



MICROCHIP PIC12(L)F1822/PIC16(L)F1823

PIC12(L)F1822/PIC16(L)F1823 Family Silicon Errata and Data Sheet Clarification

The PIC12(L)F1822/PIC16(L)F1823 family devices that you have received conform functionally to the current Device Data Sheet (DS40001413E), 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 PIC12(L)F1822/PIC16(L)F1823 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 (A9).

Data Sheet clarifications and corrections start on [page 12](#), 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 a hardware debugger:

1. Using the appropriate interface, connect the device to the hardware debugger.
2. Open an MPLAB IDE project.
3. Configure the MPLAB IDE project for the appropriate device and hardware debugger.
4. Based on the version of MPLAB IDE you are using, do one of the following:
 - a) For MPLAB IDE 8, select *Programmer > Reconnect*.
 - b) For MPLAB X IDE, select *Window > Dashboard* and click the **Refresh Debug Tool Status** icon ().

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 DEVREV values for the various PIC12(L)F1822/PIC16(L)F1823 silicon revisions are shown in [Table 1](#).

TABLE 1: SILICON DEVREV VALUES

Part Number	DEVICE ID<13:0> ^{(1),(2)}			
	DEV<8:0>	Revision ID for Silicon Revision		
		A6	A8	A9
PIC12F1822	10 0111 000	0 0110	0 1000	0 1001
PIC12LF1822	10 1000 000	0 0110	0 1000	0 1001
PIC16F1823	10 0111 001	0 0110	0 1000	0 1001
PIC16LF1823	10 1000 001	0 0110	0 1000	0 1001

Note 1: The Device ID is located in the configuration memory at address 8006h.

2: Refer to the “PIC12(L)F1822/PIC16(L)F182X Memory Programming Specification” (DS41390) for detailed information on Device and Revision IDs for your specific device.

PIC12(L)F1822/PIC16(L)F1823

TABLE 2: SILICON ISSUE SUMMARY

Module	Feature	Item Number	Issue Summary	Affected Revisions ⁽¹⁾		
				A6	A8	A9
Oscillator	HS Oscillator	1.1	HS Oscillator min. VDD.	X		
Oscillator	Clock Switching	1.2	Clock switching can cause a single corrupted instruction.	X	X	
Oscillator	Oscillator Start-up Timer (OST) bit	1.3	OST bit remains set.	X	X	
ADC	Analog-to-Digital Converter	2.1	ADC conversion does not complete.	X		
APFCON	Remappable T1Gate	3.1	T1Gate is not remappable.	X		
Enhanced Capture Compare PWM (ECCP)	Enhanced PWM	4.1	PWM 0% duty cycle direction change.	X		
Enhanced Capture Compare PWM (ECCP)	Enhanced PWM	4.2	PWM 0% duty cycle port steering.	X		
Clock Switching	OSTS Status Bit	5.1	Remains clear when 4xPLL enabled.	X	X	X
Timer1 Gate	T1Gate Toggle mode	6.1	T1Gate flip-flop does not clear.	X		
In-Circuit Serial Programming™ (ICSP™)	Low-Voltage Programming	7.1	Bulk Erase not available with LVP.	X		
BOR	Wake-up from Sleep	8.1	Device may BOR Reset when waking-up from Sleep.	X	X	
Enhanced Universal Synchronous Asynchronous Receiver (EUSART)	16-Bit High-Speed Asynchronous mode	9.1	Works improperly at maximum rate.	X	X	X
Enhanced Universal Synchronous Asynchronous Receiver (EUSART)	Auto-Baud Detect	9.2	Auto-Baud Detect may store incorrect count value in the SPBRG registers.	X	X	
Master Synchronous Serial Port (MSSP)	SPI Master mode	10.1	Buffer Full (BF) bit or MSSP Interrupt Flag (SSPIF) bit becomes set half SCK cycle too early.	X	X	X
Master Synchronous Serial Port (MSSP)	SPI Slave mode	10.2	SPI master releasing Slave Select during Slave Sleep mode corrupts data.	X	X	X
Master Synchronous Serial Port (MSSP)	SPI Slave mode	10.3	SPI master enabling Slave Select too early could lose received data in Slave mode.	X	X	X
Master Synchronous Serial Port (MSSP)	SPI Slave mode	10.4	WCOL is erroneously set in SPI Slave mode during Sleep.	X	X	X

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

PIC12(L)F1822/PIC16(L)F1823

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 (**A9**).

1. Module: Oscillator

1.1 HS Oscillator

The HS oscillator requires a minimum voltage of 3.0 volts (at 65°C or less) to operate at 20 MHz.

Work around

None.

Affected Silicon Revisions

A6	A8	A9					
X							

1.2 Clock Switching

When switching clock sources between INTOSC clock source and an external clock source, one corrupted instruction may be executed after the switch occurs.

This issue does not affect Two-Speed Start-up or the Fail-Safe Clock Monitor operation.

Work around

When switching from an external oscillator clock source, first switch to 16 MHz HFINTOSC. Once running at 16 MHz HFINTOSC, configure IRCF to run at desired internal oscillator frequency.

When switching from an internal oscillator (INTOSC) to an external oscillator clock source, first switch to HFINTOSC High-Power mode (8 MHz or 16 MHz). Once running from HFINTOSC, switch to the external oscillator clock source.

Affected Silicon Revisions

A6	A8	A9					
X	X						

1.3 Oscillator Start-up Timer (OST) bit

During the Two-Speed Start-up sequence, the OST is enabled to count 1024 clock cycles. After the count is reached, the OSTS bit is set, the system clock is held low until the next falling edge of the external crystal (LP, XT or HS mode), before switching to the external clock source.

When an external oscillator is configured as the primary clock and Fail-Safe Clock mode is enabled (FCMEN = 1), any of the following conditions will result in the Oscillator Start-up Timer (OST) failing to restart:

- MCLR Reset
- Wake from Sleep
- Clock change from INTOSC to Primary Clock

This anomaly will manifest itself as a clock failure condition for external oscillators which take longer than the clock failure time-out period to start.

Work around

None.

Affected Silicon Revisions

A6	A8	A9					
X	X						

PIC12(L)F1822/PIC16(L)F1823

2. Module: ADC

2.1 Analog-to-Digital Converter (ADC)

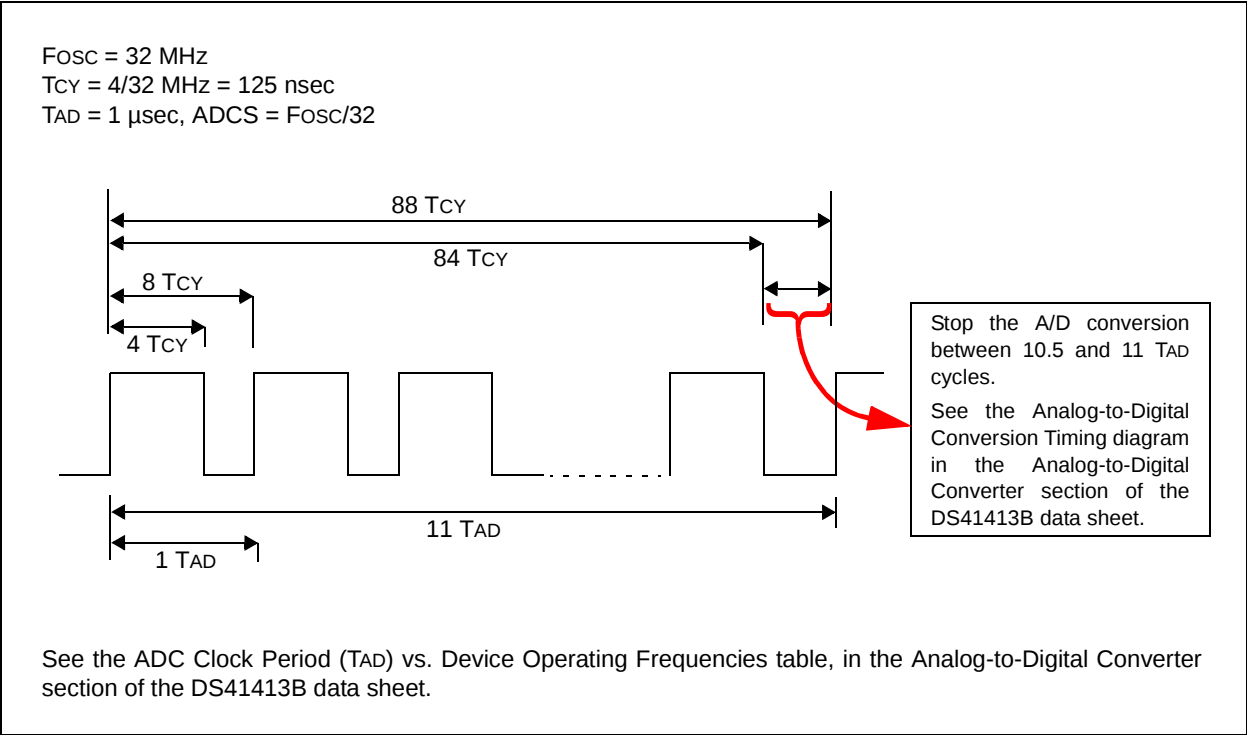
Under certain device operating conditions, the ADC conversion may not complete properly. When this occurs, the ADC Interrupt Flag (ADIF) does not get set, the GO/DONE bit does not get cleared and the conversion result does not get loaded into the ADRESH and ADRESL result registers.

Work around

Method 1: Select the dedicated RC oscillator as the ADC conversion clock source, and perform all conversions with the device in Sleep.

Method 2: Provide a fixed delay in software to stop the A-to-D conversion manually, after all 10 bits are converted, but before the conversion would complete automatically. The conversion is stopped by clearing the GO/DONE bit in software. The GO/DONE bit must be cleared during the last ½ TAD cycle, before the conversion would have completed automatically. Refer to [Figure 1](#) for details.

FIGURE 1: INSTRUCTION CYCLE DELAY CALCULATION EXAMPLE



In [Figure 1](#), 88 instruction cycles (TCY) will be required to complete the full conversion. Each TAD cycle consists of 8 TCY periods. A fixed delay is provided to stop the A/D conversion after 86 instruction cycles and terminate the conversion at the correct time as shown in the figure above.

Note: The exact delay time will depend on the choice of FOSC and the TAD divisor (ADCS) selection. The TCY counts shown in the timing diagram above apply to this example only. Refer to [Table 3](#) for the required delay counts for other configurations.

EXAMPLE 1: CODE EXAMPLE OF INSTRUCTION CYCLE DELAY

```
BSF    ADCON0, GO/DONE; Start ADC conversion
                        ; Provide 86
                        ; instruction cycle
                        ; delay here
BCF    ADCON0, GO/DONE; Terminate the
                        ; conversion manually
MOVF   ADRESH, W      ; Read conversion
                        ; result
```

For other combinations of FOSC, TAD values and instruction cycle delay counts, refer to [Table 3](#).

PIC12(L)F1822/PIC16(L)F1823

TABLE 3: INSTRUCTION CYCLE DELAY COUNTS FOR OTHER Fosc AND TAD COMBINATIONS

Fosc	TAD	Instruction Cycle Delay Counts
32 MHz	Fosc/64	172
	Fosc/32	86
16 MHz	Fosc/64	172
	Fosc/32	86
	Fosc/16	43
8 MHz	Fosc/32	86
	Fosc/16	43

Affected Silicon Revisions

A6	A8	A9					
X							

3. Module: APFCON

3.1 Timer1 Gate

The APFCON register is normally used to remap the T1 Gate to an alternate pin. The T1GSEL bit of the APFCON register is found to be not writable and therefore the T1Gate pin cannot be remapped. The default value for the T1GSEL bit is 0 and, therefore, the T1Gate will be found on RA4. This affects the PIC16(L)F1823 devices only.

Work around

None.

Affected Silicon Revisions

A6	A8	A9					
X							

4. Module: Enhanced Capture Compare PWM (ECCP)

4.1 Enhanced PWM

When the PWM is configured for Full-Bridge mode and the duty cycle is set to 0%, writing the Pxm<1:0> bits to change the direction has no effect on PxA and PxC outputs.

Work around

Increase the duty cycle to a value greater than 0% before changing directions.

Affected Silicon Revisions

A6	A8	A9					
X							

4.2 Enhanced PWM

In PWM mode, when the duty cycle is set to 0% and the STRxSYNC bit is set, writing the STRxA, STRxB, STRxC and the STRxD bits to enable/disable steering to port pins has no effect on the outputs.

Work around

Increase the duty cycle to a value greater than 0% before enabling/disabling steering to port pins.

Affected Silicon Revisions

A6	A8	A9					
X							

5. Module: Clock Switching

5.1 OSTS Status Bit

When the 4xPLL is enabled, the Oscillator Start-up Time-out Status (OSTS) bit always remains clear.

Work around

None.

Affected Silicon Revisions

A6	A8	A9					
X	X	X					

6. Module: Timer1 Gate

6.1 Timer1 Gate Toggle mode

When Timer1 Gate Toggle mode is enabled, it is possible to measure the full-cycle length of a Timer1 Gate signal. To perform this function, the Timer1 Gate source is routed through a flip-flop that changes state on every incrementing edge of the gate signal. Timer1 Gate Toggle mode is enabled by setting the T1GTM bit of the T1GCON register. When working properly, clearing either the T1GTM bit or the TMR1ON bit would also clear the output value of this flip-flop, and hold it clear. This is done in order to control which edge is being measured. The issue that exists is that clearing the TMR1ON bit does not clear the output value of the flip-flop and hold it clear.

Work around

Clear the T1GTM bit in the T1GCON register to clear and hold clear the output value of the flip-flop.

Affected Silicon Revisions

A6	A8	A9					
X							

PIC12(L)F1822/PIC16(L)F1823

7. Module: In-Circuit Serial Programming™ (ICSP™)

7.1 Low-Voltage Programming

The Bulk Erase feature is not available with Low-Voltage Programming mode.

A Bulk Erase of the program Flash memory or data memory cannot be executed in Low-Voltage Programming mode.

Work around

Method 1: If ICSP Low-Voltage Programming mode is required, use row erases to erase the program memory, as described in the Program/Verify mode section of the Programming Specification. Data memory must be overwritten with the desired values.

Method 2: Use the ICSP High-Voltage Programming mode if a Bulk Erase is required.

Note: Only the Bulk Erase feature will erase the program or data memory if the code or data protection is enabled. Method 2 must be used if the code or data protection is enabled.

Affected Silicon Revisions

A6	A8	A9					
X							

8. Module: BOR

8.1 BOR Reset

This issue affects only the PIC12LF1822/PIC16LF1823 devices. The devices may undergo a BOR Reset when waking-up from Sleep and BOR is re-enabled. A BOR Reset may also occur the moment the software BOR is enabled.

Under certain voltage and temperature conditions and when either SBODEN or BOR_NSLEEP is selected, the devices may occasionally reset when waking-up from Sleep or BOR is enabled.

Note: This issue pertains only to the LF product versions, PIC12LF1822 and PIC16LF1823.

Work around

Method 1: In applications where BOR use is not critical, turn off the BOR in the Configuration Word.

Method 2: Set the FVREN bit of the FVRCON register. Maintain this bit on at all times.

Method 3: When BOR module is needed only during run-time, use the software-enabled BOR by setting the SBODEN option on the Configuration Word. BOR should be turned off by software before Sleep, then follow the below sequence for turning BOR on after Wake-up:

- Wake-up event occurs
- Turn on FVR (FVREN bit of the FVRCON register)
- Wait until FVRRDY bit is set
- Wait 15 μ s after the FVR Ready bit is set
- Manually turn on the BOR

Method 4: Use the software-enabled BOR as described in Method 3, but use the following sequence:

- Switch to internal 32 kHz oscillator immediately before Sleep
- Upon wake-up, turn on FVR (FVREN bit of the FVRCON register)
- Manually turn on the BOR
- Switch the clock back to the preferred clock source

Note: When using the software BOR follow the steps in Methods 3 or 4 above when enabling BOR for the first time during program execution.

Affected Silicon Revisions

A6	A8	A9					
X	X						

9. Module: Enhanced Universal Synchronous Asynchronous Receiver (EUSART)

9.1 16-Bit High-Speed Asynchronous Mode

The EUSART provides unexpected operation when the 16-Bit High-Speed Asynchronous mode is selected and the Baud Rate Generator Data register values are loaded with zero ('0'). The use of this configuration for EUSART communication is not recommended. The configuration is shown below in the following table:

Configuration Bits			BRG Data Registers	
SYNC	BRG16	BRGH	SPBRGH Value	SPBRGL Value
0	1	1	00000000	00000000

Work around

Ensure that the SPBRGH or the SPBRGL register is loaded with a non-zero value.

Affected Silicon Revisions

A6	A8	A9					
X	X	X					

PIC12(L)F1822/PIC16(L)F1823

9.2 Auto-Baud Detect

When using automatic baud detection (ABDEN), on occasion, an incorrect count value can be stored at the end of auto-baud detection in the SPBRGH:SPBRGL (SPBRG) registers. The SPBRG value may be off by several counts. This condition happens sporadically when the device clock frequency drifts to a frequency where the SPBRG value oscillates between two different values. The issue is present regardless of the baud rate Configuration bit settings.

Work around

When using auto-baud, it is a good practice to always verify the obtained value of SPBRG, to ensure it remains within the application specifications. Two recommended methods are shown below.

For additional auto-baud information, see Technical Brief TB3069, *"Use of Auto-Baud for Reception of LIN Serial Communications Devices: Mid-Range and Enhanced Mid-Range"*.

EXAMPLE 2: METHOD 1 – EUSART AUTO-BAUD DETECT WORK AROUND

In firmware, define default, minimum and maximum auto-baud (SPBRG) values according to the application requirements.

For example, if the application runs at 9600 baud at 16 MHz then, the default SPBRG value would be (assuming 16-bit/Asynchronous mode) 0x67. The minimum and maximum allowed values can be calculated based on the application. In this example, a +/-5% tolerance is required, so tolerance is $0x67 * 5\% = 0x05$.

```
#define SPBRG_16BIT    *((*int)&SPBRG;                // define location for 16-bit SPBRG value

const int DEFAULT_BAUD = 0x0067;                    // Default Auto-Baud value
const int TOL = 0x05;                               // Baud Rate % tolerance
const int MIN_BAUD = DEFAULT_BAUD - TOL;             // Minimum Auto-Baud Limit
const int MAX_BAUD = DEFAULT_BAUD + TOL;             // Maximum Auto-Baud Limit
.
.
.
ABDEN = 1;                                           // Start Auto-Baud
while (ABDEN);                                       // Wait until Auto-Baud completes

if((SPBRG_16BIT > MAX_BAUD)|| (SPBRG_16BIT < MIN_BAUD))
{
    SPBRG_16BIT = DEFAULT_BAUD;                     // Compare if value is within limits
                                                    // if out of spec, use DEFAULT_BAUD
}
.
.
.                                                    // if in spec, continue using the
                                                    // Auto-Baud value in SPBRG
```

PIC12(L)F1822/PIC16(L)F1823

EXAMPLE 3: METHOD 2 – EUSART AUTO-BAUD DETECT WORK AROUND

Similar to Method 1, define default, minimum and maximum auto-baud (SPBRG) values. In firmware, compute a running average of SPBRG. If the new SPBRG value falls outside the minimum or maximum limits, then use the current running average value (Average_Baud), otherwise use the auto-baud SPBRG value and calculate a new running average.

For example, if the application runs at 9600 baud at 16 MHz then, the default SPBRG value would be (assuming 16-bit/ Asynchronous mode) 0x67. The minimum and maximum allowed values can be calculated based on the application. In this example, a +/-5% tolerance is required, so tolerance is $0x67 * 5\% = 0x05$.

```
#define SPBRG_16BIT    *((*int)&SPBRG;                // define location for 16-bit SPBRG value

const int DEFAULT_BAUD = 0x0067;                      // Default Auto-Baud value
const int TOL = 0x05;                                 // Baud Rate % tolerance
const int MIN_BAUD = DEFAULT_BAUD - TOL;              // Minimum Auto-Baud Limit
const int MAX_BAUD = DEFAULT_BAUD + TOL;              // Maximum Auto-Baud Limit

int Average_Baud;                                     // Define Average_Baud variable
int Integrator;                                       // Define Integrator variable
.
.
.
Average_Baud = DEFAULT_BAUD;                          // Set initial average Baud rate
Integrator = DEFAULT_BAUD*15;                          // The running 16 count average
.
.
.
ABDEN = 1;                                             // Start Auto-Baud
while (ABDEN);                                         // Wait until Auto-Baud completes

Integrator+ = SPBRG_16BIT;
Average_Baud = Integrator/16;
if((SPBRG_16BIT > MAX_BAUD)|| (SPBRG_16BIT < MIN_BAUD))
{
    SPBRG_16BIT = Average_Baud;                       // Check if value is within limits
                                                    // If out of spec, use previous average
}
else
{
    Integrator+ = SPBRG_16BIT;                         // If in spec, calculate the running
    Average_Baud = Integrator/16;                     // average but continue using the
    Integrator- = Average_Baud;                       // Auto-Baud value in SPBRG
}
.
.
.
```

Affected Silicon Revisions

A6	A8	A9					
X	X						

PIC12(L)F1822/PIC16(L)F1823

10. Module: Master Synchronous Serial Port (MSSP)

10.1 SPI Master mode

When the MSSP is used in SPI Master mode and the CKE bit is clear (CKE = 0), the Buffer Full (BF) bit and the MSSP Interrupt Flag (SSPIF) bit becomes set half an SCK cycle early. If the user software immediately reacts to either of the bits being set, a write collision may occur as indicated by the WCOL bit being set.

Work around

To avoid a write collision one of the following methods should be used:

Method 1: Add a software delay of one SCK period after detecting the completed transfer (the BF bit or SSPIF bit becomes set) and prior to writing to the SSP1BUF register. Verify the WCOL bit is clear after writing to SSP1BUF. If the WCOL bit is set, clear the bit in software and rewrite the SSP1BUF register.

Method 2: As part of the MSSP initialization procedure, set the CKE bit (CKE = 1).

Affected Silicon Revisions

A6	A8	A9					
X	X	X					

10.2 SPI Slave Mode

When the MSSP module is configured in SPI Slave mode with $\overline{SS}$ pin control enabled (SSPM = 0100) and the device is in Sleep mode during SPI activity, if the SPI master releases the $\overline{SS}$ line ($\overline{SS}$ goes high) before the device wakes from Sleep and updates SSP1BUF, the received data will be lost.

Work around

Method 1: The SPI master must wait a minimum of parameter SP83 (1.5 Tcy + 40 ns) after the last SCK edge and the additional wake-up time from Sleep (device dependent) before releasing the $\overline{SS}$ line.

Method 2: If both the master and slave devices have an available pin, once the slave has completed the transaction and BF or SSPIF is set, the slave could toggle an output to inform the master that the transaction is complete and that it is safe to release the $\overline{SS}$ line.

Affected Silicon Revisions

A6	A8	A9					
X	X	X					

10.3 SPI Slave Mode

When the MSSP module is configured in SPI Slave mode with $\overline{SS}$ pin control enabled (SSPM = 0100) and the device is in Sleep mode during SPI activity, if the SPI master enables $\overline{SS}$ ($\overline{SS}$ goes low) within 1 Tcy before Sleep is executed, the data written into the SSP1BUF by the slave for transmission will remain in the SSP1BUF, and the byte received by the slave will be completely discarded. The MSb of the data byte that is currently loaded into SSP1BUF will be transmitted on each of the eight SCK clocks, resulting in either a 0x00 or 0xFF address to be incorrectly transmitted. This issue typically occurs when the device wakes up from Sleep to process data and immediately goes back to Sleep during the next transmission.

Work around

The SPI slave must wait a minimum of $2.25 * Tcy$ from the time the $\overline{SS}$ line becomes active ($\overline{SS}$ goes low) before executing the Sleep command.

Affected Silicon Revisions

A6	A8	A9					
X	X	X					

10.4 SPI Slave Mode

When the MSSP module is configured with either of the Slave modes listed below and Sleep is executed during transmission, the WCOL bit is erroneously set. Although the WCOL is set, it does not cause a break in transmission or reception.

Mode 1: SPI Slave mode with $\overline{SS}$ disabled (SSPM = 0101) and CKE = 0.

Mode 2: SPI Slave mode with $\overline{SS}$ enabled (SSPM = 0100) and $\overline{SS}$ not set and then cleared before each consecutive transmission. This typically occurs during multiple byte transmissions in which the master does not release the $\overline{SS}$ line until all transmission has completed.

Work around

Method 1: The WCOL bit can be ignored since the issue does not interfere with MSSP hardware.

Method 2: Clear the SSPEN bit after each transaction, then set SSPEN before next transaction.

Affected Silicon Revisions

A6	A8	A9					
X	X	X					

PIC12(L)F1822/PIC16(L)F1823

Data Sheet Clarifications

The following typographic corrections and clarifications are to be noted for the latest version of the device data sheet (DS40001413E):

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

None.

APPENDIX A: DOCUMENT REVISION HISTORY

Rev A Document (6/2010)

Initial release of this document.

Rev B Document (11/2010)

Updated errata to the new format; Added Silicon Revision A8; Added Module 5: Clock Switching.

Rev C Document (03/2011)

Added Modules 6, 7 and 8.

Rev D Document (02/2012)

Updated Table 1; Added Modules 1.2, 1.3 and 1.4; Added Module 9, EUSART; Other minor corrections.

Data Sheet Clarifications: Added Module 1, Oscillator.

Rev E Document (10/2012)

Added MPLAB X IDE; Added Silicon Revision A9; Updated Table 2; Updated Module 8, BOR; Other minor corrections.

Data Sheet Clarifications: Removed Module 1, Oscillator.

Rev F Document (12/2014)

Added Module 10, MSSP; Other minor corrections.

Rev G Document (07/2015)

Added Modules 10.2 to 10.4 (MSSP); Removed Module 1.2 (Oscillator: HFINTOSC Ready/Stable Bit).

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.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, FlashFlex, flexPWR, JukeBlox, KEELOQ, KEELOQ logo, Klear, LANCheck, MediaLB, MOST, MOST logo, MPLAB, OptoLyzer, PIC, PICSTART, PIC³² logo, RightTouch, SpyNIC, SST, SST Logo, SuperFlash and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

The Embedded Control Solutions Company and mTouch are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, BodyCom, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, ECAN, In-Circuit Serial Programming, ICSP, Inter-Chip Connectivity, KlearNet, KlearNet logo, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, RightTouch logo, REAL ICE, SQI, Serial Quad I/O, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

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

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

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

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

© 2010-2015, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN: 978-1-63277-647-1

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949 ==

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

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

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

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

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

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

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

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110

Canada - Toronto
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: 852-2943-5100
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 - Dongguan
Tel: 86-769-8702-9880

China - Hangzhou
Tel: 86-571-8792-8115
Fax: 86-571-8792-8116

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

ASIA/PACIFIC

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

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

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: 91-20-3019-1500

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

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 - Dusseldorf
Tel: 49-2129-3766400

Germany - Karlsruhe
Tel: 49-721-625370

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

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

Italy - Venice
Tel: 39-049-7625286

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

Poland - Warsaw
Tel: 48-22-3325737

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

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820