The PIC32MX575/675/695/775/795 family devices that you have received conform functionally to the current Device Data Sheet (DS60001156K), 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 the PIC32MX575/675/695/775/795 silicon.

Data Sheet clarifications and corrections start on page 18, 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 web site (www.microchip.com).

**TABLE 1: SILICON DEVREV VALUES**

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Device ID(1)</th>
<th>Revision ID for Silicon Revision(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>A0  A1  A3  A4  A5</td>
</tr>
<tr>
<td>PIC32MX575F256H</td>
<td>0x4317053</td>
<td>0x0  0x1  0x3  0x4  0x5</td>
</tr>
<tr>
<td>PIC32MX675F256H</td>
<td>0x430B053</td>
<td></td>
</tr>
<tr>
<td>PIC32MX775F256H</td>
<td>0x4303053</td>
<td></td>
</tr>
<tr>
<td>PIC32MX575F512H</td>
<td>0x4309053</td>
<td></td>
</tr>
<tr>
<td>PIC32MX675F512H</td>
<td>0x430C053</td>
<td></td>
</tr>
<tr>
<td>PIC32MX695F512H</td>
<td>0x4325053</td>
<td></td>
</tr>
<tr>
<td>PIC32MX775F512H</td>
<td>0x430D053</td>
<td></td>
</tr>
<tr>
<td>PIC32MX795F512H</td>
<td>0x430E053</td>
<td></td>
</tr>
<tr>
<td>PIC32MX575F256L</td>
<td>0x4333053</td>
<td></td>
</tr>
<tr>
<td>PIC32MX675F256L</td>
<td>0x4305053</td>
<td></td>
</tr>
</tbody>
</table>

**Note 1:** Refer to the “Memory Organization” and “Special Features” chapters in the current Device Data Sheet (DS60001156K) for detailed information on Device and Revision IDs for your specific device.
**TABLE 1: SILICON DEVREV VALUES (CONTINUED)**

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Device ID(1)</th>
<th>Revision ID for Silicon Revision(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIC32MX775F256L</td>
<td>0x4312053</td>
<td>A0 0x0 A1 0x1 A3 0x3 A4 0x4 A5 0x5</td>
</tr>
<tr>
<td>PIC32MX575F512L</td>
<td>0x430F053</td>
<td></td>
</tr>
<tr>
<td>PIC32MX675F512L</td>
<td>0x4311053</td>
<td></td>
</tr>
<tr>
<td>PIC32MX695F512L</td>
<td>0x4341053</td>
<td></td>
</tr>
<tr>
<td>PIC32MX775F512L</td>
<td>0x4307053</td>
<td></td>
</tr>
<tr>
<td>PIC32MX795F512L</td>
<td>0x4307053</td>
<td></td>
</tr>
</tbody>
</table>

*Note 1:* Refer to the “Memory Organization” and “Special Features” chapters in the current Device Data Sheet (DS60001156K) for detailed information on Device and Revision IDs for your specific device.

**TABLE 2: SILICON ISSUE SUMMARY**

<table>
<thead>
<tr>
<th>Module</th>
<th>Feature</th>
<th>Item #</th>
<th>Issue Summary</th>
<th>Affected Revisions(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>I2C</td>
<td></td>
<td>1.</td>
<td>The SDA line state may not be detected correctly.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>Ethernet</td>
<td>RMII 10 MB</td>
<td>2.</td>
<td>Pause frames are sent at 10 times the normal rate.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>ADC</td>
<td>Interrupt</td>
<td>3.</td>
<td>The interrupt generated by the module cannot be cleared when the module is disabled.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>Parallel</td>
<td>Slave Mode</td>
<td>4.</td>
<td>A PMP interrupt used to wake the device will not be reflected in the interrupt flag until the end of the write strobe.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>Master Port</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Output</td>
<td></td>
<td>5.</td>
<td>Output Compare Fault detection is not asynchronous.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>Compare</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPI</td>
<td></td>
<td>6.</td>
<td>The SPIBUSY and SRMT bits assert 1 bit time before the end of the transaction.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>UART</td>
<td></td>
<td>7.</td>
<td>The UTXBF bit deasserts one Peripheral Bus (PB) clock after the interrupt is generated.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>USB</td>
<td>USB PLL</td>
<td>8.</td>
<td>The USBPLL does not automatically suspend in Idle mode.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>Output</td>
<td></td>
<td>9.</td>
<td>In PWM mode, the output waveform is one PB clock longer than the expected value.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>Compare</td>
<td>PWM</td>
<td>10.</td>
<td>A Fault interrupt will not be generated if firmware clears the Fault while the Fault is still asserted.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>Output</td>
<td>PWM Fault Input</td>
<td>11.</td>
<td>In Pattern Match mode, the DMA module may not append all of the CRC results to the result buffer.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>Compare</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DMA</td>
<td>Pattern Match</td>
<td>12.</td>
<td>In Synchronized External Clock mode, the first period of the count is short.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>Timers</td>
<td>External Clock</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPI</td>
<td>Frame Slave</td>
<td>13.</td>
<td>Outgoing data corruption occurs when the frame signal is coincident with the clock.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>Mode</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CAN</td>
<td></td>
<td>14.</td>
<td>TXABAT, TXLARB, and TXERR bits may erroneously be cleared by an aborted read of the CxFIFOCONn register.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
<tr>
<td>CAN</td>
<td></td>
<td>15.</td>
<td>Requested aborts to a TX message via setting the ABAT (CxC&lt;CON&lt;27&gt;) bit or clearing the TXREQ (CxFIFOCON&lt;3&gt;) bit may not complete.</td>
<td>A0 X A1 X A3 X A4 X A5</td>
</tr>
</tbody>
</table>

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

*Note 2:* This issue has been corrected for this revision of silicon. Refer to the specific issue for details regarding the correction.
### TABLE 2: SILICON ISSUE SUMMARY (CONTINUED)

<table>
<thead>
<tr>
<th>Module</th>
<th>Feature</th>
<th>Item #</th>
<th>Issue Summary</th>
<th>Affected Revisions&lt;sup&gt;(1)&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAN</td>
<td>—</td>
<td>16</td>
<td>The FRESET (CxFIFOCONn&lt;14&gt;) and UINC (CxFIFOCONn&lt;13&gt;) bits are not settable via a normal SFR write.</td>
<td>A0 A1 A3 A4 A5</td>
</tr>
<tr>
<td>CAN</td>
<td>DeviceNet™</td>
<td>17</td>
<td>DeviceNet™ filtering does not function.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Output Compare</td>
<td>PWM Fault Input Mode</td>
<td>18</td>
<td>A Fault may be erroneously cleared due to an aborted read.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>SPI</td>
<td>Slave Mode</td>
<td>19</td>
<td>In Slave mode with the STXISEL (SPIxCON&lt;3:2&gt;) bits = 00, a TX buffer underrun condition will not assert the TX interrupt flag.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>USB</td>
<td>—</td>
<td>20</td>
<td>The TOKBUSY bit does not correctly indicate status when a transfer completes within the Start of Frame (SOF) threshold.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>USB</td>
<td>Host Mode</td>
<td>21</td>
<td>In Host mode, the interval between the first two SOF packets may be less than what is specified by the USB specification.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Watchdog Timer</td>
<td>—</td>
<td>22</td>
<td>When code-protect is enabled, the WDT is not held in Reset during the POR RAM Clear Sequence (RCS).</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Oscillator</td>
<td>Clock Switch and Two-Speed Start-Up</td>
<td>23</td>
<td>Clock switching and Two-Speed Start-up may cause a general exception when the reserved bit 8 of the DDPCON register is ‘0’.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Oscillator</td>
<td>Clock Switch</td>
<td>24</td>
<td>Clock source switching may cause a general exception or POR when switching from a slow clock to a fast clock.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>SPI</td>
<td>Slave Mode</td>
<td>25</td>
<td>A wake-up interrupt may not be clearable.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>PORTS</td>
<td>—</td>
<td>26</td>
<td>I/O pins do not tri-state immediately, if previously driven high.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>SPI</td>
<td>—</td>
<td>27</td>
<td>Byte writes to the SPIxSTAT register are not decoded correctly.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>SPI</td>
<td>Frame Mode</td>
<td>28</td>
<td>Recovery from an underrun requires multiple SPI clock periods.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>CAN</td>
<td>—</td>
<td>29</td>
<td>The TXABAT bit status may be incorrect after an abort.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>UART</td>
<td>IrDA®</td>
<td>30</td>
<td>The IrDA minimum bit time is not detected at all baud rates.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>UART</td>
<td>IrDA®</td>
<td>31</td>
<td>Transmit (TX) data is corrupted when BRG values greater than 0x200 are used.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>JTAG</td>
<td>—</td>
<td>32</td>
<td>On 64-pin devices, the TMS pin requires an external pull-up.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>UART</td>
<td>—</td>
<td>33</td>
<td>The TRMT (UxSTA&lt;8&gt;) bit is asserted before the transmission is complete.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>UART</td>
<td>UART Receive Buffer Overrun Error Status</td>
<td>34</td>
<td>The OERR (UxSTA&lt;1&gt;) bit does not get cleared on a module Reset. The OERR bit retains its value even after the UART module is reinitialized.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>ADC</td>
<td>Conversion Trigger from INTO Interrupt</td>
<td>35</td>
<td>The ADC module conversion triggers occur on the rising edge of the INTO signal even when INTO is configured to generate an interrupt on the falling edge.</td>
<td>X X X X X X</td>
</tr>
</tbody>
</table>

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

**Note 2:** This issue has been corrected for this revision of silicon. Refer to the specific issue for details regarding the correction.
TABLE 2: SILICON ISSUE SUMMARY (CONTINUED)

<table>
<thead>
<tr>
<th>Module</th>
<th>Feature</th>
<th>Item #</th>
<th>Issue Summary</th>
<th>Affected Revisions(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>JTAG</td>
<td>Boundary Scan</td>
<td>36.</td>
<td>Pin 100 on 100-pin packages and pin A1 on 121-pin packages do not respond to boundary scan commands.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>DMA</td>
<td>Suspend Status</td>
<td>37.</td>
<td>The DMABUSY status bit (DMACON&lt;11&gt;) may not reflect the correct status if the DMA module is suspended.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Voltage</td>
<td>Regulator BOR</td>
<td>38.</td>
<td>Device may not exit BOR state if a BOR event occurs.</td>
<td>X X</td>
</tr>
<tr>
<td>Output</td>
<td>Compare PWM Mode</td>
<td>39.</td>
<td>If the Output Compare module is configured for a 0% duty cycle (OCxRS = 0), a glitch may occur on the next cycle.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Oscillator</td>
<td>Clock Switch</td>
<td>40.</td>
<td>If a Fail-Safe Clock Monitor (FSCM) event occurs when Primary Oscillator (POsc) mode is used, firmware clock switch requests to switch from FRC mode will fail.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>i²C</td>
<td>Slave Mode</td>
<td>41.</td>
<td>The i²C module does not respond to address 0x78 when the STRICT and A10M bits are cleared in the i2CxCON register.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>USB</td>
<td>Idle Interrupt</td>
<td>42.</td>
<td>Idle interrupts cease if the IDLEIF interrupt flag is cleared.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Constant Data</td>
<td>Access from Flash</td>
<td>43.</td>
<td>A Data Bus Exception (DBE) may occur if an interrupt is encountered by the CPU while it is accessing constant data from Flash memory.</td>
<td>X X X See Note 2 See Note 2</td>
</tr>
<tr>
<td>CPU</td>
<td>Data Write to a Peripheral</td>
<td>44.</td>
<td>A data write operation by the CPU to a peripheral may be repeated if an interrupt occurs during initial write operation.</td>
<td>X X X See Note 2 See Note 2</td>
</tr>
<tr>
<td>Clock Out</td>
<td></td>
<td>45.</td>
<td>A clock signal is present on the CLKO pin, regardless of the clock source and setting of the CLKO Enable Configuration bit, during a Power-on Reset (POR) condition.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Input</td>
<td>Idle Mode and Sleep Mode</td>
<td>46.</td>
<td>All input capture modes selectable by the ICM&lt;2:0&gt; (ICxCON&lt;2:0&gt;) bits, with the exception of Interrupt-only mode, will not work when the CPU enters Idle mode or Sleep mode.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Host</td>
<td></td>
<td>47.</td>
<td>The USB bus might not be returned to the J-state following an acknowledgment packet when running low-speed through a hub.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Pull-ups</td>
<td></td>
<td>48.</td>
<td>Internal pull-up resistors may not guarantee a logical ‘1’ on non-5V tolerant pins when they are configured as digital inputs.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Pull-ups</td>
<td></td>
<td>49.</td>
<td>Internal pull-up resistors may not guarantee a logical ‘1’ on 5V tolerant pins when they are configured as digital inputs.</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Slave Addresses</td>
<td></td>
<td>50.</td>
<td>When the i²C module is operating as a Slave, some reserved bus addresses may be acknowledged (ACKed) when they should be not acknowledged (NAKed).</td>
<td>X X X X X X</td>
</tr>
<tr>
<td>Synchronization</td>
<td></td>
<td>51.</td>
<td>On a RX FIFO overflow, shift registers stop receiving data, which causes the UART to lose synchronization.</td>
<td>X X X X X X</td>
</tr>
</tbody>
</table>

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

Note 2: This issue has been corrected for this revision of silicon. Refer to the specific issue for details regarding the correction.
## TABLE 2: SILICON ISSUE SUMMARY (CONTINUED)

<table>
<thead>
<tr>
<th>Module</th>
<th>Feature</th>
<th>Item #</th>
<th>Issue Summary</th>
<th>Affected Revisions&lt;sup&gt;(1)&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A0  A1 A3 A4 A5</td>
</tr>
<tr>
<td>USB Low Speed Mode</td>
<td>Low-Speed Mode</td>
<td>52.</td>
<td>USB Low-Speed Device and Host modes are not supported.</td>
<td>X     X   X   X   X</td>
</tr>
</tbody>
</table>

**Note**
1: Only those issues indicated in the last column apply to the current silicon revision.
2: This issue has been corrected for this revision of silicon. Refer to the specific issue for details regarding the correction.
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 (A5).

1. Module: I2C

The I2C modules, with the exception of I2C1 and I2C2, may not detect state of SDA line correctly:

• In Master mode, module may encounter a bus collision when performing a Start condition.
• In Slave mode, module may not Acknowledge the first packet sent after enabling the I2C module. In this case, it will return a NACK instead of an ACK.

Work around
Master Mode:
1. Use another I2C node on the bus to sequence I2C bus transactions, such as the Start event.
2. Connect an unused general-purpose I/O pin to the SDAx pin of the I2C module to be used.
   
The user software must perform the following sequence of operations to execute a Start condition on the I2C bus:
   
a) With the I2C module disabled, clear the LAT bit of the general-purpose I/O pin that is connected to the SDAx pin. Then, clear the corresponding TRIS bit to make sure that the I/O pin is pulled low.
   
b) Enable the I2C module by setting the ON (I2CxCON<15>) bit; but do not configure the I2CxBRG register at this time.
   
c) Execute a software delay loop of at least 10 µs.
   
d) Set the TRIS bit of the I/O pin connected to the SDAx pin. This will make it an input pin, thereby ensuring that it goes to a high logic state.
   
e) Execute a software delay loop of at least 10 µs.
   
f) Configure the I2CxBRG register with the value required by the application.
   
g) Issue a Start condition by setting the SEN (I2CxCON<0>) bit as needed. I2C communications can now proceed normally.

Slave Mode:
The I2C master device on the bus must either pull the SDA line low, and then high again, prior to sending the first packet to the device, or must resend the first packet.

2. Module: Ethernet

In 10 MB RMII mode only, pause frames are sent at 10 times the normal rate. This reduces the available network bandwidth if the device is connected to the network via a hub. This does not reduce functionality or violate specifications.

Work around
If bandwidth is a concern, connect the PIC32 device to a network using an Ethernet switch.

3. Module: ADC

The interrupt generated by the ADC module cannot be cleared when the ADC module is disabled.

Work around
Ensure the interrupt is serviced and the interrupt flag is cleared before turning off the ADC module.

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 (A5).

A0 A1 A3 A4 A5
X X

A0 A1 A3 A4 A5
X X X X X

A0 A1 A3 A4 A5
X X X X X
4. **Module: Parallel Master Port**

   In Slave mode, a PMP interrupt will wake the device; however, the interrupt source will not be reflected in the interrupt flag until the end of the write strobe.

   **Work arounds**

   There are two possible solutions to this issue:

   1. If multiple wake-up sources are to be used, firmware can poll all of the configured wake-up source interrupt flags. If none are set, assume the source was the PMP.
   2. Firmware can wait for a period exceeding the write strobe length, and then poll the PMP interrupt flag.

   **Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

5. **Module: Output Compare**

   The Fault input detection is not asynchronous. There is a one to two Peripheral Bus (PB) clock delay between the Fault input assertion and the shutdown of the appropriate Output Compare output pin.

   **Work around**

   Ensure that the device driven by the Output Compare module can tolerate this shutdown delay.

   **Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

6. **Module: SPI**

   The SPIBUSY (SPIxCON<11>) and SRMT (SPIxCON<7>) bits assert one bit time before the end of the transaction.

   **Note:** SPI operation with the DMA module is not affected by this issue.

   **Work arounds**

   There are two possible solutions to this issue:

   1. Firmware must provide a one bit time delay between the assertion of these bits and performing any operation that requires the transaction to be complete.
   2. Use DMA module to transfer data to/from...
7. Module: UART

The UTXBF bit (UxSTA<9>) clears one PB clock cycle after the interrupt is generated. When using a PB bus divisor other than 1:1 and polling the UART transmit interrupt flag with the next instruction reading the UTXBF bit, the result may not reflect the actual UTXBF status.

Work arounds

There are two possible solutions to this issue:

1. Only use a PB bus divisor of 1:1.
2. If firmware is polling the transmit interrupt flag and the UTXBF flag, insert a read of the UxSTA register between these operations and discard the result. This read will ensure the status of the UTXBF flag is correct when the next read of this register occurs.

Affected Silicon Revisions

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

8. Module: USB

When the USBSIDL bit (UxCNFG1<4>) is set, the USBPLL does not automatically suspend in Idle mode.

Work around

Use firmware to manually suspend the USB clock before entering Sleep mode.

Affected Silicon Revisions

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

9. Module: Output Compare

In PWM mode, the output waveform is one Peripheral Bus (PB) clock longer than the expected value.

Work around

Load OCxRS with a value one less than the number expected to achieve the desired output.

Affected Silicon Revisions

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>
10. Module: Output Compare

In PWM mode, if firmware attempts to clear the OCFLT bit (OCxCON<4>) while the Fault still exists, a second interrupt will not be generated for this Fault when firmware exits the Interrupt Service Routine (ISR). The OCFLT bit will remain set while a Fault is detected.

**Work around**

In the ISR, clear the OCFLT bit, and test the OCFLT bit before exiting the ISR. If the bit is set, set the OCx interrupt to generate a second interrupt.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

11. Module: DMA

In Pattern Match mode, the DMA module may not append all of the CRC results to the result buffer.

**Work around**

Use firmware to read the CRC result and append it to the result buffer.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

12. Module: Timers

When the Timer module is first enabled and the prescaler value is greater than one, the number of input clocks required to increment the timer from zero to one is one input clock, not the value stated by the prescaler.

**Work around**

None.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

13. Module: SPI

Outgoing data will be corrupted when in Frame Slave mode with the FRMCNT<2:0> bits (SPIxCON<26:24>) greater than zero and the Frame pulse is coincident with the clock.

**Work around**

1. There is no work around for operation when the Frame pulse is coincident with the clock.
2. Provide a frame signal that precedes the clock signal.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

14. Module: CAN

The TXABAT, TXLARB, and TXERR bits may erroneously be cleared by an aborted read of the CxFIFOCONn register. An aborted read occurs when a load instruction in the CPU pipeline has started execution, but is aborted due to an interrupt.

**Work around**

Disable interrupts before reading the contents of the CxFIFOCONn register, and then re-enable interrupts after reading the register.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
15. Module: CAN

Requested aborts to a TX message via setting the ABAT bit (CxCON<27>) or clearing the TXREQ bit (CxFIFOCON<3>) may not complete. The CAN bus protocol is not violated.

**Work around**

1. After a general abort request, firmware should poll until the BUSY (CxCON<7>) bit = 0, or wait two message times. If the ABAT bit remains high, the message was successfully aborted and the module must be reset by clearing and setting the ON (CxCON<15>) bit.
2. After a FIFO specific abort request, firmware should poll until the BUSY bit = 0, or wait two message times. If the TXREQ bit remains high, the message was successfully aborted and the FIFO must be reset by setting the FRESET (CxFIFOCONn<14>) bit and polling until FRESET = 0.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

16. Module: CAN

The FRESET bit (CxFIFOCONn<14>) and the UINC bit (CxFIFOCONn<13>) are not settable through a normal Special Function Register (SFR) write.

**Work around**

Use the SET register operations to change the state of these bits.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

17. Module: CAN

The DeviceNet™ message filtering does not function.

**Work around**

Use hardware to filter the Standard Identifier (SID) and use firmware to decode the DeviceNet™ identifier.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

18. Module: Output Compare

The Output Compare module may reinitialize or clear a Fault on an aborted read of the OCxCON register. An aborted read occurs when a read instruction in the CPU pipeline has started execution, but is aborted due to an interrupt.

**Work around**

Disable interrupts before reading the contents of the OCxCON register, and then re-enable interrupts after reading the register.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

19. Module: SPI

In Slave mode with the STXISEL<1:0> (SPIxCON<3:2>) bits = 00, a TX buffer underrun condition will not assert the TX interrupt flag.

**Work around**

Use any other legal value of STXISEL<1:0> (i.e., ‘01’, ‘10’, or ‘11’).

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

20. Module: USB

The TOKBUSY bit (UxCON<5>) does not correctly indicate status when a transfer completes within the Start of Frame (SOF) threshold.

**Work around**

Use a firmware semaphore to track when a token is written to the UxTOK register. Firmware then clears the semaphore when the transfer is complete.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
21. Module: USB

In Host mode, the interval between the first two SOF packets may be less than what is specified by the USB specification.

**Work around**
None.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

22. Module: Watchdog Timer

When code-protect is enabled, the Watchdog Timer (WDT) is not held in Reset during the POR RAM Clear Sequence (RCS). If the WDT period does not exceed the RCS period, the WDT will reset the device and the RCS sequence will restart.

**Work around**
Use WDT periods equal to or longer than 128 ms. Since the RCS and WDT run concurrently, firmware will have a reduced period in which to service the WDT for the first time.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

23. Module: Oscillator

Clock switching and Two-Speed Start-up may cause a general exception when the reserved bit 8 of the DDPCON register is '0'.

**Work around**
Ensure that the reserved bit 8 of the DDPCON register to set to '1'. For example,

```c
DDPCON |= 0x100;
```

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

24. Module: Oscillator

Clock source switching may cause a general exception or POR when switching from a slow clock to a fast clock.

**Work around**
Clock source switches should be performed by first switching to the FRC, and then switching to the target clock source.

**Note:** If the peripheral library is being used, clock switching is performed automatically through the FRC.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

25. Module: SPI

In Slave mode, when entering Sleep mode after a SPI transfer with SPI interrupts enabled, a false interrupt may be generated that wakes the device. This interrupt can be cleared; however, entering Sleep may cause the condition to occur again.

**Work around**
Do not use SPI in Slave mode as a wake-up source from Sleep mode.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

26. Module: PORTS

When an I/O pin is set to output a logic high signal, and is then changed to an input using the TRISx registers, the I/O pin should immediately tri-state and let the pin float. Instead, the pin will continue to partially drive a logic high signal out for a period of time.

**Work around**
The pin should be driven low, prior to being tri-stated, if it is desirable for the pin to tri-state quickly.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
27. Module: SPI

Byte writes to the SPIxSTAT register are not decoded correctly. A byte write to byte zero of SPIxSTAT is actually performed on both byte zero and byte one. A byte write to byte one of SPIxSTAT is ignored.

Work around
Only perform word operations on the SPIxSTAT register.

Affected Silicon Revisions

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

28. Module: SPI

In Frame mode, the SPI module is not immediately ready for further transfers after clearing the SPITUR bit (SPIxSTAT<8>). The SPITUR bit will be cleared by hardware before the SPI state machine is prepared for the next operation.

Work around
Firmware must wait at least four bit times before writing to the SPI registers after clearing the SPITUR bit.

Affected Silicon Revisions

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

29. Module: CAN

When an abort request occurs concurrently with a successful message transmission, and additional messages remain in the FIFO, these remaining messages are not transmitted and the TXABAT bit (CxFIFOCONn<6>) does not reflect the abort.

Work around
The actual FIFO status can be determined by the FIFO pointers, CxFIFOCIn and CxFIFOUAn.

Affected Silicon Revisions

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

30. Module: UART

The UART module is not fully IrDA® compliant. The module does not detect the 1.6 μs minimum bit width at all baud rates as defined in the IrDA® specification. The module does detect the 3-/16-bit width at all baud rates.

Work around
None.

Affected Silicon Revisions

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

31. Module: UART

In IrDA® mode with baud clock output enabled, the UART transmit (TX) data is corrupted when the Baud Rate Generator (BRG) value is greater than 0x200.

Work around
Use the Peripheral Bus (PB) divisor to lower the PB frequency such that the required UART BRG value is less than 0x201.

Affected Silicon Revisions

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

32. Module: JTAG

On 64-pin devices an external pull-up resistor is required on the TMS pin for proper JTAG.

Work around
Connect a 100k to 200k pull-up to the TMS pin.

Affected Silicon Revisions

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
33. Module: UART

The TRMT bit (UxSTA<8>) is asserted during the Stop bit generation, not after the Stop bit has been sent.

**Work around**

If firmware needs to be aware when the transmission is complete, firmware should add a half bit time delay after the TRMT bit is asserted.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

34. Module: UART

The OERR (UxSTA<1>) bit does not get cleared on a module Reset. If the OERR bit is set and the module is disabled, the OERR bit retains its status even after the UART module is reinitialized.

**Work around**

The user software must check this bit in the UART module initialization routine and clear it if it is set.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

35. Module: ADC

When the ADC module is configured to start conversion on an external interrupt (SSRC<2:0> (ADxCON1<7:5> bits) = 001), the start of conversion always occurs on a rising edge detected at the INT0 pin, even when the INT0 pin has been configured to generate an interrupt on a falling edge (INT0EP (INTCON<0>) bit = 0).

**Work around**

Generate ADC conversion triggers on the rising edge of the INT0 signal.

Alternatively, use external circuitry to invert the signal appearing at the INT0 pin, so that a falling edge of the input signal is detected as a rising edge by the INT0 pin.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

36. Module: JTAG

Pin 100 on 100-pin packages and pin A1 on 121-pin packages do not respond to boundary scan commands.

**Work around**

None.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

37. Module: DMA

If the DMA module is suspended by setting the DMA Suspend bit, SUSPEND (DMACON<12>), the DMA Module Busy bit, DMABUSY (DMACON<11>), may continue to show a Busy status, when the DMA module completes transaction.

**Work around**

Use the Channel Busy bit, CHBUSY (DCHxCON<15>), to check the status of the DMA channel.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

38. Module: Voltage Regulator

Device may not exit BOR state if a BOR event occurs.

**Work arounds**

1. VDD must remain within the published specification (see parameter DC10 of the device data sheet).
2. Reset the device by providing a POR condition.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
39. Module: Output Compare

If the Output Compare module is configured for a 0% duty cycle (OCxRS register = 0), a glitch may occur on the next cycle.

**Work around**
The Output Compare module should be disabled and then re-enabled to achieve a 0% duty cycle.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

40. Module: Oscillator

If the Primary Oscillator (POsc) mode is implemented and a Fail-Safe Clock Monitor (FSCM) event occurs (failure of the external primary clock), the internal clock source will switch to the FRC oscillator. Subsequent firmware clock switch requests from the FRC oscillator to other clock sources will fail and the device will continue to execute on the FRC oscillator. Upon repair of the external clock source and a power-on state, the device will resume operation with the primary oscillator clock source.

**Work around**
None.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

41. Module: I2C

The slave address, 0x78, is one of a group of reserved addresses. It is used as the upper byte of a 10-bit address when 10-bit addressing is enabled. The I2C module control register allows the programmer to enable both 10-bit addressing and strict enforcement of reserved addressing, with the A10M (I2CxCON<10>) and STRICT (I2CxCON<11>) bits, respectively. When both bits are cleared, the device should respond to the reserved address 0x78, but does not.

**Work around**
None.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

42. Module: USB

If the bus has been idle for more than 3 ms, the IDLEIF interrupt flag is set. If software clears the interrupt flag, and the bus remains idle, the IDLEIF interrupt flag will not be set again.

**Work around**
Software can leave the IDLEIF bit set until it has received some indication of bus resumption. (Resume, Reset, SOF, or Error).

**Note:** Resume and Reset are the only interrupts that should be following IDLEIF assertion. If the IDLEIF bit is set, it should be okay to suspend the USB module. This will require software to clear the IDLEIE interrupt enable bit to exit the USB Interrupt Service Routine (ISR) (if using interrupt driven code).

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>
43. Module: CPU

When both Prefetch and Instruction Cache are enabled, a Data Bus Exception (DBE) may occur if an interrupt is encountered by the CPU while it is accessing constant data (not instructions) from Flash memory.

**Work arounds**

To avoid a DBE, use one of the following two solutions:

1. Structure application code, such that interrupts are not used while the CPU is accessing data from Flash memory.
2. Disable either the Prefetch module or CPU cache functionality as follows (by default both are disabled on a Power-on Reset (POR)):
   a) To disable the Prefetch module, set the Predictive Prefetch Enable bits, PRE-FEN<1:0>, in the Cache Control Register, CHECON<6:5>, to '00'.
   b) To disable CPU cache, set the Kseg0 bits, K0<2:0>, in the CP0 Configuration Register, Config<2:0>, to '010'.

**Note:** Disabling either the cache or Prefetch module will have minimum performance degradation, with a typical application realizing 10 percent or less performance impact.

**Corrected Revisions**

On corrected revisions, an interrupt occurring during CPU access of constant data (not instructions) from Flash memory will be delayed for up to two System Clock (SYSCLK) cycles.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

---

44. Module: CPU

During normal operation, if a CPU write operation to a peripheral is interrupted by an incoming interrupt, it should be aborted (not completed) and resumed after the interrupt is serviced. However, some of these write operations may not be aborted, resulting in a double write to peripherals by the CPU (the first write during the interrupt and the second write after the interrupt is serviced).

**Work around**

Most peripherals are not affected by this issue, as a double write will not have a negative impact. However, the following communication peripherals will double-send data if their respective transmit buffers are written twice: SPI, I2C, UART, and PMP. To avoid double transmission of data, utilize DMA to transfer data to these peripherals or disable interrupts while writing to these peripherals.

**Corrected Revisions**

On corrected revisions, an interrupt occurring during CPU write operation to a peripheral will be delayed for up to two Peripheral Bus Clock (PBCLK) cycles.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

---

45. Module: Oscillator

A clock signal is present on the CLKO pin, regardless of the clock source and setting of the CLKO Enable Configuration bit, OSCIOFNC (DEVCFG1<10>), during a Power-on Reset (POR) condition.

**Work around**

Do not connect the CLKO pin to a device that would be adversely affected by rapid pin toggling or a frequency other than that defined by the oscillator configuration. Do not use the CLKO pin as an input if the device connected to the CLKO pin would be adversely affected by the pin driving a signal out.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
46. Module: Input Capture

All input capture modes selectable by the ICM<2:0> (ICxCON<2:0>) bits, with the exception of Interrupt-only mode, will not work when the CPU enters Idle mode or Sleep mode.

**Work around**

Configure the Input Capture module for Interrupt-only mode (ICM<2:0> = 111) when the CPU is in Sleep mode or Idle mode.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

47. Module: USB

While operating in Host mode and attached to a low-speed device through a full-speed USB hub, the Host may persistently drive the bus to an SE0 state (both D+/D- as '0'), which would be interpreted as a bus Reset condition by the hub; or the host may persistently drive the bus to a J-state, which would make the hub detach condition undetectable by the host.

**Work around**

Connect low-speed devices directly to the Host USB port and not through a USB hub.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

48. Module: Non-5V Tolerant Pins

When internal pull-ups are enabled on non-5V tolerant pins, the level as measured on the pin and available to external device inputs, may not exceed the minimum value of $V_{IH}$, and therefore, qualify as a logic "high". However, with respect to PIC32 devices, so long as the load does not exceed -50 µA, the internal pull-ups are guaranteed to be recognized as a logic "high" internally to the PIC32 device.

**Work around**

It is recommend to only use external pull-ups:

- To guarantee a logic "high" for external logic input circuits outside of the PIC32 device
- For PIC32 device inputs, if the external load exceeds -50 µA

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

49. Module: 5V Tolerant Pins

When internal pull-ups are enabled on 5V tolerant pins, the level as measured on the pin and available to external device inputs, may not exceed the minimum value of $V_{IH}$, and therefore, qualify as a logic "high". However, with respect to PIC32 devices, so long as the load does not exceed -50 µA, the internal pull-ups are guaranteed to be recognized as a logic "high" internally to the PIC32 device.

**Work around**

It is recommend to only use external pull-ups:

- To guarantee a logic "high" for external logic input circuits outside of the PIC32 device
- For PIC32 device inputs, if the external load exceeds -50 µA

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

50. Module: I2C

When the I2C module is operating as a Slave, some reserved bus addresses may be acknowledged (ACKed) when they should be not acknowledged (NAKed).

As a result, there will be multiple data NAK interrupts until the Stop condition is asserted.

**Work around**

When the address interrupt arrives, check the address to determine if it is actually a reserved address. If the address is a reserved address, set a flag and use the flag to ignore subsequent data interrupts. When the Stop condition occurs, clear the flag.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
51. Module: UART

During a RX FIFO overflow condition, the shift register stops receiving data. This causes the UART to lose synchronization with the serial data stream. The only way to recover from this is to turn the UART OFF and ON until it synchronizes. This could require several OFF/ON sequences.

**Work arounds**

**Work around 1:**
Avoid the RX overrun condition by ensuring that the UARTx module has a high enough interrupt priority such that other peripheral interrupt processing latencies do not exceed the time to overrun the UART RX buffer based on the application baud rate. Alternately or in addition to, set the URXISEL bits in the UxSTA register to generate an earlier RX interrupt based on RX FIFO fill status to buy more time for interrupt latency processing requirements.

**Work around 2:**
If avoiding RX FIFO overruns is not possible, implement a ACK/NAK software handshake protocol to repeat lost packet transfers after restoring UART synchronization.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

52. Module: USB Low Speed Mode

USB Low-Speed mode is not functional in both Device and Host modes due to signal integrity compliance issues.

**Work around**
Use USB Full-Speed mode.

**Affected Silicon Revisions**

<table>
<thead>
<tr>
<th>A0</th>
<th>A1</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
Data Sheet Clarifications

The following typographic corrections and clarifications are to be noted for the latest version of the device data sheet (DS60001156K):

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

There are no current data sheet clarifications to report.
APPENDIX A: REVISION HISTORY

Initial release of this document; issued for revision A0 silicon.
Includes silicon issues 1 (I2C), 2 (Ethernet), 3 (ADC), 4 (Parallel Master Port), 5 (Output Compare), 6 (SPI) and 7 (UART).

Added silicon issues 8 (USB), 9-10 (Output Compare), 11 (DMA), 12 (Timers), 13 (SPI), 14-17 (CAN), 18 (Output Compare), 19 (SPI), 20-21 (USB), 22 (Watchdog Timer), 23 (Oscillator) and 24 (Oscillator).

The document title was changed to PIC32MX575/675/695/775/795 Family Silicon Errata and Data Sheet Clarification.
Added devices to Table 1: Silicon DEVREV Values.
Modified silicon issue 1 (I²C™).
Added silicon issues 25 (SPI), 26 (PORTS), 27-28 (SPI), 29 (CAN), 30-31 (UART), 32 (JTAG), 33 (UART) and 34 (UART), and added data sheet clarification issue 1 (There are no current data sheet clarifications to report.).

Removed data sheet clarification 1.
Added silicon issues 35 (ADC), 36 (JTAG) and 37 (DMA).

Added silicon issue 38 (Voltage Regulator).

Updated the current silicon revision to A1 throughout the document. Added silicon issue 39 (Output Compare) and data sheet clarification 1 (There are no current data sheet clarifications to report.).

Updated issue 19 (SPI).
Added silicon issues 40 (Oscillator), 41 (I²C), and 42 (USB).

Updated the current silicon revision to A3 throughout the document.

Added silicon issues 43 (CPU), 44 (CPU), and 45 (Oscillator).

Added silicon issue 46 (Input Capture) and 47 (USB).

Rev. L Document (9/2012)
Updated the current silicon revision to A4 throughout the document.
Updated silicon issue 6 (SPI), 43 (CPU) and 44 (CPU).
Updated the note in the Silicon DEVREV Values table (see Table 1).

Rev. M Document (2/2013)
Updated the current silicon revision to A5 throughout the document.
The Note in silicon issue 42 (USB) was updated.

Added silicon issues 48 (Non-5V Tolerant Pins) and 49 (5V Tolerant Pins).

Rev. O Document (10/2016)
Updated silicon issues 50 (I²C) and 51 (UART).

Rev. P Document (08/2019)
Added “There are no current data sheet clarifications to report.” under Data Sheet Clarifications.

Rev. Q Document (04/2020)
Added silicon issue 52. Module: “USB Low Speed Mode”.

Rev. R Document (04/2020)
Removed obsolete Data Sheet clarifications for Feature List.
Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip’s Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable.”

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

Information contained in this publication 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. 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 ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. 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, JukeBox, KeeLog, 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, SensiGenuity, SpyNIC, SSI, SSI Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TempTrackr, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

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, Vite, WinPath, and ZL are registered 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.

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.

© 2009-2020, Microchip Technology Incorporated, All Rights Reserved.

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

ISBN: 978-1-5224-5947-7

© 2009-2020 Microchip Technology Inc.
## AMERICAS

**Corporate Office**  
2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 480-792-7200  
Fax: 480-792-7277  
Technical Support:  
http://www.microchip.com/support  
Web Address:  
www.microchip.com

- **Atlanta**  
Duluth, GA  
Tel: 678-957-9614  
Fax: 678-957-1455
- **Austin, TX**  
Tel: 512-257-3370
- **Boston**  
Westborough, MA  
Tel: 774-760-0087  
Fax: 774-760-0088
- **Chicago**  
Itasca, IL  
Tel: 630-285-0071  
Fax: 630-285-0075
- **Dallas**  
Addison, TX  
Tel: 972-818-7243  
Fax: 972-818-2924
- **Detroit**  
Novi, MI  
Tel: 248-848-4000
- **Houston, TX**  
Tel: 281-884-5983
- **Indianapolis**  
Noblesville, IN  
Tel: 317-773-8323  
Fax: 317-773-5453  
Tel: 317-536-2380
- **Los Angeles**  
Mission Viejo, CA  
Tel: 949-462-9523  
Fax: 949-462-9608  
Tel: 951-273-7800
- **Raleigh, NC**  
Tel: 919-844-7510
- **New York, NY**  
Tel: 631-435-6000
- **San Jose, CA**  
Tel: 408-735-9110  
Tel: 408-436-4270
- **Canada - Toronto**  
Tel: 905-695-1980  
Fax: 905-695-2078

## ASIA/PACIFIC

- **Australia - Sydney**  
Tel: 61-2-9868-6733
- **China - Beijing**  
Tel: 86-10-23869-7000
- **China - Chengdu**  
Tel: 86-28-8665-5511
- **China - Chongqing**  
Tel: 86-23-8980-9588
- **China - Dongguan**  
Tel: 86-769-8702-9880
- **China - Guangzhou**  
Tel: 86-20-8755-8029
- **China - Hangzhou**  
Tel: 86-571-8792-8115
- **China - Hong Kong SAR**  
Tel: 852-2943-5100
- **China - Nanjing**  
Tel: 86-25-8473-2460
- **China - Qingdao**  
Tel: 86-632-8502-7355
- **China - Shanghai**  
Tel: 86-21-3326-8000
- **China - Shenyang**  
Tel: 86-24-2334-2829
- **China - Shenzhen**  
Tel: 86-755-8864-2200
- **China - Suzhou**  
Tel: 86-186-6233-1526
- **China - Wuhan**  
Tel: 86-27-5980-5300
- **China - Xian**  
Tel: 86-29-8833-7252
- **China - Xiamen**  
Tel: 86-592-2388138
- **China - Zhuhai**  
Tel: 86-756-3210040
- **India - Bangalore**  
Tel: 91-80-3090-4444
- **India - New Delhi**  
Tel: 91-11-4160-8631
- **India - Pune**  
Tel: 91-20-4121-0141
- **Japan - Osaka**  
Tel: 81-6-6152-7160
- **Japan - Tokyo**  
Tel: 81-3-6880-3770
- **Korea - Daegu**  
Tel: 82-53-744-4301
- **Korea - Seoul**  
Tel: 82-2-554-7200
- **Malaysia - Kuala Lumpur**  
Tel: 603-7651-7906
- **Malaysia - Penang**  
Tel: 60-4-227-8870
- **Philippines - Manila**  
Tel: 63-2-634-9065
- **Singapore**  
Tel: 65-6334-8870
- **Taiwan - Hsin Chu**  
Tel: 886-3-577-8366
- **Taiwan - Kaohsiung**  
Tel: 886-7-213-7830
- **Taiwan - Taipei**  
Tel: 886-2-2508-8600
- **Thailand - Bangkok**  
Tel: 66-2-694-1351
- **Vietnam - Ho Chi Minh**  
Tel: 84-28-5448-2100

## EUROPE

- **Austria - Wels**  
Tel: 43-7242-2244-39  
Fax: 43-7242-2244-393
- **Denmark - Copenhagen**  
Tel: 45-4485-5910  
Fax: 45-4485-2829
- **Finland - Espoo**  
Tel: 358-9-4520-820
- **France - Paris**  
Tel: 33-1-69-53-63-20  
Fax: 33-1-69-30-90-79
- **Germany - Garching**  
Tel: 49-8931-9700
- **Germany - Haan**  
Tel: 49-2129-3766400
- **Germany - Heilbronn**  
Tel: 49-7131-72400
- **Germany - Karlsruhe**  
Tel: 49-721-625370
- **Germany - Munich**  
Tel: 49-89-627-144-0  
Fax: 49-89-627-144-44
- **Germany - Rosenheim**  
Tel: 49-8031-354-560
- **Israel - Ra'anana**  
Tel: 972-9-744-7705
- **Italy - Milan**  
Tel: 39-0331-742611  
Fax: 39-0331-466781
- **Italy - Padova**  
Tel: 39-049-7625286
- **Netherlands - Drunen**  
Tel: 31-416-690399  
Fax: 31-416-690340
- **Norway - Trondheim**  
Tel: 47-7288-4388
- **Poland - Warsaw**  
Tel: 48-22-3325737
- **Romania - Bucharest**  
Tel: 40-21-407-87-50
- **Spain - Madrid**  
Tel: 34-91-708-08-90  
Fax: 34-91-708-08-91
- **Sweden - Gothenburg**  
Tel: 46-31-704-60-40
- **Sweden - Stockholm**  
Tel: 46-8-5090-4654
- **UK - Wokingham**  
Tel: 44-118-921-5800  
Fax: 44-118-921-5820