dsPIC33FJXXXMCX06/X08/X10 Family Silicon Errata and Data Sheet Clarification

The dsPIC33FJXXXMCX06/X08/X10 family devices that you have received conform functionally to the current Device Data Sheet (DS70287**C**), except for the anomalies described in this document.

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

The errata described in this document will be addressed in future revisions of the dsPIC33FJXXXMCX06/X08/X10 silicon.

> **Note:** This document summarizes all silicon errata issues from all revisions of silicon, previous as well as current. Only the issues indicated in the last column of Table 2 apply to the current silicon revision (**A4**).

Data sheet clarifications and corrections start on page 28, 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).

For example, to identify the silicon revision level using MPLAB IDE in conjunction with MPLAB ICD 2 or PICkit™ 3:

1. Using the appropriate interface, connect the device to the MPLAB ICD 2 programmer/debugger or PICkit 3.
2. From the main menu in MPLAB IDE, select *Configure>Select Device*, and then select the target part number in the dialog box.
3. Select the MPLAB hardware tool (*Debugger>Select Tool*).
4. Perform a "Connect" operation to the device (*Debugger>Connect*). Depending on the development tool used, the part number *and* Device Revision ID value appear in the **Output** window.

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

The Device and Revision ID values for the various dsPIC33FJXXXMCX06/X08/X10 silicon revisions are shown in Table 1.

TABLE 1: SILICON DEVREV VALUES

Part Number	Device ID[(1)]	Revision ID for Silicon Revision[(2)]			Part Number	Device ID[(1)]	Revision ID for Silicon Revision[(2)]		
		A2	A3	A4			A2	A3	A4
dsPIC33FJ64MC506	0x0089	0x3002	0x3004	0x3040	dsPIC33FJ128MC510	0x00A3	0x3002	0x3004	0x3040
dsPIC33FJ64MC508	0x008A				dsPIC33FJ128MC706	0x00A9			
dsPIC33FJ64MC510	0x008B				dsPIC33FJ128MC708	0x00AE			
dsPIC33FJ64MC706	0x0091				dsPIC33FJ128MC710	0x00AF			
dsPIC33FJ64MC710	0x0097				dsPIC33FJ256MC510	0x00B7			
dsPIC33FJ128MC506	0x00A1				dsPIC33FJ256MC710	0x00BF			

Note 1: The Device and Revision IDs (DEVID and DEVREV) are located at the last two implemented addresses in program memory.

2: Refer to the *"dsPIC33F/PIC24H Flash Programming Specification"* (DS70152) for detailed information on Device and Revision IDs for your specific device.

TABLE 2: SILICON ISSUE SUMMARY

Module	Feature	Item Number	Issue Summary	Affected Revisions(1)		
				A2	A3	A4
Doze Mode	—	1.	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.	X	X	X
ADC	12-bit Mode	2.	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.	X	X	X
ADC	10-bit Mode	3.	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.	X	X	X
CPU	EXCH Instruction	4.	The EXCH instruction does not execute correctly.	X	X	X
CPU	DISI Instruction	5.	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.	X	X	X
Output Compare	PWM Mode	6.	The output compare module will miss one compare event when the duty cycle register value is updated from 0x0000 to 0x0001.	X	X	X
SPI	Frame Master Mode	7.	The SPI module will fail to generate frame synchronization pulses in Frame Master mode.	X	X	X
SPI	Slave Select	8.	The SPI module slave select functionality will not work correctly.	X	X	X
SPI	Sampling	9.	The SMP bit does not have any effect when the SPI module is configured for a 1:1 prescale factor in Master mode.	X	X	X
ECAN™	Data Transmission	10.	ECAN transmissions may be incorrect if any buffers other than Buffer 0 are enabled as transmit buffers.	X	X	X
ECAN	Data Transmission	11.	Under specific conditions, the first five bits of a transmitted identifier may not match the value in the transmit buffer ID register.	X	X	X
ECAN	Loopback Mode	12.	The ECAN module (ECAN1 or ECAN2) does not function correctly in Loopback mode.	X	X	X
I^2C™	Bus Collision	13.	The Bus Collision Status bit does not get set when a bus collision occurs during a Restart or Stop event.	X	X	X
ADC	Sleep Mode	14.	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.	X	X	X
Doze Mode	—	15.	The address error trap, stack error trap, math error trap, and DMA error trap will not wake-up a device from Doze mode.	X	X	X
JTAG	Flash Programming	16.	JTAG programming does not work.	X	X	X
UART	—	17.	With the parity option enabled, a parity error may occur if the Baud Rate Generator (BRG) contains an odd value.	X	X	X
UART	—	18.	The Receive Buffer Overrun Error Status bit may get set before the UART FIFO has overflowed.	X	X	X
UART	High-Speed Mode	19.	UART receptions may be corrupted if the BRG is set up for 4x mode.	X	X	X
UART	—	20.	The UTXISEL0 bit is always read back as zero.	X	X	X
UART	High-Speed Mode	21.	The auto-baud feature may not calculate the correct baud rate when the BRG is set up for 4x mode.	X	X	X

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

TABLE 2: SILICON ISSUE SUMMARY (CONTINUED)

Module	Feature	Item Number	Issue Summary	Affected Revisions[(1)]		
				A2	A3	A4
UART	Auto-Baud	22.	With the auto-baud feature selected, the Sync Break character (0x55) may be loaded into the FIFO as data.	X	X	X
I^{2}C	—	23.	A write collision does not prevent the transmit register from being written.	X	X	X
I^{2}C	Slave Mode	24.	The ACKSTAT bit reflects the received ACK/NACK status for master transmissions, but not for slave transmissions.	X	X	X
I^{2}C	Slave Mode	25.	The D_A Status bit is not set on a slave write to the transmit register.	X	X	X
Interrupt Controller	Idle Mode	26.	If a clock failure occurs when the device is in Idle mode, the oscillator failure trap does not vector to the Trap Service Routine (TSR).	X	X	X
Internal Voltage Regulator	Sleep Mode	27.	An $\overline{\text{MCLR}}$ wake-up from Sleep mode does not wait for the on-chip voltage regulator to power-up.	X	X	X
ECAN	Receive Overflow	28.	The C1RXOVF2 and C2RXOVF2 registers always read back as 0x0000.	X	X	X
Oscillator	FRC Accuracy	29.	Internal FRC accuracy parameters are not within the published data sheet specifications.	X	X	X
SPI	SDI1 Pin	30.	SPI1 functionality for pin 34 (U1RX/SDI1/RF2) is erroneously enabled by the SPI2 module.	X	X	X
UART	Auto-Baud	31.	The auto-baud feature measures baud rate inaccurately for certain baud rate and clock speed combinations.	X	X	X
Device ID Register	—	32.	The content of the Device ID register changes from the factory programmed value.	X	X	X
DMA	Sleep and Idle Modes	33.	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.	X	X	X
DMA	Doze Mode	34.	A DMA error trap may not be generated when the device is in Doze mode.	X	X	X
Output Compare	Dual Compare Match Mode	35.	In Dual Compare Match mode, the OCx output is not reset when the OCxR and OCxRS registers are loaded with values that have a difference of 1.	X	X	X
UART	High-Speed Mode	36.	When the UART is in 4x mode (BRGH = 1) and using two Stop bits (STSEL = 1), it may sample the first Stop bit instead of the second one.	X	X	X
UART	Auto-Baud	37.	When an auto-baud is detected, the receive interrupt may occur twice.	X	X	X
DMA	NULL Data Write Mode	38.	NULL Data Peripheral Write mode for the DMA channel does not function.	X	X	X
DMA	Error Traps	39.	DMA request Fault condition does not generate a DMA error trap.	X	X	X
DMA	NULL Data Write Mode	40.	DMA channel writes an additional NULL value to the peripheral register.	X	X	X
CPU	REPEAT Instruction	41.	Any instruction executed inside a REPEAT loop that produces a Read-After-Write stall condition, results in the instruction being executed fewer times than was intended.	X	X	X

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

TABLE 2: SILICON ISSUE SUMMARY (CONTINUED)

Module	Feature	Item Number	Issue Summary	Affected Revisions[1]		
				A2	A3	A4
Oscillator	FRC Tuning	42.	For certain values of the TUN<5:0> bits (OSCTUN<5:0>), the resultant frequencies are incorrect.	X	X	X
UART	IR Mode	43.	The 16x baud clock signal on the BCLK pin is present only when the module is transmitting.	X	X	X
SPI	SCKx Pins	44.	The SPIxCON1 DISSCK bit does not influence port functionality.	X	X	X
I^2C	SFR Writes	45.	The BCL bit in I2CSTAT can be cleared only with 16-bit operation and can be corrupted with 1-bit or 8-bit operations on I2CSTAT.	X	X	X
I^2C	10-bit Addressing Mode	46.	When the I^2C module is configured for 10-bit addressing using the same address bits (A10 and A9) as other I^2C devices, the A10 and A9 bits may not work as expected.	X	X	X
I^2C	10-bit Addressing Mode	47.	When the I^2C module is configured as a 10-bit slave with an address of 0x102, the I2CxRCV register content for the lower address byte is 0x01 rather than 0x02.	X	X	X
I^2C	I/O Port Functionality	48.	With the I^2C module enabled, the PORT bits and external Interrupt Input functions (if any) associated with SCL and SDA pins will not reflect the actual digital logic levels on the pins.	X	X	X
I^2C	10-bit Addressing Mode	49.	The 10-bit slave does not set the RBF flag or load the I2CxRCV register on address match if the Least Significant bits of the address are the same as the 7-bit reserved addresses.	X	X	X
Internal Voltage Regulator	IPD Current	50.	When the VREGS bit (RCON<8>) is set to a logic '0', the device may reset and higher sleep current may be observed.	X	X	X
PSV Operations	—	51.	An address error trap occurs in certain addressing modes when accessing the first four bytes of any PSV page.	X	X	X
UART	Interrupts	52.	The UART error interrupt may not occur, or may occur at an incorrect time, if multiple errors occur during a short period of time.	X	X	X
UART	IR Mode	53.	When the UART module is operating in 8-bit mode (PDSEL = 0x) and using the IrDA® encoder/decoder (IREN = 1), the module incorrectly transmits a data payload of 80h as 00h.	X	X	X
ECAN	Sleep Mode	54.	The WAKIF bit in the CxINTF register cannot be cleared by software instruction after the device has been interrupted from Sleep by activity on the CAN bus.	X	X	X
I^2C	—	55.	After the ACKSTAT bit is set when receiving a NACK, it may be cleared by the reception of a Start or Stop bit.	X	X	X
PWM	Single-Shot Mode	56.	There is a glitch in the PWMxL signal in Single-Shot mode with complementary output. Another glitch occurs when resuming from a Fault condition in Free-Running mode with complementary output.	X	X	X
PWM	Latched Fault	57.	Subsequent faults in the same timer cycle are missed during a latched fault.	X	X	X
PWM	PWM Generator 2	58.	Under certain conditions, the PWM outputs, PWM2H and PWM2L, do not generate PWM signals.	X	X	X
PWM	PWM Counter Register	59.	PTMR does keep counting down after halting code execution in Debug mode.	X	X	X

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

TABLE 2: SILICON ISSUE SUMMARY (CONTINUED)

Module	Feature	Item Number	Issue Summary	Affected Revisions(1)		
				A2	A3	A4
PWM	Idle Mode	60.	Fault-driven Wake-up from Idle mode does not function.	X	X	X
PWM	Doze Mode	61.	The Motor Control PWM module generates more interrupts than expected when Doze mode is used and the output postscaler value is different than 1:1.	X	X	X
QEI	Interrupt Generator	62.	The QEI module does not generate an interrupt in a particular overflow condition.	X	X	X
SPI	Transmit Operation	63.	Writing to the SPIxBUF register as soon as TBF bit is cleared will cause the SPI module to ignore written data.	X	X	X
UART	Break Character Generation	64.	The UART module will not generate back-to-back Break characters.	X	X	X
QEI	Timer Gated Accumulation Mode	65.	When Timer Gated Accumulation is enabled, the QEI does not generate an interrupt on every falling edge.	X	X	X
QEI	Timer Gated Accumulation Mode	66.	When Timer Gated Accumulation is enabled, and an external signal is applied, the POSCNT increments and generates an interrupt after a match with MAXCNT.	X	X	X
I/O	SDO1 Pin	67.	The SDO1 pin may toggle while the device is being programmed via PGECx/PGEDx pin pairs.	X	X	X
SPI	Slave FRMDLY	68.	The SPI communication in Framed mode does not function correctly if the Slave SPI frame delay bit (FRMDLY) is set to '1'.	X	X	X
ADC	Current Consumption in Sleep Mode	69.	If the ADC module is in an enabled state when the device enters Sleep Mode, the power-down current (IPD) of the device may exceed the device data sheet specifications.	X	X	X
CPU	DISI	70.	The DISI disable interrupt duration may be extended based on the instruction coding sequence following a DISI instruction.	X	X	X

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

Silicon Errata Issues

> **Note:** This document summarizes all silicon errata issues from all revisions of silicon, previous as well as current. Only the issues indicated by the shaded column in the following tables apply to the current silicon revision (**A4**).

1. Module: 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.

Work around

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

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

2. Module: ADC

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

Work around

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

1. The specifications in Table 3 reflect 11-bit ADC operation. RIN source impedance is recommended as 200 ohms and sample time is recommended as 3 TAD 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 Least Significant Bytes (LSBs), and DNL is <±1 LSB with no missing codes. Maximum conversion rate is 300 ksps.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

TABLE 3: 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 VREF+/VREF-						
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 VREF+/VREF-						
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 TAD	—	—	—

3. Module: 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 4.

Work around

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

TABLE 4: 600 KSPS OPERATION

Param No.	Symbol	Min.	Typical	Max.	Units	Conditions
AD17	RIN	—	—	200	Ohm	10-bit
ADC Accuracy – Measurements taken with External VREF+/VREF-						
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 VREF+/VREF-						
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 TAD	—	—	—

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

4. Module: CPU

The EXCH instruction does not execute correctly.

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 C Compiler for dsPIC® DSCs (formerly known as the MPLAB C30 C Compiler), specify the compiler option, -merrata=exch (*Project>Build Options>Projects>MPLAB C30>Use Alternate Settings*)

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

5. Module: CPU

The DISI instruction will not disable interrupts when a DISI instruction is executed in the same instruction cycle that the DISI counter decrements to zero. For example, when user code executes a DISI #7, interrupts for 7 + 1 cycles (7 + the DISI instruction itself) are disabled. In that case, the DISI instruction uses a counter that 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 – the DISI counter is loaded with the new value, and interrupts remain disabled until the counter becomes zero.

Work around

When multiple DISI instructions are executed in 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.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

6. Module: Output Compare

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.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

7. Module: SPI

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`). Synchronization pulses also will not be generated if FRMDLY = `0` and SMP = `0`. However, the module functions correctly in Frame Slave mode, and also when FRMDLY = `0` and SMP = `1`.

Work around

If DMA is not being used, manually drive the $\overline{\text{SSx}}$ 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 period (depending on the data word size, which is configured using the MODE 16-bit).

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

8. Module: SPI

The SPI module slave select functionality (enabled by setting SSEN = `1`) will not function correctly. Whether the $\overline{\text{SSx}}$ 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, poll the $\overline{\text{SSx}}$ pin state using the Change Notification (CN) pin associated to the $\overline{\text{SSx}}$ pin as follows:

1. Disable the SPIx module by clearing the SPIEN bit in the SPIxSTAT register.
2. Clear the SSEN bit in the SPIxCON1 register to allow the I/O port to control the $\overline{\text{SSx}}$ pin.
3. Ensure that the CNx pin is configured as a digital input by setting the associated bit in the TRISx register.
4. Enable interrupts for the selected CNx pin by setting the appropriate bits in the CNEN1 and CNEN2 registers.
5. Turn on the weak pull-up device for the selected CNx pins by setting the appropriate bits in the CNPU1 and CNPU2 registers.
6. Clear the CNIF interrupt flag in the IFSx register.
7. Select the desired interrupt priority for CNx interrupts using the CNIP<2:0> control bits in the IPCx register.
8. Enable CNx interrupts using the CNIE control bit in the IECx register.
9. In the CNx Interrupt Service Routine, read the PORTx register associated to the $\overline{\text{SSx}}$ pin:
    a) if the PORTx bit is '`0`' – enable the SPIx module by setting the SPIEN bit, and perform the required data read/write.
    b) if the PORTx bit is '`1`' – disable the SPIx module by setting the SPIEN bit, clear the SPI interrupt flag (SPIxIF), perform a dummy read of the SPIxBUF register, and return from the Interrupt Service Routine (ISR).

If DMA is being used, no work around exists.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

9. Module: SPI

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.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

10. Module: ECAN™

If any buffers other than Buffer 0 are enabled as transmit buffers (i.e., if the TXEN bits other than TXEN0 are set to ‘1’), incorrect ID and data transmissions will occur intermittently.

Work around

Enable only Buffer 0 for transmission.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

11. Module: ECAN

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.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

12. Module: ECAN

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

Work around

Do not use Loopback mode.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

13. Module: I^{2}C™

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

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

14. Module: ADC

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. That is, 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 when it is configured to generate an interrupt after each INT0 triggered conversion (SMPI<3:0> = 0000).

Work around

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

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

15. Module: Doze Mode

None of the following error traps will wake-up a device from Doze mode: the address error trap, stack error trap, math error trap, or the DMA error trap.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

16. Module: JTAG

JTAG programming does not work.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

17. 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 of the 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>).

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

18. Module: UART

The Receive Buffer Overrun Error Status bit, OERR (UxSTA<1>), may be 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.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

19. 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.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

20. Module: UART

The UTXISEL0 bit (UxSTA<13>) is always read as zero, regardless of the value written to it. 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.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

21. 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.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

22. 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).

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

23. Module: I^2C

Writing to I2CxTRN during a Start bit transmission generates a write collision, indicated by the IWCOL bit (I2CxSTAT<7>) 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.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

24. Module: I^2C

The ACKSTAT bit (I2CxSTAT<15>) 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 whether 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

The SDA pin should be connected to any other available I/O pin on the device. After transmitting a byte, the slave should poll the SDA line (subject to a time-out period that is dependent on the application) to determine whether an ACK (‘0’) or a NACK (‘1’) was received.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

25. Module: I^2C

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

Work around

Use the D_A Status bit only for determining slave reception status. Do not use it for determining slave transmission status.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

26. Module: Interrupt Controller

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. Regardless, the Trap Service Routine must be included in the user application.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

27. Module: Internal Voltage Regulator

If a $\overline{\text{MCLR}}$ 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 (BOR).

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

28. Module: ECAN

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.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

29. Module: 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 5.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

TABLE 5: INTERNAL FRC ACCURACY

AC Characteristics		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +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 ≤ TA ≤ +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.

30. 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, including the following devices:

- dsPIC33FJ64MC506
- dsPIC33FJ64MC706
- dsPIC33FJ128MC506
- dsPIC33FJ128MC706

Work around

Two conditions apply:

1. If the SPI2 module is used, pin 34 cannot be used as an I/O (RF2). Using another I/O pin is recommended.
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.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

31. 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 the user software.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

32. 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 (excluding Configuration Memory 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), of the “*dsPIC33F Family Reference Manual*”.
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, please contact your local Microchip representative.

Steps 1 through 5 in the work around are implemented in MPLAB IDE version 8.00 or higher for the MPLAB ICD 2, MPLAB REAL ICE™ in-circuit emulator and PM3 tools.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

33. 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.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

34. Module: DMA

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

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

35. Module: Output Compare

When the output compare module is operated in the Dual Compare Match mode, a timer compare match with the value in the OCxR register sets the OCx output, producing a rising edge on the OCx pin. Then, when a timer compare match with the value in the OCxRS register occurs, the OCx output is reset, producing a falling edge on the OCx pin.

The above statement applies to all conditions except when the difference between OCxR and OCxRS is 1. In this case, the output compare module may miss the Reset compare event, and cause the OCx pin to remain continuously high. This condition will remain until the difference between values in the OCxR and OCxRS registers is made greater than 1.

Work around

Ensure in software that the difference between values in OCxR and OCxRS registers is maintained greater than 1.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

36. Module: UART

When the UART is in 4x mode (BRGH = `1`) and using two Stop bits (STSEL = `1`), it may sample the first Stop bit instead of the second one.

This issue does not affect the other UART configurations.

Work around

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

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

37. Module: UART

When an auto-baud is detected, the receive interrupt may occur twice. The first interrupt occurs at the beginning of the Start bit and the second occurs after reception of the Sync field character.

Work around

If an extra interrupt is detected, ignore the additional interrupt.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

38. Module: DMA

When the DMA channel is configured for NULL Data Peripheral Write mode (DMAxCON<11> = 1), it does not execute a NULL (all zeros) write to the peripheral address.

Work around

Use two DMA channels to receive data from the peripheral module. One channel must be configured to transfer data from the peripheral to DMA RAM, while another channel must be configured to transfer dummy data from the DMA RAM to the peripheral. Both channels must be set up for the same DMA request.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

39. Module: DMA

A low priority DMA channel request can be preempted by a higher priority DMA channel request. For example, if DMA Channel 0 has a higher priority than DMA Channel 1. A request to DMA channel 1 will be pending while DMA Channel 0 is processing its request. If DMA Channel 1 receives another request while it is in a pending request state, the DMA module does not generate a DMA error trap event.

Work around

None. Using higher priority DMA channels for servicing sources of frequent requests significantly reduces the possibility of the condition described above occurring, but does not completely eliminate it.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

40. Module: DMA

When the DMA channel is configured for One Shot mode with NULL write enabled, the channel will write an extra NULL to the peripheral register after completing the last transfer. In the case of the SPI module and the SPIxBUF register, this would cause the SPI module to perform an extra receive operation.

Work around

None. In the case of using DMA NULL write with the SPI module, perform a dummy read of the SPIxBUF register, after the DMA transfer is completed, to clear the SPIRBF flag and prevent an unexpected overflow condition on the next SPI receive operation.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

41. Module: CPU

Any instruction executed inside a REPEAT loop that produces a Read-After-Write stall condition results in the instruction being executed fewer times than was intended.

An example of such code is:

```
repeat #0xf
inc [w1],[++w1]
```

Work around

Avoid repeating an instruction that creates a stall using a REPEAT instruction. Instead, use the DO instruction while using the dsPIC33F device. A code example is shown below:

```
    DO #0xf, end
    inc [w1],[++w1]
        end: nop
```

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

42. Module: Oscillator

For certain values of the TUN<5:0> bits (OSCTUN<5:0>), the resultant frequencies do not match the expected values.

As shown in Table 6, the actual frequencies obtained for different values of the TUN<5:0> bits are listed in terms of percentage change relative to the center frequency of 7.3728 MHz. The frequency errors listed in the table are approximate and may vary slightly from device to device.

It is recommended that the user application includes a way to measure the exact oscillator frequency in order to verify the frequencies listed below.

Work around

Configure your peripherals and other system parameters based on the actual frequencies listed in Table 6.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

TABLE 6:

TUN<5:0>	Expected Change from 7.3728 MHz	Actual Change from 7.3728 MHz
000000	—	—
000001	+0.375%	+0.375%
000010	+0.75%	+0.75%
000011	+1.125%	+1.125%
000100	+1.5%	+1.5%
000101	+1.875%	+1.875%
000110	+2.25%	+2.25%
000111	+2.625%	+2.625%
001000	+3%	+3%
001001	+3.375%	+3.375%
001010	+3.75%	+3.75%
001011	+4.125%	+4.125%
001100	+4.5%	+4.5%
001101	+4.875%	+4.875%
001110	+5.25%	+5.25%
001111	+5.625%	+5.625%
010000	+6%	+8.325%
010001	+6.375%	+8.7%
010010	+6.75%	+9.075%
010011	+7.125%	+9.45%
010100	+7.5%	+9.825%
010101	+7.875%	+10.2%
010110	+8.25%	+10.575%
010111	+8.625%	+10.95%
011000	+9%	+11.325%
011001	+9.375%	+11.7%
011010	+9.75%	+12.075%
011011	+10.125%	+12.45%
011100	+10.5%	+12.825%
011101	+10.875%	+13.2%
011110	+11.25%	+13.575%
011111	+11.625%	+13.95%
100000	-12%	-12%
100001	-11.625%	-11.625%
100010	-11.25%	-11.25%
100011	-10.875%	-10.875%
100100	-10.5%	-10.5%
100101	-10.125%	-10.125%
100110	-9.75%	-9.75%
100111	-9.375%	-9.375%
101000	-9%	-9%
101001	-8.625%	-8.625%
101010	-8.25%	-8.25%
101011	-7.875%	-7.875%
101100	-7.5%	-7.5%
101101	-7.125%	-7.125%
101110	-6.75%	-6.75%
101111	-6.375%	-6.375%
110000	-6%	-3.675%
110001	-5.625%	-3.3%
110010	-5.25%	-2.925%
110011	-4.875%	-2.55%
110100	-4.5%	-2.175%
110101	-4.125%	-1.8%
110110	-3.75%	-1.425%
110111	-3.375%	-1.05%
111000	-3%	-0.675%
111001	-2.625%	-0.3%
111010	-2.25%	+0.075%
111011	-1.875%	+0.45%
111100	-1.5%	+0.825%
111101	-1.125%	+1.2%
111110	-0.75%	+1.575%
111111	-0.375%	+1.95%

43. Module: UART

When the UART is configured for IR interface operations (UxMODE<9:8> = 11), the 16x baud clock signal on the BCLK pin is present only when the module is transmitting. The pin is idle at all other times.

Work around

Configure one of the output compare modules to generate the required baud clock signal when the UART is receiving data or in an Idle state.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

44. Module: SPI

Setting the DISSCK bit in the SPIxCON1 register does not allow the user application to use the SCK pin as a general purpose I/O pin.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

45. Module: I^{2}C

The BCL bit in I2CSTAT can be cleared only with 16-bit operation, and can be corrupted with 1-bit or 8-bit operations on I2CSTAT.

Work around

Use 16-bit operations to clear BCL.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

46. Module: I^{2}C

If there are two I^{2}C devices on the bus, one of them acts as the master receiver and the other acts as the slave transmitter. If both devices are configured for 10-bit Addressing mode, and have the same value in the A10 and A9 bits of their addresses: then, when the slave select address is sent from the master, both the master and slave acknowledge it. When the master sends out the read operation, both the master and the slave enter into Read mode, and both of them transmit the data. The resultant data will be the ANDing of the two transmissions.

Work around

In all I^{2}C devices, the addresses, as well as bits A10 and A9, should be different.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

47. Module: I^{2}C

When the I^{2}C module is configured as a 10-bit slave with an address of 0x102, the I2CxRCV register content for the lower address byte is 0x01, rather than 0x02. However, the I^{2}C module acknowledges both address bytes.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

48. Module: I^{2}C

With the I^{2}C module enabled, the PORT bits and external interrupt input functions associated with the SCL and SDA pins (if any) do not reflect the actual digital logic levels on the pins.

Work around

If the SDA and/or SCL pins need to be polled, these pins should be connected to other port pins to be read correctly. This issue *does not* affect the operation of the I^{2}C module.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

49. Module: I^{2}C

In 10-bit Addressing mode, some address matches do not set the RBF flag or load the receive register, I2CxRCV, if the lower address byte matches the reserved addresses. In particular, these include all addresses with the form XX0000XXXX and XX1111XXXX, with the following exceptions:

- 001111000X
- 011111001X
- 101111010X
- 111111011X

Work around

Ensure that the lower address byte in 10-bit Addressing mode does not match any 7-bit reserved addresses.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

50. Module: Internal Voltage Regulator

When the VREGS bit (RCON<8>) is set to a logic '0', the device may reset, and higher sleep current may be observed.

Work around

Ensure that the VREGS bit (RCON<8>) is set to a logic '1' for device Sleep mode operation.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

51. Module: PSV Operations

An address error trap occurs in certain addressing modes when accessing the first four bytes of a PSV page. This only occurs when using the following addressing modes:

- MOV.D
- Register Indirect Addressing (Word or Byte mode) with pre/post-decrement

Work around

Do not perform PSV accesses to any of the first four bytes using the above addressing modes. For applications using the C language, MPLAB C Compiler for dsPIC DSCs (formerly known as the MPLAB C30 C Compiler), version 3.11 or higher, provides the following command-line switch that implements a work around for the erratum.

```
-merrata=psv_trap
```

Refer to the `readme.txt` file in the MPLAB C Compiler for dsPIC DSCs, v3.11 tool suite for further details.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

52. Module: UART

The UART error interrupt may not occur, or may occur at an incorrect time, if multiple errors occur during a short period of time.

Work around

Read the error flags in the UxSTA register whenever a byte is received to verify the error status. In most cases, these bits will be correct, even if the UART error interrupt fails to occur.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

53. Module: UART

When the UART is operating in 8-bit mode (PDSEL = 0x) and using the IrDA® encoder/decoder (IREN = 1), the module incorrectly transmits a data payload of 80h as 00h.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

54. Module: ECAN

The WAKIF bit in the CxINTF register cannot be cleared by software instruction after the device is interrupted from Sleep by activity on the CAN bus.

When the device wakes up from Sleep due to CAN bus activity, the ECAN module is placed in operational mode. The ECAN event interrupt occurs due to the WAKIF flag. Attempts to clear the flag in the Interrupt Service Routine may not clear the flag. The WAKIF bit being set will not cause repetitive Interrupt Service Routine execution.

Work around

Although the WAKIF bit does not clear, the device Sleep and ECAN Wake function continue to work as expected. If the ECAN event is enabled, the CPU will enter the Interrupt Service Routine due to the WAKIF flag getting set. The application can maintain a secondary flag, which tracks the device Sleep and Wake events.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

55. Module: I^{2}C

When the I^{2}C module is operating in Master mode, after the ACKSTAT bit is set when receiving a NACK from the slave, it may be cleared by the reception of a Start or Stop bit.

Work around

Store the value of the ACKSTAT bit immediately after receiving a NACK.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

56. Module: PWM

Under certain conditions, devices in the motor control family have a glitch in the PWMxL signal. The glitch is a brief high pulse during the low portion of the duty cycle. This error occurs when the module is configured in Single-Shot mode (PTMOD<1:0> = `01`) with complementary output. It also occurs when resuming from a Fault condition in Free-Running mode (PTMOD<1:0) = `00`) with complementary output.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

57. Module: PWM

If a Fault is cleared by user software during Latched mode, while the current PWM cycle remains active, any additional faults that arrive before the end of the current PWM cycle will not be detected. The PWM outputs will return to the non-Fault state at the end of that cycle and remain there, even if the fault is currently active. No interrupt will be sent to the interrupt controller.

Work around

While in the Latched mode of operation, do not attempt to clear faults during the same cycle in which they arrive.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

58. Module: PWM

The PWM outputs, PWM2H and PWM2L, do not generate PWM signals if the following two conditions are met:

1. FLTBCON register is written with any value.
2. Fault pin B ($\overline{\text{FLTB}}$) is low while the FLTBCON register is being written.

Work around

There are three possible work arounds for this issue:

1. Do not write to the FLTBCON register.
2. If Fault B is needed for the application, always keep the $\overline{\text{FLTB}}$ pin connected to logic high while writing to the FLTBCON register.
3. Set the $\overline{\text{FLTA}}$ pin to a logic low while writing to the FLTBCON register.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

59. Module: PWM

If the PTDIR bit is set, when PTMR is counting down, and the CPU execution is halted, after a breakpoint is reached, PTMR will start counting up as if PTDIR was zero.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

60. Module: PWM

An active Fault while the Motor Control PWM module is stopped in Idle mode does not result in an interrupt request being sent to the processor. Therefore, the processor does not wake-up from Idle mode.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

61. Module: PWM

When the device is operated in Doze mode and the motor control PWM module has a postscaler set to any value different than 1:1 (PTOPS > 0 in PxTCON register), the motor control PWM module generates more interrupts than expected.

Work around

Do not use Doze mode with the motor control PWM if the time base output postscaler is different than 1:1.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

62. Module: QEI

The Quadrature Encoder Interface (QEI) module does not generate an interrupt when MAXCNT is set to 0xFFFF and the following events occur:

1. POSCNT underflows from 0x0000 to 0xFFFF
2. POSCNT stops
3. POSCNT overflows from 0xFFFF to 0x0000

This sequence of events occurs when the motor is running in one direction, which causes POSCNT to underflow to 0xFFFF. Then, if the motor stops and starts running in the opposite direction, an overflow from 0xFFFF to 0x0000 will be generated. The QEI module does not generate an interrupt when this condition occurs.

Work around

To prevent this condition from occurring, set MAXCNT to 0x7FFF, which will cause an interrupt to be generated by the QEI module.

In addition, a global variable could be used to keep track of bit 15, so that when an overflow or underflow condition is present on POSCNT, the variable will toggle bit 15. Example 1 shows the code required for this global variable.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

EXAMPLE 1:

```
unsigned int POSCNT_b15 = 0;
unsigned int Motor_Position = 0;

int main(void)
{
            // ... User's code

            MAXCNT = 0x7FFF;      // Instead of 0xFFFF

            Motor_Position = POSCNT_b15 + POSCNT;

            // ... User's code
}
void __attribute__((__interrupt__)) _QEIInterrupt(void)
{
            IFSxbits.QEIIF = 0;   // Clear QEI interrupt flag
                                  // x=2 for dsPIC30F
                                  // x=3 for dsPIC33F
            POSCNT_b15 ^= 0x8000; // Overflow or Underflow
}
```

63. Module: SPI

Writing to the SPIxBUF register as soon as the TBF bit is cleared will cause the SPI module to ignore the written data. Applications which use SPI with DMA will not be affected by this erratum.

Work around

After the TBF bit is cleared, wait for a minimum duration of one SPI Clock before writing to the SPIxBUF register.

Alternately, do one of the following:

a) Poll the RBF bit and wait for it to get set before writing to the SPIxBUF register
b) Poll the SPI Interrupt flag and wait for it to get set before writing to the SPIxBUF register
c) Use an SPI Interrupt Service Routine (ISR)
d) Use DMA

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

64. Module: UART

The UART module will not generate consecutive break characters. Trying to perform a back-to-back Break character transmission will cause the UART module to transmit the dummy character that is used to generate the first Break character instead of transmitting the second Break character. Break characters are generated correctly if they are followed by a non-Break character transmission.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

65. Module: QEI

When the TQCS and TQGATE bits in the QEIxCON register are set, a QEI interrupt should be generated after an input pulse on the QEA input. This interrupt is not generated in the affected silicon.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

66. Module: QEI

When the TQCS and TQGATE bits in the QEIxCON register are set, the POSCNT counter should not increment but erroneously does; and if allowed to increment to match MAXCNT, a QEI interrupt will be generated.

Work around

To prevent the erroneous increment of POSCNT while running the QEI in Timer Gated Accumulation mode, initialize MAXCNT = 0.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

67. Module: I/O

While the device is being programmed via the PGECx/PGEDx pin pair, the device pin with SDO1 functionality may start toggling.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

68. Module: SPI

Regardless of the Slave setting for the Frame delay bit (FRMDLY = 0 or FRMDLY = 1), the Slave always acts as if the sync pulse precedes the first SPI data bit (FRMDLY = 0). The SPI will not function as described if Slave FRMDLY = 1.

Work around

None.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

69. Module: ADC

If the ADC module is in an enabled state when the device enters Sleep mode as a result of executing a `PWRSAV #0` instruction, the device power-down current (IPD) may exceed the specifications listed in the device data sheet. This may happen even if the ADC module is disabled by clearing the ADON bit prior to entering Sleep mode.

Work around

In order to remain within the IPD specifications listed in the device data sheet, the user software must completely disable the ADC module by setting the ADC Module Disable bit in the corresponding Peripheral Module Disable register (PMDx), prior to executing a `PWRSAV #0` instruction.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

70. Module: CPU

Code execution while DISI is active, (i.e., the DISICNT register is greater than zero), may extend level 1 through 6 interrupts on a cycle-by-cycle basis depending on the instruction coding sequences following the `DISI` instruction. Under such conditions, the DISICNT register instruction cycle decrement may not occur, and therefore, may extend the DISI by one instruction cycle for each occurrence of the instruction condition.

Work arounds

Use one of the following three work arounds to resolve this issue:

Work around 1

To guarantee a deterministic DISI cycle count, use either a `DISI #0` or the equivalent C30 macro `__builtin__disi(0);` in the user application code to terminate an active DISI.

Example 2 provides Assembly and C30 code examples to accomplish this option.

EXAMPLE 2:

Assembly Code Example:

```
//--------------------------------------------------------------------------------------------
// Block interrupts levels 1-6 for 0x5 instruction cycles.
//--------------------------------------------------------------------------------------------
DISI #3FFF     ;Block interrupts levels 1-6 for 0x4000 instruction cycles
MOV #52, W0    ;1-instruction cycle
BTSC W0, #0    ;2-instruction cycles
CLR W0         ;1-instruction cycle

;REPEAT #9;OPTIONAL to extend DISI from 5 to 15 instruction cycles
;NOP

DISI #0        ;1-instruction cycle (Terminate DISI, unblock any pending interrupts)
```

C30 Code Example Using C30 Defined Macros:

```
//--------------------------------------------------------------------------------------------
//Block interrupts priorities 1-6 for user-designated code segment.
//--------------------------------------------------------------------------------------------
__builtin__disi(0x3FFF);  //Block interrupts priorities 1-6 for 0x4000 instruction cycles

; CODE BLOCK TO BE PROTECTED FROM INTERRUPTS
………………..                //User code block
………………..
………………..

__builtin__disi(0);       //Re-enable interrupts priorities 1-6
```

Work around 2

An alternative to the DISI implementation, is to elevate the CPU interrupt level directly to disable same or lower priority interrupts.

Example 3 provides Assembly and C30 code examples to accomplish this option.

EXAMPLE 3:

Assembly Code Example:

```
push W0             ;save current W0
push SR             ;save current status reg
mov #0x00E0, W0     ;NOTE this blocks level 7 interrupts and  below. DISI blocks level six and below
ior SR
nop

; CODE BLOCK TO BE PROTECTED FROM INTERRUPTS
nop
nop

; restore context
pop SR
pop W0
```

C30 Code Example Using C30 Defined Macros:

```
//-------------------------------------------------------------------------------------------------
//Save current interrupt priority level in user variable "saved_to" then set CPU IPL priority to
//"ipl" which will block all other interrupts whose priority is ≤ "ipl". "ipl" cannot exceed 7.
//-------------------------------------------------------------------------------------------------
SET_AND_SAVE_CPU_IPL(saved_to, ipl);


; CODE BLOCK TO BE PROTECTED FROM INTERRUPTS
………………..                      //User code block
………………..
………………..

RESTORE_CPU_IPL(saved_to);    //Restore previous priority level
```

Work around 3

For pseudo-equivalency to DISI, another option would be to use a timer set to interrupt priority level 7 and the timer period set to the DISI count minus the (ISR interrupt latency + 1). Start the timer immediately just before the `DISI` instruction and in the timer ISR execute a `DISI #0`, and then disable and clear the timer.

Affected Silicon Revisions

A2	A3	A4					
X	X	X					

Data Sheet Clarifications

The following typographic corrections and clarifications are to be noted for the latest version of the device data sheet (DS70287**C**):

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

1. Module: DC Characteristics: I/O Pin Input Specifications

The maximum value for Parameter DI19 (VIL specifications for SDAx and SCLx pins) was stated incorrectly in Table 26-9 of the current device data sheet. Also, Parameters DI28 and DI29 (VIH specifications for SDAx and SCLx pins) values were not stated. The correct values are shown in bold type in Table 1.

2. Module: Electrical Characteristics (DC Specifications)

In Table 26-7, the maximum value of Parameter DC60b (Base IPD @ 3.3V and +85°C) is corrected to 850 µA. All other values are correct as shown.

TABLE 1: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
	VIL	Input Low Voltage					
DI18		**SDAx, SCLx**	VSS	—	0.3 VDD	V	SMBus disabled
DI19		**SDAx, SCLx**	VSS	—	**0.8**	V	SMBus enabled
	VIH	Input High Voltage					
DI28		**SDAx, SCLx**	**0.7 VDD**	**—**	**5.5**	**V**	**SMBus disabled**
DI29		**SDAx, SCLx**	**2.1**	**—**	**5.5**	**V**	**SMBus enabled**

APPENDIX A: REVISION HISTORY

Rev A Document (4/2009)

Initial release of this document; issued for revision A2, A3 and A4 silicon.

Includes silicon issues 1 (Doze Mode), 2-3 (ADC), 4-5 (CPU), 6 (Output Compare), 7-9 (SPI), 10-12 (ECAN™), 13 (I^{2}C™), 14 (ADC), 15 (Doze Mode), 16 (JTAG), 17-22 (UART), 23-25 (I^{2}C), 26 (Interrupt Controller), 27 (Internal Voltage Regulator), 28 (ECAN), 29 (Oscillator), 30 (SPI), 31 (UART), 32 (Device ID Register), 33-34 (DMA), 35 (Output Compare), 36-37 (UART), 38-40 (DMA), 41 (CPU), 42 (Oscillator), 43 (UART), 44 (SPI), 45-49 (I^{2}C), 50 (Internal Voltage Regulator), 51 (PSV Operations), 52-53 (UART), 54 (ECAN), 55 (I^{2}C), 56-61 (PWM), 62 (QEI) and 63 (SPI).

Rev B Document (8/2009)

Corrected revision IDs for the last two devices listed in Table 1 on page 2.

Updated silicon issue 10 (ECAN™).

Revised silicon issue 30 (SPI).

Updated the first sentence in the work around for silicon issue 32 (Device ID Register).

Revised silicon issue 62 (QEI).

Added silicon issues 64 (UART), 65-66 (QEI) and 67 (I/O).

Rev C Document (1/2010)

Added silicon issue 68 (SPI).

Rev D Document (6/2010)

Updated silicon issue 4 (CPU).

Added silicon issue 69 (ADC) and data sheet clarification 1 (DC Characteristics: I/O Pin Input Specifications).

Rev E Document (1/2012)

Added silicon issue 70 (CPU).

Rev F Document (1/2013)

Added data sheet clarification 2 (Electrical Characteristics (DC Specifications).

NOTES:

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.

QUALITY MANAGEMENT SYSTEM CERTIFIED BY DNV

═ ISO/TS 16949 ═

Trademarks

♻ Printed on recycled paper.

ISBN: 978-1-62076-849-5

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

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://www.microchip.com/support
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

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

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

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

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
Harbour 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: 86-10-8569-7000
Fax: 86-10-8528-2104

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

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hangzhou
Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

China - Hong Kong SAR
Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

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: 86-755-8864-2200
Fax: 86-755-8203-1760

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

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

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 - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo
Tel: 81-3-6880- 3770
Fax: 81-3-6880-3771

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

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

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

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: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7828
Fax: 886-7-330-9305

Taiwan - Taipei
Tel: 886-2-2508-8600
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

11/29/12