Product Change Notification: SYST-08PBUK426

Date:

09-Oct-2026

Product Category:

Digital Signal Controllers

Notification Subject:

dsPIC33AK128MC106 Family Silicon Errata and Data Sheet Clarification

Affected CPNs:

SYST-08PBUK426_Affected_CPN_10092026.pdf
SYST-08PBUK426_Affected_CPN_10092026.csv

Notification Text:

SYST-08PBUK426

Microchip has released a new Document for the dsPIC33AK128MC106 Family Silicon Errata and Data Sheet Clarification of devices. If you are using one of these devices please read the document located at **dsPIC33AK128MC106 Family Silicon Errata and Data Sheet Clarification**.

Notification Status: Final

Description of Change:

Added silicon issues 29 (CPU), 30 (SPI) and 31 (Debug).

Impacts to Data Sheet: Refer to **DS70005539C**

Reason for Change: To Improve Productivity

Change Implementation Status: Complete

Date Document Changes Effective: 09 Oct 2026

NOTE: Please be advised that this is a change to the document only the product has not been changed.

Markings to Distinguish Revised from Unrevised Devices: N/A

Attachments:

dsPIC33AK128MC106 Family Silicon Errata and Data Sheet Clarification

Please contact your local **Microchip sales office** with questions or concerns regarding this notification.

Terms and Conditions:

If you wish to receive Microchip PCNs via email please register for our PCN email service at our **PCN home page** select register then fill in the required fields. You will find instructions about registering for Microchips PCN email service in the **PCN FAQ** section.

If you wish to change your PCN profile, including opt out, please go to the **PCN home page** select login and sign into your myMicrochip account. Select a profile option from

Affected Catalog Part Numbers (CPN)

DSPIC33AK128MC105T-H/M7

DSPIC33AK128MC105T-H/PT

DSPIC33AK128MC105T-H/PTVAO

DSPIC33AK128MC105T-I/M7

DSPIC33AK128MC105T-I/PT

DSPIC33AK128MC106-E/M7

DSPIC33AK128MC106-E/PT

DSPIC33AK128MC106-H/M7

DSPIC33AK128MC106-H/PT

DSPIC33AK128MC106-I/M7

DSPIC33AK128MC106-I/PT

DSPIC33AK128MC106T-E/M7

DSPIC33AK128MC106T-E/PT

DSPIC33AK128MC106T-H/M7

DSPIC33AK128MC106T-H/PT

DSPIC33AK128MC106T-I/M7

DSPIC33AK128MC106T-I/PT

DSPIC33AK32MC102-E/M7

DSPIC33AK32MC102-E/SS

DSPIC33AK32MC102-H/M7

DSPIC33AK32MC102-H/SS

DSPIC33AK32MC102-I/M7

DSPIC33AK32MC102-I/SS

DSPIC33AK32MC102T-E/M7

DSPIC33AK128MC102T-H/M7

DSPIC33AK128MC102T-H/SS

DSPIC33AK128MC102T-I/M7

DSPIC33AK128MC102T-I/SS

DSPIC33AK128MC103-E/M7

DSPIC33AK128MC103-E/M7VAO

DSPIC33AK128MC103-H/M7

DSPIC33AK128MC103-I/M7

DSPIC33AK128MC103T-E/M7

DSPIC33AK128MC103T-H/M7

DSPIC33AK128MC103T-I/M7

DSPIC33AK128MC105-E/M7

DSPIC33AK128MC105-E/PT

DSPIC33AK128MC105-E/PTVAO

DSPIC33AK128MC105-H/M7

DSPIC33AK128MC105-H/PT

DSPIC33AK128MC105-H/PTVAO

DSPIC33AK128MC105-I/M7

DSPIC33AK128MC105-I/PT

DSPIC33AK128MC105T-E/M7

DSPIC33AK128MC105T-E/PT

DSPIC33AK64MC105-H/PT

DSPIC33AK64MC105-H/PTVAO

DSPIC33AK64MC105-I/M7

DSPIC33AK64MC105-I/PT

DSPIC33AK64MC105T-E/M7

DSPIC33AK64MC105T-E/PT

DSPIC33AK64MC105T-H/M7

DSPIC33AK64MC105T-H/PT

DSPIC33AK64MC105T-H/PTVAO

DSPIC33AK64MC105T-I/M7

DSPIC33AK64MC105T-I/PT

DSPIC33AK64MC106-E/M7

DSPIC33AK64MC106-E/PT

DSPIC33AK64MC106-H/M7

DSPIC33AK64MC106-H/PT

DSPIC33AK64MC106-I/M7

DSPIC33AK64MC106-I/PT

DSPIC33AK64MC106T-E/M7

DSPIC33AK64MC106T-E/PT

DSPIC33AK64MC106T-H/M7

DSPIC33AK64MC106T-H/PT

DSPIC33AK64MC106T-I/M7

DSPIC33AK64MC106T-I/PT

DSPIC33AK64MC102T-E/M7

DSPIC33AK64MC102T-E/SS

DSPIC33AK64MC102T-H/M7

DSPIC33AK64MC102T-H/M7VAO

DSPIC33AK64MC102T-H/SS

DSPIC33AK64MC102T-I/M7

DSPIC33AK64MC102T-I/SS

DSPIC33AK64MC103-E/M7

DSPIC33AK64MC103-H/M7

DSPIC33AK64MC103-I/M7

DSPIC33AK64MC103T-E/M7

DSPIC33AK64MC103T-H/M7

DSPIC33AK64MC103T-I/M7

DSPIC33AK64MC105-E/M7

DSPIC33AK64MC105-E/PT

DSPIC33AK64MC105-H/M7

DSPIC33AK128MC102-E/M7

DSPIC33AK128MC102-E/SS

DSPIC33AK128MC102-H/M7

DSPIC33AK128MC102-H/M7VAO

DSPIC33AK128MC102-H/SS

DSPIC33AK128MC102-I/M7

DSPIC33AK128MC102-I/SS

DSPIC33AK128MC102T-E/M7

DSPIC33AK128MC102T-E/SS

DSPIC33AK128MC105T-E/PTVAO

DSPIC33AK32MC102T-E/SS

DSPIC33AK32MC102T-H/M7

DSPIC33AK32MC102T-H/SS

DSPIC33AK32MC102T-I/M7

DSPIC33AK32MC102T-I/SS

DSPIC33AK32MC103-E/M7

DSPIC33AK32MC103-H/M7

DSPIC33AK32MC103-I/M7

DSPIC33AK32MC103T-E/M7

DSPIC33AK32MC103T-H/M7

DSPIC33AK32MC103T-I/M7

DSPIC33AK32MC105-E/M7

DSPIC33AK32MC105-E/PT

DSPIC33AK32MC105-H/M7

DSPIC33AK32MC105-H/PT

DSPIC33AK32MC105-I/M7

DSPIC33AK32MC105-I/PT

DSPIC33AK32MC105T-E/M7

DSPIC33AK32MC105T-E/PT

DSPIC33AK32MC105T-H/M7

DSPIC33AK32MC105T-H/PT

DSPIC33AK32MC105T-I/M7

DSPIC33AK32MC105T-I/PT

DSPIC33AK32MC106-E/M7

DSPIC33AK32MC106-E/PT

DSPIC33AK32MC106-H/M7

DSPIC33AK32MC106-H/PT

DSPIC33AK32MC106-I/M7

DSPIC33AK32MC106-I/PT

DSPIC33AK32MC106T-E/M7

DSPIC33AK32MC106T-E/PT

DSPIC33AK32MC106T-H/M7

DSPIC33AK32MC106T-H/PT

DSPIC33AK32MC106T-I/M7

DSPIC33AK32MC106T-I/PT

DSPIC33AK64MC102-E/M7

DSPIC33AK64MC102-E/SS

DSPIC33AK64MC102-H/M7

DSPIC33AK64MC102-H/M7VAO

DSPIC33AK64MC102-H/SS

DSPIC33AK64MC102-I/M7

DSPIC33AK64MC102-I/SS

dsPIC33AK128MC106 Family Silicon Errata and Data Sheet Clarification

The dsPIC33AK128MC106 family devices that you have received conform functionally to the current device data sheet (DS70005539**C**), except for the anomalies described in this document.

The silicon issues discussed in the following pages are for silicon revisions with the Device and Revision IDs listed in Table 1. The silicon issues are summarized in Table 2.

The errata described in this document will be addressed in future revisions of dsPIC33AK128MC106 family silicon.

Note:	This document summarizes all silicon errata issues from all revisions of silicon, previous as well as current. Only the issues indicated in the last column of Table 2 apply to the current silicon revision (**A2**).

Data sheet clarifications and corrections start on Page 12, following the discussion of silicon issues.

The silicon revision level can be identified using the current version of MPLAB® IDE and Microchip's programmers, debuggers and emulation tools, which are available at the Microchip corporate website (www.microchip.com).

For example, to identify the silicon revision level using MPLAB X IDE in conjunction with a hardware debugger:

1. Using the appropriate interface, connect the device to the hardware debugger.
2. Open an MPLAB X IDE project.
3. Configure the MPLAB X IDE project for the appropriate device and hardware debugger.
4. For MPLAB X IDE, select *Window > Dashboard* and click the **Refresh Debug Tool Status** icon ( ).
5. Depending on the development tool used, the part number *and* Device Revision ID value appear in the **Output** window.

Note:	If you are unable to extract the silicon revision level, please contact your local Microchip sales office for assistance.

The DEVREV values for the various silicon revisions of the dsPIC33AK128MC106 family are shown in Table 1.

TABLE 1: SILICON DEVREV VALUES

Part Number	Device ID(1)	Revision ID for Silicon Revision(1)
		A2
dsPIC33AK32MC102	0x9D00	02h
dsPIC33AK32MC103	0x9D01	
dsPIC33AK32MC105	0x9D02	
dsPIC33AK32MC106	0x9D03	
dsPIC33AK64MC102	0x9D10	
dsPIC33AK64MC103	0x9D11	
dsPIC33AK64MC105	0x9D12	
dsPIC33AK64MC106	0x9D13	
dsPIC33AK128MC102	0x9D20	
dsPIC33AK128MC103	0x9D21	
dsPIC33AK128MC105	0x9D22	
dsPIC33AK128MC106	0x9D23	

Note 1: Refer to the *"dsPIC33AK128MC106 Family Flash Programming Specification"* (DS70005470) for detailed information on Device and Revision IDs for your specific device.

© 2024-2026 Microchip Technology Inc. and its subsidiaries

TABLE 2: SILICON ISSUE SUMMARY

Module	Feature	Item Number	Issue Summary	Affected Revisions[1]
				A2
ADC	Sign Bit	1.	The ADC Result Sign bit is not extended in Differential-mode.	X
CCP	Output	2.	When ASDGM = 1, the CCP generates an output with the shut-down gate input at 0.	X
CCP	Capture	3.	When CCSEL= 1 and IGSM= 1, CCP will capture the timer value without an edge.	X
CCP	Overflow	4.	The overflow condition is not clear when ICBNE = 0.	X
CPU	PCTRAP	5.	PCTRAP captures the trap origination address of the math error trap even when disabled.	X
DMA	Channel Overrun	6.	The DMA channel overrun is not set when triggered by software or external interrupts.	X
DMA	Ping-Pong Operation	7.	The DMA channel ping-pong operation works only once for One-Shot and Continuous Transfer modes	X
DMA	Ping-Pong Channel	8.	The Ping-Pong Channel Enable bit is not reset by hardware in Repeated Transfer modes of DMA.	X
DMA	Bus Read Error	9.	The DMA Bus Read Error Fault bit is not set.	X
I/O	$\overline{\text{MCLR}}$	10.	When the$\overline{\text{MCLR}}$ pin is driven low, the I/O pins take five cycles to tristate.	X
I/O	Change Notification	11.	The change notification interrupt fails when in sleep.	X
I2C	Host: 1 MHz Baud	12.	The bus collision may get asserted when the STOP bit is sent from the HOST at 1 MHz baud rate.	X
INT	Math Error Trap	13.	The COVB Math error trap cannot be disabled.	X
INT	Edge Detect	14.	The INT4 External interrupts are raised only during a positive edge of trigger irrespective of the Edge Detect Polarity Select bit.	X
OSC/CLK	POSC	15.	POSCRDY sets the early on clock switch to POSC.	X
OSC/CLK	Clk Divider	16.	The CLKxDIV register does not get reset on $\overline{\text{MCLR}}$.	X
OSC/CLK	FRC	17.	FRC is always enabled.	X
PTG	Software Trigger	18.	The PTGSWT bit is not cleared in hardware.	X
PWM	PCI	19.	The PCI SR latch does not reset in RESET Dominant mode.	X
PWM	TRIGx	20.	When TRIGx = EOC and CAHALF is changed by software, the trigger event can occur in an opposite phase.	X
QEI	Index Event	21.	The index event is missed when the clock divider (INTDIV[2:0]) is configured for greater than 1.	X
QEI	Index Counter	22.	The position counter is loaded with 0x00000001 on an index event for PIMOD[2:0] = 0b001.	X
SPI	Ignore Overflow	23.	The SPI Receiver does not continue to run on overflow with an Ignore Overflow bit set.	X
SPI	Interrupt	24.	The CPU is not waking up from sleep with a SPI watermark interrupt.	X
UART	RX Interrupt	25.	The UART Transmit collision is not suppressing a RX interrupt.	X
UART	Buffer Overflow	26.	The address is not causing the buffer overflow in Address Detect mode.	X
WDT	Wake-up	27.	The RCON.WDTO bit is not set when WDT wakes up the device from PWRSAV modes.	X
CPU	Trap	28.	The address error trap generated on a legal indirect+offset mov.	X

Note 1: Only those issues indicated in the last column apply to the current silicon revision.

© 2024-2026 Microchip Technology Inc. and its subsidiaries

TABLE 2: SILICON ISSUE SUMMARY (CONTINUED)

Module	Feature	Item Number	Issue Summary	Affected Revisions(1)
				A2
CPU	Interrupt Vectoring	29.	Interrupt vectoring and simultaneous PBU/cache invalidation may stall the CPU indefinitely.	X
SPI	Host Mode	30.	SPI Host mode may exhibit an unreliable operation when using the peripheral bus clock.	X
Debug	Breakpoints	31.	When a hardware breakpoint is placed immediately after a conditional branch instruction, the breakpoint can be missed and will result in no debugger halt.	X

Note 1: Only those issues indicated in the last column apply to the current silicon revision.

Silicon Errata Issues

Note:	This document summarizes all silicon errata issues from all revisions of silicon, previous as well as current. Only the issues indicated by the shaded column in the following tables apply to the current silicon revision (**A2**).

1. Module: ADC

The ADC Result Sign bit is not extended in Differential-mode (DIFF = 1). This affects the operation of the ADC's comparator and accumulators.

Work around

Extend the Sign bit in software after reading the buffer. There is no workaround for ADC's internal comparator and accumulator.

Affected Silicon Revisions

A2							
X							

2. Module: CCP

When ASDGM = 1 and shutdown gate input is low (OCFx = 0), a CCP output is still generated.

Work around

Read the auto shutdown fault RP pin through LAT register bits, and if the LAT value reads as zero, use the CCP Software Shutdown/Gate Control bit to turn off the CCP.

Affected Silicon Revisions

A2							
X							

3. Module: CCP

When CCP is configured for input capture and One-Shot mode (IGSM = 1), the CCP will capture the timer value without an edge.

Work around

Once CCP is turned ON, and before enabling gate capture, re-arm the capture module using Input Capture Gate Arm bit.

Affected Silicon Revisions

A2							
X							

4. Module: CCP

When CCP is configured for Input Capture mode (CCM = 1) and an overflow condition occurs (ICOV = 1), the read capture buffer will not accept the next value until ICBNE is cleared. However, an interrupt is still generated.

Work around

Clear the ICOV bit in software; this will also cause the ICBNE bit to be cleared.

Affected Silicon Revisions

A2							
X							

5. Module: CPU

In any trap event, the trap origination address register (PCTRAP) is loaded with the value of the Program Counter (PC) associated with the instruction that caused the trap. However, it is not expected to capture the trap origination address of math error traps when they are disabled.

Work around

There is no workaround. Alternatively, enable math error traps, ignore the PCTRAP value and clear PCTRAP before exiting the math error trap handler. This re-enables the PCTRAP to capture the trap origination address of any other traps.

Affected Silicon Revisions

A2							
X							

© 2024-2026 Microchip Technology Inc. and its subsidiaries

6. Module: DMA

A DMA channel is configured to perform the data transfer using a software trigger or an external interrupt. When the transaction is still ongoing and repeated, a trigger is issued via software or external interrupt and the overrun flag is not set as expected. Consequently, this means that any data loss occurring due to repeated software or external triggers is not being recorded.

Work around

There is no workaround if the DMA channel is triggered by an external interrupt. In case the DMA channel is triggered by software, the application is expected to make sure that another software trigger is not issued when the DMA channel is busy.

Affected Silicon Revisions

A2							
X							

7. Module: DMA

A pair of DMA channels is configured in either One-Shot or Continuous Transfer modes and enabled to operate in Ping-Pong mode. The two available channels for ping-pong work only for the first pair of triggers. The count register of the DMA channels is not reloaded as expected, causing the transactions to halt.

Work around

Reload the count, source and destination address registers manually after transfer completion of the respective channel and before re-enabling the respective channel.

Affected Silicon Revisions

A2							
X							

8. Module: DMA

A pair of DMA channels is configured for either repeated one-shot or continuous transfers and is enabled to operate in Ping-Pong mode. For the configured ping-pong pair, after each channel completes its transaction, the PCHAEN bit is not reset by the hardware. Therefore, the ping-pong operation ceases.

Work around

The application software is expected to reset the Ping-Pong Channel Enable bit (PCHAEN) explicitly for each DMA channel once the channel has completed its ongoing transaction.

Affected Silicon Revisions

A2							
X							

9. Module: DMA

To initiate a read error trap, it is necessary to activate the Read Error Trap Enable (RETEN) bit. On setting the RETEN bit, a read error trap is triggered, which concurrently sets the DMA Bus Read Error Fault bit (DMASTATx.DMAFLT2) to high. Conversely, if the RETEN bit is not enabled, the DMA Bus Read Error Fault bit (DMASTATx.DMAFLT2) will not be set. Therefore, to guarantee the detection of bus read errors in the DMA through the Bus Read Fault bit, RETEN needs to be enabled, although this action may result in the generation of undesired traps.

Work around

Set the RETEN bit explicitly in the application software. When the trap is generated, ignore the trap and exit from the respective ISR.

Affected Silicon Revisions

A2							
X							

10. Module: I/O

When the $\overline{\text{MCLR}}$ pin is driven to a low state, all I/O pins should tristate asynchronously. However, it takes five CPU clock cycles.

Work around

None.

Affected Silicon Revisions

A2							
X							

11. Module: I/O

When change notification is configured to generate an interrupt on the positive edge and the device enters Sleep mode, if the first edge is high to low, the device fails to wake up on the subsequent low to high input edge.

Work around

None. However, in cases when only a single I/O is used for waking up the device from sleep using the I/O change notification feature in the application, use an external interrupt as the wake-up source.

Affected Silicon Revisions

A2							
X							

12. Module: I^2C

Depending on various factors (MIPS, pull-up resistor, pull-up voltage, SMBus/I^2C voltage levels), bus collision may get asserted when the STOP bit is sent from the host at 1 MHz baud rate.

Work around

None.

Affected Silicon Revisions

A2							
X							

13. Module: INT

The COVTE bit of the INTCON4 register is used to enable/disable math error traps caused by a catastrophic overflow of Accumulator A or B. However, a catastrophic overflow of Accumulator B raises a math error trap, irrespective of the status of the COVTE bit.

Work around

None.

Affected Silicon Revisions

A2							
X							

14. Module: INT

The INTxEP bit in the INTCON2 register can be used to select polarity of external interrupt triggers, allowing them to trigger on either the positive or negative edge. However, INT4 interrupt triggers only on the positive edge trigger, irrespective of the state of the INT4EP bit in INTCON2 register.

Work around

There is no workaround for the negative polarity trigger on external interrupt INT4. If it is required to set up external interrupts to trigger on the negative edge of signal, use INT0-INT3 external interrupts.

Affected Silicon Revisions

A2							
X							

15. Module: OSC/CLK

The Status bit POSCRDY, which indicates the POSC source is ready to use, can be set before the POSC circuit has reached a sufficient amplitude. A subsequent clock switch to POSC can result in a corrupt clock.

Work around

Software should configure the clock settings (OSCCTRL & OSCCFG) while POSCEN = `0` to ensure a stable clock switch. After POSCEN = `1`, software should wait until the POSCRDY Status bit is asserted.

EXAMPLE 1:

```
OSCCTRLbits.POSCEN = 1;   //Enable POSC

while(OSCCTRLbits.POSCRDY == 0);

CLK1CONbits.OSWEN = 1; //initiate clk switch

while(CLK1CONbits.OSWEN != 0);
//wait clk switch to complete
```

Affected Silicon Revisions

A2							
X							

© 2024-2026 Microchip Technology Inc. and its subsidiaries

16. Module: OSC/CLK

When the device is reset using the MCLR pin, the CLKxDIV registers are not reset to default values specified in the device data sheet.

Work around

Use software to reset the CLKxDIV register following an MCLR reset.

Affected Silicon Revisions

A2							
X							

17. Module: OSC/CLK

The FRCEN bit has no effect, and FRC is always enabled.

Work around

None.

Affected Silicon Revisions

A2							
X							

18. Module: PTG

When executing back-to-back wait for software trigger step commands, the PTGSWT bit is not cleared by hardware.

Work around

Manually clear the PTGSWT bit before setting it again, between the two PTGSW trigger commands.

Affected Silicon Revisions

A2							
X							

19. Module: PWM

When the PCI input is high, the PCI Active signal is terminated when there is no termination signal.

Work around

None. Only the PCI active signal is affected, and the PWM outputs remain halted.

Affected Silicon Revisions

A2							
X							

20. Module: PWM

A trigger event is generated at the wrong phase when the trigger value is programmed to be equal to EOC, and the trigger compare event time (CAHALF) is changed on the fly.

Work around

Minimum difference between EOC and trigger values should be at least 64.

Affected Silicon Revisions

A2							
X							

21. Module: QEI

When QEI is configured with Position Counter Initialization Mode bits (PIMOD[2:0]) equal to 0b001, 0b010, 0b011, 0b100 or 0b111, any index event initializes the position counter. However, if the clock divider value is configured by INTDIV[2:0] bits greater than 1, and if the index event occurs between two clock edges, then the index event will be missed.

Work around

Index pulse width should be sufficiently high enough so that the event is captured by the QEI clock.

Affected Silicon Revisions

A2							
X							

© 2024-2026 Microchip Technology Inc. and its subsidiaries

22. Module: QEI

When QEI is configured in Quadrature mode (CCM[1:0] = 0b00) with PIMOD[2:0] = 0b001, any index event is supposed to reset the position counter. However, an index event will not reset the position counter, but it will be loaded with 0x00000001.

Work around

User application can enable interrupt on index event and load the position counter with zero. The index event should be sufficiently large enough so that software can reset the position counter before the next count up or count down pulse.

Affected Silicon Revisions

A2							
X							

23. Module: SPI

In the event of an overflow, the SPI reception is suspended when the Ignore Overflow bit is activated, and the SPIROV bit is not cleared.

Work around

Clear the SPIROV bit during the reception of the next byte that triggered the overflow.
Additionally, SPIBUF must be read at least once.

Affected Silicon Revisions

A2							
X							

24. Module: SPI

In Sleep mode, SPI receive watermark interrupts are not asserted even though the configured number of elements has been received, resulting in the device not waking up from sleep.

Work around

Ensure that the device enters Sleep mode with SPI receive buffer full interrupts enabled for properly waking up from sleep.

Affected Silicon Revisions

A2							
X							

25. Module: UART

When the TXCIE bit is set and the TXCIF = 1, the RX watermark interrupt is not suppressed, causing an unintentional interrupt.

Work around

Ignore the RX watermark interrupt when the TXCIF flag is raised.

Affected Silicon Revisions

A2							
X							

26. Module: UART

In Address Detect mode, the address byte does not cause the buffer overflow if an address is received when the RX buffer is full.

Work around

Read the data from the RX buffer before it is full.

Affected Silicon Revisions

A2							
X							

27. Module: WDT

The WDTO bit (RCON[4]) fails to set when WDT is the wake-up source from Sleep and Idle modes.

Work around

None. Alternatively, enable WDT interrupts in order to determine whether the source of wake-up from Power-Save modes was the WDT.

Affected Silicon Revisions

A2							
X							

© 2024-2026 Microchip Technology Inc. and its subsidiaries

28. Module: CPU

An address error trap is generated in Indirect Register Offset Addressing mode (represented as [Ws + Wb] within instructions), when the value of Ws (Source) used is beyond 24-bits wide (size of address path). The issue may occur even when the effective address (Ws + Wb) is a valid address.

Work around

Either of the following workarounds can be used to overcome the issue:

1) Define the value of Ws within 24-bits of address range.

2) When using the XC-DSC V3.20 or newer C Compiler for dsPIC® DSCs, enable the compiler option: `-merrata=base_offset (Project Properties>XC_DSC>xc-dsc-gcc>Additonal options)`

Affected Silicon Revisions

A2							
X							

29. Module: CPU

Interrupt vectoring and simultaneous PBU/cache invalidation may stall the CPU indefinitely.

If a Prefetch Branch Unit (PBU) cache invalidation request is issued and followed immediately by the CPU reading an IVT entry from Flash for exception processing, the CPU may enter a perpetually stalled state. The CPU will only recover in response to a hardware reset (watchdog timeout, MCLR, POR, etc.). Cache invalidation requests capable of triggering this erratum have two possible sources:

1. Manual cache invalidation: Firmware or DMA writes a '1' to the CHECON[CHEINV] bit.
2. RTSP operations: Firmware or DMA writes a '1' to the NVMCON[WR] bit while CHECON[CHECOH] == 1 (all NVM addresses are below the threshold on single-partition only masksets).

Work around

Either of the following workarounds can be used to overcome the issue.

Work around 1

Place the IVT in RAM instead of Flash. This option is recommended for best performance as all interrupts will have two to five clocks less latency, and up to three clocks of latency jitter will be removed each time the CPU fetches an IVT entry.

Example 1:

```
// Copies IVT from Flash into RAM and begins using the RAM copy. Avoids cache invalidation errata.
// Call once at startup/initialization.
void SetIVTInRAM(void)
{
    static __attribute__((aligned(0x40))) void * ram_ivt[__IVT_NUM];
    // __IVT_NUM requires XC-DSC v3.20 or newer; if undefined, change to highest implemented
    // vector number

    if(IVTBASE != (uint32_t)ram_ivt)
    {
        __builtin_memcpy(ram_ivt, (const void*)IVTBASE, sizeof(ram_ivt));

        uint32_t PACCON1Save = PACCON1;
        PACCON1 |= 0x000D0000UL;        // BMXIRAMLWR/BMXIRAMHWR/IVTBASEWR must be set
       // Infinite loop if locked. SetIVTInRAM() MUST be called before code that locks BMXIRAM/IVTBASE
        while((PACCON1 & 0x000D000DUL) != 0x000D0000UL);

        // Specify ram_ivt[]'s address range as read-only + executable
        int originalIPLMask = __builtin_write_DISICTL(7);
        BMXIRAML = (uint32_t)ram_ivt;
        BMXIRAMH = ((uint32_t)ram_ivt) + sizeof(ram_ivt);
        IVTBASE = (uint32_t)ram_ivt;
        PACCON1 = PACCON1Save;
        __builtin_write_DISICTL(originalIPLMask);
    }
}
```

Work around 2

Temporarily block all IVT vector fetches while triggering cache invalidation events. This option suspends interrupts, but continues to allow traps to be detected and immediately serviced by a common handler routine.

Example 2:

```
void __attribute__((interrupt, nocontext)) IVTReadFailException(void)
{
    // Common trap handler code - will be reentrant if multiple dissimilar traps
    // of increasing priority take place.
    // Individual flag bits in INTCON1 and INTCON3 to INTCON5 may be polled to
    // determine the nature of this trap. Only flags that are serviced should be
    // cleared if this function is allowed to return so as to preserve status
    // for recursive instances.
}

// Cache invalidation errata workaround
void CacheInvalidationEvent(void)
{
// Fallback address to vector to when the IVT can't be read via the CPU-I bus
    VFA = (uint32_t)IVTReadFailException;

    uint32_t IVTBaseSave = IVTBASE;
    int DISICTLSave = __builtin_write_DISICTL(7);

    IVTBASE = 0;                            // Illegal IVT base address: CPU jumps to [VFA] if used
    NVMCONbits.WR = 1;  /* and/or */  CHECONbits.CHEINV = 1;
    (void)CHECON;                           // Dummy SFR read for prior SFR write to exit CPU pipeline
    IVTBASE = IVTBaseSave;                  // Reenable IVT usage
    __builtin_write_DISICTL(DISICTLSave);   // Reenable interrupts}
}
```

Since this workaround blocks interrupts and RTSP code often sets NVMCON[WR] many times before executing from any of the erased/reprogrammed address ranges, it is suggested that the CHECON[CHECOH] bit be cleared at reset/code initialization. In such cases, this workaround would only need to be applied to manual cache invalidation, not NVMCONbits.WR = 1 operations. After all RTSP operations are complete, one manually triggered cache invalidation can be performed if there is any chance stale instructions remain in the cache from before starting the RTSP session.

Example 3:

```
int main(void)
{
    CHECONbits.CHECOH = 0;      // Disable automatic cache invalidation each time NVMCONbits.WR is
                                // set - do manual invalidation only

    // Do all RTSP/bootloader operations

    CacheInvalidationEvent();   // Set CHECONbits.CHEINV = 1 in here, not NVMCONbits.WR

    // Safe to begin executing from the reprogrammed flash regions now
```

Affected Silicon Revisions

A2							
X							

© 2024-2026 Microchip Technology Inc. and its subsidiaries

30. Module: SPI

When the SPI module operates in Host mode (MSTEN = 1) with the peripheral bus selected as the clock source (MCLKEN = 0), the module may exhibit unreliable operation. Internal synchronization timing cannot be guaranteed across all operating conditions. This issue occurs regardless of SPI clock polarity/phase settings (CKP, CKE, SMP bits) and affects both dedicated SPI pins and remappable (PPS) pins. Client mode operation is not affected.

Work around

Configure the SPI module to use the dedicated clock generator (CLKGENx) as the clock source by setting SPIxCON1.MCLKEN = 1 when the module is used in Host mode (MSTEN = 1).

Affected Silicon Revisions

A2							
X							

31. Module: Debug

When the Program Counter matches certain hardware breakpoint addresses, a halt in Debug mode may be missed. Breakpoint addresses subject to being missed are the non-speculative execution addresses associated with a conditional branch, as well as the targets of dynamic program flow change instructions, such as computed branch offsets in C switch statements, function return addresses and function pointer destinations. The breakpoint will be missed if the NVM is busy at the time of the address match, resulting in breakpoints at uncached addresses being susceptible, while the same breakpoint address reached via a looping construct will normally trigger a successful debug halt.

Work around 1

Enable software breakpoints.

Work around 2

Add a second hardware breakpoint or relocate program breakpoints suspected of being missed to the next address or line of code.

Affected Silicon Revisions

A2							
X							

Data Sheet Clarifications

The following typographic corrections and clarifications are to be noted for the latest version of the device data sheet (DS70005539**C**).

Note:	Corrections are shown in **bold**. Where possible, the original bold text formatting has been removed for clarity.

None.

© 2024-2026 Microchip Technology Inc. and its subsidiaries

APPENDIX A: DOCUMENT REVISION HISTORY

Rev A Document (7/2024)

Initial version of this document; issued for silicon revision A2.

Rev B Document (3/2025)

Added silicon issue 28 (CPU).

Rev C Document (10/2026)

Added silicon issues 29 (CPU), 30 (SPI) and 31 (Debug).

© 2024-2026 Microchip Technology Inc. and its subsidiaries

Microchip Information

Trademarks

The “Microchip” name and logo, the “M” logo, and other names, logos, and brands are registered and unregistered trademarks of Microchip Technology Incorporated or its affiliates and/or subsidiaries in the United States and/or other countries (“Microchip Trademarks”). Information regarding Microchip Trademarks can be found at https://www.microchip.com/en-us/about/legal-information/microchip-trademarks.

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP "AS IS". MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.