event not generated																															
			Generated	1	Event generated																															

8.19.7.14.1.2 EVENTS_DMA.RX.READY

Address offset: 0x150

Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.19.7.14.1.3 EVENTS_DMA.RX.BUSERROR

Address offset: 0x154

An error occurred during the bus transfer.

When this event is generated, the address which caused the error can be read from the **BUSERRORADDRESS** register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	BUSERROR			An error occurred during the bus transfer.																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.19.7.14.1.4 EVENTS_DMA.RX.MATCH[n] (n=0..3)

Address offset: $0x158 + (n \times 0x4)$

Pattern match is detected on the DMA data bus.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	MATCH			Pattern match is detected on the DMA data bus.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.19.7.14.2 EVENTS_DMA.TX

Peripheral events.

8.19.7.14.2.1 EVENTS_DMA.TX.END

Address offset: 0x168

Generated after all MAXCNT bytes have been transferred

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	END			Generated after all MAXCNT bytes have been transferred																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.19.7.14.2.2 EVENTS_DMA.TX.READY

Address offset: 0x16C

Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.19.7.14.2.3 EVENTS_DMA.TX.BUSERROR

Address offset: 0x170

An error occurred during the bus transfer.

When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	BUSERROR			An error occurred during the bus transfer.																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.19.7.15 PUBLISH_STARTED

Address offset: 0x180

Publish configuration for event **STARTED**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																							
A	RW	CHIDX		[0..255]				DPPI channel that event STARTED will publish to																																																							
B	RW	EN																																																													
			Disabled	0				Disable publishing																																																							
			Enabled	1				Enable publishing																																																							

8.19.7.16 PUBLISH_STOPPED

Address offset: 0x184

Publish configuration for event **STOPPED**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event STOPPED will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.19.7.17 PUBLISH_END

Address offset: 0x188

Publish configuration for event **END**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															

8.19.7.18 PUBLISH_DMA

Publish configuration for events

8.19.7.18.1 PUBLISH_DMA.RX

Publish configuration for events

8.19.7.18.1.1 PUBLISH_DMA.RX.END

Address offset: 0x1CC

Publish configuration for event **END**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event END will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.19.7.18.1.2 PUBLISH_DMA.RX.READY

Address offset: 0x1D0

Publish configuration for event **READY**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event READY will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.19.7.18.1.3 PUBLISH_DMA.RX.BUSERERROR

Address offset: 0x1D4

Publish configuration for event **BUSERERROR**

When this event is generated, the address which caused the error can be read from the BUSERERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event BUSERERROR will publish to																													
B	RW	EN																																	
			Disabled	0		Disable publishing																													
			Enabled	1		Enable publishing																													

8.19.7.18.1.4 PUBLISH_DMA.RX.MATCH[n] (n=0..3)

Address offset: 0x1D8 + (n × 0x4)

Publish configuration for event **MATCH[n]**

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					B																								A A A A A A A A							
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	CHIDX		[0..255]	DPPI channel that event MATCH[n] will publish to																															
B	RW	EN																																		
			Disabled	0	Disable publishing																															
			Enabled	1	Enable publishing																															

8.19.7.18.2 PUBLISH_DMA.TX

Publish configuration for events

8.19.7.18.2.1 PUBLISH_DMA.TX.END

Address offset: 0x1E8

Publish configuration for event **END**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																											
ID				B																								A				A	A	A	A	A	A	A																								
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																								
A	RW	CHIDX		[0..255]		DPPI channel that event END will publish to																																																								
B	RW	EN																																																												
			Disabled	0	Disable publishing																																																									
			Enabled	1	Enable publishing																																																									

8.19.7.18.2 PUBLISH_DMA.TX.READY

Address offset: 0x1EC

Publish configuration for event **READY**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A	A																								
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																							
A	RW	CHIDX		[0..255]				DPPI channel that event READY will publish to																																																							
B	RW	EN																																																													
			Disabled	0				Disable publishing																																																							
			Enabled	1				Enable publishing																																																							

8.19.7.18.3 PUBLISH_DMA.TX.BUSERROR

Address offset: 0x1F0

Publish configuration for event **BUSERROR**

When this event is generated, the address which caused the error can be read from the **BUSERRORADDRESS** register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event BUSERROR will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.19.7.19 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				I H G F E D C B																A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	END_START			Shortcut between event END and task START																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
B-E	RW	DMA_RX_MATCH[i]_DMA_RX_ENABLEMA +1)%4] (i=0..3)			Shortcut between event DMA.RX.MATCH[n] and task DMA.RX.ENABLEMATCH[(i+1)%4]																														
					Allows daisy-chaining match events.																														
			Disabled	0	Disable shortcut																														
		Enabled	1	Enable shortcut																															
F-I	RW	DMA_RX_MATCH[i]_DMA_RX_DISABLEMATCH[i] (i=0..3)			Shortcut between event DMA.RX.MATCH[i] and task DMA.RX.DISABLEMATCH[i]																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														

8.19.7.20 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	STARTED			Write '1' to enable interrupt for event STARTED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	STOPPED			Write '1' to enable interrupt for event STOPPED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	END			Write '1' to enable interrupt for event END																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	DMARXEND			Write '1' to enable interrupt for event DMARXEND																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
E	RW	DMARXREADY			Write '1' to enable interrupt for event DMARXREADY																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
F	RW	DMARXBUSERROR			Write '1' to enable interrupt for event DMARXBUSERROR																														
					When this event is generated, the address which caused the error can be read from the <code>BUSERRORADDRESS</code> register.																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
G-J	RW	DMARXMATCH[i] (i=0..3)			Write '1' to enable interrupt for event DMARXMATCH[i]																														

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			M L K J I H G F E D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
K	RW	DMATXEND			Write '1' to enable interrupt for event DMATXEND																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
L	RW	DMATXREADY			Write '1' to enable interrupt for event DMATXREADY																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
M	RW	DMATXBUSERROR			Write '1' to enable interrupt for event DMATXBUSERROR																													
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

8.19.7.21 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				M L K J I H G F E D																														C B A	
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	STARTED			Write '1' to disable interrupt for event STARTED																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	STOPPED			Write '1' to disable interrupt for event STOPPED																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	END			Write '1' to disable interrupt for event END																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	DMARXEND			Write '1' to disable interrupt for event DMARXEND																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
E	RW	DMARXREADY			Write '1' to disable interrupt for event DMARXREADY																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			M L K J I H G F E D																														C B A	
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
F	RW	DMARXBUSERROR			Write '1' to disable interrupt for event DMARXBUSERROR																													
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
G-J	RW	DMARXMATCH[i] (i=0..3)			Write '1' to disable interrupt for event DMARXMATCH[i]																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
K	RW	DMATXEND			Write '1' to disable interrupt for event DMATXEND																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
L	RW	DMATXREADY			Write '1' to disable interrupt for event DMATXREADY																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
M	RW	DMATXBUSERROR			Write '1' to disable interrupt for event DMATXBUSERROR																													
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

8.19.7.22 ENABLE

Address offset: 0x500

Enable SPIM

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																																				
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	ENABLE			Enable or disable SPIM																															
			Disabled	0	Disable SPIM																															
			Enabled	7	Enable SPIM																															

8.19.7.23 PRESCALER

Address offset: 0x52C

The prescaler is used to set the SPI frequency.

The prescaler divides the core clock by the divisor to make the SPI clock. The resulting frequency is given by 'core clock' / DIVISOR. Different instances of the SPIM might have different core clocks. The SPIM core clock and divisor limits is given in the instance table in [Instances](#) on page 563.

Note that a low prescaler setting may require changing the default [RXDELAY](#) value to ensure correct sampling.

Only even numbers is allowed for the divisor.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																														A	A	A	A	A	A	A	A
Reset 0x00000040				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0		
ID	R/W	Field	Value ID	Value				Description																													
A	RW	DIVISOR		2..126				Core clock to SCK divisor																													

8.19.7.24 CONFIG

Address offset: 0x554

Configuration register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	ORDER				Bit order																													
			MsbFirst	0	Most significant bit shifted out first																														
			LsbFirst	1	Least significant bit shifted out first																														
B	RW	CPHA				Serial clock (SCK) phase																													
			Leading	0	Sample on leading edge of clock, shift serial data on trailing edge																														
			Trailing	1	Sample on trailing edge of clock, shift serial data on leading edge																														
C	RW	CPOL				Serial clock (SCK) polarity																													
			ActiveHigh	0	Active high																														
			ActiveLow	1	Active low																														

8.19.7.25 IFTIMING.RXDELAY

Address offset: 0x5AC

Sample delay for input serial data on SDI

If the value is written larger than the maximum value, the maximum value will be used.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000002				0 1 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	RXDELAY		[7..0]	Sample delay for input serial data on SDI. The value specifies the number of SPIM core clock cycles delay from the the sampling edge of SCK (leading edge for CONFIG.CPHA = 0, trailing edge for CONFIG.CPHA = 1) until the input serial data is sampled. As an example, if RXDELAY = 0 and CONFIG.CPHA = 0, the input serial data is sampled on the rising edge of SCK.																														

8.19.7.26 IFTIMING.CSNDUR

Address offset: 0x5B0

Minimum duration between edge of CSN and edge of SCK. When SHORTS.END_START is used, this is also the minimum duration CSN must stay high between transactions.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A																															
Reset 0x00000002				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CSNDUR		[0xFF..0]				<p>Minimum duration between edge of CSN and edge of SCK. When SHORTS.END_START is used, this is the minimum duration CSN must stay high between transactions. The value is specified in number of SPIM core clock cycles.</p> <p>Note that for low values of CSNDUR, the system turnaround time will dominate the actual time between transactions.</p>																											

8.19.7.27 DCXCNT

Address offset: 0x5B4

DCX configuration

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	DCXCNT		0x0..0xF				This register specifies the number of command bytes preceding the data bytes. The PSEL.DCX line will be low during transmission of command bytes and high during transmission of data bytes. Value 0xF indicates that all bytes are command bytes.																											

8.19.7.28 CSNPOL

Address offset: 0x5B8

Polarity of CSN output

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CSNPOL[i] (i=0..0)			Polarity of CSN output																														
			LOW	0	Active low (idle state high)																														
			HIGH	1	Active high (idle state low)																														

8.19.7.29 ORC

Address offset: 0x5C0

Byte transmitted after TXD.MAXCNT bytes have been transmitted in the case when RXD.MAXCNT is greater than TXD.MAXCNT

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ORC						Byte transmitted after TXD.MAXCNT bytes have been transmitted in the case when RXD.MAXCNT is greater than TXD.MAXCNT.																											

8.19.7.30 PSEL.SCK

Address offset: 0x600

Pin select for SCK

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C																								B B B A A A A A							
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	PIN		[0..31]	Pin number																														
B	RW	PORT		[0..7]	Port number																														
C	RW	CONNECT			Connection																														
			Disconnected	1	Disconnect																														
			Connected	0	Connect																														

8.19.7.31 PSEL.MOSI

Address offset: 0x604

Pin select for SDO signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C																								B B B A A A A A							
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PIN		[0..31]				Pin number																											
B	RW	PORT		[0..7]				Port number																											
C	RW	CONNECT						Connection																											
			Disconnected	1				Disconnect																											
			Connected	0				Connect																											

8.19.7.32 PSEL.MISO

Address offset: 0x608

Pin select for SDI signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C																								B B B A A A A A							
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	PIN		[0..31]		Pin number																													
B	RW	PORT		[0..7]		Port number																													
C	RW	CONNECT				Connection																													
			Disconnected	1		Disconnect																													
			Connected	0		Connect																													

8.19.7.33 PSEL.DCX

Address offset: 0x60C

Pin select for DCX signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				C																								B				B	B	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	PIN		[0..31]		Pin number																															
B	RW	PORT		[0..7]		Port number																															
C	RW	CONNECT				Connection																															
			Disconnected	1		Disconnect																															
			Connected	0		Connect																															

8.19.7.34 PSEL.CSN

Address offset: 0x610

Pin select for CSN

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				C																								B				B	B	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
ID	R/W	Field	Value ID	Value				Description																													
A	RW	PIN		[0..31]				Pin number																													
B	RW	PORT		[0..7]				Port number																													
C	RW	CONNECT						Connection																													
			Disconnected	1				Disconnect																													
			Connected	0				Connect																													

8.19.7.35 DMA.RX.PTR

Address offset: 0x704

RAM buffer start address

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x20000000				0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						RAM buffer start address for this EasyDMA channel. This address is a word aligned Data RAM address.																											

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.19.7.36 DMA.RX.MAXCNT

Address offset: 0x708

Maximum number of bytes in channel buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

8.19.7.37 DMA.RX.AMOUNT

Address offset: 0x70C

Number of bytes transferred in the last transaction, updated after the END event.

Also updated after each MATCH event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	AMOUNT		[1..0xffff]				Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.																											

8.19.7.38 DMA.RX.LIST

Address offset: 0x714

EasyDMA list type

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A																												A	A		
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																															
A	RW	TYPE			List type																															
			Disabled	0	Disable EasyDMA list																															
			ArrayList	1	Use array list																															

8.19.7.39 DMA.RX.TERMINATEONBUSERROR

Address offset: 0x71C

Terminate the transaction if a BUSERROR event is detected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ENABLE																																	
			Disabled	0				Disable																											
			Enabled	1				Enable																											

8.19.7.40 DMA.RX.BUSERRORADDRESS

Address offset: 0x720

Address of transaction that generated the last BUSERROR event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	ADDRESS																																	

8.19.7.41 DMA.RX.MATCH

Registers to control the behavior of the pattern matcher engine

8.19.7.41.1 DMA.RX.MATCH.CONFIG

Address offset: 0x724

Configure individual match events

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				H G F E																D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-D	RW	ENABLE[i] (i=0..3)			Enable match filter i																														
			Disabled	0	Match filter disabled																														
			Enabled	1	Match filter enabled																														
E-H	RW	ONESHOT[i] (i=0..3)			Configure match filter i as one-shot or sticky																														
					One-shot match filters can be used together with shortcuts to check for continuous data sequences by disabling the filter if the next data is not a match.																														
					Note: The presence of these shorts depends on the configuration of the peripheral integrating this EasyDMA.																														
			Continuous	0	Match filter stays enabled until disabled by task																														
		Oneshot	1	Match filter stays enabled until next data word is received																															

Note: The presence of these shorts depends on the configuration of the peripheral integrating this EasyDMA.

8.19.7.41.2 DMA.RX.MATCH.CANDIDATE[n] (n=0..3)

Address offset: 0x728 + (n × 0x4)

The data to look for - any match will trigger the MATCH[n] event, if enabled.

Note: This register can be updated while a transfer is in progress, but the new value will not take effect until a match has been found or the transfer is done. That makes it possible to write a new set of match words which will be searched for immediately after the event triggers.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	RW	DATA																		Data to look for															

8.19.7.42 DMA.TX.PTR

Address offset: 0x73C

RAM buffer start address

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x20000000				0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						RAM buffer start address for this EasyDMA channel. This address is a word aligned Data RAM address.																											

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.19.7.43 DMA.TX.MAXCNT

Address offset: 0x740

Maximum number of bytes in channel buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW	MAXCNT		[1..0xffff]								Maximum number of bytes in channel buffer																							

8.19.7.44 DMA.TX.AMOUNT

Address offset: 0x744

Number of bytes transferred in the last transaction, updated after the END event.

Also updated after each MATCH event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	AMOUNT		[1..0xffff]				Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.																											

8.19.7.45 DMA.TX.LIST

Address offset: 0x74C

EasyDMA list type

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																														
ID				A																																			A	A																									
Reset 0x00000000				0																																			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																									
A	RW	TYPE						List type																																																									
			Disabled	0				Disable EasyDMA list																																																									
			ArrayList	1				Use array list																																																									

8.19.7.46 DMA.TX.TERMINATEONBUSERROR

Address offset: 0x754

Terminate the transaction if a BUSERROR event is detected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE																																	
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.19.7.47 DMA.TX.BUSERROADDRESS

Address offset: 0x758

Address of transaction that generated the last BUSERROR event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	ADDRESS																																	

8.20 SPIS — Serial peripheral interface target with EasyDMA

The SPI target peripheral (SPIS) with EasyDMA provides a full duplex, 4-wire synchronous serial communication interface.

The main features of SPIS are the following:

- EasyDMA direct transfer to and from RAM
- SPI mode [0..3]
- Individual selection of I/O pins
- Hardware-based semaphore mechanisms for synchronizing access to data buffers by SPIS and CPU

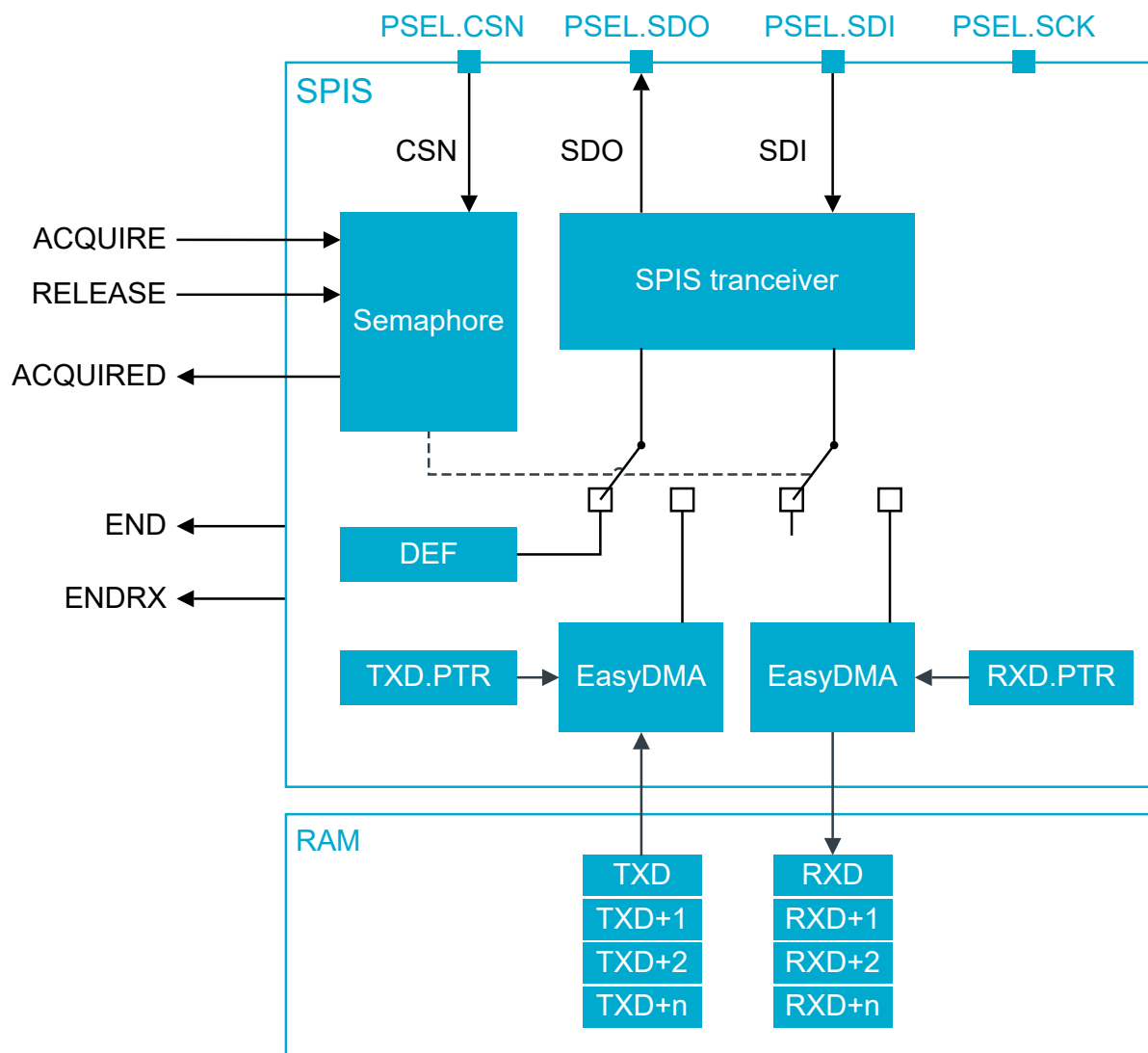


Figure 137: SPIS

8.20.1 SPI modes

SPIS supports SPI modes [0..3]. Modes CPOL and CPHA are set in the CONFIG register.

Mode	Clock polarity (CPOL)	Clock phase (CPHA)
SPI_MODE0	0 (Active High)	0 (Sample on Leading)
SPI_MODE1	0 (Active High)	1 (Sample on Trailing)
SPI_MODE2	1 (Active Low)	0 (Sample on Leading)
SPI_MODE3	1 (Active Low)	1 (Sample on Trailing)

Table 56: SPI modes

8.20.2 Shared resources

The SPIS peripheral shares registers and other resources with peripherals that have the same ID as SPIS. Before SPIS can be configured and used, all peripherals that have the same ID as SPIS must be disabled.

Disabling a peripheral with the same ID as SPIS will not reset any shared SPIS registers. Configure all SPIS registers to ensure they operate correctly.

See the Instantiation table in [Instantiation](#) on page 216 for details on peripherals and their IDs.

8.20.3 EasyDMA

SPIS implements EasyDMA for accessing RAM without CPU involvement.

SPIS implements the EasyDMA channels found in the following table.

Channel	Type	Register Cluster
TXD	READER	TXD
RXD	WRITER	RXD

Table 57: SPIS EasyDMA Channels

For detailed information regarding the use of EasyDMA, see [EasyDMA](#) on page 34.

If RXD.MAXCNT is greater than TXD.MAXCNT, the remaining transmitted bytes will contain the value defined in the ORC register.

The END event indicates that EasyDMA is finished accessing the RAM buffer.

8.20.4 SPIS operation

SPIS uses two memory pointers. RXD.PTR points to the RXD buffer (receive buffer) and TXD.PTR points to the TXD buffer (transmit buffer). Because these buffers are located in RAM, which can be accessed by both SPIS and the CPU, a hardware based semaphore mechanism is implemented to enable safe sharing.

The CPU must acquire the SPI semaphore before it can safely update the RXD.PTR and TXD.PTR pointers. The ACQUIRE task must be triggered for the CPU to receive the ACQUIRED event and have access to the semaphore. When the CPU has updated the RXD.PTR and TXD.PTR pointers, the CPU must release the semaphore before SPIS can acquire it.

The CPU releases the semaphore by triggering the RELEASE task, as illustrated in the following figure. Triggering the RELEASE task when the CPU does not have access to the semaphore will have no effect. See [Semaphore operation](#) on page 589 for more information.

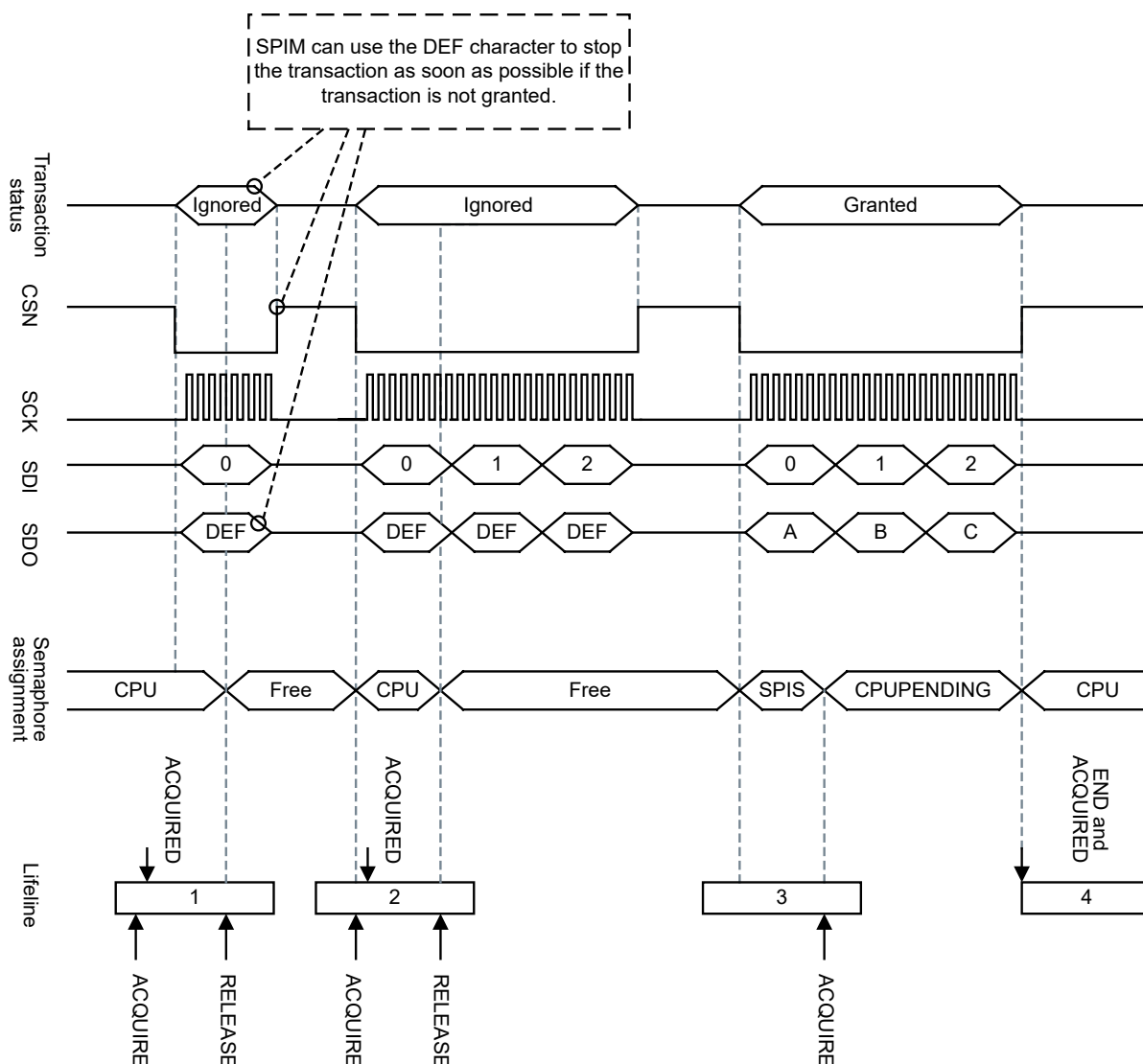


Figure 138: SPI transaction when shortcut between END and ACQUIRE is enabled

If the CPU is not able to reconfigure TXD.PTR and RXD.PTR between granted transactions, the same TX data will be clocked out and the RX buffers will be overwritten. To prevent this from happening, the END_ACQUIRE shortcut can be used. With this shortcut enabled, the semaphore will be handed over to the CPU automatically after the granted transaction has completed. This enables the CPU to update the TXPTR and RXPTR between every granted transaction.

The ENDRX event is generated when the RX buffer has been filled.

The RXD.MAXCNT register specifies the maximum number of bytes SPIS can receive in one granted transaction. If SPIS receives more than RXD.MAXCNT number of bytes, an OVERFLOW will be indicated in the STATUS register and the incoming bytes will be discarded.

The TXD.MAXCNT parameter specifies the maximum number of bytes SPIS can transmit in one granted transaction. If SPIS is forced to transmit more than TXD.MAXCNT number of bytes, an OVERREAD will be indicated in the STATUS register and the ORC character will be clocked out.

The RXD.AMOUNT and TXD.AMOUNT registers are updated when a granted transaction is complete. The TXD.AMOUNT register indicates how many bytes were read from the TX buffer in the last transaction. ORC (over-read) characters are not included in this number. Similarly, the RXD.AMOUNT register indicates how many bytes were written into the RX buffer in the last transaction.

8.20.5 Semaphore operation

The semaphore is a mechanism implemented inside the SPIS peripheral that prevents SPIS and CPU from accessing data buffers simultaneously.

By default, the semaphore is assigned to the CPU after the SPIS peripheral is enabled. An ACQUIRED event will not be generated for this initial semaphore handover. If the ACQUIRE task is triggered while the semaphore is assigned to the CPU, an ACQUIRED event will be generated immediately. The following figure illustrates the transitions between states in the semaphore based on the relevant tasks and events.

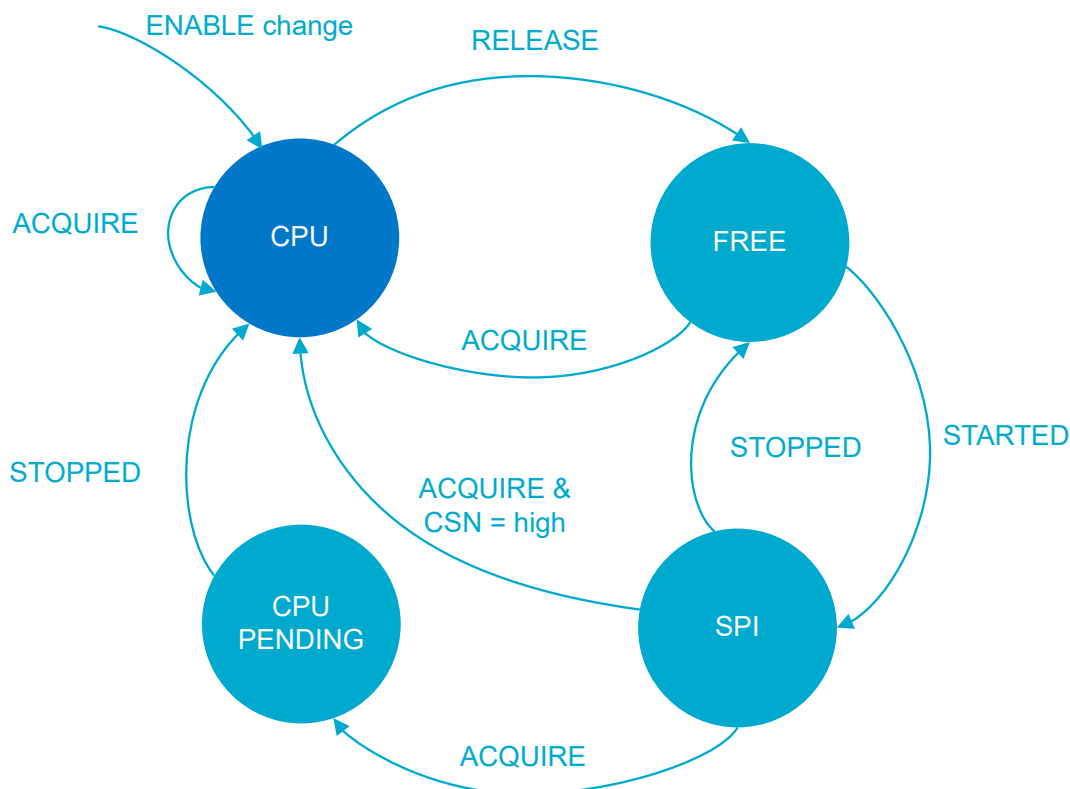


Figure 139: SPI semaphore FSM

Note: The semaphore mechanism does not prevent the CPU from performing read or write access to the RXD.PTR register, TXD.PTR registers, or RAM that these pointers are pointing to. The semaphore is only telling when these can be updated by the CPU so that safe sharing is achieved.

SPIS will try to acquire the semaphore when the STARTED event is detected. If SPIS does not obtain the semaphore, the transaction will be ignored and the semaphore is retained by the CPU. All incoming data on SDI will be discarded and the DEF (default) character will be clocked out on the SDO line throughout the transaction. This is also true if the semaphore is released by the CPU during the transaction. If a race condition occurs where the CPU and SPIS try to acquire the semaphore at the same time, as illustrated in lifeline item 2 in figure [SPI transaction when shortcut between END and ACQUIRE is enabled](#) on page 588, the CPU is given the semaphore.

If SPIS acquires the semaphore, the transaction will be granted. The incoming data on SDI will be stored in the RXD buffer and the data in the TXD buffer will be clocked out on SDO.

When a transaction is complete and CSN goes HIGH, SPIS will automatically release the semaphore and generate the END event.

SPIS can be granted multiple transactions in a row as long as the semaphore is available.

If the CPU tries to acquire the semaphore while it is assigned to SPIS, an immediate handover will not be granted. After the granted transaction is complete, SPIS releases the semaphore to the CPU. If the `END_ACQUIRE` shortcut is enabled and the CPU has triggered the `ACQUIRE` task during a granted transaction, only one `ACQUIRE` request will be served following the `END` event.

8.20.6 Pin configuration

The CSN, SCK, SDI, and SDO signals associated with SPIS are mapped to physical pins according to the configuration specified in the `PSEL.CSN`, `PSEL.SCK`, `PSEL.MOSI`, and `PSEL.MISO` registers, respectively. If the `CONNECT` field is set to `Disconnected`, the associated SPIS signal will not be connected to any physical pins.

These registers and their configurations are only used when SPIS is enabled, and retained as long as the device is in System ON mode. See [POWER — Power control](#) on page 97 for more information about power modes. When the peripheral is disabled, the pins behave as regular GPIOs, and use the configuration in their respective `OUT` bit field and `PIN_CNF[n]` register. Only configure `PSEL.CSN`, `PSEL.SCK`, `PSEL.MOSI`, and `PSEL.MISO` when SPIS is disabled.

Before enabling SPIS, the pins used by SPIS must be configured in the GPIO peripheral as described in [GPIO configuration before enabling peripheral](#) on page 590. This ensures that the pins are driven correctly if SPIS becomes temporarily disabled, or if the device enters System OFF mode. This configuration must be retained in the GPIO for the selected pins to be recognized by an external SPI controller.

The SDO line is set `HIGH` as long as SPIS is not selected with CSN.

Only one peripheral can be assigned to drive a particular GPIO pin at a time. Failing to do so may result in unpredictable behavior.

SPI signal	SPI pin	Direction	Output value	Comment
CSN	As specified in <code>PSEL.CSN</code>	Input	Not applicable	
SCK	As specified in <code>PSEL.SCK</code>	Input	Not applicable	
SDI	As specified in <code>PSEL.MOSI</code>	Input	Not applicable	
SDO	As specified in <code>PSEL.MISO</code>	Input	Not applicable	Emulates that SPIS is not selected.

Table 58: GPIO configuration before enabling peripheral

8.20.7 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
SPI00 : S	GLOBAL	0x5004A000	US	S	SA	No	SPI peripheral SPI00
SPI00 : NS		0x4004A000					
SPI20 : S	GLOBAL	0x500C6000	US	S	SA	No	SPI peripheral SPI20
SPI20 : NS		0x400C6000					
SPI21 : S	GLOBAL	0x500C7000	US	S	SA	No	SPI peripheral SPI21
SPI21 : NS		0x400C7000					
SPI22 : S	GLOBAL	0x500C8000	US	S	SA	No	SPI peripheral SPI22
SPI22 : NS		0x400C8000					
SPI30 : S	GLOBAL	0x50104000	US	S	SA	No	SPI peripheral SPI30
SPI30 : NS		0x40104000					

Configuration

Instance	Domain	Configuration
SPI00 : S	GLOBAL	Use GPIO port P2
SPI00 : NS		
SPI20 : S	GLOBAL	Use GPIO port P1, or dedicated pins on P2
SPI20 : NS		
SPI21 : S	GLOBAL	Use GPIO port P1, or dedicated pins on P2
SPI21 : NS		
SPI22 : S	GLOBAL	Use GPIO port P1, or dedicated pins on P2
SPI22 : NS		
SPI30 : S	GLOBAL	Use GPIO port P0
SPI30 : NS		

Register overview

Register	Offset	TZ	Description
TASKS_ACQUIRE	0x014		Acquire SPI semaphore
TASKS_RELEASE	0x018		Release SPI semaphore, enabling the SPI slave to acquire it
TASKS_DMA.RX.ENABLEMATCH[n]	0x030		Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.
TASKS_DMA.RX.DISABLEMATCH[n]	0x040		Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.
SUBSCRIBE_ACQUIRE	0x094		Subscribe configuration for task ACQUIRE
SUBSCRIBE_RELEASE	0x098		Subscribe configuration for task RELEASE
SUBSCRIBE_DMA.RX.ENABLEMATCH[n]	0x0B0		Subscribe configuration for task ENABLEMATCH[n]
SUBSCRIBE_DMA.RX.DISABLEMATCH[n]	0x0C0		Subscribe configuration for task DISABLEMATCH[n]
EVENTS_END	0x104		Granted transaction completed
EVENTS_ACQUIRED	0x118		Semaphore acquired
EVENTS_DMA.RX.END	0x14C		Generated after all MAXCNT bytes have been transferred
EVENTS_DMA.RX.READY	0x150		Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.
EVENTS_DMA.RX.BUSERROR	0x154		An error occurred during the bus transfer.
EVENTS_DMA.RX.MATCH[n]	0x158		Pattern match is detected on the DMA data bus.
EVENTS_DMA.TX.END	0x168		Generated after all MAXCNT bytes have been transferred

Register	Offset	TZ	Description
EVENTS_DMA.TX.READY	0x16C		Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.
EVENTS_DMA.TX.BUSERROR	0x170		An error occurred during the bus transfer.
PUBLISH_END	0x184		Publish configuration for event END
PUBLISH_ACQUIRED	0x198		Publish configuration for event ACQUIRED
PUBLISH_DMA.RX.END	0x1CC		Publish configuration for event END
PUBLISH_DMA.RX.READY	0x1D0		Publish configuration for event READY
PUBLISH_DMA.RX.BUSERROR	0x1D4		Publish configuration for event BUSERROR
PUBLISH_DMA.RX.MATCH[n]	0x1D8		Publish configuration for event MATCH[n]
PUBLISH_DMA.TX.END	0x1E8		Publish configuration for event END
PUBLISH_DMA.TX.READY	0x1EC		Publish configuration for event READY
PUBLISH_DMA.TX.BUSERROR	0x1F0		Publish configuration for event BUSERROR
SHORTS	0x200		Shortcuts between local events and tasks
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
SEMSTAT	0x400		Semaphore status register
STATUS	0x440		Status from last transaction
ENABLE	0x500		Enable SPI slave
CONFIG	0x554		Configuration register
DEF	0x55C		Default character. Character clocked out in case of an ignored transaction.
ORC	0x5C0		Over-read character
PSEL.SCK	0x600		Pin select for SCK
PSEL.MISO	0x604		Pin select for SDO signal
PSEL.MOSI	0x608		Pin select for SDI signal
PSEL.CSN	0x610		Pin select for CSN signal
DMA.RX.PTR	0x704		RAM buffer start address
DMA.RX.MAXCNT	0x708		Maximum number of bytes in channel buffer
DMA.RX.AMOUNT	0x70C		Number of bytes transferred in the last transaction, updated after the END event. Also updated after each MATCH event.
DMA.RX.TERMINATEONBUSERROR	0x71C		Terminate the transaction if a BUSERROR event is detected.
DMA.RX.BUSERRORADDRESS	0x720		Address of transaction that generated the last BUSERROR event.
DMA.RX.MATCH.CONFIG	0x724		Configure individual match events
DMA.RX.MATCH.CANDIDATE[n]	0x728		The data to look for - any match will trigger the MATCH[n] event, if enabled.
DMA.TX.PTR	0x73C		RAM buffer start address
DMA.TX.MAXCNT	0x740		Maximum number of bytes in channel buffer
DMA.TX.AMOUNT	0x744		Number of bytes transferred in the last transaction, updated after the END event. Also updated after each MATCH event.
DMA.TX.TERMINATEONBUSERROR	0x754		Terminate the transaction if a BUSERROR event is detected.
DMA.TX.BUSERRORADDRESS	0x758		Address of transaction that generated the last BUSERROR event.

8.20.7.1 TASKS_ACQUIRE

Address offset: 0x014

Acquire SPI semaphore

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_ACQUIRE						Acquire SPI semaphore																											
			Trigger	1				Trigger task																											

8.20.7.2 TASKS_RELEASE

Address offset: 0x018

Release SPI semaphore, enabling the SPI slave to acquire it

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	W	TASKS_RELEASE			Release SPI semaphore, enabling the SPI slave to acquire it																																	
			Trigger	1	Trigger task																																	

8.20.7.3 TASKS_DMA

Peripheral tasks.

8.20.7.3.1 TASKS_DMA.RX

Peripheral tasks.

8.20.7.3.1.1 TASKS_DMA.RX.ENABLEMATCH[n] (n=0..3)

Address offset: 0x030 + (n × 0x4)

Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	ENABLEMATCH			Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.																															
			Trigger	1	Trigger task																															

8.20.7.3.1.2 TASKS_DMA.RX.DISABLEMATCH[n] (n=0..3)

Address offset: 0x040 + (n × 0x4)

Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	W	DISABLEMATCH			Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.																																	
			Trigger	1	Trigger task																																	

8.20.7.4 SUBSCRIBE_ACQUIRE

Address offset: 0x094

Subscribe configuration for task [ACQUIRE](#)

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																													
ID					B																								A										A	A	A	A	A	A	A																				
Reset 0x00000000					0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value	ID	Value		Description																																																										
A	RW	CHIDX			[0..255]		DPPI channel that task ACQUIRE will subscribe to																																																										
B	RW	EN																																																															
			Disabled	0	Disable subscription																																																												
			Enabled	1	Enable subscription																																																												

8.20.7.5 SUBSCRIBE_RELEASE

Address offset: 0x098

Subscribe configuration for task **RELEASE**

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
ID					B																								A A A A A A A A A A															
Reset 0x00000000					0 0																																							
ID	R/W	Field	Value	ID	Value				Description																																			
A	RW	CHIDX			[0..255]				DPPI channel that task RELEASE will subscribe to																																			
B	RW	EN																																										
			Disabled	0	Disable subscription																																							
			Enabled	1	Enable subscription																																							

8.20.7.6 SUBSCRIBE_DMA

Subscribe configuration for tasks

8.20.7.6.1 SUBSCRIBE_DMA.RX

Subscribe configuration for tasks

8.20.7.6.1.1 SUBSCRIBE_DMA.RX.ENABLEMATCH[n] (n=0..3)

Address offset: 0x0B0 + (n × 0x4)

Subscribe configuration for task **ENABLEMATCH[n]**

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																													
ID					B																								A										A	A	A	A	A	A	A																				
Reset 0x00000000					0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value	ID	Value		Description																																																										
A	RW	CHIDX			[0..255]		DPPI channel that task ENABLEMATCH[n] will subscribe to																																																										
B	RW	EN																																																															
			Disabled	0	Disable subscription																																																												
			Enabled	1	Enable subscription																																																												

8.20.7.6.1.2 SUBSCRIBE_DMA.RX.DISABLEMATCH[n] (n=0..3)

Address offset: 0x0C0 + (n × 0x4)

Subscribe configuration for task **DISABLEMATCH[n]**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that task <code>DISABLEMATCH[n]</code> will subscribe to																																																									
B	RW	EN																																																													
			Disabled	0	Disable subscription																																																										
			Enabled	1	Enable subscription																																																										

8.20.7.7 EVENTS_END

Address offset: 0x104

Granted transaction completed

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_END			Granted transaction completed																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.20.7.8 EVENTS_ACQUIRED

Address offset: 0x118

Semaphore acquired

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_ACQUIRED			Semaphore acquired																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.20.7.9 EVENTS_DMA

Peripheral events.

8.20.7.9.1 EVENTS_DMA.RX

Peripheral events.

8.20.7.9.1.1 EVENTS_DMA.RX.END

Address offset: 0x14C

Generated after all MAXCNT bytes have been transferred

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A																															
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																															
A	RW	END			Generated after all MAXCNT bytes have been transferred																															
			NotGenerated	0	Event not generated																															
			Generated	1	Event generated																															

8.20.7.9.1.2 EVENTS_DMA.RX.READY

Address offset: 0x150

Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.20.7.9.1.3 EVENTS_DMA.RX.BUSERROR

Address offset: 0x154

An error occurred during the bus transfer.

When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	BUSERROR			An error ocured during the bus transfer.																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.20.7.9.1.4 EVENTS_DMA.RX.MATCH[n] (n=0..3)

Address offset: 0x158 + (n × 0x4)

Pattern match is detected on the DMA data bus.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	MATCH						Pattern match is detected on the DMA data bus.																											
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.20.7.9.2 EVENTS_DMA.TX

Peripheral events.

8.20.7.9.2.1 EVENTS_DMA.TX.END

Address offset: 0x168

Generated after all MAXCNT bytes have been transferred

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																					A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																
A	RW	END			Generated after all MAXCNT bytes have been transferred																																
			NotGenerated	0	Event not generated																																
			Generated	1	Event generated																																

8.20.7.9.2.2 EVENTS_DMA.TX.READY

Address offset: 0x16C

Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																													
A	RW	READY				Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.																													
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.20.7.9.2.3 EVENTS_DMA.TX.BUSERROR

Address offset: 0x170

An error occurred during the bus transfer.

When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	BUSERROR			An error ocured during the bus transfer.																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.20.7.10 PUBLISH_END

Address offset: 0x184

Publish configuration for event [END](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event END will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.20.7.11 PUBLISH_ACQUIRED

Address offset: 0x198

Publish configuration for event **ACQUIRED**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event ACQUIRED will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.20.7.12 PUBLISH_DMA

Publish configuration for events

8.20.7.12.1 PUBLISH_DMA.RX

Publish configuration for events

8.20.7.12.1.1 PUBLISH_DMA.RX.END

Address offset: 0x1CC

Publish configuration for event **END**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event END will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.20.7.12.1.2 PUBLISH_DMA.RX.READY

Address offset: 0x1D0

Publish configuration for event **READY**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event READY will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.20.7.12.1.3 PUBLISH_DMA.RX.BUSERERROR

Address offset: 0x1D4

Publish configuration for event **BUSERERROR**

When this event is generated, the address which caused the error can be read from the BUSERERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event BUSERERROR will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.20.7.12.1.4 PUBLISH_DMA.RX.MATCH[n] (n=0..3)

Address offset: 0x1D8 + (n × 0x4)

Publish configuration for event **MATCH[n]**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event MATCH[n] will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.20.7.12.2 PUBLISH_DMA.TX

Publish configuration for events

8.20.7.12.2.1 PUBLISH_DMA.TX.END

Address offset: 0x1E8

Publish configuration for event **END**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event END will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.20.7.12.2 PUBLISH_DMA.TX.READY

Address offset: 0x1EC

Publish configuration for event **READY**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																													
ID				B																								A												A	A	A	A	A	A	A																		
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																										
A	RW	CHIDX		[0..255]		DPPI channel that event READY will publish to																																																										
B	RW	EN																																																														
			Disabled	0		Disable publishing																																																										
			Enabled	1		Enable publishing																																																										

8.20.7.12.3 PUBLISH_DMA.TX.BUSERROR

Address offset: 0x1F0

Publish configuration for event **BUSERROR**

When this event is generated, the address which caused the error can be read from the **BUSERRORADDRESS** register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event BUSERROR will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.20.7.13 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				I H G F E D C B																										A					
Reset 0x00000000				0 0																															
D	R/W	Field	Value ID	Value	Description																														
A	RW	END_ACQUIRE			Shortcut between event END and task ACQUIRE																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
B-E	RW	DMA_RX_MATCH[i]_DMA_RX_ENABLEMA +1)%4] (i=0..3)			Shortcut between event DMA.RX.MATCH[n] and task DMA.RX.ENABLEMATCH[(i+1)%4]																														
					Allows daisy-chaining match events.																														
			Disabled	0	Disable shortcut																														
		Enabled	1	Enable shortcut																															
F-I	RW	DMA_RX_MATCH[i]_DMA_RX_DISABLEMATCH[i] (i=0..3)			Shortcut between event DMA.RX.MATCH[n] and task DMA.RX.DISABLEMATCH[n]																														
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														

8.20.7.14 INTENSET

Address offset: 0x304

Enable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																		
ID			L K J I H G F E D C																										B				A				
Reset 0x00000000			0 0																																		
ID	R/W	Field	Value ID	Value	Description																																
A	RW	END			Write '1' to enable interrupt for event END																																
			Set	1	Enable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
B	RW	ACQUIRED			Write '1' to enable interrupt for event ACQUIRED																																
			Set	1	Enable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
C	RW	DMARXEND			Write '1' to enable interrupt for event DMARXEND																																
			Set	1	Enable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
D	RW	DMARXREADY			Write '1' to enable interrupt for event DMARXREADY																																
			Set	1	Enable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
E	RW	DMARXBUSEROR			Write '1' to enable interrupt for event DMARXBUSEROR																																
					When this event is generated, the address which caused the error can be read from the BUSERORADDRESS register.																																
			Set	1	Enable																																
			Disabled	0	Read: Disabled																																
F-I	RW	DMARXMATCH[i] (i=0..3)			Write '1' to enable interrupt for event DMARXMATCH[i]																																
			Set	1	Enable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
J	RW	DMATXEND			Write '1' to enable interrupt for event DMATXEND																																

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																			
ID				L K J I H G F E D C																								B								A			
Reset 0x00000000				0 0																																			
ID	R/W	Field	Value ID	Value		Description																																	
			Set	1		Enable																																	
			Disabled	0		Read: Disabled																																	
			Enabled	1		Read: Enabled																																	
K	RW	DMATXREADY				Write '1' to enable interrupt for event DMATXREADY																																	
			Set	1		Enable																																	
			Disabled	0		Read: Disabled																																	
			Enabled	1		Read: Enabled																																	
L	RW	DMATXBUSEROR				Write '1' to enable interrupt for event DMATXBUSEROR																																	
						When this event is generated, the address which caused the error can be read from the BUSERORADDRESS register.																																	
			Set	1		Enable																																	
			Disabled	0		Read: Disabled																																	
			Enabled	1		Read: Enabled																																	

8.20.7.15 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			L K J I H G F E D C																B								A							
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	END			Write '1' to disable interrupt for event END																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
B	RW	ACQUIRED			Write '1' to disable interrupt for event ACQUIRED																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
C	RW	DMARXEND			Write '1' to disable interrupt for event DMARXEND																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
D	RW	DMARXREADY			Write '1' to disable interrupt for event DMARXREADY																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
E	RW	DMARXBUSEROR			Write '1' to disable interrupt for event DMARXBUSEROR																													
					When this event is generated, the address which caused the error can be read from the BUSERORADDRESS register.																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
F-I	RW	DMARXMATCH[i] (i=0..3)			Write '1' to disable interrupt for event DMARXMATCH[i]																													
			Clear	1	Disable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				L K J I H G F E D C																B								A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
J	RW	DMATXEND			Write '1' to disable interrupt for event DMATXEND																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
K	RW	DMATXREADY			Write '1' to disable interrupt for event DMATXREADY																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
L	RW	DMATXBUSERROR			Write '1' to disable interrupt for event DMATXBUSERROR																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.20.7.16 SEMSTAT

Address offset: 0x400

Semaphore status register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000001				0 1																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	SEMSTAT			Semaphore status																														
			Free	0	Semaphore is free																														
			CPU	1	Semaphore is assigned to CPU																														
			SPIS	2	Semaphore is assigned to SPI slave																														
			CPUPending	3	Semaphore is assigned to SPI but a handover to the CPU is pending																														

8.20.7.17 STATUS

Address offset: 0x440

Status from last transaction

Individual bits are cleared by writing a '1' to the bits that shall be cleared.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	OVERREAD			TX buffer over-read detected, and prevented																														
			NotPresent	0	Read: error not present																														
			Present	1	Read: error present																														
			Clear	1	Write: clear error on writing '1'																														
B	RW	OVERFLOW			RX buffer overflow detected, and prevented																														
			NotPresent	0	Read: error not present																														
			Present	1	Read: error present																														
			Clear	1	Write: clear error on writing '1'																														

8.20.7.18 ENABLE

Address offset: 0x500

Enable SPI slave

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE			Enable or disable SPI slave																														
			Disabled	0	Disable SPI slave																														
			Enabled	2	Enable SPI slave																														

8.20.7.19 CONFIG

Address offset: 0x554

Configuration register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ORDER			Bit order																														
			MsbFirst	0	Most significant bit shifted out first																														
			LsbFirst	1	Least significant bit shifted out first																														
B	RW	CPHA			Serial clock (SCK) phase																														
			Leading	0	Sample on leading edge of clock, shift serial data on trailing edge																														
			Trailing	1	Sample on trailing edge of clock, shift serial data on leading edge																														
C	RW	CPOL			Serial clock (SCK) polarity																														
			ActiveHigh	0	Active high																														
			ActiveLow	1	Active low																														

8.20.7.20 DEF

Address offset: 0x55C

Default character. Character clocked out in case of an ignored transaction.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																													A A A A A A A A							
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	DEF			Default character. Character clocked out in case of an ignored transaction.																															

8.20.7.21 ORC

Address offset: 0x5C0

Over-read character

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ORC						Over-read character. Character clocked out after an over-read of the transmit buffer.																											

8.20.7.22 PSEL.SCK

Address offset: 0x600

Pin select for SCK

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C																								B B B A A A A							
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PIN		[0..31]				Pin number																											
B	RW	PORT		[0..7]				Port number																											
C	RW	CONNECT						Connection																											
			Disconnected	1				Disconnect																											
			Connected	0				Connect																											

8.20.7.23 PSEL.MISO

Address offset: 0x604

Pin select for SDO signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C																								B B B A A A A							
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PIN		[0..31]				Pin number																											
B	RW	PORT		[0..7]				Port number																											
C	RW	CONNECT						Connection																											
			Disconnected	1				Disconnect																											
			Connected	0				Connect																											

8.20.7.24 PSEL.MOSI

Address offset: 0x608

Pin select for SDI signal

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C																								B B B A A A A							
Reset 0xFFFFFFFF				1 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PIN		[0..31]				Pin number																											
B	RW	PORT		[0..7]				Port number																											
C	RW	CONNECT						Connection																											
			Disconnected	1				Disconnect																											
			Connected	0				Connect																											

8.20.7.25 PSEL.CSN

Address offset: 0x610

Pin select for CSN signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				C																												B	B	B	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1				
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	PIN		[0..31]		Pin number																																	
B	RW	PORT		[0..7]		Port number																																	
C	RW	CONNECT				Connection																																	
			Disconnected	1		Disconnect																																	
			Connected	0		Connect																																	

8.20.7.26 DMA.RX.PTR

Address offset: 0x704

RAM buffer start address

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.20.7.27 DMA.RX.MAXCNT

Address offset: 0x708

Maximum number of bytes in channel buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																				A A A A A A A A A A A A A A A A A A A A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	MAXCNT		[1..0xffff]				Maximum number of bytes in channel buffer																											

8.20.7.28 DMA.RX.AMOUNT

Address offset: 0x70C

Number of bytes transferred in the last transaction, updated after the END event.

Also updated after each MATCH event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	AMOUNT		[1..0xffff]				Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.																											

8.20.7.29 DMA.RX.TERMINATEONBUSERROR

Address offset: 0x71C

Terminate the transaction if a BUSERROR event is detected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ENABLE																																	
			Disabled	0				Disable																											
			Enabled	1				Enable																											

8.20.7.30 DMA.RX.BUSERRORADDRESS

Address offset: 0x720

Address of transaction that generated the last BUSERROR event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	ADDRESS																																	

8.20.7.31 DMA.RX.MATCH

Registers to control the behavior of the pattern matcher engine

8.20.7.31.1 DMA.RX.MATCH.CONFIG

Address offset: 0x724

Configure individual match events

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				H G F E																D C B A															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value	Description																														
A-D	RW	ENABLE[i] (i=0..3)			Enable match filter i																														
			Disabled	0	Match filter disabled																														
			Enabled	1	Match filter enabled																														
E-H	RW	ONESHOT[i] (i=0..3)			Configure match filter i as one-shot or sticky																														
					One-shot match filters can be used together with shortcuts to check for continuous data sequences by disabling the filter if the next data is not a match.																														
					Note: The presence of these shorts depends on the configuration of the peripheral integrating this EasyDMA.																														
			Continuous	0	Match filter stays enabled until disabled by task																														
			Oneshot	1	Match filter stays enabled until next data word is received																														

8.20.7.31.2 DMA.RX.MATCH.CANDIDATE[n] (n=0..3)

Address offset: 0x728 + (n × 0x4)

The data to look for - any match will trigger the MATCH[n] event, if enabled.

Note: This register can be updated while a transfer is in progress, but the new value will not take effect until a match has been found or the transfer is done. That makes it possible to write a new set of match words which will be searched for immediately after the event triggers.

Bit number									31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID									A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID						Value						Description																										
A	RW	DATA													Data to look for																										

8.20.7.32 DMA.TX.PTR

Address offset: 0x73C

RAM buffer start address

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x20000000				0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						RAM buffer start address for this EasyDMA channel. This address is a word aligned Data RAM address.																											

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.20.7.33 DMA.TX.MAXCNT

Address offset: 0x740

Maximum number of bytes in channel buffer

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value				Description																													
A	RW	MAXCNT		[1..0xffff]				Maximum number of bytes in channel buffer																													

8.20.7.34 DMA.TX.AMOUNT

Address offset: 0x744

Number of bytes transferred in the last transaction, updated after the END event.

Also updated after each MATCH event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	AMOUNT		[1..0xffff]				Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.																											

8.20.7.35 DMA.TX.TERMINATEONBUSERROR

Address offset: 0x754

Terminate the transaction if a BUSERROR event is detected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ENABLE																																	
			Disabled	0				Disable																											
			Enabled	1				Enable																											

8.20.7.36 DMA.TX.BUSERRORADDRESS

Address offset: 0x758

Address of transaction that generated the last BUSERROR event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	ADDRESS																																	

8.21 TEMP — Temperature sensor

The temperature sensor (TEMP) measures die temperature over the temperature range of the device. Linearity compensation can be implemented if required by the application.

The main features of TEMP are:

- Temperature range is greater than or equal to operating temperature of the device

- Resolution is 0.25 degrees
- TEMP analog electronics power down after temperature measurement is completed

TEMP is started by triggering the START task.

When the temperature measurement is completed, a DATARDY event will be generated and the result of the measurement can be read from the TEMP register.

To achieve the measurement accuracy stated in the electrical specification, the crystal oscillator must be selected as the HFCLK source, see [CLOCK — Clock control](#) on page 76 for more information.

When the temperature measurement is completed, TEMP analog electronics power down to save power.

TEMP only supports one-shot operation, meaning that every TEMP measurement has to be explicitly started using the START task.

8.21.1 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
TEMP : S	GLOBAL	0x500D7000	US	S	NA	No	Temperature sensor TEMP
TEMP : NS		0x400D7000					

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Start temperature measurement
TASKS_STOP	0x004		Stop temperature measurement
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
EVENTS_DATARDY	0x100		Temperature measurement complete, data ready
PUBLISH_DATARDY	0x180		Publish configuration for event DATARDY
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
TEMP	0x508		Temperature in °C (0.25° steps)
A0	0x520		Slope of 1st piece wise linear function
A1	0x524		Slope of 2nd piece wise linear function
A2	0x528		Slope of 3rd piece wise linear function
A3	0x52C		Slope of 4th piece wise linear function
A4	0x530		Slope of 5th piece wise linear function
A5	0x534		Slope of 6th piece wise linear function
A6	0x538		Slope of 7th piece wise linear function
B0	0x540		y-intercept of 1st piece wise linear function
B1	0x544		y-intercept of 2nd piece wise linear function
B2	0x548		y-intercept of 3rd piece wise linear function
B3	0x54C		y-intercept of 4th piece wise linear function
B4	0x550		y-intercept of 5th piece wise linear function
B5	0x554		y-intercept of 6th piece wise linear function
B6	0x558		y-intercept of 7th piece wise linear function
T0	0x560		End point of 1st piece wise linear function
T1	0x564		End point of 2nd piece wise linear function
T2	0x568		End point of 3rd piece wise linear function

Register	Offset	TZ	Description
T3	0x56C		End point of 4th piece wise linear function
T4	0x570		End point of 5th piece wise linear function
T5	0x574		End point of 6th piece wise linear function

8.21.1.1 TASKS_START

Address offset: 0x000

Start temperature measurement

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A																																	
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	W	TASKS_START			Start temperature measurement																																	
			Trigger	1	Trigger task																																	

8.21.1.2 TASKS_STOP

Address offset: 0x004

Stop temperature measurement

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_STOP						Stop temperature measurement																											
			Trigger	1				Trigger task																											

8.21.1.3 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task **START**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																		
ID				B																								A																A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
ID	R/W	Field	Value ID	Value				Description																																													
A	RW	CHIDX		[0..255]				DPPI channel that task START will subscribe to																																													
B	RW	EN																																																			
			Disabled	0				Disable subscription																																													
			Enabled	1				Enable subscription																																													

8.21.1.4 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task **STOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task STOP will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.21.1.5 EVENTS_DATARDY

Address offset: 0x100

Temperature measurement complete, data ready

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_DATARDY			Temperature measurement complete, data ready																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.21.1.6 PUBLISH_DATARDY

Address offset: 0x180

Publish configuration for event **DATARDY**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event DATARDY will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.21.1.7 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	DATARDY			Write '1' to enable interrupt for event DATARDY																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.21.1.8 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	DATARDY			Write '1' to disable interrupt for event DATARDY																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.21.1.9 TEMP

Address offset: 0x508

Temperature in °C (0.25° steps)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	TEMP						Temperature in °C (0.25° steps)																											
																																Result of temperature measurement. Die temperature in °C, 2's complement format, 0.25 °C steps			
																																11 bits sign extended to 32 bits, with 2 LSBs as fractional bits.			
																																Decision point: DATARDY			

8.21.1.10 A0

Address offset: 0x520

Slope of 1st piece wise linear function

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A																															
Reset 0x000002D6					0 1 0 1 1 0 1 0 1 1 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	A0			Slope of 1st piece wise linear function																															

8.21.1.11 A1

Address offset: 0x524

Slope of 2nd piece wise linear function

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A																															
Reset 0x0000032D					0 1 1 0 0 1 0 1 1 0 1																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	A1			Slope of 2nd piece wise linear function																															

8.21.1.12 A2

Address offset: 0x528

Slope of 3rd piece wise linear function

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
ID																									A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000384					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0								
ID	R/W	Field	Value ID		Value		Description																																					
A	RW	A2					Slope of 3rd piece wise linear function																																					

8.21.1.13 A3

Address offset: 0x52C

Slope of 4th piece wise linear function

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
ID																									A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x000003E9					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0	1								
ID	R/W	Field	Value ID		Value		Description																																					
A	RW	A3					Slope of 4th piece wise linear function																																					

8.21.1.14 A4

Address offset: 0x530

Slope of 5th piece wise linear function

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
ID																									A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x0000046F					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	1	1	1	1	1							
ID	R/W	Field	Value ID		Value		Description																																					
A	RW	A4					Slope of 5th piece wise linear function																																					

8.21.1.15 A5

Address offset: 0x534

Slope of 6th piece wise linear function

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								
ID																									A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000522					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0							
ID	R/W	Field	Value ID		Value		Description																																					
A	RW	A5					Slope of 6th piece wise linear function																																					

8.21.1.16 A6

Address offset: 0x538

Slope of 7th piece wise linear function

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																					A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x000005B7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	1	0	1	1	1
ID	R/W	Field	Value ID	Value	Description																											
A	RW	A6			Slope of 7th piece wise linear function																											

8.21.1.17 B0

Address offset: 0x540

y-intercept of 1st piece wise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

8.21.1.18 B1

Address offset: 0x544

y-intercept of 2nd piece wise linear function

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																					A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000F76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	1	1	1	0	1	1	0
ID	R/W	Field	Value ID	Value	Description																											
A	RW	B1			y-intercept of 2nd piece wise linear function																											

8.21.1.19 B2

Address offset: 0x548

y-intercept of 3rd piece wise linear function

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0							
ID																									A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000F8A					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	1	0	1	0						
ID	R/W	Field	Value ID	Value	Description																																						
A	RW	B2			y-intercept of 3rd piece wise linear function																																						

8.21.1.20 B3

Address offset: 0x54C

y-intercept of 4th piece wise linear function

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A																															

8.21.1.21 B4

Address offset: 0x550

y-intercept of 5th piece wise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

8.21.1.22 B5

Address offset: 0x554

y-intercept of 6th piece wise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000207				0 1 1 1 1																															
ID	R/W	Field	Value ID	ValueDescription																															
A	RW	B5		v-intercept of 6th piece wise linear function																															

8.21.1.23 B6

Address offset: 0x558

y-intercept of 7th piece wise linear function

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0							
ID																									A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000558					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	0	0	0						
ID	R/W	Field	Value ID	Value	Description																																						
A	RW	B6			y-intercept of 7th piece wise linear function																																						

8.21.1.24 T0

Address offset: 0x560

End point of 1st piece wise linear function

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

8.21.1.25 T1

Address offset: 0x564

End point of 2nd piece wise linear function

[illegible]

8.21.1.26 T2

Address offset: 0x568

End point of 3rd piece wise linear function

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0											
ID																													A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x0000001F					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1									
ID	R/W	Field	Value	ID	Value	Description																																									
A	RW	T2				End point of 3rd piece wise linear function																																									

8.21.1.27 T3

Address offset: 0x56C

End point of 4th piece wise linear function

Bit number						31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0					
ID																														A	A	A	A	A	A	A	A					
Reset 0x00000038						0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0					
ID	R/W	Field	Value ID			Value			Description																																	
A	RW	T3							End point of 4th piece wise linear function																																	

8.21.1.28 T4

Address offset: 0x570

End point of 5th piece wise linear function

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0												
ID																													A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x0000004F					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	1											
ID	R/W	Field		Value	ID	Value		Description																																								
A	RW	T4				End point of 5th piece wise linear function																																										

8.21.1.29 T5

Address offset: 0x574

End point of 6th piece wise linear function

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0											
ID																													A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000066					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1	0										
ID	R/W	Field	Value	ID	Value		Description																																								
A	RW	T5					End point of 6th piece wise linear function																																								

8.22 TIMER — Timer/counter

The TIMER peripheral is a general purpose timer allowing time intervals to be defined by user input.

The main features of TIMER are:

- Two modes of operation: Timer mode and Counter mode
- Multiple capture/compare registers
- Compare event for every capture/compare registers
- 4-bit (1/2X) prescaler
- Configurable number of bits used by the TIMER: 8, 16, 24 or 32 bits
- TIMER runs on the high-frequency clock source (HFCLK)

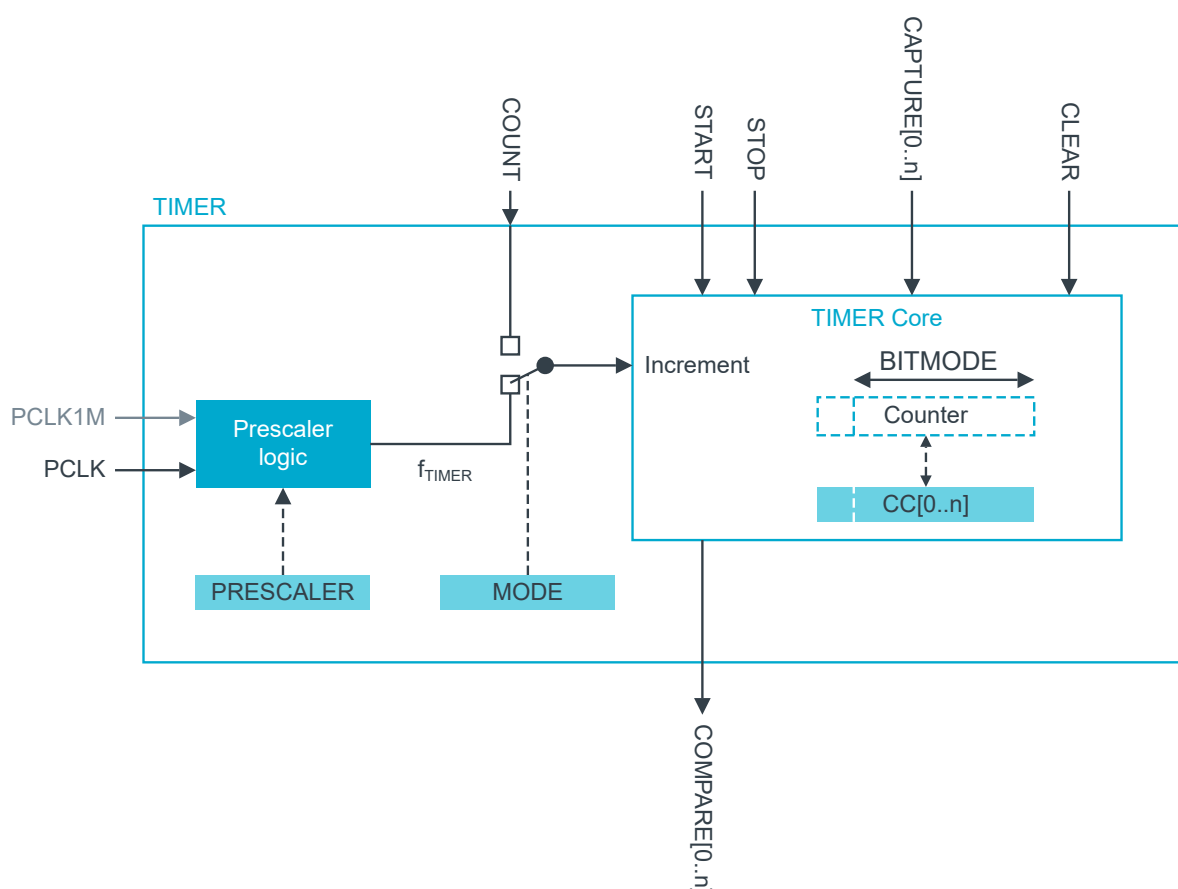


Figure 140: Block schematic for timer/counter

TIMER runs on the high-frequency clock source (HFCLK) and includes a four-bit (1/2X) prescaler that can divide the timer input clock (PCLK) from the HFCLK controller. The TIMER base frequency is always given as PCLK divided by the prescaler value.

The PPI system allows a TIMER event to trigger a task on another system peripheral on the device. The PPI system also enables the TIMER task/event feature to generate periodic output and PWM signals to any GPIO. The number of GPIO inputs or outputs used at the same time is limited by the number of GPIOTE channels.

TIMER can operate in two modes: Timer mode and Counter mode. In both modes, TIMER is started by triggering the START task, and stopped by triggering the STOP task. After TIMER stops, it can resume timing/counting by triggering the START task again. When timing/counting resumes, TIMER continues from the value it was on prior to stopping.

In Timer mode, TIMER's internal Counter register is incremented by one for every tick of the timer frequency f_{TIMER} , as illustrated in [Block schematic for timer/counter](#) on page 618. The timer frequency is derived from PCLK as shown in the following example, using the values specified in the PRESCALER register.

$$f_{\text{TIMER}} = \text{PCLK} / (2^{\text{PRESCALER}})$$

For timers using PCLK16M as PCLK, when $f_{\text{TIMER}} \leq 1$ MHz, TIMER uses PCLK1M instead of PCLK for reduced power consumption. Clock source selection between PCLK and PCLK1M is automatic according to the TIMER base frequency set by the prescaler.

In Counter mode, the TIMER's internal Counter register is incremented by one each time the COUNT task is triggered, meaning the timer frequency and the prescaler are not utilized in Counter mode. Similarly, the COUNT task has no effect in Timer mode.

TIMER's maximum value is configured by changing the bit-width of the timer in register [BITMODE](#) on page 626.

[PRESCALER](#) on page 626 and [BITMODE](#) on page 626 must only be updated when TIMER is stopped. If these registers are updated while TIMER is started, unpredictable behavior may occur.

When TIMER is incremented beyond its maximum value, the Counter register will overflow and TIMER will automatically start over from zero.

The Counter register can be cleared by triggering the CLEAR task. This will explicitly set the internal value to zero.

TIMER implements multiple capture/compare registers.

Independent of prescaler settings, the accuracy of TIMER is equivalent to one tick of the timer frequency f_{TIMER} as illustrated in [Block schematic for timer/counter](#) on page 618.

8.22.1 Capture

TIMER implements one capture task for every available capture/compare register.

Every time the CAPTURE[n] task is triggered, the counter value is copied to the CC[n] register.

8.22.2 Compare

TIMER implements one COMPARE event for every available capture/compare register.

When the counter value becomes equal to the value specified in a capture compare register CC[n], the corresponding compare event COMPARE[n] is generated.

[BITMODE](#) on page 626 specifies how many Counter and capture/compare register bits are used when the comparison is performed. Other bits are ignored.

The COMPARE event can be configured to operate in one-shot mode by configuring the corresponding ONESHOTEN[n] register. After writing CC[n], a COMPARE[n] event is generated the first time the Counter matches CC[n].

8.22.3 Task delays

After TIMER is started, the CLEAR, COUNT, and STOP tasks are guaranteed to take effect within one clock cycle of the PCLK.

8.22.4 Task priority

If the START task and the STOP task are triggered at the same time, meaning within the same period of PCLK, the STOP task is prioritized.

If one or more of the CAPTURE tasks and the CLEAR task are triggered at the same time, that is, within the same period of PCLK, the CAPTURE tasks are prioritized. This means that the CC registers will capture the counter value before the CLEAR tasks are triggered.

8.22.5 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
TIMER00 : S	GLOBAL	0x50055000	US	S	NA	No	Timer TIMER00
TIMER00 : NS		0x40055000					
TIMER10 : S	GLOBAL	0x50085000	US	S	NA	No	Timer TIMER10
TIMER10 : NS		0x40085000					
TIMER20 : S	GLOBAL	0x500CA000	US	S	NA	No	Timer TIMER20
TIMER20 : NS		0x400CA000					
TIMER21 : S	GLOBAL	0x500CB000	US	S	NA	No	Timer TIMER21
TIMER21 : NS		0x400CB000					
TIMER22 : S	GLOBAL	0x500CC000	US	S	NA	No	Timer TIMER22
TIMER22 : NS		0x400CC000					
TIMER23 : S	GLOBAL	0x500CD000	US	S	NA	No	Timer TIMER23
TIMER23 : NS		0x400CD000					
TIMER24 : S	GLOBAL	0x500CE000	US	S	NA	No	Timer TIMER24
TIMER24 : NS		0x400CE000					

Configuration

Instance	Domain	Configuration
TIMER00 : S TIMER00 : NS	GLOBAL	Peripheral clock frequency (PCLK) is 128 MHz
		The system is able to configure the TIMER peripheral input clock frequency (PCLK) before it reaches TIMER, and calculations of PRESCALER value must take the actual PCLK frequency into account
TIMER10 : S TIMER10 : NS	GLOBAL	6 capture compare channels implemented
		Peripheral clock frequency (PCLK) is 32 MHz
TIMER20 : S TIMER20 : NS	GLOBAL	8 capture compare channels implemented
		Peripheral clock frequency (PCLK) is 16 MHz
TIMER21 : S TIMER21 : NS	GLOBAL	6 capture compare channels implemented
		Peripheral clock frequency (PCLK) is 16 MHz
TIMER22 : S TIMER22 : NS	GLOBAL	6 capture compare channels implemented
		Peripheral clock frequency (PCLK) is 16 MHz
TIMER23 : S TIMER23 : NS	GLOBAL	6 capture compare channels implemented
		Peripheral clock frequency (PCLK) is 16 MHz
TIMER24 : S TIMER24 : NS	GLOBAL	6 capture compare channels implemented
		Peripheral clock frequency (PCLK) is 16 MHz

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Start Timer
TASKS_STOP	0x004		Stop Timer
TASKS_COUNT	0x008		Increment Timer (Counter mode only)
TASKS_CLEAR	0x00C		Clear time
TASKS_CAPTURE[n]	0x040		Capture Timer value to CC[n] register
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
SUBSCRIBE_COUNT	0x088		Subscribe configuration for task COUNT
SUBSCRIBE_CLEAR	0x08C		Subscribe configuration for task CLEAR
SUBSCRIBE_CAPTURE[n]	0x0C0		Subscribe configuration for task CAPTURE[n]
EVENTS_COMPARE[n]	0x140		Compare event on CC[n] match
PUBLISH_COMPARE[n]	0x1C0		Publish configuration for event COMPARE[n]
SHORTS	0x200		Shortcuts between local events and tasks
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
MODE	0x504		Timer mode selection
BITMODE	0x508		Configure the number of bits used by the TIMER
PRESCALER	0x510		Timer prescaler register
CC[n]	0x540		Capture/Compare register n
ONESHOTEN[n]	0x580		Enable one-shot operation for Capture/Compare channel n

8.22.5.1 TASKS_START

Address offset: 0x000

Start Timer

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	TASKS_START			Start Timer																															
			Trigger	1	Trigger task																															

8.22.5.2 TASKS_STOP

Address offset: 0x004

Stop Timer

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	TASKS_STOP			Stop Timer																															
			Trigger	1	Trigger task																															

8.22.5.3 TASKS_COUNT

Address offset: 0x008

Increment Timer (Counter mode only)

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																							A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value	Description																																		
A	W	TASKS_COUNT			Increment Timer (Counter mode only)																																		
			Trigger	1	Trigger task																																		

8.22.5.4 TASKS_CLEAR

Address offset: 0x00C

Clear time

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	TASKS_CLEAR			Clear time																															
			Trigger	1	Trigger task																															

8.22.5.5 TASKS_CAPTURE[n] (n=0..7)

Address offset: 0x040 + (n × 0x4)

Capture Timer value to CC[n] register

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	W	TASKS_CAPTURE			Capture Timer value to CC[n] register																															
			Trigger	1	Trigger task																															

8.22.5.6 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task **START**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task START will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.22.5.7 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task **STOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task STOP will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.22.5.8 SUBSCRIBE_COUNT

Address offset: 0x088

Subscribe configuration for task **COUNT**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CHIDX		[0..255]	DPPI channel that task COUNT will subscribe to																														
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.22.5.9 SUBSCRIBE_CLEAR

Address offset: 0x08C

Subscribe configuration for task **CLEAR**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task CLEAR will subscribe to																													
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.22.5.10 SUBSCRIBE_CAPTURE[n] (n=0..7)

Address offset: 0x0C0 + (n × 0x4)

Subscribe configuration for task **CAPTURE[n]**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task CAPTURE[n] will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A-H	RW	COMPARE[i] (i=0..7)						Enable or disable interrupt for event COMPARE[i]																											
			Disabled	0				Disable																											
			Enabled	1				Enable																											

8.22.5.15 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-H	RW	COMPARE[i] (i=0..7)			Write '1' to enable interrupt for event COMPARE[i]																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.22.5.16 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-H	RW	COMPARE[i] (i=0..7)			Write '1' to disable interrupt for event COMPARE[i]																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.22.5.17 MODE

Address offset: 0x504

Timer mode selection

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	MODE			Timer mode																														
			Timer	0	Select Timer mode																														
			Counter	1	Select Counter mode																														
					This enumerator is deprecated.																														
			LowPowerCounter	2	Select Low Power Counter mode																														

8.22.5.18 BITMODE

Address offset: 0x508

Configure the number of bits used by the TIMER

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	BITMODE			Timer bit width																														
			16Bit	0	16 bit timer bit width																														
			08Bit	1	8 bit timer bit width																														
			24Bit	2	24 bit timer bit width																														
			32Bit	3	32 bit timer bit width																														

8.22.5.19 PRESCALER

Address offset: 0x510

Timer prescaler register

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000004				0 1 0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PRESCALER		[0..9]				Prescaler value																											

8.22.5.20 CC[n] (n=0..7)

Address offset: 0x540 + (n × 0x4)

Capture/Compare register n

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value																Description															
A	RW	CC																		Capture/Compare value															

Only the number of bits indicated by BITMODE will be used by the TIMER.

8.22.5.21 ONESHOTEN[n] (n=0..7)

Address offset: 0x580 + (n × 0x4)

Enable one-shot operation for Capture/Compare channel n

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ONESHOTEN			Enable one-shot operation																														
					Configures the corresponding compare-channel for one-shot operation																														
			Disable	0	Disable one-shot operation																														
					Compare event is generated every time the Counter matches CC[n]																														
			Enable	1	Enable one-shot operation																														
					Compare event is generated the first time the Counter matches CC[n] after CC[n] has been written																														

8.23 TWIM — I²C compatible two-wire interface master with EasyDMA

TWI master with EasyDMA (TWIM) provides a half duplex, two wire synchronous serial communication interface which supports multiple slaves in the same bus.

The main features of TWIM are:

- I²C compatible
- EasyDMA direct transfer to/from RAM
- Individual selection of I/O pins
- Support for clock stretching
- Transmissions can be suspended and resumed

The two-wire interface can communicate with a bi-directional wired-AND bus with two lines (SCL, SDA). The protocol makes it possible to interconnect up to 127 individually addressable devices. TWIM is not compatible with CBUS.

Individual selection of GPIO pins ensures flexibility in device pinout and efficient use of board space and signal routing.

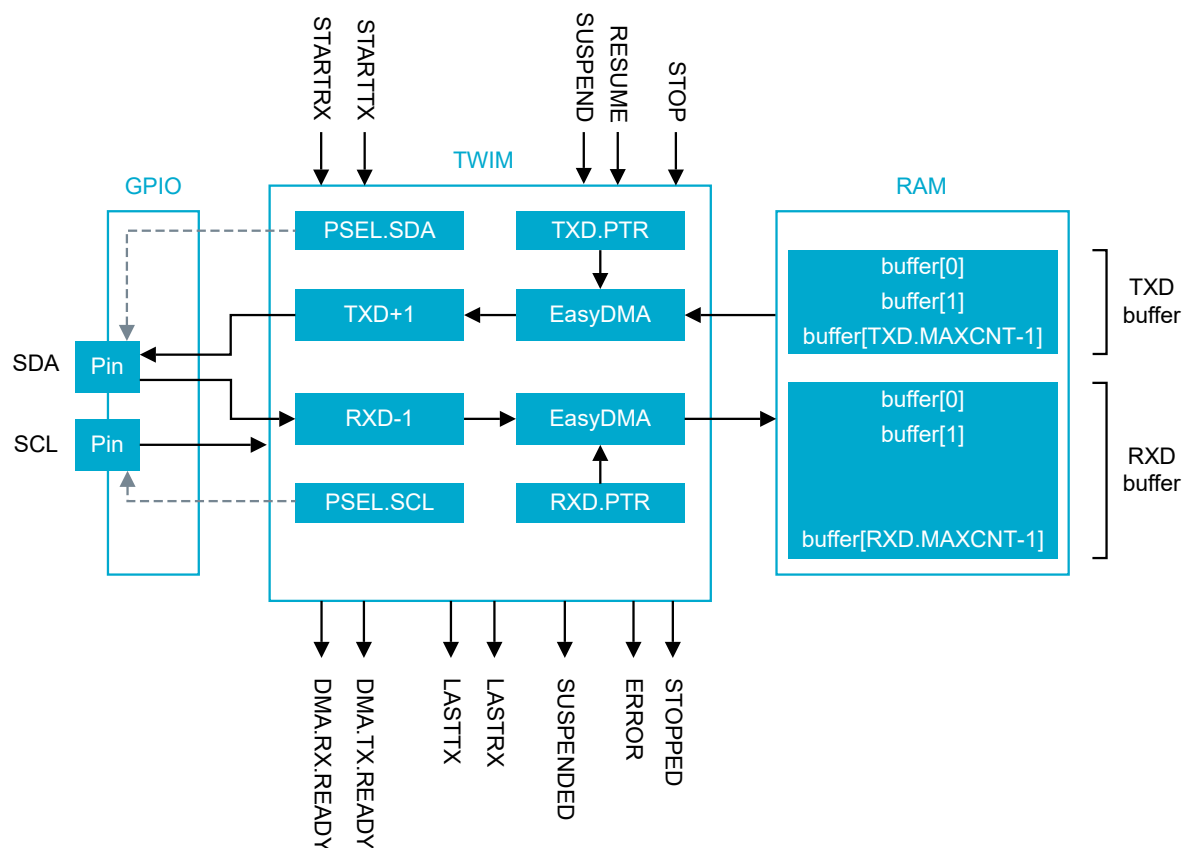


Figure 141: TWI master with EasyDMA

A typical TWI setup consists of one master and one or more slaves, as illustrated in the following figure. TWIM is only able to operate as a single master on the TWI bus. Multi-master bus configuration is not supported.

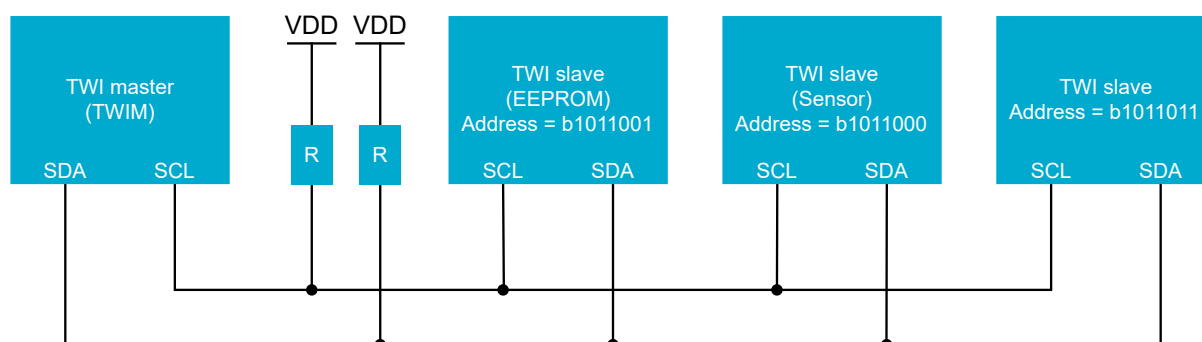


Figure 142: A typical TWI setup comprising one master and three slaves

TWIM supports clock stretching performed by the slaves. The SCK pulse following a stretched clock cycle may be shorter than specified by the I²C specification.

TWIM is started by triggering the STARTTX or STARTRX tasks, and stopped by triggering the STOP task. After a STOP task, TWIM generates a STOPPED event when it has stopped.

After TWIM has been started, the STARTTX or STARTRX tasks should not be triggered again until TWIM has issued a LASTRX, LASTTX, or STOPPED event.

TWIM can be suspended using the SUSPEND task, such as when using the TWI master in a low priority interrupt context. When TWIM enters suspend state, it will automatically issue a SUSPENDED event while performing a continuous clock stretching until it is instructed to resume operation via a RESUME task.

TWIM cannot be stopped while it is suspended, thus the STOP task has to be issued after the TWI master has been resumed.

Note: Any ongoing byte transfer will be allowed to complete before the suspend is enforced. A SUSPEND task has no effect unless TWIM is actively involved in a transfer.

If a NACK is clocked in from the slave, TWIM generates an ERROR event.

8.23.1 Shared resources

TWIM shares registers and other resources with other peripherals that have the same ID as the TWI master. Therefore, you must disable all peripherals that have the same ID as TWIM before it can be configured and used.

Disabling a peripheral that has the same ID as TWIM will not reset any of the registers that are shared with TWIM. It is therefore important to configure all relevant registers explicitly to secure that TWIM operates correctly.

The Instantiation table in [Instantiation](#) on page 216 shows which peripherals have the same ID as the TWI.

8.23.2 EasyDMA

TWIM implements EasyDMA for accessing RAM without CPU involvement.

TWIM implements the EasyDMA channels found in the following table.

Channel	Type	Register Cluster
TXD	READER	TXD
RXD	WRITER	RXD

Table 59: TWIM EasyDMA Channels

For detailed information regarding the use of EasyDMA, see [EasyDMA](#) on page 34.

The RXD.PTR, TXD.PTR, RXD.MAXCNT, and TXD.MAXCNT registers are double-buffered. They can be updated and prepared for the next RX or TX transmission immediately after having received the EVENTS_DMA.RX.READY or EVENTS_DMA.TX.READY event.

The STOPPED event indicates that EasyDMA has finished accessing the buffer in RAM.

8.23.3 Master write sequence

A TWIM write sequence is started by triggering the STARTTX task. After the STARTTX task has been triggered, TWIM generates a start condition on the TWI bus, followed by clocking out the address and the READ/WRITE bit set to 0 (WRITE=0, READ=1).

The address must match the address of the slave device that the master wants to write to. The READ/WRITE bit is followed by an ACK/NACK bit (ACK=0 or NACK=1) generated by the slave.

After receiving the ACK bit, TWIM clocks out the data bytes found in the transmit buffer located in RAM at the address specified in the TXD.PTR register. Each byte clocked out from TWIM will be followed by an ACK/NACK bit clocked in from the slave.

A typical TWIM write sequence is shown in the following figure, including clock stretching performed by TWIM following a SUSPEND task.

A SUSPENDED event indicates that the SUSPEND task has taken effect. This event can be used to synchronize the software.

TWIM will generate a LASTTX event when it starts to transmit the last byte.

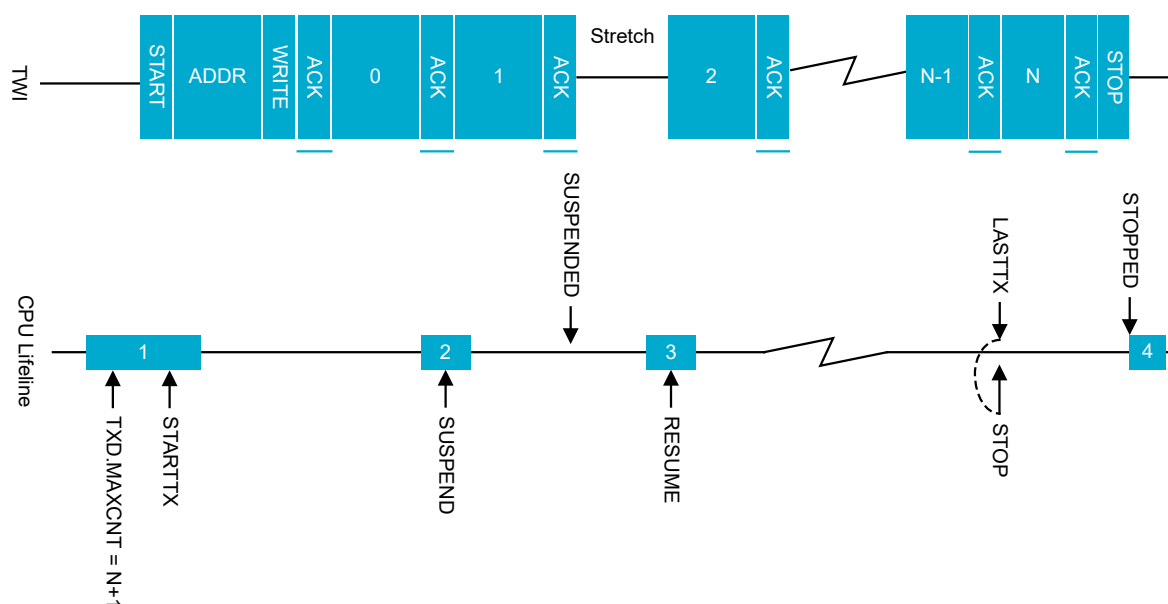


Figure 143: TWIM writing data to a slave

TWIM is stopped by triggering the STOP task. This task should be triggered during the transmission of the last byte to secure that TWIM will stop as fast as possible after sending the last byte. The shortcut between LASTTX and STOP can alternatively be used to accomplish this.

Note: TWIM does not stop by itself when the entire RAM buffer has been sent, or when an error occurs. The STOP task must be issued, through the use of a local or PPI shortcut, or in software as part of the error handler.

8.23.4 Master read sequence

A TWIM read sequence is started by triggering the STARTRX task. After the STARTRX task has been triggered, TWIM generates a start condition on the TWI bus, followed by clocking out the address and the READ/WRITE bit set to 1 (WRITE = 0, READ = 1). The address must match the address of the slave device that the master wants to read from. The READ/WRITE bit is followed by an ACK/NACK bit (ACK=0 or NACK = 1) generated by the slave.

After sending the ACK bit, the TWI slave sends data to the master using the clock generated by TWIM.

Data received will be stored in RAM at the address specified in the RXD.PTR register. The TWI master will generate an ACK after all but the last byte have been received from the slave. TWIM generates a NACK after the last byte received to indicate that the read sequence shall stop.

A typical TWIM read sequence is illustrated in the following figure, including clock stretching performed by TWIM following a SUSPEND task.

A SUSPENDED event indicates that the SUSPEND task has taken effect. This event can be used to synchronize the software.

TWIM generates a LASTRX event when it is ready to receive the last byte. If RXD.MAXCNT > 1, the LASTRX event is generated after sending the ACK of the previously received byte. If RXD.MAXCNT = 1, the LASTRX event is generated after receiving the ACK following the address and READ bit.

TWIM is stopped by triggering the STOP task. This task must be triggered before the NACK bit is supposed to be transmitted. The STOP task can be triggered at any time during the reception of the last byte. It is recommended to use the shortcut between LASTRX and STOP to accomplish this.

TWIM does not stop by itself when the RAM buffer is full, or when an error occurs. The STOP task must be issued, through the use of a local or PPI shortcut, or in software as part of the error handler.

TWIM cannot be stopped while suspended, so the STOP task must be issued after TWIM has been resumed.

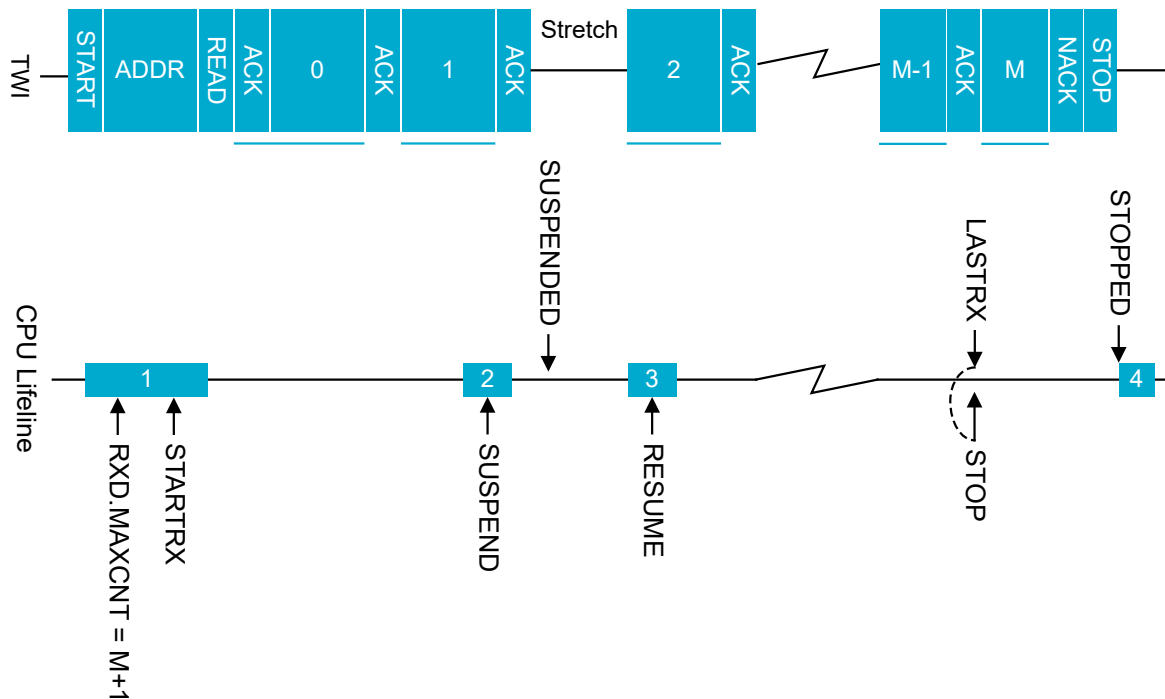


Figure 144: TWIM reading data from a slave

8.23.5 Master repeated start sequence

A typical repeated start sequence is when TWIM writes two bytes to the slave followed by reading four bytes from the slave. This example uses shortcuts to perform the simplest type of repeated start sequence, i.e. one write followed by one read. The same approach can be used to perform a repeated start sequence where the sequence is read followed by write.

The following figure shows an example of a repeated start sequence where TWIM writes two bytes followed by reading four bytes from the slave.

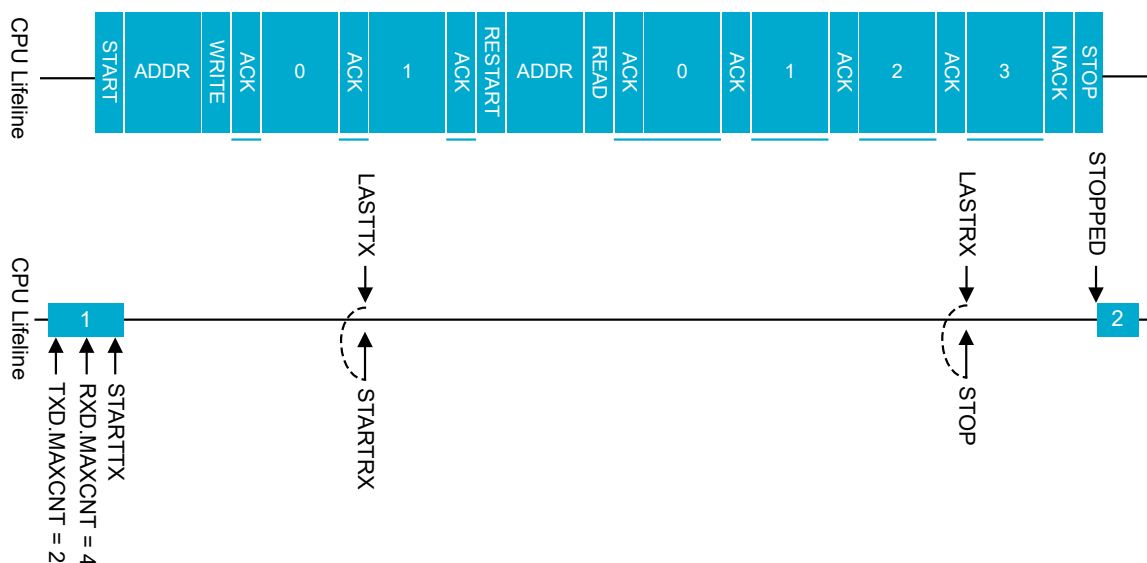


Figure 145: Master repeated start sequence

If a more complex repeated start sequence is needed, and the TWI firmware drive is serviced in a low priority interrupt, it may be necessary to use the SUSPEND task and SUSPENDED event to guarantee that the correct tasks are generated at the correct time. A double repeated start sequence using the SUSPEND task to secure safe operation in low priority interrupts is shown in the following figure.

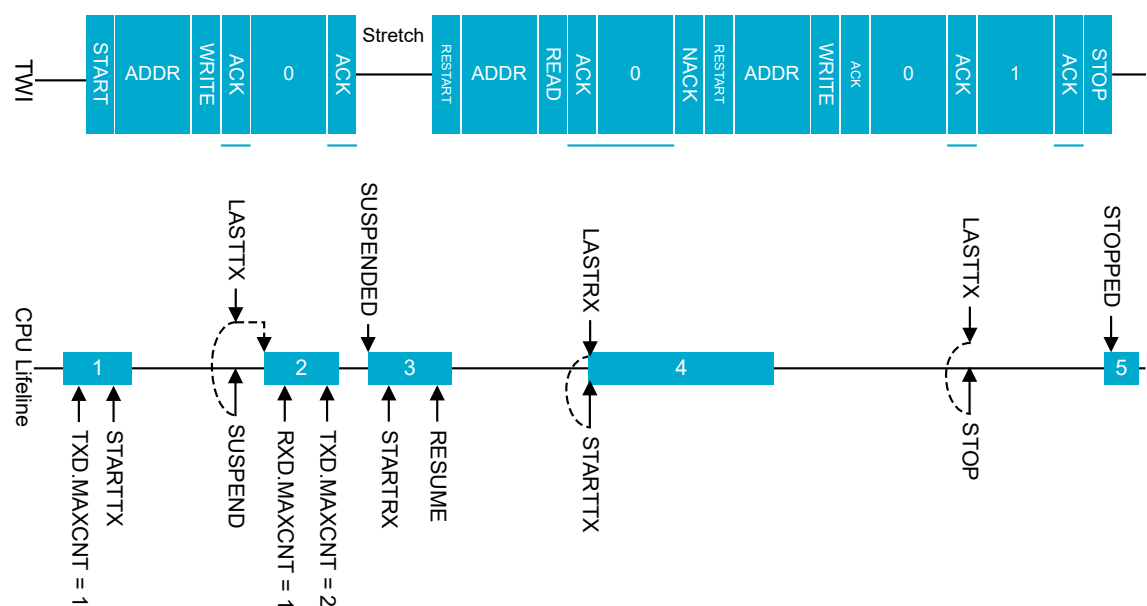


Figure 146: Double repeated start sequence

8.23.6 Low power

To ensure lowest possible power consumption when the peripheral is not needed stop and disable TWIM.

When the STOP task is sent, the software shall wait until the STOPPED event is received as a response before disabling the peripheral through the ENABLE register. If the peripheral is already stopped, the STOP task is not required.

8.23.7 Master mode pin configuration

The SCL and SDA signals are mapped to physical pins using the PSEL.SCL and PSEL.SDA registers.

The PSEL.SCL and PSEL.SDA registers and their configurations are only used as long as the TWI master is enabled, and retained only as long as the device is in System ON mode. When the peripheral is disabled, the pins will behave as regular GPIOs, and use the configuration in their respective OUT bit field and PIN_CNF[n] register. PSEL.SCL and PSEL.SDA must only be configured when the TWI master is disabled.

To secure correct signal levels on the pins used by TWIM while in System OFF mode, and when TWIM is disabled, these pins must be configured in the GPIO peripheral as described in the following table.

Only one peripheral can be assigned to drive a particular GPIO pin at a time. Failing to do so may result in unpredictable behavior.

TWI master signal	TWI master pin	Direction	Output value	Drive strength
SCL	As specified in PSEL.SCL	Input	Not applicable	SOD1
SDA	As specified in PSEL.SDA	Input	Not applicable	SOD1

Table 60: GPIO configuration before enabling peripheral

8.23.8 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
TWIM20 : S	GLOBAL	0x500C6000	US	S	SA	No	Two-wire interface controller TWIM20
TWIM20 : NS		0x400C6000					
TWIM21 : S	GLOBAL	0x500C7000	US	S	SA	No	Two-wire interface controller TWIM21
TWIM21 : NS		0x400C7000					
TWIM22 : S	GLOBAL	0x500C8000	US	S	SA	No	Two-wire interface controller TWIM22
TWIM22 : NS		0x400C8000					
TWIM30 : S	GLOBAL	0x50104000	US	S	SA	No	Two-wire interface controller TWIM30
TWIM30 : NS		0x40104000					

Configuration

Instance	Domain	Configuration
TWIM20 : S	GLOBAL	Use GPIO port P1, or dedicated pins on P2
TWIM20 : NS		CURRENTAMOUNT register not included.
TWIM21 : S	GLOBAL	Use GPIO port P1, or dedicated pins on P2
TWIM21 : NS		CURRENTAMOUNT register not included.
TWIM22 : S	GLOBAL	Use GPIO port P1, or dedicated pins on P2
TWIM22 : NS		CURRENTAMOUNT register not included.
TWIM30 : S	GLOBAL	Use GPIO port P0
TWIM30 : NS		CURRENTAMOUNT register not included.

Register overview

Register	Offset	TZ	Description
TASKS_STOP	0x004		Stop TWI transaction. Must be issued while the TWI master is not suspended.
TASKS_SUSPEND	0x00C		Suspend TWI transaction
TASKS_RESUME	0x010		Resume TWI transaction

Register	Offset	TZ	Description
TASKS_DMA.RX.START	0x028		Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.
TASKS_DMA.RX.STOP	0x02C		Stops operation using easyDMA. This does not trigger an END event.
TASKS_DMA.RX.ENABLEMATCH[n]	0x030		Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.
TASKS_DMA.RX.DISABLEMATCH[n]	0x040		Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.
TASKS_DMA.TX.START	0x050		Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.
TASKS_DMA.TX.STOP	0x054		Stops operation using easyDMA. This does not trigger an END event.
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
SUBSCRIBE_SUSPEND	0x08C		Subscribe configuration for task SUSPEND
SUBSCRIBE_RESUME	0x090		Subscribe configuration for task RESUME
SUBSCRIBE_DMA.RX.START	0x0A8		Subscribe configuration for task START
SUBSCRIBE_DMA.RX.STOP	0x0AC		Subscribe configuration for task STOP
SUBSCRIBE_DMA.RX.ENABLEMATCH[n]	0x0B0		Subscribe configuration for task ENABLEMATCH[n]
SUBSCRIBE_DMA.RX.DISABLEMATCH[n]	0x0C0		Subscribe configuration for task DISABLEMATCH[n]
SUBSCRIBE_DMA.TX.START	0x0D0		Subscribe configuration for task START
SUBSCRIBE_DMA.TX.STOP	0x0D4		Subscribe configuration for task STOP
EVENTS_STOPPED	0x104		TWI stopped
EVENTS_ERROR	0x114		TWI error
EVENTS_SUSPENDED	0x128		SUSPEND task has been issued, TWI traffic is now suspended.
EVENTS_LASTRX	0x134		Byte boundary, starting to receive the last byte
EVENTS_LASTTX	0x138		Byte boundary, starting to transmit the last byte
EVENTS_DMA.RX.END	0x14C		Generated after all MAXCNT bytes have been transferred
EVENTS_DMA.RX.READY	0x150		Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.
EVENTS_DMA.RX.BUSERROR	0x154		An error occurred during the bus transfer.
EVENTS_DMA.RX.MATCH[n]	0x158		Pattern match is detected on the DMA data bus.
EVENTS_DMA.TX.END	0x168		Generated after all MAXCNT bytes have been transferred
EVENTS_DMA.TX.READY	0x16C		Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.
EVENTS_DMA.TX.BUSERROR	0x170		An error occurred during the bus transfer.
PUBLISH_STOPPED	0x184		Publish configuration for event STOPPED
PUBLISH_ERROR	0x194		Publish configuration for event ERROR
PUBLISH_SUSPENDED	0x1A8		Publish configuration for event SUSPENDED
PUBLISH_LASTRX	0x1B4		Publish configuration for event LASTRX
PUBLISH_LASTTX	0x1B8		Publish configuration for event LASTTX
PUBLISH_DMA.RX.END	0x1CC		Publish configuration for event END
PUBLISH_DMA.RX.READY	0x1D0		Publish configuration for event READY
PUBLISH_DMA.RX.BUSERROR	0x1D4		Publish configuration for event BUSERROR
PUBLISH_DMA.RX.MATCH[n]	0x1D8		Publish configuration for event MATCH[n]
PUBLISH_DMA.TX.END	0x1E8		Publish configuration for event END
PUBLISH_DMA.TX.READY	0x1EC		Publish configuration for event READY
PUBLISH_DMA.TX.BUSERROR	0x1F0		Publish configuration for event BUSERROR
SHORTS	0x200		Shortcuts between local events and tasks
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
ERRORSRC	0x4C4		Error source
ENABLE	0x500		Enable TWIM
FREQUENCY	0x524		TWI frequency. Accuracy depends on the HFCLK source selected.
ADDRESS	0x588		Address used in the TWI transfer
PSEL.SCL	0x600		Pin select for SCL signal
PSEL.SDA	0x604		Pin select for SDA signal

Register	Offset	TZ	Description
DMA.RX.PTR	0x704		RAM buffer start address
DMA.RX.MAXCNT	0x708		Maximum number of bytes in channel buffer
DMA.RX.AMOUNT	0x70C		Number of bytes transferred in the last transaction, updated after the END event. Also updated after each MATCH event.
DMA.RX.TERMINATEONBUSERROR	0x71C		Terminate the transaction if a BUSERROR event is detected.
DMA.RX.BUSERRORADDRESS	0x720		Address of transaction that generated the last BUSERROR event.
DMA.RX.MATCH.CONFIG	0x724		Configure individual match events
DMA.RX.MATCH.CANDIDATE[n]	0x728		The data to look for - any match will trigger the MATCH[n] event, if enabled.
DMA.TX.PTR	0x73C		RAM buffer start address
DMA.TX.MAXCNT	0x740		Maximum number of bytes in channel buffer
DMA.TX.AMOUNT	0x744		Number of bytes transferred in the last transaction, updated after the END event. Also updated after each MATCH event.
DMA.TX.TERMINATEONBUSERROR	0x754		Terminate the transaction if a BUSERROR event is detected.
DMA.TX.BUSERRORADDRESS	0x758		Address of transaction that generated the last BUSERROR event.

8.23.8.1 TASKS_STOP

Address offset: 0x004

Stop TWI transaction. Must be issued while the TWI master is not suspended.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																							A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value	ID	Value	Description																																	
A	W	TASKS_STOP				Stop TWI transaction. Must be issued while the TWI master is not suspended.																																	
		Trigger		1		Trigger task																																	

8.23.8.2 TASKS_SUSPEND

Address offset: 0x00C

Suspend TWI transaction

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																								A			
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID		Value					Description																																	
A	W	TASKS_SUSPEND							Suspend TWI transaction																																		
				Trigger	1					Trigger task																																	

8.23.8.3 TASKS_RESUME

Address offset: 0x010

Resume TWI transaction

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																A											
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																						
A	W	TASKS_RESUME			Resume TWI transaction																																						
			Trigger	1	Trigger task																																						

8.23.8.4 TASKS_DMA

Peripheral tasks.

8.23.8.4.1 TASKS_DMA.RX

Peripheral tasks.

8.23.8.4.1.1 TASKS_DMA.RX.START

Address offset: 0x028

Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	START						Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.																											
			Trigger	1				Trigger task																											

8.23.8.4.1.2 TASKS_DMA.RX.STOP

Address offset: 0x02C

Stops operation using easyDMA. This does not trigger an END event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	STOP						Stops operation using easyDMA. This does not trigger an END event.																											
			Trigger	1				Trigger task																											

8.23.8.4.1.3 TASKS_DMA.RX.ENABLEMATCH[n] (n=0..3)

Address offset: 0x030 + (n × 0x4)

Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				A																																
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																												
A	W	ENABLEMATCH						Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.																												
			Trigger	1				Trigger task																												

8.23.8.4.1.4 TASKS_DMA.RX.DISABLEMATCH[n] (n=0..3)

Address offset: 0x040 + (n × 0x4)

Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.

Bit number										31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																			
ID										A																			
Reset 0x00000000										0 0																			
ID	R/W	Field	Value ID	Value		Description																							
A	W	DISABLEMATCH				Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.																							
			Trigger	1		Trigger task																							

8.23.8.4.2 TASKS_DMA.TX

Peripheral tasks.

8.23.8.4.2.1 TASKS_DMA.TX.START

Address offset: 0x050

Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				A																																		
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value				Description																														
A	W	START						Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.																														
			Trigger	1				Trigger task																														

8.23.8.4.2.2 TASKS_DMA.TX.STOP

Address offset: 0x054

Stops operation using easyDMA. This does not trigger an END event.

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID										A																																	
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field		Value ID	Value					Description																																	
A	W	STOP								Stops operation using easyDMA. This does not trigger an END event.																																	
			Trigger		1						Trigger task																																

8.23.8.5 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task STOP

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task STOP will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.23.8.6 SUBSCRIBE_SUSPEND

Address offset: 0x08C

Subscribe configuration for task **SUSPEND**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task SUSPEND will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.23.8.7 SUBSCRIBE_RESUME

Address offset: 0x090

Subscribe configuration for task **RESUME**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task RESUME will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.23.8.8 SUBSCRIBE_DMA

Subscribe configuration for tasks

8.23.8.8.1 SUBSCRIBE_DMA.RX

Subscribe configuration for tasks

8.23.8.8.1.1 SUBSCRIBE_DMA.RX.START

Address offset: 0x0A8

Subscribe configuration for task **START**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task START will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.23.8.8.1.2 SUBSCRIBE_DMA.RX.STOP

Address offset: 0x0AC

Subscribe configuration for task **STOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task STOP will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.23.8.8.1.3 SUBSCRIBE_DMA.RX.ENABLEMATCH[n] (n=0..3)

Address offset: 0x0B0 + (n × 0x4)

Subscribe configuration for task **ENABLEMATCH[n]**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CHIDX		[0..255]	DPPI channel that task ENABLEMATCH[n] will subscribe to																														
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.23.8.8.1.4 SUBSCRIBE_DMA.RX.DISABLEMATCH[n] (n=0..3)

Address offset: 0x0C0 + (n × 0x4)

Subscribe configuration for task **DISABLEMATCH[n]**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task DISABLEMATCH[n] will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.23.8.8.2 SUBSCRIBE_DMA.TX

Subscribe configuration for tasks

8.23.8.8.2.1 SUBSCRIBE_DMA.TX.START

Address offset: 0x0D0

Subscribe configuration for task [START](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				B																													A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
ID	R/W	Field	Value ID	Value		Description																																	
A	RW	CHIDX		[0..255]		DPPI channel that task START will subscribe to																																	
B	RW	EN																																					
			Disabled	0	Disable subscription																																		
			Enabled	1	Enable subscription																																		

8.23.8.8.2.2 SUBSCRIBE_DMA.TX.STOP

Address offset: 0x0D4

Subscribe configuration for task [STOP](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A					A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that task STOP will subscribe to																																
B	RW	EN																																				
			Disabled	0	Disable subscription																																	
			Enabled	1	Enable subscription																																	

8.23.8.9 EVENTS_STOPPED

Address offset: 0x104

TWI stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_STOPPED						TWI stopped																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.23.8.10 EVENTS_ERROR

Address offset: 0x114

TWI error

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_ERROR			TWI error																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.23.8.11 EVENTS_SUSPENDED

Address offset: 0x128

SUSPEND task has been issued, TWI traffic is now suspended.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_SUSPENDED			SUSPEND task has been issued, TWI traffic is now suspended.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.23.8.12 EVENTS_LASTRX

Address offset: 0x134

Byte boundary, starting to receive the last byte

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID					A																																
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																
A	RW	EVENTS_LASTRX			Byte boundary, starting to receive the last byte																																
			NotGenerated	0	Event not generated																																
			Generated	1	Event generated																																

8.23.8.13 EVENTS_LASTTX

Address offset: 0x138

Byte boundary, starting to transmit the last byte

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_LASTTX			Byte boundary, starting to transmit the last byte																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.23.8.14 EVENTS_DMA

Peripheral events.

8.23.8.14.1 EVENTS_DMA.RX

Peripheral events.

8.23.8.14.1.1 EVENTS_DMA.RX.END

Address offset: 0x14C

Generated after all MAXCNT bytes have been transferred

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID					A																																
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value	Description																																
A	RW	END			Generated after all MAXCNT bytes have been transferred																																
			NotGenerated	0	Event not generated																																
			Generated	1	Event generated																																

8.23.8.14.1.2 EVENTS_DMA.RX.READY

Address offset: 0x150

Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.23.8.14.1.3 EVENTS_DMA.RX.BUSERROR

Address offset: 0x154

An error occurred during the bus transfer.

When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	BUSERROR						An error occurred during the bus transfer.																											
								When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.23.8.14.1.4 EVENTS_DMA.RX.MATCH[n] (n=0..3)

Address offset: 0x158 + (n × 0x4)

Pattern match is detected on the DMA data bus.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	MATCH			Pattern match is detected on the DMA data bus.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.23.8.14.2 EVENTS_DMA.TX

Peripheral events.

8.23.8.14.2.1 EVENTS_DMA.TX.END

Address offset: 0x168

Generated after all MAXCNT bytes have been transferred

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	END			Generated after all MAXCNT bytes have been transferred																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.23.8.14.2.2 EVENTS_DMA.TX.READY

Address offset: 0x16C

Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.23.8.14.2.3 EVENTS_DMA.TX.BUSERROR

Address offset: 0x170

An error occurred during the bus transfer.

When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	BUSERROR			An error occurred during the bus transfer.																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.23.8.15 PUBLISH_STOPPED

Address offset: 0x184

Publish configuration for event STOPPED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event STOPPED will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.23.8.16 PUBLISH_ERROR

Address offset: 0x194

Publish configuration for event ERROR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event ERROR will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.23.8.17 PUBLISH_SUSPENDED

Address offset: 0x1A8

Publish configuration for event SUSPENDED

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event SUSPENDED will publish to																													
B	RW	EN																																	
			Disabled	0		Disable publishing																													
			Enabled	1		Enable publishing																													

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CHIDX		[0..255]	DPPI channel that event READY will publish to																														
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.23.8.20.1.3 PUBLISH_DMA.RX.BUSERERROR

Address offset: 0x1D4

Publish configuration for event **BUSERERROR**

When this event is generated, the address which caused the error can be read from the BUSERERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event BUSERERROR will publish to																													
B	RW	EN																																	
			Disabled	0		Disable publishing																													
			Enabled	1		Enable publishing																													

8.23.8.20.1.4 PUBLISH_DMA.RX.MATCH[n] (n=0..3)

Address offset: 0x1D8 + (n × 0x4)

Publish configuration for event **MATCH[n]**

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					B																								A A A A A A A A							
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	CHIDX		[0..255]	DPPI channel that event MATCH[n] will publish to																															
B	RW	EN																																		
			Disabled	0	Disable publishing																															
			Enabled	1	Enable publishing																															

8.23.8.20.2 PUBLISH_DMA.TX

Publish configuration for events

8.23.8.20.2.1 PUBLISH_DMA.TX.END

Address offset: 0x1E8

Publish configuration for event **END**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event END will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.23.8.20.2 PUBLISH_DMA.TX.READY

Address offset: 0x1EC

Publish configuration for event **READY**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				B																								A				A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value				Description																															
A	RW	CHIDX		[0..255]				DPPI channel that event READY will publish to																															
B	RW	EN																																					
			Disabled	0				Disable publishing																															
			Enabled	1				Enable publishing																															

8.23.8.20.3 PUBLISH_DMA.TX.BUSERROR

Address offset: 0x1F0

Publish configuration for event **BUSERROR**

When this event is generated, the address which caused the error can be read from the **BUSERRORADDRESS** register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A	A																								
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event BUSERROR will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.23.8.21 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				M L K J I H G F																E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	LASTTX_DMA_RX_START				Shortcut between event LASTTX and task DMA.RX.START																													
			Disabled	0	Disable shortcut																														
			Enabled	1	Enable shortcut																														
B	RW	LASTTX_SUSPEND				Shortcut between event LASTTX and task SUSPEND																													

8.23.8.22 INTEN

Enable or disable interrupt

4503 018 v0.8

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				O N M L K J I H G F																E D C				B				A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
H	RW	DMARXBUSERROR			Enable or disable interrupt for event DMARXBUSERROR																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
I-L	RW	DMARXMATCH[i] (i=0..3)			Enable or disable interrupt for event DMARXMATCH[i]																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
M	RW	DMATXEND			Enable or disable interrupt for event DMATXEND																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
N	RW	DMATXREADY			Enable or disable interrupt for event DMATXREADY																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														
O	RW	DMATXBUSERROR			Enable or disable interrupt for event DMATXBUSERROR																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.23.8.23 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				O N M L K J I H G F																E D C				B				A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	STOPPED			Write '1' to enable interrupt for event STOPPED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	ERROR			Write '1' to enable interrupt for event ERROR																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	SUSPENDED			Write '1' to enable interrupt for event SUSPENDED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	LASTRX			Write '1' to enable interrupt for event LASTRX																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
E	RW	LASTTX			Write '1' to enable interrupt for event LASTTX																														
			Set	1	Enable																														

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				O N M L K J I H G F																E D				C				B				A					
Reset 0x00000000				0 0																																	
ID	R/W	Field	Value	ID	Value	Description																															
F	RW	DMARXEND	Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
			Set	1	Write '1' to enable interrupt for event DMARXEND																																
			Set	1	Enable																																
			Disabled	0	Read: Disabled																																
G	RW	DMARXREADY	Enabled	1	Read: Enabled																																
			Set	1	Write '1' to enable interrupt for event DMARXREADY																																
			Set	1	Enable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
H	RW	DMARXBUSEROR	Set	1	Write '1' to enable interrupt for event DMARXBUSEROR																																
			Set	1	When this event is generated, the address which caused the error can be read from the BUSERORADDRESS register.																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
			Set	1	Enable																																
I-L	RW	DMARXMATCH[i] (i=0..3)	Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
			Set	1	Write '1' to enable interrupt for event DMARXMATCH[i]																																
			Set	1	Enable																																
			Disabled	0	Read: Disabled																																
M	RW	DMATXEND	Enabled	1	Read: Enabled																																
			Set	1	Write '1' to enable interrupt for event DMATXEND																																
			Set	1	Enable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
N	RW	DMATXREADY	Set	1	Write '1' to enable interrupt for event DMATXREADY																																
			Set	1	Enable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
			Set	1	Enable																																
O	RW	DMATXBUSEROR	Set	1	Write '1' to enable interrupt for event DMATXBUSEROR																																
			Set	1	When this event is generated, the address which caused the error can be read from the BUSERORADDRESS register.																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
			Set	1	Enable																																

8.23.8.24 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				O N M L K J I H G F																E D				C				B				A			
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	STOPPED						Write '1' to disable interrupt for event STOPPED																											
			Clear	1				Disable																											
			Disabled	0				Read: Disabled																											
			Enabled	1				Read: Enabled																											
B	RW	ERROR						Write '1' to disable interrupt for event ERROR																											

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				O N M L K J I H G F																E D C				B				A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	SUSPENDED			Write '1' to disable interrupt for event SUSPENDED																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	LASTRX			Write '1' to disable interrupt for event LASTRX																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
E	RW	LASTTX			Write '1' to disable interrupt for event LASTTX																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
F	RW	DMARXEND			Write '1' to disable interrupt for event DMARXEND																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
G	RW	DMARXREADY			Write '1' to disable interrupt for event DMARXREADY																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
H	RW	DMARXBUSERROR			Write '1' to disable interrupt for event DMARXBUSERROR																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
I-L	RW	DMARXMATCH[i] (i=0..3)			Write '1' to disable interrupt for event DMARXMATCH[i]																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
M	RW	DMATXEND			Write '1' to disable interrupt for event DMATXEND																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
N	RW	DMATXREADY			Write '1' to disable interrupt for event DMATXREADY																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
O	RW	DMATXBUSERROR			Write '1' to disable interrupt for event DMATXBUSERROR																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.23.8.25 ERRORSRC

Address offset: 0x4C4

Error source

Bit number		31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																			
ID																																			C	B	A
Reset 0x00000000		0 0																																			
ID	R/W	Field	Value ID	Value	Description																																
A	RW	OVERRUN W1C			Overrun error																																
					A new byte was received before previous byte got transferred into RXD buffer. (Previous data is lost)																																
			NotReceived	0	Error did not occur																																
			Received	1	Error occurred																																
B	RW	ANACK W1C			NACK received after sending the address (write '1' to clear)																																
			NotReceived	0	Error did not occur																																
			Received	1	Error occurred																																
C	RW	DNACK W1C			NACK received after sending a data byte (write '1' to clear)																																
			NotReceived	0	Error did not occur																																
			Received	1	Error occurred																																

8.23.8.26 ENABLE

Address offset: 0x500

Enable TWIM

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE			Enable or disable TWIM																														
			Disabled	0	Disable TWIM																														
			Enabled	6	Enable TWIM																														

8.23.8.27 FREQUENCY

Address offset: 0x524

TWI frequency. Accuracy depends on the HFCLK source selected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

8.23.8.28 ADDRESS

Address offset: 0x588

Address used in the TWI transfer

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																													A					A	A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value	Description																																		
A	RW	ADDRESS			Address used in the TWI transfer																																		

8.23.8.29 PSEL.SCL

Address offset: 0x600

Pin select for SCL signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0							
ID				C																								B B B A A A A A														
Reset 0xFFFFFFFF				1 1																																						
ID	R/W	Field	Value ID	Value		Description																																				
A	RW	PIN		[0..31]		Pin number																																				
B	RW	PORT		[0..7]		Port number																																				
C	RW	CONNECT				Connection																																				
			Disconnected	1		Disconnect																																				
			Connected	0		Connect																																				

8.23.8.30 PSEL.SDA

Address offset: 0x604

Pin select for SDA signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				C																								B				B	B	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
ID	R/W	Field	Value ID	Value				Description																													
A	RW	PIN		[0..31]				Pin number																													
B	RW	PORT		[0..7]				Port number																													
C	RW	CONNECT						Connection																													
			Disconnected	1				Disconnect																													
			Connected	0				Connect																													

8.23.8.31 DMA.RX.PTR

Address offset: 0x704

RAM buffer start address

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x20000000				0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						RAM buffer start address for this EasyDMA channel. This address is a word aligned Data RAM address.																											

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.23.8.32 DMA.RX.MAXCNT

Address offset: 0x708

Maximum number of bytes in channel buffer

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	MAXCNT		[1..0xFFFF]				Maximum number of bytes in channel buffer																											

8.23.8.33 DMA.RX.AMOUNT

Address offset: 0x70C

Number of bytes transferred in the last transaction, updated after the END event.

Also updated after each MATCH event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value		Description																															
A	R	AMOUNT		[1..0xFFFF]		Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.																															

8.23.8.34 DMA.RX.TERMINATEONBUSERROR

Address offset: 0x71C

Terminate the transaction if a BUSERROR event is detected.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				A																																		
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value				Description																														
A	RW	ENABLE																																				
			Disabled	0				Disable																														
			Enabled	1				Enable																														

8.23.8.35 DMA.RX.BUSERRORADDRESS

Address offset: 0x720

Address of transaction that generated the last BUSERROR event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	ADDRESS																																	

8.23.8.36 DMA.RX.MATCH

Registers to control the behavior of the pattern matcher engine

8.23.8.36.1 DMA.RX.MATCH.CONFIG

Address offset: 0x724

Configure individual match events

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				H G F E																												D C B A			
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-D	RW	ENABLE[i] (i=0..3)			Enable match filter i																														
			Disabled	0	Match filter disabled																														
			Enabled	1	Match filter enabled																														
E-H	RW	ONESHOT[i] (i=0..3)			Configure match filter i as one-shot or sticky																														
					One-shot match filters can be used together with shortcuts to check for continuous data sequences by disabling the filter if the next data is not a match.																														
					Note: The presence of these shorts depends on the configuration of the peripheral integrating this EasyDMA.																														
			Continuous	0	Match filter stays enabled until disabled by task																														
			Oneshot	1	Match filter stays enabled until next data word is received																														

8.23.8.36.2 DMA.RX.MATCH.CANDIDATE[n] (n=0..3)

Address offset: 0x728 + (n × 0x4)

The data to look for - any match will trigger the MATCH[n] event, if enabled.

Note: This register can be updated while a transfer is in progress, but the new value will not take effect until a match has been found or the transfer is done. That makes it possible to write a new set of match words which will be searched for immediately after the event triggers.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW	DATA										Data to look for																							

8.23.8.37 DMA.TX.PTR

Address offset: 0x73C

RAM buffer start address

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x20000000				0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						RAM buffer start address for this EasyDMA channel. This address is a word aligned Data RAM address.																											

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.23.8.38 DMA.TX.MAXCNT

Address offset: 0x740

Maximum number of bytes in channel buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW	MAXCNT		[1..0xFFFF]								Maximum number of bytes in channel buffer																							

8.23.8.39 DMA.TX.AMOUNT

Address offset: 0x744

Number of bytes transferred in the last transaction, updated after the END event.

Also updated after each MATCH event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	AMOUNT		[1..0xFFFF]				Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.																											

8.23.8.40 DMA.TX.TERMINATEONBUSERROR

Address offset: 0x754

Terminate the transaction if a BUSERROR event is detected.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				A																																		
Reset 0x00000000				0 0																																		
ID	R/W	Field	Value ID	Value				Description																														
A	RW	ENABLE																																				
			Disabled	0				Disable																														
			Enabled	1				Enable																														

8.23.8.41 DMA.TX.BUSERRORADDRESS

Address offset: 0x758

Address of transaction that generated the last BUSERROR event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	ADDRESS																																	

8.23.9 Pullup resistor

Figure 147: Recommended TWIM pullup value vs. line capacitance

- The I²C specification allows a line capacitance of 400 pF at most.
- The value of internal pullup resistor (R_{PU}) for nRF54L15/10/05 can be found in [GPIO — General purpose input/output](#) on page 273.

8.24 TWIS — I²C compatible two-wire interface target with EasyDMA

The TWI target peripheral (TWIS) with EasyDMA provides a half duplex, two-wire synchronous serial communication interface.

The main features of TWIS are the following:

- I²C compatible
- Supports 100 kbps and 400 kbps bit rate
- EasyDMA direct transfer to and from RAM
- Individual selection of I/O pins
- Support for clock stretching

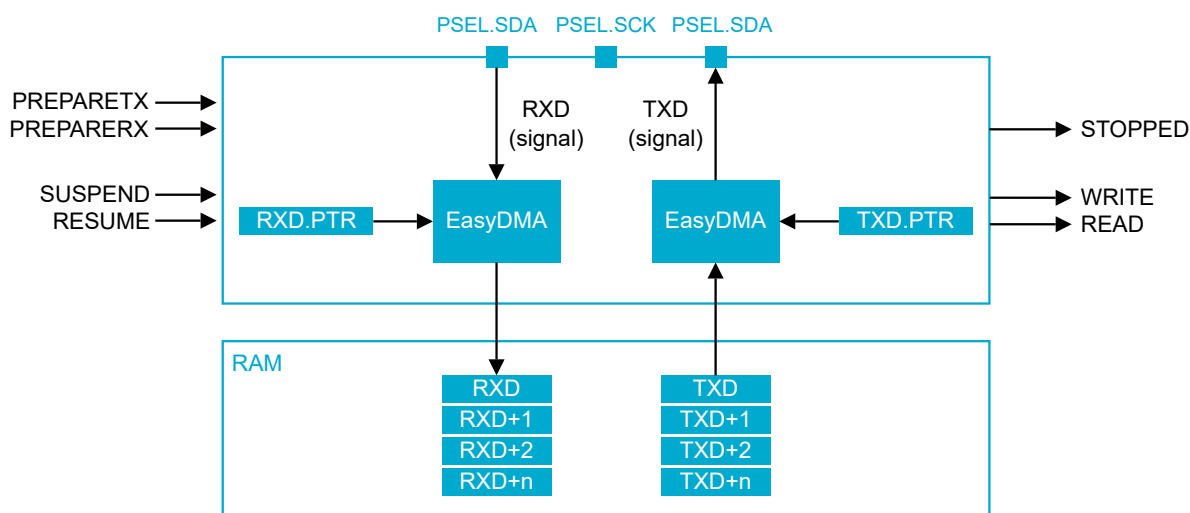


Figure 148: TWIS with EasyDMA

A typical TWI setup consists of one controller and one or more targets, as seen in the following figure. Only a single controller can be used on the TWI bus.

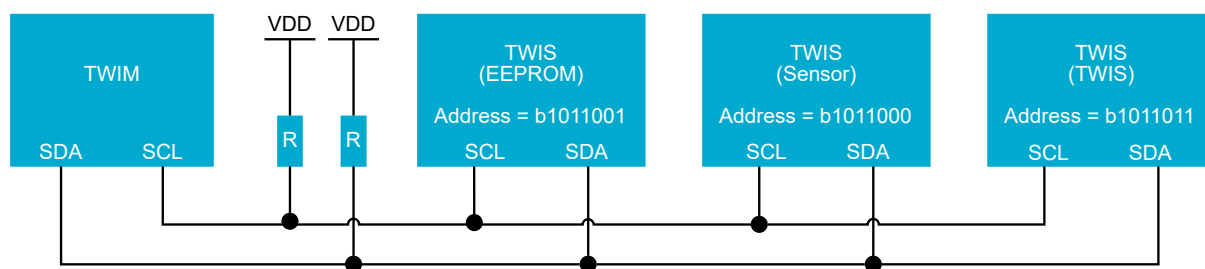


Figure 149: Typical TWI setup with one controller and three targets

8.24.1 State machine

The following figure shows the TWIS state machine.

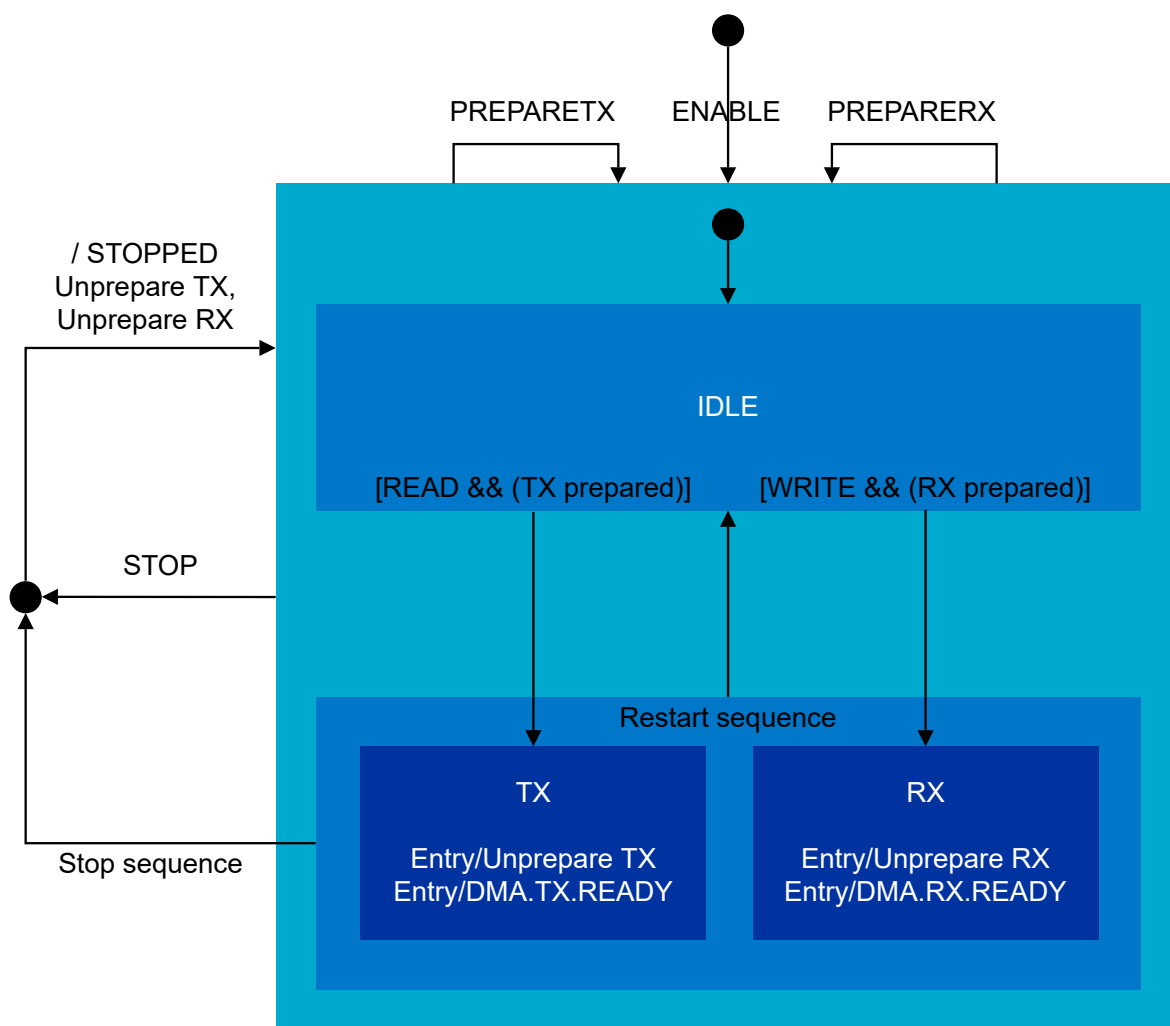


Figure 150: TWIS state machine

The following table contains descriptions of the symbols used in the state machine.

Symbol	Type	Description
ENABLE	Register	TWIS enabled via the ENABLE register.
PREPARETX	Task	The TASKS_PREPARETX task was triggered.
STOP	Task	The TASKS_STOP task was triggered.
PREPARERX	Task	The TASKS_PREPARERX task was triggered.
STOPPED	Event	The EVENTS_STOPPED event was generated.
DMA.RX.READY	Event	The EVENTS_DMA.RX.READY event was generated.
DMA.TX.READY	Event	The EVENTS_DMA.TX.READY event was generated.
TX prepared	Internal	Internal flag indicating that a TASKS_PREPARETX task was triggered.
RX prepared	Internal	Internal flag indicating that a TASKS_PREPARERX task was triggered.
Unprepare TX	Internal	Clears the TX prepared flag until the next TASKS_PREPARETX task.
Unprepare RX	Internal	Clears the RX prepared flag until the next TASKS_PREPARERX task.
Stop condition	TWI protocol	A TWI stop condition was detected.
Restart condition	TWI protocol	A TWI restart condition was detected.

Table 61: TWI slave state machine symbols

TWIS supports clock stretching. In order to use this feature, the controller must also support clock stretching for the feature to execute properly. TWIS operates in a low-power mode while waiting for the TWI controller to initiate a transfer. As long as TWIS is not addressed, it will remain in this mode.

For TWIS to run correctly, PSEL.SCL, PSEL.SDA, CONFIG, and the ADDRESS[n] registers must be configured, the SCL and SDA lines must both be high, before enabling TWIS through the ENABLE register. Similarly, changing these settings must be performed while TWIS is disabled. Failing to do so may result in unpredictable behavior.

8.24.2 Shared resources

The TWIS peripheral shares registers and other resources with peripherals that have the same ID as TWIS. Before TWIS can be configured and used, all peripherals that have the same ID as TWIS must be disabled.

Disabling a peripheral with the same ID as TWIS will not reset any shared TWIS registers. Configure all TWIS registers to ensure they operate correctly.

See the Instantiation table in [Instantiation](#) on page 216 for details on peripherals and their IDs.

8.24.3 EasyDMA

TWIS implements EasyDMA for accessing RAM without CPU involvement.

TWIS implements the EasyDMA channels found in the following table.

Channel	Type	Register Cluster
TXD	READER	TXD
RXD	WRITER	RXD

Table 62: TWIS EasyDMA Channels

For detailed information regarding the use of EasyDMA, see [EasyDMA](#) on page 34.

The STOPPED event indicates that EasyDMA is finished accessing the buffer in RAM.

8.24.4 Read command response

Before TWIS can respond to a read command, it must be configured and enabled in the ENABLE register. When enabled, TWIS is in the IDLE state.

A read command is started when TWIM generates a start condition on the TWI bus. This is followed by clocking out the address and setting the READ/WRITE bit to 1 (READ=1, WRITE=0). The READ/WRITE bit is followed by an ACK/NACK bit (ACK = 0, NACK = 1) response from the TWIS.

TWIS can listen for two addresses at a time. This is configured in the ADDRESS registers and the CONFIG register.

TWIS only acknowledges (ACK) the read command if the address presented by the controller matches one of the addresses the target is configured to listen for. TWIS will generate a READ event when it acknowledges the read command.

TWIS only detects a read command from the IDLE state.

TWIS will set an internal **TX prepared** flag when the PREPARETX task is triggered.

When the read command is received, TWIS will enter the TX state if the internal **TX prepared** flag is set.

If the internal **TX prepared** flag is not set when the read command is received, TWIS will stretch the controller's clock until the PREPARETX task is triggered and the internal **TX prepared** flag is set.

TWIS will generate the EVENTS_DMA.TX.READY event and clear the **TX prepared** flag when it enters the TX state. In this state, TWIS will send the data bytes found in the transmit buffer to the controller using the controller's clock.

TWIS returns to the IDLE state if the TWIS receives a restart command when it is in the TX state.

TWIS is stopped when it receives the stop condition from TWIM. A STOPPED event will be generated when the transaction has stopped. TWIS will clear the **TX prepared** flag and go back to the IDLE state when it has stopped.

The transmit buffer is located in RAM at the address specified in the TXD.PTR register. TWIS will only be able to send TXD.MAXCNT bytes from the transmit buffer for each transaction. If TWIM forces TWIS to send more than TXD.MAXCNT bytes, the target will send the byte specified in the ORC register to the controller instead. If this happens, an ERROR event will be generated.

The EasyDMA configuration registers RXD.PTR, TXD.PTR, RXD.AMOUNT, and TXD.AMOUNT, are latched when the EVENTS_DMA.TX.READY event is generated.

TWIS can be forced to stop by triggering the STOP task. A STOPPED event will be generated when TWIS has stopped. TWIS will clear the **TX prepared** flag and return to the IDLE state when it has stopped, see [Terminate an ongoing TWI transaction](#) on page 663.

Each byte sent from TWIS will be followed by an ACK/NACK bit sent from the controller. TWIM will generate a NACK following the last byte that it wants to receive to tell the target to release the bus allowing TWIM to generate the stop condition. The TXD.AMOUNT register can be queried after a transaction to see how many bytes were sent.

A typical TWIS read command response is illustrated in the following figure, including clock stretching following a SUSPEND task.

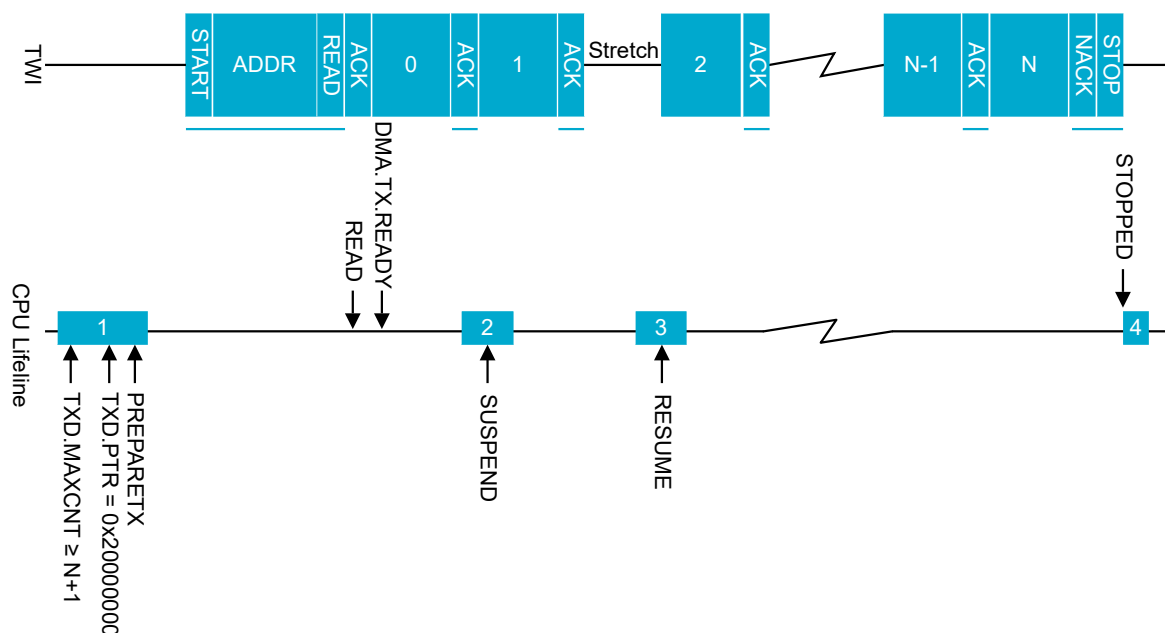


Figure 151: TWIS responding to a read command

8.24.5 Write command response

Before TWIS can respond to a write command, TWIS must be configured and enabled in the ENABLE register. When enabled, TWIS is in the IDLE state.

A write command is started when TWIM generates a start condition on the TWI bus. This is followed by clocking out the address and setting the READ/WRITE bit to 0 (READ = 1, WRITE = 0). The READ/WRITE bit is followed by an ACK/NACK bit (ACK = 0, NACK = 1) response from the TWIS.

TWIS can listen for two addresses at a time. This is configured in the ADDRESS registers and the CONFIG register.

TWIS only acknowledges (ACK) the write command if the address presented by the controller matches one of the addresses the target is configured to listen for. TWIS will generate a WRITE event when it acknowledges the write command.

TWIS only detects a write command from the IDLE state.

TWIS will set an internal **RX prepared** flag when the PREPARERX task is triggered.

When the write command is received, TWIS will enter the RX state if the internal **RX prepared** flag is set.

If the internal **RX prepared** flag is not set when the write command is received, TWIS will start stretching the master's clock after the first data byte, not allowing the master to send the stop condition. Clock is stretched until the PREPARERX task is triggered and the internal **RX prepared** flag is set.

TWIS will generate the EVENTS_DMA.RX.READY event and clear the internal **RX prepared** flag when it enters the RX state. In this state, TWIS will be able to receive the bytes sent by the TWIM.

TWIS returns to the IDLE state if TWIS receives a restart command when it is in the RX state.

TWIS is stopped when it receives the stop condition from TWIM. A STOPPED event will be generated when the transaction has stopped. TWIS will clear the internal **RX prepared** flag and go back to the IDLE state when it has stopped.

The receive buffer is located in RAM at the address specified in the RXD.PTR register. TWIS can only receive as many bytes as specified in the RXD.MAXCNT register. If TWIM tries to send more bytes to TWIS than it

can receive, the extra bytes are discarded and NACKed by the target. If this happens, an ERROR event will be generated.

The EasyDMA configuration registers, RXD.PTR, TXD.PTR, RXD.AMOUNT, and TXD.AMOUNT, are latched when the EVENTS_DMA.RX.READY event is generated.

TWIS can be forced to stop by triggering the STOP task. A STOPPED event will be generated when TWIS has stopped. TWIS will clear the internal **RX prepared** flag and return to the IDLE state when it has stopped, see [Terminate an ongoing TWI transaction](#) on page 663.

TWIS will generate an ACK after every byte received from the controller. The RXD.AMOUNT register can be queried after a transaction to see how many bytes were received.

A typical TWIS write command response is illustrated in the following figure, including clock stretching following a SUSPEND task.

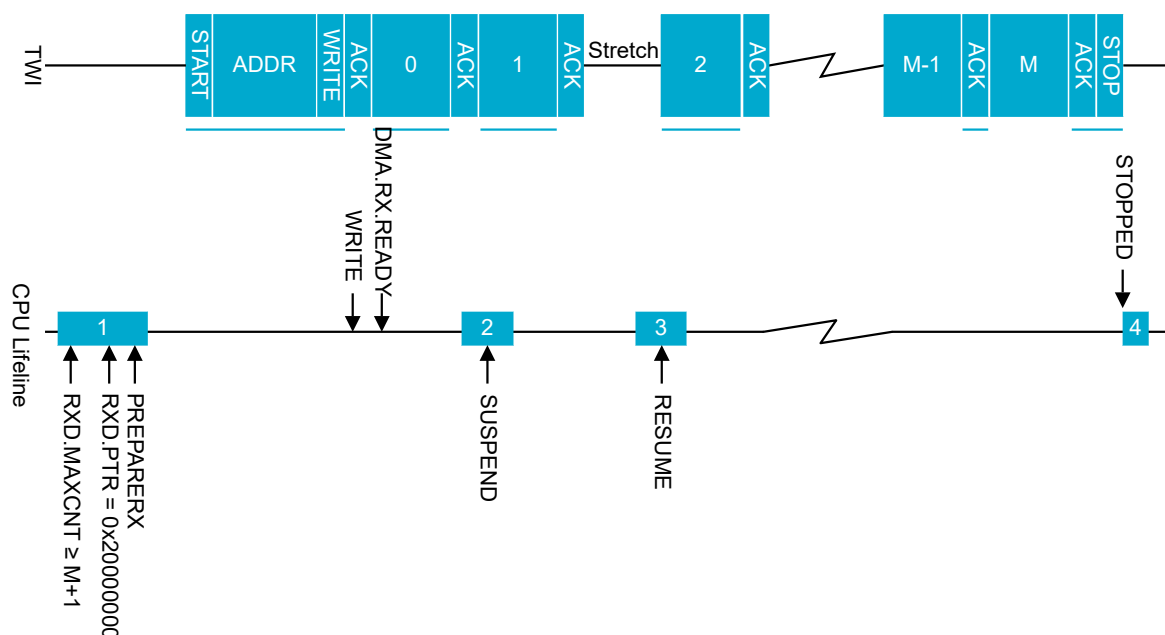


Figure 152: TWIS responding to a write command

8.24.6 TWI controller repeated start sequence

A repeated start sequence is where the TWI controller writes two bytes to TWIS, followed by reading four bytes from the target. This is shown in the following figure.

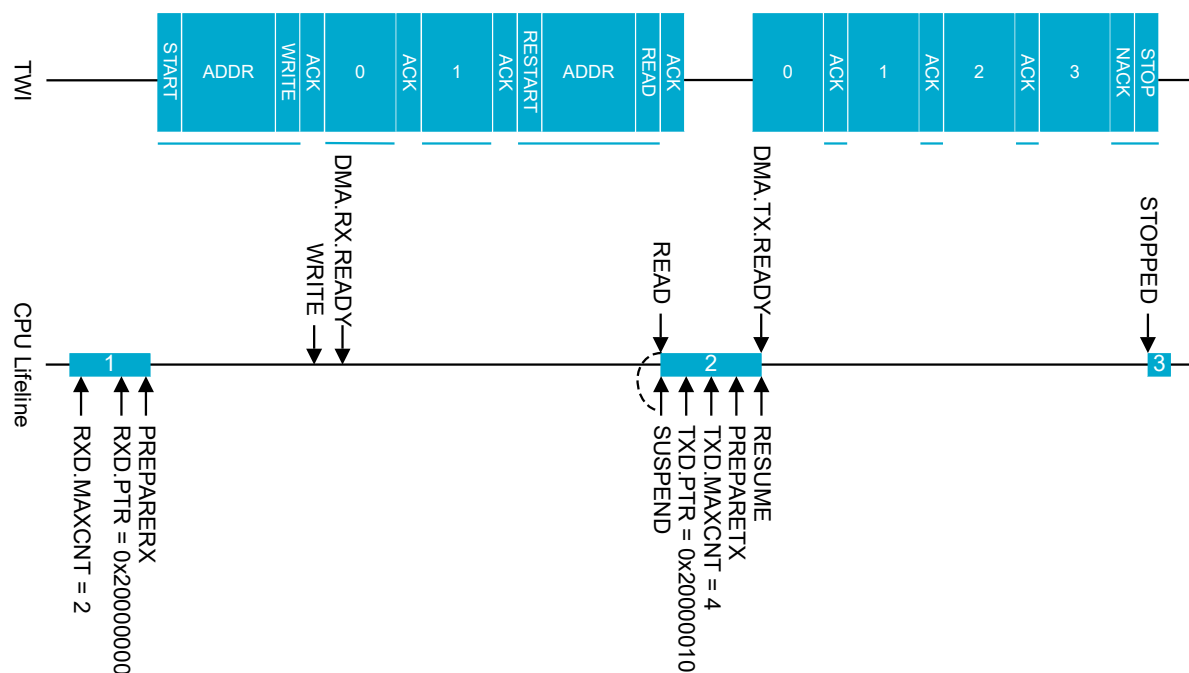


Figure 153: Repeated start sequence

In this example, the receiver does not know in advance what the controller wants to read. This information is in the first two received bytes of the write in the repeated start sequence. For the CPU to process the received data before TWIS replies to the read command, the SUSPEND task is triggered. This is enabled through a shortcut from the READ event generated when the read command is received. When the CPU has processed the incoming data and prepared the correct data response, the CPU will resume the transaction by triggering the RESUME task.

8.24.7 Terminate an ongoing TWI transaction

In some situations, an ongoing transaction must be terminated. This can happen when the external TWI controller is not responding correctly, for example.

To stop an ongoing transaction, trigger the STOP task. A STOPPED event will be generated when TWIS stops. It is not dependent on the STOP condition being generated on the TWI bus. TWIS will release the bus when it has stopped and returns to its IDLE state.

8.24.8 Low power

When the peripheral is not needed, stop and disable TWIM for lowest possible power consumption.

When the STOP task is sent, the software must wait until the STOPPED event is received before disabling the peripheral through the ENABLE register. If the peripheral is already stopped, the STOP task is not needed.

8.24.9 Target mode pin configuration

The SCL and SDA signals are mapped to physical pins using the PSEL.SCL and PSEL.SDA registers.

The PSEL.SCL and PSEL.SDA registers and their configurations are only used when TWIS is enabled, and retained while the device is in System ON mode. When the peripheral is disabled, the pins function as regular GPIOs, and use the configuration in their respective OUT bit field and PIN_CNF[n] register. Only configure PSEL.SCL and PSEL.SDA when TWIS is disabled.

When in System OFF mode or when TWIS is disabled, the TWIS pins must be configured in the GPIO peripheral as described in the following table to secure correct signal levels.

Only one peripheral can be assigned to drive a GPIO pin at a time. Failing to do so may result in unpredictable behavior.

TWIS signal	TWIS pin	Direction	Output value	Drive strength
SCL	As specified in PSEL.SCL	Input	Not applicable	S0D1
SDA	As specified in PSEL.SDA	Input	Not applicable	S0D1

Table 63: GPIO configuration before enabling peripheral

8.24.10 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
TWIS20 : S	GLOBAL	0x500C6000	US	S	SA	No	Two-wire interface target TWIS20
TWIS20 : NS		0x400C6000					
TWIS21 : S	GLOBAL	0x500C7000	US	S	SA	No	Two-wire interface target TWIS21
TWIS21 : NS		0x400C7000					
TWIS22 : S	GLOBAL	0x500C8000	US	S	SA	No	Two-wire interface target TWIS22
TWIS22 : NS		0x400C8000					
TWIS30 : S	GLOBAL	0x50104000	US	S	SA	No	Two-wire interface target TWIS30
TWIS30 : NS		0x40104000					

Configuration

Instance	Domain	Configuration
TWIS20 : S	GLOBAL	Use GPIO port P1, or dedicated pins on P2
TWIS20 : NS		CURRENTAMOUNT register not included.
TWIS21 : S	GLOBAL	Use GPIO port P1, or dedicated pins on P2
TWIS21 : NS		CURRENTAMOUNT register not included.
TWIS22 : S	GLOBAL	Use GPIO port P1, or dedicated pins on P2
TWIS22 : NS		CURRENTAMOUNT register not included.
TWIS30 : S	GLOBAL	Use GPIO port P0
TWIS30 : NS		CURRENTAMOUNT register not included.

Register overview

Register	Offset	TZ	Description
TASKS_STOP	0x004		Stop TWI transaction
TASKS_SUSPEND	0x00C		Suspend TWI transaction
TASKS_RESUME	0x010		Resume TWI transaction
TASKS_PREPARERX	0x020		Prepare the TWI slave to respond to a write command
TASKS_PREPARETX	0x024		Prepare the TWI slave to respond to a read command
TASKS_DMA.RX.ENABLEMATCH[n]	0x030		Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.
TASKS_DMA.RX.DISABLEMATCH[n]	0x040		Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.

Register	Offset	TZ	Description
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
SUBSCRIBE_SUSPEND	0x08C		Subscribe configuration for task SUSPEND
SUBSCRIBE_RESUME	0x090		Subscribe configuration for task RESUME
SUBSCRIBE_PREPARERX	0x0A0		Subscribe configuration for task PREPARERX
SUBSCRIBE_PREPARETX	0x0A4		Subscribe configuration for task PREPARETX
SUBSCRIBE_DMA.RX.ENABLEMATCH[n]	0x0B0		Subscribe configuration for task ENABLEMATCH[n]
SUBSCRIBE_DMA.RX.DISABLEMATCH[n]	0x0C0		Subscribe configuration for task DISABLEMATCH[n]
EVENTS_STOPPED	0x104		TWI stopped
EVENTS_ERROR	0x114		TWI error
EVENTS_WRITE	0x13C		Write command received
EVENTS_READ	0x140		Read command received
EVENTS_DMA.RX.END	0x14C		Generated after all MAXCNT bytes have been transferred
EVENTS_DMA.RX.READY	0x150		Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.
EVENTS_DMA.RX.BUSERROR	0x154		An error occurred during the bus transfer.
EVENTS_DMA.RX.MATCH[n]	0x158		Pattern match is detected on the DMA data bus.
EVENTS_DMA.TX.END	0x168		Generated after all MAXCNT bytes have been transferred
EVENTS_DMA.TX.READY	0x16C		Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.
EVENTS_DMA.TX.BUSERROR	0x170		An error occurred during the bus transfer.
PUBLISH_STOPPED	0x184		Publish configuration for event STOPPED
PUBLISH_ERROR	0x194		Publish configuration for event ERROR
PUBLISH_WRITE	0x1BC		Publish configuration for event WRITE
PUBLISH_READ	0x1C0		Publish configuration for event READ
PUBLISH_DMA.RX.END	0x1CC		Publish configuration for event END
PUBLISH_DMA.RX.READY	0x1D0		Publish configuration for event READY
PUBLISH_DMA.RX.BUSERROR	0x1D4		Publish configuration for event BUSERROR
PUBLISH_DMA.RX.MATCH[n]	0x1D8		Publish configuration for event MATCH[n]
PUBLISH_DMA.TX.END	0x1E8		Publish configuration for event END
PUBLISH_DMA.TX.READY	0x1EC		Publish configuration for event READY
PUBLISH_DMA.TX.BUSERROR	0x1F0		Publish configuration for event BUSERROR
SHORTS	0x200		Shortcuts between local events and tasks
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
ERRORSRC	0x4D0		Error source
MATCH	0x4D4		Status register indicating which address had a match
ENABLE	0x500		Enable TWIS
ADDRESS[n]	0x588		TWI slave address n
CONFIG	0x594		Configuration register for the address match mechanism
ORC	0x5C0		Over-read character. Character sent out in case of an over-read of the transmit buffer.
PSEL.SCL	0x600		Pin select for SCL signal
PSEL.SDA	0x604		Pin select for SDA signal
DMA.RX.PTR	0x704		RAM buffer start address
DMA.RX.MAXCNT	0x708		Maximum number of bytes in channel buffer
DMA.RX.AMOUNT	0x70C		Number of bytes transferred in the last transaction, updated after the END event. Also updated after each MATCH event.
DMA.RX.TERMINATEONBUSERROR	0x71C		Terminate the transaction if a BUSERROR event is detected.
DMA.RX.BUSERRORADDRESS	0x720		Address of transaction that generated the last BUSERROR event.
DMA.RX.MATCH.CONFIG	0x724		Configure individual match events
DMA.RX.MATCH.CANDIDATE[n]	0x728		The data to look for - any match will trigger the MATCH[n] event, if enabled.
DMA.TX.PTR	0x73C		RAM buffer start address
DMA.TX.MAXCNT	0x740		Maximum number of bytes in channel buffer

Register	Offset	TZ	Description
DMA.TX.AMOUNT	0x744		Number of bytes transferred in the last transaction, updated after the END event. Also updated after each MATCH event.
DMA.TX.TERMINATEONBUSERROR	0x754		Terminate the transaction if a BUSERROR event is detected.
DMA.TX.BUSERRORADDRESS	0x758		Address of transaction that generated the last BUSERROR event.

8.24.10.1 TASKS_STOP

Address offset: 0x004

Stop TWI transaction

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																					A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value	ID	Value		Description																														
A	W	TASKS_STOP					Stop TWI transaction																														
			Trigger		1		Trigger task																														

8.24.10.2 TASKS_SUSPEND

Address offset: 0x00C

Suspend TWI transaction

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																					A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value	ID	Value				Description																												
A	W	TASKS_SUSPEND							Suspend TWI transaction																												
			Trigger		1				Trigger task																												

8.24.10.3 TASKS_RESUME

Address offset: 0x010

Resume TWI transaction

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																							A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value	ID	Value				Description																														
A	W	TASKS_RESUME							Resume TWI transaction																														
			Trigger		1				Trigger task																														

8.24.10.4 TASKS_PREPARERX

Address offset: 0x020

Prepare the TWI slave to respond to a write command

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_PREPARERX						Prepare the TWI slave to respond to a write command																											
			Trigger	1				Trigger task																											

8.24.10.5 TASKS_PREPARETX

Address offset: 0x024

Prepare the TWI slave to respond to a read command

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_PREPARETX						Prepare the TWI slave to respond to a read command																											
			Trigger	1				Trigger task																											

8.24.10.6 TASKS_DMA

Peripheral tasks.

8.24.10.6.1 TASKS_DMA.RX

Peripheral tasks.

8.24.10.6.1.1 TASKS_DMA.RX.ENABLEMATCH[n] (n=0..3)

Address offset: 0x030 + (n × 0x4)

Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	ENABLEMATCH						Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.																											
			Trigger	1				Trigger task																											

8.24.10.6.1.2 TASKS_DMA.RX.DISABLEMATCH[n] (n=0..3)

Address offset: 0x040 + (n × 0x4)

Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	DISABLEMATCH						Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.																											
			Trigger	1				Trigger task																											

8.24.10.7 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task **STOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CHIDX		[0..255]	DPPI channel that task STOP will subscribe to																														
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.24.10.8 SUBSCRIBE_SUSPEND

Address offset: 0x08C

Subscribe configuration for task **SUSPEND**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CHIDX		[0..255]	DPPI channel that task SUSPEND will subscribe to																														
B	RW	EN	Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.24.10.9 SUBSCRIBE_RESUME

Address offset: 0x090

Subscribe configuration for task **RESUME**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task RESUME will subscribe to																													
B	RW	EN																																	
			Disabled	0	Disable subscription																														
			Enabled	1	Enable subscription																														

8.24.10.10 SUBSCRIBE_PREPARERX

Address offset: 0x0A0

Subscribe configuration for task **PREPARERX**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task PREPARERX will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.24.10.11 SUBSCRIBE_PREPARETX

Address offset: 0x0A4

Subscribe configuration for task [PREPARETX](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task PREPARETX will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.24.10.12 SUBSCRIBE_DMA

Subscribe configuration for tasks

8.24.10.12.1 SUBSCRIBE_DMA.RX

Subscribe configuration for tasks

8.24.10.12.1.1 SUBSCRIBE_DMA.RX.ENABLEMATCH[n] (n=0..3)

Address offset: 0x0B0 + (n × 0x4)

Subscribe configuration for task [ENABLEMATCH\[n\]](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that task ENABLEMATCH[n] will subscribe to																													
B	RW	EN																																	
			Disabled	0		Disable subscription																													
			Enabled	1		Enable subscription																													

8.24.10.12.1.2 SUBSCRIBE_DMA.RX.DISABLEMATCH[n] (n=0..3)

Address offset: 0x0C0 + (n × 0x4)

Subscribe configuration for task [DISABLEMATCH\[n\]](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															

8.24.10.13 EVENTS_STOPPED

Address offset: 0x104

TWI stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_STOPPED			TWI stopped																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.24.10.14 EVENTS_ERROR

Address offset: 0x114

TWI error

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_ERROR			TWI error																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.24.10.15 EVENTS_WRITE

Address offset: 0x13C

Write command received

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_WRITE			Write command received																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.24.10.16 EVENTS_READ

Address offset: 0x140

Read command received

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																							A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value	ID	Value	Description																																	
A	RW	EVENTS_READ				Read command received																																	
			NotGenerated	0		Event not generated																																	
			Generated	1		Event generated																																	

8.24.10.17 EVENTS_DMA

Peripheral events.

8.24.10.17.1 EVENTS_DMA.RX

Peripheral events.

8.24.10.17.1.1 EVENTS_DMA.RX.END

Address offset: 0x14C

Generated after all MAXCNT bytes have been transferred

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																						A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value	ID	Value	Description																																
A	RW	END				Generated after all MAXCNT bytes have been transferred																																
			NotGenerated	0		Event not generated																																
			Generated	1		Event generated																																

8.24.10.17.1.2 EVENTS_DMA.RX.READY

Address offset: 0x150

Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID					A																																		
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value	ID	Value	Description																																	
A	RW	READY				Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.																																	
			NotGenerated	0		Event not generated																																	
			Generated	1		Event generated																																	

8.24.10.17.1.3 EVENTS_DMA.RX.BUSERROR

Address offset: 0x154

An error occurred during the bus transfer.

When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	BUSERROR			An error occurred during the bus transfer.																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.24.10.17.1.4 EVENTS_DMA.RX.MATCH[n] (n=0..3)

Address offset: 0x158 + (n × 0x4)

Pattern match is detected on the DMA data bus.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	MATCH						Pattern match is detected on the DMA data bus.																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.24.10.17.2 EVENTS_DMA.TX

Peripheral events.

8.24.10.17.2.1 EVENTS_DMA.TX.END

Address offset: 0x168

Generated after all MAXCNT bytes have been transferred

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	END						Generated after all MAXCNT bytes have been transferred																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.24.10.17.2.2 EVENTS_DMA.TX.READY

Address offset: 0x16C

Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.24.10.17.2.3 EVENTS_DMA.TX.BUSERERROR

Address offset: 0x170

An error occurred during the bus transfer.

When this event is generated, the address which caused the error can be read from the BUSERERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	BUSERERROR				An error occurred during the bus transfer.																													
						When this event is generated, the address which caused the error can be read from the BUSERERRORADDRESS register.																													
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.24.10.18 PUBLISH_STOPPED

Address offset: 0x184

Publish configuration for event STOPPED

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A															A	A	A	A	A	A	A														
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																							
A	RW	CHIDX		[0..255]				DPPI channel that event STOPPED will publish to																																																							
B	RW	EN																																																													
			Disabled	0				Disable publishing																																																							
			Enabled	1				Enable publishing																																																							

8.24.10.19 PUBLISH_ERROR

Address offset: 0x194

Publish configuration for event ERROR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event ERROR will publish to																													
B	RW	EN																																	
			Disabled	0		Disable publishing																													
			Enabled	1		Enable publishing																													

8.24.10.20 PUBLISH_WRITE

Address offset: 0x1BC

Publish configuration for event **WRITE**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event WRITE will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.24.10.21 PUBLISH_READ

Address offset: 0x1C0

Publish configuration for event **READ**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event READ will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.24.10.22 PUBLISH_DMA

Publish configuration for events

8.24.10.22.1 PUBLISH_DMA.RX

Publish configuration for events

8.24.10.22.1.1 PUBLISH_DMA.RX.END

Address offset: 0x1CC

Publish configuration for event **END**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event END will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.24.10.22.1.2 PUBLISH_DMA.RX.READY

Address offset: 0x1D0

Publish configuration for event **READY**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																													
ID				B																								A												A	A	A	A	A	A	A																		
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																										
A	RW	CHIDX		[0..255]		DPPI channel that event READY will publish to																																																										
B	RW	EN																																																														
			Disabled	0	Disable publishing																																																											
			Enabled	1	Enable publishing																																																											

8.24.10.22.1.3 PUBLISH_DMA.RX.BUSERROR

Address offset: 0x1D4

Publish configuration for event **BUSERROR**

When this event is generated, the address which caused the error can be read from the **BUSERRORADDRESS** register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event BUSERROR will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.24.10.22.1.4 PUBLISH_DMA.RX.MATCH[n] (n=0..3)

Address offset: 0x1D8 + (n × 0x4)

Publish configuration for event **MATCH[n]**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A										A	A	A	A	A	A	A																			
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that event MATCH[n] will publish to																																																									
B	RW	EN																																																													
			Disabled	0	Disable publishing																																																										
			Enabled	1	Enable publishing																																																										

8.24.10.22.2 PUBLISH_DMA.TX

Publish configuration for events

8.24.10.22.2.1 PUBLISH_DMA.TX.END

Address offset: 0x1E8

Publish configuration for event [END](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event END will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.24.10.22.2.2 PUBLISH_DMA.TX.READY

Address offset: 0x1EC

Publish configuration for event [READY](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that event READY will publish to																											
B	RW	EN																																	
			Disabled	0				Disable publishing																											
			Enabled	1				Enable publishing																											

8.24.10.22.2.3 PUBLISH_DMA.TX.BUSERERROR

Address offset: 0x1F0

Publish configuration for event [BUSERERROR](#)

When this event is generated, the address which caused the error can be read from the [BUSERERRORADDRESS](#) register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CHIDX		[0..255]	DPPI channel that event BUSERERROR will publish to																														
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.24.10.23 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M L K J I H G F E D C																B								A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
L	RW	DMATXEND	Disabled	0	Disable																														
			Enabled	1	Enable																														
			Enable or disable interrupt for event DMATXEND																																
M	RW	DMATXREADY	Disabled	0	Disable																														
			Enabled	1	Enable																														
			Enable or disable interrupt for event DMATXREADY																																
N	RW	DMATXBUSERROR	Disabled	0	Disable																														
			Enabled	1	Enable																														
			Enable or disable interrupt for event DMATXBUSERROR																																
When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																																			
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.24.10.25 INTENSET

Address offset: 0x304

Enable interrupt

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			N M L K J I H G F E D C																B								A							
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	STOPPED			Write '1' to enable interrupt for event STOPPED																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
B	RW	ERROR			Write '1' to enable interrupt for event ERROR																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
C	RW	WRITE			Write '1' to enable interrupt for event WRITE																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
D	RW	READ			Write '1' to enable interrupt for event READ																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
E	RW	DMARXEND			Write '1' to enable interrupt for event DMARXEND																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
F	RW	DMARXREADY			Write '1' to enable interrupt for event DMARXREADY																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			N M L K J I H G F E D C																B								A							
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
G	RW	DMARXBUSERROR			Write '1' to enable interrupt for event DMARXBUSERROR																													
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
		Enabled	1	Read: Enabled																														
H-K	RW	DMARXMATCH[i] (i=0..3)			Write '1' to enable interrupt for event DMARXMATCH[i]																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
L	RW	DMATXEND			Write '1' to enable interrupt for event DMATXEND																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
M	RW	DMATXREADY			Write '1' to enable interrupt for event DMATXREADY																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
N	RW	DMATXBUSERROR			Write '1' to enable interrupt for event DMATXBUSERROR																													
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
		Enabled	1	Read: Enabled																														

8.24.10.26 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M L K J I H G F E D C																B								A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	STOPPED			Write '1' to disable interrupt for event STOPPED																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	ERROR			Write '1' to disable interrupt for event ERROR																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	WRITE			Write '1' to disable interrupt for event WRITE																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	READ			Write '1' to disable interrupt for event READ																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N M L K J I H G F E D C																B								A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
E	RW	DMARXEND	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event DMARXEND																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
F	RW	DMARXREADY	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event DMARXREADY																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
G	RW	DMARXBUSERROR	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event DMARXBUSERROR																														
			Disabled	0	When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			Enabled	1	Disable																														
H-K	RW	DMARXMATCH[i] (i=0..3)	Enabled	1	Read: Disabled																														
			Clear	1	Write '1' to disable interrupt for event DMARXMATCH[i]																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
L	RW	DMATXEND	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event DMATXEND																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
M	RW	DMATXREADY	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event DMATXREADY																														
			Disabled	0	Disable																														
			Enabled	1	Read: Disabled																														
N	RW	DMATXBUSERROR	Enabled	1	Read: Enabled																														
			Clear	1	Write '1' to disable interrupt for event DMATXBUSERROR																														
			Disabled	0	When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			Enabled	1	Disable																														

8.24.10.27 ERRORSRC

Address offset: 0x4D0

Error source

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																			
ID				C B A																																			
Reset 0x00000000				0 0																																			
ID	R/W	Field	Value ID	Value	Description																																		
A	RW	OVERFLOW W1C			RX buffer overflow detected, and prevented																																		
			NotDetected	0	Error did not occur																																		
			Detected	1	Error occurred																																		
B	RW	DNACK W1C			NACK sent after receiving a data byte																																		
			NotReceived	0	Error did not occur																																		
			Received	1	Error occurred																																		
C	RW	OVERREAD W1C			TX buffer over-read detected, and prevented																																		
			NotDetected	0	Error did not occur																																		
			Detected	1	Error occurred																																		

8.24.10.28 MATCH

Address offset: 0x4D4

Status register indicating which address had a match

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	MATCH		[0..1]	Indication of which address in ADDRESS that matched the incoming address																														

8.24.10.29 ENABLE

Address offset: 0x500

Enable TWIS

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE			Enable or disable TWIS																														
			Disabled	0	Disable TWIS																														
			Enabled	9	Enable TWIS																														

8.24.10.30 ADDRESS[n] (n=0..1)

Address offset: 0x588 + (n × 0x4)

TWI slave address n

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ADDRESS						TWI slave address																											

8.24.10.31 CONFIG

Address offset: 0x594

Configuration register for the address match mechanism

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					B A																															
Reset 0x00000001					0 1																															
ID	R/W	Field	Value ID	Value	Description																															
A-B	RW	ADDRESS[i] (i=0..1)			Enable or disable address matching on ADDRESS[i]																															
			Disabled	0	Disabled																															
			Enabled	1	Enabled																															

8.24.10.32 ORC

Address offset: 0x5C0

Over-read character. Character sent out in case of an over-read of the transmit buffer.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

8.24.10.33 PSEL.SCL

Address offset: 0x600

Pin select for SCL signal

Bit number		31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																											
ID		C																																		B	B	B	A	A	A	A	A		
Reset 0xFFFFFFFF		1 1																																											
ID	R/W	Field	Value ID	Value	Description																																								
A	RW	PIN		[0..31]	Pin number																																								
B	RW	PORT		[0..7]	Port number																																								
C	RW	CONNECT			Connection																																								
			Disconnected	1	Disconnect																																								
			Connected	0	Connect																																								

8.24.10.34 PSEL.SDA

Address offset: 0x604

Pin select for SDA signal

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID		C																							B					A				
Reset 0xFFFFFFFF		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
ID	R/W	Field	Value ID	Value	Description																													
A	RW	PIN		[0..31]	Pin number																													
B	RW	PORT		[0..7]	Port number																													
C	RW	CONNECT			Connection																													
			Disconnected	1	Disconnect																													
			Connected	0	Connect																													

8.24.10.35 DMA.RX.PTR

Address offset: 0x704

RAM buffer start address

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A		
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
A	RW	PTR			RAM buffer start address for this EasyDMA channel. This address is a word aligned Data RAM address.																																	

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.24.10.36 DMA.RX.MAXCNT

Address offset: 0x708

Maximum number of bytes in channel buffer

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0											
ID																												A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0										
ID	R/W	Field	Value ID	Value				Description																																						
A	RW	MAXCNT		[1..0xFFFF]				Maximum number of bytes in channel buffer																																						

8.24.10.37 DMA.RX.AMOUNT

Address offset: 0x70C

Number of bytes transferred in the last transaction, updated after the END event.

Also updated after each MATCH event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	AMOUNT		[1..0xFFFF]								Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.																							

8.24.10.38 DMA.RX.TERMINATEONBUSERROR

Address offset: 0x71C

Terminate the transaction if a BUSERROR event is detected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ENABLE																																	
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.24.10.39 DMA.RX.BUSERRORADDRESS

Address offset: 0x720

Address of transaction that generated the last BUSERROR event.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A		
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																		
A	RW	ADDRESS																																					

8.24.10.40 DMA.RX.MATCH

Registers to control the behavior of the pattern matcher engine

8.24.10.40.1 DMA.RX.MATCH.CONFIG

Address offset: 0x724

Configure individual match events

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				H G F E																D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-D	RW	ENABLE[i] (i=0..3)			Enable match filter i																														
			Disabled	0	Match filter disabled																														
			Enabled	1	Match filter enabled																														
E-H	RW	ONESHOT[i] (i=0..3)			Configure match filter i as one-shot or sticky																														
					One-shot match filters can be used together with shortcuts to check for continuous data sequences by disabling the filter if the next data is not a match.																														
					Note: The presence of these shorts depends on the configuration of the peripheral integrating this EasyDMA.																														
			Continuous	0	Match filter stays enabled until disabled by task																														
			Oneshot	1	Match filter stays enabled until next data word is received																														

8.24.10.40.2 DMA.RX.MATCH.CANDIDATE[n] (n=0..3)

Address offset: 0x728 + (n × 0x4)

The data to look for - any match will trigger the MATCH[n] event, if enabled.

Note: This register can be updated while a transfer is in progress, but the new value will not take effect until a match has been found or the transfer is done. That makes it possible to write a new set of match words which will be searched for immediately after the event triggers.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	DATA						Data to look for																											

8.24.10.41 DMA.TX.PTR

Address offset: 0x73C

RAM buffer start address

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						RAM buffer start address for this EasyDMA channel. This address is a word aligned Data RAM address.																											

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.24.10.42 DMA.TX.MAXCNT

Address offset: 0x740

Maximum number of bytes in channel buffer

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value				Description																													
A	RW	MAXCNT		[1..0xFFFF]				Maximum number of bytes in channel buffer																													

8.24.10.43 DMA.TX.AMOUNT

Address offset: 0x744

Number of bytes transferred in the last transaction, updated after the END event.

Also updated after each MATCH event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	AMOUNT		[1..0xFFFF]				Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.																											

8.24.10.44 DMA.TX.TERMINATEONBUSERROR

Address offset: 0x754

Terminate the transaction if a BUSERROR event is detected.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ENABLE																																	
			Disabled	0				Disable																											
			Enabled	1				Enable																											

8.24.10.45 DMA.TX.BUSERRORADDRESS

Address offset: 0x758

Address of transaction that generated the last BUSERROR event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ADDRESS																																	

8.25 UARTE — Universal asynchronous receiver/transmitter with EasyDMA

The Universal asynchronous receiver/transmitter with EasyDMA peripheral (UARTE) provides a full-duplex, asynchronous serial communication interface with hardware flow control.

The main features of UARTE are the following:

- Full-duplex operation
- EasyDMA direct transfer to and from RAM
- Individual selection of I/O pins
- Slow instances with up to 1 Mbps baud rate
- Optional even and odd parity bit checking and generation
- One or two stop bits
- Configurable data frame size: 4 bit to 9 bit
- 9-bit mode support with address matching in RX
- Automatic hardware flow control
- Supports return to the IDLE state between transactions (when using HW flow control)
- Interrupt generation after programmable timeout

- Compare match filter for generating events or interrupts

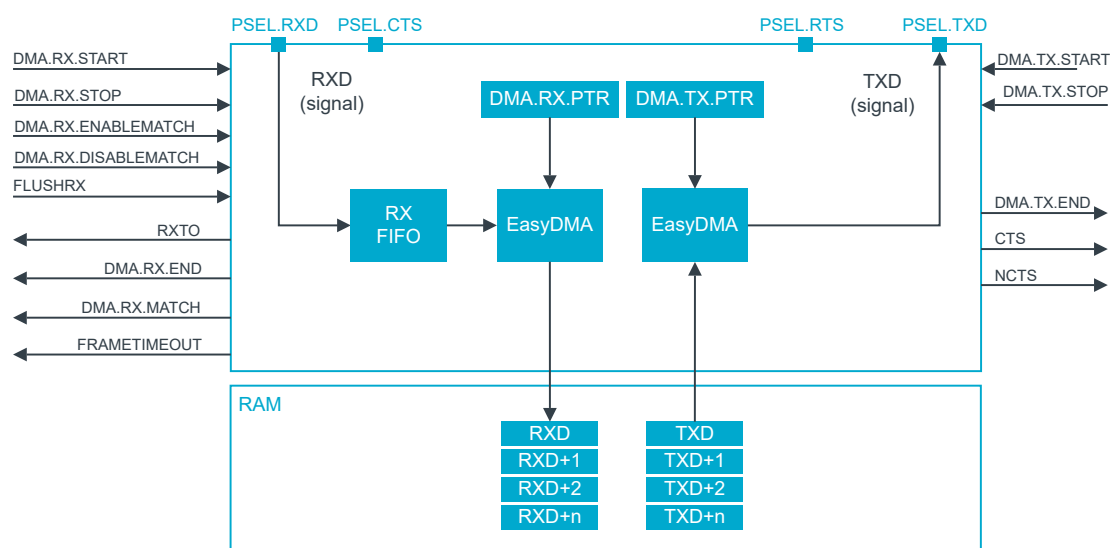


Figure 154: UARTE configuration

Note: The external crystal oscillator must be enabled to obtain sufficient clock accuracy for stable communication. See [CLOCK — Clock control](#) on page 76 for more information.

8.25.1 EasyDMA

UARTE implements EasyDMA for reading and writing to and from RAM.

If the DMA.TX.PTR and the DMA.RX.PTR are not pointing to the RAM region, an EasyDMA transfer may result in a HardFault or RAM corruption. See [Memory](#) on page 19 for more information about each memory region.

The DMA.RX.PTR, DMA.TX.PTR, DMA.RX.MAXCNT, and DMA.TX.MAXCNT registers are double-buffered. They can be updated and prepared for the next reception or transmission immediately after having received the DMA.RX.READY or DMA.TX.READY events.

The DMA.RX.END and DMA.TX.END events indicate that the EasyDMA is finished accessing the RX or TX buffer in RAM.

For detailed information regarding the use of EasyDMA, see [EasyDMA](#) on page 34.

8.25.2 Transmission

The first step of a DMA transmission is storing bytes in the transmit buffer and configuring EasyDMA. This is achieved by writing the initial address pointer to DMA.TX.PTR, and the number of bytes to transmit from the RAM buffer to DMA.TX.MAXCNT. The UARTE transmission is started by triggering the DMA.TX.START task.

After each byte has been sent over the TXD line, a TXDRDY event is generated.

When then bytes have been transmitted, the DMA.TX.END event is generated.

A UARTE transmission sequence is stopped by triggering the DMA.TX.STOP task. A TXSTOPPED event will be generated when the UARTE transmitter has stopped.

If the DMA.TX.END event has not been generated when the UARTE transmitter stops, UARTE will generate the DMA.TX.END event explicitly even though all bytes specified in the DMA.TX.MAXCNT register have not been transmitted.

If flow control is enabled in the HWFC field in the CONFIG register, a transmission will be automatically suspended when CTS is deactivated, and resumed when CTS is activated again, as shown in the following figure. A byte that is in transmission when CTS is deactivated will finish transmitting before the transmission is suspended.

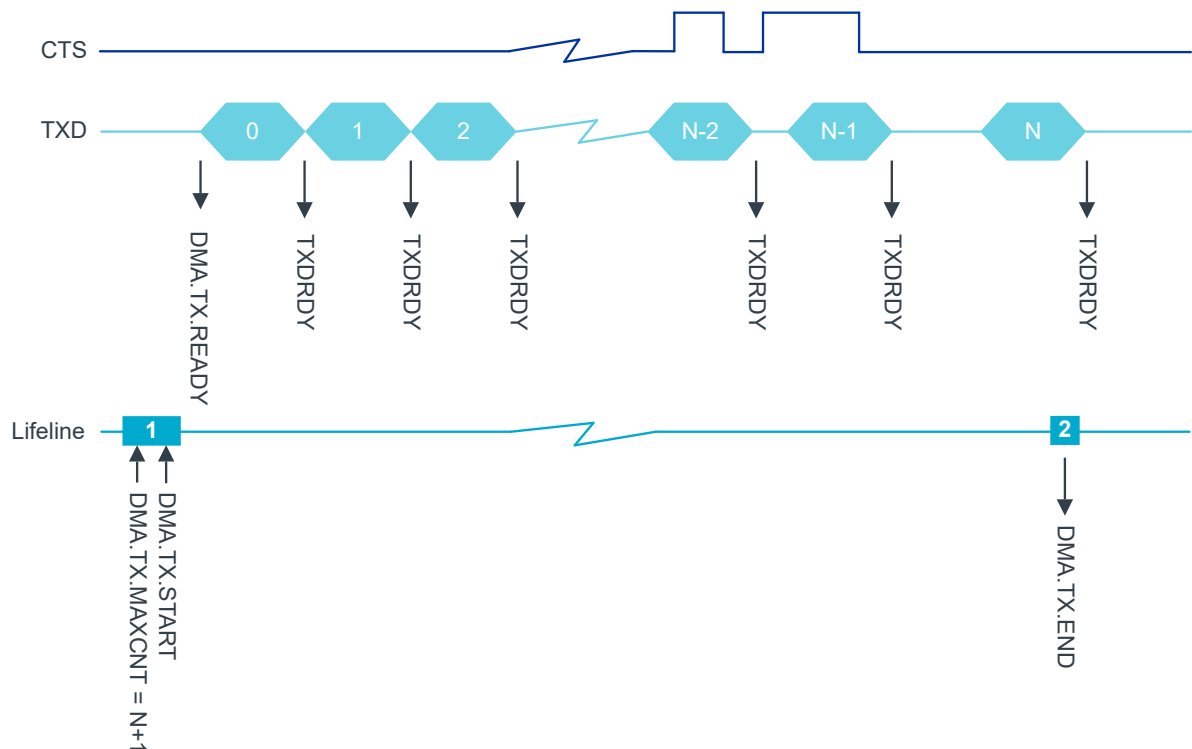


Figure 155: UART transmission

The UART transmitter is least active when it is stopped, consuming the least amount of energy. This is before it is started via `DMA.TX.START`, or after it has been stopped via `DMA.TX.STOP` and the `TXSTOPPED` event has been generated. See [POWER — Power control](#) on page 97 for more information about power modes.

8.25.3 Reception

The UART receiver is started by triggering the `DMA.RX.START` task. The UART receiver uses EasyDMA to store incoming data in an RX buffer in RAM.

The RX buffer is located at the address specified in the `DMA.RX.PTR` register. The `DMA.RX.PTR` register is double-buffered and can be updated and prepared for the next `DMA.RX.START` task immediately after the `EVENTS_DMA.RX.READY` event is generated. The size of the RX buffer is specified in the `DMA.RX.MAXCNT` register. UART generates an `DMA.RX.END` event when it has filled up the RX buffer, as seen in the following figure.

For each byte received over the RXD line, an `RXDRDY` event is generated. This event is likely to occur before the corresponding data has been transferred to RAM.

The `DMA.RX.AMOUNT` register can be queried following an `DMA.RX.END` event to see how many new bytes have been transferred to the RX buffer in RAM since the previous `DMA.RX.END` event.

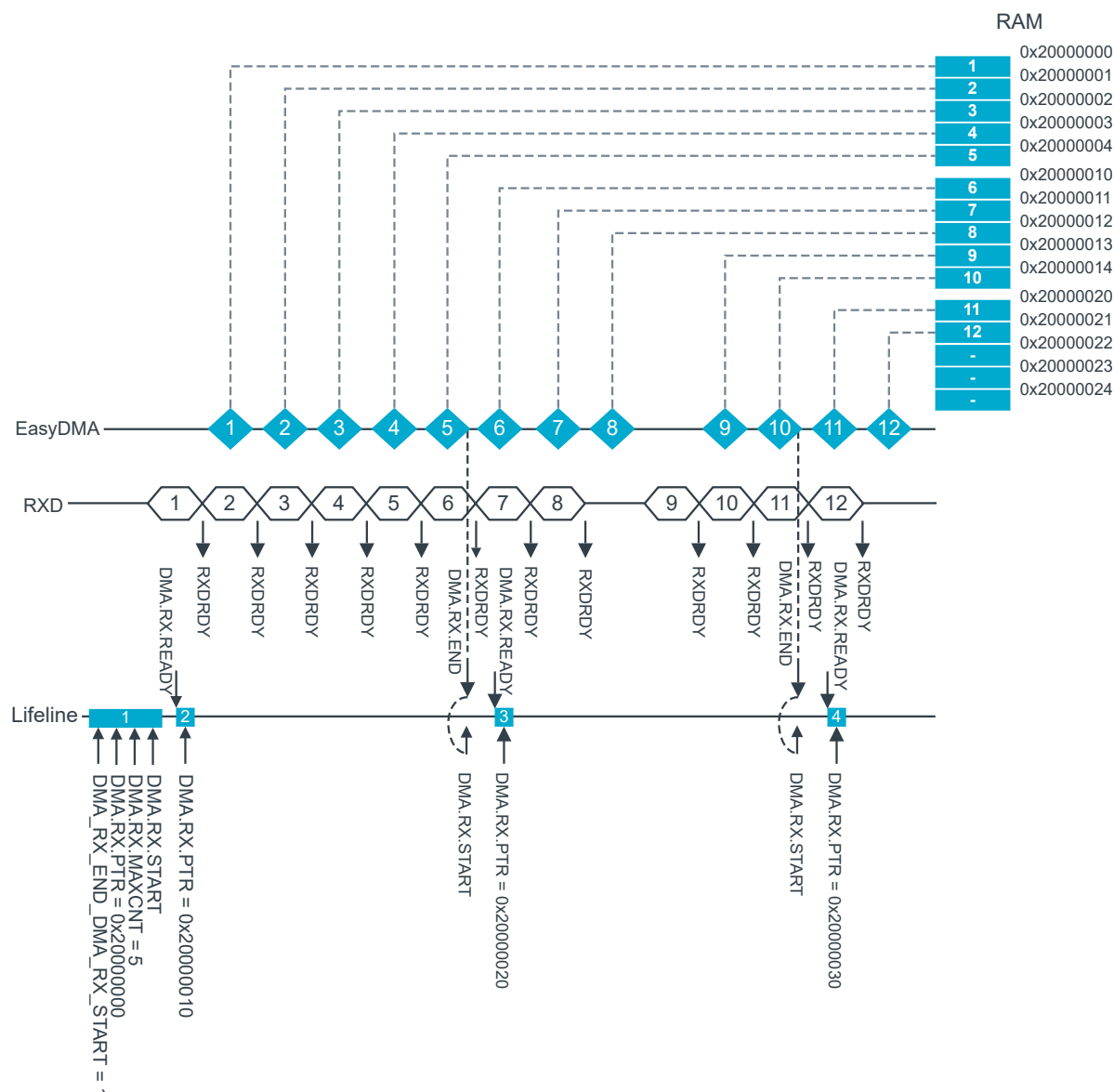


Figure 156: UARTE reception

The UARTE receiver is stopped by triggering the DMA.RX.STOP task. An RXTO event is generated when UARTE has stopped. UARTE makes sure that an impending DMA.RX.END event is generated before the RXTO event is generated. This means that UARTE guarantees that no DMA.RX.END event is generated after RXTO, unless UARTE is restarted or a FLUSHRX command is issued after the RXTO event is generated.

Note: If the DMA.RX.END event has not been generated when the UARTE receiver stops, indicating that all pending content in the RX FIFO has been moved to the RX buffer, UARTE generates the DMA.RX.END event explicitly even though the RX buffer is not full. In this scenario, the DMA.RX.END event is generated before the RXTO event is generated.

To determine the amount of bytes the RX buffer has received, the CPU can read the DMA.RX.AMOUNT register following the DMA.RX.END event or the RXTO event.

If sent in succession immediately after the RTS signal is deactivated, UARTE can receive up to four bytes after the DMA.RX.STOP task has been triggered.

After the RXTO event is generated, the internal RX FIFO can still contain data. To move this data to RAM, the FLUSHRX task must be triggered. The RX buffer must be emptied, or the DMA.RX.PTR register

must be updated before the FLUSHRX task is triggered. This ensures the data in the RX buffer is not overwritten. To make sure that all data in the RX FIFO is moved to the RX buffer, the DMA.RX.MAXCNT register must be set to $\text{DMA.RX.MAXCNT} > 4$, as seen in the following figure. The UARTE will generate the `EVENTS_DMA.RX.END` event after completing the FLUSHRX task even if the RX FIFO was empty or if the RX buffer does not fill up. After the `DMA.RX.END` event, the `DMA.RX.AMOUNT` register holds the actual amount of bytes transferred to the RX buffer.

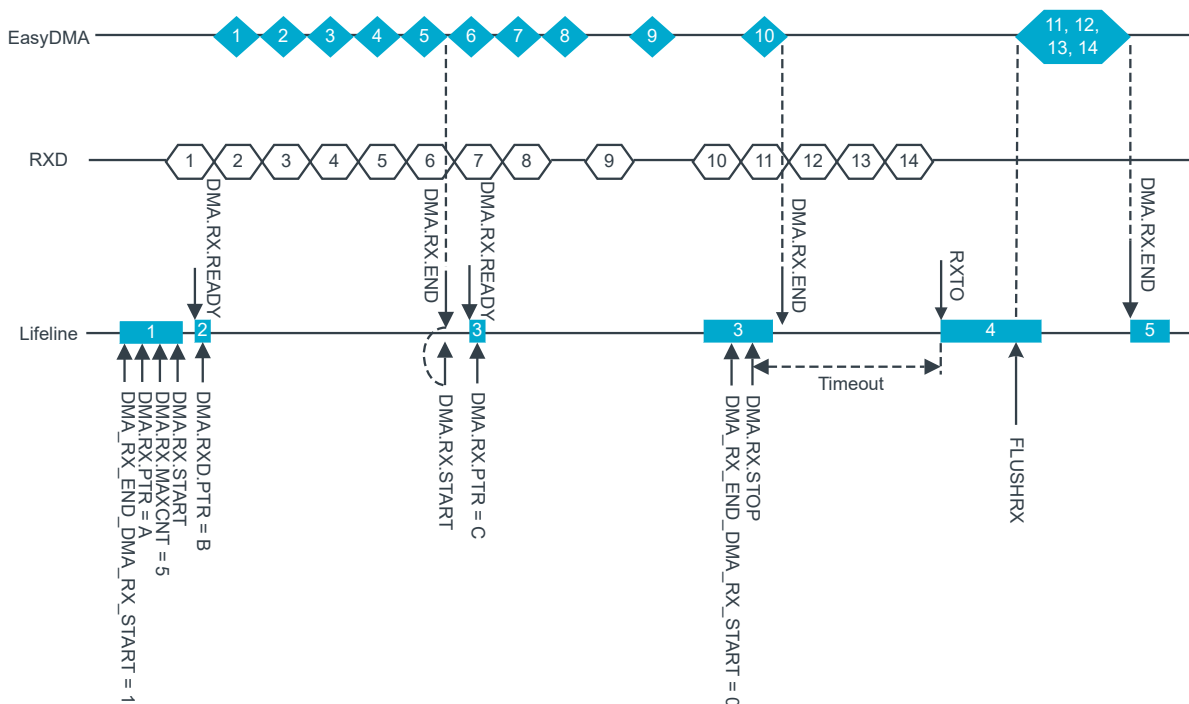


Figure 157: UARTE reception with forced stop through `DMA.RX.STOP`

If hardware flow control is enabled in the `HWFC` field in the `CONFIG` register, the `RTS` signal will be deactivated when the receiver is stopped via the `DMA.RX.STOP` task, or when UARTE can only receive four more bytes in its internal RX FIFO.

With flow control disabled, the UARTE will function in the same way as when the flow control is enabled, except that the `RTS` line will not be used. This means that no signal will be generated when UARTE is only able to receive four additional bytes in its internal RX FIFO. Data received when the internal RX FIFO is full, will be lost.

The UARTE receiver is least active when it is stopped, consuming the least amount of energy. This is before it is started via `DMA.RX.START`, or after it has been stopped via `DMA.RX.STOP` and the `RXT0` event has been generated. See [POWER — Power control](#) on page 97 for more information about power modes.

8.25.4 Data frame size

UARTE implements a configurable data frame size of 4 bits to 9 bits and is set in the register `CONFIG` on page 716. If a value greater than 9 or less than 4 is written to this register, the frame size will be set to 8 bits and the register will read back a value of 8.

When UARTE is used with the 9 bit frame size, a 9th bit is added after the MSB of the 8 bit data frame, and before the parity and stop bits, as shown in the following figure. This bit indicates if the 8 bit data is an address or data. If the 9th bit is 1, the 8 bit data is interpreted as an address. If the 9th bit is 0, the 8 bit data is interpreted as data.

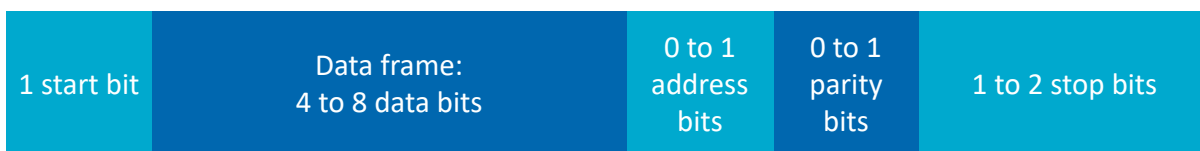


Figure 158: UARTE frame

When UART is in RX configured with a 9 bit frame size, all frames are ignored until a frame with the address bit set is received, and the 8 remaining bits of the data frame matches the address set in the register [ADDRESS](#) on page 716. The frames following the matching address are received as an 8 bit data frame until the next frame where the address bit set is received, the address bit is not stored. If the address does not match [ADDRESS](#), the following frames are ignored.

If the parity bit is enabled, the address bit is not included in the parity calculation.

When UART TX is started, the first byte in the buffer read by EasyDMA is treated as an address, and transmitted with the address bit set to 1. The next bits in the buffer are treated as data and transmitted with the address bit set to 0.

When UARTE uses a data frame size less than an 8 bits, the data is trimmed from an 8 bit frame size in the RAM buffer for TX, and padded before being stored in the RAM buffer for RX. The ENDIAN field in the register [CONFIG](#) defines if the data is trimmed from MSB or LSB of the 8 bit buffer frame.

8.25.5 Frame timeout interrupt

UARTE can generate an event after a programmable timeout.

If enabled with the FRAMETIMEOUT field in register [CONFIG](#) on page 716, a counter starts when the event [EVENTS_RXDRDY](#) on page 701 is generated, and counts the number of periods given in register [FRAMETIMEOUT](#) on page 716. The period time of the counter is equal to the period of one bit on the UART TX line given by the baud rate set in [BAUDRATE](#) on page 715. If the task [TASKS_DMA.RX.STOP](#) on page 697 is not triggered before the timeout expires, then the event [EVENTS_FRAMETIMEOUT](#) is generated. The counter is reset when the event [EVENTS_RXDRDY](#) on page 701 is generated. UARTE reception can be stopped on timeout by setting the short from the [FRAMETIMEOUT](#) event to the [DMA.RX.STOP](#) task in [SHORTS](#) on page 709.

This feature can be used to support variable length UART transmission with no end of transmission tag. After the last UART frames are received within the configured timeout, an interrupt is generated and the data can be processed.

The following figure shows an example where the UART receives a transmission with size of two frames, and the [FRAMETIMEOUT](#) register is set to 16.

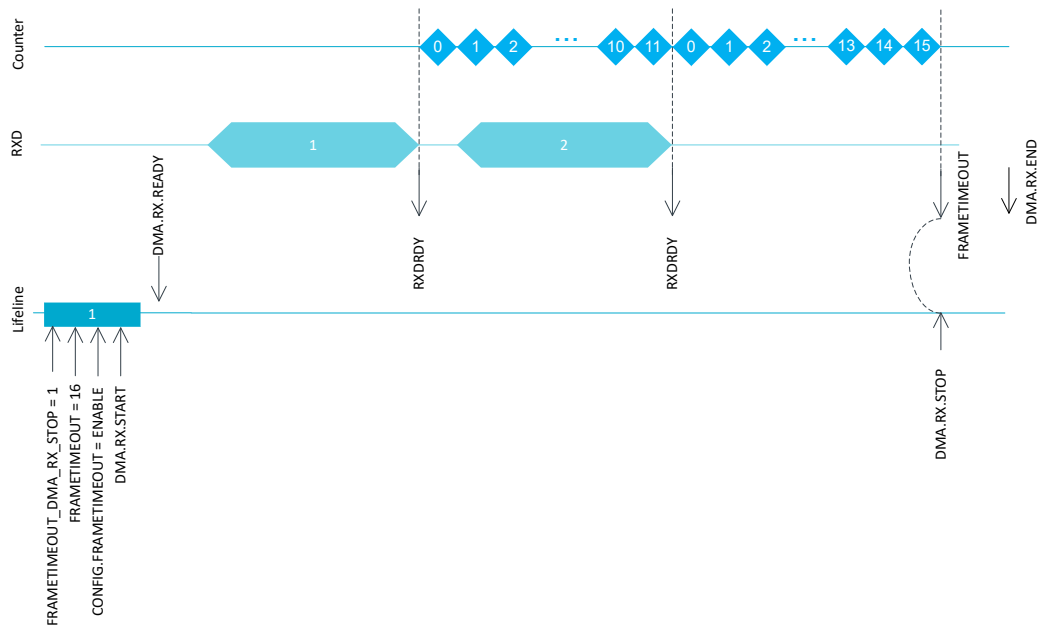


Figure 159: UARTe reception frame timeout

The minimum value of the `FRAMETIMEOUT` register must be set to a value larger than the configured UART frame length.

8.25.6 Error conditions

An ERROR event, in the form of a framing error, will be generated if a valid stop bit is not detected in a frame. Another ERROR event, in the form of a break condition, will be generated if the RXD line is held active low for longer than the length of a data frame. A framing error is always generated before a break condition occurs.

An ERROR event will not stop reception. If the error was a parity error, the received byte is still transferred into RAM along with any following bytes. If a framing error occurs (wrong stop bit), that byte will not be stored in RAM but the next incoming bytes will.

8.25.7 Using the UARTe without flow control

If flow control is not enabled, the interface will behave as if the CTS and RTS lines are held active.

8.25.8 Parity and stop bit configuration

Automatic even parity generation for both transmission and reception can be configured using the register `CONFIG` on page 716. If odd parity is required, it can be configured using the register `CONFIG` on page 716. See the register description for details.

The amount of stop bits can be configured in the register `CONFIG` on page 716.

8.25.9 Compare match filter

UARTe has a compare match filter that can watch for a specific sequence of data. This feature is implemented in EasyDMA on the RX channel.

UARTe can generate events or interrupts when the specific data sequence is received. The event `EVENTS_DMA.RX.MATCH[n]` ($n=0..3$) on page 703 is generated when there is a match in the data stream being received. The number of `MATCH` can be different for each instance. See [Registers](#) on page 694 for how many `MATCH` events are implemented per instance.

Register [DMA.RX.MATCH.CANDIDATE\[n\]](#) ($n=0..3$) on page 720 configures the pattern for comparison. The filter is enabled with the corresponding ENABLE bit in register [DMA.RX.MATCH.CONFIG](#) on page 719. The [DMA.RX.ENABLEMATCH](#) task can be used to set that bit, and the [DMA.RX.DISABLEMATCH](#) task will clear that bit. If the ONESHOT[i] field in the [CONFIG](#) register is set, the corresponding ENABLE[i] bit will be cleared on a successful match.

For detailed information regarding the use of pattern matching in the EasyDMA engine, see [EasyDMA](#) on page 34.

8.25.10 Low power

To ensure lowest possible power consumption when the peripheral is not needed, stop and disable UARTE.

The DMA.TX.STOP and DMA.RX.STOP tasks are not always needed (the peripheral might already be stopped). If DMA.TX.STOP or DMA.RX.STOP is sent, software waits until the TXSTOPPED or RXTO event is received before disabling the peripheral through the ENABLE register.

8.25.11 Pin configuration

The RXD, CTS (Clear To Send, active low), RTS (Request To Send, active low), and TXD signals associated with UARTE are mapped to physical pins according to the configuration specified in the PSEL.n registers.

These registers and their configurations are only used when UARTE is enabled, and retained while the device is in System ON mode. The PSEL.n registers can be configured only when UARTE is disabled.

To ensure correct behavior when in System OFF mode, the pins must be configured in the GPIO peripheral as described in the following table.

Only one peripheral can be assigned to drive a particular GPIO pin at a time. Failing to do so may result in unpredictable behavior.

UARTE signal	UARTE pin	Direction	Output value
RXD	As specified in PSEL.RXD	Input	Not applicable
CTS	As specified in PSEL.CTS	Input	Not applicable
RTS	As specified in PSEL.RTS	Output	1
TXD	As specified in PSEL.TXD	Output	1

Table 64: GPIO configuration before enabling peripheral

8.25.12 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
UARTE00 : S	GLOBAL	0x5004A000	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE00
UARTE00 : NS		0x4004A000					
UARTE20 : S	GLOBAL	0x500C6000	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE20
UARTE20 : NS		0x400C6000					
UARTE21 : S	GLOBAL	0x500C7000	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE21
UARTE21 : NS		0x400C7000					
UARTE22 : S	GLOBAL	0x500C8000	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE22
UARTE22 : NS		0x400C8000					
UARTE30 : S	GLOBAL	0x50104000	US	S	SA	No	Universal asynchronous receiver/transmitter UARTE30
UARTE30 : NS		0x40104000					

Configuration

Instance	Domain	Configuration
UARTE00 : S UARTE00 : NS	GLOBAL	Use GPIO port P2
		The core frequency scales with the CPU frequency, see PLL.FREQ (Retained) on page 96
		Timeout interrupt is included.
		Supports data frame sizes 4, 5, 6, 7, 8, and 9 bits.
UARTE20 : S UARTE20 : NS	GLOBAL	Peripheral clock frequency is 128 MHz.
		Use GPIO port P1, or dedicated pins on P2
		Timeout interrupt is included.
		Supports data frame sizes 4, 5, 6, 7, 8, and 9 bits.
UARTE21 : S UARTE21 : NS	GLOBAL	Peripheral clock frequency is 16 MHz.
		Use GPIO port P1, or dedicated pins on P2
		Timeout interrupt is included.
		Supports data frame sizes 4, 5, 6, 7, 8, and 9 bits.
UARTE22 : S UARTE22 : NS	GLOBAL	Peripheral clock frequency is 16 MHz.
		Use GPIO port P1, or dedicated pins on P2
		Timeout interrupt is included.
		Supports data frame sizes 4, 5, 6, 7, 8, and 9 bits.
UARTE30 : S UARTE30 : NS	GLOBAL	Peripheral clock frequency is 16 MHz.
		Use GPIO port P0
		Timeout interrupt is included.
		Supports data frame sizes 4, 5, 6, 7, 8, and 9 bits.

Register overview

Register	Offset	TZ	Description
TASKS_FLUSHRX	0x01C		Flush RX FIFO into RX buffer
TASKS_DMA.RX.START	0x028		Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.
TASKS_DMA.RX.STOP	0x02C		Stops operation using easyDMA. This does not trigger an END event.
TASKS_DMA.RX.ENABLEMATCH[n]	0x030		Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.
TASKS_DMA.RX.DISABLEMATCH[n]	0x040		Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.
TASKS_DMA.TX.START	0x050		Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.
TASKS_DMA.TX.STOP	0x054		Stops operation using easyDMA. This does not trigger an END event.
SUBSCRIBE_FLUSHRX	0x09C		Subscribe configuration for task FLUSHRX
SUBSCRIBE_DMA.RX.START	0x0A8		Subscribe configuration for task START
SUBSCRIBE_DMA.RX.STOP	0x0AC		Subscribe configuration for task STOP
SUBSCRIBE_DMA.RX.ENABLEMATCH[n]	0x0B0		Subscribe configuration for task ENABLEMATCH[n]
SUBSCRIBE_DMA.RX.DISABLEMATCH[n]	0x0C0		Subscribe configuration for task DISABLEMATCH[n]
SUBSCRIBE_DMA.TX.START	0x0D0		Subscribe configuration for task START
SUBSCRIBE_DMA.TX.STOP	0x0D4		Subscribe configuration for task STOP
EVENTS_CTS	0x100		CTS is activated (set low). Clear To Send.
EVENTS_NCTS	0x104		CTS is deactivated (set high). Not Clear To Send.
EVENTS_TXDRDY	0x10C		Data sent from TXD
EVENTS_RXDRDY	0x110		Data received in RXD (but potentially not yet transferred to Data RAM)
EVENTS_ERROR	0x114		Error detected
EVENTS_RXTO	0x124		Receiver timeout
EVENTS_TXSTOPPED	0x130		Transmitter stopped
EVENTS_DMA.RX.END	0x14C		Generated after all MAXCNT bytes have been transferred
EVENTS_DMA.RX.READY	0x150		Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.
EVENTS_DMA.RX.BUSERROR	0x154		An error occurred during the bus transfer.
EVENTS_DMA.RX.MATCH[n]	0x158		Pattern match is detected on the DMA data bus.
EVENTS_DMA.TX.END	0x168		Generated after all MAXCNT bytes have been transferred
EVENTS_DMA.TX.READY	0x16C		Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.
EVENTS_DMA.TX.BUSERROR	0x170		An error occurred during the bus transfer.
EVENTS_FRAMETIMEOUT	0x174		Timed out due to bus being idle while receiving data.
PUBLISH_CTS	0x180		Publish configuration for event CTS
PUBLISH_NCTS	0x184		Publish configuration for event NCTS
PUBLISH_TXDRDY	0x18C		Publish configuration for event TXDRDY
PUBLISH_RXDRDY	0x190		Publish configuration for event RXDRDY
PUBLISH_ERROR	0x194		Publish configuration for event ERROR
PUBLISH_RXTO	0x1A4		Publish configuration for event RXTO
PUBLISH_TXSTOPPED	0x1B0		Publish configuration for event TXSTOPPED
PUBLISH_DMA.RX.END	0x1CC		Publish configuration for event END
PUBLISH_DMA.RX.READY	0x1D0		Publish configuration for event READY
PUBLISH_DMA.RX.BUSERROR	0x1D4		Publish configuration for event BUSERROR
PUBLISH_DMA.RX.MATCH[n]	0x1D8		Publish configuration for event MATCH[n]
PUBLISH_DMA.TX.END	0x1E8		Publish configuration for event END
PUBLISH_DMA.TX.READY	0x1EC		Publish configuration for event READY
PUBLISH_DMA.TX.BUSERROR	0x1F0		Publish configuration for event BUSERROR
PUBLISH_FRAMETIMEOUT	0x1F4		Publish configuration for event FRAMETIMEOUT
SHORTS	0x200		Shortcuts between local events and tasks
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt

Register	Offset	TZ	Description
INTENCLR	0x308		Disable interrupt
ERRORSRC	0x480		Error source
ENABLE	0x500		Enable UART
BAUDRATE	0x524		Baud rate. Accuracy depends on the HFCLK source selected.
CONFIG	0x56C		Configuration of parity, hardware flow control, framesize, and packet timeout.
ADDRESS	0x574		Set the address of the UARTE for RX when used in 9 bit data frame mode.
FRAMETIMEOUT	0x578		Set the number of UARTE bits to count before triggering packet timeout.
PSEL.TXD	0x604		Pin select for TXD signal
PSEL.CTS	0x608		Pin select for CTS signal
PSEL.RXD	0x60C		Pin select for RXD signal
PSEL.RTS	0x610		Pin select for RTS signal
DMA.RX.PTR	0x704		RAM buffer start address
DMA.RX.MAXCNT	0x708		Maximum number of bytes in channel buffer
DMA.RX.AMOUNT	0x70C		Number of bytes transferred in the last transaction, updated after the END event. Also updated after each MATCH event.
DMA.RX.TERMINATEONBUSERROR	0x71C		Terminate the transaction if a BUSERROR event is detected.
DMA.RX.BUSERRORADDRESS	0x720		Address of transaction that generated the last BUSERROR event.
DMA.RX.MATCH.CONFIG	0x724		Configure individual match events
DMA.RX.MATCH.CANDIDATE[n]	0x728		The data to look for - any match will trigger the MATCH[n] event, if enabled.
DMA.TX.PTR	0x73C		RAM buffer start address
DMA.TX.MAXCNT	0x740		Maximum number of bytes in channel buffer
DMA.TX.AMOUNT	0x744		Number of bytes transferred in the last transaction, updated after the END event. Also updated after each MATCH event.
DMA.TX.TERMINATEONBUSERROR	0x754		Terminate the transaction if a BUSERROR event is detected.
DMA.TX.BUSERRORADDRESS	0x758		Address of transaction that generated the last BUSERROR event.

8.25.12.1 TASKS_FLUSHRX

Address offset: 0x01C

Flush RX FIFO into RX buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_FLUSHRX						Flush RX FIFO into RX buffer																											
			Trigger	1				Trigger task																											

8.25.12.2 TASKS_DMA

Peripheral tasks.

8.25.12.2.1 TASKS_DMA.RX

Peripheral tasks.

8.25.12.2.1.1 TASKS_DMA.RX.START

Address offset: 0x028

Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	START						Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.																											
			Trigger	1				Trigger task																											

8.25.12.2.1.2 TASKS_DMA.RX.STOP

Address offset: 0x02C

Stops operation using easyDMA. This does not trigger an END event.

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																								A			
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID		Value					Description																																	
A	W	STOP								Stops operation using easyDMA. This does not trigger an END event.																																	
			Trigger	1	Trigger task																																						

8.25.12.2.1.3 TASKS_DMA.RX.ENABLEMATCH[n] (n=0..3)

Address offset: 0x030 + (n × 0x4)

Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																							A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value	Description																																		
A	W	ENABLEMATCH			Enables the MATCH[n] event by setting the ENABLE[n] bit in the CONFIG register.																																		
			Trigger	1	Trigger task																																		

8.25.12.2.1.4 TASKS_DMA.RX.DISABLEMATCH[n] (n=0..3)

Address offset: 0x040 + (n × 0x4)

Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A																															
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																															
A	W	DISABLEMATCH			Disables the MATCH[n] event by clearing the ENABLE[n] bit in the CONFIG register.																															
			Trigger	1	Trigger task																															

8.25.12.2.2 TASKS_DMA.TX

Peripheral tasks.

8.25.12.2.2.1 TASKS_DMA.TX.START

Address offset: 0x050

Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	START						Starts operation using easyDMA to load the values. See peripheral description for operation using easyDMA.																											
			Trigger	1				Trigger task																											

8.25.12.2.2 TASKS_DMA.TX.STOP

Address offset: 0x054

Stops operation using easyDMA. This does not trigger an END event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	STOP						Stops operation using easyDMA. This does not trigger an END event.																											
			Trigger	1				Trigger task																											

8.25.12.3 SUBSCRIBE_FLUSHRX

Address offset: 0x09C

Subscribe configuration for task [FLUSHRX](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that task FLUSHRX will subscribe to																																
B	RW	EN																																				
			Disabled	0	Disable subscription																																	
			Enabled	1	Enable subscription																																	

8.25.12.4 SUBSCRIBE_DMA

Subscribe configuration for tasks

8.25.12.4.1 SUBSCRIBE_DMA.RX

Subscribe configuration for tasks

8.25.12.4.1.1 SUBSCRIBE_DMA.RX.START

Address offset: 0x0A8

Subscribe configuration for task [START](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task START will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.25.12.4.1.2 SUBSCRIBE_DMA.RX.STOP

Address offset: 0x0AC

Subscribe configuration for task **STOP**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task STOP will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.25.12.4.1.3 SUBSCRIBE_DMA.RX.ENABLEMATCH[n] (n=0..3)

Address offset: 0x0B0 + (n × 0x4)

Subscribe configuration for task **ENABLEMATCH[n]**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task ENABLEMATCH[n] will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.25.12.4.1.4 SUBSCRIBE_DMA.RX.DISABLEMATCH[n] (n=0..3)

Address offset: 0x0C0 + (n × 0x4)

Subscribe configuration for task **DISABLEMATCH[n]**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task DISABLEMATCH[n] will subscribe to																											
B	RW	EN																																	
			Disabled	0				Disable subscription																											
			Enabled	1				Enable subscription																											

8.25.12.4.2 SUBSCRIBE_DMA.TX

Subscribe configuration for tasks

8.25.12.4.2.1 SUBSCRIBE_DMA.TX.START

Address offset: 0x0D0

Subscribe configuration for task [START](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				B																								A				A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value				Description																													
A	RW	CHIDX		[0..255]				DPPI channel that task START will subscribe to																													
B	RW	EN																																			
			Disabled	0				Disable subscription																													
			Enabled	1				Enable subscription																													

8.25.12.4.2.2 SUBSCRIBE_DMA.TX.STOP

Address offset: 0x0D4

Subscribe configuration for task [STOP](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that task STOP will subscribe to																																
B	RW	EN																																				
			Disabled	0	Disable subscription																																	
			Enabled	1	Enable subscription																																	

8.25.12.5 EVENTS_CTS

Address offset: 0x100

CTS is activated (set low). Clear To Send.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_CTS			CTS is activated (set low). Clear To Send.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.25.12.6 EVENTS_NCTS

Address offset: 0x104

CTS is deactivated (set high). Not Clear To Send.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_NCTS			CTS is deactivated (set high). Not Clear To Send.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.25.12.7 EVENTS_TXDRDY

Address offset: 0x10C

Data sent from TXD

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_TXDRDY			Data sent from TXD																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.25.12.8 EVENTS_RXDRDY

Address offset: 0x110

Data received in RXD (but potentially not yet transferred to Data RAM)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																				A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	EVENTS_RXDRDY				Data received in RXD (but potentially not yet transferred to Data RAM)																															
			NotGenerated	0	Event not generated																																
			Generated	1	Event generated																																

8.25.12.9 EVENTS_ERROR

Address offset: 0x114

Error detected

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_ERROR			Error detected																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.25.12.10 EVENTS_RXTO

Address offset: 0x124

Receiver timeout

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_RXT0			Receiver timeout																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.25.12.11 EVENTS_TXSTOPPED

Address offset: 0x130

Transmitter stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_TXSTOPPED			Transmitter stopped																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.25.12.12 EVENTS_DMA

Peripheral events.

8.25.12.12.1 EVENTS_DMA.RX

Peripheral events.

8.25.12.12.1.1 EVENTS_DMA.RX.END

Address offset: 0x14C

Generated after all MAXCNT bytes have been transferred

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	ValueDescription																															
A	RW	END		Generated after all MAXCNT bytes have been transferred																															
			NotGenerated	0Event not generated																															
			Generated	1Event generated																															

8.25.12.12.1.2 EVENTS_DMA.RX.READY

Address offset: 0x150

Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.25.12.12.1.3 EVENTS_DMA.RX.BUSERROR

Address offset: 0x154

An error occurred during the bus transfer.

When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	BUSERROR			An error occurred during the bus transfer.																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.25.12.12.1.4 EVENTS_DMA.RX.MATCH[n] (n=0..3)

Address offset: 0x158 + (n × 0x4)

Pattern match is detected on the DMA data bus.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	MATCH			Pattern match is detected on the DMA data bus.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.25.12.12.2 EVENTS_DMA.TX

Peripheral events.

8.25.12.12.2.1 EVENTS_DMA.TX.END

Address offset: 0x168

Generated after all MAXCNT bytes have been transferred

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	END			Generated after all MAXCNT bytes have been transferred																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.25.12.12.2.2 EVENTS_DMA.TX.READY

Address offset: 0x16C

Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	READY			Generated when EasyDMA has buffered the .PTR and .MAXCNT registers for the channel, allowing them to be written to prepare for the next sequence.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.25.12.12.2.3 EVENTS_DMA.TX.BUSERROR

Address offset: 0x170

An error occurred during the bus transfer.

When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	BUSERROR			An error occurred during the bus transfer.																														
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.25.12.13 EVENTS_FRAMETIMEOUT

Address offset: 0x174

Timed out due to bus being idle while receiving data.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENTS_FRAMETIMEOUT						Timed out due to bus being idle while receiving data.																											
			NotGenerated	0				Event not generated																											
			Generated	1				Event generated																											

8.25.12.14 PUBLISH_CTS

Address offset: 0x180

Publish configuration for event **CTS**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event CTS will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.25.12.15 PUBLISH_NCTS

Address offset: 0x184

Publish configuration for event **NCTS**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event NCTS will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.25.12.16 PUBLISH_TXDRDY

Address offset: 0x18C

Publish configuration for event **TXDRDY**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event TXDRDY will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.25.12.17 PUBLISH_RXDRDY

Address offset: 0x190

Publish configuration for event **RXDRDY**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0																																		
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that event RXDRDY will publish to																																
B	RW	EN																																				
			Disabled	0	Disable publishing																																	
			Enabled	1	Enable publishing																																	

8.25.12.18 PUBLISH_ERROR

Address offset: 0x194

Publish configuration for event **ERROR**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0																																		
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that event ERROR will publish to																																
B	RW	EN																																				
			Disabled	0	Disable publishing																																	
			Enabled	1	Enable publishing																																	

8.25.12.19 PUBLISH_RXTO

Address offset: 0x1A4

Publish configuration for event **RXTO**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				B																												A	A	A	A	A	A	A	A
Reset 0x00000000				0																																			
ID	R/W	Field	Value ID	Value				Description																															
A	RW	CHIDX		[0..255]				DPPI channel that event RXTO will publish to																															
B	RW	EN																																					
			Disabled	0				Disable publishing																															
			Enabled	1				Enable publishing																															

8.25.12.20 PUBLISH_TXSTOPPED

Address offset: 0x1B0

Publish configuration for event **TXSTOPPED**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0																																		
ID	R/W	Field	Value ID	Value		Description																																
A	RW	CHIDX		[0..255]		DPPI channel that event TXSTOPPED will publish to																																
B	RW	EN																																				
			Disabled	0	Disable publishing																																	
			Enabled	1	Enable publishing																																	

8.25.12.21 PUBLISH_DMA

Publish configuration for events

8.25.12.21.1 PUBLISH_DMA.RX

Publish configuration for events

8.25.12.21.1.1 PUBLISH_DMA.RX.END

Address offset: 0x1CC

Publish configuration for event [END](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																													
ID				B																								A										A	A	A	A	A	A	A																				
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																								
A	RW	CHIDX		[0..255]				DPPI channel that event END will publish to																																																								
B	RW	EN																																																														
			Disabled	0				Disable publishing																																																								
			Enabled	1				Enable publishing																																																								

8.25.12.21.1.2 PUBLISH_DMA.RX.READY

Address offset: 0x1D0

Publish configuration for event [READY](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																													
ID				B																								A												A	A	A	A	A	A	A																		
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																								
A	RW	CHIDX		[0..255]				DPPI channel that event READY will publish to																																																								
B	RW	EN																																																														
			Disabled	0				Disable publishing																																																								
			Enabled	1				Enable publishing																																																								

8.25.12.21.1.3 PUBLISH_DMA.RX.BUSERROR

Address offset: 0x1D4

Publish configuration for event [BUSERROR](#)

When this event is generated, the address which caused the error can be read from the [BUSERRORADDRESS](#) register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															

8.25.12.21.1.4 PUBLISH_DMA.RX.MATCH[n] (n=0..3)

Address offset: 0x1D8 + (n × 0x4)

Publish configuration for event **MATCH[n]**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0						
ID				B																								A				A	A	A	A	A	A				
Reset 0x00000000				0																																					
ID	R/W	Field	Value ID	Value		Description																																			
A	RW	CHIDX		[0..255]		DPPI channel that event MATCH[n] will publish to																																			
B	RW	EN																																							
			Disabled	0	Disable publishing																																				
			Enabled	1	Enable publishing																																				

8.25.12.21.2 PUBLISH_DMA.TX

Publish configuration for events

8.25.12.21.2.1 PUBLISH_DMA.TX.END

Address offset: 0x1E8

Publish configuration for event **END**

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																											
ID				B																								A				A	A	A	A	A	A	A																								
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																								
A	RW	CHIDX		[0..255]		DPPI channel that event END will publish to																																																								
B	RW	EN																																																												
			Disabled	0	Disable publishing																																																									
			Enabled	1	Enable publishing																																																									

8.25.12.21.2.2 PUBLISH_DMA.TX.READY

Address offset: 0x1EC

Publish configuration for event **READY**

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event READY will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.25.12.21.2.3 PUBLISH_DMA.TX.BUSERERROR

Address offset: 0x1F0

Publish configuration for event **BUSERERROR**

When this event is generated, the address which caused the error can be read from the **BUSERERRORADDRESS** register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	CHIDX		[0..255]		DPPI channel that event BUSERROR will publish to																													
B	RW	EN																																	
			Disabled	0	Disable publishing																														
			Enabled	1	Enable publishing																														

8.25.12.22 PUBLISH_FRAMETIMEOUT

Address offset: 0x1F4

Publish configuration for event [FRAMETIMEOUT](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															

8.25.12.23 SHORTS

Address offset: 0x200

Shortcuts between local events and tasks

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				L K J I H G F E D C																B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	DMA_RX_END_DMA_RX_START			Shortcut between event DMA.RX.END and task DMA.RX.START																														
		Disabled	0	Disable shortcut																															
		Enabled	1	Enable shortcut																															
B	RW	DMA_RX_END_DMA_RX_STOP			Shortcut between event DMA.RX.END and task DMA.RX.STOP																														
		Disabled	0	Disable shortcut																															
		Enabled	1	Enable shortcut																															
C	RW	DMA_TX_END_DMA_TX_STOP			Shortcut between event DMA.TX.END and task DMA.TX.STOP																														
		Disabled	0	Disable shortcut																															
		Enabled	1	Enable shortcut																															
D-G	RW	DMA_RX_MATCH[i]_DMA_RX_ENABLEMA +1)%4] (i=0..3)			Shortcut between event DMA.RX.MATCH[n] and task DMA.RX.ENABLEMATCH[(i+1)%4]																														
				Allows daisy-chaining match events.																															
		Disabled	0	Disable shortcut																															
		Enabled	1	Enable shortcut																															
H-K	RW	DMA_RX_MATCH[i]_DMA_RX_DISABLEMATCH[i] (i=0..3)			Shortcut between event DMA.RX.MATCH[n] and task DMA.RX.DISABLEMATCH[n]																														
		Disabled	0	Disable shortcut																															
		Enabled	1	Enable shortcut																															
L	RW	FRAMETIMEOUT_DMA_RX_STOP			Shortcut between event FRAMETIMEOUT and task DMA.RX.STOP																														
		Disabled	0	Disable shortcut																															

8.25.12.24 INTEN

Enable or disable interrupt

4503_018 v0.8

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				R										Q	P	O	N	M	L	K	J	I	H	G					F		E			D	C	B		A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																	
P	RW	DMATXREADY			Enable or disable interrupt for event DMATXREADY																																	
			Disabled	0	Disable																																	
			Enabled	1	Enable																																	
Q	RW	DMATXBUSERROR			Enable or disable interrupt for event DMATXBUSERROR																																	
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																																	
			Disabled	0	Disable																																	
			Enabled	1	Enable																																	
R	RW	FRAMETIMEOUT			Enable or disable interrupt for event FRAMETIMEOUT																																	
			Disabled	0	Disable																																	
			Enabled	1	Enable																																	

8.25.12.25 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				R Q P O N M L K J I H																G				F				E D C				B A			
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CTS			Write '1' to enable interrupt for event CTS																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	NCTS			Write '1' to enable interrupt for event NCTS																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
C	RW	TXDRDY			Write '1' to enable interrupt for event TXDRDY																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
D	RW	RXDRDY			Write '1' to enable interrupt for event RXDRDY																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
E	RW	ERROR			Write '1' to enable interrupt for event ERROR																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
F	RW	RXTO			Write '1' to enable interrupt for event RXTO																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
G	RW	TXSTOPPED			Write '1' to enable interrupt for event TXSTOPPED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			R Q P O N M L K J I H																G				F				E D C				B A			
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
H	RW	DMARXEND			Write '1' to enable interrupt for event DMARXEND																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
I	RW	DMARXREADY			Write '1' to enable interrupt for event DMARXREADY																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
J	RW	DMARXBUSERERROR			Write '1' to enable interrupt for event DMARXBUSERERROR																													
					When this event is generated, the address which caused the error can be read from the BUSERERRORADDRESS register.																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
		Enabled	1	Read: Enabled																														
K-N	RW	DMARXMATCH[i] (i=0..3)			Write '1' to enable interrupt for event DMARXMATCH[i]																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
O	RW	DMATXEND			Write '1' to enable interrupt for event DMATXEND																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
P	RW	DMATXREADY			Write '1' to enable interrupt for event DMATXREADY																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													
Q	RW	DMATXBUSERERROR			Write '1' to enable interrupt for event DMATXBUSERERROR																													
					When this event is generated, the address which caused the error can be read from the BUSERERRORADDRESS register.																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
		Enabled	1	Read: Enabled																														
R	RW	FRAMETIMEOUT			Write '1' to enable interrupt for event FRAMETIMEOUT																													
			Set	1	Enable																													
			Disabled	0	Read: Disabled																													
			Enabled	1	Read: Enabled																													

8.25.12.26 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																	
ID				R Q P O N M L K J I H																		G				F				E D C				B A			
Reset 0x00000000				0 0																																	
ID	R/W	Field	Value ID	Value				Description																													
A	RW	CTS						Write '1' to disable interrupt for event CTS																													
			Clear	1				Disable																													
			Disabled	0				Read: Disabled																													

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			R Q P O N M L K J I H																G				F		E D C				B A					
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
B	RW	NCTS	Enabled	1	Read: Enabled																													
			Clear	1	Write '1' to disable interrupt for event NCTS																													
			Disabled	0	Disable																													
			Enabled	1	Read: Disabled																													
C	RW	TXDRDY	Enabled	1	Read: Enabled																													
			Clear	1	Write '1' to disable interrupt for event TXDRDY																													
			Disabled	0	Disable																													
			Enabled	1	Read: Disabled																													
D	RW	RXDRDY	Enabled	1	Read: Enabled																													
			Clear	1	Write '1' to disable interrupt for event RXDRDY																													
			Disabled	0	Disable																													
			Enabled	1	Read: Disabled																													
E	RW	ERROR	Enabled	1	Read: Enabled																													
			Clear	1	Write '1' to disable interrupt for event ERROR																													
			Disabled	0	Disable																													
			Enabled	1	Read: Disabled																													
F	RW	RXTO	Enabled	1	Read: Enabled																													
			Clear	1	Write '1' to disable interrupt for event RXTO																													
			Disabled	0	Disable																													
			Enabled	1	Read: Disabled																													
G	RW	TXSTOPPED	Enabled	1	Read: Enabled																													
			Clear	1	Write '1' to disable interrupt for event TXSTOPPED																													
			Disabled	0	Disable																													
			Enabled	1	Read: Disabled																													
H	RW	DMARXEND	Enabled	1	Read: Enabled																													
			Clear	1	Write '1' to disable interrupt for event DMARXEND																													
			Disabled	0	Disable																													
			Enabled	1	Read: Disabled																													
I	RW	DMARXREADY	Enabled	1	Read: Enabled																													
			Clear	1	Write '1' to disable interrupt for event DMARXREADY																													
			Disabled	0	Disable																													
			Enabled	1	Read: Disabled																													
J	RW	DMARXBUSERERROR	Enabled	1	Read: Enabled																													
			Clear	1	Write '1' to disable interrupt for event DMARXBUSERERROR																													
			Disabled	0	When this event is generated, the address which caused the error can be read from the BUSERERRORADDRESS register.																													
			Enabled	1	Read: Disabled																													
K-N	RW	DMARXMATCH[i] (i=0..3)	Enabled	1	Read: Enabled																													
			Clear	1	Write '1' to disable interrupt for event DMARXMATCH[i]																													
			Disabled	0	Disable																													
			Enabled	1	Read: Disabled																													
O	RW	DMATXEND	Enabled	1	Read: Enabled																													
			Clear	1	Write '1' to disable interrupt for event DMATXEND																													
			Disabled	0	Disable																													
			Enabled	1	Read: Disabled																													
P	RW	DMATXREADY	Enabled	1	Read: Enabled																													
			Clear	1	Write '1' to disable interrupt for event DMATXREADY																													
			Disabled	0	Disable																													

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID		R Q P O N M L K J I H										G F E D C B A																					
Reset 0x00000000		0 0																															
ID	R/W	Field	Value ID	Value	Description																												
Q	RW	DMATXBUSERROR	Enabled	1	Read: Enabled																												
					Write '1' to disable interrupt for event DMATXBUSERROR																												
					When this event is generated, the address which caused the error can be read from the BUSERRORADDRESS register.																												
			Clear	1	Disable																												
			Disabled	0	Read: Disabled																												
R	RW	FRAMETIMEOUT	Enabled	1	Read: Enabled																												
					Write '1' to disable interrupt for event FRAMETIMEOUT																												
			Clear	1	Disable																												
			Disabled	0	Read: Disabled																												
			Enabled	1	Read: Enabled																												

8.25.12.27 ERRORSRC

Address offset: 0x480

Error source

This register is read/write one to clear.

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			D C B A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	RW	OVERRUN W1C			Overrun error																													
					A start bit is received while the previous data still lies in RXD. (Previous data is lost.)																													
			NotPresent	0	Read: error not present																													
			Present	1	Read: error present																													
B	RW	PARITY W1C			Parity error																													
					A character with bad parity is received, if HW parity check is enabled.																													
			NotPresent	0	Read: error not present																													
			Present	1	Read: error present																													
C	RW	FRAMING W1C			Framing error occurred																													
					A valid stop bit is not detected on the serial data input after all bits in a character have been received.																													
			NotPresent	0	Read: error not present																													
			Present	1	Read: error present																													
D	RW	BREAK W1C			Break condition																													
					The serial data input is '0' for longer than the length of a data frame. (The data frame length is 10 bits without parity bit and 11 bits with parity bit.)																													
			NotPresent	0	Read: error not present																													
			Present	1	Read: error present																													

8.25.12.28 ENABLE

Address offset: 0x500

Enable UART

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	ENABLE				Enable or disable UARTE																													
			Disabled	0		Disable UARTE																													
			Enabled	8		Enable UARTE																													

8.25.12.29 BAUDRATE

Address offset: 0x524

Baud rate. Accuracy depends on the HFCLK source selected.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x04000000				0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																														
A	RW	BAUDRATE			Baud rate																														
			Baud1200	0x0004F000	1200 baud (actual rate: 1205) when UARTE has 16 MHz peripheral clock frequency																														
			Baud2400	0x0009D000	2400 baud (actual rate: 2396) when UARTE has 16 MHz peripheral clock frequency																														
			Baud4800	0x0013B000	4800 baud (actual rate: 4808) when UARTE has 16 MHz peripheral clock frequency																														
			Baud9600	0x00275000	9600 baud (actual rate: 9598) when UARTE has 16 MHz peripheral clock frequency																														
			Baud14400	0x003AF000	14400 baud (actual rate: 14401) when UARTE has 16 MHz peripheral clock frequency																														
			Baud19200	0x004EA000	19200 baud (actual rate: 19208) when UARTE has 16 MHz peripheral clock frequency																														
			Baud28800	0x0075C000	28800 baud (actual rate: 28777) when UARTE has 16 MHz peripheral clock frequency																														
			Baud31250	0x00800000	31250 baud when UARTE has 16 MHz peripheral clock frequency																														
			Baud38400	0x009D0000	38400 baud (actual rate: 38369) when UARTE has 16 MHz peripheral clock frequency																														
			Baud56000	0x00E50000	56000 baud (actual rate: 55944) when UARTE has 16 MHz peripheral clock frequency																														
			Baud57600	0x00EB0000	57600 baud (actual rate: 57554) when UARTE has 16 MHz peripheral clock frequency																														
			Baud76800	0x013A9000	76800 baud (actual rate: 76923) when UARTE has 16 MHz peripheral clock frequency																														
			Baud115200	0x01D60000	115200 baud (actual rate: 115108) when UARTE has 16 MHz peripheral clock frequency																														
			Baud230400	0x03B00000	230400 baud (actual rate: 231884) when UARTE has 16 MHz peripheral clock frequency																														
			Baud250000	0x04000000	250000 baud when UARTE has 16 MHz peripheral clock frequency																														
			Baud460800	0x07400000	460800 baud (actual rate: 457143) when UARTE has 16 MHz peripheral clock frequency																														
			Baud921600	0x0F000000	921600 baud (actual rate: 941176) when UARTE has 16 MHz peripheral clock frequency																														
			Baud1M	0x10000000	1 megabaud when UARTE has 16 MHz peripheral clock frequency																														

Configuration of parity, hardware flow control, framesize, and packet timeout.

8.25.12.31 ADDRESS

Set the address of the UARTE for RX when used in 9 bit data frame mode.

8.25.12.32 FRAMETIMEOUT

Set the number of UARTE bits to count before triggering packet timeout.

Bit number	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																								A	A	A	A	A	A	A	A	A
Reset 0x00000010	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
ID	R/W	Field	Value ID	Value	Description																											
A	RW	COUNTERTOP			Number of UARTE bits before timeout.																											

8.25.12.33 PSEL.TXD

Address offset: 0x604

Pin select for TXD signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				C																								B				B	B	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			
ID	R/W	Field	Value ID	Value				Description																														
A	RW	PIN		[0..31]				Pin number																														
B	RW	PORT		[0..7]				Port number																														
C	RW	CONNECT						Connection																														
			Disconnected	1				Disconnect																														
			Connected	0				Connect																														

8.25.12.34 PSEL.CTS

Address offset: 0x608

Pin select for CTS signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				C																								B				B	B	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			
ID	R/W	Field	Value ID	Value				Description																														
A	RW	PIN		[0..31]				Pin number																														
B	RW	PORT		[0..7]				Port number																														
C	RW	CONNECT						Connection																														
			Disconnected	1				Disconnect																														
			Connected	0				Connect																														

8.25.12.35 PSEL.RXD

Address offset: 0x60C

Pin select for RXD signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				C																								B				B	B	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			
ID	R/W	Field	Value ID	Value		Description																																
A	RW	PIN		[0..31]		Pin number																																
B	RW	PORT		[0..7]		Port number																																
C	RW	CONNECT				Connection																																
			Disconnected	1		Disconnect																																
			Connected	0		Connect																																

8.25.12.36 PSEL.RTS

Address offset: 0x610

Pin select for RTS signal

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID				C																												B	B	B	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1				
ID	R/W	Field	Value ID	Value				Description																															
A	RW	PIN		[0..31]				Pin number																															
B	RW	PORT		[0..7]				Port number																															
C	RW	CONNECT						Connection																															
			Disconnected	1				Disconnect																															
			Connected	0				Connect																															

8.25.12.37 DMA.RX.PTR

Address offset: 0x704

RAM buffer start address

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x20000000				0 0 1 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	PTR						RAM buffer start address for this EasyDMA channel. This address is a word aligned Data RAM address.																											

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.25.12.38 DMA.RX.MAXCNT

Address offset: 0x708

Maximum number of bytes in channel buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																				A A A A A A A A A A A A A A A A A A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW	MAXCNT		[1..0xffff]								Maximum number of bytes in channel buffer																							

8.25.12.39 DMA.RX.AMOUNT

Address offset: 0x70C

Number of bytes transferred in the last transaction, updated after the END event.

Also updated after each MATCH event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	AMOUNT		[1..0xffff]				Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.																											

8.25.12.40 DMA.RX.TERMINATEONBUSERROR

Address offset: 0x71C

Terminate the transaction if a BUSERROR event is detected.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ENABLE																																	
			Disabled	0				Disable																											
			Enabled	1				Enable																											

8.25.12.41 DMA.RX.BUSERRORADDRESS

Address offset: 0x720

Address of transaction that generated the last BUSERROR event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	ADDRESS																																	

8.25.12.42 DMA.RX.MATCH

Registers to control the behavior of the pattern matcher engine

8.25.12.42.1 DMA.RX.MATCH.CONFIG

Address offset: 0x724

Configure individual match events

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				H G F E																								D C B A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-D	RW	ENABLE[i] (i=0..3)			Enable match filter i																														
			Disabled	0	Match filter disabled																														
			Enabled	1	Match filter enabled																														
E-H	RW	ONESHOT[i] (i=0..3)			Configure match filter i as one-shot or continuous																														
					One-shot operation will disable the filter on a match, while Continuous operation will keep it enabled until explicitly disabled.																														
			Continuous	0	Match filter stays enabled until disabled by task																														
			Oneshot	1	Match filter stays enabled until next data word is received																														

8.25.12.42.2 DMA.RX.MATCH.CANDIDATE[n] (n=0..3)

Address offset: 0x728 + (n × 0x4)

The data to look for - any match will trigger the MATCH[n] event, if enabled.

Note: This register can be updated while a transfer is in progress, but the new value will not take effect until either the DMA is restarted or the match event is generated. That makes it possible to write a new set of match words which will be searched for immediately after the event triggers.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW	DATA										Data to look for																							

8.25.12.43 DMA.TX.PTR

Address offset: 0x73C

RAM buffer start address

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A																															
Reset 0x20000000					0 0 1 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	PTR			RAM buffer start address for this EasyDMA channel. This address is a word aligned Data RAM address.																															

Note: See the memory chapter for details about which memories are available for EasyDMA.

8.25.12.44 DMA.TX.MAXCNT

Address offset: 0x740

Maximum number of bytes in channel buffer

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	RW	MAXCNT		[1..0xffff]																Maximum number of bytes in channel buffer															

8.25.12.45 DMA.TX.AMOUNT

Address offset: 0x744

Number of bytes transferred in the last transaction, updated after the END event.

Also updated after each MATCH event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	AMOUNT		[1..0xffff]				Number of bytes transferred in the last transaction. In case of NACK error, includes the NACK'ed byte.																											

8.25.12.46 DMA.TX.TERMINATEONBUSERROR

Address offset: 0x754

Terminate the transaction if a BUSERROR event is detected.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				A																																
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																												
A	RW	ENABLE																																		
			Disabled	0				Disable																												
			Enabled	1				Enable																												

8.25.12.47 DMA.TX.BUSERRORADDRESS

Address offset: 0x758

Address of transaction that generated the last BUSERROR event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	ADDRESS																																	

8.26 VPR — RISC-V CPU

VPR is a small, efficient CPU developed by Nordic Semiconductor.

VPR is compatible with the RISC-V instruction set and implements the following extensions:

- E – Integer instruction set with 16 registers
- M – Multiply and divide extension
- C – Compressed extension (compressed instructions)

VPR implements the machine mode CPU mode as well as the RISC-V CLIC specification for the interrupt controller.

VPR is optimized for implementing software-defined peripheral functions. The software-defined peripheral running on VPR does not start on its own, but must be started by the application core processor by performing the following steps:

1. Configure VPR program counter (PC) to point to the peripheral binary image by using register [INITPC](#) on page 738.
2. Start VPR by using register [CPURUN](#) on page 738.

8.26.1 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
VPR00 : S	GLOBAL	0x5004C000	US	NS	NSA	No	FLPR - VPR peripheral registers
VPR00 : NS		0x4004C000					

Configuration

Instance	Domain	Configuration
VPR00 : S VPR00 : NS	GLOBAL	Boot vector (INIT_PC_RESET_VALUE): 0x00000000
		Self-booting (VPR_START_RESET_VALUE): 0
		VPR RAM base address (RAM_BASE_ADDR): 0x20000000
		VPR RAM size (RAM_SZ): 20 (Value in bytes is computed as 2^(RAM size))
		Retain registers in Deep Sleep mode: 0
		Restore VPR context at VPR reset using register [NRF_MEMCONF->POWER1.RET].MEM[0]
		VPR context save address: 0x2003FE00
		VPR remap address: 0x00000000
		VEVIF tasks: 16..22
		Mask of supported VEVIF tasks: 0x007F0000
		VEVIF DPPI channels: 0..3
		Mask of supported VEVIF DPPI channels: 0x000F0000
		VEVIF events: 16..22
		Mask of supported VEVIF events: 0x00100000
		Debugger interface register offset: 0x5004C400

Register overview

Register	Offset	TZ	Description
TASKS_TRIGGER[n]	0x000		VPR task [n] register
SUBSCRIBE_TRIGGER[n]	0x080		Subscribe configuration for task TASKS_TRIGGER[n]
EVENTS_TRIGGERED[n]	0x100		VPR event [n] register
PUBLISH_TRIGGERED[n]	0x180		Publish configuration for event EVENTS_TRIGGERED[n]
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
INTPEND	0x30C		Pending interrupts
DEBUGIF.DATA0	0x410		Abstract Data 0. Read/write data for argument 0
DEBUGIF.DATA1	0x414		Abstract Data 1. Read/write data for argument 1
DEBUGIF.DMCONTROL	0x440		Debug Module Control
DEBUGIF.DMSTATUS	0x444		Debug Module Status
DEBUGIF.HARTINFO	0x448		Hart Information
DEBUGIF.HALTSUM1	0x44C		Halt Summary 1

Register	Offset	TZ	Description
DEBUGIF.HAWINDOWSEL	0x450		Hart Array Window Select
DEBUGIF.HAWINDOW	0x454		Hart Array Window
DEBUGIF.ABSTRACTCS	0x458		Abstract Control and Status
DEBUGIF.ABSTRACTCMD	0x45C		Abstract command
DEBUGIF.ABSTRACTAUTO	0x460		Abstract Command Autoexec
DEBUGIF.CONFSTRPTR[n]	0x464		Configuration String Pointer [n]
DEBUGIF.NEXTDM	0x474		Next Debug Module
DEBUGIF.PROGBUF[n]	0x480		Program Buffer [n]
DEBUGIF.AUTHDATA	0x4C0		Authentication Data
DEBUGIF.HALTSUM2	0x4D0		Halt Summary 2
DEBUGIF.HALTSUM3	0x4D4		Halt Summary 3
DEBUGIF.SBADDRESS3	0x4DC		System Bus Address 127:96
DEBUGIF.SBCS	0x4E0		System Bus Access Control and Status
DEBUGIF.SBADDRESS0	0x4E4		System Bus Address 31:0
DEBUGIF.SBADDRESS1	0x4E8		System Bus Address 63:32
DEBUGIF.SBADDRESS2	0x4EC		System Bus Address 95:64
DEBUGIF.SBDATA0	0x4F0		System Bus Data 31:0
DEBUGIF.SBDATA1	0x4F4		System Bus Data 63:32
DEBUGIF.SBDATA2	0x4F8		System Bus Data 95:64
DEBUGIF.SBDATA3	0x4FC		System Bus Data 127:96
DEBUGIF.HALTSUM0	0x500		Halt summary 0
CPURUN	0x800		State of the CPU after a core reset
INITPC	0x808		Initial value of the PC at CPU start.

8.26.1.1 TASKS_TRIGGER[n] (n=16..22)

Address offset: $0x000 + (n \times 0x4)$

VPR task [n] register

If RTPs are not enabled: writes to registers 15..0 are ignored, other read/write accesses operate normally

If RTPs are enabled: reads or writes to these registers will be stalled if there is a simultaneous VPR CSR write access

to any CSR register.

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			A																															
Reset 0x00000000			0 0																															
ID	R/W	Field	Value ID	Value	Description																													
A	W	TASKS_TRIGGER			VPR task [n] register																													
					If RTPs are not enabled: writes to registers 15..0 are ignored, other read/write accesses operate normally																													
					If RTPs are enabled: reads or writes to these registers will be stalled if there is a simultaneous VPR CSR write access																													
					to any CSR register.																													
			Trigger	1	Trigger task																													

8.26.1.2 SUBSCRIBE_TRIGGER[n] (n=0..3)

Address offset: $0x080 + (n \times 0x4)$

Subscribe configuration for task TASKS_TRIGGER[n]

If RTPs are not enabled: writes to registers 15..0 are ignored, other read/write accesses operate normally

If RTPs are enabled: reads or writes to these registers will be stalled if there is a simultaneous VPR CSR write access to any CSR register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EN						Subscription enable bit																											
								Its value depends on OR between bit 31 and bit 0 of previously written value																											
			Disabled	0x0				Disable subscription																											
			Enabled	0x1				Enable subscription																											

8.26.1.3 EVENTS_TRIGGERED[n] (n=16..22)

Address offset: $0x100 + (n \times 0x4)$

VPR event [n] register

If RTPs are not enabled: writes to registers 15..0 are ignored, other read/write accesses operate normally

If RTPs are enabled: reads or writes to these registers will be stalled if there is a simultaneous VPR CSR write access

to any CSR register.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value	ID	Value	Description																													
A	RW	EVENTS_TRIGGERED				VPR event [n] register																													
						If RTPs are not enabled: writes to registers 15..0 are ignored, other read/write accesses operate normally																													
						If RTPs are enabled: reads or writes to these registers will be stalled if there is a simultaneous VPR CSR write access																													
						to any CSR register.																													
			NotGenerated	0		Event not generated																													
			Generated	1		Event generated																													

8.26.1.4 PUBLISH_TRIGGERED[n] (n=0..3)

Address offset: $0x180 + (n \times 0x4)$

Publish configuration for event [EVENTS_TRIGGERED\[n\]](#)

If RTPs are not enabled: writes to registers 15..0 are ignored, other read/write accesses operate normally

If RTPs are enabled: reads or writes to these registers will be stalled if there is a simultaneous VPR CSR write access

to any CSR register.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EN						Publication enable bit																											
								Its value depends on OR between bit 31 and bit 0 of previously written value																											
			Disabled	0x0				Disable publishing																											
			Enabled	0x1				Enable publishing																											

8.26.1.5 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
	RW	TRIGGERED[i] (i=16..22)			Enable or disable interrupt for event TRIGGERED[i]																														
					If RTPs are not enabled: writes to registers 15..0 are ignored, other read/write accesses operate normally																														
					If RTPs are enabled: reads or writes to these registers will be stalled if there is a simultaneous VPR CSR write access																														
					to any CSR register.																														
			Disabled	0	Disable																														
			Enabled	1	Enable																														

8.26.1.6 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
	RW	TRIGGERED[i] (i=16..22)			Write '1' to enable interrupt for event TRIGGERED[i]																														
					If RTPs are not enabled: writes to registers 15..0 are ignored, other read/write accesses operate normally																														
					If RTPs are enabled: reads or writes to these registers will be stalled if there is a simultaneous VPR CSR write access																														
					to any CSR register.																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.26.1.7 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
	RW	TRIGGERED[i] (i=16..22)			Write '1' to disable interrupt for event TRIGGERED[i]																														
					If RTPs are not enabled: writes to registers 15..0 are ignored, other read/write accesses operate normally																														
					If RTPs are enabled: reads or writes to these registers will be stalled if there is a simultaneous VPR CSR write access																														
					to any CSR register.																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.26.1.8 INTPEND

Address offset: 0x30C

Pending interrupts

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
	R	TRIGGERED[i] (i=16..22)			Read pending status of interrupt for event TRIGGERED[i]																														
					If RTPs are not enabled: writes to registers 15..0 are ignored, other read/write accesses operate normally																														
					If RTPs are enabled: reads or writes to these registers will be stalled if there is a simultaneous VPR CSR write access																														
					to any CSR register.																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														

8.26.1.9 DEBUGIF.DATA0

Address offset: 0x410

Abstract Data 0. Read/write data for argument 0

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	DATA0		Abstract Data 0																															

8.26.1.10 DEBUGIF.DATA1

Address offset: 0x414

Abstract Data 1. Read/write data for argument 1

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	DATA1						Abstract Data 1																											

8.26.1.11 DEBUGIF.DMCONTROL

Address offset: 0x440

Debug Module Control

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				K J I H G F F F F F F F F F E E E E E E E E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	DMACTIVE			Reset signal for the debug module.																														
			Disabled	0	Reset the debug module itself																														
			Enabled	1	Normal operation																														
B	RW	NDMRESET			Reset signal output from the debug module to the system.																														
			Inactive	0	Reset inactive																														
			Active	1	Reset active																														
C	W	CLRRESETHALTREQ			Clear the halt on reset request.																														
			NoOperation	0	No operation when written 0.																														
			Clear	1	Clears the halt on reset request																														
D	W	SETRESETHALTREQ			Set the halt on reset request.																														
			NoOperation	0	No operation when written 0.																														
			Clear	1	Sets the halt on reset request																														
E	W	HARTSELHI			The high 10 bits of hartsel.																														
F	W	HARTSELLO			The low 10 bits of hartsel.																														
G	W	HASEL			Definition of currently selected harts.																														
			Single	0	Single hart selected.																														
			Multiple	1	Multiple harts selected																														
H	W	ACKHAVERESET			Clear the havereset.																														
			NoOperation	0	No operation when written 0.																														
			Clear	1	Clears the havereset for selected harts.																														
I	RW	HARTRESET			Reset harts.																														
			Deasserted	0	Reset de-asserted.																														
			Asserted	1	Reset asserted.																														
J	W	RESUMEREQ			Resume currently selected harts.																														
			NoOperation	0	No operation when written 0.																														
			Resumed	1	Currently selected harts resumed.																														
K	W	HALTREQ			Halt currently selected harts.																														
			Clear	0	Clears halt request bit for all currently selected harts.																														
			Halt	1	Currently selected harts halted.																														

8.26.1.12 DEBUGIF.DMSTATUS

Address offset: 0x444

Debug Module Status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				R																Q P O N M L K J I H G F E D C B A A A A															
Reset 0x00400082				0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	VERSION			Version of the debug module.																														
			NotPresent	0	Debug module not present.																														
			V011	1	There is a Debug Module and it conforms to version 0.11 of this specification.																														
			V013	2	There is a Debug Module and it conforms to version 0.13 of this specification.																														
			NonConform	15	There is a Debug Module but it does not conform to any available version of the spec.																														
B	R	CONFSTRPTRVALID			Configuration string.																														
			NotRelevant	0	The confstrptr0..confstrptr3 holds information which is not relevant to the configuration string.																														
			Address	1	The confstrptr0..confstrptr3 holds the address of the configuration string.																														
C	R	HASRESETHALTREQ			Halt-on-reset support status.																														
			No	0	Halt-on-reset is supported.																														
			Yes	1	Halt-on-reset is not supported.																														
D	R	AUTHBUSY			Authentication busy status.																														
			No	0	The authentication module is ready.																														
			Yes	1	The authentication module is busy.																														
E	R	AUTHENTICATED			Authentication status.																														
			No	0	Authentication required before using the debug module.																														
			Yes	1	Authentication passed.																														
F	R	ANYHALTED			Any currently selected harts halted status.																														
			No	0	None of the currently selected harts halted.																														
			Yes	1	Any of the currently selected harts halted.																														
G	R	ALLHALTED			All currently selected harts halted status.																														
			No	0	Not all of the currently selected harts halted.																														
			Yes	1	All of the currently selected harts halted.																														
H	R	ANYRUNNING			Any currently selected harts running status.																														
			No	0	None of the currently selected harts running.																														
			Yes	1	Any of the currently selected harts running.																														
I	R	ALLRUNNING			All currently selected harts running status.																														
			No	0	Not all of the currently selected harts running.																														
			Yes	1	All of the currently selected harts running.																														
J	R	ANYUNAVAIL			Any currently selected harts unavailable status.																														
			No	0	None of the currently selected harts unavailable.																														
			Yes	1	Any of the currently selected harts unavailable.																														
K	R	ALLUNAVAIL			All currently selected harts unavailable status.																														
			No	0	Not all of the currently selected harts unavailable.																														
			Yes	1	All of the currently selected harts unavailable.																														
L	R	ANYNONEXISTENT			Any currently selected harts nonexistent status.																														
			No	0	None of the currently selected harts nonexistent.																														
			Yes	1	Any of the currently selected harts nonexistent.																														
M	R	ALLNONEXISTENT			All currently selected harts nonexistent status.																														
			No	0	Not all of the currently selected harts nonexistent.																														
			Yes	1	All of the currently selected harts nonexistent.																														
N	R	ANYRESUMEACK			Any currently selected harts acknowledged last resume request.																														
			No	0	None of the currently selected harts acknowledged last resume request.																														
			Yes	1	Any of the currently selected harts acknowledged last resume request.																														
O	R	ALLRESUMEACK			All currently selected harts acknowledged last resume																														
			No	0	Not all of the currently selected harts acknowledged last resume request.																														

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0							
ID				R										Q										P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A	A	A	A
Reset 0x00400082				0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0					
ID	R/W	Field	Value ID	Value	Description																																					
P	R	ANYHAVERESET	Yes	1	All of the currently selected harts acknowledged last resume request.																																					
			No	0	Any currently selected harts have been reset and reset is not acknowledged.																																					
			Yes	1	None of the currently selected harts have been reset and reset is not acknowledged.																																					
Q	R	ALLHAVERESET	Yes	1	Any of the currently selected harts have been reset and reset is not acknowledged.																																					
			No	0	All currently selected harts have been reset and reset is not acknowledge																																					
			Yes	1	Not all of the currently selected harts have been reset and reset is not acknowledge.																																					
R	R	IMPEBREAK	No	0	All of the currently selected harts have been reset and reset is not acknowledge.																																					
			Yes	1	Implicit ebreak instruction at the non-existent word immediately after the Program Buffer.																																					
			No	0	No implicit ebreak instruction.																																					
			Yes	1	Implicit ebreak instruction.																																					

8.26.1.13 DEBUGIF.HARTINFO

Address offset: 0x448

Hart Information

This register gives information about the hart currently selected by hartsel. This register is optional. If it is not present it should read all-zero. If this register is included, the debugger can do more with the Program Buffer by writing programs which explicitly access the data and/or dscratch registers

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D D D D																C B B B B A A A A A A A A A A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	DATAADDR		-2048 .. 2047	Data Address																														
If dataaccess is 0: The number of the first CSR dedicated to shadowing the data registers. If dataaccess is 1: Address of RAM where the data registers are shadowed. This address is sign extended and easily addressed with a load or store using x0 as the address register.																																			
B	R	DATASIZE		0 .. 12	Data Size																														
If dataaccess is 0: Number of CSRs dedicated to shadowing the data registers. If dataaccess is 1: Number of 32-bit words in the memory map dedicated to shadowing the data registers. Since there are at most 12 data registers, the value in this register must be 12 or smaller																																			
C	R	DATAACCESS			Data Access																														
			No	0	The data registers are shadowed in the hart by CSRs. Each CSR is DXLEN bits in size, and corresponds to a single argument.																														
			Yes	1	The data registers are shadowed in the hart’s memory map. Each register takes up 4 bytes in the memory map.																														
D	R	NSCRATCH			Number of dscratch registers																														
Number of dscratch registers available for the debugger to use during program buffer execution, starting from dscratch0. The debugger can make no assumptions about the contents of these registers between commands.																																			

8.26.1.14 DEBUGIF.HALTSUM1

Address offset: 0x44C

Halt Summary 1

Each bit in this read-only register indicates whether any of a group of harts is halted or not. Unavailable/nonexistent harts are not considered to be halted. This register might not be present if fewer than 33 harts are connected to this DM. The LSB reflects the halt status of harts hartsel[19:10] 0x0 through 0x1f. The MSB reflects the halt status of harts hartsel[19:10] 0x3e0 through 0x3ff.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	HALTSUM1						Halt Summary 1																											

8.26.1.15 DEBUGIF.HAWINDOWSEL

Address offset: 0x450

Hart Array Window Select

This register selects which of the 32-bit portion of the hart array mask register is accessible in hawindow

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																														
ID				A A																														

8.26.1.16 DEBUGIF.HAWINDOW

Address offset: 0x454

Hart Array Window

This register provides R/W access to a 32-bit portion of the hart array mask register. The position of the window is determined by hawindowssel. I.e. bit 0 refers to hart hawindowssel x 32, while bit 31 refers to hart hawindowssel x 32 + 31. Since some bits in the hart array mask register may be constant 0, some bits in this register may be constant 0, depending on the current value of hawindowssel.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	MASKDATA						Mask data.																											

8.26.1.17 DEBUGIF.ABSTRACTCS

Address offset: 0x458

Abstract Control and Status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D D D D D										C										B B B			A A A A								
Reset 0x01000002				0 0 0 0 0 0 0 0 1 0 1 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	DATACOUNT			Number of data registers that are implemented as part of the abstract command interface. Valid sizes are 1..12.																														
B	RW	CMDERR	NoError	0	No error.																														
			Busy	1	An abstract command was executing while command, abstractcts, or abstractauto was written, or when one of the data or progbuf registers was read or written. This status is only written if cmderr contains 0																														
			NotSupported	2	The requested command is notsupported, regardless of whether the hart is running or not.																														
			Exception	3	An exception occurred while executing the command (e.g. while executing theProgram Buffer).																														
			HaltResume	4	The abstract command couldn't execute because the hart wasn't in the required state (running/halted). or unavailable.																														
			Bus	5	The abstract command failed due to abus error (e.g. alignment, access size, or timeout).																														
			Other	7	The command failed for another reason.																														
C	R	BUSY			Abstract command execution status.																														
			NotBusy	0	Not busy.																														
			Busy	1	An abstract command is currently being executed. This bit is set as soon as command is written, and is not cleared until that command has completed.																														
D	R	PROGBUFSIZE			Size of the Program Buffer, in 32-bit words. Valid sizes are 0 - 1.																														

8.26.1.18 DEBUGIF.ABSTRACTCMD

Address offset: 0x45C

Abstract command

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				B	B	B	B	B	B	B	B	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																													
A	W	CONTROL				This Field is interpreted in a command specific manner, described for each abstract command.																													
B	W	CMDTYPE				The type determines the overall functionality of this abstract command.																													
			REGACCESS	0	Register Access Command																														
			QUICKACCESS	1	Quick Access Command																														
			MEMACCESS	2	Memory Access Command																														

8.26.1.19 DEBUGIF.ABSTRACTAUTO

Address offset: 0x460

Abstract Command Autoexec

This register is optional. Including it allows more efficient burst accesses. A debugger can detect whether it is support by setting bits and reading them back. Writing this register while an abstract command is executing causes cmderr to become 1 (busy) once the command completes (busy becomes 0).

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID		B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B
Reset 0x00000000		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value	ID	Value	Description																											
A	R	AUTOEXECDATA				When a bit in this field is 1, read or write accesses to the corresponding data word cause the command in command to be executed again.																											
B	R	AUTOEXECPROGBUF				When a bit in this field is 1, read or write accesses to the corresponding progbuf word cause the command in command to be executed again.																											

8.26.1.20 DEBUGIF.CONFSTRPTR[n] (n=0..3)

Address offset: 0x464 + (n × 0x4)

Configuration String Pointer [n]

When confstrptrvalid is set, reading this register returns bits 31:0 of the configuration string pointer. Reading the other confstrptr registers returns the upper bits of the address. When system bus mastering is implemented, this must be an address that can be used with the System Bus Access module. Otherwise, this must be an address that can be used to access the configuration string from the hart with ID 0.32 RISC-V External Debug Support Version 0.14.0-DRAFT If confstrptrvalid is 0, then the confstrptr registers hold identifier information.

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID		A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value	ID	Value	Description																											
A	R	ADDR				Address																											

8.26.1.21 DEBUGIF.NEXTDM

Address offset: 0x474

Next Debug Module

If there is more than one DM accessible on this DMI, this register contains the base address of the next one in the chain, or 0 if this is the last one in the chain.

Bit number		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID		A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value	ID	Value	Description																											
A	R	ADDR				Address																											

8.26.1.22 DEBUGIF.PROGBUF[n] (n=0..15)

Address offset: 0x480 + (n × 0x4)

Program Buffer [n]

progbuf0 through progbuf15 provide read/write access to the optional program buffer. progbufsize indicates how many of them are implemented starting at progbuf0, counting up. Accessing these registers while an abstract command is executing causes cmderr to be set to 1 (busy) if it is 0. Attempts to write them while busy is set does not change their value. The values in these registers may not be preserved after an abstract command is executed. The only guarantees on their contents are the ones offered by the command in question. If the command fails, no assumptions can be made about the contents of these registers.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	DATA										Data																							

8.26.1.23 DEBUGIF.AUTHDATA

Address offset: 0x4C0

Authentication Data

This register serves as a 32-bit serial port to/from the authentication module. When authbusy is clear, the debugger can communicate with the authentication module by reading or writing this register. There is no separate mechanism to signal overflow/underflow.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	DATA										Data																							

8.26.1.24 DEBUGIF.HALTSUM2

Address offset: 0x4D0

Halt Summary 2

Each bit in this read-only register indicates whether any of a group of harts is halted or not. Unavailable/nonexistent harts are not considered to be halted. This register might not be present if fewer than 1025 harts are connected to this DM. The LSB reflects the halt status of harts hartsel[19:15] 0x0 through hartsel[19:15] 0x3ff. The MSB reflects the halt status of harts hartsel[19:15] 0x7c00 through hartsel[19:15] 0x7fff

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	HALTSUM2										Halt Summary 2																							

8.26.1.25 DEBUGIF.HALTSUM3

Address offset: 0x4D4

Halt Summary 3

Each bit in this read-only register indicates whether any of a group of harts is halted or not. Unavailable/nonexistent harts are not considered to be halted. This register might not be present if fewer than 32769 harts are connected to this DM. The LSB reflects the halt status of harts 0x0 through 0x7fff. The MSB reflects the halt status of harts 0xf8000 through 0xfffff

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	R	HALTSUM3										Halt Summary 3																							

8.26.1.26 DEBUGIF.SBADDRESS3

Address offset: 0x4DC

System Bus Address 127:96

If sbasize is less than 97, then this register is not present. When the system bus master is busy, writes to this register will set sbbusyerror and don't do anything else.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																															
A	R	ADDRESS			Accesses bits 127:96 of the physical address in sbaddress (if the system address bus is that wide).																															

8.26.1.27 DEBUGIF.SBCS

Address offset: 0x4E0

System Bus Access Control and Status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				N N N M L K J J J I H G G G F F F F F F F E D C B A																															
Reset 0x20000000				0 0 1 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	SBACCESS8	sbaccess8	1	8-bit system bus accesses are supported.																														
B	R	SBACCESS16	sbaccess16	1	16-bit system bus accesses are supported.																														
C	R	SBACCESS32	sbaccess32	1	32-bit system bus accesses are supported.																														
D	R	SBACCESS64	sbaccess64	1	64-bit system bus accesses are supported.																														
E	R	SBACCESS128	sbaccess128	1	128-bit system bus accesses are supported.																														
F	R	SBASIZE			Width of system bus addresses in bits. (0 indicates there is no bus access support.)																														
G	R	SBERROR			When the Debug Module’s system bus master encounters an error, this field gets set. The bits in this field remain set until they are cleared by writing 1 to them. While this field is non-zero, no more system bus accesses can be initiated by the Debug Module. An implementation may report Other error (7) for any error condition.																														
			Normal	0	There was no bus error.																														
			Timeout	1	There was a timeout.																														
			Address	2	A bad address was accessed.																														
			Alignment	3	There was an alignment error.																														
			Size	4	An access of unsupported size was requested.																														
			Other	7	Other.																														
H	R	SBREADONDATA	sbreadondata	1	Every read from sbdata0 automatically triggers a system bus read at the (possibly autoincremented) address.																														
I	R	SBAUTOINCREMENT	sbautoincrement	1	sbaddress is incremented by the access size (in bytes) selected in sbaccess after every system bus access.																														

Bit number			31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID			N N N								M L K J J J I H G G G F F F F F F F E D C B A																							
Reset 0x20000000			0 0 1 0																															
ID	R/W	Field	Value ID	Value	Description																													
J	R	SBACCESS			Select the access size to use for system bus accesses. If sbaccess has an unsupported value when the DM starts a bus access, the access is not performed and sberror is set to 4.																													
			size8	0	8-bit.																													
			size16	1	16-bit.																													
			size32	2	32-bit.																													
			size64	3	64-bit.																													
			size128	4	128-bit.																													
K	R	SBREADONADDR	sbreadonaddr	1	Every write to sbaddress0 automatically triggers a system bus read at the new address.																													
L	R	SBBUSY			(Whether the system bus itself is busy is related, but not the same thing.) This bit goes high immediately when a read or write is requested for any reason, and does not go low until the access is fully completed. Writes to sbcs while sbbusy is high result in undefined behavior. A debugger must not write to sbcs until it reads sbbusy as 0.																													
			notbusy	0	System bus master is not busy.																													
			busy	1	System bus master is busy.																													
M	R	SBBUSYERROR			Set when the debugger attempts to read data while a read is in progress, or when the debugger initiates a new access while one is already in progress (while sbbusy is set). It remains set until it’s explicitly cleared by the debugger. While this field is set, no more system bus accesses can be initiated by the Debug Module.																													
			noerror	0	No error.																													
			error	1	Debugger access attempted while one in progress.																													
N	R	SBVERSION	version0	0	The System Bus interface conforms to mainline drafts of thia RISC-V External Debug Support spec older than 1 January, 2018.																													
			version1	1	The System Bus interface conforms to RISC-V External Debug Support version 0.14.0-DRAFT. Other values are reserved for future versions.																													

8.26.1.28 DEBUGIF.SBADDRESS0

Address offset: 0x4E4

System Bus Address 31:0

If sbasize is 0, then this register is not present. When the system bus master is busy, writes to this register will set sbbusyerror and don't do anything else. If sberror is 0, sbbusyerror is 0, and sbreadonaddr is set then writes to this register start the following: 1. Set sbbusy. 2. Perform a bus read from the new value of sbaddress. 3. If the read succeeded and sbautoincrement is set, increment sbaddress. 4. Clear sbbusy.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	ADDRESS						Accesses bits 31:0 of the physical address in sbaddress.																											

8.26.1.29 DEBUGIF.SBADDRESS1

Address offset: 0x4E8

System Bus Address 63:32

If sbasize is less than 33, then this register is not present. When the system bus master is busy, writes to this register will set sbbusyerror and don't do anything else.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																															
A	R	ADDRESS			Accesses bits 63:32 of the physical address in sbaddress (if the system address bus is that wide).																															

8.26.1.30 DEBUGIF.SBADDRESS2

Address offset: 0x4EC

System Bus Address 95:64

If sbasize is less than 65, then this register is not present. When the system bus master is busy, writes to this register will set sbbusyerror and don't do anything else.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	ADDRESS						Accesses bits 95:64 of the physical address in sbaddress (if the system address bus is that wide).																											

8.26.1.31 DEBUGIF.SBDATA0

Address offset: 0x4F0

System Bus Data 31:0

If all of the sbaccess bits in sbcs are 0, then this register is not present. Any successful system bus read updates sbdata. If the width of the read access is less than the width of sbdata, the contents of the remaining high bits may take on any value. If either sberror or sbbusyerror isn't 0 then accesses do nothing. If the bus master is busy then accesses set sbbusyerror, and don't do anything else. Writes to this register start the following: 1. Set sbbusy. 2. Perform a bus write of the new value of sbdata to sbaddress. 3. If the write succeeded and sbautoincrement is set, increment sbaddress. 4. Clear sbbusy. Reads from this register start the following: 1. "Return" the data. 2. Set sbbusy. 3. If sbreadondata is set: (a) Perform a system bus read from the address contained in sbaddress, placing the result in sbdata. (b) If sbautoincrement is set and the read was successful, increment sbaddress. 4. Clear sbbusy. Only sbdata0 has this behavior. The other sbdata registers have no side effects. On systems that have buses wider than 32 bits, a debugger should access sbdata0 after accessing the other sbdata registers

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	DATA						Accesses bits 31:0 of sbdata																											

8.26.1.32 DEBUGIF.SBDATA1

Address offset: 0x4F4

System Bus Data 63:32

If sbaccess64 and sbaccess128 are 0, then this register is not present. If the bus master is busy then accesses set sbbusyerror, and don't do anything else.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	DATA						Accesses bits 63:32 of sbdata (if the system bus is that wide).																											

8.26.1.33 DEBUGIF.SBDATA2

Address offset: 0x4F8

System Bus Data 95:64

This register only exists if sbaccess128 is 1. If the bus master is busy then accesses set sbbusyerror, and don't do anything else

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A			
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value																									Description									
A	R	DATA																											Accesses bits 95:64 of sbdata (if the system bus is that wide).									

8.26.1.34 DEBUGIF.SBDATA3

Address offset: 0x4FC

System Bus Data 127:96

This register only exists if sbaccess128 is 1. If the bus master is busy then accesses set sbbusyerror, and don't do anything else

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	DATA						Accesses bits 127:96 of sbdata (if the system bus is that wide).																											

8.26.1.35 DEBUGIF.HALTSUM0

Address offset: 0x500

Halt summary 0

Each bit in this read-only register indicates whether one specific hart is halted or not. Unavailable/nonexistent harts are not considered to be halted. This register might not be present if fewer than 2 harts are connected to this DM. The LSB reflects the halt status of hart hartsel[19:5] 0x0, and the MSB reflects halt status of hart hartsel[19:5] 0xf

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	HALTSUM0						Halt summary 0																											

8.26.1.36 CPURUN

Address offset: 0x800

State of the CPU after a core reset

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EN			Controls CPU running state after a core reset.																														
			Stopped	0	CPU stopped. If this is the CPU state after a core reset, setting this bit will change the CPU state to CPU running.																														
			Running	1	CPU running. If this is the CPU state after a core reset, clearing this bit will change the CPU state to CPU stopped after a core reset.																														

8.26.1.37 INITPC

Address offset: 0x808

Initial value of the PC at CPU start.

Note: This address must be 64 bit aligned

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	INITPC						Initial value of the PC at CPU start.																											
								This value should be set before setting CPURUN.EN. After setting CPURUN.EN, this register can be reconfigured to prepare the CPU to start from a new initial PC upon receiving a reset request																											

8.26.2 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
VPRCLIC	FLPR	0xF0000000	HF	NS	NA	No	VPR CLIC registers

Configuration

Instance	Domain	Configuration
VPRCLIC	FLPR	<p>Supported interrupts (IRQNUM): 0..270</p> <p>VEVIF tasks: 0..31</p> <p>Mask of supported VEVIF tasks: 0xFFFFFFFF</p> <p>VPR counter (CNT0) interrupt handler number (COUNTER_IRQ_NUM): 31</p> <p>CLIC configuration for VPR 1.2 enabled</p>

Register overview

Register	Offset	TZ	Description
CLIC.CLICCFG	0x0000		CLIC configuration.
CLIC.CLICINFO	0x0004		CLIC information.
CLIC.CLICINT[n]	0x1000		Interrupt control register for IRQ number [n].

8.26.2.1 CLIC.CLICCFG

Address offset: 0x0000

CLIC configuration.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C C B B B B A																															
Reset 0x00000011				0 1 0 0 0 1																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	NVBITS			Selective interrupt hardware vectoring.																														
			Implemented	1	Selective interrupt hardware vectoring is implemented																														
B	R	NLBITS			Interrupt level encoding.																														
					Indicates how many upper bits are assigned to encode the interrupt level in CLICINT[n].PRIORITY																														
			Eight	8	8 bits = interrupt levels encoded in eight bits																														
C	R	NMBITS			Interrupt privilege mode.																														
					Indicates how many bits represent privilege mode																														
			ModeM	0	All interrupts are M-mode only																														

8.26.2.2 CLIC.CLICINFO

Address offset: 0x0004

CLIC information.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				C C C C C C C								B B B B B B B								A A A A A A A A A A A A A A A A A															
Reset 0x00401FFF				0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1																															
ID	R/W	Field	Value ID	Value				Description																											
A	R	NUMINTERRUPTS						Maximum number of interrupts supported.																											
B	R	VERSION						Version																											
C	R	NUMTRIGGER						Number of maximum interrupt triggers supported																											

8.26.2.3 CLIC.CLICINT[n] (n=0..270)

Address offset: $0x1000 + (n \times 0x4)$

Interrupt control register for IRQ number [n].

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				H H H H H H H H G G F F E D D D D D D C B B B B B B B A																															
Reset 0x3FC30000				0 0 1 1 1 1 1 1 1 1 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	IP			Interrupt Pending bit.																														
			NotPending	0	Interrupt not pending																														
			Pending	1	Interrupt pending																														
B	R	READ1			Read as 0, write ignored.																														
C	RW	IE			Interrupt enable bit.																														
			Disabled	0	Interrupt disabled																														
			Enabled	1	Interrupt enabled																														
D	R	READ2			Read as 0, write ignored.																														
E	R	SHV			Selective Hardware Vectoring.																														
			Vectored	1	Hardware vectored																														
F	R	TRIG			Trigger type and polarity for each interrupt input.																														
			EdgeTriggered	1	Interrupts are edge-triggered																														
G	R	MODE			Privilege mode.																														
			MachineMode	3	Machine mode																														
H	RW	PRIORITY			Interrupt priority level																														
			PRIOLEVEL0	0x3F	Priority level 0																														
			PRIOLEVEL1	0x7F	Priority level 1																														
			PRIOLEVEL2	0xBF	Priority level 2																														
			PRIOLEVEL3	0xFF	Priority level 3																														

8.27 WDT — Watchdog timer

The countdown watchdog timer (WDT) uses the low-frequency clock source (LFCLK) and offers configurable and robust protection against application lock-up.

The main features of WDT are:

- Generates watchdog reset
- Optional pause of WDT when the CPU is sleeping or when it is stopped by the debugger
- Optional generation of non-maskable interrupt (NMI)
- Runs off the low-frequency clock source (LFCLK)

WDT must be configured before it is started. After configuration, WDT is started by triggering the START task.

When WDT is running, its configuration registers (CRV, RREN, and CONFIG) are blocked for further configuration.

WDT can be paused while the CPU is sleeping, or when the debugger has halted the CPU. WDT is implemented as a down-counter that generates a TIMEOUT event when it wraps over after counting down to 0. When WDT is started by the START task, the watchdog counter is loaded with the value specified in the CRV register. This counter is also reloaded with the value specified in the CRV register when a reload request is granted.

The timeout period for the watchdog is given by the following equation:

$$\text{timeout [s]} = (\text{CRV} + 1) / 32768$$

When started, WDT will make the 32.768 kHz RC oscillator start if no other 32.768 kHz clock source is running and generating the 32.768 kHz system clock, see chapter [CLOCK — Clock control](#) on page 76.

8.27.1 Reload criteria

WDT has eight separate reload request registers. These registers are used to request WDT to reload its counter with the value specified in the CRV register. To reload the watchdog counter, write 0x6E524635 to all enabled reload registers.

One or more RR registers can be individually enabled through the RREN register.

8.27.2 Temporarily pausing the watchdog

By default, the watchdog will be active counting down the down-counter while the CPU is sleeping. It is possible to configure the watchdog to automatically pause when the CPU is sleeping or when it is stopped by the debugger.

Entering System OFF mode will stop and disable the watchdog.

8.27.3 Watchdog reset

A TIMEOUT event automatically leads to a watchdog reset.

If the watchdog is configured to generate an interrupt on the TIMEOUT event, the watchdog reset is postponed by two 32.768 kHz clock cycles after the TIMEOUT event is generated. Once the TIMEOUT event is generated, and unless the watchdog is stopped, the impending watchdog reset will occur.

The watchdog can be reset from several reset sources, see [Reset behavior](#) on page 108. After a reset, the watchdog configuration registers are available for configuration.

See [RESET — Reset control](#) on page 106 for more information about reset sources.

The TIMEOUT event will also generate NMI interrupt, when NMI interrupt is supported. See the the instance's configuration in [Instantiation](#) on page 216 to see if NMI is supported.

8.27.4 Stopping the watchdog

By default, the watchdog cannot be stopped. It is possible to configure the watchdog to allow the STOP task.

To stop the watchdog, perform the following steps.

1. Set the [CONFIG](#) register's STOPEN field to `Enable` during watchdog configuration.
2. Write the special value 0x6E524635 to the [TSEN](#) register.
3. Invoke the STOP task.

When these conditions are met, the watchdog is stopped and a STOPPED event is issued.

When the watchdog is stopped, its configuration registers [CRV](#), [RREN](#), and [CONFIG](#) are no longer blocked.

Note: It is recommended to write zeros to [TSEN](#) on page 747 after the watchdog has stopped, to avoid runaway code triggering the STOP task.

8.27.5 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
WDT30	GLOBAL	0x50108000	HF	S	NA	No	Watchdog timer WDT30
WDT31 : S	GLOBAL	0x50109000	US	S	NA	No	Watchdog timer WDT31
WDT31 : NS		0x40109000					

Configuration

Instance	Domain	Configuration
WDT30	GLOBAL	Supports non-maskable interrupts (NMI).
WDT31 : S	GLOBAL	Does not generate non-maskable interrupts.
WDT31 : NS		

Register overview

Register	Offset	TZ	Description
TASKS_START	0x000		Start WDT
TASKS_STOP	0x004		Stop WDT
SUBSCRIBE_START	0x080		Subscribe configuration for task START
SUBSCRIBE_STOP	0x084		Subscribe configuration for task STOP
EVENTS_TIMEOUT	0x100		Watchdog timeout
EVENTS_STOPPED	0x104		Watchdog stopped
PUBLISH_TIMEOUT	0x180		Publish configuration for event TIMEOUT
PUBLISH_STOPPED	0x184		Publish configuration for event STOPPED
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
NMIENSET	0x324		Enable interrupt
NMIENCLR	0x328		Disable interrupt
RUNSTATUS	0x400		Run status
REQSTATUS	0x404		Request status
CRV	0x504		Counter reload value
RREN	0x508		Enable register for reload request registers
CONFIG	0x50C		Configuration register
TSEN	0x520		Task stop enable
RR[n]	0x600		Reload request n

8.27.5.1 TASKS_START

Address offset: 0x000

Start WDT

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																																	A									
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																					
A	W	TASKS_START			Start WDT																																					
			Trigger	1	Trigger task																																					

8.27.5.2 TASKS_STOP

Address offset: 0x004

Stop WDT

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	W	TASKS_STOP						Stop WDT																											
			Trigger	1				Trigger task																											

8.27.5.3 SUBSCRIBE_START

Address offset: 0x080

Subscribe configuration for task [START](#)

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B																								A A A A A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CHIDX		[0..255]				DPPI channel that task START will subscribe to																											
B	RW	EN																																	
			Disabled	0			Disable subscription																												
			Enabled	1			Enable subscription																												

8.27.5.4 SUBSCRIBE_STOP

Address offset: 0x084

Subscribe configuration for task [STOP](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																												
ID				B																								A				A	A	A	A	A	A	A																									
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																																																									
A	RW	CHIDX		[0..255]		DPPI channel that task STOP will subscribe to																																																									
B	RW	EN																																																													
			Disabled	0	Disable subscription																																																										
			Enabled	1	Enable subscription																																																										

8.27.5.5 EVENTS_TIMEOUT

Address offset: 0x100

Watchdog timeout

Bit number										31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																								A			
Reset 0x00000000										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																																						
A	RW	EVENTS_TIMEOUT			Watchdog timeout																																						
			NotGenerated	0	Event not generated																																						
			Generated	1	Event generated																																						

8.27.5.6 EVENTS_STOPPED

Address offset: 0x104

Watchdog stopped

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	EVENTS_STOPPED			Watchdog stopped																														
			NotGenerated	0	Event not generated																														
			Generated	1	Event generated																														

8.27.5.7 PUBLISH_TIMEOUT

Address offset: 0x180

Publish configuration for event [TIMEOUT](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value				Description																														
A	RW	CHIDX		[0..255]				DPPI channel that event TIMEOUT will publish to																														
B	RW	EN																																				
			Disabled	0	Disable publishing																																	
			Enabled	1	Enable publishing																																	

8.27.5.8 PUBLISH_STOPPED

Address offset: 0x184

Publish configuration for event [STOPPED](#)

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				B																								A				A	A	A	A	A	A	A
Reset 0x00000000				0 0																																		
ID	R/W	Field	Value ID	Value				Description																														
A	RW	CHIDX		[0..255]				DPPI channel that event STOPPED will publish to																														
B	RW	EN																																				
			Disabled	0				Disable publishing																														
			Enabled	1				Enable publishing																														

8.27.5.9 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	TIMEOUT			Write '1' to enable interrupt for event TIMEOUT																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	STOPPED			Write '1' to enable interrupt for event STOPPED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.27.5.10 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	TIMEOUT			Write '1' to disable interrupt for event TIMEOUT																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	STOPPED			Write '1' to disable interrupt for event STOPPED																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.27.5.11 NMIENSET

Address offset: 0x324

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	TIMEOUT			Write '1' to enable interrupt for event TIMEOUT																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	STOPPED			Write '1' to enable interrupt for event STOPPED																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

8.27.5.12 NMIENCLR

Address offset: 0x328

Disable interrupt

Bit number		31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																			
ID																																			B	A	
Reset 0x00000000		0 0																																			
ID	R/W	Field	Value ID	Value	Description																																
A	RW	TIMEOUT			Write '1' to disable interrupt for event TIMEOUT																																
			Clear	1	Disable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																
B	RW	STOPPED			Write '1' to disable interrupt for event STOPPED																																
			Clear	1	Disable																																
			Disabled	0	Read: Disabled																																
			Enabled	1	Read: Enabled																																

8.27.5.13 RUNSTATUS

Address offset: 0x400

Run status

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	RUNSTATUSWDT			Indicates whether or not WDT is running																														
			NotRunning	0	Watchdog is not running																														
			Running	1	Watchdog is running																														

8.27.5.14 REQSTATUS

Address offset: 0x404

Request status

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					H G F E D C B A																															
Reset 0x00000001					0 1																															
ID	R/W	Field	Value ID	Value	Description																															
A-H	R	RR[i] (i=0..7)			Request status for RR[i] register																															
			DisabledOrRequested0		RR[i] register is not enabled, or are already requesting reload																															
			EnabledAndUnrequested		RR[i] register is enabled, and are not yet requesting reload																															

8.27.5.15 CRV

Address offset: 0x504

Counter reload value

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0xFFFFFFFF				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
ID	R/W	Field	Value ID	Value				Description																											
A	RW	CRV		[0xF..0xFFFFFFFF]				Counter reload value in number of cycles of the 32.768 kHz clock																											

8.27.5.16 RREN

Address offset: 0x508

Enable register for reload request registers

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																												H	G	F	E	D	C	B	A
Reset 0x00000001				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
ID	R/W	Field	Value ID	Value		Description																													
A-H	RW	RR[i] (i=0..7)				Enable or disable RR[i] register																													
			Disabled	0		Disable RR[i] register																													
			Enabled	1		Enable RR[i] register																													

8.27.5.17 CONFIG

Address offset: 0x50C

Configuration register

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				C																												B		A		
Reset 0x00000001				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
ID	R/W	Field	Value ID	Value	Description																															
A	RW	SLEEP			Configure WDT to either be paused, or kept running, while the CPU is sleeping																															
			Pause	0	Pause WDT while the CPU is sleeping																															
			Run	1	Keep WDT running while the CPU is sleeping																															
B	RW	HALT			Configure WDT to either be paused, or kept running, while the CPU is halted by the debugger																															
			Pause	0	Pause WDT while the CPU is halted by the debugger																															
			Run	1	Keep WDT running while the CPU is halted by the debugger																															
C	RW	STOPEN			Allow stopping WDT																															
			Disable	0	Do not allow stopping WDT																															
			Enable	1	Allow stopping WDT																															

8.27.5.18 TSEN

Address offset: 0x520

Task stop enable

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	W	TSEN						Allow stopping WDT																											
			Enable	0x6E524635				Value to allow stopping WDT																											

9 Debug and trace

The debug and trace system is a flexible and powerful mechanism for non-intrusive debugging.

The main features of the debug and trace system are the following:

- Access port connection for Arm Cortex-M33
 - Eight breakpoints
 - Four watchpoint comparators
 - Instrumentation trace macrocell (ITM)
 - Embedded trace macrocell (ETM)
 - Access protection through APPROTECT, ERASEPROTECT, and SECUREAPPROTECT
- Serial wire debug (SWD) interface, protocol version 2 with multidrop support
- Control-access port (CTRL-AP) that enables control of the device when other debug access ports (DAP) have been disabled by the access port protection
- Trace port interface unit (TPIU)
 - 4-bit parallel trace of ITM and ETM trace data
 - Serial wire output (SWO) trace of ITM data

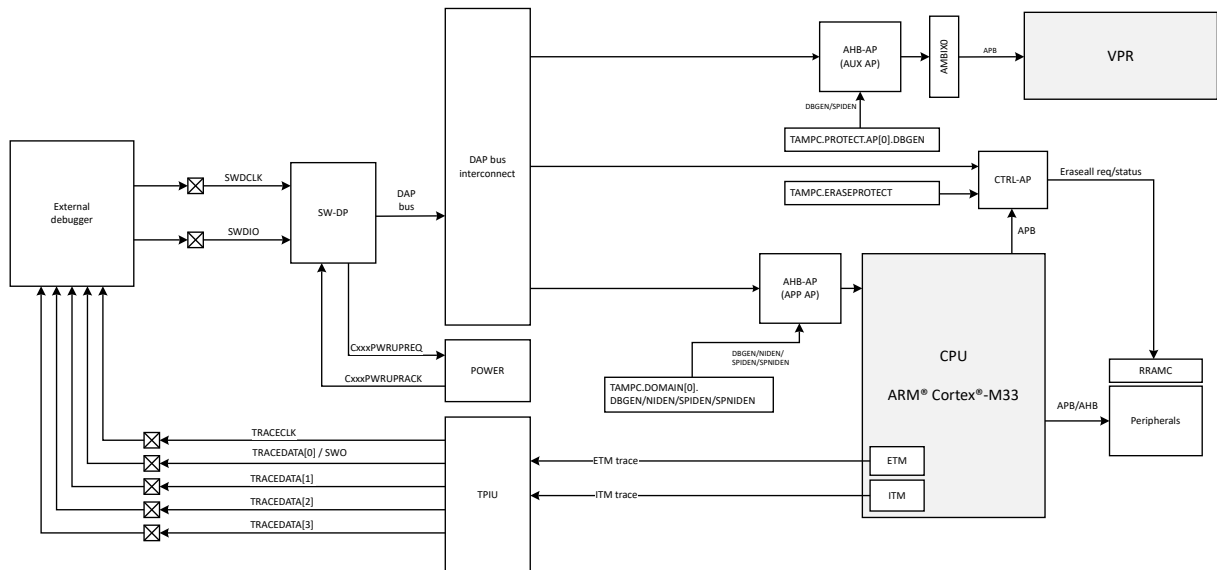


Figure 160: Debug and trace overview

9.1 Debug access port

An external debugger can access the device through the debug access port (DAP).

The DAP implements a standard serial wire debug (SWD) Arm CoreSight™ protocol with a two-pin serial interface (**SWDCLK** and **SWDIO**).

The **SWDIO** pin has an internal pull-up resistor. The **SWDCLK** pin has an internal pull-down resistor.

There are several access ports that connect to different parts of the system, as shown in the following table.

AP ID	Type	Description
0	AHB-AP	CM33 access port
1	AHB-AP	AUX access port
2	CTRL-AP	Control access port

Table 65: Access port overview

The AHB-AP is a standard Arm component. See the *ArmCoreSight SoC-400 Technical Reference Manual* revision r3p2 for more information. The control access port (CTRL-AP) is proprietary and described in more detail in [CTRL-AP - Control access port](#) on page 755.

9.1.1 Registers

Register overview

Register	Offset	TZ	Description
TARGETID	0x042		<p>The TARGETID register provides information about the target when the host is connected to a single device.</p> <p>The TARGETID register is accessed by a read of DP register 0x4 when the DPBANKSEL bit in the SELECT register is set to 0x2.</p>
DLPIDR	0x043		<p>The DLPIDR register provides information about the serial wire debug protocol version.</p> <p>Accessed by a read of DP register 0x4 when the DPBANKSEL bit in the SELECT register is set to 0x3.</p>

9.1.1.1 TARGETID

Address offset: 0x042

The TARGETID register provides information about the target when the host is connected to a single device.

The TARGETID register is accessed by a read of DP register 0x4 when the DPBANKSEL bit in the SELECT register is set to 0x2.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				C	C	C	C	B	B	B	B	B	B	B	B	B	B	B	B					A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	TDESIGNER						An 11-bit code JEDEC JEP106 continuation code and identity code. The ID identifies the designer of the part.																											
			NordicSemi	0x144				Nordic Semiconductor ASA.																											
B	R	TPARTNO						Part number.																											
C	R	TREVISION						Target revision.																											

9.1.1.2 DLPIDR

Address offset: 0x043

The DLPIDR register provides information about the serial wire debug protocol version.

Accessed by a read of DP register 0x4 when the DPBANKSEL bit in the SELECT register is set to 0x3.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID				B	B	B	B																					A	A	A	A					
Reset 0x00000001				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
ID	R/W	Field	Value ID	Value				Description																												
A	R	PROTVSN						Protocol version.																												
			SWDPv2	1				SW protocol version 2.																												
B	R	TINSTANCE						Target instance.																												
								This value is set by the UICR.TINSTANCE register.																												

9.2 Access port protection

The control access port is always accessible from the debugger, while access to the system resources through each core's individual access ports (AHB-AP) can be protected in different ways.

The following tables give an overview of the access port protection methods.

Registers	Description
UICR.APPROTECT, TAMPC.PROTECT.DOMAIN[0].- DBGEN, and TAMPC.PROTECT.- DOMAIN[0].NIDEN	These registers control the generation of the Arm Cortex-M33 AHB-AP DBGEN and NIDEN signals, which controls all non-secure access through the Arm Cortex-M33 AHB-AP. This is used to provide readback protection of the non-volatile memory contents. See also Arm Cortex-M33 access port protection for non-secure debug access on page 752. For more information about the DBGEN and NIDEN signals, see the <i>Arm CoreSight SoC-400 Technical Reference Manual, Revision r3p2</i> .
UICR.SECUREAPPROTECT, TAMPC.PROTECT.DOMAIN[0].- SPIDEN, and TAMPC.PROTECT.- DOMAIN[0].SPNIDEN	These registers control the generation of the Arm Cortex-M33 AHB-AP SPIDEN and SPNIDEN signals, which blocks all secure access through the Arm Cortex-M33 AHB-AP. This means that only the non-secure code can be debugged and accessed. To enable access to the secure access port, APPROTECT must be unprotected. See also Arm Cortex-M33 access port protection for secure debug access on page 752. For more information about the SPIDEN and SPNIDEN signals, see the <i>Arm CoreSight SoC-400 Technical Reference Manual, Revision r3p2</i> .
UICR.AUXAPPROTECT, TAMPC.PROTECT.AP[0].DBGEN	These registers control the generation of the AHB-AP DBGEN signal, which controls debug access to the VPR AHB-AP. This is used to provide debug capability for VPR.
UICR.ERASEPROTECT and TAMPC.PROTECT.ERASEPROTECT	Disables the CTRL-AP.ERASEALL and RRAMC ERASEALL functionality. This can be used together with APPROTECT to provide read-back and re-purposing protection.

Table 66: Access port protection overview

UICR and TAMPC are combined to enable or disable the access port protection. The access port is normally protected, and is opened when the following conditions are met:

1. UICR.APPROTECT must be Unprotected.
2. The corresponding TAMPC.PROTECT register must be written by firmware. See [TAMPC](#) signal protector for details on the procedure.

The following tables lists the available APPROTECT combinations.

Application core UICR.APPROTECT	TAMPC.PROTECT CTRL.VALUE register (DBGEN or NIDEN)	TAMPC.PROTECT CTRL.LOCK register (DBGEN or NIDEN)	DBGEN or NIDEN	Debug access to Application core AHB- AP
Protected	Low (0)	1 (Locked by UICR)	0	Not permitted
Unprotected	x (Software controlled*)	x (Software controlled)	0	Not permitted, but software controlled. For usage see TAMPC — Tamper controller on page 192.

Table 67: Arm Cortex-M33 access port protection for non-secure debug access

*) After reset, the TAMPC VALUE register is 0, preventing debug access to application core. On-chip software must write to TAMPC VALUE register before the debug access port is opened. See Access port unlocking below.

Application core UICR.SECURE- APPROTECT	TAMPC.PROTECT CTRL.VALUE register (SPIDEN or SPNIDEN)	TAMPC.PROTECT CTRL.LOCK register (DBGEN or NIDEN)	SPIDEN or SPNIDEN	Secure debug access to Application core AHB-AP
Protected	Low (0) (Locked by UICR)	1 (Locked by UICR)	0	Not permitted
Unprotected	x (Software controlled*)	x (Software controlled)	0	Not permitted, but software controlled. For usage see TAMPC — Tamper controller on page 192.

Table 68: Arm Cortex-M33 access port protection for secure debug access

*) After reset, the TAMPC VALUE register is 0, preventing debug access to application core. On-chip software must write to TAMPC VALUE register before the debug access port is opened. See Access port unlocking below.

Register TAMPC CTRL VALUE		Debug access	
SPIDEN	DBGEN	Secure mode invasive AHB-AP access	Non-secure mode invasive AHB-AP access
0	0	No	No
0	1	No	Yes
1	0	No	No
1	1	Yes	Yes

Table 69: Arm Cortex-M33 debug authentication signals

The access port is also open after the completion of the CTRL-AP.ERASEALL operation. After completing the erase operation, CTRL-AP will temporarily unprotect AHB-AP. AHB-AP will be protected when one of the following conditions are met:

- Power-on reset
- Brown-out reset
- Watchdog timer reset
- Pin reset

The following figure is an example on how a device with access port protection enabled can be erased, programmed, and configured to allow debugging. Operations sent from debugger as well as registers written by firmware will affect the access port state. The operation named Reset* is one of the conditions listed above.

The TAMPC write must follow a sequence, for more details see [TAMPC — Tamper controller](#) on page 192.

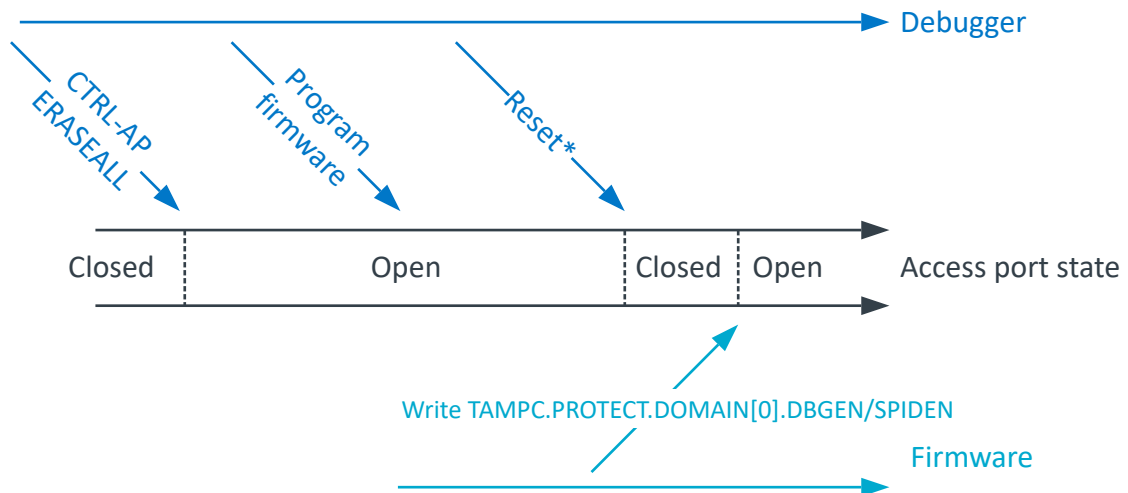


Figure 161: Access port unlocking

The debugger can read the access port protection status in the core's AHB-AP, using the Arm AHB-AP Control/Status Word register (CSW), defined in the *Arm CoreSight SoC-400 Technical Reference Manual, Revision r3p2*. The DbgStatus field indicates that the AHB-AP can perform AHB transfers, while the SPISStatus field indicates if secure AHB transfers are permitted. For a list of all debug access ports, see [Debug access port](#) on page 749.

9.3 Debug interface mode

Before the external debugger can connect to an access port, the debugger must first request the device to power up through CxxxPWRUPREQ in the SWJ-DP.

The device remains in debug interface mode when the debugger requests power through CxxxPWRUPREQ. Otherwise, the device is in Normal mode. When a debug session is over, the device must be set to Normal mode by the external debugger, followed by a pin reset. This reduces overall power consumption.

Some peripherals behave differently in debug interface mode compared to Normal mode. These differences are described in more detail in the corresponding peripheral chapter.

For details on how to use the debug capabilities, read the debug documentation of your IDE.

If the device is in System OFF when power is requested from CxxxPWRUPREQ, the system wakes up and the DIF flag in [RESETREAS](#) on page 109 is set.

9.4 Real-time debug

The device supports real-time debugging. This allows interrupts to execute to completion in real time when breakpoints are set in Thread mode or lower priority interrupts.

Real-time debugging enables setting a breakpoint for single-stepping through code. This prevents real-time event-driven threads from running at a higher priority. For example, this enables the device to continue to service high-priority interrupts of an external controller or sensor, without failure or loss of state synchronization, while stepping through code in a low-priority thread.

9.5 Multidrop serial wire debug

Multidrop serial wire debug (SWD) allows simultaneous access to an unlimited number of devices through a single connection. This is useful for connectivity-constrained products that have several chips with multidrop support.

To select a target in a multidrop capable product, the debugger must write the correct `TINSTANCE`, `TPARTNO`, and `TDESIGNER` fields into the SW-DP `TARGETSEL` register. Values for these fields are located in the registers [TARGETID](#) on page 750 and [DLPIDR](#) on page 750.

For more information about multidrop SWD, see the *Arm Debug Interface Architecture Specification*, ADIV5.0 to ADIV5.2.

9.6 Trace

The device supports ETM and ITM trace.

Trace data from the ETM and ITM is sent to an external debugger through a 4-bit wide parallel trace port (TPIU), as illustrated in [Debug and trace overview](#) on page 749.

In addition to parallel trace mode, the TPIU supports serial trace mode through the serial wire output (SWO) trace protocol. Parallel and serial trace modes cannot be used at the same time. ETM trace is only supported in parallel trace mode. ITM trace is supported in both parallel and serial trace mode. See the debug documentation of the IDE for more information.

TPIU trace pins are multiplexed with GPIOs. The **SWO** and **TRACEDATA[0]** pins can use the same GPIO. The **SWO** pin can also use a separate GPIO on P2. See [Pin assignments](#) on page 793 for more information.

Trace speed is configured in the register [TRACEPORTSPEED](#). Trace pin speed is determined by the GPIO drive setting of the multiplexed pins. See [GPIO — General purpose input/output](#) on page 273 for information on drive settings.

9.6.1 Enabling the trace port

A specific sequence of operations must be performed to enable the trace port.

1. Enable trace and debug using the following code.

```
NRF_TAD_S->ENABLE = TAD_ENABLE_ENABLE_Msk;
```

2. Set drive strength to the highest possible value to ensure fast operation. Do this for all trace pins that will be used.

```
#define TRACE_PIN_CLEAR      (~(GPIO_PIN_CNF_CTRLSEL_Msk | GPIO_PIN_CNF_DRIVE0_Msk \
                             | GPIO_PIN_CNF_DRIVE1_Msk))

#define TRACE_PIN_CONFIG    ((GPIO_PIN_CNF_DRIVE0_E0 << GPIO_PIN_CNF_DRIVE0_Pos) \
                             | (GPIO_PIN_CNF_DRIVE1_E1 << GPIO_PIN_CNF_DRIVE1_Pos))

// Clear the bitfields before configuring to make sure the correct value is written
NRF_P2_S->PIN_CNF[TRACE_TRACECLK_PIN] &= TRACE_PIN_CLEAR;
NRF_P2_S->PIN_CNF[TRACE_TRACEDATA0_PIN] &= TRACE_PIN_CLEAR;
NRF_P2_S->PIN_CNF[TRACE_TRACEDATA1_PIN] &= TRACE_PIN_CLEAR;
NRF_P2_S->PIN_CNF[TRACE_TRACEDATA2_PIN] &= TRACE_PIN_CLEAR;
NRF_P2_S->PIN_CNF[TRACE_TRACEDATA3_PIN] &= TRACE_PIN_CLEAR;

NRF_P2_S->PIN_CNF[TRACE_TRACECLK_PIN] |= TRACE_PIN_CONFIG;
NRF_P2_S->PIN_CNF[TRACE_TRACEDATA0_PIN] |= TRACE_PIN_CONFIG;
NRF_P2_S->PIN_CNF[TRACE_TRACEDATA1_PIN] |= TRACE_PIN_CONFIG;
NRF_P2_S->PIN_CNF[TRACE_TRACEDATA2_PIN] |= TRACE_PIN_CONFIG;
NRF_P2_S->PIN_CNF[TRACE_TRACEDATA3_PIN] |= TRACE_PIN_CONFIG;
```

3. Trace port speed is configured as a prescaled version of the CPU frequency and must be at least half the CPU frequency to avoid dropping trace packets.

```
NRF_TAD_S->TRACEPORTSPEED = TAD_TRACEPORTSPEED_TRACEPORTSPEED_DIV2;
```

Note: Do not run the trace port at less than half the CPU frequency, as this risks dropping trace packets.

4. Configure Arm CoreSight components. See documentation for Arm CoreSight for more information.

9.7 CTRL-AP - Control access port

The control access port (CTRL-AP) is a custom access port that enables control of the device when other access ports (AP) have been disabled by the access port protection.

For an overview of the other debug access ports, see [DAP](#).

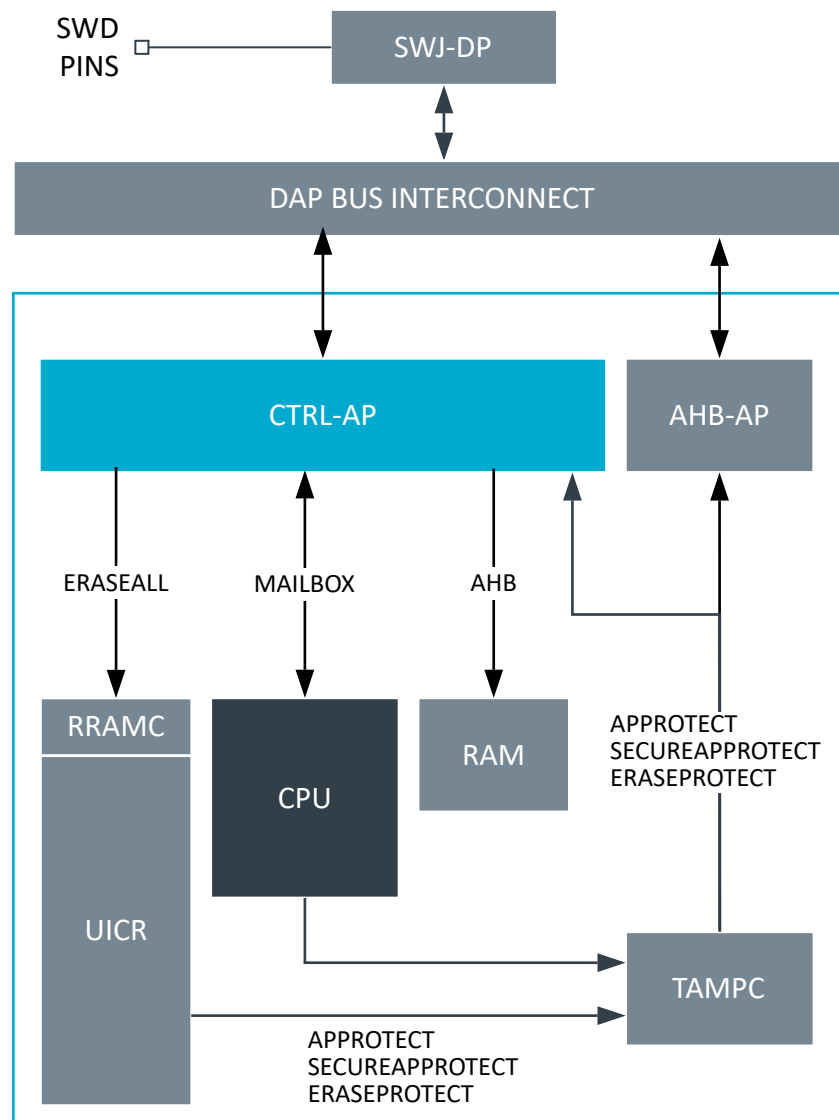


Figure 162: Control access port details

Access port protection (APPROTECT) blocks the debugger access to the AHB-AP, and prevents read and write access to all CPU registers and memory-mapped addresses. To enable port protection access for both secure and non-secure modes, use the registers UICR.SECUREAPPROTECT and UICR.APPROTECT.

Erase protection (ERASEPROTECT) protects the entire non-volatile memory, SICR, and UICR, from being erased. Erase protection is enabled using UICR.ERASEPROTECT. Erase protection is disabled using register ERASEPROTECT.DISABLE. For more information about disabling erase protection, see [Erase protection](#) on page 759.

CTRL-AP has the following features:

- Resetting the device
- Erase all
- Mailbox interface
- [INFO.PARTNO](#) and [INFO.HWREVISION](#) registers

There are two sets of registers in the peripheral:

1. Peripheral registers - Accessed by a CPU using APB interface.
2. Debugger registers - Accessed by an external debugger using CTRL-AP interface.

9.7.1 Reset request

The debugger can request the device to perform a reset.

The register [RESET](#) is used to request the reset. Once the reset is performed, the reset reason is accessible through the RESETREAS register. For more information about reset, see [RESET — Reset control](#) on page 106.

9.7.2 Erase all

The erase all function lets the debugger trigger an erase of non-volatile memory, user information configuration registers (UICR), secure information configuration region (SICR), RAM, all peripheral settings, and also temporarily removes the access port protection.

To trigger an erase all function, follow the sequence in the following flowchart. After the sequence has completed, the access port protection is removed until the next pin reset, power-on reset, brown-out reset, or watchdog timer reset.

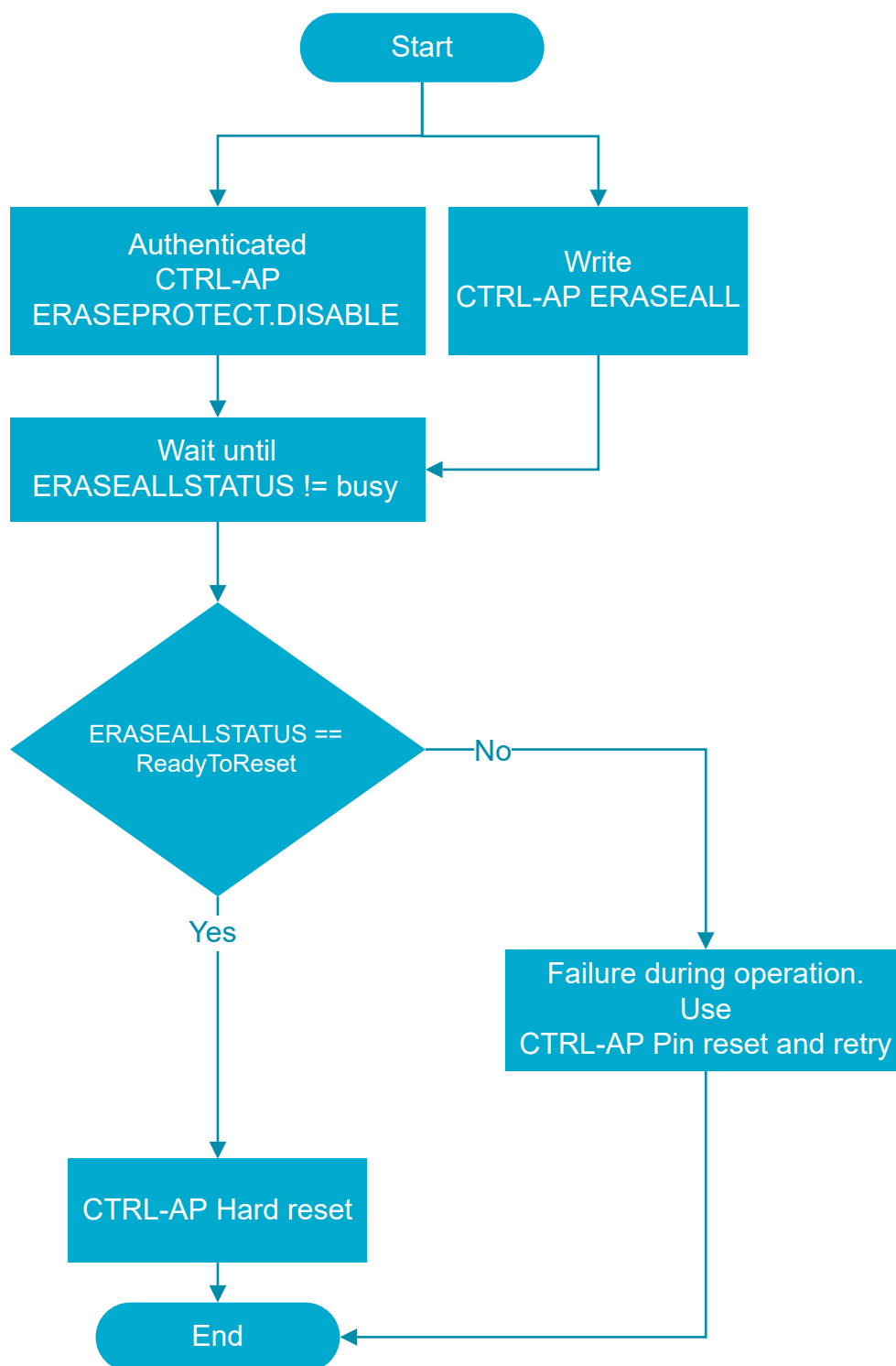


Figure 163: Erase all states

Erase all protection

It is possible to prevent the debugger from performing an erase all operation by using either the TAMPC `TAMPC.PROTECT.ERASEPROTECT` register, or the `UICR.ERASEPROTECT` register followed by a device reset.

Once this has been configured, the CTRL-AP `ERASEALL` operation is disabled.

The register `ERASEPROTECT.STATUS` on page 762 holds the status for erase protection.

9.7.2.1 Erase protection

Erase protection can be used to prevent a device from being erased.

The Erase all function can be initiated by the non-volatile memory controller, or by the control access port. The Erase all operation is enabled in some of the access port protection combinations, as listed in the following table.

Access port protection state				Erase all operation		
APPROTECT	SECUREAP-PROTECT	ERASEPROTECT	ERASEPROTECT.LOCK	RRAMC ERASEALL	CTRL-AP ERASEALL	CTRL-AP ERASE-PROTECT.-DISABLE
Unprotected	Unprotected	Unprotected	Any	Allowed	Allowed	Allowed
Unprotected	Protected	Unprotected	Any	Not allowed	Allowed	Allowed
Protected	Unprotected	Unprotected	Any	Not allowed	Allowed	Allowed
Protected	Protected	Unprotected	Any	Not allowed	Allowed	Allowed
Any	Any	Protected	Unlocked	Not allowed	Not allowed	Allowed
Any	Any	Protected	Locked	Not allowed	Not allowed	Not allowed

Table 70: Erase protection

The debugger can read the erase protection status in the register [ERASEPROTECT.STATUS](#) on page 762.

The erase protection can be disabled through a cooperation between on-board firmware and debugger. If ERASEPROTECT has been enabled, both the debugger and on-chip firmware must write the same non-zero 32-bit KEY value into registers [ERASEPROTECT.DISABLE](#) on page 762 and [ERASEPROTECT.DISABLE](#) on page 768 respectively to disable the erase protection. When both registers have been written with the same non-zero 32-bit KEY value, the device is automatically erased as described in [Erase all](#) on page 757. The access ports will be re-enabled on the next reset once the secure erase sequence has completed.

The write-once register should be set to Locked as early as possible in the start-up sequence, preferably as soon as the on-chip firmware has determined it does not need to communicate with a debugger over the CTRL-AP mailbox interface. Once written, it will not be possible to remove the erase protection until the next next pin reset, power-on reset, brown-out reset, or watchdog timer reset.

9.7.3 Mailbox interface

CTRL-AP implements a mailbox interface which enables the CPU to communicate with a debugger over the SWD interface.

The mailbox interface consists of a transmit register [MAILBOX.TXDATA](#) on page 763 with its corresponding status register [MAILBOX.TXSTATUS](#) on page 763, and a receive register [MAILBOX.RXDATA](#) on page 763 with its corresponding status register [MAILBOX.RXSTATUS](#) on page 763. Status bits in registers TXSTATUS/RXSTATUS will be set and cleared automatically when registers TXDATA/RXDATA are written to and read from, independently of the direction.

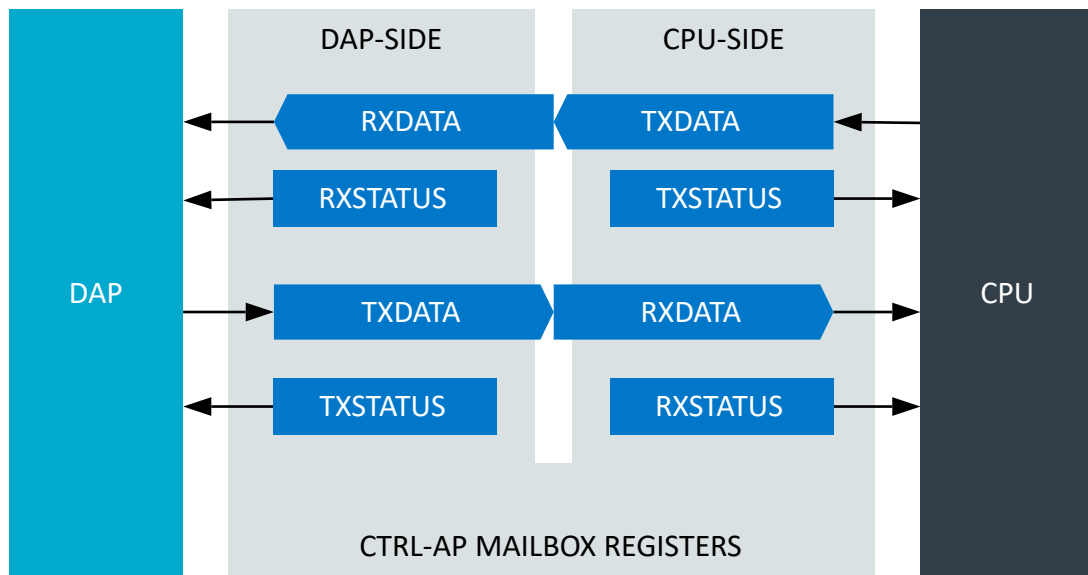


Figure 164: Mailbox register interface

Mailbox transfer sequence

1. Sender writes TXDATA
2. CTRL-AP sets sender's TXSTATUS to DataPending
3. CTRL-AP sets receiver's RXSTATUS to DataPending
4. Receiver reads RXDATA
5. CTRL-AP sets receiver's RXSTATUS to NoDataPending
6. CTRL-AP sets sender's TXSTATUS to NoDataPending

Events

EVENTS_RXREADY is generated when **MAILBOX.RXSTATUS** changes to DataPending. This indicates that a debugger has written new data to **MAILBOX.RXDATA**.

EVENTS_TXDONE is generated when the **MAILBOX.TXSTATUS** changes to NoDataPending. This indicates that a debugger has read the data from **MAILBOX.TXDATA**.

9.7.4 Device information

Device information such as part number and hardware revision can be read using CTRL-AP.

CTRL-AP provides the following information about the device:

- CTRL-AP identification register, IDR - See [IDR](#)
- Part number - See [INFO.PARTNO](#)
- Hardware revision - See [INFO.HWREVISION](#)

The information is available even for protected devices.

9.7.5 Debugger registers

CTRL-AP has a set of registers that can only be accessed from the debugger over the SWD interface. These are not accessible from the CPU.

9.7.5.1 Debug side registers

Register overview

Register	Offset	TZ	Description
RESET	0x000		System reset request and status
ERASEALL	0x004		Perform a secure erase of the device, where flash, SRAM, and UICR will be erased in sequence. The device will be returned to factory default settings upon next reset.
ERASEALLSTATUS	0x008		This is the status register for the ERASEALL operation.
ERASEPROTECT.STATUS	0x00C		Erase protection status.
ERASEPROTECT.DISABLE	0x010		This register disables ERASEPROTECT and performs Erase all.
APPROTECT.STATUS	0x014		This is the status register for the access port protection.
MAILBOX.TXDATA	0x020		Data sent from the debugger to the device.
MAILBOX.TXSTATUS	0x024		Status to indicate if data sent from the debugger to the device has been read.
MAILBOX.RXDATA	0x028		Data sent from the device to the debugger.
MAILBOX.RXSTATUS	0x02C		Status to indicate if data sent from the device to the debugger has been read.
INFO.PARTNO	0x030		Part number of the device
INFO.HWREVISION	0x034		Hardware Revision of the device
IDR	0x0FC		CTRL-AP Identification Register, IDR

9.7.5.1.1 RESET

Address offset: 0x000

System reset request and status

Only the enumerated values are supported, writing other values has unpredictable effect.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	W	RESET			Reset request																														
			NoReset	0	Write to release reset.																														
					Reading '0' means reset is not active.																														
			SoftReset	1	Write to hold device in soft reset. Use NoReset to release device from reset.																														
					Reading '1' means reset is active.																														
			HardReset	2	Write to hold device in hard reset. Use NoReset to release device from reset.																														
					Reading '2' means reset is active.																														
			PinReset	4	Write to hold device in pin reset. Use NoReset to release device from reset.																														
					Reading '4' means reset is active.																														

9.7.5.1.2 ERASEALL

Address offset: 0x004

Perform a secure erase of the device, where flash, SRAM, and UICR will be erased in sequence. The device will be returned to factory default settings upon next reset.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																				A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	W	ERASEALL				Return device to factory default settings																															
			NoOperation	0	No operation																																
			Erase	1	Erase flash, SRAM, and UICR in sequence																																

9.7.5.1.3 ERASEALLSTATUS

Address offset: 0x008

This is the status register for the ERASEALL operation.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	ERASEALLSTATUS			Status bits for the ERASEALL operation																														
			Ready	0	ERASEALL is ready																														
			ReadyToReset	1	Device is ready to be reset																														
			Busy	2	ERASEALL is busy (on-going)																														
			Error	3	Error during ERASEALL																														

9.7.5.1.4 ERASEPROTECT.STATUS

Address offset: 0x00C

Erase protection status.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	PALL			Erase protection status.																														
			Enabled	1	Erase protection is enabled.																														
			Disabled	0	Erase protection is not enabled and ERASEALL can be performed.																														

9.7.5.1.5 ERASEPROTECT.DISABLE

Address offset: 0x010

This register disables ERASEPROTECT and performs Erase all.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	KEY						The Erase all sequence will be initiated if value of the KEY fields are non-zero and the KEY fields match on both the CPU and debugger sides.																											

9.7.5.1.6 APPROTECT.STATUS

Address offset: 0x014

This is the status register for the access port protection.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																
ID				B																																A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																
ID	R/W	Field	Value ID	Value																																Description																															
A	R	APPROTECT																																		Status bit for access port protection.																															
			Enabled	1	APPROTECT is enabled																																																														
			Disabled	0	APPROTECT is disabled																																																														
B	R	SECUREAPPROTECT																																		This is the status register for the secure access port protection.																															
			Enabled	1	SECUREAPPROTECT is enabled																																																														
			Disabled	0	SECUREAPPROTECT is disabled																																																														

9.7.5.1.7 MAILBOX.TXDATA

Address offset: 0x020

Data sent from the debugger to the device.

Writing to this register will automatically set field DataPending in register TXSTATUS.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	RW	Data						Data sent from debugger																											

9.7.5.1.8 MAILBOX.TXSTATUS

Address offset: 0x024

Status to indicate if data sent from the debugger to the device has been read.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	Status			Status of register DATA																														
			NoDataPending	0	No data pending in register TXDATA																														
			DataPending	1	Data pending in register TXDATA																														

9.7.5.1.9 MAILBOX.RXDATA

Address offset: 0x028

Data sent from the device to the debugger.

Reading from this register will automatically set field NoDataPending in register RXSTATUS.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																														
ID				A A																														

9.7.5.1.10 MAILBOX.RXSTATUS

Address offset: 0x02C

Status to indicate if data sent from the device to the debugger has been read.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				A																																	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	R	Status				Status of register DATA																															
			NoDataPending	0	No data pending in register RXDATA																																
			DataPending	1	Data pending in register RXDATA																																

9.7.5.1.11 INFO.PARTNO

Address offset: 0x030

Part number of the device

This register is retained on system on idle.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	R	PARTNO						Part number																											

9.7.5.1.12 INFO.HWREVISION

Address offset: 0x034

Hardware Revision of the device

This register is retained on system on idle.

Bit number									31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID									A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID			Value				Description																															
A	R	ID								Part Variant																															

9.7.5.1.13 IDR

Address offset: 0x0FC

CTRL-AP Identification Register, IDR

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				E E E E D D D C C C C C C C B B B B A A A A A A A A																															
Reset 0x32880000				0 0 1 1 0 0 1 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	R	APID				AP Identification																													
B	R	CLASS				Access Port (AP) class																													
			NotDefined	0x0	No defined class																														
			MEMAP	0x8	Memory Access Port																														
C	R	JEP106ID				JEDEC JEP106 identity code																													
D	R	JEP106CONT				JEDEC JEP106 continuation code																													
E	R	REVISION				Revision																													

9.7.6 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
CTRLAP : S	GLOBAL	0x50052000	US	S	NSA	No	Control access port CPU side
CTRLAP : NS		0x40052000					

Register overview

Register	Offset	TZ	Description
EVENTS_RXREADY	0x100		RXSTATUS is changed to DataPending.
EVENTS_TXDONE	0x104		TXSTATUS is changed to NoDataPending.
INTEN	0x300		Enable or disable interrupt
INTENSET	0x304		Enable interrupt
INTENCLR	0x308		Disable interrupt
INTPEND	0x30C		Pending interrupts
MAILBOX.RXDATA	0x400		Data sent from the debugger to the CPU.
MAILBOX.RXSTATUS	0x404		Status to indicate if data sent from the debugger to the CPU has been read.
MAILBOX.TXDATA	0x480		Data sent from the CPU to the debugger.
MAILBOX.TXSTATUS	0x484		Status to indicate if data sent from the CPU to the debugger has been read.
ERASEPROTECT.LOCK	0x500		This register locks the ERASEPROTECT.DISABLE register from being written until next reset.
ERASEPROTECT.DISABLE	0x504		This register disables the ERASEPROTECT register and performs an ERASEALL operation.
RESET	0x520		System reset request.

9.7.6.1 EVENTS_RXREADY

Address offset: 0x100

RXSTATUS is changed to DataPending.

New data is available in MAILBOX.RXDATA.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																																							A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
ID	R/W	Field	Value	ID	Value	Description																																	
A	RW	EVENTS_RXREADY				RXSTATUS is changed to DataPending.																																	
						New data is available in MAILBOX.RXDATA.																																	
			NotGenerated	0		Event not generated																																	
			Generated	1		Event generated																																	

9.7.6.2 EVENTS_TXDONE

Address offset: 0x104

TXSTATUS is changed to NoDataPending.

MAILBOX.TXDATA has been read.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID																																							A
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value	Description																																		
A	RW	EVENTS_TXDONE			TXSTATUS is changed to NoDataPending.																																		
					MAILBOX.TXDATA has been read.																																		
			NotGenerated	0	Event not generated																																		
			Generated	1	Event generated																																		

9.7.6.3 INTEN

Address offset: 0x300

Enable or disable interrupt

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID																																				B	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
ID	R/W	Field	Value ID	Value		Description																															
A	RW	RXREADY				Enable or disable interrupt for event RXREADY																															
						New data is available in MAILBOX.RXDATA.																															
			Disabled	0		Disable																															
			Enabled	1		Enable																															
B	RW	TXDONE				Enable or disable interrupt for event TXDONE																															
						MAILBOX.TXDATA has been read.																															
			Disabled	0		Disable																															
			Enabled	1		Enable																															

9.7.6.4 INTENSET

Address offset: 0x304

Enable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	RXREADY			Write '1' to enable interrupt for event RXREADY																														
					New data is available in MAILBOX.RXDATA.																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	TXDONE			Write '1' to enable interrupt for event TXDONE																														
					MAILBOX.TXDATA has been read.																														
			Set	1	Enable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

9.7.6.5 INTENCLR

Address offset: 0x308

Disable interrupt

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	RXREADY			Write '1' to disable interrupt for event RXREADY																														
					New data is available in MAILBOX.RXDATA.																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														
B	RW	TXDONE			Write '1' to disable interrupt for event TXDONE																														
					MAILBOX.TXDATA has been read.																														
			Clear	1	Disable																														
			Disabled	0	Read: Disabled																														
			Enabled	1	Read: Enabled																														

9.7.6.6 INTPEND

Address offset: 0x30C

Pending interrupts

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	RXREADY			Read pending status of interrupt for event RXREADY																														
					New data is available in MAILBOX.RXDATA.																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														
B	R	TXDONE			Read pending status of interrupt for event TXDONE																														
					MAILBOX.TXDATA has been read.																														
			NotPending	0	Read: Not pending																														
			Pending	1	Read: Pending																														

9.7.6.7 MAILBOX.RXDATA

Address offset: 0x400

Data sent from the debugger to the CPU.

Reading from this register will automatically set field NoDataPending in register RXSTATUS.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	R	RXDATA																		Data received from debugger.															

9.7.6.8 MAILBOX.RXSTATUS

Address offset: 0x404

Status to indicate if data sent from the debugger to the CPU has been read.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																																							A
Reset 0x00000000				0 0																																			
ID	R/W	Field	Value ID	Value		Description																																	
A	R	RXSTATUS				Status of data in register RXDATA.																																	
			NoDataPending	0	No data is pending in register RXDATA.																																		
			DataPending	1	Data is pending in register RXDATA.																																		

9.7.6.9 MAILBOX.TXDATA

Address offset: 0x480

Data sent from the CPU to the debugger.

Writing to this register will automatically set field DataPending in register TXSTATUS.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value																Description															
A	RW	TXDATA																		Data sent to debugger.															

9.7.6.10 MAILBOX.TXSTATUS

Address offset: 0x484

Status to indicate if data sent from the CPU to the debugger has been read.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	R	TXSTATUS			Status of data in register TXDATA.																														
			NoDataPending	0	No data is pending in register TXDATA.																														
			DataPending	1	Data is pending in register TXDATA.																														

9.7.6.11 ERASEPROTECT.LOCK

Address offset: 0x500

This register locks the ERASEPROTECT.DISABLE register from being written until next reset.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
ID																																							A
Reset 0x00000000				0 0																																			
ID	R/W	Field	Value ID	Value		Description																																	
A	RW1	LOCK				Lock ERASEPROTECT.DISABLE register from being written until next reset.																																	
			Unlocked	0	Register ERASEPROTECT.DISABLE is writeable.																																		
			Locked	1	Register ERASEPROTECT.DISABLE is read-only.																																		

9.7.6.12 ERASEPROTECT.DISABLE

Address offset: 0x504

This register disables the ERASEPROTECT register and performs an ERASEALL operation.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A	W1	KEY						The ERASEALL sequence is initiated if the value of the KEY fields are non-zero and the KEY fields match on both the CPU and debugger sides.																											

9.7.6.13 RESET

Address offset: 0x520

System reset request.

Only the enumerated values are supported, writing other values has unpredictable effect.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	W	RESET						Reset request																											
			NoReset	0				No reset is generated																											
			SoftReset	1				Perform a device soft reset																											
			HardReset	2				Perform a device hard reset																											
			PinReset	4				Perform a device pin reset																											

9.8 TAD - Trace and debug control

Configuration interface for trace and debug

Please refer to the [Trace](#) section for more information about how to configure the trace and debug interface.

Note: Although there are PSEL registers for the trace port, each function can only be mapped to a single pin due to pin speed requirements. Setting the PIN field to anything else will not have any effect. See [Pin assignment chapter](#) for more information.

9.8.1 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
TAD : S	GLOBAL	0x50053000	US	S	NA	No	Empty instance abstract
TAD : NS		0x40053000					

Register overview

Register	Offset	TZ	Description
SYSPWRUPREQ	0x400		System power-up request
DBGPWRUPREQ	0x404		Debug power-up request
ENABLE	0x500		Enable debug domain and aquire selected GPIOs
TRACEPORTSPEED	0x518		Trace port speed
			This register is retained.
TINSTANCE	0x520		SW-DP Target instance

9.8.1.1 SYSPWRUPREQ

Address offset: 0x400

System power-up request

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	ACTIVE			Activate power-up request																														
			NotActive	0	Power-up request not active																														
			Active	1	Power-up request active																														

9.8.1.2 DBGPWRUPREQ

Address offset: 0x404

Debug power-up request

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ACTIVE						Activate power-up request																											
			NotActive	0				Power-up request not active																											
			Active	1				Power-up request active																											

9.8.1.3 ENABLE

Address offset: 0x500

Enable debug domain and aquire selected GPIOs

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ENABLE																																	
			DISABLED	0				Disable debug domain and release selected GPIOs																											
			ENABLED	1				Enable debug domain and aquire selected GPIOs																											

9.8.1.4 TRACEPORTSPEED (Retained)

Address offset: 0x518

Trace port speed

This register is retained.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	TRACEPORTSPEED			Trace port speed is divided from CPU clock. The TRACECLK pin output will be divided again by two from the trace port clock.																														
			DIV1	0	Trace port speed equals CPU clock																														
			DIV2	1	Trace port speed equals CPU clock divided by 2																														
			DIV4	2	Trace port speed equals CPU clock divided by 4																														
			DIV32	3	Trace port speed equals CPU clock divided by 32																														

9.8.1.5 TINSTANCE

Address offset: 0x520

SW-DP Target instance

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																			
ID																																					A	A	A	A
Reset 0x00000000					0 0																																			
ID	R/W	Field	Value ID	Value	Description																																			
A	RW	TINSTANCE			TINSTANCE bits are used in the SW-DP DLPIDR.TINSTANCE field.																																			

9.9 ETM — Embedded trace macrocell

The ARM embedded trace macrocell implements instruction, data and event tracing.

This document only provides a register-level description of this ARM component. See the [Arm® Embedded Trace Macrocell Architecture Specification](#) for more details

9.9.1 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
ETM	APPLICATION	0xE0041000	HF	NS	NA	No	Embedded trace macrocell

Register overview

Register	Offset	TZ	Description
TRCPRGCTLR	0x004		Enables the trace unit.

Register	Offset	TZ	Description
TRCPROCSCLR	0x008		Controls which PE to trace. Might ignore writes when the trace unit is enabled or not idle. Before writing to this register, ensure that TRCSTATR.IDLE == 1 so that the trace unit can synchronize with the chosen PE. Implemented if TRCIDR3.NUMPROC is greater than zero.
TRCSTATR	0x00C		Idle status bit
TRCCONFIGR	0x010		Controls the tracing options This register must always be programmed as part of trace unit initialization. Might ignore writes when the trace unit is enabled or not idle.
TRCEVENTCTLOR	0x20		Controls the tracing of arbitrary events. If the selected event occurs a trace element is generated in the trace stream according to the settings in TRCEVENTCTL1R.DATAEN and TRCEVENTCTL1R.INSTEN.
TRCEVENTCTL1R	0x24		Controls the behavior of the events that TRCEVENTCTLOR selects. This register must always be programmed as part of trace unit initialization. Might ignore writes when the trace unit is enabled or not idle.
TRCSTALLCTLR	0x2C		Enables trace unit functionality that prevents trace unit buffer overflows. Might ignore writes when the trace unit is enabled or not idle. Must be programmed if TRCIDR3.STALLCTL == 1.
TRCTSCTLR	0x30		Controls the insertion of global timestamps in the trace streams. When the selected event is triggered, the trace unit inserts a global timestamp into the trace streams. Might ignore writes when the trace unit is enabled or not idle. Must be programmed if TRCCONFIGR.TS == 1.
TRCSYNCPR	0x34		Controls how often trace synchronization requests occur. Might ignore writes when the trace unit is enabled or not idle. If writes are permitted then the register must be programmed.
TRCCCTLR	0x38		Sets the threshold value for cycle counting. Might ignore writes when the trace unit is enabled or not idle. Must be programmed if TRCCONFIGR.CCI==1.
TRCBCTLR	0x3C		Controls which regions in the memory map are enabled to use branch broadcasting. Might ignore writes when the trace unit is enabled or not idle. Must be programmed if TRCCONFIGR.BB == 1.
TRCTRACEIDR	0x40		Sets the trace ID for instruction trace. If data trace is enabled then it also sets the trace ID for data trace, to (trace ID for instruction trace) + 1. This register must always be programmed as part of trace unit initialization. Might ignore writes when the trace unit is enabled or not idle.
TRCQCTLR	0x44		Controls when Q elements are enabled. Might ignore writes when the trace unit is enabled or not idle. This register must be programmed if it is implemented and TRCCONFIGR.QE is set to any value other than 0b00.
TRCVICTLR	0x080		Controls instruction trace filtering. Might ignore writes when the trace unit is enabled or not idle. Only returns stable data when TRCSTATR.PMSTABLE == 1. Must be programmed, particularly to set the value of the SSSTATUS bit, which sets the state of the start/stop logic.

Register	Offset	TZ	Description
TRCVIIECTLR	0x084		ViewInst exclude control. Might ignore writes when the trace unit is enabled or not idle. This register must be programmed when one or more address comparators are implemented.
TRCVISSCTLR	0x088		Use this to set, or read, the single address comparators that control the ViewInst start/stop logic. The start/stop logic is active for an instruction which causes a start and remains active up to and including an instruction which causes a stop, and then the start/stop logic becomes inactive. Might ignore writes when the trace unit is enabled or not idle. If implemented then this register must be programmed.
TRCVIPCSCTLR	0x08C		Use this to set, or read, which PE comparator inputs can control the ViewInst start/stop logic. Might ignore writes when the trace unit is enabled or not idle. If implemented then this register must be programmed.
TRCVDCTLR	0x0A0		Controls data trace filtering. Might ignore writes when the trace unit is enabled or not idle. This register must be programmed when data tracing is enabled, that is, when either TRCCONFIGR.DA == 1 or TRCCONFIGR.DV == 1.
TRCVDSACCTLR	0x0A4		ViewData include / exclude control. Might ignore writes when the trace unit is enabled or not idle. This register must be programmed when one or more address comparators are implemented.
TRCVDARCCTLR	0x0A8		ViewData include / exclude control. Might ignore writes when the trace unit is enabled or not idle. This register must be programmed when one or more address comparators are implemented.
TRCSEQEVR[n]	0x100		Moves the sequencer state according to programmed events. Might ignore writes when the trace unit is enabled or not idle. When the sequencer is used, all sequencer state transitions must be programmed with a valid event.
TRCSEQRSTEV	0x118		Moves the sequencer to state 0 when a programmed event occurs. Might ignore writes when the trace unit is enabled or not idle. When the sequencer is used, all sequencer state transitions must be programmed with a valid event.
TRCSEQSTR	0x11C		Use this to set, or read, the sequencer state. Might ignore writes when the trace unit is enabled or not idle. Only returns stable data when TRCSTATR.PMSTABLE == 1. When the sequencer is used, all sequencer state transitions must be programmed with a valid event.
TRCEXTINSEL	0x120		Use this to set, or read, which external inputs are resources to the trace unit. Might ignore writes when the trace unit is enabled or not idle. Only returns stable data when TRCSTATR.PMSTABLE == 1. When the sequencer is used, all sequencer state transitions must be programmed with a valid event.
TRCNRDLVR[n]	0x140		This sets or returns the reload count value for counter n. Might ignore writes when the trace unit is enabled or not idle.
TRCNTCTLR[n]	0x150		Controls the operation of counter n. Might ignore writes when the trace unit is enabled or not idle.

Register	Offset	TZ	Description
TRCNCNTVR[n]	0x160		This sets or returns the value of counter n. The count value is only stable when TRCSTATR.PMSTABLE == 1. If software uses counter n then it must write to this register to set the initial counter value. Might ignore writes when the trace unit is enabled or not idle.
TRCRSCTLR[n]	0x200		Controls the selection of the resources in the trace unit. Might ignore writes when the trace unit is enabled or not idle. If software selects a non-implemented resource then CONSTRAINED UNPREDICTABLE behavior of the resource selector occurs, so the resource selector might fire unexpectedly or might not fire. Reads of the TRCRSCTLRn might return UNKNOWN.
TRCSSCCR0	0x280		Controls the single-shot comparator.
TRCSSCSR0	0x2A0		Indicates the status of the single-shot comparators. TRCSSCSR0 is sensitive to instruction addresses.
TRCSSPCICR0	0x2C0		Selects the processor comparator inputs for Single-shot control.
TRCPDCR	0x310		Controls the single-shot comparator.
TRCPDSR	0x314		Indicates the power down status of the ETM.
TRCITATBIDR	0xEE4		Sets the state of output pins.
TRCITIATBINR	0xEF4		Reads the state of the input pins.
TRCITIATBOUTr	0xEFC		Sets the state of the output pins.
TRCITCTRL	0xF00		Enables topology detection or integration testing, by putting ETM-M33 into integration mode.
TRCCCLAIMSET	0xFA0		Sets bits in the claim tag and determines the number of claim tag bits implemented.
TRCCCLAIMCLR	0xFA4		Clears bits in the claim tag and determines the current value of the claim tag.
TRCAUTHSTATUS	0xFB8		Indicates the current level of tracing permitted by the system
TRCDEVARCH	0xFBC		The TRCDEVARCH identifies ETM-M33 as an ETMv4.2 component
TRCDEVTYPE	0xFCC		Controls the single-shot comparator.
TRCPIDR[n]	0xFD0		Coresight peripheral identification registers.
TRCCIDR[n]	0xFF0		Coresight component identification registers.

9.9.1.1 TRCPRGCTLR

Address offset: 0x004

Enables the trace unit.

Bit number					31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID					A																															
Reset 0x00000000					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value	ID	Value	Description																														
A	RW	EN				Trace unit enable bit																														
			Disabled	0		The trace unit is disabled. All trace resources are inactive and no trace is generated.																														
			Enabled	1		The trace unit is enabled.																														

9.9.1.2 TRCPROCSCLR

Address offset: 0x008

Controls which PE to trace.

Might ignore writes when the trace unit is enabled or not idle.

Before writing to this register, ensure that TRCSTATR.IDLE == 1 so that the trace unit can synchronize with the chosen PE.

Implemented if TRCIDR3.NUMPROC is greater than zero.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																													
ID																												A											A	A	A	A																						
Reset 0x00000000				0																																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																																																								
A	RW	PROCSEL						PE select bits that select the PE to trace.																																																								

9.9.1.3 TRCSTATR

Address offset: 0x00C

Idle status bit

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	IDLE			Trace unit enable bit																														
			NotIdle	0	The trace unit is not idle.																														
			Idle	1	The trace unit is idle.																														
B	RW	PMSTABLE			Programmers' model stable bit																														
			NotStable	0	The programmers' model is not stable.																														
			Stable	1	The programmers' model is stable.																														

9.9.1.4 TRCCONFIGR

Address offset: 0x010

Controls the tracing options

This register must always be programmed as part of trace unit initialization.

Might ignore writes when the trace unit is enabled or not idle.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				M L K J J I H G G G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	LOADASPOINST			Instruction P0 load field. This field controls whether load instructions are traced as P0 instructions.																														
			No	0	Do not trace load instructions as P0 instructions.																														
			Yes	1	Trace load instructions as P0 instructions.																														
B	RW	STOREASPOINST			Instruction P0 field. This field controls whether store instructions are traced as P0 instructions.																														
			No	0	Do not trace store instructions as P0 instructions.																														
			Yes	1	Trace store instructions as P0 instructions.																														
C	RW	BB			Branch broadcast mode bit.																														
			Disabled	0	Branch broadcast mode is disabled.																														
			Enabled	1	Branch broadcast mode is enabled.																														
D	RW	CCI			Cycle counting instruction trace bit.																														
			Disabled	0	Cycle counting in the instruction trace is disabled.																														
			Enabled	1	Cycle counting in the instruction trace is enabled.																														
E	RW	CID			Context ID tracing bit.																														
			Disabled	0	Context ID tracing is disabled.																														
			Enabled	1	Context ID tracing is enabled.																														
F	RW	VMID			Virtual context identifier tracing bit.																														
			Disabled	0	Virtual context identifier tracing is disabled.																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				M L K J J I H G G G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
G	RW	COND	Enabled	1	Virtual context identifier tracing is enabled.																														
			Disabled	0	Conditional instruction tracing bit.																														
			LoadOnly	1	Conditional instruction tracing is disabled.																														
			StoreOnly	2	Conditional load instructions are traced.																														
			LoadAndStore	3	Conditional store instructions are traced.																														
			All	7	Conditional load and store instructions are traced.																														
H	RW	TS	Enabled	1	Global timestamp tracing bit.																														
			Disabled	0	Global timestamp tracing is disabled.																														
			Enabled	1	Global timestamp tracing is enabled.																														
I	RW	RS	Enabled	1	Return stack enable bit.																														
			Disabled	0	Return stack is disabled.																														
			Enabled	1	Return stack is enabled.																														
J	RW	QE	Enabled	3	Q element enable field.																														
			Disabled	0	Q elements are disabled.																														
			OnlyWithoutInstCou	1	Q elements with instruction counts are enabled. Q elements without instruction counts are disabled.																														
			Enabled	3	Q elements with and without instruction counts are enabled.																														
K	RW	VMIDOPT	VTTBR_EL2	0	Control bit to select the Virtual context identifier value used by the trace unit, both for trace generation and in the Virtual context identifier comparators.																														
			CONTEXTIDR_EL2	1	VTTBR_EL2.VMID is used. If the trace unit supports a Virtual context identifier larger than the VTTBR_EL2.VMID, the upper unused bits are always zero. If the trace unit supports a Virtual context identifier larger than 8 bits and if the VTCR_EL2.VS bit forces use of an 8-bit Virtual context identifier, bits [15:8] of the trace unit Virtual context identifier are always zero.																														
			CONTEXTIDR_EL2	1	CONTEXTIDR_EL2 is used.																														
L	RW	DA	Enabled	1	Data address tracing bit.																														
			Disabled	0	Data address tracing is disabled.																														
			Enabled	1	Data address tracing is enabled.																														
M	RW	DV	Enabled	1	Data value tracing bit.																														
			Disabled	0	Data value tracing is disabled.																														
			Enabled	1	Data value tracing is enabled.																														

9.9.1.5 TRCEVENTCTL0R

Address offset: 0x20

Controls the tracing of arbitrary events.

If the selected event occurs a trace element is generated in the trace stream according to the settings in TRCEVENTCTL1R.DATAEN and TRCEVENTCTL1R.INSTEN.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENT		[0:255]				Select which event should generate trace elements.																											

9.9.1.6 TRCEVENTCTL1R

Address offset: 0x24

Controls the behavior of the events that TRCEVENTCTL0R selects.

This register must always be programmed as part of trace unit initialization.

Might ignore writes when the trace unit is enabled or not idle.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																													
ID																																G F														E D C B A			
Reset 0x00000000				0 0																																													
ID	R/W	Field	Value ID	Value	Description																																												
A-D	RW	INSTEN[i] (i=0..3)			Instruction event enable field.																																												
			Disabled	0	The trace unit does not generate an Event element.																																												
			Enabled	1	The trace unit generates an Event element for event i, in the instruction trace stream.																																												
E	RW	DATAEN			Data event enable bit.																																												
			Disabled	0	The trace unit does not generate an Event element if event 0 occurs.																																												
			Enabled	1	The trace unit generates an Event element in the data trace stream if event 0 occurs.																																												
F	RW	ATB			AMBA Trace Bus (ATB) trigger enable bit.																																												
			Disabled	0	ATB trigger is disabled.																																												
			Enabled	1	ATB trigger is enabled. If a CoreSight ATB interface is implemented then when event 0 occurs the trace unit generates an ATB event.																																												
G	RW	LPOVERRIDE			Low-power state behavior override bit. Controls how a trace unit behaves in low-power state.																																												
			Disabled	0	Trace unit low-power state behavior is not affected. That is, the trace unit is enabled to enter low-power state.																																												
			Enabled	1	Trace unit low-power state behavior is overridden. That is, entry to a low-power state does not affect the trace unit resources or trace generation.																																												

9.9.1.7 TRCSTALLCTLR

Address offset: 0x2C

Enables trace unit functionality that prevents trace unit buffer overflows.

Might ignore writes when the trace unit is enabled or not idle.

Must be programmed if TRCIDR3.STALLCTL == 1.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																				G F E D C B				A A A A											
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	LEVEL		[15:0]	Threshold level field.																														
					If LEVEL is nonzero then a trace unit might suppress the generation of:																														
					Global timestamps in the instruction trace stream and the data trace stream.																														
					Cycle counting in the instruction trace stream, although the cumulative cycle count remains correct.																														
			Min	0	Zero invasion. This setting has a greater risk of a FIFO overflow																														
			Max	15	Maximum invasion occurs but there is less risk of a FIFO overflow.																														

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
B	RW	ISTALL			Instruction stall bit. Controls if a trace unit can stall the PE when the instruction trace buffer space is less than LEVEL.																														
			Disabled	0	The trace unit must not stall the PE.																														
			Enabled	1	The trace unit can stall the PE.																														
C	RW	DSTALL			Data stall bit. Controls if a trace unit can stall the PE when the data trace buffer space is less than LEVEL.																														
			Disabled	0	The trace unit must not stall the PE.																														
			Enabled	1	The trace unit can stall the PE.																														
D	RW	INSTPRIORITY			Prioritize instruction trace bit. Controls if a trace unit can prioritize instruction trace when the instruction trace buffer space is less than LEVEL.																														
			Disabled	0	The trace unit must not prioritize instruction trace.																														
			Enabled	1	The trace unit can prioritize instruction trace. A trace unit might prioritize instruction trace by preventing output of data trace, or other means which ensure that the instruction trace has a higher priority than the data trace.																														
E	RW	DATADISCARDLOAD			Data discard field. Controls if a trace unit can discard data trace elements on a load when the data trace buffer space is less than LEVEL.																														
			Disabled	0	The trace unit must not discard any data trace elements.																														
			Enabled	1	The trace unit can discard P1 and P2 elements associated with data loads.																														
F	RW	DATADISCARDSTORE			Data discard field. Controls if a trace unit can discard data trace elements on a store when the data trace buffer space is less than LEVEL.																														
			Disabled	0	The trace unit must not discard any data trace elements.																														
			Enabled	1	The trace unit can discard P1 and P2 elements associated with data stores.																														
G	RW	NOOVERFLOW			Trace overflow prevention bit.																														
			Disabled	0	Trace overflow prevention is disabled.																														
			Enabled	1	Trace overflow prevention is enabled. This might cause a significant performance impact.																														

9.9.1.8 TRCTSCTLR

Address offset: 0x30

Controls the insertion of global timestamps in the trace streams.

When the selected event is triggered, the trace unit inserts a global timestamp into the trace streams.

Might ignore writes when the trace unit is enabled or not idle.

Must be programmed if TRCCONFIGR.TS == 1.

Bit number					31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID					A A A A A A A A																															
Reset 0x00000000					0 0																															
ID	R/W	Field	Value ID	Value	Description																															
A	RW	EVENT		[0:255]	Select which event should generate time stamps.																															

9.9.1.9 TRCSYNCPR

Address offset: 0x34

Controls how often trace synchronization requests occur.

Might ignore writes when the trace unit is enabled or not idle.

If writes are permitted then the register must be programmed.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																																A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value		Description																													
A	RW	PERIOD		[31:0]		Controls how many bytes of trace, the sum of instruction and data, that a trace unit can generate before a trace synchronization request occurs. The number of bytes is always a power of two, calculated by 2^PERIOD																													
			Disabled	0		Trace synchronization requests are disabled. This setting does not disable other types of trace synchronization request.																													

9.9.1.10 TRCCCCTLR

Address offset: 0x38

Sets the threshold value for cycle counting.

Might ignore writes when the trace unit is enabled or not idle.

Must be programmed if TRCCONFIGR.CCI==1.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															

9.9.1.11 TRCBBCTLR

Address offset: 0x3C

Controls which regions in the memory map are enabled to use branch broadcasting.

Might ignore writes when the trace unit is enabled or not idle.

Must be programmed if TRCCONFIGR.BB == 1.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A-H	RW	RANGE[i] (i=0..7)				Address range field. Selects which address range comparator pairs are in use with branch broadcasting. Each field represents an address range comparator pair, so field[i] controls the selection of address range comparator pair i.																													
			Disabled	0		The address range that address range comparator pair i defines, is not selected.																													
			Enabled	1		The address range that address range comparator pair n defines, is selected.																													

9.9.1.12 TRCTRACEIDR

Address offset: 0x40

Sets the trace ID for instruction trace. If data trace is enabled then it also sets the trace ID for data trace, to (trace ID for instruction trace) + 1.

This register must always be programmed as part of trace unit initialization.

Might ignore writes when the trace unit is enabled or not idle.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value			Description																												
A	RW	TRACEID					Trace ID field. Sets the trace ID value for instruction trace.																												
							Bit[0] must be zero if data trace is enabled. If data trace is enabled then a trace unit sets the trace ID for data trace, to TRACEID+1.																												

9.9.1.13 TRCQCTLR

Address offset: 0x44

Controls when Q elements are enabled.

Might ignore writes when the trace unit is enabled or not idle.

This register must be programmed if it is implemented and TRCCONFIGR.QE is set to any value other than 0b00.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-H	RW	RANGE[i] (i=0..7)			Specifies the address range comparators to be used for controlling Q elements.																														
			Disabled	0	Address range comparator i is disabled.																														
			Enabled	1	Address range comparator i is selected for use.																														
I	RW	MODE			Selects whether the address range comparators selected by the RANGE field indicate address ranges where the trace unit is permitted to generate Q elements or address ranges where the trace unit is not permitted to generate Q elements:																														
			Exclude	0	Exclude mode. The address range comparators selected by the RANGE field indicate address ranges where the trace unit cannot generate Q elements. If no ranges are selected, Q elements are permitted across the entire memory map.																														
			Include	1	Include mode. The address range comparators selected by the RANGE field indicate address ranges where the trace unit can generate Q elements. If all the implemented bits in RANGE are set to 0 then Q elements are disabled.																														

9.9.1.14 TRCVICTLR

Address offset: 0x080

Controls instruction trace filtering.

Might ignore writes when the trace unit is enabled or not idle.

Only returns stable data when TRCSTATR.PMSTABLE == 1.

Must be programmed, particularly to set the value of the SSSTATUS bit, which sets the state of the start/stop logic.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
ID																				L				K	J	I	H	G	F	E					D				C	B					A				A	A	A	A																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																													
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				L K J I H G F E																D C B								A A A A							
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
			Disabled	0	This event is not filtered.																														
			Enabled	1	This event is filtered.																														
			Stopped	0	The start/stop logic is in the stopped state.																														
			Started	1	The start/stop logic is in the started state.																														
B	RW	SSSTATUS			When TRCIDR4.NUMACPAIRS > 0 or TRCIDR4.NUMPC > 0, this bit returns the status of the start/stop logic.																														
			Stopped	0	The start/stop logic is in the stopped state.																														
			Started	1	The start/stop logic is in the started state.																														
			Stopped	0	The start/stop logic is in the stopped state.																														
			Started	1	The start/stop logic is in the started state.																														
C	RW	TRCRESET			Controls whether a trace unit must trace a Reset exception.																														
			Disabled	0	The trace unit does not trace a Reset exception unless it traces the exception or instruction immediately prior to the Reset exception.																														
			Enabled	1	The trace unit always traces a Reset exception.																														
			Disabled	0	The trace unit does not trace a Reset exception unless it traces the exception or instruction immediately prior to the Reset exception.																														
			Enabled	1	The trace unit always traces a Reset exception.																														
D	RW	TRCERR			When TRCIDR3.TRCERR==1, this bit controls whether a trace unit must trace a System error exception.																														
			Disabled	0	The trace unit does not trace a System error exception unless it traces the exception or instruction immediately prior to the System error exception.																														
			Enabled	1	The trace unit always traces a System error exception, regardless of the value of ViewInst.																														
			Disabled	0	The trace unit does not trace a System error exception unless it traces the exception or instruction immediately prior to the System error exception.																														
			Enabled	1	The trace unit always traces a System error exception, regardless of the value of ViewInst.																														
E-H	RW	EXLEVEL[i]_S (i=0..3)			In Secure state, each bit controls whether instruction tracing is enabled for the corresponding Exception level i.																														
			Disabled	1	The trace unit does not generate instruction trace, in Secure state, for Exception level i.																														
			Enabled	0	The trace unit generates instruction trace, in Secure state, for Exception level i.																														
			Disabled	1	The trace unit does not generate instruction trace, in Secure state, for Exception level i.																														
			Enabled	0	The trace unit generates instruction trace, in Secure state, for Exception level i.																														
I-L	RW	EXLEVEL[i]_NS (i=0..3)			In Non-secure state, each bit controls whether instruction tracing is enabled for the corresponding Exception level i.																														
			Disabled	1	The trace unit does not generate instruction trace, in Non-secure state, for Exception level i.																														
			Enabled	0	The trace unit generates instruction trace, in Non-secure state, for Exception level i.																														
			Disabled	1	The trace unit does not generate instruction trace, in Non-secure state, for Exception level i.																														
			Enabled	0	The trace unit generates instruction trace, in Non-secure state, for Exception level i.																														

9.9.1.15 TRCVIIECTLR

Address offset: 0x084

ViewInst exclude control.

Might ignore writes when the trace unit is enabled or not idle.

This register must be programmed when one or more address comparators are implemented.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				P O N M L K J I																H G F E D C B A																		
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A-H	RW	INCLUDE[i] (i=0..7)				Include range field. Selects which address range comparator pairs are in use with ViewInst include control.																																
			Disabled	0	The address range that address range comparator pair i defines, is not selected for ViewInst include control.																																	
			Enabled	1	The address range that address range comparator pair i defines, is selected for ViewInst include control.																																	
I-P	RW	EXCLUDE[i] (i=0..7)				Exclude range field. Selects which address range comparator pairs are in use with ViewInst exclude control.																																
			Disabled	0	The address range that address range comparator pair i defines, is not selected for ViewInst exclude control.																																	
			Enabled	1	The address range that address range comparator pair i defines, is selected for ViewInst exclude control.																																	

9.9.1.16 TRCVISSCTLR

Address offset: 0x088

Use this to set, or read, the single address comparators that control the ViewInst start/stop logic. The start/stop logic is active for an instruction which causes a start and remains active up to and including an instruction which causes a stop, and then the start/stop logic becomes inactive.

Might ignore writes when the trace unit is enabled or not idle.

If implemented then this register must be programmed.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				P O N M L K J I																H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-H	RW	START[i] (i=0..7)			Selects which single address comparators are in use with ViewInst start/stop control, for the purpose of starting trace.																														
			Disabled	0	The single address comparator i, is not selected as a start resource.																														
			Enabled	1	The single address comparator i, is selected as a start resource.																														
I-P	RW	STOP[i] (i=0..7)			Selects which single address comparators are in use with ViewInst start/stop control, for the purpose of stopping trace																														
			Disabled	0	The single address comparator i, is not selected as a stop resource.																														
			Enabled	1	The single address comparator i, is selected as a stop resource.																														

9.9.1.17 TRCVIPCSSCTLR

Address offset: 0x08C

Use this to set, or read, which PE comparator inputs can control the ViewInst start/stop logic.

Might ignore writes when the trace unit is enabled or not idle.

If implemented then this register must be programmed.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
ID				P O N M L K J I																H G F E D C B A																		
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
ID	R/W	Field	Value ID	Value		Description																																
A-H	RW	START[i] (i=0..7)				Selects which PE comparator inputs are in use with ViewInst start/stop control, for the purpose of starting trace																																
			Disabled	0	The single PE comparator input i, is not selected as a start resource.																																	
			Enabled	1	The single PE comparator input i, is selected as a start resource.																																	
I-P	RW	STOP[i] (i=0..7)				Selects which PE comparator inputs are in use with ViewInst start/stop control, for the purpose of stopping trace.																																
			Disabled	0	The single PE comparator input i, is not selected as a stop resource.																																	
			Enabled	1	The single PE comparator input i, is selected as a stop resource.																																	

9.9.1.18 TRCVDCTLR

Address offset: 0x0A0

Controls data trace filtering.

Might ignore writes when the trace unit is enabled or not idle.

This register must be programmed when data tracing is enabled, that is, when either TRCCONFIGR.DA == 1 or TRCCONFIGR.DV == 1.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				L K J I I H G F E D C B A																															
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																														
A-H	RW	EVENT[i] (i=0..7)			Event unit enable bit.																														
			Disabled	0	The trace event is not selected for trace filtering.																														
			Enabled	1	The trace event is selected for trace filtering.																														
I	RW	SPREL			Controls whether a trace unit traces data for transfers that are relative to the Stack Pointer (SP).																														
			Enabled	0	The trace unit does not affect the tracing of SP-relative transfers.																														
			DataOnly	2	The trace unit does not trace the address portion of SP-relative transfers. If data value tracing is enabled then the trace unit generates a P1 data address element.																														
			Disabled	3	The trace unit does not trace the address or value portions of SP-relative transfers.																														
J	RW	PCREL			Controls whether a trace unit traces data for transfers that are relative to the Program Counter (PC).																														
			Enabled	0	The trace unit does not affect the tracing of PC-relative transfers.																														
			Disabled	1	The trace unit does not trace the address or value portions of PC-relative transfers.																														
K	RW	TBI			Controls which information a trace unit populates in bits[63:56] of the data address.																														
			SignExtend	0	The trace unit assigns bits[63:56] to have the same value as bit[55] of the data address, that is, it sign-extends the value.																														
			Copy	1	The trace unit assigns bits[63:56] to have the same value as bits[63:56] of the data address.																														
L	RW	TRCEXDATA			Controls the tracing of data transfers for exceptions and exception returns on Armv6-M, Armv7-M, and Armv8-M PEs.																														
			Disabled	0	Exception and exception return data transfers are not traced.																														
			Enabled	1	Exception and exception return data transfers are traced if the other aspects of ViewData indicate that the data transfers must be traced.																														

9.9.1.19 TRCVDSACCTLR

Address offset: 0x0A4

ViewData include / exclude control.

Might ignore writes when the trace unit is enabled or not idle.

This register must be programmed when one or more address comparators are implemented.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				P O N M L K J I																H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-H	RW	INCLUDE[i] (i=0..7)			Selects which single address comparators are in use with ViewData include control.																														
			Disabled	0	The single address comparator i, is not selected for ViewData include control.																														
			Enabled	1	The single address comparator i, is selected for ViewData include control.																														
I-P	RW	EXCLUDE[i] (i=0..7)			Selects which single address comparators are in use with ViewData exclude control.																														
			Disabled	0	The single address comparator i, is not selected for ViewData exclude control.																														
			Enabled	1	The single address comparator i, s selected for ViewData exclude control.																														

9.9.1.20 TRCVDARCCTLR

Address offset: 0x0A8

ViewData include / exclude control.

Might ignore writes when the trace unit is enabled or not idle.

This register must be programmed when one or more address comparators are implemented.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				P O N M L K J I																H G F E D C B A															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-H	RW	INCLUDE[i] (i=0..7)			Include range field. Selects which address range comparator pairs are in use with ViewData include control.																														
			Disabled	0	The address range that address range comparator i defines, is not selected for ViewData include control.																														
			Enabled	1	The address range that address range comparator i defines, is selected for ViewData include control.																														
I-P	RW	EXCLUDE[i] (i=0..7)			Exclude range field. Selects which address range comparator pairs are in use with ViewData exclude control.																														
			Disabled	0	The address range that address range comparator i defines, is not selected for ViewData exclude control.																														
			Enabled	1	The address range that address range comparator i defines, s selected for ViewData exclude control.																														

9.9.1.21 TRCSEQEVR[n] (n=0..2)

Address offset: 0x100 + (n × 0x4)

Moves the sequencer state according to programmed events.

Might ignore writes when the trace unit is enabled or not idle.

When the sequencer is used, all sequencer state transitions must be programmed with a valid event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				P O N M L K J I H G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-H	RW	F[i] (i=0..7)			Forward field.																														
			Disabled	0	The trace event does not affect the sequencer.																														
			Enabled	1	When the event occurs then the sequencer state moves from state n to state n+1.																														
I-P	RW	B[i] (i=0..7)			Backward field.																														
			Disabled	0	The trace event does not affect the sequencer.																														
			Enabled	1	When the event occurs then the sequencer state moves from state n+1 to state n.																														

9.9.1.22 TRCSEQRSTEV

Address offset: 0x118

Moves the sequencer to state 0 when a programmed event occurs.

Might ignore writes when the trace unit is enabled or not idle.

When the sequencer is used, all sequencer state transitions must be programmed with a valid event.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EVENT		[0:255]				Select which event should reset the sequencer.																											

9.9.1.23 TRCSEQSTR

Address offset: 0x11C

Use this to set, or read, the sequencer state.

Might ignore writes when the trace unit is enabled or not idle.

Only returns stable data when TRCSTATR.PMSTABLE == 1.

When the sequencer is used, all sequencer state transitions must be programmed with a valid event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
ID				A																																A	
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
ID	R/W	Field	Value ID	Value		Description																															
A	RW	STATE				Sets or returns the state of the sequencer.																															
			State0	0	The sequencer is in state 0.																																
			State1	1	The sequencer is in state 1.																																
			State2	2	The sequencer is in state 2.																																
			State3	3	The sequencer is in state 3.																																

9.9.1.24 TRCEXTINSEL

Address offset: 0x120

Use this to set, or read, which external inputs are resources to the trace unit.

Might ignore writes when the trace unit is enabled or not idle.

Only returns stable data when TRCSTATR.PMSTABLE == 1.

When the sequencer is used, all sequencer state transitions must be programmed with a valid event.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D	D	D	D	D	D	D	D	C	C	C	C	C	C	C	C	B	B	B	B	B	B	B	B	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																											
A-D	RW	SEL[i] (i=0..3)		[0:255]				Each field in this collection selects an external input as a resource for the trace unit.																											

9.9.1.25 TRCCNTRLDVR[n] (n=0..3)

Address offset: 0x140 + (n × 0x4)

This sets or returns the reload count value for counter n.

Might ignore writes when the trace unit is enabled or not idle.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ID																			A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value				Description																												
A	RW	VALUE		[0:65535]				Contains the reload value for counter n. When a reload event occurs for counter n then the trace unit copies the VALUEn field into counter n.																												

9.9.1.26 TRCCNTCTLR[n] (n=0..3)

Address offset: 0x150 + (n × 0x4)

Controls the operation of counter n.

Might ignore writes when the trace unit is enabled or not idle.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D C B B B B B B B A A A A A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	CNTEVENT		[0:255]	Selects an event, that when it occurs causes counter n to decrement.																														
B	RW	RLDEVENT		[0:255]	Selects an event, that when it occurs causes a reload event for counter n.																														
C	RW	RLDSELF			Controls whether a reload event occurs for counter n, when counter n reaches zero.																														
			Disabled	0	The counter is in Normal mode.																														
			Enabled	1	The counter is in Self-reload mode.																														
D	RW	CNTCHAIN			For TRCCNTCTLR3 and TRCCNTCTLR1, this bit controls whether counter n decrements when a reload event occurs for counter n-1.																														
			Disabled	0	Counter n does not decrement when a reload event for counter n-1 occurs.																														
			Enabled	1	Counter n decrements when a reload event for counter n-1 occurs. This concatenates counter n and counter n-1, to provide a larger count value.																														

9.9.1.27 TRCCNTVR[n] (n=0..3)

Address offset: 0x160 + (n × 0x4)

This sets or returns the value of counter n.

The count value is only stable when TRCSTATR.PMSTABLE == 1.

If software uses counter n then it must write to this register to set the initial counter value.

Might ignore writes when the trace unit is enabled or not idle.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	VALUE		[0:65535]				Contains the count value of counter n.																											

9.9.1.28 TRCRSCTLR[n] (n=2..31)

Address offset: 0x200 + (n × 0x4)

Controls the selection of the resources in the trace unit.

Might ignore writes when the trace unit is enabled or not idle.

If software selects a non-implemented resource then CONSTRAINED UNPREDICTABLE behavior of the resource selector occurs, so the resource selector might fire unexpectedly or might not fire. Reads of the TRCRSCTLRn might return UNKNOWN.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	EN						Trace unit enable bit																											
			Disabled	0				The trace unit is disabled. All trace resources are inactive and no trace is generated.																											
			Enabled	1				The trace unit is enabled.																											

9.9.1.29 TRCSSCCR0

Address offset: 0x280

Controls the single-shot comparator.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	RST						Enables the single-shot comparator resource to be reset when it occurs, to enable another comparator match to be detected																											
			Disabled	0				Multiple matches can not be detected.																											
			Enabled	1				Multiple matches can occur.																											

9.9.1.30 TRCSSCSR0

Address offset: 0x2A0

Indicates the status of the single-shot comparators. TRCSSCSR0 is sensitive to instruction addresses.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																			
ID				E																																D C B A			
Reset 0x00000000				0 0																																			
ID	R/W	Field	Value ID	Value	Description																																		
A	RW	INST			Instruction address comparator support																																		
			False	0	Single-shot instruction address comparisons not supported.																																		
			True	1	Single-shot instruction address comparisons supported.																																		
B	RW	DA			Data address comparator support																																		
			False	0	Data address comparisons not supported.																																		
			True	1	Data address comparisons supported.																																		
C	RW	DV			Data value comparator support																																		
			False	0	Data value comparisons not supported.																																		
			True	1	Data value comparisons supported.																																		
D	RW	PC			Process counter value comparator support																																		
			False	0	Process counter value comparisons not supported.																																		
			True	1	Process counter value comparisons supported.																																		
E	RW	STATUS			Single-shot status. This indicates whether any of the selected comparators have matched.																																		
			NoMatch	0	Match has not occurred.																																		
			Match	1	Match has occurred at least once.																																		

9.9.1.31 TRCSSPCICR0

Address offset: 0x2C0

Selects the processor comparator inputs for Single-shot control.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-D	RW	PC[i] (i=0..3)			Selects processor comparator i inputs for Single-shot control																														
			Disabled	0	Processor comparator i is not selected for Single-shot control.																														
			Enabled	1	Processor comparator i is selected for Single-shot control.																														

9.9.1.32 TRCPDCR

Address offset: 0x310

Controls the single-shot comparator.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	PU			Power up request, to request that power to ETM and access to the trace registers is maintained.																														
			Disabled	0	Power not requested.																														
			Enabled	1	Power requested.																														

9.9.1.33 TRCPDSR

Address offset: 0x314

Indicates the power down status of the ETM.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A	RW	POWER			Indicates ETM is powered up																														
			NotPoweredUp	0	ETM is not powered up. All registers are not accessible.																														
			PoweredUp	1	ETM is powered up. All registers are accessible.																														
B	RW	STICKYPD			Sticky power down state.																														
					This bit is set to 1 when power to the ETM registers is removed, to indicate that programming state has been lost. It is cleared after a read of the TRCPDSR																														
			NotPoweredDown	0	Trace register power has not been removed since the TRCPDSR was last read.																														
			PoweredDown	1	Trace register power has been removed since the TRCPDSR was last read.																														

9.9.1.34 TRCITATBIDR

Address offset: 0xEE4

Sets the state of output pins.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				G F E D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A-G	RW	ID[i] (i=0..6)						Drives the ATIDMI[i] output pin.																											

9.9.1.35 TRCITIATBINR

Address offset: 0xEF4

Reads the state of the input pins.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ATVALID						Returns the value of the ATVALIDMI input pin.																											
B	RW	AFREADY						Returns the value of the AFREADYMI input pin.																											

9.9.1.36 TRCITIATBOUTr

Address offset: 0xEFC

Sets the state of the output pins.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											
A	RW	ATVALID						Drives the ATVALIDMI output pin.																											
B	RW	AFREADY						Drives the AFREADYMI output pin.																											

9.9.1.37 TRCITCTRL

Address offset: 0xF00

Enables topology detection or integration testing, by putting ETM-M33 into integration mode.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	RW	IME				Integration mode enable																													
			Disabled	0		ETM is not in integration mode.																													
			Enabled	1		ETM is in integration mode.																													

9.9.1.38 TRCCLAIMSET

Address offset: 0xFA0

Sets bits in the claim tag and determines the number of claim tag bits implemented.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-D	RW	SET[i] (i=0..3)			Claim tag set register																														
			NotSet	0	Claim tag i is not set.																														
			Set	1	Claim tag i is set.																														
			Claim	1	Set claim tag i.																														

9.9.1.39 TRCCLAIMCLR

Address offset: 0xFA4

Clears bits in the claim tag and determines the current value of the claim tag.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D C B A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value	Description																														
A-D	RW	CLR[i] (i=0..3)			Claim tag clear register																														
			NotSet	0	Claim tag i is not set.																														
			Set	1	Claim tag i is set.																														
			Clear	1	Clear claim tag i.																														

9.9.1.40 TRCAUTHSTATUS

Address offset: 0xFB8

Indicates the current level of tracing permitted by the system

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				D D C C B B A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value								Description																							
A	RW	NSID										Non-secure Invasive Debug																							

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																												D	D	C	C	B	B	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																														
B	RW	NSNID	NotImplemented	0	The feature is not implemented.																														
			Implemented	1	The feature is implemented.																														
			Non-secure Non-Invasive Debug																																
			NotImplemented	0	The feature is not implemented.																														
C	RW	SID	Implemented	1	The feature is implemented.																														
			Secure Invasive Debug																																
			NotImplemented	0	The feature is not implemented.																														
			Implemented	1	The feature is implemented.																														
D	RW	SNID	Secure Non-Invasive Debug																																
			NotImplemented	0	The feature is not implemented.																														
			Implemented	1	The feature is implemented.																														

9.9.1.41 TRCDEVARCH

Address offset: 0xFBC

The TRCDEVARCH identifies ETM-M33 as an ETMv4.2 component

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				D	D	D	D	D	D	D	D	D	D	C	B	B	B	B	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ID	R/W	Field	Value ID	Value	Description																														
A	R	ARCHID			Architecture ID																														
			ETMv42	0x4A13	Component is an ETMv4 component																														
B	R	REVISION			Architecture revision																														
			v2	2	Component is part of architecture 4.2																														
C	R	PRESENT			This register is implemented																														
			Absent	0	The register is not implemented.																														
			Present	1	The register is implemented.																														
D	R	ARCHITECT			Defines the architect of the component																														
			Arm	0x23B	This peripheral was architected by Arm.																														

9.9.1.42 TRCDEVTYPE

Address offset: 0xFCC

Controls the single-shot comparator.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID				B B B B A A A A																															
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value		Description																													
A	R	MAJOR				The main type of the component																													
			TraceSource	3	Peripheral is a trace source.																														
B	R	SUB				The sub-type of the component																													
			ProcessorTrace	1	Peripheral is a processor trace source.																														

9.9.1.43 TRCPIDR[n] (n=0..7)

Address offset: 0xFD0 + (n × 0x4)

Coresight peripheral identification registers.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																																			
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											

9.9.1.44 TRCCIDR[n] (n=0..3)

Address offset: 0xFF0 + (n × 0x4)

Coresight component identification registers.

Bit number				31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
ID																																			
Reset 0x00000000				0 0																															
ID	R/W	Field	Value ID	Value				Description																											

9.10 TPIU — Trace port interface unit

The Arm Cortex-M33 TPIU bridges the on-chip trace data from the ETM and the ITM, with separate IDs, to a data stream.

The Arm Cortex-M33 TPIU encapsulates IDs where required, and an external Trace Port Analyzer (TPA) captures the data stream. See the [Arm Cortex-M33 Processor Technical Reference Manual](#) for more details.

9.10.1 Registers

Instances

Instance	Domain	Base address	TrustZone			Split access	Description
			Map	Att	DMA		
TPIU	APPLICATION	0xE0040000	HF	NS	NA	No	Trace port interface unit (Trace and Debug)

10 Hardware and layout

10.1 Pin assignments

The pin assignment figures and tables describe the pinouts for the product variants of the device.

As a general rule, peripherals must use GPIO pins in their own domain for all peripheral functions when selected in the PSEL register. Dedicated clock pin requirements are listed in [Clock pins](#) on page 794. In addition, there are some dedicated pin functions that allow pin connections between different power domains.

The block diagram shows which peripheral and port belong together, see [Block diagram](#) on page 15.

GPIO ports have their own properties. For details, see [GPIO — General purpose input/output](#) on page 273.

10.1.1 Dedicated pins

The device has some pins dedicated for specific purposes. GPIO pin routing and configuration is flexible. Some pins have limitations or recommendations for configuration and use.

Peripheral	Description
UARTE20/21	Can use any pin on P1. Can connect across power domains to dedicated pins on P2.
SPIM00	Has dedicated pins on P2. For 32 MHz operation, the pins must be configured using extra high drive E0/E1 configuration in the DRIVE0/1 fields of the PIN_CNF GPIO register.
SPIM20/21	Can use any pins on P1; see notes on clock pins . Can be connect across power domains to dedicated pins on P2.
SPIS20/21	Can use any pins on P1; see notes on clock pins . Can connect across power domains to dedicated pins on P2.
TRACE	Has dedicated pins that must be configured using extra high drive E0/E1 configuration in the DRIVE0/1 fields of the PIN_CNF GPIO register.
GRTC	Has dedicated pins for clock and PWM output.
TAMPC	Has dedicated pins for active shield input and output.
FLPR	Uses dedicated pins on P2 for emulated peripherals such as QSPI.
RADIO	Uses dedicated pins on P1 for antenna switch control (DFEGPIO for direction finding).
NFC	Uses dedicated pins as listed in the pin assignments table for the selected device. These pins are configured as NFC antenna pins from reset. To use the pins for Digital I/O, NFC function must be disabled in the NFCT — Near field communication tag on page 350 peripheral.

Table 71: Dedicated pin functions

Cross power-domain notes

Selected pins of P2 can be used for some serial interfaces in the peripheral domain — SPIM, SPIS, and UARTE. This is not the most power-efficient way of connecting these serial interfaces, but adds flexibility when designing a circuit board. When setting up the peripheral's PSEL register for cross-domain connections, it must be connected only to the corresponding function listed in the pin assignments table. For example, the peripheral's PSEL.SCK register must use the P2 SCK pin from the pin assignment table. The pin assignments table shows which pins can be configured for cross power-domain connections.

10.1.2 Clock pins

The device has dedicated clock pins.

Some peripherals have clock signals. Dedicated clock pins have been optimized to ensure correct timing relationship between clock and data signals for these peripherals. See the following table for which peripheral signals must use clock pins. The pin assignment table identifies clock pins.

Clock pins can also be used as regular I/O data pins.

The peripheral data signal must be configured to use pins close to the clock pin. This ensures that the internal paths from the peripheral to the pin have the same delay, so that the data and clock signals reach the pins at the same time.

For high-speed signals, the printed circuit board (PCB) layout must use short PCB traces of identical length. This makes sure any delays are kept to a minimum, with close to identical delay on the clock and data path.

The following table shows which peripheral signals must use clock pins.

Peripheral	Signal	Clock pin required
SPIM/SPIS	SDO	
	SDI	
	SCK	x
	CSN	
	DCX	
TWIM/TWIS	SDA	
	SCL	x
PDM	DIN	
	CLK	x
I2S	MCK	x
	LRCK	
	SCK	X
	SDIN	
	SDOUT	
TRACE	TRACEDATA[]	
	TRACECLK	x (dedicated pin)
GRTC	CLKOUT32K	x (dedicated pin)
	PWMOUT	x (dedicated pin)
	CLK16M	x (dedicated pin)

Table 72: List of peripheral signals and clock pin requirement

10.1.3 QFN48 pin assignments

The QFN48 pin assignment figure and table describe the pinouts for this variant of the device.

Pins that can be used as clock signals are shown in the figure with the pin number in red color, and in the tables with a cross in the "Clock pin" column. For more information about clock pins, see [Clock pins](#) on page 794.

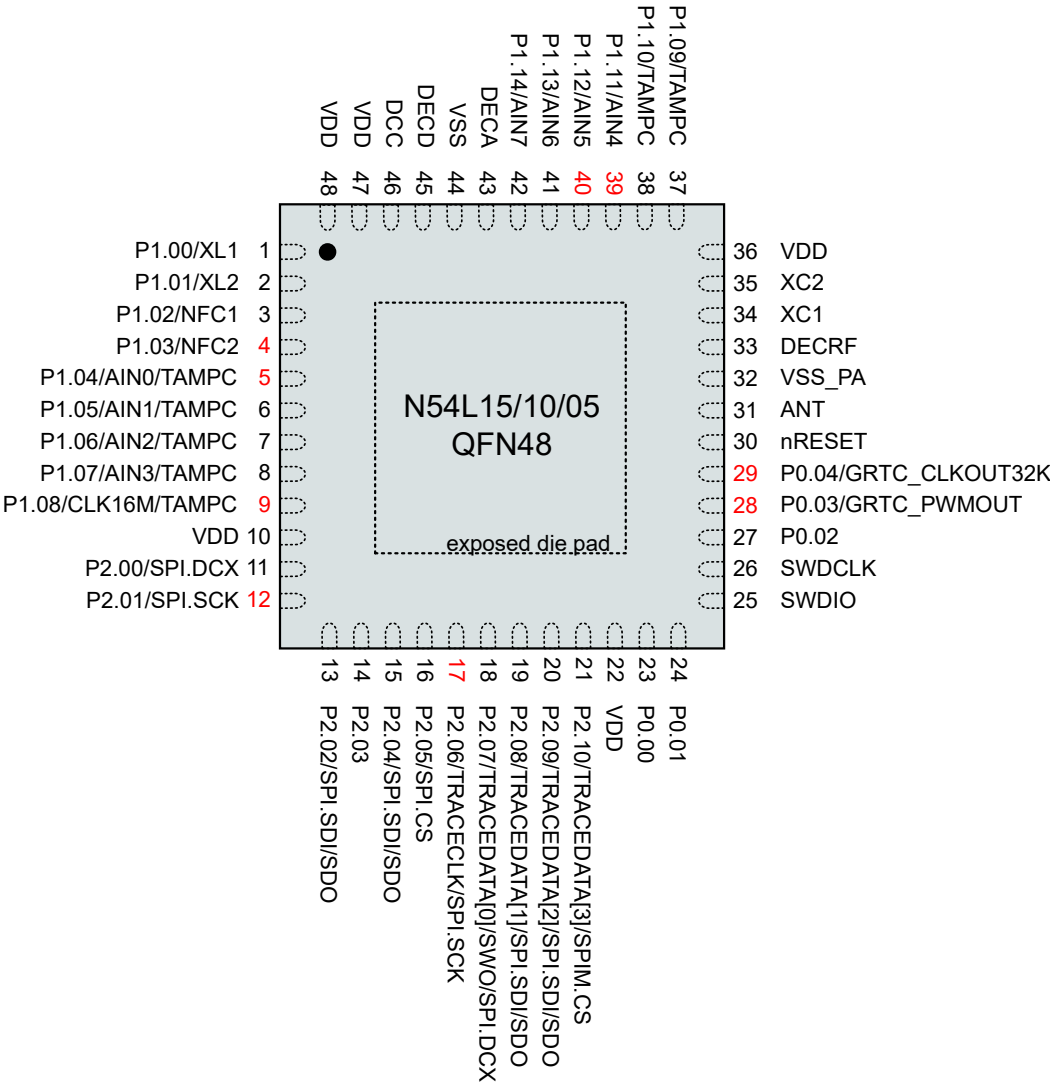


Figure 165: QFN48 pin assignments, top view

Pin	Clock pin	Name	Function	Description	Dedicated function
1		P1 . 00 XL1	Digital I/O Analog input	General purpose I/O Connection for 32.768 kHz crystal	
2		P1 . 01 XL2	Digital I/O Analog input	General purpose I/O Connection for 32.768 kHz crystal	
3		P1 . 02 NFC1	Digital I/O NFC input	General purpose I/O NFC antenna connection	
4	x	P1 . 03 NFC2	Digital I/O NFC input	General purpose I/O NFC antenna connection	
5	x	P1 . 04 ASO [0] AIN0	Digital I/O Digital I/O Analog input	General purpose I/O TAMPC active shield 0 output Analog input	TAMPC
6		P1 . 05 ASI [0] RADIO [6] AIN1	Digital I/O Digital I/O Digital I/O Analog input	General purpose I/O TAMPC active shield 0 input RADIO DFEGPIO Analog input	TAMPC RADIO
7		P1 . 06 ASO [1] AIN2	Digital I/O Digital I/O Analog input	General purpose I/O TAMPC active shield 1 output Analog input	TAMPC
8		P1 . 07 ASI [1] AIN3	Digital I/O Digital I/O Analog input	General purpose I/O TAMPC active shield 1 input Analog input	TAMPC
9	x	P1 . 08 CLK16M EXTREF	Digital I/O Digital I/O Analog input	General purpose I/O GRTC HF clock output External reference for SAADC	
10		VDD	Power	Power supply	
11		P2 . 00	Digital I/O Digital I/O Digital I/O	General purpose I/O SPIM DCX UARTE RXD	SPIM00/20 UARTE00/20

Pin	Clock pin	Name	Function	Description	Dedicated function
			Digital I/O	QSPI D3	FLPR (QSPI)
12	x	P2 . 01	Digital I/O Digital I/O Digital I/O Digital I/O	General purpose I/O SPIM SCK SPIS SCK QSPI SCK	SPIM00/20 SPIS00/S20 FLPR
13		P2 . 02	Digital I/O Digital I/O Digital I/O Digital I/O Digital I/O Digital I/O	General purpose I/O SPIM SDO SPIS SDO UARTE TXD QSPI D0 Serial wire output (SWO)	SPIM00/20 SPIS00/20 UARTE00/20 FLPR Trace
14		P2 . 03	Digital I/O Digital I/O	General purpose I/O QSPI D2	FLPR
15		P2 . 04	Digital I/O Digital I/O Digital I/O Digital I/O Digital I/O	General purpose I/O SPIM SDI SPIS SDI UARTE CTS QSPI D1	SPIM00/20 SPIS00/20 UARTE00/20 FLPR
16		P2 . 05	Digital I/O Digital I/O Digital I/O Digital I/O	General purpose I/O SPIM CS UARTE RTS QSPI CS	SPIM00/20 UARTE00/20 FLPR
17	x	P2 . 06 TRACECLK	Digital I/O Digital I/O Digital I/O Digital I/O	General purpose I/O SPIM SCK SPIS SCK Trace clock	SPIM00/21 SPIS20/21 Trace
18		P2 . 07 TRACEDATA [0] SWO	Digital I/O Digital I/O Digital I/O Digital I/O	General purpose I/O Trace data Serial wire output (SWO) SPIM DCX	Trace Trace SPIM00/21

Pin	Clock pin	Name	Function	Description	Dedicated function
			Digital I/O	UARTE RXD	UARTE00/21
19		P2 . 08	Digital I/O	General purpose I/O	
		TRACEDATA [1]	Digital I/O	Trace data	Trace
			Digital I/O	SPIM SDO	SPIM00/21
			Digital I/O	SPIS SDO	SPIS00/21
			Digital I/O	UARTE TXD	UARTE00/21
20		P2 . 09	Digital I/O	General purpose I/O	
		TRACEDATA [2]	Digital I/O	Trace data	Trace
			Digital I/O	SPIM SDI	SPIM00/21
			Digital I/O	SPIS SDI	SPIS00/21
			Digital I/O	UARTE CTS	UARTE00/21
21		P2 . 10	Digital I/O	General purpose I/O	
		TRACEDATA [3]	Digital I/O	Trace data	Trace
			Digital I/O	SPIM CS	SPIM00/21
			Digital I/O	UARTE RTS	UARTE00/21
22		VDD	Power	Power supply	
23		P0 . 00	Digital I/O	General purpose I/O	
24		P0 . 01	Digital I/O	General purpose I/O	
25		SWDIO	Debug	Serial wire data. Bidirectional with standard-drive and on-chip pull-down.	
26		SWDCLK	Debug	Serial wire clock. Input with on-chip pull-up.	
27		P0 . 02	Digital I/O	General purpose I/O	
28	x	P0 . 03	Digital I/O	General purpose I/O	GRTC
		GRTCPWM	Digital I/O	GRTC PWM output	
29	x	P0 . 04	Digital I/O	General purpose I/O	GRTC
		GRTCLFCLKOUT	Digital I/O	GRTC LF clock output	
30		nRESET	Reset	Pin reset with on-chip pull-up	

Pin	Clock pin	Name	Function	Description	Dedicated function
31		ANT	RF	Single ended radio antenna connection	See Reference circuitry on page 808 for guidelines on how to ensure good RF performance
32		VSS_PA	Power	Ground (radio supply)	
33		DECRF	Power	0.9 V regulator supply decoupling	Must be connected to DECA. See Reference circuitry on page 808.
34		XC1	Analog input	Connection for 32 MHz crystal	
35		XC2	Analog input	Connection for 32 MHz crystal	
36		VDD	Power	Power supply	
37		P1.09 ASO[2] RADIO[0]	Digital I/O Digital I/O Digital I/O	General purpose I/O TAMPC active shield 2 output RADIO DFEGPIO	TAMPC RADIO
38		P1.10 ASI[2] RADIO[1]	Digital I/O Digital I/O Digital I/O	General purpose I/O TAMPC active shield 2 input RADIO DFEGPIO	TAMPC RADIO
39	x	P1.11 ASO[3] RADIO[2] AIN4	Digital I/O Digital I/O Digital I/O Analog input	General purpose I/O TAMPC active shield 3 output RADIO DFEGPIO Analog input	TAMPC RADIO
40	x	P1.12 ASI[3] RADIO[3] AIN5	Digital I/O Digital I/O Digital I/O Analog input	General purpose I/O TAMPC active shield 3 input RADIO DFEGPIO Analog input	TAMPC RADIO

Pin	Clock pin	Name	Function	Description	Dedicated function
41		P1 . 13 RADIO [4] AIN6	Digital I/O Digital I/O Analog input	General purpose I/O RADIO DFEGPIO Analog input	RADIO
42		P1 . 14 RADIO [5] AIN7	Digital I/O Digital I/O Analog input	General purpose I/O RADIO DFEGPIO Analog input	RADIO
43		DECA	Power	0.9 V regulator supply decoupling	Must be connected to DECRF
44		VSS	Power	Ground	
45		DECD	Power	0.9 V regulator supply decoupling	
46		DCC	Power	DC/DC regulator output	
47		VDD	Power	Power supply	
48		VDD	Power	Power supply	
49		VSS	Power	Ground pad (die pad)	

Table 73: QFN48 pin assignments

For the device to function properly, exposed die pad (pin 49) must be connected to ground (**VSS**, pins 32 and 44).

10.1.4 WLCSP300 pin assignments

The WLCSP300 pin assignment figure and table describe the pinouts for this variant of the device.

Pins that can be used as clock signals are shown in the figure with the pin number in red color, and in the tables with a cross in the "Clock pin" column. For more information about clock pins, see [Clock pins](#) on page 794.

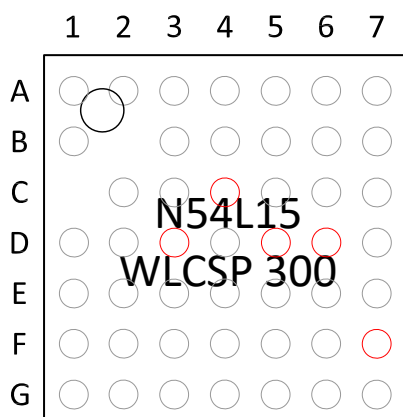


Figure 166: WLCSP300 pin assignments, top view

Pin	Clock pin	Name	Function	Description	Dedicated function
A1		XC1	Analog input	Connection for 32 MHz crystal	
A2		XC2	Analog input	Connection for 32 MHz crystal	
A3	x	P1 . 12 ASI [3] RADIO [3] AIN5	Digital I/O Digital I/O Digital I/O Analog input	General purpose I/O TAMPC active shield 3 input RADIO DFEGPIO Analog input	TAMPC RADIO
A4		DECA	Power	0.9 V regulator supply decoupling	Must be connected to DECRF
A5		DECD	Power	0.9 V regulator supply decoupling	
A6		VSS	Power	Ground	
A7		DCC	Power	DC/DC regulator output	
B1		DECRF	Power	0.9 V regulator supply decoupling	Must be connected to DECA. See Reference circuitry on page 808.
B3		P1 . 09 ASO [2] RADIO [0]	Digital I/O Digital I/O Digital I/O	General purpose I/O TAMPC active shield 2 output RADIO DFEGPIO	TAMPC RADIO
B4		P1 . 13 RADIO [4] AIN6	Digital I/O Digital I/O Analog input	General purpose I/O RADIO DFEGPIO Analog input	RADIO
B5		P1 . 14 RADIO [5] AIN7	Digital I/O Digital I/O Analog input	General purpose I/O RADIO DFEGPIO Analog input	RADIO
B6		P1 . 15	Digital I/O	General purpose I/O	
B7		VDD	Power	Power supply	
C2		VSS_PA	Power	Ground (radio supply)	

Pin	Clock pin	Name	Function	Description	Dedicated function
C3		P1 . 10 ASI [2] RADIO [1]	Digital I/O Digital I/O Digital I/O	General purpose I/O TAMPC active shield 2 input RADIO DFEGPIO	TAMPC RADIO
C4	x	P1 . 11 ASO [3] RADIO [2] AIN4	Digital I/O Digital I/O Digital I/O Analog input	General purpose I/O TAMPC active shield 3 output RADIO DFEGPIO Analog input	TAMPC RADIO
C5		P1 . 00 XL1	Digital I/O Analog input	General purpose I/O Connection for 32.768 kHz crystal	
C6		P1 . 01 XL2	Digital I/O Analog input	General purpose I/O Connection for 32.768 kHz crystal	
C7		P1 . 02 NFC1	Digital I/O NFC input	General purpose I/O NFC antenna connection	
D1		ANT	RF	Single ended radio antenna connection	See Reference circuitry on page 808 for guidelines on how to ensure good RF performance
D2		nRESET	Reset	Pin reset with on-chip pull-up	
D3	x	P0 . 04 GRTCLFCLKOUT	Digital I/O Digital I/O	General purpose I/O GRTC LF clock output	GRTC
D4		P2 . 08 TRACEDATA [1]	Digital I/O Digital I/O Digital I/O Digital I/O	General purpose I/O Trace data SPIM SDO SPIS SDO UARTE TXD	Trace SPIM00/21 SPIS00/21 UARTE00/21
D5	x	P1 . 03 NFC2	Digital I/O NFC input	General purpose I/O NFC antenna connection	
D6	x	P1 . 04	Digital I/O	General purpose I/O	

Pin	Clock pin	Name	Function	Description	Dedicated function
		ASO [0]	Digital I/O	TAMPC active shield 0 output	TAMPC
		AIN0	Analog input	Analog input	
D7		P1 . 06	Digital I/O	General purpose I/O	
		ASO [1]	Digital I/O	TAMPC active shield 1 output	TAMPC
		AIN2	Analog input	Analog input	
E1	x	P0 . 03	Digital I/O	General purpose I/O	
		GRTCPWM	Digital I/O	GRTC PWM output	GRTC
E2		P0 . 02	Digital I/O	General purpose I/O	
E3		SWDCLK	Debug	Serial wire clock. Input with on-chip pull-up.	
E4		P2 . 07	Digital I/O	General purpose I/O	
		TRACEDATA [0]	Digital I/O	Trace data	Trace
		SWO	Digital I/O	Serial wire output (SWO)	Trace
			Digital I/O	SPIM DCX	SPIM00/21
			Digital I/O	UARTE RXD	UARTE00/21
E5		P2 . 03	Digital I/O	General purpose I/O	
			Digital I/O	QSPI D2	FLPR
E6		P1 . 05	Digital I/O	General purpose I/O	
		ASI [0]	Digital I/O	TAMPC active shield 0 input	TAMPC
		RADIO [6]	Digital I/O	RADIO DFEGPIO	RADIO
		AIN1	Analog input	Analog input	
E7		P1 . 07	Digital I/O	General purpose I/O	
		ASI [1]	Digital I/O	TAMPC active shield 1 input	TAMPC
		AIN3	Analog input	Analog input	
F1		P0 . 01	Digital I/O	General purpose I/O	
F2		SWDIO	Debug	Serial wire data. Bidirectional with standard-drive and on-chip pull-down.	

Pin	Clock pin	Name	Function	Description	Dedicated function
F3		P2 . 09 TRACEDATA [2]	Digital I/O	General purpose I/O	
			Digital I/O	Trace data	Trace
			Digital I/O	SPIM SDI	SPIM00/21
			Digital I/O	SPIS SDI	SPIS00/21
			Digital I/O	UARTE CTS	UARTE00/21
F4	x	P2 . 06	Digital I/O	General purpose I/O	
			Digital I/O	SPIM SCK	SPIM00/21
			Digital I/O	SPIS SCK	SPIS20/21
		TRACECLK	Digital I/O	Trace clock	Trace
F5		P2 . 04	Digital I/O	General purpose I/O	
			Digital I/O	SPIM SDI	SPIM00/20
			Digital I/O	SPIS SDI	SPIS00/20
			Digital I/O	UARTE CTS	UARTE00/20
			Digital I/O	QSPI D1	FLPR
F6		P2 . 02	Digital I/O	General purpose I/O	
			Digital I/O	SPIM SDO	SPIM00/20
			Digital I/O	SPIS SDO	SPIS00/20
			Digital I/O	UARTE TXD	UARTE00/20
			Digital I/O	QSPI D0	FLPR
			Digital I/O	Serial wire output (SWO)	Trace
F7	x	P1 . 08	Digital I/O	General purpose I/O	
		CLK16M	Digital I/O	GRTC HF clock output	
		EXTREF	Analog input	External reference for SAADC	
G1		P0 . 00	Digital I/O	General purpose I/O	
G2		P2 . 10 TRACEDATA [3]	Digital I/O	General purpose I/O	
			Digital I/O	Trace data	Trace
			Digital I/O	SPIM CS	SPIM00/21
			Digital I/O	UARTE RTS	UARTE00/21
G3		VDD	Power	Power supply	

Pin	Clock pin	Name	Function	Description	Dedicated function
G4		VSS	Power	Ground	
G5		P2.05	Digital I/O Digital I/O Digital I/O Digital I/O	General purpose I/O SPIM CS UARTE RTS QSPI CS	SPIM00/20 UARTE00/20 FLPR
G6		P2.00	Digital I/O Digital I/O Digital I/O Digital I/O	General purpose I/O SPIM DCX UARTE RXD QSPI D3	SPIM00/20 UARTE00/20 FLPR (QSPI)
G7	x	P2.01	Digital I/O Digital I/O Digital I/O Digital I/O	General purpose I/O SPIM SCK SPIS SCK QSPI SCK	SPIM00/20 SPIS00/S20 FLPR

Table 74: WLCSP300 pin assignments

For the device to function properly, exposed die pad (pin 49) must be connected to ground (VSS, pins 32 and 44).

10.2 Mechanical specifications

The mechanical specifications for the packages show the dimensions in millimeters.

10.2.1 QFN48 6x6 mm package

Dimensions in millimeters for the QFN48 6x6 mm package.

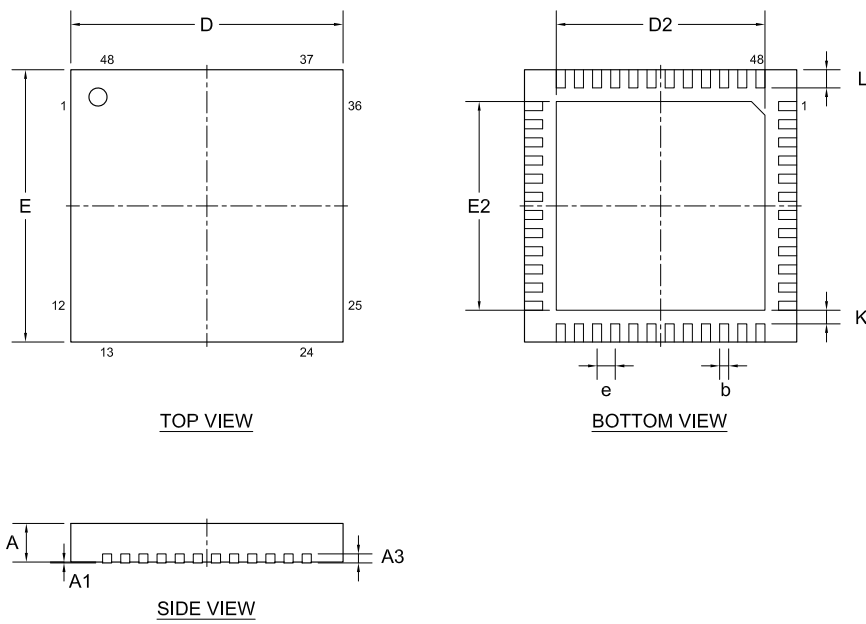


Figure 167: QFN48 6x6 mm package

	A	A1	A3	b	D, E	D2, E2	e	K	L
Min.	0.80	0.00		0.15		4.50		0.20	0.35
Nom.	0.85	0.04	0.20	0.20	6.00	4.60	0.40		0.40
Max.	0.90	0.05		0.25		4.70			0.45

Table 75: QFN48 dimensions in millimeters

10.2.2 WLCSP 300 μ m pitch package

Dimensions in millimeters for the WLCSP 300 μ m pitch package.

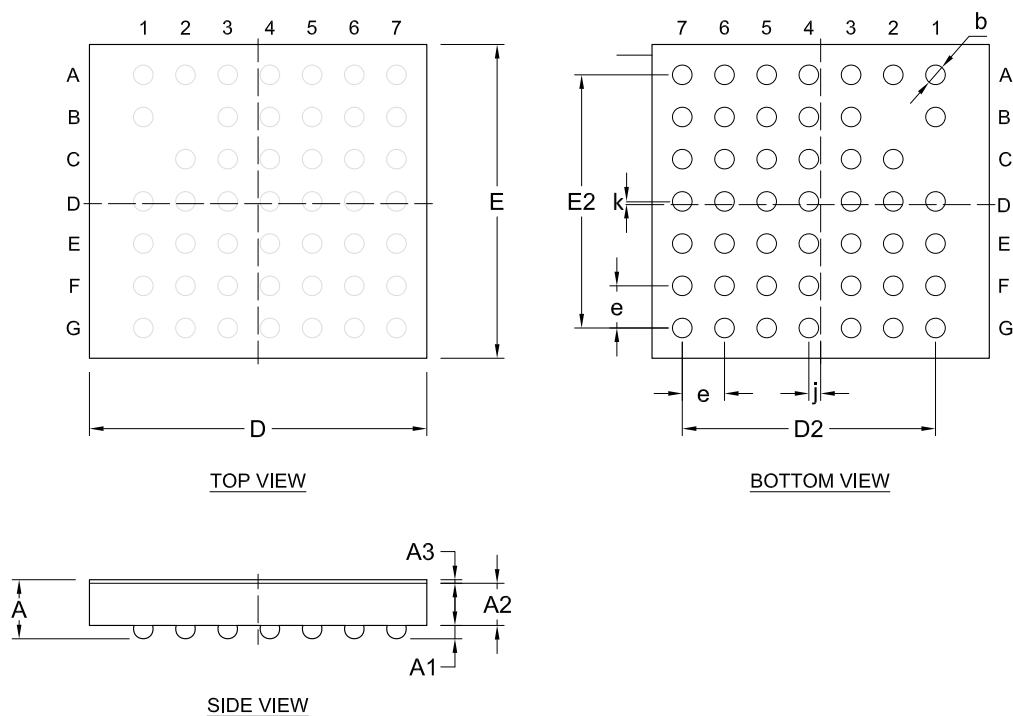


Figure 168: WLCSP 300 μm pitch package

	A	A1	A2	A3	D	E	D2	E2	e	b	j	k
Min.	0.362	0.065	0.275	0.022	2.421	2.215				0.14		
Nom.	0.42		0.3	0.025	2.451	2.245	1.8	1.8	0.3		0.072	0.013
Max.	0.478	0.125	0.325	0.028	2.481	2.275				0.2		

Table 76: WLCSP 300 μm pitch package dimensions in millimeters

10.3 Reference circuitry

To ensure good RF performance when designing PCBs, it is highly recommended to use the PCB layouts and component values provided by Nordic Semiconductor.

Documentation for the different package reference circuits, including Altium Designer files, PCB layout files, and PCB production files can be downloaded from www.nordicsemi.com.

In this section, there are reference circuits for all product variants, showing the components and component values to support on-chip features in a design.

10.3.1 Circuit configuration 1 for QFAA QFN-48

Config no.	Supply configuration	Enabled features
	VDD	NFC
Config 1	Battery/External regulator	No

Table 77: Circuit configurations for QFN48

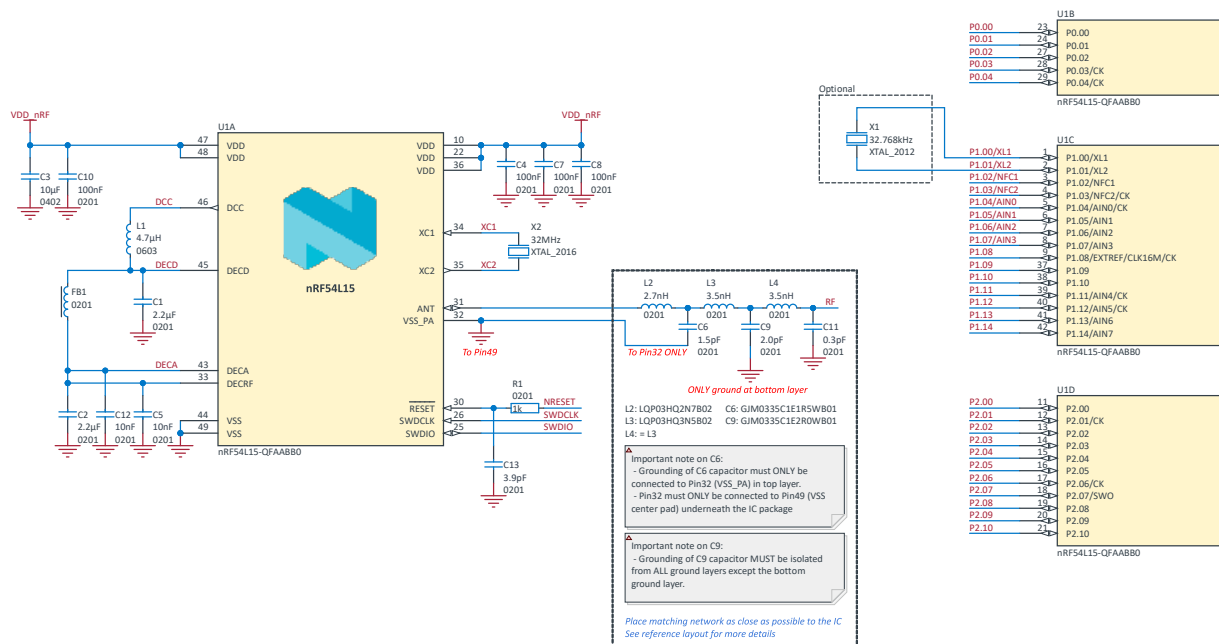


Figure 169: Circuit configuration 1 schematic

Note: For PCB reference layouts, see the product page for the device on www.nordicsemi.com.

Designator	Value	Description	Footprint
C1, C2	2.2 μ F	Capacitor, X6T, $\pm 20\%$, 2.5 V	0201
C3	10 μ F	Capacitor, X6S, $\pm 20\%$, 6.3 V	0402
C4, C7, C8, C10	100 nF	Capacitor, X7R, $\pm 10\%$	0201
C5, C12	10 nF	Capacitor, X7R, 6.3 V	0201
C6	1.5 pF	Capacitor, NP0, ± 0.05 pF, 25 V, High Q	0201
C9	2.0 pF	Capacitor, NP0, ± 0.05 pF, 25 V	0201
C11	0.3 pF	Capacitor, COG, ± 0.01 pF, 50 V	0201
C13	3.9 pF	Capacitor, COG, ± 0.25 pF, 50 V	0201
FB1	120 Ω	Ferrite bead, 120 Ω at 100 MHz, 200 mA, 500 m Ω Max	0201
L1	4.7 μ H	Inductor, 120 mA, $\pm 20\%$, 650 m Ω	0603
L2	2.7 nH	Inductor, 600 mA, ± 0.1 nH, 120 m Ω	0201
L3, L4	3.5 nH	Inductor, 500 mA, ± 0.1 nH, 170 m Ω	0201
R1	1 k Ω	Resistor, $\pm 1\%$, 0.05 W	0201
U1	nRF54L15-QFAA	Multiprotocol Bluetooth Low Energy, IEEE 802.15.4, and 2.4GHz proprietary System on Chip	QFN-48
X1	32.768 kHz	Crystal SMD 2012, 32.768 kHz, CI = 9 pF, Total tol: ± 20 ppm	XTAL_2012
X2	32 MHz	Crystal SMD 2016, 32 MHz, CI = 8 pF, Total Tol: ± 40 ppm. For frequency tolerance requirements, see 32 MHz crystal oscillator (HFXO) on page 820.	XTAL_2016

Table 78: Bill of material for circuit configuration 1

Note: The antenna filtering components are subject to change.

10.3.2 Circuit configuration 1 for CAAA WLCSP 300 μ m pitch

Config no.	Supply configuration	Enabled features
	VDD	NFC
Config 1	Battery/External regulator	No

Table 79: Circuit configurations for WLCSP 300 μ m

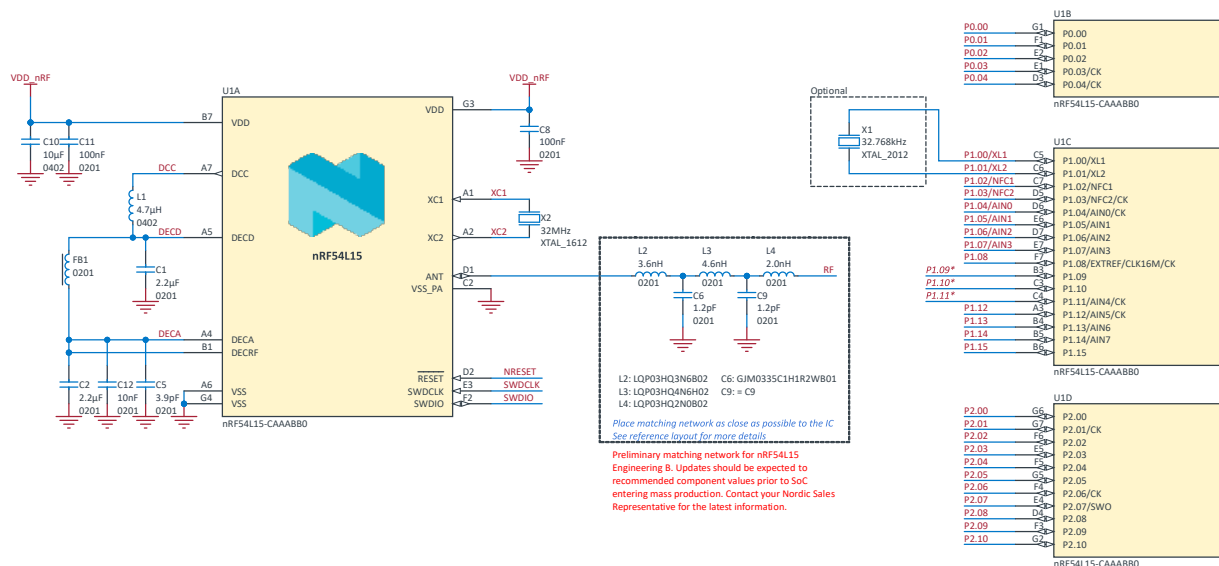


Figure 170: Circuit configuration 1 schematic

Note: For PCB reference layouts, see the product page for the device on www.nordicsemi.com.

Designator	Value	Description	Footprint
C1, C2	2.2 μ F	Capacitor, X6T, $\pm 20\%$, 2.5 V	0201
C5	3.9 pF	Capacitor, NPO, ± 0.05 pF, 25 V	0201
C6, C9	1.2 pF	Capacitor, NPO, ± 0.05 pF, 50 V	0201
C8, C11	100 nF	Capacitor, X7R, $\pm 10\%$	0201
C10	10 μ F	Capacitor, X6S, $\pm 20\%$, 6.3 V	0402
C12	10 nF	Capacitor, X7R, 6.3 V	0201
FB1	120 Ω	Ferrite bead, 120 Ω at 100 MHz, 200 mA, 500 m Ω Max	0201
L1	4.7 μ H	Inductor, 300 mA, $\pm 20\%$, 1.08 Ω	0402
L2	3.6 nH	Inductor, 400 mA, ± 0.1 nH, 170 m Ω	0201
L3	4.6 nH	Inductor, 300 mA, $\pm 3\%$, 250 m Ω	0201
L4	2.0 nH	Inductor, 600 mA, ± 0.1 nH, 120 m Ω	0201
U1	nRF54L15-CAAA	Multiprotocol Bluetooth Low Energy, IEEE 802.15.4, and 2.4GHz proprietary System on Chip	CSP300
X1	32.768 kHz	Crystal SMD 2012, 32.768 kHz, CI = 9 pF, Total tol: ± 20 ppm	XTAL_2012
X2	32 MHz	Crystal SMD 1612, 32 MHz, CI = 8 pF, Total Tol: ± 40 ppm, Aging ± 1 ppm/year. For frequency tolerance requirements, see 32 MHz crystal oscillator (HFXO) on page 820.	XTAL_1612

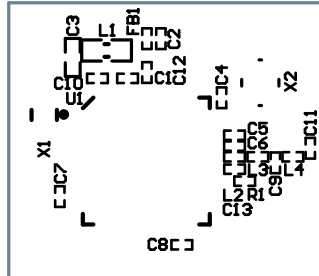
Table 80: Bill of material for circuit configuration 1

Note: The antenna filtering components are subject to change.

10.3.3 PCB layout example

The PCB layout in the following figure is a reference layout for Circuit configuration no. 1 for QFAA QFN48. Notice how the capacitor C6 is grounded. It is not directly connected to the ground plane, but grounded via pin 32 and to the VSS die pad. This is done to create additional filtering of harmonic components.

For all available reference layouts, see the product page for nRF54L15/10/05 on www.nordicsemi.com.



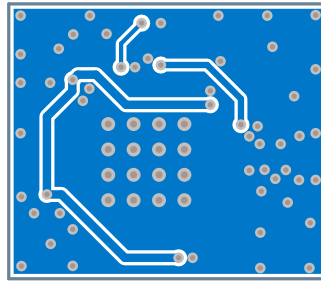


Figure 175: Bottom layer

10.3.4 PMIC support

The nRF54L Series is comprehensively supported by Nordic Semiconductor's own range of PMICs (Power Management Integrated Circuits). These PMICs are meticulously designed to enhance the performance and efficiency of the nRF54L Series devices. This integration ensures the longest battery life and the highest reliability for the end application. The synergy between the nRF54L Series and the Nordic PMICs highlights Nordic Semiconductor's commitment to providing a complete and cohesive solution for their customers' needs in wireless technology applications.

10.4 Reflow conditions

The maximum amount of reflow is three, with each reflow at 260°C and 30 seconds.

11 Electrical specification

11.1 Current consumption

Because the power and clock management system is constantly adjusting the different power and clock sources, estimating an application's current consumption can be challenging. To facilitate the estimation process, a set of current consumption scenarios is provided to show the typical current drawn from the supply pins.

Each scenario specifies a set of operations and conditions applying to the given scenario. All scenarios are listed in the following sections.

11.1.1 Conditions

The following table shows a set of common conditions used in all scenarios, unless otherwise stated.

Condition	Value	Note
Supply	3 V on VDD	
Temperature	25°C	
CPU	WFI (wait for interrupt)/WFE (wait for event) sleep	
Peripherals	All IDLE	
Clock	HFCLK = HFINT running at 128 MHz LFCLK = Not running	
Regulator	DC/DC	
RAM	16 kB	In System ON, RAM value refers to the amount of RAM that is powered. The remainder of RAM is powered off and not retained. In System OFF, RAM value refers to amount of RAM that is retained.
External components	As reference circuitry	See Reference circuitry on page 808 for details.
Cache enabled	Yes	Only applies when the CPU is running from non-volatile memory.
Compiler version	GCC version 10.3.1 20210621	
Compiler flags	<code>-O0 -fno-strict-aliasing -fno-delete-null-pointer-checks -fomit-frame-pointer -ffunction-sections -fmax-errors=1 -mcpu=cortex-m33 -mthumb -falign-functions=16 -mcmse</code>	

Table 81: Current consumption scenarios, common conditions

11.1.2 CURRENT Electrical specification

11.1.2.1 Sleep

Symbol	Description	Min.	Typ.	Max.	Units
I _{OFF0}	System OFF, Wake on pin, 0 KB RAM retained		0.6		uA
I _{OFF1}	System OFF, Wake on pin + GRTC, LFXO, 0 KB RAM retained		0.8		uA
I _{ON_IDLE0}	System ON, Wake on pin, 0 KB RAM retained		0.7		uA
I _{ON_IDLE1}	System ON, Wake on pin, 64 KB RAM retained		1.3		uA
I _{ON_IDLE2}	System ON, Wake on pin, 96 KB RAM retained		1.5		uA
I _{ON_IDLE3}	System ON, Wake on pin, 128 KB RAM retained		1.8		uA
I _{ON_IDLE4}	System ON, Wake on pin, 192 KB RAM retained		2.4		uA
I _{ON_IDLE5}	System ON, Wake on pin, 256 KB RAM retained		3.0		uA
I _{ON_IDLE6}	System ON, Wake on pin + GRTC, LFXO, 64 KB RAM retained		1.5		uA
I _{ON_IDLE7}	System ON, Wake on pin + GRTC, LFXO, 128 KB RAM retained		2.0		uA
I _{ON_IDLE8}	System ON, Wake on pin + GRTC, LFXO, 256 KB RAM retained		3.1		uA
I _{ON_IDLE9}	System ON, Wake on pin + GRTC, LFRC, 256 KB RAM retained		3.7		uA

11.1.2.2 CPU running

The CPU running parameters are obtained using the following compiler version:

Compiler: Arm version 6.16 (armclang)

Compiler flags:

```
--target=arm-arm-none-eabi -c -g -masm=auto -Wno-unused-value -mcpu=cortex-m33 -mfloat-abi=hard -mfpu=fpv5-sp-d16 -flto -Omax
```

Linker flags:

```
--lto --remove
```

Symbol	Description	Min.	Typ.	Max.	Units
I _{APPCPU0}	CPU running Coremark @ 128 MHz from NVM, Cache enabled		2.6		mA
I _{APPCPU1}	CPU running Coremark @ 128 MHz from RAM, Cache disabled		2.9		mA

11.1.2.3 RADIO transmitting/receiving

Symbol	Description	Min.	Typ.	Max.	Units
I _{RADIO_RX0}	Radio RX @ 1 Mbps, HFXO		3.3		mA
I _{RADIO_RX1}	Radio RX @ 2 Mbps, HFXO		3.6		mA
I _{RADIO_TX0}	Radio TX @ 0 dBm, HFXO		4.8		mA
I _{RADIO_TX1}	Radio TX @ 4 dBm, HFXO		6.6		mA
I _{RADIO_TX2}	Radio TX @ 8 dBm, HFXO		9.8		mA

11.1.2.4 SAADC active

Symbol	Description	Min.	Typ.	Max.	Units
I _{SAADC0}	SAADC @ 2 Msps, HFXO, 256 KB RAM powered		1.4		mA

11.1.2.5 TIMER active

Symbol	Description	Min.	Typ.	Max.	Units
I _{TIMER0}	TIMER00 running @ 128MHz, 16 KB RAM powered		450		uA
I _{TIMER1}	TIMER20 running @ 16MHz, 16 KB RAM powered		142		uA
I _{TIMER2}	TIMER20 running @ 1MHz, 16 KB RAM powered		121		uA
I _{TIMER3}	TIMER10 running @ 32MHz, HFXO, 16 KB RAM powered		240		mA

11.2 CLOCK Electrical specification

11.2.1 High frequency clock source (HFCLK)

Symbol	Description	Min.	Typ.	Max.	Units
f _{NOM}	Nominal output frequency		64/128		MHz

11.2.2 32.768 kHz clock source (LFCLK)

Symbol	Description	Min.	Typ.	Max.	Units
f _{NOM_LFCLK}	Nominal output frequency		32.768		kHz

11.3 COMP Electrical specification

11.3.1 COMP Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
t _{PROPDLY,LP}	Propagation delay, low-power mode ²		0.5		μs
t _{PROPDLY,HS}	Propagation delay, high-speed mode ²		0.1		μs
I _{SOURCE}	Configurable input current provided by the output driven current source.				μA
I _{SOURCE,A}	Current when register ISOURCE=len2uA5	1.4	2.2	3.6	μA
I _{SOURCE,B}	Current when register ISOURCE=len5uA	3.3	4.5	6.1	μA
I _{SOURCE,C}	Current when register ISOURCE=len10uA	6.7	8.9	11.3	μA
V _{DIFFHYST}	Optional hysteresis applied to differential input	40	60	80	mV
t _{INT_REF,START}	Startup time for the internal bandgap reference		1.5	2.2	μs
E _{INT_REF}	Internal bandgap reference error	-4.7	0	4.0	%
V _{INPUTOFFSET}	Input offset	-15	0	15	mV
t _{COMP,START}	Startup time for the comparator core		3		μs

11.4 CPU Electrical specification

11.4.1 CPU performance

² Propagation delay is with 10 mV overdrive.

The CPU performance metrics are derived from benchmarks executed on highly optimized firmware. The firmware is compiled using the specified compiler version and settings to ensure accurate and reliable performance data.

Compiler: Arm version 6.16 (armclang)

Compiler flags:

```
--target=arm-arm-none-eabi -c -g -masm=auto -Wno-unused-value -mcpu=cortex-m33 -mfloat-abi=hard -mfpv5-sp-d16 -flto -Omax
```

Linker flags:

```
--lto --remove
```

Symbol	Description	Min.	Typ.	Max.	Units
CM _{RRAM} CACHE128	CPU running CoreMark at 128 MHz from RRAM, cache enabled		500		CoreMark
CM _{rram} 128/MHz	CoreMark per MHz, running from RRAM, cache enabled, HFX0128M		3.90		CoreMark/ MHz
CM _{RAM} 128	CPU running CoreMark at 128 MHz from RAM		464		CoreMark

11.4.2 CPU wakeup times

Symbol	Description	Min.	Typ.	Max.	Units
t _{R2ON}	Time from pin reset to CPU executes the first instruction		60		μs
t _{OFF2ON}	Time from wake-up from System OFF mode to CPU executes the first instruction		60		μs
t _{IDLE2CPU}	Wakeup time from CPU sleep (WFI,WFE) to CPU executes the next instruction		13		μs
t _{IDLE2CPU,CONSTLAT}	Wakeup time from CPU sleep (WFI,WFE) to CPU executes the next instruction in constant latency sub-mode		9		μs

11.5 GPIO Electrical specification

11.5.1 GPIO Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
V _{IH}	Input high voltage	0.7 x VDD		VDD	V
V _{IL}	Input low voltage	VSS		0.3 x VDD	V
V _{OH,SD}	Output high voltage, standard drive, 0.5 mA, VDD ≥ 1.7	VDD - 0.4		VDD	V
V _{OH,HDL}	Output high voltage, high drive, 5 mA, VDD ≥ 2.7 V	VDD - 0.4		VDD	V
V _{OH,HDL}	Output high voltage, high drive, 3 mA, VDD ≥ 1.7 V	VDD - 0.4		VDD	V
V _{OL,SD}	Output low voltage, standard drive, 0.5 mA, VDD ≥ 1.7	VSS		VSS + 0.4	V
I _{OL,SD}	Current at VSS+0.4 V, output set low, standard drive, VDD ≥ 1.7	1	3	4	mA
I _{OL,HDL}	Current at VSS+0.4 V, output set low, high drive, VDD ≥ 1.7 V	3			mA
I _{OL,ED}	Current at VSS+0.4 V, output set low, extra drive, VDD ≥ 1.7 V	16			mA
I _{OH,SD}	Current at VDD-0.4 V, output set high, standard drive, VDD ≥ 1.7	1	3	4	mA
I _{OH,HDL}	Current at VDD-0.4 V, output set high, high drive, VDD ≥ 1.7 V	4			mA
I _{OH,ED}	Current at VDD-0.4 V, output set high, extra drive, VDD ≥ 1.7 V	14			mA
I _{GPIO,TOTAL}	Recommended maximum sustained current drawn by all GPIOs			15	mA
t _{HRF,12pF}	Rise/Fall time, high drive mode, 20-80%, 12 pF load ¹		4		ns
t _{ERF,12pF}	Rise/Fall time, extra drive mode, 20-80%, 12 pF load ¹		0.9		ns

Symbol	Description	Min.	Typ.	Max.	Units
R_{PU}	Pull-up resistance	12	14	16	k Ω
R_{PD}	Pull-down resistance	12	14	18	k Ω
C_{PAD}	Pad capacitance		1		pF
C_{PAD_NFC}	Pad capacitance on NFC pads		5		pF

11.6 I2S Electrical specification

11.6.1 I2S timing specification

Symbol	Description	Min.	Typ.	Max.	Units
t_{S_SDIN}	SDIN setup time before SCK rising	20			ns
t_{H_SDIN}	SDIN hold time after SCK rising	15			ns
t_{S_SDOUT}	SCK falling edge to SDOUT valid	10			ns
t_{H_SDOUT}	SDOUT hold time after SCK falling	10			ns
t_{SCK_LRCK}	SCLK falling to LRCK edge	-5	0	+5	ns
f_{MCK}	MCK frequency			8000	kHz
f_{LRCK}	LRCK frequency			100	kHz
f_{SCK}	SCK frequency			8000	kHz
DC_{CK}	Clock duty cycle (MCK, LRCK, SCK)	45		55	%

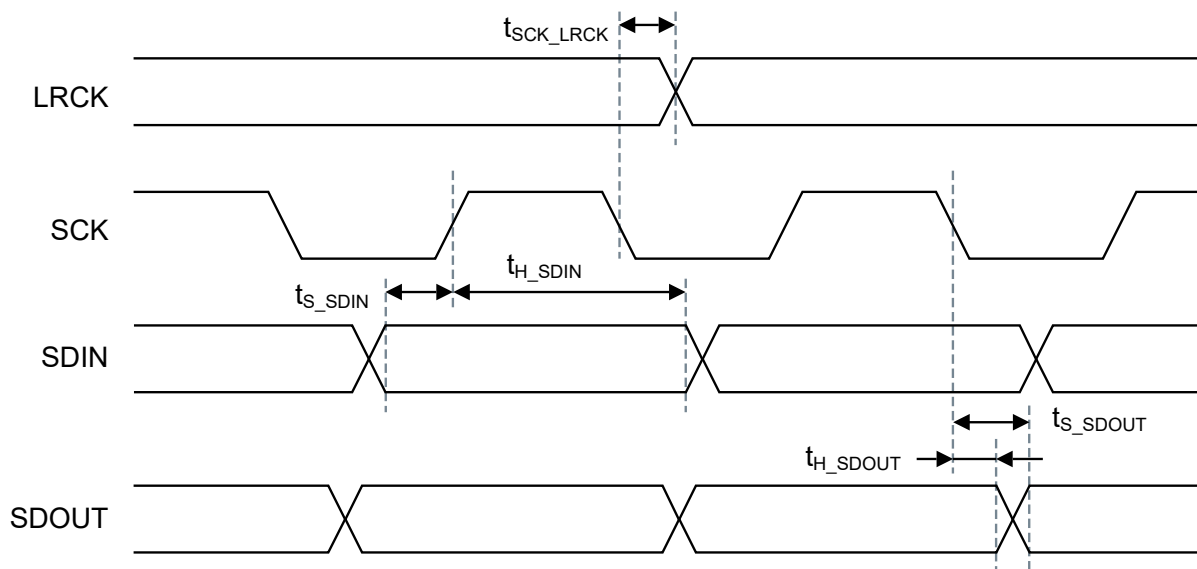


Figure 176: I2S timing diagram

11.7 LPCOMP Electrical specification

11.7.1 LPCOMP Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
$t_{LPCANADET}$	Time from VIN crossing (≥ 50 mV above threshold) to ANADETECT signal generated		1.5		μs
$V_{INPOFFSET}$	Input offset including reference ladder error	-18		18	mV
V_{HYST}	Optional hysteresis		43		mV
$t_{STARTUP}$	Startup time for LPCOMP	10	32	62	μs

11.8 NFCT Electrical specification

11.8.1 NFCT Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
f_c	Frequency of operation	12.55	13.56	13.57	MHz
C_{MI}	Carrier modulation index	95			%
DR	Data Rate		106		kbps
V_{sense}	Peak differential field detect threshold level on NFC1-NFC2, with input being high impedance in sense mode		1.w3		Vp
I_{max}	Maximum input current on NFCT pins		80	130	mA

11.8.2 NFCT Timing Parameters

Symbol	Description	Min.	Typ.	Max.	Units
$t_{activate}$	Time from task_ACTIVATE in SENSE or DISABLE state to ACTIVATE_A or IDLE state, excluding voltage supply and oscillator startup times			625	μs
t_{sense}	Time from remote field is present in SENSE mode to FIELDDETECTED event is asserted		7.2		μs

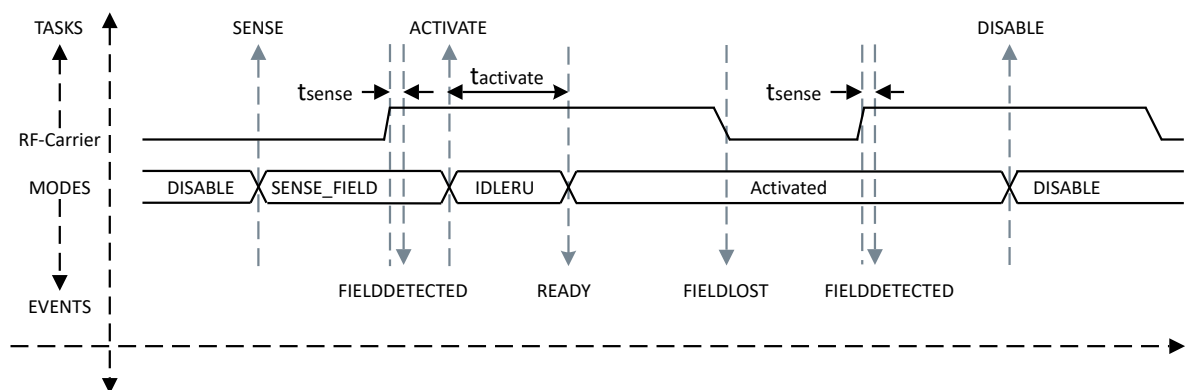


Figure 177: NFCT timing parameters (Shortcuts for FIELDDETECTED and FIELDLOST are disabled)

11.9 OSCILLATORS Electrical specification

11.9.1 32 MHz crystal oscillator (HFXO)

Symbol	Description	Min.	Typ.	Max.	Units
f_{HFXO}	External crystal frequency		32		MHz
$f_{\text{TOL_HFXO}}$	Frequency tolerance requirement for 2.4 GHz proprietary radio applications			±60	ppm
$f_{\text{TOL_HFXO_BLE}}$	Frequency tolerance requirement, Bluetooth Low Energy applications			±40	ppm
$C_{\text{L_HFXO}}$	Load capacitance	6		9	pF

11.9.2 32.768 kHz crystal oscillator (LFXO)

Symbol	Description	Min.	Typ.	Max.	Units
f_{LFXO}	External crystal frequency		32.768		kHz
$f_{\text{TOL_LFXO_BLE}}$	Frequency tolerance requirement, Bluetooth Low Energy applications			±500	ppm
$f_{\text{TOL_LFXO_ANT}}$	Frequency tolerance requirement for ANT applications			±50	ppm
$C_{\text{L_LFXO}}$	Load capacitance	6		9	pF

11.10 PPI Electrical specification

11.10.1 Typical PPI latencies

Symbol	Description	Min.	Typ.	Max.	Units
t_{PPI}	PPI latency between same power-domain peripherals in RUN state (i.e. PCLK's are running)		2		cycles

11.11 PDM Electrical specification

11.11.1 PDM Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
$f_{\text{PDM_CLK_64}}$	PDM clock speed. PDMCLKCTRL = Default (Setting needed for 16 kHz sample frequency @ RATIO = Ratio64)		1.032		MHz
$f_{\text{PDM_CLK_80}}$	PDM clock speed. PDMCLKCTRL = 1280K (Setting needed for 16 kHz sample frequency @ RATIO = Ratio80)		1.280		MHz
$t_{\text{PDM_JITTER}}$	Jitter in PDM clock output				ns
$T_{\text{dPDM_CLK}}$	PDM clock duty cycle	40	50	60	%
$t_{\text{PDM_DATA}}$	Decimation filter delay			5	ms
$t_{\text{PDM_cv}}$	Allowed clock edge to data valid			125	ns
$t_{\text{PDM_ci}}$	Allowed (other) clock edge to data invalid	0			ns
$t_{\text{PDM_s}}$	Data setup time at $f_{\text{PDM_CLK}} = 1.024 \text{ MHz}$ or 1.280 MHz	363			ns
$t_{\text{PDM_h}}$	Data hold time at $f_{\text{PDM_CLK}} = 1.024 \text{ MHz}$ or 1.280 MHz	0			ns
$G_{\text{PDM_default}}$	Default (reset) absolute gain of the PDM module		3.2		dB

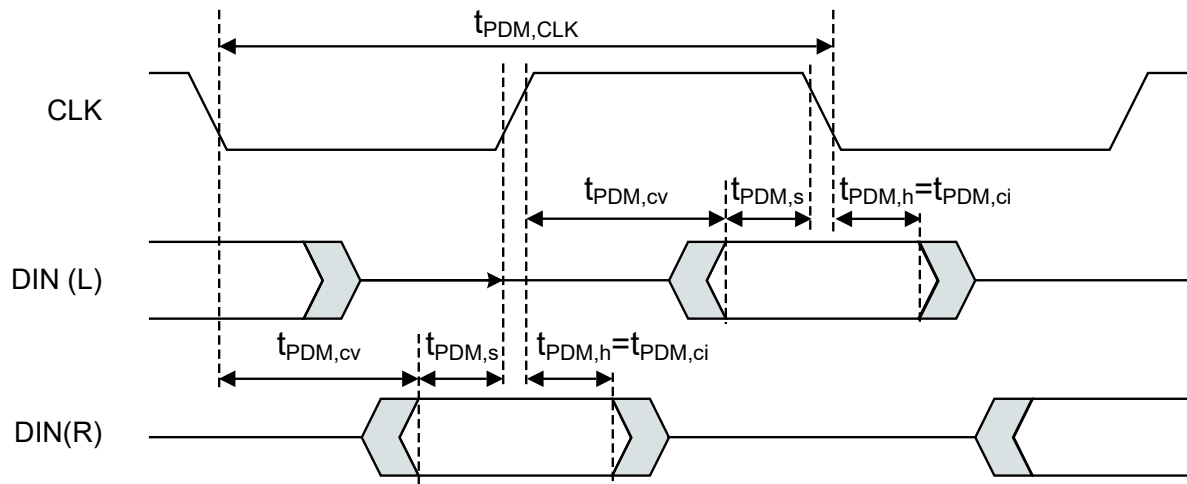


Figure 178: PDM timing diagram

11.12 QDEC Electrical specification

11.12.1 QDEC Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
t_{SAMPLE}	Time between sampling signals from quadrature decoder	128		131072	μs
t_{LED}	Time from LED is turned on to signals are sampled	0		511	μs

11.13 RADIO Electrical specification

11.13.1 General radio characteristics

Symbol	Description	Min.	Typ.	Max.	Units
f_{OP}	Operating frequencies	2360		2500	MHz
$f_{\text{PLL,CH,SP}}$	PLL channel spacing		1.0		MHz
$f_{\text{DELTA,1M}}$	Frequency deviation @ 1 Mbps		± 170		kHz
$f_{\text{DELTA,BLE,1M}}$	Frequency deviation @ Bluetooth LE 1 Mbps		± 250		kHz
$f_{\text{DELTA,2M}}$	Frequency deviation @ 2 Mbps		± 320		kHz
$f_{\text{DELTA,BLE,2M}}$	Frequency deviation @ Bluetooth LE 2 Mbps		± 500		kHz
f_{SKBPS}	On-the-air data rate	125		4000	kbps
$f_{\text{chip, IEEE 802.15.4}}$	Chip rate in IEEE 802.15.4 mode		2000		kchip/s

11.13.2 Radio current consumption (transmitter)

Symbol	Description	Min.	Typ.	Max.	Units
$I_{TX,MaxdBm,DCDC,QFN}$	TX only run current for QFN package, DC/DC, 3 V, P_{RF} at maximum power setting		9.1		mA
$I_{TX,MaxdBm,DCDC,CSP}$	TX only run current for CSP package, DC/DC, 3 V, P_{RF} at maximum power setting		9.7		mA
$I_{TX,0dBm,DCDC}$	TX only run current DC/DC, 3 V, $P_{RF} = 0$ dBm		3.7		mA
$I_{TX,MINUS4dBm,DCDC}$	TX only run current DC/DC, 3 V, $P_{RF} = -4$ dBm		2.8		mA
$I_{TX,MINUS8dBm,DCDC}$	TX only run current DC/DC, 3 V, $P_{RF} = -8$ dBm		2.2		mA
$I_{TX,MINUS12dBm,DCDC}$	TX only run current DC/DC, 3 V, $P_{RF} = -12$ dBm		1.9		mA
$I_{TX,MINUS16dBm,DCDC}$	TX only run current DC/DC, 3 V, $P_{RF} = -16$ dBm		1.7		mA
$I_{TX,MINUS40dBm,DCDC}$	TX only run current DC/DC, 3 V, $P_{RF} = -40$ dBm		1.2		mA
$I_{START,TX}$	TX start-up current, $P_{RF} = 4$ dBm		..		mA

11.13.3 Radio current consumption (Receiver)

Symbol	Description	Min.	Typ.	Max.	Units
$I_{RX,1M,DCDC}$	RX only run current DC/DC, 3 V, 1 Mbps/1 Mbps Bluetooth LE mode		2.1		mA
$I_{RX,2M,DCDC}$	RX only run current DC/DC, 3 V, 2 Mbps/2 Mbps Bluetooth LE mode		2.1		mA
$I_{START,RX,1M,DCDC}$	RX start-up current DC/DC, 3 V, 1 Mbps/1 Mbps Bluetooth LE mode		1.6		mA

11.13.4 Transmitter specification

Symbol	Description	Min.	Typ.	Max.	Units
$P_{RF,qfn}$	Maximum output power for QFN package		7		dBm
$P_{RF,CSP}$	Maximum output power for CSP package		8		dBm
P_{RFCR}	RF power accuracy	-2		2	dB
$P_{RF1,1}$	1st Adjacent Channel Transmit Power 1 MHz (1 Mbps)		-25 ³		dBc
$P_{RF2,1}$	2nd Adjacent Channel Transmit Power 2 MHz (1 Mbps)		-48 ³		dBc
$P_{RF1,2}$	1st Adjacent Channel Transmit Power 2 MHz (2 Mbps)		-25 ⁴		dBc
$P_{RF2,2}$	2nd Adjacent Channel Transmit Power 4 MHz (2 Mbps)		-52 ⁴		dBc
E_{vm}	Error vector magnitude in IEEE 802.15.4 mode		2		%rms
$P_{harm2nd, IEEE 802.15.4}$	2nd harmonics in IEEE 802.15.4 mode		-63		dBm
$P_{harm3rd, IEEE 802.15.4}$	3rd harmonics in IEEE 802.15.4 mode		-68		dBm

³ MODE = Nrf_1Mbit

⁴ MODE = Nrf_2Mbit

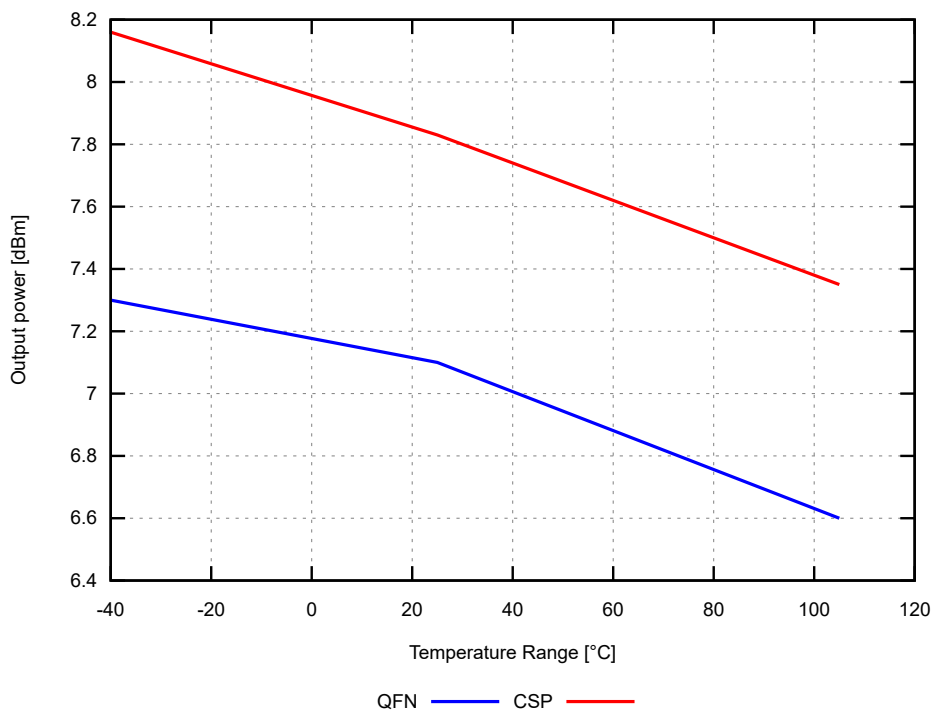


Figure 179: Output power, 1 Mbps Bluetooth low energy mode, at maximum TXPOWER setting (typical values)

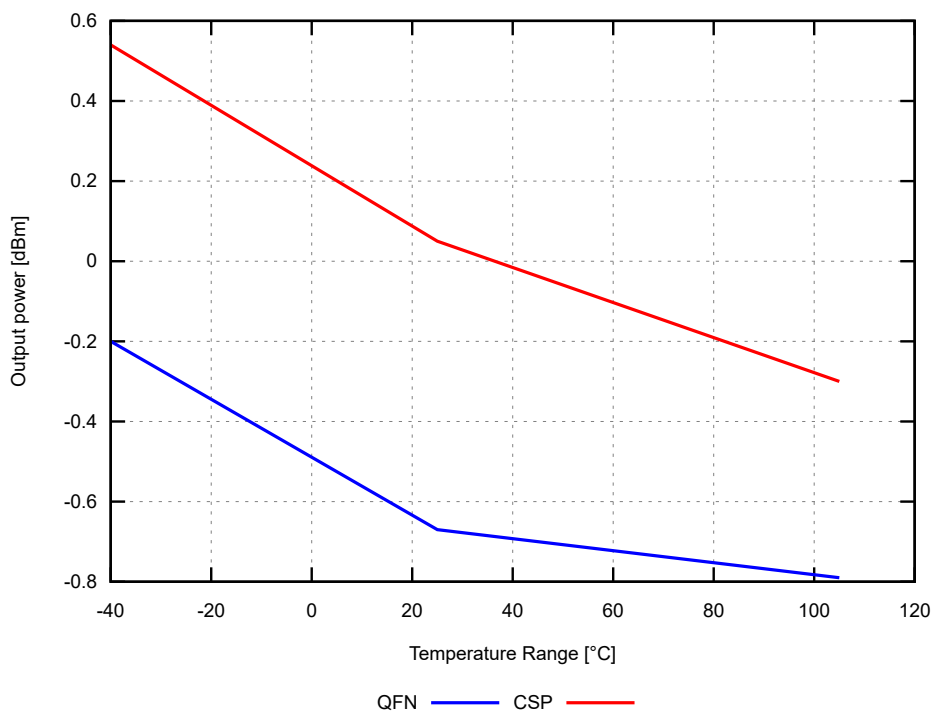


Figure 180: Output power, 1 Mbps Bluetooth low energy mode, at 0 dBm setting (typical values)

11.13.5 Receiver operation

Symbol	Description	Min.	Typ.	Max.	Units
P _{RX,MAX}	Maximum received signal strength at < 0.1% PER		0		dBm
P _{SENS,IT,1M}	Sensitivity, 1 Mbps nRF mode ideal transmitter ⁵		-93		dBm
P _{SENS,IT,2M}	Sensitivity, 2 Mbps nRF mode ideal transmitter ⁵		-90		dBm
P _{SENS,IT,4M}	Sensitivity, 4 Mbps nRF mode ideal transmitter ⁵		-90		dBm
P _{SENS,IT,SP,1M,BLE}	Sensitivity, 1 Mbps Bluetooth LE ideal transmitter, packet length ≤ 37 bytes BER = 1E-3 ⁶		-96 ⁷		dBm
P _{SENS,IT,LP,1M,BLE}	Sensitivity, 1 Mbps Bluetooth LE ideal transmitter, packet length ≥ 128 bytes BER = 1E-4		-95		dBm
P _{SENS,IT,SP,2M,BLE}	Sensitivity, 2 Mbps Bluetooth LE ideal transmitter, packet length ≤ 37 bytes		-94		dBm
P _{SENS,IT,BLE LE125k}	Sensitivity, 125 kbps Bluetooth LE mode		-104		dBm
P _{SENS,IT,BLE LE500k}	Sensitivity, 500 kbps Bluetooth LE mode		-99		dBm
P _{SENS,IEEE 802.15.4}	Sensitivity in IEEE 802.15.4 mode		-102		dBm

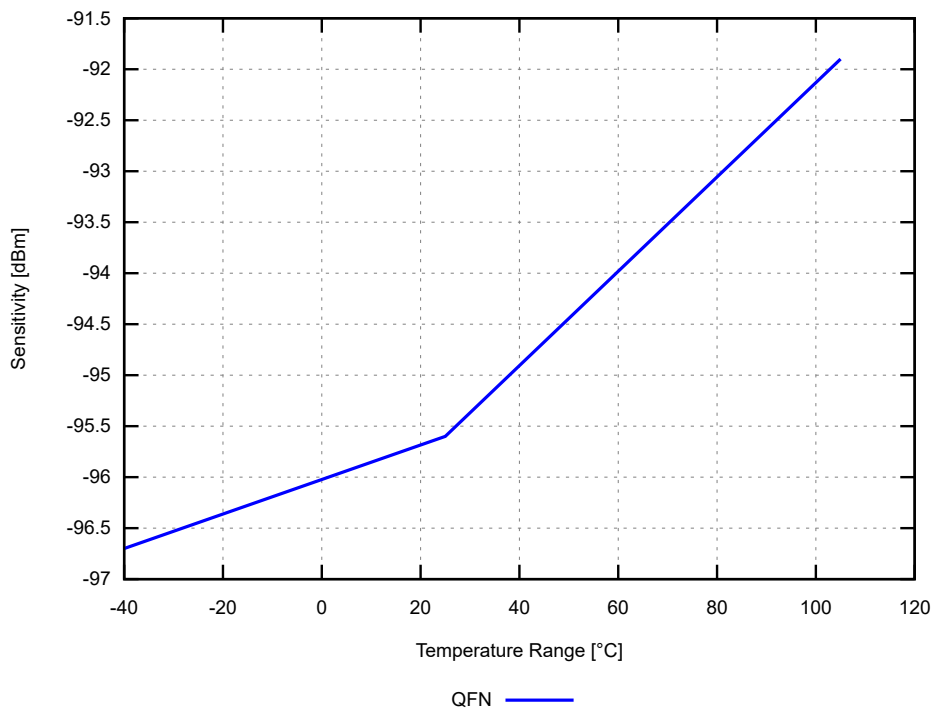


Figure 181: Sensitivity, 1 Mbps Bluetooth LE ideal transmitter, packet length ≤ 37 bytes BER = 1E-3 (typical values)

11.13.6 RX selectivity

RX selectivity with equal modulation on interfering signal⁸

⁵ Typical sensitivity applies when ADDR0 is used for receiver address correlation. When ADDR[1...7] are used for receiver address correlation, the typical sensitivity for this mode is degraded by 3 dB.

⁶ As defined in the *Bluetooth Core Specification v4.0 Volume 6: Core System Package (Low Energy Controller Volume)*.

⁷ QFN package sensitivity is degraded by 1dB

⁸ Desired signal level at P_{IN} = -67 dBm. One interferer is used, having equal modulation as the desired signal. The input power of the interferer where the sensitivity equals BER = 0.1% is presented.

Symbol	Description	Min.	Typ.	Max.	Units
C/I _{1M,co-channel}	1 Mbps mode, co-channel interference		9		dB
C/I _{1M,-1MHz}	1 Mbps mode, Adjacent (-1 MHz) interference		-4		dB
C/I _{1M,+1MHz}	1 Mbps mode, Adjacent (+1 MHz) interference		-9		dB
C/I _{1M,-2MHz}	1 Mbps mode, Adjacent (-2 MHz) interference		-28		dB
C/I _{1M,+2MHz}	1 Mbps mode, Adjacent (+2 MHz) interference		-40		dB
C/I _{1M,-3MHz}	1 Mbps mode, Adjacent (-3 MHz) interference		-39		dB
C/I _{1M,+3MHz}	1 Mbps mode, Adjacent (+3 MHz) interference		-43		dB
C/I _{1M,±6MHz}	1 Mbps mode, Adjacent (≥6 MHz) interference		-48		dB
C/I _{1MBLE,co-channel}	1 Mbps Bluetooth LE mode, co-channel interference		-6		dB
C/I _{1MBLE,-1MHz}	1 Mbps Bluetooth LE mode, Adjacent (-1 MHz) interference		-2		dB
C/I _{1MBLE,+1MHz}	1 Mbps Bluetooth LE mode, Adjacent (+1 MHz) interference		-6		dB
C/I _{1MBLE,-2MHz}	1 Mbps Bluetooth LE mode, Adjacent (-2 MHz) interference		-29		dB
C/I _{1MBLE,+2MHz}	1 Mbps Bluetooth LE mode, Adjacent (+2 MHz) interference		-43		dB
C/I _{1MBLE,≥3MHz}	1 Mbps Bluetooth LE mode, Adjacent (≥3 MHz) interference		-46		dB
C/I _{1MBLE,image}	Image frequency interference		-29		dB
C/I _{1MBLE,image,1MHz}	Adjacent (1 MHz) interference to in-band image frequency		-39		dB
C/I _{2M,co-channel}	2 Mbps mode, co-channel interference		.10		dB
C/I _{2M,-2MHz}	2 Mbps mode, Adjacent (-2 MHz) interference		-5		dB
C/I _{2M,+2MHz}	2 Mbps mode, Adjacent (+2 MHz) interference		-9		dB
C/I _{2M,-4MHz}	2 Mbps mode, Adjacent (-4 MHz) interference		-27		dB
C/I _{2M,+4MHz}	2 Mbps mode, Adjacent (+4 MHz) interference		-42		dB
C/I _{2M,-6MHz}	2 Mbps mode, Adjacent (-6 MHz) interference		-38		dB
C/I _{2M,+6MHz}	2 Mbps mode, Adjacent (+6 MHz) interference		-45		dB
C/I _{2M,≥12MHz}	2 Mbps mode, Adjacent (≥12 MHz) interference		-50		dB
C/I _{2MBLE,co-channel}	2 Mbps Bluetooth LE mode, co-channel interference		6		dB
C/I _{2MBLE,-2MHz}	2 Mbps Bluetooth LE mode, Adjacent (-2 MHz) interference		-2		dB
C/I _{2MBLE,+2MHz}	2 Mbps Bluetooth LE mode, Adjacent (+2 MHz) interference		-6		dB
C/I _{2MBLE,-4MHz}	2 Mbps Bluetooth LE mode, Adjacent (-4 MHz) interference		-29		dB
C/I _{2MBLE,+4MHz}	2 Mbps Bluetooth LE mode, Adjacent (+4 MHz) interference		-44		dB
C/I _{2MBLE,≥6MHz}	2 Mbps Bluetooth LE mode, Adjacent (≥6 MHz) interference		-46		dB
C/I _{4M,co-channel}	4 Mbps mode, co-channel interference		6		dB
C/I _{4M,-4MHz}	4 Mbps mode, Adjacent (-4 MHz) interference		-4		dB
C/I _{4M,+4MHz}	4 Mbps mode, Adjacent (+4 MHz) interference		-11		dB
C/I _{4M,-8MHz}	4 Mbps mode, Adjacent (-8 MHz) interference		-27		dB
C/I _{4M,+8MHz}	4 Mbps mode, Adjacent (+8 MHz) interference		-46		dB
C/I _{4M,-12MHz}	4 Mbps mode, Adjacent (-12 MHz) interference		-40		dB
C/I _{4M,+12MHz}	4 Mbps mode, Adjacent (+12 MHz) interference		-50		dB
C/I _{4M,≥24MHz}	4 Mbps mode, Adjacent (≥24 MHz) interference		-56		dB
C/I _{2MBLE,image}	Image frequency interference		-29		dB
C/I _{2MBLE,image,2MHz}	Adjacent (2 MHz) interference to in-band image frequency		-38		dB
C/I _{125k BLE LR,co-channel}	125 kbps Bluetooth LE LR mode, co-channel interference		1		dB
C/I _{125k BLE LR,-1MHz}	125 kbps Bluetooth LE LR mode, Adjacent (-1 MHz) interference		-13		dB
C/I _{125k BLE LR,+1MHz}	125 kbps Bluetooth LE LR mode, Adjacent (+1 MHz) interference		-16		dB
C/I _{125k BLE LR,-2MHz}	125 kbps Bluetooth LE LR mode, Adjacent (-2 MHz) interference		-36		dB
C/I _{125k BLE LR,+2MHz}	125 kbps Bluetooth LE LR mode, Adjacent (+2 MHz) interference		-52		dB
C/I _{125k BLE LR,≥3MHz}	125 kbps Bluetooth LE LR mode, Adjacent (≥3 MHz) interference		-55		dB
C/I _{125k BLE LR,image}	Image frequency interference		-36		dB
C/I _{IEEE 802.15.4,-5MHz}	IEEE 802.15.4 mode, Adjacent (-5 MHz) rejection		-35		dB
C/I _{IEEE 802.15.4,+5MHz}	IEEE 802.15.4 mode, Adjacent (+5 MHz) rejection		-38		dB
C/I _{IEEE 802.15.4,±10MHz}	IEEE 802.15.4 mode, Alternate (±10 MHz) rejection		-50		dB

11.13.7 RX intermodulation

RX intermodulation. Desired signal level at $P_{IN} = -64$ dBm. Two interferers with equal input power are used. The interferer closest in frequency is not modulated, the other interferer is modulated equal with the desired signal. The input power of the interferers where the sensitivity equals $BER = 1E-3$ is presented.

Symbol	Description	Min.	Typ.	Max.	Units
$P_{IMD,5TH,1M}$	IMD performance, 1 Mbps, 5th offset channel, packet length ≤ 37 bytes				dBm
$P_{IMD,5TH,1M,BLE}$	IMD performance, Bluetooth LE 1 Mbps, 5th offset channel, packet length ≤ 37 bytes		-19		dBm
$P_{IMD,5TH,2M}$	IMD performance, 2 Mbps, 5th offset channel, packet length ≤ 37 bytes				dBm
$P_{IMD,5TH,2M,BLE}$	IMD performance, Bluetooth LE 2 Mbps, 5th offset channel, packet length ≤ 37 bytes		-16		dBm

11.13.8 Radio timing

Symbol	Description	Min.	Typ.	Max.	Units
$t_{TXEN,BLE,1M}$	Time between TXEN task and READY event after channel FREQUENCY configured (1 Mbps Bluetooth LE and 150 μ s TIFS)		140		μ s
$t_{TXEN,FAST,BLE,1M}$	Time between TXEN task and READY event after channel FREQUENCY configured (1 Mbps Bluetooth LE with fast ramp-up and 150 μ s TIFS)		40		μ s
$t_{TXDIS,BLE,1M}$	When in TX, delay between DISABLE task and DISABLED event for MODE = Nrf_1Mbit and MODE = Ble_1Mbit		2		μ s
$t_{RXEN,BLE,1M}$	Time between the RXEN task and READY event after channel FREQUENCY configured (1 Mbps Bluetooth LE)		134		μ s
$t_{RXEN,FAST,BLE,1M}$	Time between the RXEN task and READY event after channel FREQUENCY configured (1 Mbps Bluetooth LE with fast ramp-up)		40		μ s
$t_{RXDIS,BLE,1M}$	When in RX, delay between DISABLE task and DISABLED event for MODE = Nrf_1Mbit and MODE = Ble_1Mbit		1		μ s
$t_{TXDIS,BLE,2M}$	When in TX, delay between DISABLE task and DISABLED event for MODE = Nrf_2Mbit and MODE = Ble_2Mbit		2		μ s
$t_{RXDIS,BLE,2M}$	When in RX, delay between DISABLE task and DISABLED event for MODE = Nrf_2Mbit and MODE = Ble_2Mbit		1		μ s
$t_{TXEN,IEEE\ 802.15.4}$	Time between TXEN task and READY event after channel FREQUENCY configured (IEEE 802.15.4 mode)		130		μ s
$t_{TXEN,FAST,IEEE\ 802.15.4}$	Time between TXEN task and READY event after channel FREQUENCY configured (IEEE 802.15.4 mode with fast ramp-up)		40		μ s
$t_{TXDIS,IEEE\ 802.15.4}$	When in TX, delay between DISABLE task and DISABLED event (IEEE 802.15.4 mode)		18		μ s
$t_{RXEN,IEEE\ 802.15.4}$	Time between the RXEN task and READY event after channel FREQUENCY configured (IEEE 802.15.4 mode)		130		μ s
$t_{RXEN,FAST,IEEE\ 802.15.4}$	Time between the RXEN task and READY event after channel FREQUENCY configured (IEEE 802.15.4 mode with fast ramp-up)		40		μ s
$t_{RXDIS,IEEE\ 802.15.4}$	When in RX, delay between DISABLE task and DISABLED event (IEEE 802.15.4 mode)		0.2		μ s
$t_{RX-to-TX\ turnaround}$	Maximum TX-to-RX or RX-to-TX turnaround time in IEEE 802.15.4 mode		17		μ s

11.13.9 Received signal strength indicator (RSSI) specifications

Symbol	Description	Min.	Typ.	Max.	Units
RSSI _{ACC}	RSSI accuracy in the range -90 to -30 dBm		±2		dB
RSSI _{RESOLUTION}	RSSI resolution		1		dB
RSSI _{PERIOD}	RSSI sampling time from RSSI _{START} task		0.25		µs
RSSI _{SETTLE}	RSSI settling time after signal level change		15	20	µs

11.13.10 Jitter

Symbol	Description	Min.	Typ.	Max.	Units
t _{DISABLED} JITTER	Jitter on DISABLED event relative to END event when shortcut between END and DISABLE is enabled		0.25		µs
t _{READY} JITTER	Jitter on READY event relative to TXEN and RXEN task		0.25		µs

11.13.11 IEEE 802.15.4 mode energy detection constants

Symbol	Description	Min.	Typ.	Max.	Units
ED_RSSISCALE	Scaling value when converting between hardware-reported value and dBm	4	4	4	
ED_RSSIOFFS	Offset value when converting between hardware-reported value and dBm	-92	-92	-92	

11.14 REGULATORS Electrical specification

11.14.1 Recommended operating conditions

Symbol	Description	Min.	Typ.	Max.	Units
V _{DD,POR}	VDD supply voltage needed during power-on reset.	1.75			V
V _{DD}	VDD operating voltage.	1.7		3.6	V

11.14.2 Power-fail comparator

Symbol	Description	Min.	Typ.	Max.	Units
V _{POF}	Voltage level warning thresholds (falling supply voltage). Levels are configurable between min. and max. in increments of 100 mV.	1.7		3.2	V
V _{POFTOL}	Threshold voltage tolerance.	-2		2	%
V _{POFHYST}	Threshold voltage hysteresis.	40	50	55	mV
V _{BOR,OFF}	Brownout reset voltage range System OFF mode. Brownout only applies to the voltage on VDD.	1.56		1.64	V
V _{BOR,ON}	Brownout reset voltage range System ON mode. Brownout only applies to the voltage on VDD.	1.57		1.64	V

11.15 RESET Electrical specification

11.15.1 Startup times

Symbol	Description	Min.	Typ.	Max.	Units
$t_{POR,10\mu s}$	Time measured as time in power-on reset after supply reaches minimum operating voltage, with VDD rise time from 1 μs to 100ms.		0.2	2	ms
t_{PINR}	Reset time when using pin reset, depending on pin capacitance		..		
$t_{PINR,500nF}$	500 nF capacitance at reset pin		..		ms
$t_{PINR,10\mu F}$	10 μF capacitance at reset pin		..		ms

11.16 RRAMC Electrical specification

11.16.1 RRAM programming

Symbol	Description	Min.	Typ.	Max.	Units
$n_{ENDURANCE}$	Number of times a 128-bit word line can be written		10000		
$t_{WRITE,UNBUFFERED}$	Time to write a 32-bit word using unbuffered write		65		μs
$t_{WRITE,WRITEBUFSIZE=1}$	Average time to write a 32-bit word in a stream of sequentially address ordered writes, using WRITEBUFSIZE=1		22		μs

11.17 SAADC Electrical specification

11.17.1 SAADC Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
f_{SAMPLE}	Maximum sampling rate			2000	kHz

11.18 SPIM Electrical specification

11.18.1 Timing specifications

Symbol	Description	Min.	Typ.	Max.	Units
f_{SPIM}	Bit rates for SPIM ⁹			8	Mbps
$f_{SPIM,HS}$	Bit rates for high-speed SPIM instances ¹⁰			32 ¹¹	Mbps
$t_{SPIM,START}$	Time from START task to transmission started		1		μs
$t_{SPIM,CCLK}$	SCK period	31.25			ns
$t_{SPIM,RSCK,LD}$	SCK rise time, standard drive ¹²			$t_{RF,25pF}$	

⁹ High bit rates may require GPIOs to be set as High Drive, see GPIO chapter for more details.

¹⁰ High bit rates may require GPIOs to be set as High Drive, see GPIO chapter for more details.

¹¹ For SPIM00 when running at 128 MHz and VDD=1.8V.

¹² At 25pF load, including GPIO pin capacitance, see GPIO spec.

Symbol	Description	Min.	Typ.	Max.	Units
$t_{\text{SPIM,RSCK,HD}}$	SCK rise time, high drive ¹²			$t_{\text{HRF},25\text{pF}}$	
$t_{\text{SPIM,FSCK,LD}}$	SCK fall time, standard drive ¹²			$t_{\text{RF},25\text{pF}}$	
$t_{\text{SPIM,FSCK,HD}}$	SCK fall time, high drive ¹²			$t_{\text{HRF},25\text{pF}}$	
$t_{\text{SPIM,WHSCK}}$	SCK high time ¹²	$(t_{\text{CSCK}}/2) - t_{\text{RSCK}} - 1.5 \text{ ns}$			
$t_{\text{SPIM,WLSCK}}$	SCK low time ¹²	$(t_{\text{CSCK}}/2) - t_{\text{FSCK}} - 1.5 \text{ ns}$			
$t_{\text{SPIM,SUMI}}$	SDI to CLK edge setup time	19			ns
$t_{\text{SPIM,HMI}}$	CLK edge to SDI hold time	18			ns
$t_{\text{SPIM,VMO}}$	CLK edge to SDO valid			59	ns
$t_{\text{SPIM,VMO,HS}}$	CLK edge to SDO valid, for high-speed SPIM instances			g^{13}	ns
$t_{\text{SPIM,HMO}}$	SDO hold time after CLK edge	20			ns

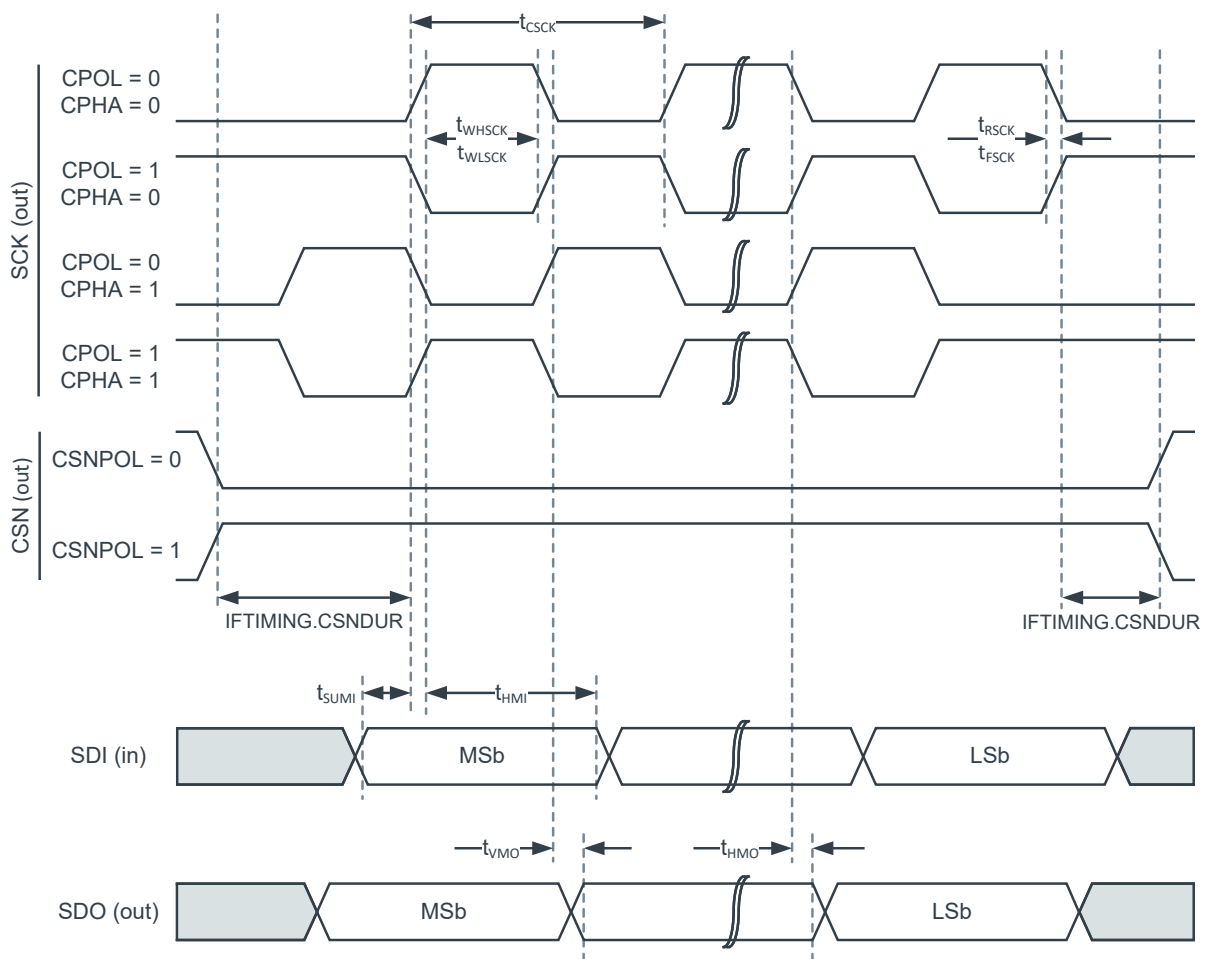


Figure 182: SPIM timing diagram

¹³ For SPIM00

11.19 SPIS Electrical specification

11.19.1 SPIS slave interface electrical specifications

Symbol	Description	Min.	Typ.	Max.	Units
f_{SPIS}	Bit rates for SPIS ¹⁴			8 ¹⁵	Mbps
$t_{\text{SPIS,START}}$	Time from RELEASE task to receive/transmit (CSN active)		1		μs

11.19.2 Serial Peripheral Interface Slave (SPIS) timing specifications

Symbol	Description	Min.	Typ.	Max.	Units
$t_{\text{SPIS,CCKIN}}$	SCK input period	125			ns
$t_{\text{SPIS,RFCKIN}}$	SCK input rise/fall time			30	ns
$t_{\text{SPIS,WHCKIN}}$	SCK input high time	30			ns
$t_{\text{SPIS,WLCKIN}}$	SCK input low time	30			ns
$t_{\text{SPIS,SUCSN}}$	CSN to CLK setup time	1000 ¹⁶			ns
$t_{\text{SPIS,HCSN}}$	CLK to CSN hold time	1000			ns
$t_{\text{SPIS,ASA}}$	CSN to SDO driven			70	ns
$t_{\text{SPIS,ASO}}$	CSN to SDO valid ¹⁷			1000	ns
$t_{\text{SPIS,DISSO}}$	CSN to SDO disabled ¹⁷			70	ns
$t_{\text{SPIS,CWH}}$	CSN inactive time	300			ns
$t_{\text{SPIS,VSO}}$	CLK edge to SDO valid			53	ns
$t_{\text{SPIS,HSO}}$	SDO hold time after CLK edge	13			ns
$t_{\text{SPIS,SUSI}}$	SDI to CLK edge setup time	19			ns
$t_{\text{SPIS,HSI}}$	CLK edge to SDI hold time	10			ns

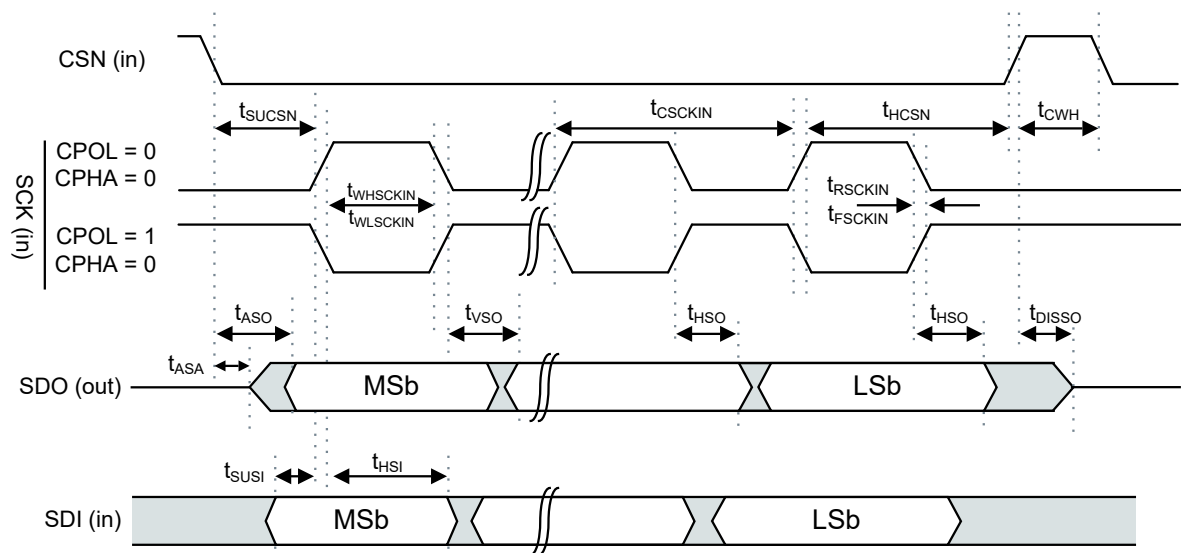


Figure 183: SPIS timing diagram, CPHA = 0

¹⁴ High bit rates may require GPIOs to be set as High Drive, see GPIO chapter for more details.

¹⁵ The actual maximum data rate depends on the master's CLK to SDO and SDI setup and hold timings.

¹⁶ Excluding any start-up delay for the high frequency clock in low power mode.

¹⁷ At 25pF load, including GPIO capacitance, see GPIO spec.

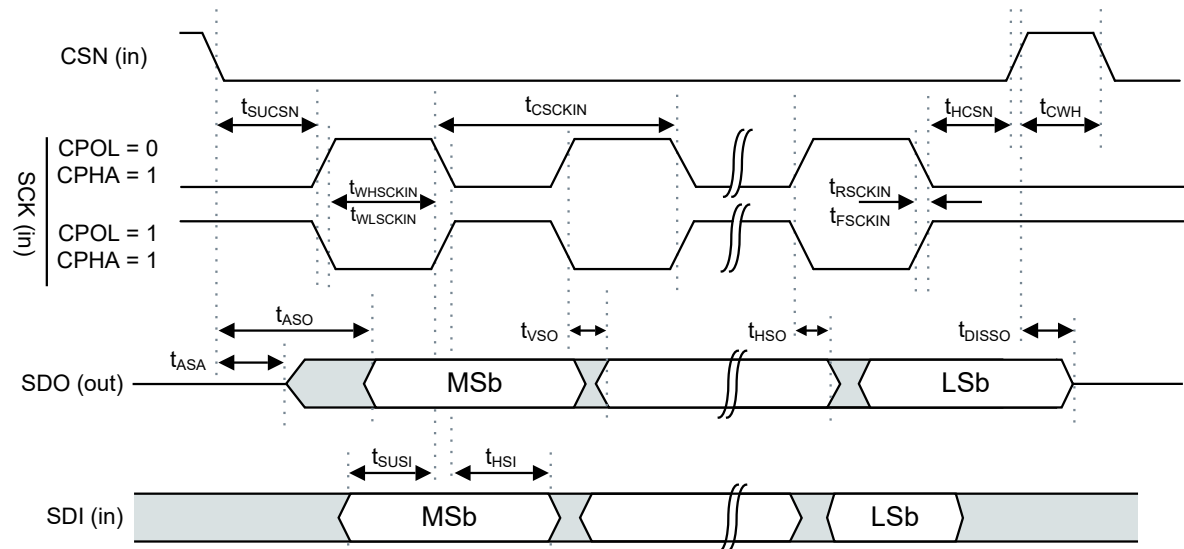


Figure 184: SPIS timing diagram, CPHA = 1

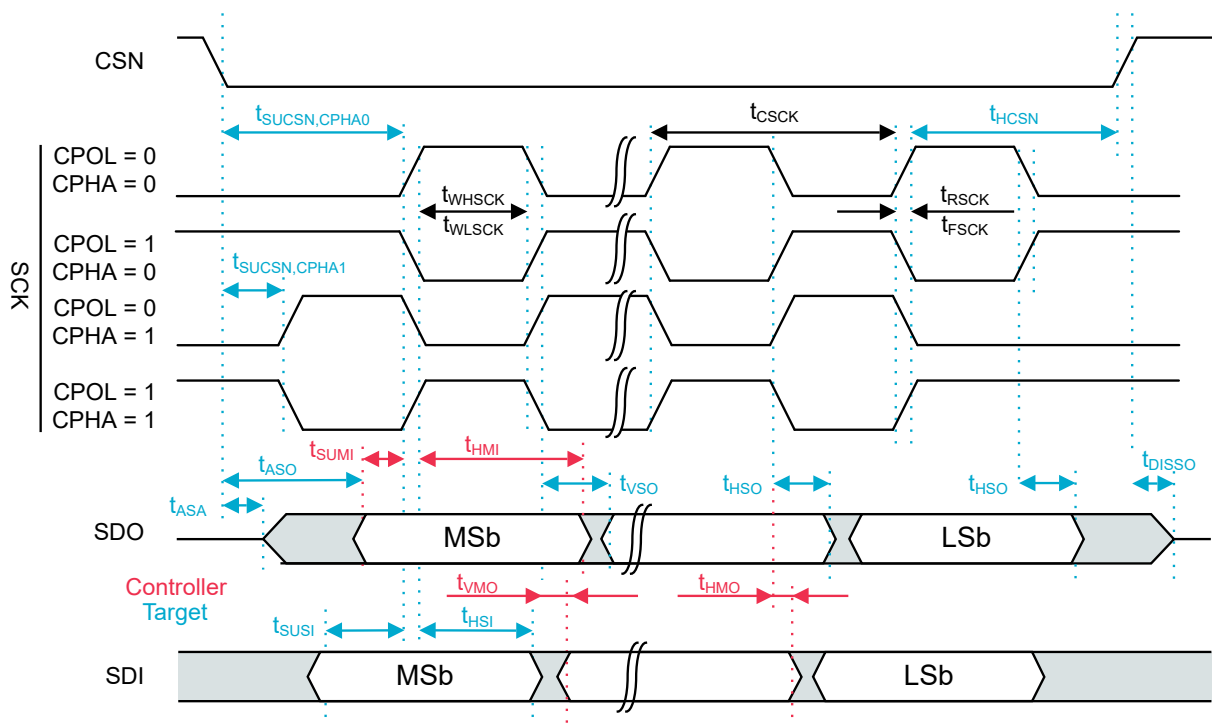


Figure 185: Common SPIM and SPIS timing diagram

11.20 SWDP Electrical specification

11.20.1 SW-DP

Symbol	Description	Min.	Typ.	Max.	Units
R_{pull}	Internal SWDIO and SWDCLK pull up/down resistance		14		kΩ
f_{SWDCLK}	SWDCLK frequency	0.125		8	MHz

11.20.2 Trace port

Symbol	Description	Min.	Typ.	Max.	Units
T _{cyc}	Clock period, as defined by Arm in Embedded Trace Macrocell Architecture Specification	15.625		250	ns

11.21 TEMP Electrical specification

11.21.1 Temperature Sensor Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
t _{TEMP}	Time required for temperature measurement		36		μs
T _{TEMP,RANGE}	Temperature sensor range	-20		70	°C
T _{TEMP,RANGE,EXT}	Temperature sensor extended temperature range	-40		105	°C
T _{TEMP,ACC}	Temperature sensor accuracy	-5		5	°C
T _{TEMP,ACC,EXT}	Temperature sensor accuracy, extended temperature range	-7		7	°C
T _{TEMP,RES}	Temperature sensor resolution		0.25		°C
T _{TEMP,STB}	Sample to sample stability at constant device temperature			±0.25	°C
T _{TEMP,OFFST}	Sample offset at 25°C	-3		3	°C

11.22 TWIM Electrical specification

11.22.1 TWIM interface electrical specifications

Symbol	Description	Min.	Typ.	Max.	Units
f _{TWIM,SCL}	Bit rates for TWIM ¹⁸	100		400	kbps
t _{TWIM,START}	Time from STARTRX/STARTTX task to transmission started		1.5		μs

11.22.2 Two Wire Interface Master (TWIM) timing specifications

Symbol	Description	Min.	Typ.	Max.	Units
t _{TWIM,SU_DATI}	Input data setup time before positive edge on SCL – all modes	20			ns
t _{TWIM,HD_DATO}	Output data hold time after negative edge on SCL – 100, 250 and 400 kbps	500		625	ns
t _{TWIM,HD_STA,100kbps}	TWIM master hold time for START and repeated START condition, 100 kbps	10000			ns
t _{TWIM,HD_STA,250kbps}	TWIM master hold time for START and repeated START condition, 250 kbps	4000			ns
t _{TWIM,HD_STA,400kbps}	TWIM master hold time for START and repeated START condition, 400 kbps	2400			ns
t _{TWIM,SU_STO,100kbps}	TWIM master setup time from SCL high to STOP condition, 100 kbps	5000			ns
t _{TWIM,SU_STO,250kbps}	TWIM master setup time from SCL high to STOP condition, 250 kbps	2000			ns
t _{TWIM,SU_STO,400kbps}	TWIM master setup time from SCL high to STOP condition, 400 kbps	1250			ns
t _{R,100kbps}	Rise time of both SDA and SCL signals, 100kbps			1000	ns
t _{F,100kbps}	Fall time of both SDA and SCL signals, 100kbps			300	ns
t _{R,400kbps}	Rise time of both SDA and SCL signals, 400kbps			300	ns
t _{F,400kbps}	Fall time of both SDA and SCL signals, 400kbps			300	ns
t _{TWIM,BUF,100kbps}	TWIM master bus free time between STOP and START conditions, 100 kbps	5200			ns

¹⁸ High bit rates or stronger pull-ups may require GPIOs to be set as High Drive, see [GPIO — General purpose input/output](#) on page 273 for more details.



11.23.1 TWIS slave timing specifications

Figure 187: TWIS timing diagram, 1 byte transaction



11.24 UARTE Electrical specification

11.24.1 UARTE electrical specification

Symbol	Description	Min.	Typ.	Max.	Units
f _{UARTE}	Baud rate for UARTE ²⁰ .			4000	kbps
t _{UARTE,CTSH}	CTS high time	0.5			μs
t _{UARTE,START}	Time from STARTRX/STARTTX task to transmission started		0.5		μs

11.25 WDT Electrical specification

11.25.1 Watchdog Timer Electrical Specification

Symbol	Description	Min.	Typ.	Max.	Units
t _{WDT}	Time out interval	458 μs		36 h	

²⁰ High baud rates may require GPIOs to be set as High Drive, see GPIO chapter for more details.

12 Recommended operating conditions

The operating conditions are the physical parameters that the device can operate within.

Symbol	Parameter	Min.	Nom.	Max.	Units
VDD	VDD supply voltage	1.7		3.5	V
VDD _{EXT}	VDD supply voltage under extended operating temperature	1.7		3.3	V
VDD _{POR}	VDD supply voltage needed during power-on reset	1.75			V
TA	Operating temperature	-40	25	85	°C
TA _{EXT}	Extended operating temperature	85		105	°C

Table 82: Recommended operating conditions

12.1 Extended Operating Temperature

The operating temperature range for the device is defined in [Recommended operating conditions](#) on page 835. The range extends from TA minimum to TA_{EXT} maximum.

Some electrical parameters are valid only for the TA operating temperature conditions. When this is the case, an additional parameter for the TA_{EXT} extended operating temperature condition is provided.

12.2 WLCSP light sensitivity

All WLCSP package variants are sensitive to visible and close-range infrared light. This means that a final product design must shield the chip properly, either by final product encapsulation or by shielding/coating of the WLCSP device.

All WLCSP package variants have a backside coating, where the marking side of the device is covered with a light absorbing film, while the side edges and the ball side of the device are still exposed and need to be protected.

13 Absolute maximum ratings

Maximum ratings are the extreme limits to which the chip can be exposed for a limited amount of time without permanently damaging it. Exposure to absolute maximum ratings for prolonged periods of time may affect the reliability of the device.

For accelerated lifetime testing (HTOL, etc.), supply voltage should not exceed the recommended operating conditions max value, see [Recommended operating conditions](#) on page 835.

	Min.	Max.	Unit
VDD	-0.3	3.6	V

Table 83: Supply voltage

	Min.	Max.	Unit
V _{I/O} , VDD ≤ 3.5 V	-0.3	VDD + 0.3	V
V _{I/O} , VDD > 3.5 V		3.6	V

Table 84: I/O pin voltage

	Min.	Max.	Unit
Storage temperature	-40	+105	°C
Moisture Sensitivity Level (MSL)		2	
ESD Human Body Model (HBM)		1	kV
ESD Charged Device Model (CDM)		500	V

Table 85: Environmental QFN package types

	Min.	Max.	Unit
Endurance	10,000		write/ rewrite cycles
Retention at 85°C	10		years
Retention at 105°C	2		years

Table 86: RRAM memory



14 Ordering information

This chapter contains information on device marking, ordering codes, and container sizes.

14.1 Device marking

The nRF54L15/10/05 package is marked as shown in the following figure.

N	5	4	L	<D	D>
<P	P>	<V	V>	<H>	<P>
<Y	Y>	<W	W>	<L	L>

Figure 188: Device marking

14.2 Box labels

The following figures show the box labels.

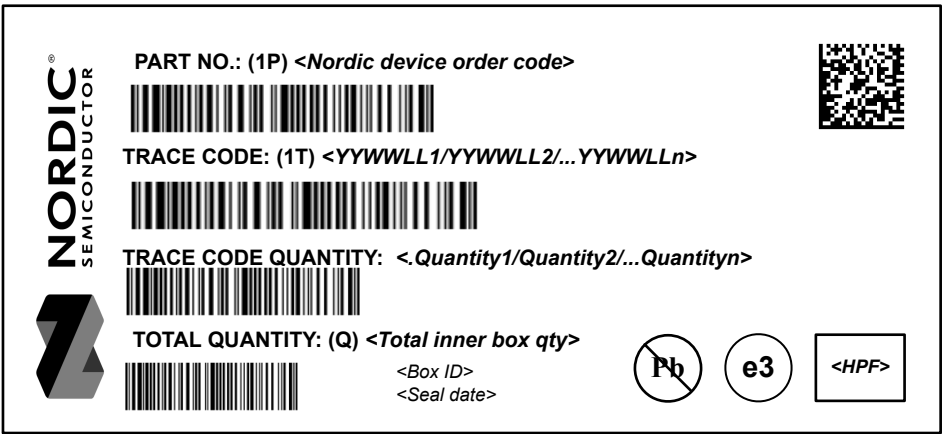


Figure 189: Inner box label












	
FROM: 	TO: 
PART NO: (1P) <Nordic device order code>  <div data-bbox="1019 600 1163 656" style="border: 1px solid black; padding: 2px; float: right;"><H><P><F></div>	
CUSTOMER PO NO: (K) <Customer Purchase Order No.>  <div data-bbox="1090 701 1160 770" style="border: 1px solid black; border-radius: 50%; padding: 2px; float: right; text-align: center;">No</div>	
SALES ORDER NO: (14K) <Nordic Sales Order+Sales order line no.+ Delivery line no.> 	
SHIPMENT ID.: 2K <Nordic's shipment ID.> 	
QUANTITY: (Q) <Total quantity> 	
COUNTRY OF ORIGIN.: 4L <2- character code of COO> 	CARTON NO: x/n
DELIVERY NO.: (9K) <Shipper's shipment no.> 	GROSS WEIGHT: <div data-bbox="837 1272 935 1339" style="border: 1px solid black; width: 60px; height: 30px; display: inline-block;"></div> KGS 

Figure 190: Outer box label

14.3 Order code

The following are the order codes and definitions for the device.

n	R	F	5	4	L	<D	D>	-	<P	P>	<V	V>	-	<C	C>
---	---	---	---	---	---	----	----	---	----	----	----	----	---	----	----

Figure 191: Order code

Abbreviation	Definition and implemented codes
N54L/nRF54L	nRF54L series product
<DD>	Device code
<PP>	Package variant code
<VV>	Function variant code
<H><P><F>	Build code H - Hardware version code P - Production configuration code (production site, etc.) F - Firmware version code (only visible on shipping container label)
<YY><WW><LL>	Tracking code YY - Year code WW - Assembly week number LL - Wafer lot code
<CC>	Container code

Table 87: Abbreviations

14.4 Code ranges and values

Defined here are the code ranges and values.

<DD>	Device
15	nRF54L15
10	nRF54L10
05	nRF54L05

Table 88: Device codes

<PP>	Package	Size (mm)	Pin/Ball count	Pitch (mm)
QF	QFN48	6x6	48	0.4
CA	CSP300	2.4x2.2	46	0.3

Table 89: Package variant codes

Device	<VV>	Flash (kB)	RAM (kB)
nRF54L15	AA	1524	256
nRF54L10	AA	1022	192
nRF54L05	AA	500	96

Table 90: Function variant codes

<H>	Description
[A . . Z]	Hardware version/revision identifier (incremental)

Table 91: Hardware version codes

<P>	Description
[0 . . 9]	Production device identifier (incremental)
[A . . Z]	Engineering device identifier (incremental)

Table 92: Production configuration codes

<F>	Description
[A . . N, P . . Z]	Version of preprogrammed firmware
[0]	Delivered with preprogrammed firmware to enable debug access

Table 93: Production version codes

<YY>	Description
[16 . . 99]	Production year: 2016 to 2099

Table 94: Year codes

<WW>	Description
[1 . . 52]	Week of production

Table 95: Week codes

<LL>	Description
[AA . . ZZ]	Wafer production lot identifier

Table 96: Lot codes

<CC>	Description
R7	7" Reel
R	13" Reel

Table 97: Container codes

14.5 Product options

Defined here are the product options for the device.

The following table lists the ordering code, as well as the minimum ordering quantity (MOQ).

Order code	MOQ
nRF54L15-QFAA-R7	1000
nRF54L15-QFAA-R	3000
nRF54L15-CAAA-R7	1500
nRF54L15-CAAA-R	7000

Table 98: nRF54L15 order codes

Order code	MOQ
nRF54L10-QFAA-R7	1000
nRF54L10-QFAA-R	3000

Table 99: nRF54L10 order codes

Order code	MOQ
nRF54L05-QFAA-R7	1000
nRF54L05-QFAA-R	3000

Table 100: nRF54L05 order codes

Order code	Description
nRF54L15-DK	nRF54L15 Development Kit

Table 101: Development tools order code

14.6 Preprogrammed firmware

The device is preprogrammed with firmware to allow debug access over the SWD interface.

The preprogrammed firmware disables access port protection by writing the TAMPC.PROTECT registers, and thus ensures the device can be programmed using the SWD interface without an Erase all function.

For more information about debug access protection, see [Access port protection](#) on page 751.

15 Legal notices

By using this documentation you agree to our terms and conditions of use. Nordic Semiconductor may change these terms and conditions at any time without notice.

Liability disclaimer

Nordic Semiconductor ASA reserves the right to make changes without further notice to the product to improve reliability, function, or design. Nordic Semiconductor ASA does not assume any liability arising out of the application or use of any product or circuits described herein.

Nordic Semiconductor ASA does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. If there are any discrepancies, ambiguities or conflicts in Nordic Semiconductor's documentation, the Datasheet prevails.

Nordic Semiconductor ASA reserves the right to make corrections, enhancements, and other changes to this document without notice.

Customer represents that, with respect to its applications, it has all the necessary expertise to create and implement safeguards that anticipate dangerous consequences of failures, monitor failures and their consequences, and lessen the likelihood of failures that might cause harm, and to take appropriate remedial actions.

Nordic Semiconductor ASA assumes no liability for applications assistance or the design of customers' products. Customers are solely responsible for the design, validation, and testing of its applications as well as for compliance with all legal, regulatory, and safety-related requirements concerning its applications.

Nordic Semiconductor ASA's products are not designed for use in life-critical medical equipment, support appliances, devices, or systems where malfunction of Nordic Semiconductor ASA's products can reasonably be expected to result in personal injury. Customer may not use any Nordic Semiconductor ASA's products in life-critical medical equipment unless adequate design and operating safeguards by customer's authorized officers have been made. Customer agrees that prior to using or distributing any life-critical medical equipment that include Nordic Semiconductor ASA's products, customer will thoroughly test such systems and the functionality of such products as used in such systems.

Customer will fully indemnify Nordic Semiconductor ASA and its representatives against any damages, costs, losses, and/or liabilities arising out of customer's non-compliance with this section.

RoHS and REACH statement

Refer to for complete hazardous substance reports, material composition reports, and latest version of Nordic's RoHS and REACH statements.

Trademarks

All trademarks, service marks, trade names, product names, and logos appearing in this documentation are the property of their respective owners.

Copyright notice

© 2024 Nordic Semiconductor ASA. All rights are reserved. Reproduction in whole or in part is prohibited without the prior written permission of the copyright holder.

