# PIC24HJXXXGPX06/X08/X10

## PIC24HJXXXGPX06/X08/X10 Rev. A2 Silicon Errata

The PIC24H (Rev. A2) devices you received were found to conform to the specifications and functionality described in the following documents:

- DS70175 – "*PIC24H Family Data Sheet*"
- DS70157 – "*dsPIC30F/33F Programmer's Reference Manual*"
- DS70046 – "*dsPIC30F Family Reference Manual*"

The exceptions to the specifications in the documents listed above are described in this section. The specific devices for which these exceptions are described are listed below:

- PIC24HJ64GP206
- PIC24HJ64GP210
- PIC24HJ64GP506
- PIC24HJ64GP510
- PIC24HJ128GP206
- PIC24HJ128GP210
- PIC24HJ128GP306
- PIC24HJ128GP310
- PIC24HJ128GP506
- PIC24HJ128GP510
- PIC24HJ256GP206
- PIC24HJ256GP210
- PIC24HJ256GP610

PIC24H Rev. A2 silicon is identified by performing a "Reset and Connect" operation to the device using MPLAB® ICD 2 with MPLAB IDE v7.40 or later. The output window will show a successful connection to the device specified in *Configure>Select Device*.

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

## Silicon Errata Summary

The following list summarizes the errata described in further detail through the remainder of this document:

1. Doze Mode

   When Doze mode is enabled, any writes to a peripheral SFR can cause other updates to that register to cease to function for the duration of the current CPU clock cycle.

2. 12-bit Analog-to-Digital Converter (ADC) Module

   For this revision of silicon, the 12-bit ADC module INL, DNL and signal acquisition time parameters are not within the published data sheet specifications.

3. 10-bit ADC Module

   For this revision of silicon, the 10-bit ADC module DNL, conversion speed and signal acquisition time parameters are not within the published data sheet specifications.

4. DMA Module: Interaction with EXCH Instruction

   The EXCH instruction does not execute correctly when one of the operands contains a value equal to the address of the DMAC SFRs.

5. DISI Instruction

   The DISI instruction will not disable interrupts if a DISI instruction is executed in the same instruction cycle that the DISI counter decrements to zero.

6. Output Compare Module

   The output compare module will produce a glitch on the output when an I/O pin is initially set high, and the module is configured to drive the pin low at a specified time.

7. Output Compare Module in PWM Mode

   The output compare module will miss one compare event when the duty cycle register value is updated from 0x0000 to 0x0001.

8. SPI Module in Frame Master Mode

   The SPI module will fail to generate frame synchronization pulses in Frame Master mode if FRMDLY = 1.

9. SPI Module in Slave Select Mode

   The SPI module Slave Select functionality will not work correctly.

10. SPI Module

    The SMP bit does not have any effect when the SPI module is configured for a 1:1 prescale factor in Master mode.

11. ECAN™ Module

    ECAN transmissions may be incorrect if multiple transmit buffers are simultaneously queued for transmission.

12. ECAN Module

    Under specific conditions, the first five bits of a transmitted identifier may not match the value in the transmit buffer ID register.

# PIC24HJXXXGPX06/X08/X10

13. ECAN Module Loopback Mode

    The ECAN module (ECAN1 or ECAN2) does not function correctly in Loopback mode.

14. I²C™ Module

    The Bus Collision Status bit does not get set when a bus collision occurs during a Restart or Stop event.

15. INT0, ADC and Sleep/Idle Mode

    ADC event triggers from the INT0 pin will not wake-up the device from Sleep or Idle mode if the SMPI bits are non-zero.

16. Doze Mode and Traps

    The address error trap, stack error trap, math error trap and DMA error trap will not wake-up a device from Doze mode.

17. JTAG Programming

    JTAG programming does not work.

18. UART

    With the parity option enabled, a parity error may occur if the Baud Rate Generator (BRG) contains an odd value.

19. UART

    The Receive Buffer Overrun Error Status bit may get set before the UART FIFO has overflowed.

20. UART

    UART receptions may be corrupted if the BRG is set up for 4x mode.

21. UART

    The UTXISEL0 bit is always read back as zero.

22. UART

    The auto-baud feature may not calculate the correct baud rate when the BRG is set up for 4x mode.

23. UART

    With the auto-baud feature selected, the sync break character (0x55) may be loaded into the FIFO as data.

24. ECAN Module

    Buffers 6 and 7 may intermittently transmit the wrong message type.

25. I²C Module

    A write collision does not prevent the transmit register from being written.

26. I²C Module

    The ACKSTAT bit only reflects the received ACK/NACK status for Master transmissions, but not for Slave transmissions.

27. I²C Module

    The D_A Status bit does not get set on a slave write to the transmit register.

28. Traps and Idle Mode

    If a clock failure occurs when the device is in Idle mode, the oscillator failure trap does not vector to the Trap Service Routine.

29. $\overline{\text{MCLR}}$ Wake-up from Sleep Mode

    An $\overline{\text{MCLR}}$ wake-up from Sleep mode does not wait for the on-chip voltage regulator to power up.

30. ECAN Module

    The C1RXOVF2 and C2RXOVF2 registers always read back as 0x0000.

31. FRC Oscillator

    Internal FRC accuracy parameters are not within the published data sheet specifications.

32. SPI

    SPI1 functionality for pin 34 (U1RX/SDI1/RF2) is erroneously enabled by the SPI2 module.

33. UART

    The auto-baud feature measures baud rate inaccurately for certain baud rate and clock speed combinations.

34. Device ID Register

    The content of the Device ID register changes from the factory programmed value.

35. DMA Module

    DMA data transfers that are active in Single-Shot mode while the device is in Sleep or Idle mode may result in more data transfers than expected.

36. Doze Mode and Traps

    A DMA error trap may not be generated when the device is in Doze mode.

37. DCI Module

    When using more than one transmit buffer, the DCI module will corrupt the data transmitted on the CSDO line.

The following sections describe the errata and workarounds to these errata, where they may apply.

### 1.   Module:  Oscillator: Doze Mode

Enabling Doze mode slows down the CPU but allows peripherals to run at full speed. When the CPU clock is slowed down by enabling Doze mode (CLKDIV<11> = 1), any writes to a peripheral SFR can cause other updates to that register to cease to function for the duration of the current CPU clock cycle. This is only an issue if the CPU attempts to write to the same register as a peripheral while in Doze mode.

For instance, if the ADC module is active and Doze mode is enabled, the main program should avoid writing to ADCCONx registers because these registers are being used by the ADC module. If the CPU does make writes before the ADC module does, then any attempts by the ADC module to write to these registers will fail.

#### <u>Work around</u>

In Doze mode, avoid writing code that will modify SFRs which may be written to by enabled peripherals.

# PIC24HJXXXGPX06/X08/X10

## 2. Module: 12-bit ADC

When the ADC module is configured for 12-bit operation, the specifications in the data sheets are not met.

**<u>Work around</u>**

Implement the ADC module as an 11-bit ADC with a maximum conversion rate of 300 Ksps.

1. The specifications in Table 1 reflect 11-bit ADC operation. RIN source impedance is recommended as 200 ohms and sample time is recommended as 3 $T_{AD}$ to ensure compatibility on future enhanced ADC modules. Missing codes are possible every $2^7$ codes.

2. When used as a 10-bit ADC, the INL is <±2 LSBs, and DNL is <±1 LSB with no missing codes. Maximum conversion rate is 300 Ksps.

**TABLE 1: ADC PERFORMANCE (11-BIT OPERATION)**

| Param No. | Symbol | Min | Typical | Max | Units | Conditions |
|-----------|--------|-----|---------|-----|-------|------------|
| AD17 | RIN | — | — | 200 | Ohm | 12-bit |
| ADC Accuracy – Measurements taken with External V$_{REF}$+/V$_{REF}$- | | | | | | |
| AD20a | Nr | — | 12 bits | — | Bits | |
| AD21a | INL | -2 | — | 2 | LSB | |
| AD22a | DNL | -1.5 | — | 1 | LSB | |
| AD23a | GERR | 1 | 5 | 10 | LSB | |
| AD24a | EOFF | 1 | 3 | 6 | LSB | |
| ADC Accuracy – Measurements taken with Internal V$_{REF}$+/V$_{REF}$- | | | | | | |
| AD21aa | INL | -2 | — | 2 | LSB | |
| AD22aa | DNL | -1.5 | — | 1 | LSB | |
| AD23aa | GERR | 5 | 10 | 20 | LSB | |
| AD24aa | EOFF | 3 | 6 | 15 | LSB | |
| Dynamic Performance | | | | | | |
| AD33a | FNYQ | — | — | 150 | KHz | |
| AD34a | ENOB | 9.5 | 9.6 | 10.4 | Bits | |
| ADC Conversion Rate | | | | | | |
| AD56a | FCNV | — | — | 300 | Ksps | |
| AD57a | TSAMP | — | 3 $T_{AD}$ | — | — | |

### 3. Module: 10-bit ADC

When the ADC module is configured for 10-bit operation, the specifications in the data sheet are not met for operation above 500 Ksps.

For 500 Ksps, the module meets specifications except for Gain and Offset parameters AD23bb and AD24bb.

For 600 Ksps operation, the module specifications are shown in Table 2.

#### Work around

None. Future versions of the silicon will support the ADC performance stated in the data sheet.

**TABLE 2:    600 KSPS OPERATION**

| Param No. | Symbol | Min | Typ | Max | Units | Conditions |
|---|---|---|---|---|---|---|
| AD17 | RIN | — | — | 200 | Ohm | 10-bit |
| ADC Accuracy – Measurements taken with External V$_{REF}$+/V$_{REF}$- | | | | | | |
| AD20b | Nr | — | 10 bits | — | Bits | |
| AD21b | INL | -2 | — | 2 | LSB | |
| AD22b | DNL | -1.5 | — | 2 | LSB | |
| AD23b | GERR | 1 | 3 | 6 | LSB | |
| AD24b | EOFF | 1 | 2 | 5 | LSB | |
| ADC Accuracy – Measurements taken with Internal V$_{REF}$+/V$_{REF}$- | | | | | | |
| AD21bb | INL | -2 | — | 2 | LSB | |
| AD22bb | DNL | -1.5 | — | 2 | LSB | |
| AD23bb | GERR | 1 | 6 | 12 | LSB | |
| AD24bb | EOFF | 2 | 5 | 10 | LSB | |
| Dynamic Performance | | | | | | |
| AD33b | FNYQ | — | — | 300 | KHz | |
| AD34b | ENOB | 8.5 | 9.7 | 9.8 | Bits | |
| ADC Conversion Rate | | | | | | |
| AD56b | FCNV | — | — | 600 | Ksps | |
| AD57b | TSAMP | — | 3 T$_{AD}$ | — | — | |

## 4. Module: DMA Module: Interaction with EXCH Instruction

The EXCH instruction does not execute correctly when either of the two operands is numerically equal to the address of any of the DMAC SFRs for this revision of silicon.

### Work around

If writing source code in assembly, the recommended fix is to replace:

```
EXCH Wsource, Wdestination
```

with:

```
PUSH Wdestination

MOV Wsource, Wdestination

POP Wsource
```

If using the MPLAB C30 C compiler, check the disassembly listing (*View>Disassembly Listing*) for the EXCH instruction. If used, make sure the operands are not equivalent to the DMA SFRs' addresses.

## 5. Module: DISI Instruction

When a user executes a DISI #7, for example, this will disable interrupts for 7 + 1 cycles (7 + the DISI instruction itself). In this case, the DISI instruction uses a counter which counts down from 7 to 0. The counter is loaded with 7 at the end of the DISI instruction.

If the user code executes another DISI on the instruction cycle where the DISI counter has become zero, the new DISI count is loaded, but the DISI state machine does not properly re-engage and continue to disable interrupts. At this point, all interrupts are enabled. The next time the user code executes a DISI instruction, the feature will act normally and block interrupts.

In summary, it is only when a DISI execution is coincident with the current DISI count = 0, that the issue occurs. Executing a DISI instruction before the DISI counter reaches zero will not produce this error. In this case, the DISI counter is loaded with the new value, and interrupts remain disabled until the counter becomes zero.

### Work around

When executing multiple DISI instructions within the source code, make sure that subsequent DISI instructions have at least one instruction cycle between the time that the DISI counter decrements to zero and the next DISI instruction. Alternatively, make sure that subsequent DISI instructions are called before the DISI counter decrements to zero.

## 6. Module: Output Compare Module

A glitch will be produced on an output compare pin under the following conditions:

• The user software initially drives the I/O pin high using the output compare module or a write to the associated PORT register.
• The output compare module is configured and enabled to drive the pin low at some later time (OCxCON = 0x0002 or OCxCON = 0x0003).

When these events occur, the output compare module will drive the pin low for one instruction cycle ($T_{CY}$) after the module is enabled.

### Work around

None. However, the user may use a timer interrupt and write to the associated PORT register to control the pin manually.

## 7. Module: Output Compare Module in PWM Mode

The output compare module will miss a compare event when the current duty cycle register (OCxRS) value is 0x0000 (0% duty cycle), and the OCxRS register is updated with a value of 0x0001. The compare event is only missed the first time a value of 0x0001 is written to OCxRS, and the PWM output remains low for one PWM period. Subsequent PWM high and low times occur as expected.

### Work around

None. If the current OCxRS register value is 0x0000, avoid writing a value of 0x0001 to OCxRS. Instead, write a value of 0x0002; however, in this case the duty cycle will be slightly different from the desired value.

**8. Module: SPI Module in Frame Master Mode**

The SPI module will fail to generate frame synchronization pulses when configured in the Frame Master mode if the start of data is selected to coincide with the start of the frame synchronization pulse (FRMEN = 1, SPIFSD = 0, FRMDLY = 1). However, the module functions correctly in Frame Slave mode, and also in Frame Maser mode if FRMDLY = 0.

**Work around**

If DMA is not being used, manually drive the $\overline{SSx}$ pin (x = 1 or 2) high using the associated PORT register, and then drive it low after the required 1 bit-time pulse width. This operation needs to be performed when the transmit buffer is written.

If DMA is being used, and if no other peripheral modules are using DMA transfers, use a Timer interrupt to periodically generate the frame synchronization pulse (using the method described above) after every 8 or 16 bit periods (depending on the data word size, configured using the MODE16 bit).

If FRMDLY = 0, no work around is needed.

**9. Module: SPI Module in Slave Select Mode**

The SPI module Slave Select functionality (enabled by setting $\overline{SSEN}$ = 1) will not function correctly. Whether the $\overline{SSx}$ pin (x = 1 or 2) is high or low, the SPI data transfer will be completed and an interrupt will be generated.

**Work around**

If DMA is not being used, manually poll the $\overline{SSx}$ pin state in the SPI interrupt by reading the associated LAT bit:

• If the LAT bit is '0', then perform the required data read/write.
• If the LAT bit is '1', then clear the SPI interrupt flag (SPIxIF), perform a dummy read of the SPIxBUF register and return from the Interrupt Service Routine.

If DMA is being used, there is no work around.

**10. Module: SPI Module**

The SMP bit (SPIxCON1<9>, where x = 1 or 2) does not have any effect when the SPI module is configured for a 1:1 prescale factor in Master mode. In this mode, whether the SMP bit is set or cleared, the data is always sampled at the end of data output time.

**Work around**

If sampling at the middle of data output time is required, then configure the SPI module to use a clock prescale factor other than 1:1 using the PPRE<1:0> and SPRE<2:0> bits in the SPIxCON1 register.

**11. Module: ECAN Module**

If multiple ECAN transmit buffers are queued for transmission (multiple TXREQ bits are set to '1' simultaneously), then the message transmissions from the enabled buffers may interfere with one another. As a result, incorrect ID and data transmissions will occur intermittently.

**Work around**

Enable only one transmit buffer for transmission at any given time. In the user application, this can be ensured by checking that all other TXREQn bits are clear, before setting the TXREQn bit corresponding to the buffer that is to be transmitted.

**12. Module: ECAN Module**

Under specific conditions, the first five bits of a transmitted identifier may not match the value in the transmit buffer SID. If the ECAN module detects a Start-of-Frame (SOF) in the third bit of interframe space and if a message to be transmitted is pending, the first five bits of the transmitted identifier may be corrupted.

**Work around**

None.

**13. Module: ECAN Module Loopback Mode**

The ECAN module (ECAN1 or ECAN2) does not function correctly in Loopback mode.

**Work around**

Do not use Loopback mode.

# PIC24HJXXXGPX06/X08/X10

## 14. Module: I²C Module

The Bus Collision Status bit (BCL) does not get set when a bus collision occurs during a Restart or Stop event. However, the BCL bit gets set when a bus collision occurs during a Start event.

**Work around**

None.

## 15. Module: INT0, ADC and Sleep/Idle Mode

ADC event triggers from the INT0 pin will not wake-up the device from Sleep or Idle mode if the SMPI bits are non-zero. This means that if the ADC is configured to generate an interrupt after a certain number of INT0 triggered conversions, the ADC conversions will not be triggered and the device will remain in Sleep. The ADC will perform conversions and wake-up the device only if it is configured to generate an interrupt after each INT0 triggered conversion (SMPI<3:0> = 0000).

**Work around**

None. If ADC event trigger from the INT0 pin is required, initialize SMPI<3:0> to '0000' (interrupt on every conversion).

## 16. Module: Doze Mode and Traps

The address error trap, stack error trap, math error trap and DMA error trap will not wake-up a device from Doze mode.

**Work around**

None.

## 17. Module: JTAG Programming

JTAG programming does not work.

**Work around**

None.

## 18. Module: UART

With the parity option enabled, a parity error, indicated by the PERR bit (UxSTA<3>) being set, may occur if the Baud Rate Generator contains an odd value. This affects both even and odd parity options.

**Work around**

Load the Baud Rate Generator register, UxBRG, with an even value, or disable the peripheral's parity option by loading either 0b00 or 0b11 into the Parity and Data Selection bits, PDSEL<1:0> (UxMODE<2:1>).

## 19. Module: UART

The Receive Buffer Overrun Error Status bit, OERR (UxSTA<1>), may get set before the UART FIFO has overflowed. After the fourth byte is received by the UART, the FIFO is full. The OERR bit should set after the fifth byte has been received in the UART shift register. Instead, the OERR bit may set after the fourth received byte with the UART Shift register empty.

**Work around**

After four bytes have been received by the UART, the UART Receiver Interrupt Flag bit, U1RXIF (IFS0<11>) or U2RXIF (IFS1<14>), will be set, indicating the UART FIFO is full. The OERR bit may also be set. After reading the UART receive buffer, UxRXREG, four times to clear the FIFO, clear both the OERR and UxRXIF bits in software.

## 20. Module: UART

UART receptions may be corrupted if the Baud Rate Generator is set up for 4x mode (BRGH = 1).

**Work around**

Use the 16x baud rate option (BRGH = 0) and adjust the baud rate accordingly.

## 21. Module: UART

The UTXISEL0 bit (UxSTA<13>) is always read as zero regardless of the value written to it. The bit can be written to either a '0' or '1', but will always be read as zero. This will affect read-modify-write operations such as bitwise or shift operations. Using a read-modify-write instruction on the UxSTA register (e.g., BSET, BLCR) will always write the UTXISEL0 bit to zero.

**Work around**

If a UTXISEL0 value of '1' is needed, avoid using read-modify-write instructions on the UxSTA register. Copy the UxSTA register to a temporary variable and set UxSTA<13> prior to performing read-modify-write operations. Copy the new value back to the UxSTA register.

## 22. Module: UART

The auto-baud feature may not calculate the correct baud rate when the High Baud Rate Enable bit, BRGH, is set. With the BRGH bit set, the baud rate calculation used is the same as BRG = 0.

**Work around**

If the auto-baud feature is needed, use the Low Baud Rate mode by clearing the BRGH bit.

## 23. Module: UART

With the auto-baud feature selected, the sync break character (0x55) may be loaded into the FIFO as data.

### Work around

To prevent the sync break character from being loaded into the FIFO, load the UxBRG register with either 0x0000 or 0xFFFF prior to enabling the auto-baud feature (ABAUD = 1).

## 24. Module: ECAN Module

When using the ECAN module with DMA, transmit buffers 6 and 7 may transmit the wrong message type. For example, the buffer may be configured as a standard frame but might intermittently transmit extended frames.

### Work around

None. Do not use buffers 6 or 7 for transmission. Instead, use transmit buffers 0, 1, 2, 3, 4 or 5.

## 25. Module: I²C

Writing to I2CxTRN during a Start bit transmission generates a write collision, indicated by the IWCOL (I2CxSTAT<7>) bit being set. In this state, additional writes to the I2CxTRN register should be blocked. However, in this condition, the I2CxTRN register can be written, although transmissions will not occur until the IWCOL bit is cleared in software.

### Work around

After each write to the I2CxTRN register, read the IWCOL bit to ensure a collision has not occurred. If the IWCOL bit is set, it must be cleared in software and I2CxTRN register must be rewritten.

## 26. Module: I²C

The ACKSTAT bit (I2CxSTAT<15>) only reflects the received ACK/NACK status for Master transmissions, but not for Slave transmissions. As a result, a Slave cannot use this bit to determine if it received an ACK or a NACK from a Master. In future silicon revisions, the ACKSTAT bit will reflect received ACK/NACK status for both Master and Slave transmissions.

### Work around

After transmitting a byte, the Slave should poll the SDA line (subject to a time out period dependent on the application) to determine if an ACK ('0') or a NACK ('1') was received.

## 27. Module: I²C

The D_A Status bit (I2CxSTAT<5>) gets set on a slave data reception in the I2CxRCV register, but does not get set on a slave write to the I2CxTRN register. In future silicon revisions, the D_A bit will get set on a slave write to the I2CxTRN register.

### Work around

Use the D_A Status bit only for determining slave reception status and not slave transmission status.

## 28. Module: Traps and Idle Mode

If a clock failure occurs when the device is in Idle mode, the oscillator failure trap does not vector to the Trap Service Routine. Instead, the device will simply wake up from Idle mode and continue code execution if the Fail-Safe Clock Monitor (FSCM) is enabled.

### Work around

Whenever the device wakes up from Idle (assuming the FSCM is enabled) the user software should check the state of the OSCFAIL bit (INTCON1<1>) to determine whether a clock failure occurred, and then perform the appropriate clock switch operation.

## 29. Module: MCLR Wake-up from Sleep Mode

If a MCLR reset pulse causes the device to wake up from Sleep mode, the device wakes up without waiting for the on-chip voltage regulator to power up. This will subsequently result in a Brown-Out Reset.

### Work around

None.

## 30. Module: ECAN Module

The C1RXOVF2 and C2RXOVF2 registers are non-functional. They are always read back as 0x0000, even when a receive overflow has occurred.

### Work around

None.

# PIC24HJXXXGPX06/X08/X10

## 31. Module: FRC Oscillator

The device does not meet the internal FRC accuracy specifications in the data sheet (Table 23-18 of the "*PIC24H Family Data Sheet*" (DS70175)). The actual accuracy specifications are shown in Table 3.

### Work around

None.

### TABLE 3:     INTERNAL FRC ACCURACY

| AC Characteristics | | Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C $\leq$ TA $\leq$ +85°C for industrial | | | | | |
|---|---|---|---|---|---|---|---|
| Parameter No. | Characteristic | Min | Typical | Max | Units | Conditions | |
| Internal FRC Accuracy @ FRC Frequency = 7.37 MHz[1,2] | | | | | | | |
| F20 | | -3 | — | +3 | % | -40°C $\leq$ TA $\leq$ +85°C | VDD = 3.0-3.6V |

**Note 1:** Frequency calibrated at 25°C and 3.3V. TUN bits can be used to compensate for temperature drift.

**2:** Devices set to initial frequency of 7.37 MHz (±2%) at 25°C.

## 32. Module: SPI

SPI1 functionality for pin 34 (U1RX/SDI1/RF2) is enabled by the SPI2 module. As a result, two side effects occur:

1. RF2 functionality is disabled if the SPI2 module is enabled.
2. This pin will not function as SDI1 if the SPI1 module is enabled.

This issue affects 64-pin devices only:

- PIC24HJ64GP206
- PIC24HJ128GP206
- PIC24HJ256GP206
- PIC24HJ128GP306
- PIC24HJ64GP506
- PIC24HJ128GP506

### Work around

Two conditions apply:

1. If the SPI2 module is used, pin 34 cannot be used as an I/O (RF2). It is recommended to use another I/O pin.
2. If the SPI1 module is used, the SPI2 module must also be enabled to gain SDI1 functionality on pin 34. As an alternative, I/O (RF2) can be configured as an input, which will allow pin 34 to function as SDI1.

## 33. Module: UART

The auto-baud feature may miscalculate for certain baud rate and clock speed combinations, resulting in a BRG value that is greater than or less than the expected value by 1. This may result in reception or transmission failures.

### Work around

Test the auto-baud rate at various clock speed and baud rate combinations that would be used in an application. If an inaccurate BRG value is generated, manually correct the baud rate in user software.

## 34. Module: Device ID Register

On a few devices, the content of the Device ID register can change from the factory programmed default value immediately after RTSP or ICSP™ Flash programming.

As a result, development tools will not recognize these devices and will generate an error message indicating that the device ID and the device part number do not match. Additionally, some peripherals will be reconfigured and will not function as described in the device data sheet.

Refer to **Section 5. "Flash Programming"** (DS70191), of the "*dsPIC33F Family Reference Manual*" for an explanation of RTSP and ICSP Flash programming.

### Work around

All RTSP and ICSP Flash programming routines must be modified as follows:

1. No word programming is allowed. Any word programming must be replaced with row programming.
2. During row programming, load write latches as described in **5.4.2.3 "Loading Write Latches"** of **Section 5. "Flash Programming"** (DS70191).
3. After latches are loaded, reload any latch location (in a given row) that has 5 LSB set to 0x18, with the original data. For example, reload one of the following latch locations with the desired data:
   0xXXXX18, 0xXXXX38, 0xXXXX58, 0xXXXX78, 0xXXXX98, 0xXXXXB8, 0xXXXXD8, 0xXXXXF8
4. Start row programming by setting NVMOP<3:0> = '0001' (memory row program operation) in the NVMCON register.
5. After row programming is complete, verify the contents of Flash memory.
6. If Flash verification errors are found, repeat steps 2 through 5. If Flash verification errors are found after a second iteration, report this problem to Microchip.

Steps 1 through 5 in the work around are implemented in MPLAB IDE version 7.61 for the MPLAB ICD 2, MPLAB REAL ICE™ in-circuit emulator and PM3 tools.

## 35. Module: DMA

When a DMA channel is enabled in Single-Shot mode while the device is in Idle mode, and the corresponding peripheral is active and configured to operate during Idle mode, the DMA channel may not become disabled immediately upon transferring the required amount of data.

As a result, the number of bytes or words of data transferred may exceed the DMA transfer count specified in the DMAxCNT register.

For example, if DMA transfers are active for both SPI byte transmissions and receptions, and only the receive DMA channel interrupt is enabled for waking up the device from Idle mode, an extra byte will be transmitted by the time the device wakes up from Idle mode.

### Work around

None.

## 36. Module: Doze Mode and Traps

A DMA error trap may not be generated when the device is in Doze mode.

### Work around

None.

## 37. Module: DCI

If the value of BLEN in DCICON2 is greater than '0', the DCI module allows the data in registers TXBUF1, TXBUF, and TXBUF3 to be overwritten while TXBUF0 is being transmitted. This results in the loss of the original contents of TXBUF1, TXBUF2, and TXBUF3. In addition, subsequent TXBUF1-3 register values will not be synchronized with TXBUF0.

### Work around

The application software must introduce a delay at the start of the DCI Interrupt Service Routine (ISR). This delay must be long enough for the DCI module to complete transmission of TXBUF0. New values can then be written to all of the transmit registers.

## APPENDIX A:    REVISION HISTORY

Revision A (6/2006)

• Initial release of the document.

Revision B (12/2006)

• Added issues 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 and 17.

Revision C (3/2007)

• Changed document title, updated issue 8, and added issues 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30 and 31.

Revision D (6/2007)

• Added the following silicon issues: 32 (SPI), 33 (UART), 34 (Device ID Register), 35 (DMA), 36 (Doze Mode and Traps) and 37 (DCI).

# WORLDWIDE SALES AND SERVICE

## AMERICAS

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

**Atlanta**
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

**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: 972-818-7423
Fax: 972-818-2924

**Detroit**
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

**Kokomo**
Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

**Los Angeles**
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

**Santa Clara**
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

**Toronto**
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax:  905-673-6509

## ASIA/PACIFIC

**Asia Pacific Office**
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Habour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

**Australia - Sydney**
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

**China - Beijing**
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

**China - Chengdu**
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

**China - Fuzhou**
Tel: 86-591-8750-3506
Fax: 86-591-8750-3521

**China - Hong Kong SAR**
Tel: 852-2401-1200
Fax: 852-2401-3431

**China - Qingdao**
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

**China - Shanghai**
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

**China - Shenyang**
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

**China - Shenzhen**
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

**China - Shunde**
Tel: 86-757-2839-5507
Fax: 86-757-2839-5571

**China - Wuhan**
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

**China - Xian**
Tel: 86-29-8833-7250
Fax: 86-29-8833-7256

## ASIA/PACIFIC

**India - Bangalore**
Tel: 91-80-4182-8400
Fax: 91-80-4182-8422

**India - New Delhi**
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

**India - Pune**
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

**Japan - Yokohama**
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

**Korea - Gumi**
Tel: 82-54-473-4301
Fax: 82-54-473-4302

**Korea - Seoul**
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

**Malaysia - Penang**
Tel: 60-4-646-8870
Fax: 60-4-646-5086

**Philippines - Manila**
Tel: 63-2-634-9065
Fax: 63-2-634-9069

**Singapore**
Tel:  65-6334-8870
Fax: 65-6334-8850

**Taiwan - Hsin Chu**
Tel: 886-3-572-9526
Fax: 886-3-572-6459

**Taiwan - Kaohsiung**
Tel: 886-7-536-4818
Fax: 886-7-536-4803

**Taiwan - Taipei**
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

**Thailand - Bangkok**
Tel: 66-2-694-1351
Fax: 66-2-694-1350

## EUROPE

**Austria - Wels**
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

**Denmark - Copenhagen**
Tel: 45-4450-2828
Fax: 45-4485-2829

**France - Paris**
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

**Germany - Munich**
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

**Italy - Milan**
Tel: 39-0331-742611
Fax: 39-0331-466781

**Netherlands - Drunen**
Tel: 31-416-690399
Fax: 31-416-690340

**Spain - Madrid**
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

**UK - Wokingham**
Tel: 44-118-921-5869
Fax: 44-118-921-5820

12/08/06