dsPIC30F1010/202X

dsPIC30F1010/202X Family
Silicon Errata and Data Sheet Clarification

The dsPIC30F1010/202X family devices that you have received conform functionally to the current Device Data Sheet (DS70178**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 dsPIC30F1010/202X 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 (**A3**).

Data Sheet clarifications and corrections start on page 23, 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 dsPIC30F1010/202X silicon revisions are shown in Table 1.

TABLE 1: SILICON DEVREV VALUES

Part Number	Device ID(1)	Revision ID for Silicon Revision(2)		
		A1	A2	A3
dsPIC30F1010	0x0404	0x1000	0x1002	0x1003
dsPIC30F2020	0x0400			
dsPIC30F2023	0x0403			

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 *"dsPIC30F SMPS Flash Programming Specification"* (DS70284) 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)		
				A1	A2	A3
PWM	Dead Time	1.	If a value less than 0x0010 is written to the DTRx and ALTDTRx registers, either or both of the PWMHx and PWMLx outputs will not function.	X	X	X
PWM	Duty Cycle	2.	Duty cycle resolution is not 1.1 ns over the entire duty cycle range.	X	X	X
PWM	Special Event Trigger and Individual Trigger	3.	The PWM Special Event Trigger and PWM Individual Trigger do not function near the beginning of the PWM period.	X	X	X
PWM	Dead Time	4.	The dead-time registers (DTRx/ALTDTRx) must be modified only when the PWM is not running and should not be modified “on-the-fly”.	X	X	X
PWM	Override Enable	5.	The PWM override feature does not work correctly.	X	X	X
PWM	Duty Cycle	6.	Any duty cycle value less than or equal to 0x0010 causes the PWM outputs to flip to the inverted state.	X	X	X
PWM	Override Priority	7.	The PWM fault, current-limit and output override priorities do not work correctly.	X	X	X
PWM	PWM Pins	8.	In Push-Pull mode, with immediate updates enabled, the PWM pins may become swapped.	X	X	X
PWM	Operation Under Temperature	9.	The PWM module may not operate at temperatures below -20ºC.	X	X	
PWM	—	10.	The PWM output may exhibit an occasional jitter proportional to the operating speed of the dsPIC30F1010/202X device.	X	X	X
PWM	Current Reset Mode	11.	Setting the XPRES bit in the PWMCONx register should enable a current-limit source to reset the PWM period when the PWM generated is configured in Independent Time Base mode. This functionality is not working correctly.	X	X	X
ADC	Global Software Trigger	12.	The Global Software Trigger bit (GSWTRG in the ADCON register) is not reset unless the PxRDY bits in the ADSTAT register are reset.	X	X	X
ADC	Sample and Hold Timing	13.	The resolution of the PWM to ADC sample-and-hold trigger timing is up to 41.6 ns instead of the 8 ns specified in the device data sheet.	X	X	X
ADC	Interrupts	14.	Individual ADC Interrupts for the ADC pin pairs do not work.	X	X	X
ADC	Conversion Rate	15.	The maximum conversion rate for the ADC module is 1.5 Msps.	X	X	X
ADC	Sample and Hold Circuits	16.	Depending on conversion configuration, ADC inputs that do not have dedicated sample-and-hold circuits may produce inaccurate conversion results.	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)		
				A1	A2	A3
Output Compare	—	17.	The output compare module produces 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.	X	X	X
Output Compare	PWM Mode	18.	The output compare module will miss one compare event when the duty cycle register value is updated from 0x0000 to 0x0001.	X	X	X
Output Compare	Dual Compare Match Mode	19.	In Dual Compare Match mode, the OCx output is not reset when the OCxR and OCxRS registers are loaded with values having a difference of 1.	X	X	X
SPI	Slave Select Mode	20.	The SPI module slave select functionality will not work correctly.	X	X	X
SPI	Frame Master Mode	21.	The SPI module will fail to generate frame synchronization pulses in Frame Master mode if FRMDLY = 1.	X	X	X
SPI	Master Mode	22.	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
SPI	—	23.	The SPIxCON1 DISSCK bit does not influence port functionality.	X	X	X
UART	Baud Rate Generator	24.	If the Baud Rate Generator register (BRG) contains an odd value and the parity option is enabled, the module may falsely indicate parity errors.	X	X	X
UART	Receive Operation	25.	The Receive Buffer Overrun Error Status bit may be set prematurely.	X	X	X
UART	High-Speed Mode	26.	UART receptions may be corrupted in High Baud Rate mode (BRGH = 1).	X	X	X
UART	—	27.	UTXISEL0 bit in the UxSTA register is always read as zero regardless of the value written to it.	X	X	X
UART	High-Speed Mode	28.	The auto-baud feature does not work properly in High Baud Rate mode (BRGH = 1).	X	X	X
UART	Auto-Baud	29.	When the auto-baud feature is enabled, the Sync Break character (0x55) may be loaded into the FIFO as data.	X	X	X
UART	IrDA® Mode	30.	The operation of the RXINV bit in the UxMODE register is inverted.	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
UART	IrDA Mode	32.	The 16x baud clock signal on the BCLK pin is present only when the module is transmitting.	X	X	X
UART	High-Speed Mode	33.	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	FIFO Error Flags	34.	Under certain circumstances, the PERR and FERR error bits may not be correct for all bytes in the receive FIFO.	X	X	X
I^2C™	Bus Collision	35.	The Bus Collision Status bit does not get set when a bus collision occurs during a Restart or Stop event.	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)		
				A1	A2	A3
I^2C	Bus Collision	36.	The I2CxTRN register can be written to even if a write collision is detected.	X	X	X
I^2C	ACK/NACK Status	37.	The ACKSTAT bit does not reflect the status of a transmission received from an I^2C slave device.	X	X	X
I^2C	Slave Mode	38.	The D_A Status bit in the I2CxSTAT register is not set on a write to the I2CxTRN register by an I^2C slave device.	X	X	X
I^2C	1-bit and 8-bit Operation	39.	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	Addressing	40.	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	Addressing	41.	The 10-bit slave does not set the RBF flag or load the I2CxRCV register on an address match if the Least Significant bits (LSbs) of the address are the same as the 7-bit reserved addresses.	X	X	X
I^2C	Addressing	42.	If the I^2C module is configured for 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
$\overline{MCLR}$ Pin	BOR Event	43.	When the dsPIC® DSC is operated with the PLL enabled, the $\overline{MCLR}$ pin does not operate correctly when a brown-out condition occurs.	X	X	X
PSV Operations	—	44.	An address error trap occurs, in certain addressing modes, when accessing the first four bytes of any PSV page.	X	X	X
Sleep Mode	—	45.	Execution of the Sleep instruction (`PWRSAV #0`) may cause incorrect program operation after the device wakes up from Sleep. The current consumption during Sleep may also increase beyond the specifications listed in the device data sheet.	X	X	
RB7 Pin and All Multiplexed Functions	—	46.	The RB7 pin does not work. This pertains to all multiplexed functions on RB7, as well.	X		
CPU – DAW.b Instruction	`DAW.b` Instruction	47.	The Decimal Adjust instruction, `DAW.b`, may improperly clear the Carry bit, C (SR<0>).	X	X	X
EXTREF Pin	—	48.	The EXTREF pin is susceptible to voltage spikes below V_{SS}.	X		
Analog Comparator	PWM Fault/ Current-Limit and ADC Triggers	49.	The Analog Comparator may falsely switch state if an ADC conversion is triggered that coincides with two PWM edges in Independent Time Base mode.	X	X	X
ADC	Current Consumption in Sleep Mode	50.	If the ADC module is in an enabled state when the device enters Sleep Mode, the power-down current (I_{PD}) of the device may exceed the device data sheet specifications.	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 (**A3**).

1. Module: PWM

If dead time functionality is enabled (DTC<1:0> = 0 or 1 in the PWMCONx register), the minimum usable value that can be written to the dead time registers, DTRx and ALTDTRx, is 0x0010. Writing a value less than 0x0010 will cause either or both the PWMxH and PWMxL outputs not to function. As a result of this erratum, the minimum usable dead time is 16 ns. Dead time resolution is 4 ns for dead times greater than 16 ns.

Work around

The dead time must either be disabled (DTC<1:0> = 2) or DTRx and ALTDRx must have a value of 0x0010 or greater. If zero dead time is required, configure the DTC<1:0> bits in the PWMCONx register to specify no dead time.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

2. Module: PWM

The data sheet indicates that the power supply PWM module has a 1.1 ns duty cycle resolution. This is true for all values of PDCx except the following:

1. 0x0010 < PDCx < 0x0040
2. (Period – 0x0040) < PDCx < (Period – 0x0010)

In these ranges, duty cycle resolution is 16 ns. The PWM period is either the master period, PTPER, or the individual PWM generator period, PHASEx.

Work around

If possible, the system should be designed so that the PWM generator will operate in the duty cycle range where the 1.1 ns resolution is possible. For operation outside this range, the design must take into account the reduced resolution.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

3. Module: PWM

Each PWM generator can be configured to generate a trigger for the ADC module or a trigger interrupt at any point during the PWM period. The point in time during the PWM period that the trigger is set is specified in the TRIGx register for PWM Individual Trigger, or in the SEVTCMP register for the Special Event Trigger. The minimum trigger value in TRIGx or SEVTCMP is 0x0008. Values below 0x0008 result in a PWM trigger not being initiated at all. As a result, no ADC sampling or trigger interrupt will occur.

Work around

If the Special Event Trigger or the Individual Trigger is implemented, the user should perform a check in firmware to make sure that TRIGx and/or SEVTCMP is always greater than 0x0008 and less than the PWM period.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

4. Module: PWM

The dead-time registers (DTRx/ALTDTRx) must be modified only when the PWM is not running. Adjusting the dead time “on-the-fly” can result in an unpredictable glitch on the PWM output, which may cause shoot-through.

Work around

None.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

5. Module: PWM

The OVRDAT<1:0> bits in the IOCONx register should determine the state of the PWMx output pins when the OVRENH and OVRENL bits (IOCONx<9:8>) are set. However, the PWM override feature does not work correctly. The PWMxH and PWMxL pins do not exhibit the state specified by the OVRDAT<1:0> bits when only one of the override bits (OVRENH or OVRENL) is set. If both bits are set, the override state is exhibited correctly on the PWMxL and PWMxH pins.

Work around

If override capability is desired on only one of the PWM pin pairs, use the GPIO module to override the PWM outputs. This can be done using the PENH and PENL bits in the IOCONx register. When the PENH/PENL bits in the IOCONx register are cleared, the GPIO module assumes control of the PWMxH/L output pin. The GPIO module must be setup in advance for the desired override output states, and the pins must be configured as digital outputs. This includes setting the PORTx and TRISx registers correctly, which correspond to the PWMxH and PWMxL pins.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

6. Module: PWM

The dsPIC30F1010/202X device data sheet states that the minimum PWM duty cycle value is 0x0010. Duty cycle values less than 0x0010 should cause the PWM outputs to display states corresponding to a duty cycle value of 0x0000. However, when a value of 0x0010 or less is loaded into the selected duty cycle register, the outputs of the PWM generator (PWMxH and PWMxL) will exhibit a state opposite of the expected state.

For example, if the expected state of the PWM output is a continuous '0', a continuous '1' will be observed, and vice versa. The above behavior applies when the Master Duty Cycle register (MDC) or Individual Duty Cycle register (PDCx) provides the duty cycle value.

Work around

Do not load the duty cycle register with a value less than or equal to 0x0010.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

7. Module: PWM

The "*dsPIC30F1010/202X Data Sheet*" (DS70178) states the priority of PWMx pin ownership as:

- PWM Generator (lowest priority)
- Output Override
- Current-Limit Override
- Fault Override
- PENx (GPIO/PWM) Ownership (highest priority)

Instead of following the above priority scheme, the PWMx pin ownership is determined by ANDing the Output Override Data bits (OVRDAT<1:0>), Current-Limit Override Data bits (CLDAT<1:0>) and Fault Override Data bits (FLTDAT<1:0>) in the IOCONx register.

For example, the override data may be set as follows:

- OVRDAT<1:0> = `00`
- CLDAT<1:0> = `01`
- FLTDAT<1:0> = `10`

If all three overrides occur simultaneously, the following operations shown in Equation 1 will determine the state of the PWMx pin.

Therefore, when multiple overrides occur simultaneously, only the override data for the active override sources will be ANDed together, while the inactive override sources will be ignored.

If only one override is active, override priorities do not apply and operation of the PWM overrides is normal.

Work around

None.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

EQUATION 1:

PWMxH = (OVRDAT<1>) AND (CLDAT<1>) AND (FLTDAT<1>) = 0 AND 0 AND 1 = 0

PWMxL = (OVRDAT<0>) AND (CLDAT<0>) AND (FLTDAT<0>) = 0 AND 1 AND 0 = 0

8. Module: PWM

In Push-Pull mode, with immediate updates enabled, the PWM pins may become swapped.

Work around

If using the PWM module in Push-Pull mode, immediate updates must be disabled.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

9. Module: PWM

The PWM module may not operate at temperatures below -20ºC. During this condition, the PWM module will relinquish control of the associated PWM pin and the Port I/O will determine the state of the pin. In addition, the PWM module will stop generating the ADC trigger before the module relinquishes control of the PWM pins.

Work around

None.

Affected Silicon Revisions

A1	A2	A3					
X	X						

10. Module: PWM

The outputs of the PWM module may exhibit a jitter proportional to the speed of operation of the device. The jitter may be observed as a deviation in the PWM period, duty cycle or phase, and may be affected independently of each other. As a result, the maximum deviation exhibited on the PWM output pin at 30 MIPS is 8.4 ns.

The jitter is caused by silicon process variations, noise on the V_{DD} rail and the operating temperature of the dsPIC® DSC. However, for a given set of operating conditions, the maximum jitter will be the same for all three parameters, and independent of each other.

Table 3 shows the maximum jitter that may be exhibited at various operating speeds.

TABLE 3:

Speed of Operation	Maximum Jitter on PWM Output
30 MIPS	8.4 ns
20 MIPS	12.6 ns
15 MIPS	16.8 ns

The maximum jitter at any operating speed can be determined using Equation 2.

EQUATION 2:

$$\textit{Maximum jitter observed (ns)} = \frac{252}{(S)}$$

Where:

- S is the speed of operation in MIPS.

The maximum percentage error observed on the PWM output can be calculated using Equation 3.

EQUATION 3:

$$\textit{Error (\%)} = \pm\left[\frac{(x_{programmed} - x_{observed})}{x_{programmed}}\right] \cdot 100$$

Where:

- $x_{observed}$ is the observed value of parameter of interest (PWM period, duty cycle or phase).
- $x_{programmed}$ is the programmed value of parameter of interest (PWM period, duty cycle or phase).

Work around

Operate the power supply PWM module so that the percentage error in the parameter of interest (from Equation 3) is within permissible limits of the application.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

11. Module: PWM

Setting the XPRES bit in the PWMCONx register should enable a current-limit source to reset the PWM period in Independent Time Base mode. This mode is not functioning correctly.

If the selected current-limit signal (either an analog comparator or external signal) triggers after the falling edge of PWMH, then the XPRES operation functions correctly. The PWM deasserted time is truncated and the PWM period is terminated early, and a new PWM cycle begins.

If the selected current-limit signal (either an analog comparator or external signal) triggers before the falling edge of PWMH, the PWMH asserted time is truncated, and the inactive time after the falling edge PWMH remains constant.

The proper XPRES behavior is to ignore the current-limit signal until the falling edge of the PWM period.

This issue may not be a problem in applications that control inductor current above a specified minimum current level. When the inductor current falls below the specified minimum value during the PWMH off-time, the PWM period is truncated and a new cycle begins to increase the inductor current.

Work around

None.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

12. Module: ADC

In order to perform multiple analog-to-digital conversions using the global software trigger, the PxRDY bits in the ADSTAT register must be cleared. The data sheet indicates that the user can configure the ADC pin pairs to perform a conversion when the GSWTRG bit in the ADCON register is set. When the conversion is available, the user must then clear the GSWTRG bit and set it again to perform another conversion. Contrary to what the data sheet indicates, this will not initiate another conversion unless the PxRDY bits are cleared. Clearing the PxRDY bits automatically clears the GSWTRG bit.

This only applies to a polling-based approach. If an interrupt-based approach is used, the user is required to clear the PxRDY bits in the ADC Interrupt Service Routine (ISR).

Work around

The following sequence should be followed to manually trigger ADC conversions using the global software trigger (polling-based only.)

1. Set the GSWTRG bit in ADCON to initiate a conversion on channels which have the trigger source as the global software trigger (via the TRGSRCx<5:0> bits in the ADCPCx registers).
2. Check the PxRDY bits to determine when the conversion(s) is completed.
3. Clear the PxRDY bits. The GSWTRG bit will be cleared as a result of this operation.
4. Repeat steps 1 through 3 to perform additional conversions.

Alternately, the individual software trigger can be selected by setting the TRGSRCx<5:0> bits in the ADCPCx register equal to 0x01. Instead of using the global software trigger, the individual software trigger bits (ADCPCx<SWTRGx>) can be used to trigger a conversion on a given analog pin pair. In a bit-polling approach, the PENDx in the ADCPCx register should be used to determine when a conversion is completed. In an interrupt-based approach, the PxRDY bits get set when the conversion is complete. These bits must be cleared in the ADC Interrupt Service Routine in order to enable future interrupts.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

13. Module: ADC

The dedicated ADC sample-and-hold circuits can be triggered by signals from the PWM module. The dsPIC30F1010/202X data sheet indicates that the resolution of the PWM to ADC sample-and-hold trigger timing is 8 ns. The existing implementation has a 41.6 ns resolution. In other words, when the PWM to ADC trigger is fired, an ADC sample may occur 1 ns to 41.6 ns later.

Work around

None.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

14. Module: ADC

The dsPIC30F1010/202X data sheet specifies that each ADC pin pair has its own interrupt vector. These interrupts do not work on the dsPIC30F1010/202X devices.

Work around

Each ADC pin pair can be configured to initiate a global ADC interrupt by setting the corresponding IRQENx bit in the ADCPCx register. The ADBASE register can be used to create a jump table in the global ADC interrupt which will execute the appropriate ADC service routine for a particular ADC pin pair. There is an ADBASE register code example in the dsPIC30F1010/202X data sheet which illustrates using the ADBASE register in this way.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

15. Module: ADC

The data sheet indicates that the conversion rate for the ADC module is 2.0 Msps. The ADC module on the dsPIC30F1010/202X silicon has a maximum conversion rate of 1.5 Msps.

Work around

None.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

16. Module: ADC

In the dsPIC30F202X device, the ADC inputs that do not have a dedicated sample-and-hold circuit will yield inaccurate conversion results unless the work around is implemented. The channels included are AN1, AN3, AN5, AN7, AN8, AN9, AN10 and AN11 (depending on the package variant). In the dsPIC30F1010 device, all of the channels mentioned above are included, plus AN4 and AN6.

Work around

In the ADCON register, configure the ADC with Order = `0` and SEQSAMP = `1`. This configuration allows for accurate conversion of the analog channels which use the shared sample-and-hold circuit. The exception to this is the RB7 pin and all multiplexed functions (refer to silicon issue number 46.).

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

17. Module: Output Compare

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 (TCY) 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.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

18. 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 missed only 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

A1	A2	A3					
X	X	X					

19. 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

A1	A2	A3					
X	X	X					

20. Module: SPI

The SPI module slave select functionality (enabled by setting 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. This applies to the dsPIC30F2023 device only.

Note: The dsPIC30F1010/202X devices have only one SPI. All references to x = 2 are intended for software compatibility with other dsPIC DSC devices.

Work around

Manually poll the $\overline{SSx}$ pin state in the SPI interrupt by reading the associated PORT bit:

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

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

21. 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). However, the module functions correctly in Frame Slave mode, and also in Frame Master mode if FRMDLY = 0. This applies to the dsPIC30F2023 device only.

Work around

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 FRMDLY = 0, no work around is needed.

Note: The dsPIC30F1010/202X devices have only one SPI. All references to x = 2 are intended for software compatibility with other dsPIC DSC devices.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

22. 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 the data output time.

Note: The dsPIC30F1010/202X devices have only one SPI. All references to x = 2 are intended for software compatibility with other dsPIC DSC devices.

Work around

If sampling at the middle of the 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

A1	A2	A3					
X	X	X					

23. 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

A1	A2	A3					
X	X	X					

24. 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

A1	A2	A3					
X	X	X					

25. Module: UART

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

A1	A2	A3					
X	X	X					

26. Module: UART

UART receptions may be corrupted if the Baud Rate Generator (BRGH) 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

A1	A2	A3					
X	X	X					

27. 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

A1	A2	A3					
X	X	X					

28. Module: UART

The auto-baud feature may not calculate the correct baud rate when the High Baud Rate Enable bit, BRGH, is set. With BRGH 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

A1	A2	A3					
X	X	X					

29. 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

A1	A2	A3					
X	X	X					

30. Module: UART

The UART module can be used to transmit and receive IrDA® signals, with the use of an IrDA transceiver, by setting the IREN bit in the UxMODE register. In this mode, the operation of the RXINV bit enables reception of signals with an Idle state of either '1' or '0'. The operation of this bit is the inverse of the stated operation in the "*dsPIC30F1010/202X Data Sheet*" (DS70178).

The signal received from an IrDA transceiver can have an Idle state of '1' or '0'. The following table summarizes how UART receptions will occur when used with the IrDA decoder.

TABLE 4:

Type of Signal Used for Transmission	State of RXINV bit	UART reception
Idle State = 1	RXINV = 0	May be erroneous
	RXINV = 1	Error free
Idle State = 0	RXINV = 0	Error free
	RXINV = 1	May be erroneous

Work around

Invert the state of RXINV bit in the UxMODE register.

If the Idle state of the received signal is '1', configure RXINV = 1. If the Idle state of the received signal is '0', configure RXINV = 0.

Affected Silicon Revisions

A1	A2	A3					
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

A1	A2	A3					
X	X	X					

32. 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

A1	A2	A3					
X	X	X					

33. 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

A1	A2	A3					
X	X	X					

34. Module: UART

Under certain circumstances, the PERR and FERR error bits may not be correct for all bytes in the receive FIFO. This has only been observed when both of the following conditions are met:

- The UART receive interrupt is set to occur when the FIFO is full or three-quarters full (U1STA<7:6> = 1x), and
- More than two bytes with an error are received

In these two circumstances, only the first two bytes with a parity or framing error will have the corresponding bits indicate correctly. The error bits will not be set after this.

Work around

None.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

35. Module: I^2C

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

A1	A2	A3					
X	X	X					

36. 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 must be rewritten.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

37. 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

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

A1	A2	A3					
X	X	X					

38. Module: I²C

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 I2CxTRN.

Work around

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

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

39. Module: I²C

The BCL bit in I2CSTAT can be cleared only with a 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

A1	A2	A3					
X	X	X					

40. Module: I²C

If there are two I²C 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 I²C devices, the addresses, as well as bits A10 and A9, should be different.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

41. Module: I²C

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

The lower address byte in 10-bit Addressing mode should not be a reserved address.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

42. Module: I²C

If the I²C module is configured for 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 I²C module acknowledges both address bytes.

Work around

None.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

43. Module: $\overline{\text{MCLR}}$ Pin

A brown-out event occurs when V_{DD} drops below the minimum operating voltage for the device but not all the way down to V_{SS}. If the dsPIC DSC SMPS device is running with the PLL enabled and a brown-out event occurs, the device may stop running; and, the $\overline{\text{MCLR}}$ pin will not reset the device. When this occurs, the device can only be reset by cycling power to the V_{DD} pins.

It is recommended that an external Brown-out Reset (BOR) circuit be used to hold the device in Reset during a brown-out event to overcome this problem. The external BOR circuit will use the $\overline{\text{MCLR}}$ pin to hold the device in Reset. The following work around, in combination with the external BOR circuit, will ensure that the device is properly reset after a brown-out event occurs.

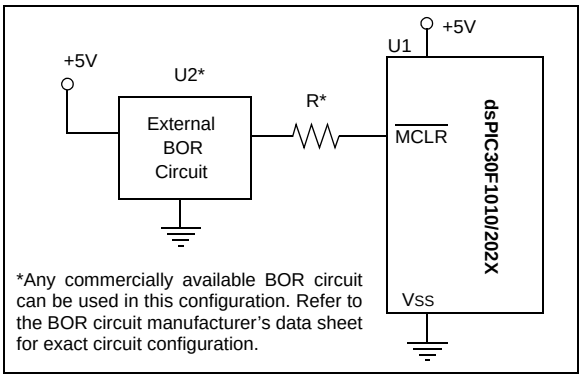
Work around

The dsPIC DSC SMPS device must be powered up with the PLL disabled, the Fail-Safe Clock Monitor enabled, and Clock Switching enabled. The PLL should be enabled in software via a clock switch after the device is reset (refer to **Section 29. "Oscillator"** (DS70268) in the "*dsPIC30F Family Reference Manual*" for details on clock switching). This ensures that the $\overline{\text{MCLR}}$ pin is functional and that the device can be reset by an external BOR circuit (see Figure 1).

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

FIGURE 1:



Use one of the following methods to perform the work around.

Method 1: Insert the code shown in Example 1 at the start of the program.

Method 2: Call the code shown in Example 1 in the beginning of code execution by including the `ClockSwitch.s` file in the project, and adding the following code:

- For assembly programming, add the following instruction at the beginning of the program:

```
.global __reset
…
…
__reset:
rcall ClockSwitch
…
…
```

- For C programming, add the following instruction at the beginning of the program:

```
int main(void)
{
        ClockSwitch;
        …
        …
}
```

EXAMPLE 1: CLOCK SWITCHING EXAMPLE

```
; This function performs a clock-switch from FRC to FRC+PLL. All other oscillator
; settings remain unchanged.
; Filename: ClockSwitch.s

_ClockSwitch:
    mov    #OSCCON+1,w4   ; Get address of high OSCCON byte
    mov    #0x0078, w0    ; 1st password for high byte access to OSCCON
    mov    #0x009A, w1    ; 2nd password for low byte access to OSCCON
    mov    #0x0001, w2    ; NOSC value for FRC+PLL
    mov.b  w0, [w4]       ; Write 1st password
    mov.b  w1, [w4]       ; Write 2nd password
    mov.b  w2, [w4]       ; Write NOSC value
    mov    #OSCCON,w4     ; Get address of low OSCCON byte
    mov    #0x0046, w0    ; 1st password for high byte access to OSCCON
    mov    #0x0057, w1    ; 2nd password for low byte access to OSCCON
    mov    #0x0001, w2    ; Set OSWEN bit
    mov.b  w0, [w4]       ; Write 1st password
    mov.b  w1, [w4]       ; Write 2nd password
    mov.b  w2, [w4]       ; Write OSWEN bit
    return
```

44. 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 for further details.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

45. Module: Sleep Mode

Execution of the Sleep instruction (`PWRSAV #0`) may cause incorrect program operation after the device wakes up from Sleep. The current consumption during Sleep may also increase beyond the specifications listed in the device data sheet.

Work arounds

To avoid this issue, any of the following three workarounds can be implemented, depending on the application requirements.

Work around 1:

Ensure that the `PWRSAV #0` instruction is located at the end of the last row of Program Flash Memory available on the target device, and fill the remainder of the row with `NOP` instructions.

This can be accomplished by replacing all occurrences of the `PWRSAV #0` instruction with a function call to a suitably aligned subroutine. The `address( )` attribute provided by the MPLAB ASM30 assembler can be used to correctly align the instructions in the subroutine. For an application written in C, the function call would be `GotoSleep( )`. For an assembly language application, the function call would be `CALL _GotoSleep`.

The Address Error Trap Service Routine software can then replace the invalid return address saved on the stack with the address of the instruction immediately following the `_GotoSleep` or `GotoSleep( )` function call. This ensures that the device continues executing the correct code sequence after waking up from Sleep mode.

Example 2 demonstrates the work around described above, as it would apply to a dsPIC30F2023 device.

EXAMPLE 2:

```
; ------------------------------------------------------------------------------------------------
.global __reset
.global _main
.global _GotoSleep
.global __AddressError
.global __INT1Interrupt
; ------------------------------------------------------------------------------------------------
    .section *, code
_main:
    BSET    INTCON2, #INT1EP  ; Set up INT pins to detect falling edge
    BCLR    IFS1, #INT1IF     ; Clear interrupt pin interrupt flag bits
    BSET    IEC1, #INT1IE     ; Enable ISR processing for INT pins
    CALL    _GotoSleep        ; Call function to enter SLEEP mode
_continue:
    BRA  _continue
; ------------------------------------------------------------------------------------------------
; Address Error Trap
__AddressError:
    BCLR    INTCON1, #ADDRERR
    ; Set program memory return address to _continue
    POP.D   W0
    MOV.B   #tblpage (_continue), W1
    MOV     #tbloffset (_continue), W0
    PUSH.D  W0
    RETFIE
; ------------------------------------------------------------------------------------------------
__INT1Interrupt:
    BCLR    IFS1, #INT1IF         ; Ensure flag is reset
    RETFIE                        ; Return from Interrupt Service Routine
; ------------------------------------------------------------------------------------------------
    .section *, code, address (0x1FC0)
_GotoSleep:
; fill remainder of the last row with NOP instructions
    .rept 31
        NOP
    .endr
; Place SLEEP instruction in the last word of program memory
    PWRSAV #0
```

Work around 2:

Instead of executing a `PWRSAV #0` instruction to put the device into Sleep mode, perform a clock switch to the 512 kHz Low-Power RC (LPRC) Oscillator with a 64:1 postscaler mode. This enables the device to operate at 0.002 MIPS, thereby significantly reducing the current consumption of the device. Similarly, instead of using an interrupt to wake-up the device from Sleep mode, perform another clock switch back to the original oscillator source to resume normal operation. Depending on the device, refer to **Section 7. "Oscillator"** (DS70054) or **Section 29. "Oscillator"** (DS70268) in the "*dsPIC30F Family Reference Manual*" (DS70046) for more details on performing a clock switch operation.

Note:	The above work around is recommended for users for whom application hardware changes are not possible.

Work around 3:

Instead of executing a `PWRSAV #0` instruction to put the device into Sleep mode, perform a clock switch to the 32 kHz Low-Power (LP) Oscillator with a 64:1 postscaler mode. This enables the device to operate at 0.000125 MIPS, thereby significantly reducing the current consumption of the device. Similarly, instead of using an interrupt to wake-up the device from Sleep mode, perform another clock switch back to the original oscillator source to resume normal operation. Depending on the device, refer to **Section 7. "Oscillator"** (DS70054) or **Section 29. "Oscillator"** (DS70268) in the "*dsPIC30F Family Reference Manual*" (DS70046) for more details on performing a clock switch operation.

Note:	The above work around is recommended for users for whom application hardware changes are possible, and also for users whose application hardware already includes a 32 kHz LP oscillator crystal.

Affected Silicon Revisions

A1	A2	A3					
X	X						

46. Module: RB7 Pin and All Multiplexed Functions

None of the functions multiplexed on this pin yields correct results.

TABLE 5: THE AFFECTED PINS ON THE dsPIC® DSC SMPS DEVICES

Device	Package	Pin Number	Functions Affected
dsPIC30F1010	28-pin SDIP	10	OSC2, CLKO, RB7
	28-pin SOIC	10	
	28-pin QFN	7	
dsPIC30F2020	28-pin SDIP	10	AN7, CMP3D, CMP4B, OSC2, CLKO, RB7
	28-pin SOIC	10	
	28-pin QFN	7	
dsPIC30F2023	44-pin TQFP	33	AN7, CMP3D, CMP4B, OSC2, CLKO, RB7
	44-pin QFN	33	

As a result of this erratum, the following issues exist:

- ADC channel AN7 does not produce the correct conversion results (dsPIC30F202X only)
- The comparator inputs, CMP3D and CMP4B, do not function (dsPIC30F202X only)
- An external crystal is not supported as an oscillator source
- The Clock Out function does not work
- RB7 cannot be used as a digital input or output

Work around

No work around is available for the OSC2 and CLKO functionality. The AN7, CMP3D, CMP4B and RB7 functions can be substituted by using other available pins.

Affected Silicon Revisions

A1	A2	A3					
X							

47. Module: CPU – DAW.b Instruction

The Decimal Adjust instruction, DAW.b, may improperly clear the Carry bit, C (SR<0>), when executed.

Work around

Check the state of the Carry bit prior to executing the DAW.b instruction. If the Carry bit has already been set, set the Carry bit again after executing the DAW.b instruction. Example 3 shows how the application should process the Carry bit during a BCD addition operation.

EXAMPLE 3: CHECK CARRY BIT BEFORE DAW.b

```
    .include “p30fxxxx.inc”
    .......
    mov.b   #0x80, w0  ;First BCD number
    mov.b   #0x80, w1  ;Second BCD number
    add.b   w0, w1, w2 ;Perform addition
    bra     NC, L0     ;If C set go to L0
    daw.b   w2         ;If not, do DAW and
    bset.b  SR, #C     ;set the carry bit
    bra     L1         ;and exit
L0:daw.b    w2
L1: ....
```

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

48. Module: EXTREF Pin

The EXTREF pin is susceptible to voltage spikes below VSS, which induces negative current flow into the pin. A voltage below VSS on any I/O pin is outside the operating specification of the device; however, the device is designed to tolerate small voltage and current transients without experiencing detrimental effects. The EXTREF pin is more susceptible to negative current spikes than the other pins on the dsPIC30F1010/202X **A3** devices.

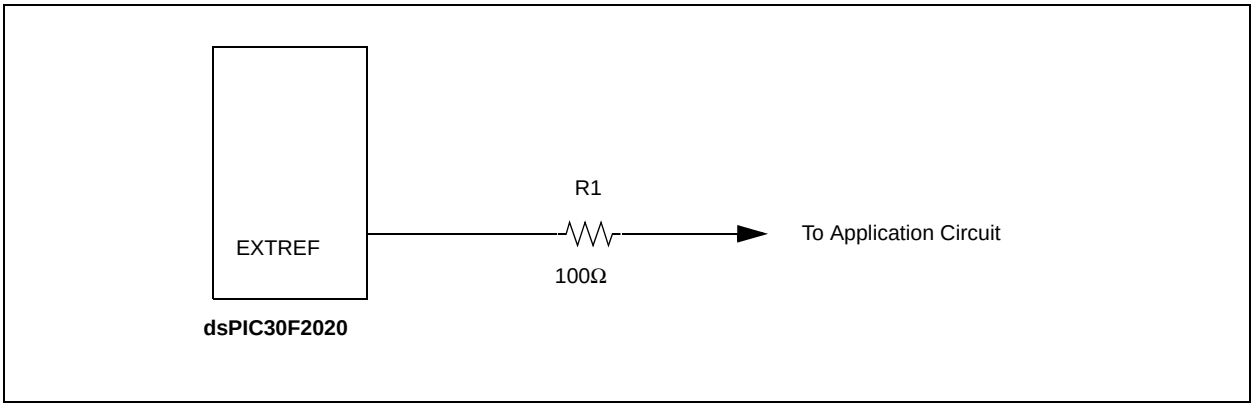
Work around

If the EXTREF pin is used in any of the configurations shown in Table 6, it is recommended that a 100 ohms 1/8W resistor be added in series with the EXTREF pin (see Figure 2).

TABLE 6: EXTREF PIN CONFIGURATIONS SUSCEPTIBLE TO NEGATIVE CURRENT TRANSIENTS

Pin Configuration	dsPIC30F1010	dsPIC30F2020	dsPIC30F2023
Digital Input	RE6	RE6	RB9
Comparator Input	N/A	N/A	CMP4D
Comparator Reference	EXTREF	EXTREF	EXTREF
Analog Input	N/A	N/A	AN9
External Timer Input	T1CK	T1CK	N/A
Input Change Notification	CN0	CN0	N/A

FIGURE 2: ADDITION OF 100 OHMS RESISTOR ON EXTREF PIN



The addition of the 100 ohms resistor effectively makes the EXTREF pin as robust as the other pins on the device with respect to tolerating voltage and current transients. In most of the configurations shown in Table 6, the 100 ohms resistor will have little effect on the operation of the pin. An exception to this is when the pin is configured as the comparator reference, EXTREF. Adding a resistor in series with the reference voltage will cause a slight offset in the gain of the comparator DAC. This gain offset is related to the voltage drop across the 100 ohms resistor.

The voltage seen at the EXTREF pin is given by Equation 4.

EQUATION 4: EXTREF VOLTAGE

$$V_{EXTREF} = \frac{V_{REF} - R \cdot (No\ of\ Comparators) \cdot V_{REF}}{40{,}000\ ohm + R}$$

Where:

- V_{EXTREF} is the voltage seen at the EXTREF pin.
- V_{REF} is the supplied comparator reference voltage.
- R is the resistor in series on the pin (100 ohms recommended).

Affected Silicon Revisions

A1	A2	A3					
X							

49. Module: Analog Comparator

Output from the Analog Comparator may falsely switch state if an ADC conversion trigger and two or more PWM edges occur at the same time. This behavior is observed when multiple PWM outputs are used in the Independent Time Base mode and the Analog Comparator is configured as a Fault or as a current-limit input to the PWM.

As a result of this erratum, the PWM module may exhibit spurious faults or current-limit events.

Work around

Ensure that the ADC conversion is triggered when multiple PWM edges are not aligned.

Affected Silicon Revisions

A1	A2	A3					
X	X	X					

50. 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

A1	A2	A3					
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 (DS70178**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 (V_{IL} specifications for SDAx and SCLx pins) and the minimum value for parameter DI29 (V_{IH} specifications for SDAx and SCLx pins) were stated incorrectly in Table 21-8 of the current device data sheet (DS70178**C**). The correct values are shown in bold type in Table 7.

TABLE 7: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 3.3V and 5.0V (±10%) (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
DI19	V_{IL}	Input Low Voltage SDA, SCL	V_{SS}	—	**0.8**	V	SMbus enabled
DI29	V_{IH}	Input High Voltage SDA, SCL	**2.1**	—	V_{DD}	V	SMbus enabled

APPENDIX A: REVISION HISTORY

Rev A Document (3/2009)

Initial release of this document; issued for revision A1, A2 and A3 silicon.

Includes silicon issues 1-11 (PWM), 12-16 (ADC), 17-19 (Output Compare), 20-23 (SPI), 24-34 (UART), 35-42 (I^2C), 43 (MCLR Pin), 44 (PSV Operations), 45 (Sleep Mode), 46 (RB7 Pin and All Multiplexed Functions), 47 (CPU – DAW.b Instruction) and 48 (EXTREF Pin).

No new silicon issues were added.

This document replaces the following errata documents:

- DS80290, *"dsPIC30F1010/202X Rev. A1 Silicon Errata"*
- DS80319, *"dsPIC30F1010/202X Rev. A2 Silicon Errata"*
- DS80391, *"dsPIC30F1010/202X Rev. A3 Silicon Errata"*

Rev B Document (7/2009)

Added silicon issue 49 (Analog Comparator).

Rev C Document (1/2010)

Updated silicon issue 6 (PWM).

Rev D Document (6/2010)

Added silicon issue 50 (ADC) and data sheet clarification 1 (DC Characteristics: I/O Pin Input Specifications).

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

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

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Trademarks

ISBN: 978-1-60932-249-6

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip received ISO/TS-16949:2002 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://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

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

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
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: 86-10-8528-2100
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 - Hong Kong SAR
Tel: 852-2401-1200
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: 86-755-8203-2660
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 - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

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: 886-3-6578-300
Fax: 886-3-6578-370

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

01/05/10