PIC32CX SG41/SG60/SG61 Family Silicon Errata and Data Sheet Clarifications

PIC32CX SG41/SG60/SG61

PIC32CX SG41/SG60/SG61 Family Errata

The PIC32CX SG41/SG60/SG61 family of devices that you have received conform functionally to the current device data sheet (DS60001715), 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 the following tables. The silicon issues are summarized in the Table of Contents following this section.

The errata described in this document will be addressed in future revisions of the PIC32CX SG41/SG60/SG61 family of devises.

Note: This document summarizes all silicon errata issues from all revisions of silicon, previous and current.

Data Sheet clarifications and corrections (if applicable) are located in the section 5. Data Sheet Clarifications, following the discussion of silicon issues.

Table 1. PIC32CX SG41 Family Silicon Device Identification

Devices	Device ID (DID[31:0])	Silicon Revision ID (DID.REVISION[3:0])
		B1
PIC32CX1025SG41100	0x61870x01	0x5
PIC32CX1025SG41128	0x61870x00	0x5

Table 2. PIC32CX SG60 Family Silicon Device Identification

Devices	Device ID (DID[31:0])	Silicon Revision ID (DID.REVISION[3:0])	
		A0	B1
PIC32CX1025SG60100	0x61800x01	-	0x5
PIC32CX1025SG60128	0x61800x00	0x3	0x5

Table 3. PIC32CX SG61 Family Silicon Device Identification

Devices	Device ID (DID[31:0])	Silicon Revision ID (DID.REVISION[3:0])
		B1
PIC32CX1025SG61100	0x61820x01	0x5
PIC32CX1025SG61128	0x61820x00	0x5

Note: Refer to the **"Device Service Unit"** chapter in the current device data sheet (DS60001715) for a detailed information on Device Identification and Revision IDs for your specific device.

© 2023 Microchip Technology Inc. and its subsidiaries

Table of Contents

© 2023 Microchip Technology Inc. and its subsidiaries

1. Silicon Errata Summary

Table 1-1. Silicon Errata Summary

Module	Feature	Item Number	Issue Summary	Affected Revisions	
				A0	B1
AC	AC Hysteresis	2.1.1	Enabling Hysteresis (COMPCTRLn.HYSTEN = 0x1) changes the threshold voltage (VTH-), which could result in unexpected behavior of the Analog Comparator.	X	
AC	Standby Sleep Mode	2.1.2	A comparison in Single Shot mode will not be completed when entering in Standby Sleep mode with RUNSTDBY = 0.	X	X
CAN	CAN Edge Filtering	2.2.1	When edge filtering is activated (CCCR.EFBI = 1) and when the end of the integration phase coincides with a falling edge at the Rx input pin, it may occur that the CAN synchronizes itself incorrectly and does not correctly receive the first bit of the frame. In this case, the CRC will detect the first bit that was received incorrectly, it will rate the received FD frame as faulty, and an error frame will be send.	X	X
CAN	Dominant Bit of Intermission	2.2.2	When NBTP.NTSEG2 is configured to zero (Phase_Seg2(N) = 1), and when there is a pending transmission request, a dominant third bit of Intermission may cause the CAN to wrongly transmit the first identifier bit dominant instead of recessive, even if this bit was configured as '1' in the Tx Buffer Element of the CAN module.	X	X
CAN	INTFLAG Status	2.2.3	Message transmitted with wrong arbitration and control fields.	X	X
CAN	DAR Mode	2.2.4	Retransmission in DAR mode due to lost arbitration.	X	X
CAN	High Priority Message (HPM) interrupt	2.2.5	Unexpected High Priority Message (HPM) interrupt	X	X
CAN	TxFIFO	2.2.6	Tx FIFO message sequence inversion	X	X
CAN	Debug Message	2.2.7	Debug message handling state machine not reset to Idle state when CCCR.INIT is set.	X	X
CAN	Transmit Message Order Inversion	2.2.8	It may happen, depending on the delay between the individual Tx requests, that in the case where multiple Tx Buffers are configured with the same Message ID the Tx Buffers are not transmitted in order of the Tx Buffer number (lowest number first).	X	X
CAN	Tx (Queue) Buffers	2.2.9	Use of multiple dedicated Tx Buffers or Tx Queue buffers configured with same Message ID.	X	X
CAN	Transmit Cancellation	2.2.10	Frame transmitted despite confirmed transmit cancellation.	X	X
Device	Standby Mode	2.3.1	STANDBY low power mode is not supported and should not be used in new designs.	X	
Device	Power Up	2.3.2	The cache lines 0 (AHB0) and 1 (AHB1), which are enabled by default at reset, may not be reset properly at power-up in some very rare circumstances.	X	X
DMAC	Linked Descriptors	2.4.1	When at least one channel using linked descriptors is already active, a channel Fetch Error (FERR) could occur on enabling a channel with no linked descriptor or the second descriptor (index 1) of the channel being enabled could be fetched by one of the already active channels using linked descriptors.	X	
NVMCTRL	NVM Read Corruption	2.5.1	NVM reads could be corrupted when mixing NVM reads with Page Buffer writes.	X	
OSCCTRL	Input Clock on FDPLLn	2.6.1	Several FDPLL unlocks could occur while the output frequency is stable.	X	
RTC	Tamper Detection Timestamp	2.7.1	The INTFLAG.TAMPER bit is not reset by reading the TIMESTAMP register.	X	X

© 2023 Microchip Technology Inc. and its subsidiaries

.....continued

Module	Feature	Item Number	Issue Summary	Affected Revisions	
				A0	B1
SERCOM I²C	SDAHOLD Timing	2.8.1	SDAHOLD timing of the SERCOM-I²C does not match the value shown in the current device data sheet.	X	X
SERCOM I²C	Repeated Start in High-Speed Host Write Operation	2.8.2	For High-Speed Host Write operations, writing CTRLB.CMD = 0x1 issues a STOP command instead of a Repeated Start making repeated start not possible in that mode.	X	X
SERCOM I²C	Repeated Start in High-Speed Host Read Operation	2.8.3	For High-Speed Host Read operations, sending a NACK (CTRLB.CMD = 0x2) forces a STOP to be issued making repeated start not possible in that mode.	X	X
SERCOM I²C	Client Mode with DMA	2.8.4	In I²C Client Transmitter mode, at the reception of a NACK, if there is still data to be sent in the DMA buffer, the DMA will push a data to the DATA register.	X	X
SERCOM I²C	I²C Client in DATA32B Mode	2.8.5	When SERCOM is configured as an I²C Client in 32-bit Data Mode (DATA32B = 1) and the I²C Host reads from the I²C Client (client transmitter) and outputs its NACK (indicating no more data is needed), the I²C Client still receives a DRDY interrupt.	X	X
SERCOM I²C	10-bit Addressing Mode	2.8.6	10-bit addressing in I²C Client mode is not functional.	X	X
SERCOM I²C	Repeated Start	2.8.7	For Host Write operations (excluding High-Speed mode) in 10-bit addressing mode, writing CTRLB.CMD = 0x1 does not issue correctly a Repeated Start command.	X	X
SERCOM I²C	Wakeup	2.8.8	When an unexpected STOP occurs on the I²C bus, the STATUS.BUSERR and INTFLAG.ERROR bits are set but may not wake the system from Sleep mode. An unexpected START will not produce this issue.	X	X
SERCOM USART	USART Auto-Baud Mode	2.9.1	In USART Auto-Baud mode, missing stop bits are not recognized as inconsistent sync (ISF) or framing (FERR) errors.	X	X
SERCOM USART	TXINV and RXINV Bits	2.9.2	The TXINV and RXINV bits in the CTRLA register have inverted functionality.	X	X
SERCOM USART	LENGTH	2.9.3	When the USART is used in 32-bit mode with hardware handshaking (CTS/RTS), the TXC interrupt flag (INTFLAG.TXC) may be set before transmission has completed. The TXC interrupt flag may incorrectly be set regardless of Data Length Enable (LENGTH.LENEN) is set to '0' or '1'.	X	X
SERCOM USART	LIN Host Delays	2.9.4	In SERCOM USART LIN Host Mode, in the case where break, sync and identifier fields are automatically transmitted when DATA is written with the identifier, the LIN Host Header delay between the sync and the ID transmission fields is not correct.	X	X
SERCOM USART	Two stop bits mode in LIN Host	2.9.5	Two stop bits mode is not supported in SERCOM USART LIN Host Mode in the case where break, sync and identifier fields are automatically transmitted when DATA is written with the identifier.	X	X
TCC	Auto Lock Feature	2.10.1	The Auto Lock (CTRL.ALOCK) feature is not functional.	X	X
TCC	MCx Interrupt Flags	2.10.2	In capture operation, MC0/MC1 interrupt status flags are not automatically cleared when CC0/CC1 register are read.	X	X
TC	MCx Interrupt Flags	2.11.1	In capture operation, MC0/MC1 interrupt status flags are not automatically cleared when CC0/CC1 register are read.	X	X
FREQM	Lost Interrupt	2.12.1	DONE interrupt may be lost.	X	X
FREQM	STATUS.BUSY	2.12.2	No timeout period for a FREQM measurement cycle.	X	X

Notes:

- Cells with 'X' indicates the issue is present in this revision of the silicon.
- Cells with '-' indicates this silicon revision does not exist for this issue.
- The blank cell indicates the issue has been corrected or does not exist in this revision of the silicon.

© 2023 Microchip Technology Inc. and its subsidiaries

2. Silicon Errata Issues

The following errata issues apply to the PIC32CX SG41/SG60/SG61 family of devices.

Note: Cells with an 'X' indicates the issue is present in this revision of the silicon.
Cells with a dash ('-') indicate this silicon revision does not exist for this issue.

Blank cells indicate the issue has been corrected or does not exist in this revision of the silicon.

2.1 Analog Comparator (AC)

2.1.1 AC Hysteresis

Enabling Hysteresis (COMPCTRLn.HYSTEN = 0x1) changes the threshold voltage (VTH-), which can result in unexpected behavior of the Analog Comparator.

Workaround

None.

Affected Silicon Revisions

A0	B1					
X						

2.1.2 Standby Sleep Mode

A comparison in Single Shot mode will not be completed when entering in Standby Sleep mode with RUNSTDBY = 0.

Workaround

Set RUNSTBDY = 1 before entering in Standby mode or wait for Standby exit to start a new comparison.

Affected Silicon Revisions

A0	B1					
X	X					

2.2 Controller Area Network (CAN)

2.2.1 CAN Edge Filtering

When edge filtering is activated (CCCR.EFBI = 1) and when the end of the integration phase coincides with a falling edge at the Rx input pin, it may occur that the CAN synchronizes itself incorrectly and does not correctly receive the first bit of the frame. In this case, the CRC will detect the first bit that was received incorrectly, it will rate the received FD frame as faulty, and an error frame will be send.

The issue occurs only when there is a falling edge at the Rx input pin (CAN_RX) within the last time quantum (tq) before the end of the integration phase. The last time quantum of the integration phase is at the sample point of the 11th recessive bit of the integration phase. When edge filtering is enabled, the bit timing logic of the CAN sees the Rx input signal delayed by the edge filtering. When the integration phase ends, edge filtering is automatically disabled. This affects the reset of the FD CRC registers at the beginning of the frame. The Classical CRC register is not affected, hence this issue does not affect the reception of Classical frames.

In CAN communication, the CAN module may enter an integrating state (either by resetting the CCCR.INIT or by protocol exception event) while a frame is active on the bus. In this case, the 11 recessive bits are counted between the acknowledge bit and the following start of frame. All nodes have synchronized at the beginning of the dominant acknowledge bit. This means that the edge of the following start of frame bit cannot fall on the sample point, therefore the issue does not occur.

© 2023 Microchip Technology Inc. and its subsidiaries

The issue occurs only when the CAN is by local errors, mis-synchronized with regard to the other nodes.

Glitch filtering as specified in ISO 11898-1:2015 is fully functional.

Edge filtering was introduced for applications where the data bit time is at least 2-tq (of nominal bit time) long. In that case, edge filtering requires at least two consecutive dominant time quanta before the counter counting the 11 recessive bits for idle detection is restarted. This means edge filtering covers the theoretical case of occasional 1-tq long dominant spikes on the CAN bus that would delay idle detection. Repeated dominant spikes on the CAN bus would disturb all CAN communication, hence the filtering to speed up idle detection will not help network performance.

When this rare event occurs, the CAN sends an error frame and the sender of the affected frame retransmits the frame. When the retransmitted frame is received, the CAN has left the integration phase and the frame will be received correctly. Edge filtering is only applied during the integration phase and it is never used during normal operation. Because the integration phase is very short with respect to "active communication time", the impact on total error frame rate is negligible. The issue has no impact on data integrity.

The CAN enters the integration phase under the following conditions:

- When CCCR.INIT is set to '0' after start-up
- After a protocol exception event (only when CCCR.PXHD = 0)

Scope:

The erratum is limited to FD frame reception when edge filtering is active (CCCR.EFBI = 1) and when the end of the integration phase coincides with a falling edge at the Rx input pin.

Effects:

The calculated CRC value does not match the CRC value of the received FD frame and the CAN module sends an error frame. After retransmission the frame is received correctly.

Workaround:

Disable edge filtering or wait on retransmission in the event that this rare event occurs.

Affected Silicon Revisions

A0	B1					
X	X					

2.2.2 Dominant Bit of Intermission

When NBTP.NTSEG2 is configured to zero (Phase_Seg2(N) = 1), and when there is a pending transmission request, a dominant third bit of intermission may cause the CAN to wrongly transmit the first identifier bit dominant instead of recessive, even if this bit was configured as '1' in the TX Buffer Element of the CAN module.

Workaround

A phase buffer segment 2 of length '1' (Phase_Seg2(N) = 1) is not sufficient to switch to the first identifier bit after the sample point in intermission where the dominant bit was detected.

The CAN protocol according to ISO 11898-1 defines that a dominant third bit of intermission causes a pending transmission to be started immediately. The received dominant bit is handled as if the CAN has transmitted a Start-of-Frame (SoF) bit.

The ISO 11898-1 specifies the minimum configuration range for Phase_Seg2(N) to be 2..8 tq. Therefore, excluding a Phase_Seg2(N) of '1' will not affect CAN conformance.

Effects:

© 2023 Microchip Technology Inc. and its subsidiaries

If NBTP.NTSEG2 = 0, it may occur that the CAN transmits the first identifier bit dominant instead of recessive.

Update configuration range of NBTP.NTSEG2 from 0..127 tq to 1..127 tq in the CAN documentation.

Affected Silicon Revisions

A0	B1					
X	X					

2.2.3 Message Transmitted with Wrong Arbitration and Control Fields

Description:

Under the following conditions a message with wrong ID, format, and DLC is transmitted:

- The CAN is in the "Receiver" (PSR.ACT ≠0b10) state, hence no pending transmission.
- A new transmission is requested before the third bit of intermission is reached.
- The CAN bus is sampled dominant at the third bit of intermission which is treated as SoF (See ISO11898-1:2015, “Section 10.4.2.2”).

Under the conditions above, the following might happen:

- The shift register is not loaded with ID, format, and DLC of the requested message.
- The CAN will start arbitration with wrong ID, format, and DLC on the next bit.
- If the ID wins arbitration, a CAN message with a valid CRC is transmitted.
- If this message is acknowledged, the ID stored in the TX event FIFO is the ID of the requested TX message and not the ID of the message transmitted on the CAN bus, hence no error is detected by the transmitting CAN.

Scope:

The erratum is limited when CAN is in the "Receiver" (PSR.ACT = 0b10) state with no pending transmission (register TXBRP == 0) and a new transmission is requested before the third bit of intermission is reached and this third bit of intermission is seen dominant.

When a transmission is requested by the CPU by writing to TXBAR, the TX message handler performs an internal arbitration and loads the pending transmit message with the highest priority into its output buffer and then sets the transmission request for the CAN Protocol Controller. The problem occurs only when the transmission request for the CAN Protocol Controller is activated in the critical time window between the sample points of the second and third bit of intermission and if that third bit of intermission is seen dominant.

This dominant level at the third bit of intermission may result from an external disturbance or may be transmitted by another node with a significantly faster clock.

Effects:

In the described case it may happen that the shift register is not loaded with arbitration and control field of the message to be transmitted. The frame is transmitted with wrong ID, format, and DLC but with the data field of the requested message. The message is transmitted in correct CAN (FD) frame format with a valid CRC.

If the message loses arbitration or is disturbed by an error, it is retransmitted with correct arbitration and control fields.

Workarounds

- **Workaround 1:** Request a new transmission only if another transmission is already pending (that is, register TXBRP ≠ 0) or when the CAN is not in the "Receiver" (when PSR.ACT ≠ 0b10) state. To avoid activating the transmission request in the critical time window between the sample points of the second and third bit of intermission, the application software can evaluate the Rx interrupt

© 2023 Microchip Technology Inc. and its subsidiaries

flags, such as IR.DRX, IR.RF0N, and IR.RF1N, which are set at the last bit of EoF when a received and accepted message becomes valid. The last bit of EoF is followed by third bits of intermission. Therefore the critical time window has safely terminated three bit times after the Rx interrupt. Now a transmission may be requested by writing to TXBAR. After the interrupt, the application has to take care that the transmission request for the CAN Protocol Controller is activated before the critical window of the following reception is reached.

- **Workaround 2:** If a transmission is to be requested while no other transmission request is already pending and the CAN bus is not idle, set the CCCR.INIT bit (which stops the CAN protocol controller), set the transmission request and clear the CCCR.INIT bit. The message currently being received when the CCCR.INIT bit is set will be lost, but no errors (or error frames) will be generated and the CAN protocol controller will re-integrate into the CAN communication immediately at the 11 recessive bits of the next End-of-Frame including intermission.
- **Workaround 3:**It is also possible to keep the number of pending transmissions always at > 0 by frequently requesting a message, then the condition "no pending transmission" is never met. The frequently requested message may be given a low priority, losing arbitration to all other messages.

Affected Silicon Revisions

A0	B1					
X	X					

2.2.4 DAR Mode

When the CAN is configured in DAR mode (CCCR.DAR = 1) the automatic retransmission for transmitted messages that have been disturbed by an error or have lost arbitration is disabled. When the transmission attempt is not successful, the TX Buffer's Transmission Request bit (TXBRP.TRPn) will be cleared and the TX Buffer's Cancellation Finished bit (TXBCF.CFn) will be set.

When the transmitted message loses arbitration at one of the first two identifier bits, chances are that instead of the bits of the actually transmitted TX Buffer, the TXBRP.TRPn and TXBCF.CFn bits of the previously started TX Buffer (or TX Buffer 0 if there is no previous transmission attempt) are written (TXBRP.TRPn = 0, TXBCF.CFn = 1).

If in this case the TXBRP.TRPn bit of the TX Buffer that lost arbitration at the first two identifier bits are not cleared, retransmission is attempted. When the CAN loses arbitration again at the immediately following retransmission, then actually and previously transmitted TX Buffer are the same and this TX Buffer's TXBRP.TRPn bit is cleared and its TXBCF.CFn bit is set.

Scope:

The erratum is limited to the case when the CAN loses arbitration at one of the first two transmitted identifier bits while in DAR mode. The problem does not occur when the transmitted message is disturbed by an error.

Effects:

In this case, it might happen that the TXBRP.TRPn bit is cleared after the second transmission attempt instead of the first. Additionally it may happen that the TXBRP.TRPn bit of the previously started TX Buffer is cleared, if it has been set again. As in this case the previously started TX Buffer has lost CAN internal arbitration against the active TX Buffer, its message has a lower identifier priority. It would also have lost arbitration on the CAN bus at the same position.

Workaround

None.

Affected Silicon Revisions

A0	B1					
X	X					

© 2023 Microchip Technology Inc. and its subsidiaries

2.2.5 High-Priority Message (HPM) interrupt

There are two configurations where the issue occurs:

Configuration A:

- At least one Standard Message ID Filter Element is configured with priority flag set (S0.SFEC = 0b100/0b101/0b110).
- No Extended Message ID Filter Element configured.
- Non-matching extended frames are accepted (GFC.ANFE = 0b00/0b01).

The HPM interrupt flag IR.HPM is set erroneously on reception of a non-high-priority extended message under the following conditions:

1. A standard HPM frame is received and accepted by a filter with priority flag set (that is, Interrupt flag IR.HPM is set as expected).
2. An extended frame is received and accepted because of the GFC.ANFE configuration (that is, Interrupt flag IR.HPM is set erroneously).

Configuration B:

- At least one Extended Message ID filter element is configured with priority flag set (F0.EFEC = 0b100/0b101/0b110).
- No Standard Message ID filter element is configured.
- Non matching standard frames are accepted (GFC.ANFS = 0b00/0b01).

The HPM interrupt flag IR.HPM is set erroneously on reception of a non high-priority standard message under the following conditions:

1. An extended HPM frame is received and accepted by a filter with priority flag set (that is, Interrupt flag IR.HPM is set as expected).
2. A standard frame is received and accepted because of the GFC.ANFS configuration (that is, Interrupt flag IR.HPM is set erroneously).

Scope:

The erratum is limited to the following configurations:

Configuration A:

No Extended Message ID filter element is configured and non-matching extended frames are accepted due to Global Filter Configuration (GFC.ANFE = 0b00/0b01).

Configuration B:

No Standard Message ID Filter Element configured and non-matching standard frames are accepted due to Global Filter Configuration (GFC.ANFS = 0b00/0b01).

Effects:

Interrupt flag IR.HPM is set erroneously at the reception of a frame with:

- Configuration A: Extended Message ID
- Configuration B: Standard Message ID

Workaround

Configuration A:

Setup an Extended Message ID filter element with the following configuration:

- F0.EFEC = 001/010: Select Rx FIFO for storage of extended frames
- F0.EFID1 = any value: The value is not relevant as all ID bits are masked out by F1.EFID2

© 2023 Microchip Technology Inc. and its subsidiaries

- F1.EFT = 10: Classic filter, F0.EFID1 = filter, F1.EFID2 = mask
- F1.EFID2 = 0: All bits of the received extended ID are masked out

Now all extended frames are stored in Rx FIFO ‘0’ or Rx FIFO ‘1’ depending on the configuration of F0.EFEC.

Configuration B:

Setup an Standard Message ID filter element with the following configuration:

- S0.SFEC = 001/010: Select Rx FIFO for storage of standard frames
- S0.SFID1 = any value: The value is not relevant as all ID bits are masked out by S0.SFID2
- S0.SFT = 10: Classic filter, S0.SFID1 = filter, S0.SFID2 = mask
- S0.SFID2 = 0: All bits of the received standard ID are masked out

Now all standard frames are stored in Rx FIFO ‘0’ or Rx FIFO ‘1’ depending on the configuration of S0.SFEC.

Affected Silicon Revisions

A0	B1					
X	X					

2.2.6 TX FIFO message sequence inversion

Assuming that there are two TX FIFO messages in the output pipeline of the TX Message Handler. Transmission of TX FIFO message 1 is started:

Position 1: TX FIFO message 1 (transmission ongoing)

Position 2: TX FIFO message 2

Position 3: Free FIFO bugger

During the transmission of TX FIFO message 1, a non TX FIFO message with a higher CAN priority is requested. Due to its priority it will be inserted into the output pipeline. The TXMH performs "message scans" to keep the output pipeline up to date with the highest priority messages from the message RAM.

After the following two message scans, the output pipeline has the following content:

Position 1: TX FIFO message 1 (transmission ongoing)

Position 2: non TX FIFO message with higher CAN priority

Position 3: TX FIFO message 2

If the transmission of TX FIFO message 1 is not successful (lost arbitration or CAN bus error) it is pushed from the output pipeline by the non TX FIFO message with higher CAN priority. The following scan again inserts TX FIFO message 1 into the output pipeline at position 3:

Position 1: non TX FIFO message with higher CAN priority (transmission ongoing)

Position 2: TX FIFO message 2

Position 3: TX FIFO message 1

This results in TX FIFO message 2 being in the output pipeline in front of TX FIFO message 1 and they are transmitted in that order, resulting in a message sequence inversion.

Scope:

The erratum describes the case when the CAN uses both, dedicated TX Buffers and a TX FIFO (TXBC.TFQM = 0) and the messages in the TX FIFO do not have the highest internal CAN priority. The described sequence inversion may also happen between two non TX FIFO messages (TX Queue or

© 2023 Microchip Technology Inc. and its subsidiaries

dedicated TX Buffers) that have the same CAN identifier and that should be transmitted in the order of their buffer numbers (not the intended use).

Effects:

In the described case it may happen that two consecutive messages from the TX FIFO exchange their positions in the transmit sequence.

Workarounds

When transmitting messages from a dedicated TX Buffer with higher priority than the messages in the TX FIFO, choose one of the following workarounds:

Workaround 1

Use two dedicated TX Buffers, for example, use TX Buffers 4 and 5 instead of the TX FIFO.

The Transmit Loop below replaces the function that fills the TX FIFO.

Write the message to TX Buffer 4

Transmit Loop:

- Request TX Buffer 4 - write TXBAR.A4
- Write message to TX Buffer 5
- Wait until transmission of TX Buffer 4 completed - IR.TC, read TXBTO.TO4
- Request TX Buffer 5 - write TXBAR.A5
- Write message to TX Buffer 4
- Wait until transmission of TX Buffer 5 completed - IR.TC, read TXBTO.TO5

Workaround 2

Ensure that only one TX FIFO element is pending for transmission at any time.

The TX FIFO elements may be filled at any time with messages to be transmitted, but their transmission requests are handled separately. Each time a TX FIFO transmission has completed and the TX FIFO becomes empty (IR.TFE = 1), the next TX FIFO element is requested.

Workaround 3

Use only a TX FIFO. Send the message with the higher priority also from TX FIFO.

Drawback: The higher priority message has to wait until the preceding messages in the TX FIFO are sent.

Affected Silicon Revisions

A0	B1					
X	X					

2.2.7 Debug Message

If the CCCR.INIT bit is set by the Host by writing to the CCCR register or when the M_CAN enters BusOff state, the debug message handling state machine stays in its current state instead of being reset to Idle state. Setting CCCR.CCE does not change RXF1S.DMS.

Scope:

The erratum is limited to the case when the Debug on CAN Support feature is active. Regular operation is not affected. In regular operation the debug message handling state machine always remains in Idle state.

Effects:

© 2023 Microchip Technology Inc. and its subsidiaries

In the described case the debug message handling state machine is stopped and remains in the current state signaled by RXF1S.DMS. In case the RXF1S.DMS = “11”, the output `m_can_dma_req` remains active.

Workaround

If the debug message handling state machine has stopped while RXF1S.DMS = “01” or RXF1S.DMS = “10”, it can be reset to Idle state by hardware reset or by reception of debug messages after CCCR.INIT is reset to zero.

If the debug message handling state machine has stopped while RXF1S.DMS = “11” with `m_can_dma_req` active, it can be reset to Idle state by hardware reset or by activating input `m_can_dma_ack`.

Affected Silicon Revisions

A0	B1					
X	X					

2.2.8 Transmit Message Order Inversion

It may happen, depending on the delay between the individual TX requests, that in the case where multiple TX Buffers are configured with the same Message ID the TX Buffers are not transmitted in order of the TX Buffer number (lowest number first).

Scope:

The erratum is limited to the case when multiple TX Buffers are configured with the same Message ID.

Effects:

In the case described it may happen, that TX Buffers configured with the same Message ID and pending TX request are not transmitted with lowest TX Buffer number first (message order inversion).

Workaround

First write the group of TX messages with same Message ID to the Message RAM and then afterwards request transmission of all these messages concurrently by a single write access to TXBAR. Before requesting a group of TX messages with this Message ID ensure that no message with this Message ID has a pending TX request.

Affected Silicon Revisions

A0	B1					
X	X					

2.2.9 TX (Queue) Buffers

Scope:

Use of multiple dedicated TX Buffers or TX Queue buffers configured with same Message ID.

Effects:

If the dedicated TX buffers with the same Message ID are not requested in ascending order or at the same time or in case of multiple TX Queue buffers with the same Message ID, it cannot be guaranteed, that these messages are transmitted in ascending order with lowest buffer number first.

Workaround

If a defined order of transmission is required, the TX FIFO can be used for transmission of messages with the same Message ID. Alternatively dedicated TX buffers with same message ID shall be

© 2023 Microchip Technology Inc. and its subsidiaries

requested in ascending order with lowest buffer number first or by a single write access to TXBAR. Alternatively a single TX Buffer can be used to transmit those messages one after the other.

Affected Silicon Revisions

A0	B1					
X	X					

2.2.10 Transmit Cancellation

If all of the following events have occurred:

1. Transmission of the TX Buffer nn was not successful.
2. The automatic retransmission mechanism has started to resend the TX Buffer nn and is currently sending the first four identifier bits.
3. The user attempts to cancel the transaction by setting the TXBCR.CRnn bit.

The hardware will send a signal stating that the transaction or transmission was successfully canceled even when it has not been TXBCF.CFnn = 1, TXBRP.TRPnn = 0. The hardware also states that frame was not transmitted on the bus TXBTO.TOnn = 0.

Other than what is signaled by TXBCF.CFnn and TXBTO.TOnn, the transmission continues until the complete frame has been sent on the CAN bus.

If the transmission is successful, the TXBTO.TOnn bit will be set. In this case new data is written to the TX Buffer nn while the transmission is still ongoing. A frame with inconsistent data may appear on the bus.

Scope:

This problem is limited to the case of transmit cancellation of CAN FD messages with more than eight data bytes, while automatic retransmission is enabled (CCCR.DAR = '0'). Transmit cancellation of Classical CAN messages and CAN FD messages with up to eight data bytes is not affected.

Effects:

When the TXBRP.TRPnn bit of the TX Buffer nn is reset by an incomplete transmit cancellation, this TX Buffer is reported to be "free". If software now writes new data to this TX Buffer while a transmission is still ongoing, this new data may be loaded into the protocol controller, leading to a data inconsistency of the transmitted frame. This means that the transmitted frame consists partly of the data available at start of frame, and data written to the TX Buffer during the ongoing transmission.

Workaround

Do not use transmit cancellation for CAN FD messages with more than eight data bytes.

Alternatively wait for the duration of the expected transmission time of the canceled TX Buffer before writing new data to that TX Buffer.

Affected Silicon Revisions

A0	B1					
X	X					

2.3 Device

2.3.1 Standby Mode

Standby Low-Power mode is not supported and should not be used in new designs.

Workaround

The Idle Low-Power mode must be considered if Hibernate or Backup sleep modes are not an option.

© 2023 Microchip Technology Inc. and its subsidiaries

Idle mode must be tuned by applying low-power techniques, such as waiting for any peripheral pending transactions to complete or switching on a slow clock source before entering the Low-Power mode.

Affected Silicon Revisions

A0	B1					
X						

2.3.2 Power Up

The cache lines 0 (AHB0) and 1 (AHB1), which are enabled by default at reset, may not be reset properly at power-up under extremely rare circumstances.

In such rare situations, different types of data corruption may happen on the two different cache lines:

- **Cache line 0 (AHB0):** A CPU fetch of the reset vector could be corrupted, preventing the application from booting correctly.
- **Cache line 1 (AHB1):** Access from one of the following Hosts (DSU, ICM, DMA) could be corrupted.

Workaround

- **Cache line 0 (AHB0):** Enable the Watchdog timer at power on with the lowest period possible by programming the WDT Enable and WDT Period bit fields from the NVM USER ROW (UROW) as follows:
  - WDT Enable = 0x1
  - WDT Period = 0x0
- **Cache line 1 (AHB1):** Clean up the Cache line 1 by toggling the NVMCTRL CTRLA.CACHEDIS1 bit before any Host (DSU, ICM, DMA) accesses as follows:
  - CACHEDIS1 = 0x1
  - CACHEDIS1 = 0x0

Affected Silicon Revisions

A0	B1					
X	X					

2.4 Direct Memory Access Controller (DMAC)

2.4.1 Linked Descriptors

When at least one channel using linked descriptors is already active, a channel Fetch Error (FERR) could occur on enabling a channel with no linked descriptor or the second descriptor (index 1) of the channel being enabled could be fetched by one of the already active channels using linked descriptors. These errors can occur when a channel is being enabled during the link request of another channel and if the channel number of the channel being enabled is lower than the channel already active.

Workaround

When enabling a channel while other channels using linked descriptors are already active, the channel number of the new channel to enable must be greater than the other channel numbers.

Affected Silicon Revisions

A0	B1					
X						

© 2023 Microchip Technology Inc. and its subsidiaries

2.5 Non-Volatile Memory Controller (NVMCTRL)

2.5.1 NVM Read Corruption

NVM reads could be corrupted when mixing NVM reads with Page Buffer writes.

Workaround

Disable cache lines before writing to the Page Buffer when executing from NVM or reading data from NVM while writing to the Page Buffer. Cache lines are disabled by writing a one to CTRLA.CACHEDIS0 and CTRLA.CACHEDIS1.

Affected Silicon Revisions

A0	B1					
X						

2.6 Oscillator Controller (OSCCTRL)

2.6.1 Input Clock on FDPLLn

FDPLL unlocks can occur while the output frequency is stable.

Workaround

Enable the lock bypass (OSCCTRL.DPLLCTRLB.LBYPASS = 1) and wake up fast (OSCCTRL.DPLLCTRLB.WUF = 1) to avoid losing FDPLL clock output during a false unlock status. The workaround does not avoid false unlock indications, but it disables the gating of the FDPLL clock output by the lock status. Therefore, the clock is issued even if the FDPLL status shows unlocked. The Clock Ready bit (OSCCTRL.DPLLSTATUS.CLKRDY) can be monitored by the application to ensure activity is present on the FDPLLn output, but the clock ready does not provide any indication of the FDPLLn lock nor frequency. A 10 ms delay is also suggested after the clock ready bit is set to allow the DPLL to achieve the target frequency.

Pseudo Code

Set OSCCTRL.DPLLCTRLB.WUF = 1 and OSCCTRL.DPLLCTRLB.LBYPASS = 1

Set DPLLCTRLA.ENABLE = 1

Wait (OSCCTRL.DPLLSTATUS.CLKRDY == 1)

Delay (10ms)

Set Source for GCLK with DPLL

Affected Silicon Revisions

A0	B1					
X						

2.7 Real-Time Counter (RTC)

2.7.1 Tamper Detection Timestamp

When DMA is enabled (CTRLB.DMAEN = 1), the INTFLAG.TAMPER bit is not reset by reading the TIMESTAMP register.

Workaround

Clear the INTFLAG.TAMPER bit by writing a ‘1’ to this bit when the Timestamp value has been read by the DMA.

© 2023 Microchip Technology Inc. and its subsidiaries

Affected Silicon Revisions

A0	B1					
X	X					

2.8 SERCOM I²C

2.8.1 SDAHOLD Timing

SDAHOLD timing of the SERCOM-I²C does not match the value shown in the current device data sheet. The following table shows the specified and real values of SDA Hold timing.

Table 2-1. SDA Hold Timing

SDA Hold Time Value	Specified SDA Hold Time	Real SDA Hold Time
0x0	Disabled	Disabled
0x1	50 ns to 100 ns	20 ns to 40 ns
0x2	300 ns to 600 ns	100 ns to 250 ns
0x3	400 ns to 800 ns	150 ns to 350 ns

Workaround

None.

Affected Silicon Revisions

A0	B1					
X	X					

2.8.2 Repeated Start in High-Speed Host Write Operation

For High-Speed Host Write operations, writing CTRLB.CMD = 0x1 issues a STOP command instead of a Repeated Start making repeated start not possible in that mode.

Workaround

None.

Affected Silicon Revisions

A0	B1					
X	X					

2.8.3 Repeated Start in High-Speed Host Read Operation

For High-Speed Host Read operations, sending a NACK (CTRLB.CMD = 0x2) forces a STOP to be issued making repeated start not possible in that mode.

Workaround

None.

Affected Silicon Revisions

A0	B1					
X	X					

2.8.4 Client Mode with DMA

In I²C Client Transmitter mode, at the reception of a NACK, if there is still data to be sent in the DMA buffer, the DMA will push a data to the DATA register. Since a NACK was received, the transfer on the I ²C bus will not occur causing the loss of this data.

© 2023 Microchip Technology Inc. and its subsidiaries

Workaround

Configure the DMA transfer size to the number of data to be received by the I²C Host. DMA cannot be used if the number of data to be received by the Host is not known.

Affected Silicon Revisions

A0	B1					
X	X					

2.8.5 I²C Client in DATA32B Mode

When SERCOM is configured as an I²C Client in 32-bit Data Mode (DATA32B = 1) and the I²C Host reads from the I²C Client (client transmitter) and outputs its NACK (indicating no more data is needed), the I²C Client still receives a DRDY interrupt.

If the CPU does not write a new data to the I²C Client DATA register, I²C Client will pull SDA line, which will result in stalling the bus permanently.

Workarounds

1. Write a dummy data to data register when a NACK is received from the Host.
2. Use command #2 (SERCOMx->I2CS.CTRLB.bit.CMD = 2) when a NACK is received from the Host.

Important: Because STATUS.RXNACK always indicates the last received ACK, to determine when a NACK is received from the I²C Host, the I²C Client software needs to consider I2CS.STATUS.RXNACK only on the second DRDY interrupt after receiving the AMATCH interrupt.

Affected Silicon Revisions

A0	B1					
X	X					

2.8.6 10-bit Addressing Mode

I²C Client 10-bit addressing mode is not functional.

Workaround

None.

Affected Silicon Revisions

A0	B1					
X	X					

2.8.7 Repeated Start

For Host Write operations (excluding High-Speed mode) in 10-bit addressing mode, writing CTRLB.CMD = 0x1 does not correctly issue a Repeated Start command..

Workaround

Write the same 10-bit address with the same direction bit to the ADDR.ADDR register to properly generate a Repeated Start.

Affected Silicon Revisions

A0	B1					
X	X					

© 2023 Microchip Technology Inc. and its subsidiaries

2.8.8 No Wakeup Upon Unexpected STOP

When an unexpected STOP occurs on the I^2C bus, the STATUS.BUSERR and INTFLAG.ERROR bits are set but may not wake the system from Sleep mode. An unexpected START will not produce this issue.

Workaround

None.

Affected Silicon Revisions

A0	B1					
X	X					

2.9 SERCOM USART

2.9.1 Auto-Baud Mode

In USART Auto-Baud mode, missing Stop bits are not recognized as inconsistent sync (ISF) or framing (FERR) errors.

Workaround

None.

Affected Silicon Revisions

A0	B1					
X	X					

2.9.2 TXINV and RXINV Bits

The TXINV and RXINV bits in the CTRLA register have inverted functionality.

Workaround

In software interpret the TXINV bit as a functionality of RXINV, and conversely, interpret the RXINV bit as a functionality of TXINV.

Affected Silicon Revisions

A0	B1				
X	X				

2.9.3 LENGTH

When the USART is used in 32-bit mode with hardware handshaking (CTS/RTS), the TXC interrupt flag (INTFLAG.TXC) may be set before transmission has completed. The TXC interrupt flag may incorrectly be set regardless of Data Length Enable bit (LENGTH.LENEN) is set to '0' or '1'.

Workaround

None.

Affected Silicon Revisions

A0	B1					
X	X					

2.9.4 LIN Host Delays

In SERCOM USART LIN Host mode (CTRLA.FORM = 0x2), in the case where break, sync, and identifier fields are automatically transmitted when data is written with the identifier (CTRLB.LINCMD = 0x2), the LIN Host Header delay between the sync and the ID transmission fields is not correct for the following cases:

© 2023 Microchip Technology Inc. and its subsidiaries

- CTRLC.HDRDLY = 0x2, where the delay between sync and ID transmission fields is 8-bit time instead of 4-bit time.
- CTRLC.HDRDLY = 0x3, where the delay between sync and ID transmission fields is 14-bit time instead of 4-bit time.

Workaround

None.

Affected Silicon Revisions

A0	B1					
X	X					

2.9.5 Two Stop Bits Mode in LIN Host

Two stop bits mode (CTRLB.SBMODE = 0x1) is not supported in SERCOM USART LIN Host mode (CTRLA.FORM = 0x2) in the case where break, sync, and identifier fields are automatically transmitted when data is written with the identifier (CTRLB.LINCMD = 0x2). Only one Stop bit is supported.

Workaround

None.

Affected Silicon Revisions

A0	B1					
X	X					

2.10 Timer/Counter for Control Applications (TCC)

2.10.1 Auto Lock Feature

The Auto Lock (CTRL.ALOCK) feature is not functional.

Workaround

None.

Affected Silicon Revisions

A0	B1					
X	X					

2.10.2 MCx Interrupt Flags

In capture operation, MC0/MC1 interrupt status flags (INTFLAG.MC0/INTFLAG.MC1) are not automatically cleared when the CC0/CC1 register is read.

Workaround

MC0/MC1 interrupt status flags must be cleared by the software (INTFLAG.MC0 = 1/INTFLAG.MC1 = 1).

Affected Silicon Revisions

A0	B1					
X	X					

2.11 Timer/Counter (TC)

2.11.1 MCx Interrupt Flags

In capture operation, MC0/MC1 interrupt status flags (INTFLAG.MC0/INTFLAG.MC1) are not automatically cleared when the CC0/CC1 register is read.

© 2023 Microchip Technology Inc. and its subsidiaries

Workaround

MC0/MC1 interrupt status flags must be cleared by software (INTFLAG.MC0 = 1/INTFLAG.MC1 = 1).

Affected Silicon Revisions

A0	B1					
X	X					

2.12 Frequency Meter (FREQM)

2.12.1 Lost Interrupt

The DONE interrupt may be lost when the measurement period (CFGA.REFNUM * the reference clock period) is less than 4 APB clock periods.

Workaround

The measurement reference period must be longer than 4 APB clock periods.

Affected Silicon Revisions

A0	B1					
X	X					

2.12.2 STATUS.BUSY

There is no timeout period for a FREQM measurement cycle. If the cycle count does not reach CFGA.REFNUM, the measurement cycle will not end and STATUS.BUSY will never deassert.

Workaround

If the measure signal may be very slow, may stop during the measurement or have a frequency of 0Hz, the application code must monitor the measurement cycle and terminate it in an appropriate period of time.

Affected Silicon Revisions

A0	B1					
X	X					

© 2023 Microchip Technology Inc. and its subsidiaries

3. Silicon Debug Related Errata Summary

Table 3-1. Silicon Debug Related Errata Summary

Module	Feature	Item Number	Issue Summary	Affected Revisions	
				A0	B1
AC	Continuous Comparisons	4.1.1	Continuous comparisons cannot be halted in Debug mode.	X	X
CAN	Bits Corruption	4.2.1	The ECR.CEL, PSR.PXE, PSR.RFDF, PSR.RBRS, PSR.RESI, PSR.DLEC and PSR.LEC bits can be corrupted by a debug access.	X	X
CAN	Bits Corruption	4.2.2	An expected clear on read of the ECR.CEL, PSR.PXE, PSR.RFDF, PSR.RBRS, PSR.RESI, PSR.DLEC and PSR.LEC bits can be unexpectedly filtered-out when the CPU is halted.	X	X
Device	Detection of a Debugger Probe	4.3.1	The detection of a debugger probe or a user reset (external reset, watchdog reset and system reset request) could fail if the "BOD33 Disable" fuse is cleared (i.e., BOD33 is enabled).	X	

Notes:

- Cells with ‘X’ indicates the issue is present in this revision of the silicon.
- Cells with ‘-’ indicates this silicon revision does not exist for this issue.
- The blank cell indicates the issue has been corrected or does not exist in this revision of the silicon.

© 2023 Microchip Technology Inc. and its subsidiaries

4. Silicon Debug Related Errata Issues

The following debug related errata issues apply to the PIC32CX SG41/SG60/SG61 family of devices.

Note: Cells with an 'X' indicate the issue is present in this revision of the silicon. Cells with a dash ('-') indicate this silicon revision does not exist for this issue.

Blank cells indicate the issue has been corrected or does not exist in this revision of the silicon.

4.1 Analog Comparator (AC)

4.1.1 Continuous Comparisons

Continuous comparisons cannot be halted in Debug mode.

Workaround

None.

Affected Silicon Revisions

A0	B1					
X	X					

4.2 Controller Area Network (CAN)

4.2.1 Bits Corruption

The ECR.CEL, PSR.PXE, PSR.RFDF, PSR.RBRS, PSR.RESI, PSR.DLEC and PSR.LEC bits can be corrupted by a debug access.

Workaround

Do not read the ECR, PSR registers with a debugger when the CPU is not halted in debug, otherwise debug access will clear those bits.

Affected Silicon Revisions

A0	B1					
X	X					

4.2.2 Bits Corruption

An expected clear on read of the ECR.CEL, PSR.PXE, PSR.RFDF, PSR.RBRS, PSR.RESI, PSR.DLEC and PSR.LEC bits can be unexpectedly filtered-out when the CPU is halted.

Workaround

Do not halt the CPU if other hosts in the application are accessing the ECR or PSR registers.

Affected Silicon Revisions

A0	B1					
X	X					

4.3 Device

4.3.1 Detection of a Debugger Probe

The detection of a debugger probe or a user reset (external reset, watchdog reset and system reset request) could fail if the "BOD33 Disable" fuse is cleared (i.e., BOD33 is enabled).

Workaround

To secure the detection of debugger probes or a user reset, enable BOD33 using the SUPC.BOD33 register instead of the "BOD33 Disable" fuse. The "BOD33 Disable" fuse must be kept set.

© 2023 Microchip Technology Inc. and its subsidiaries

Affected Silicon Revisions

A0	B1					
X						

© 2023 Microchip Technology Inc. and its subsidiaries

5. Data Sheet Clarifications

The following typographic corrections and clarifications are to be noted for the device data sheet (DS60001715**F**), and are showed in **BOLD** type:

There are currently no Data Sheet Clarifications to report.

© 2023 Microchip Technology Inc. and its subsidiaries

6. Appendix A: Revision History

Rev F - 05/2023

The following errata were added in this revision:

- TCC: 2.10.2 MCx Interrupt Flags
- TC: 2.11.1 MCx Interrupt Flags
- FREQM: 2.12.1 Lost Interrupt
- FREQM: 2.12.2 STATUS.BUSY

Updated the following errata: 2.7.1. Tamper Detection Timestamp

Rev E - 02/2023

The following errata were added in this revision:

- Device: 2.3.2 Power Up

Rev D - 12/2022

The following errata were added in this revision:

- SERCOM USART: 2.9.4 LIN Host Delays
- SERCOM USART: 2.9.5 Two Stop Bits Mode in LIN Host

Updated the verbiage in the following errata:

- TCC: 2.10.1 Auto Lock Feature
- Device: 4.3.1 Detection of a Debugger Probe

Revision C - 08/2022

This errata was restructured in this revision to include the following new sections:

- Silicon Debug Related Errata Summary
- Silicon Debug Related Errata Issues

The following errata were added in this revision:

- 2.2.10 Transmit Cancellation
- 2.3.1 Standby Mode

This version of the document includes several major revisions. The following sections were reworked to remove deprecated errata content:

- Errata Summary
- AC
- CAN
- DEVICE
- DMAC
- NVMCTRL
- OSCCTRL
- RTC
- SERCOM I^2C
- SERCOM USART
- TCC

The following Errata sections were deprecated in this revision:

© 2023 Microchip Technology Inc. and its subsidiaries

- ADC
- CCL
- DAC
- DSU
- EVSYS
- PAC
- PDEC
- PORT
- SERCOM
- SERCOM SPI
- TCC
- SUPC
- TRNG
- EIC
- OSC32KCTRL

Revision B - 04/2022

The following Errata were added in this revision:

- Device: 2.6.4 AVDD Analog Cluster
- DSU: 2.8.2. Coresight
- DSU : 2.8.3. Debugger Illegal Accesses
- EVSYS: 2.9.3. Synchronous/Resynchronized Modes
- NVMCTRL: 2.10.2 SmartEEPROM Buffered Mode
- OSCCTRL: 2.11.10 False Clock Failure Detection
- OSCCTRL: 2.11.11 Clock Switch Back Feature Limitation
- TC: 2.21.3 PER Register
- TCC: 2.22.9 STATUS Register Access
- TCC: 2.22.10 DMA One-Shot Trigger Mode
- EIC: 2.24.1 Edge Detection
- EIC: 2.24.2 Asynchronous Edge Detection
- OSC32KCTRL: 2.25.1 False Clock Failure Detection
- OSC32KCTRL: 2.25.2 Clock switch Back Feature Limitation

Updated the verbiage in the following errata:

- OSCCTRL: 2.11.8 Low-Frequency Input Clock on FDPLLn

Revision A - 11/2021

This is the initial release of this document.

© 2023 Microchip Technology Inc. and its subsidiaries

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

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

- 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 and under normal conditions.
- There are dishonest and possibly illegal methods being used in attempts to breach the code protection features of the Microchip devices. We believe that these methods require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Attempts to breach these code protection features, most likely, cannot be accomplished without violating Microchip's intellectual property rights.
- Microchip is willing to work with any customer who is concerned about the integrity of its code.
- 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. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized

© 2023 Microchip Technology Inc. and its subsidiaries

access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Information contained in this publication is provided for the sole purpose of designing with and using Microchip products. Information regarding device applications and the like 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.

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.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AnyRate, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, chipKIT, chipKIT logo, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PackeTime, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AgileSwitch, APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, FlashTec, Hyper Speed Control, HyperLight Load, IntelliMOS, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet-Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, WinPath, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, Augmented Switching, BlueSky, BodyCom, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, Espresso T1S, EtherGREEN, IdealBridge, In-Circuit Serial Programming, ICSP, INICnet, Intelligent Paralleling, Inter-Chip Connectivity, JitterBlocker, maxCrypto, maxView, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, RTAX, RTG4, SAM-ICE, Serial Quad I/O, simpleMAP, SimpliPHY, SmartBuffer, SMART-I.S., storClad, SQI, SuperSwitcher, SuperSwitcher II, Switchtec, SynchroPHY, Total Endurance, TSHARC, USBCheck, VariSense, VectorBlox, VeriPHY, ViewSpan, WiperLock, XpressConnect, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

© 2023 Microchip Technology Inc. and its subsidiaries

The Adaptec logo, Frequency on Demand, Silicon Storage Technology, and Symmcom are registered trademarks of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 2023, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN: 978-1-6683-2355-7

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

© 2023 Microchip Technology Inc. and its subsidiaries

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office	**Australia - Sydney**	**India - Bangalore**	**Austria - Wels**
2355 West Chandler Blvd.	Tel: 61-2-9868-6733	Tel: 91-80-3090-4444	Tel: 43-7242-2244-39
Chandler, AZ 85224-6199	**China - Beijing**	**India - New Delhi**	Fax: 43-7242-2244-393
Tel: 480-792-7200	Tel: 86-10-8569-7000	Tel: 91-11-4160-8631	**Denmark - Copenhagen**
Fax: 480-792-7277	**China - Chengdu**	**India - Pune**	Tel: 45-4485-5910
Technical Support:	Tel: 86-28-8665-5511	Tel: 91-20-4121-0141	Fax: 45-4485-2829
www.microchip.com/support	**China - Chongqing**	**Japan - Osaka**	**Finland - Espoo**
Web Address:	Tel: 86-23-8980-9588	Tel: 81-6-6152-7160	Tel: 358-9-4520-820
www.microchip.com	**China - Dongguan**	**Japan - Tokyo**	**France - Paris**
Atlanta	Tel: 86-769-8702-9880	Tel: 81-3-6880- 3770	Tel: 33-1-69-53-63-20
Duluth, GA	**China - Guangzhou**	**Korea - Daegu**	Fax: 33-1-69-30-90-79
Tel: 678-957-9614	Tel: 86-20-8755-8029	Tel: 82-53-744-4301	**Germany - Garching**
Fax: 678-957-1455	**China - Hangzhou**	**Korea - Seoul**	Tel: 49-8931-9700
Austin, TX	Tel: 86-571-8792-8115	Tel: 82-2-554-7200	**Germany - Haan**
Tel: 512-257-3370	**China - Hong Kong SAR**	**Malaysia - Kuala Lumpur**	Tel: 49-2129-3766400
Boston	Tel: 852-2943-5100	Tel: 60-3-7651-7906	**Germany - Heilbronn**
Westborough, MA	**China - Nanjing**	**Malaysia - Penang**	Tel: 49-7131-72400
Tel: 774-760-0087	Tel: 86-25-8473-2460	Tel: 60-4-227-8870	**Germany - Karlsruhe**
Fax: 774-760-0088	**China - Qingdao**	**Philippines - Manila**	Tel: 49-721-625370
Chicago	Tel: 86-532-8502-7355	Tel: 63-2-634-9065	**Germany - Munich**
Itasca, IL	**China - Shanghai**	**Singapore**	Tel: 49-89-627-144-0
Tel: 630-285-0071	Tel: 86-21-3326-8000	Tel: 65-6334-8870	Fax: 49-89-627-144-44
Fax: 630-285-0075	**China - Shenyang**	**Taiwan - Hsin Chu**	**Germany - Rosenheim**
Dallas	Tel: 86-24-2334-2829	Tel: 886-3-577-8366	Tel: 49-8031-354-560
Addison, TX	**China - Shenzhen**	**Taiwan - Kaohsiung**	**Israel - Ra'anana**
Tel: 972-818-7423	Tel: 86-755-8864-2200	Tel: 886-7-213-7830	Tel: 972-9-744-7705
Fax: 972-818-2924	**China - Suzhou**	**Taiwan - Taipei**	**Italy - Milan**
Detroit	Tel: 86-186-6233-1526	Tel: 886-2-2508-8600	Tel: 39-0331-742611
Novi, MI	**China - Wuhan**	**Thailand - Bangkok**	Fax: 39-0331-466781
Tel: 248-848-4000	Tel: 86-27-5980-5300	Tel: 66-2-694-1351	**Italy - Padova**
Houston, TX	**China - Xian**	**Vietnam - Ho Chi Minh**	Tel: 39-049-7625286
Tel: 281-894-5983	Tel: 86-29-8833-7252	Tel: 84-28-5448-2100	**Netherlands - Drunen**
Indianapolis	**China - Xiamen**		Tel: 31-416-690399
Noblesville, IN	Tel: 86-592-2388138		Fax: 31-416-690340
Tel: 317-773-8323	**China - Zhuhai**		**Norway - Trondheim**
Fax: 317-773-5453	Tel: 86-756-3210040		Tel: 47-72884388
Tel: 317-536-2380			**Poland - Warsaw**
Los Angeles			Tel: 48-22-3325737
Mission Viejo, CA			**Romania - Bucharest**
Tel: 949-462-9523			Tel: 40-21-407-87-50
Fax: 949-462-9608			**Spain - Madrid**
Tel: 951-273-7800			Tel: 34-91-708-08-90
Raleigh, NC			Fax: 34-91-708-08-91
Tel: 919-844-7510			**Sweden - Gothenberg**
New York, NY			Tel: 46-31-704-60-40
Tel: 631-435-6000			**Sweden - Stockholm**
San Jose, CA			Tel: 46-8-5090-4654
Tel: 408-735-9110			**UK - Wokingham**
Tel: 408-436-4270			Tel: 44-118-921-5800
Canada - Toronto			Fax: 44-118-921-5820
Tel: 905-695-1980			
Fax: 905-695-2078			

© 2023 Microchip Technology Inc. and its subsidiaries