PIC18F2455/2550/4455/4550 Family Silicon Errata and Data Sheet Clarification

The PIC18F2455/2550/4455/4550 family devices that you have received conform functionally to the current Device Data Sheet (DS39632**D**), 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 PIC18F2455/2550/4455/4550 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 (**B7**).

Data Sheet clarifications and corrections start on page 18, following the discussion of silicon issues.

The silicon revision level can be identified using the current version of MPLAB® IDE and Microchip's programmers, debuggers, and emulation tools, which are available at the Microchip corporate web site (www.microchip.com).

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 DEVREV values for the various PIC18F2455/2550/4455/4550 silicon revisions are shown in Table 1.

TABLE 1: SILICON DEVREV VALUES

Part Number	Device ID(1)	Revision ID for Silicon Revision(2)				
		A3	B4	B5	B6	B7
PIC18F2455	126Xh	2h	4h	5h	6h	7h
PIC18F2550	124Xh					
PIC18F4455	122Xh					
PIC18F4550	120Xh					

Note 1: The Device IDs (DEVID and DEVREV) are located at the last two implemented addresses of configuration memory space. They are shown in hexadecimal in the format "DEVID DEVREV".

2: Refer to the *"PIC18F2XXX/4XXX Family Flash Microcontroller Programming Specification"* (DS39622) 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)				
				A3	B4	B5	B6	B7
EUSART	Nine-Bit mode	1.	Back-to-back timing transmit corruption	X				
Timer1/3	Sixteen-Bit mode	2.	Write to TMR1H/3H may lengthen duration	X				
MSSP	Slave Transmit	3.	Slave transmit, 10 TCY not blocked	X				
Interrupts	Two-Cycle Instructions	4.	Special considerations for interrupt context save	X				
ECCP	ECCPASE Bit	5.	Do not use bit-wise operations on ECCPASE	X				
ECCP	Auto-Restart	6.	Immediate restart upon shutdown source removal	X				
ECCP	Special Event Trigger	7.	Compare mode, Special Event Trigger not like PIC18F452	X				
ADC	Offset Error	8.	Offset greater than data sheet	X				
BOR	VBOR	9.	'11' setting below minimum operating voltage	X				
USB	SIE IN Endpoint	10.	PKTDIS set, NAK, four extra bytes sent	X				
PORTD	RDPU Control Bit	11.	Access to PORTE causes RDPU to clear	X				
MSSP	Slave Addressing	12.	I^2C™ slave address masking not implemented	X				
EUSART	BAUDCON Register	13.	RXDTP, TXCKP do not exist	X				
USB	Ping-Pong Buffer	14.	Ping-Pong mode '11' not supported	X				
MSSP	SPI Slave mode	15.	SPI slave write collision	X				
MSSP	SPI mode	16.	SPI SDO output may change	X				
MSSP	I^2C™ mode	17.	I^2C pins may not initialize properly	X				
MSSP	SPI mode	18.	In SPI mode, do not poll BF bit	X				
EUSART	Async mode	19.	Extra zero bytes in Async mode	X				
EUSART	Async mode	20.	Data corruption in 9-bit Async full-duplex	X				
EUSART	Receive Buffer	21.	RCREG not valid for subsequent reads	X				
EUSART	Auto Wake-up	22.	WUE bit not clearing promptly	X				
Timer1/3	16-Bit Async	23.	16-Bit Async mode, TMR1H/3H not updated	X				
Reset	RAM	24.	Asynch Reset can alter RAM	X				
ECCP	PWM mode	25.	Dead-band delay incorrect	X	X	X	X	X
MSSP	SPI Master mode	26.	SPI master, write collision for FOSC/64 and Timer2/2	X	X	X	X	X
MSSP	I^2C mode	27.	Unaddressed I^2C slave node may respond	X				
EUSART	Auto-Baud	28.	Auto-baud sometimes does not work	X				
ADC	2 TOSC or RC Clock	29.	EIL, EDL, not meeting data sheet at 511/512	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)				
				A3	B4	B5	B6	B7
BOR	HLVD or USB	30.	Clearing SBOREN can cause BOR	X				
MSSP	I^2C mode	31.	I^2C baud rate not meeting formula	X	X	X		
MSSP	SPI slave	32.	SPI slave not meeting timing parameter 70		X	X	X	X
Timer1/3	Eight-Bit Async	33.	Need delay between consecutive writes		X			
BOR	Threshold	34.	Certain conditions move BOR threshold		X	X	X	
BOR	Threshold	35.	Table reads can move BOR threshold					X
EUSART	Synch Master	36.	Synchronous master mode, DT data changes		X	X		
MSSP	SPI Slave	37.	SPI Slave needs a series resistor		X	X	X	
MSSP	SPI modes	38.	SPI modes, incoming data not received if BF set				X	X
MSSP	I^2C Slave	39.	With I^2C slave reception, need to read data promptly	X	X	X	X	X
MSSP	I^2C Master	40.	In I^2C Master mode, narrow clock width upon slave clock stretch				X	X
EUSART	Interrupts	41.	If interrupts are enabled, 2 T_{CY} delay needed after re-enabling the module		X	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 (**B7**).

1. Module: EUSART

When performing back-to-back transmission in 9-bit mode (TX9D bit in the TXSTA register is set), an ongoing transmission's timing can be corrupted if the TX9D bit (for the next transmission) is not written immediately following the setting of TXIF. This is because any write to the TXSTA register results in a reset of the Baud Rate Generator which will effect any ongoing transmission.

Work around

Load TX9D just after TXIF is set, either by polling TXIF or by writing TX9D at the beginning of the Interrupt Service Routine, or only write to TX9D when a transmission is not in progress (TRMT = 1).

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

2. Module: Timer1/3

When Timer1/3 is operating in 16-bit mode and the prescale setting is not 1:1, a write to the TMR1H/TMR3H Buffer registers may lengthen the duration of the period between the increments of the timer for the period in which TMR1H/TMR3H were written.

Work around

Either of two work arounds can be used:

- Stop Timer1/Timer3 before writing the TMR1H/TMR3H registers
- Write TMR1L/TMR3L immediately after writing TMR1H/TMR3H

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

3. Module: MSSP

In Slave Transmit mode, when a transmission is initiated, the SSPBUF register may be written for up to 10 TCY before additional writes are blocked. The data transfer may be corrupted if SSPBUF is written during this time.

The WCOL bit is set any time an SSPBUF write occurs during a transfer.

Work around

Avoid writing SSPBUF until the data transfer is complete, indicated by the setting of the SSPIF bit (PIR1<3>).

To ensure any potential transfer in progress is not corrupted, verify that the WCOL bit (SSPCON1<7>) is clear after writing SSPBUF.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

4. Module: Interrupts

If an interrupt occurs during a two-cycle instruction that modifies the STATUS, BSR or WREG register, the unmodified value of the register will be saved to the corresponding Fast Return (Shadow) register and upon a fast return from the interrupt, the unmodified value will be restored to the STATUS, BSR or WREG register.

For example, if a high-priority interrupt occurs during the instruction, "MOVFF TEMP, WREG", the MOVFF instruction will be completed and WREG will be loaded with the value of TEMP before branching to ISR. However, the previous value of WREG will be saved to the Fast Return register during ISR branching. Upon return from the interrupt with a fast return, the previous value of WREG in the Fast Return register will be written to WREG.

This results in WREG containing the value it had before execution of "MOVFF TEMP, WREG".

Affected instructions are:

MOVFF Fs, Fd
Where Fd is WREG, BSR or STATUS

MOVSF Zs, Fd
Where Fd is WREG, BSR or STATUS

MOVSS [Zs], [Zd]
Where the destination is WREG, BSR or STATUS

Work around

Alternative work arounds are available for the Assembly Language Programming and the C Programming Language:

Assembly Language Programming

Either of two work arounds can be used:

- If any two-cycle instruction is used to modify the WREG, BSR or STATUS register, do not use the RETFIE FAST instruction to return from the interrupt. Instead, save/restore WREG, BSR and STATUS via software per Example 9-1 in the Device Data Sheet.

  Alternatively, in the case of MOVFF, use the MOVF instruction to write to WREG instead. For example, use:

```
MOVF   TEMP, W
MOVWF  BSR
```

  instead of: MOVFF TEMP, BSR.

- As another alternative, use the work around shown in Example 1. This example overwrites the Fast Return register by making a dummy call to Foo with the fast option in the high-priority service routine.

EXAMPLE 1: INTERRUPT WORK AROUND – ASSEMBLY

```
ISR @ 0x0008
  CALL Foo, FAST; store current value of WREG, BSR, STATUS for a second time
Foo:
  POP         ; clears return address of Foo call
  :           ; insert high priority ISR code here
  :
  RETFIEFAST
```

C Programming Language

The exact work around depends on the compiler in use. Please refer to your C compiler documentation for details.

If using the Microchip MPLAB® C18 C Compiler, define both high and low-priority interrupt handler functions as "low priority" by using the `pragma interruptlow` directive. This directive instructs the compiler to not use the `RETFIE FAST` instruction. If the proper high-priority interrupt bit is set in the IPRx register, the interrupt is treated as high priority in spite of the `pragma interruptlow` directive.

The code segment shown in Example 2 demonstrates the work around using the C18 compiler.

EXAMPLE 2: INTERRUPT WORK AROUND – C

```
#pragma interruptlow MyLowISR
void MyLowISR(void)
{
   // Handle low priority interrupts.
}

// Although MyHighISR is a high priority interrupt, use interruptlow pragma so that
// the compiler will not use retfie FAST.

#pragma interruptlow MyHighISR
void MyHighISR(void)
{
   // Handle high priority interrupts.
}

#pragma code highVector=0x08
void HighVector (void)
{
   _asm goto MyHighISR _endasm
}
#pragma code /* return to default code section */

#pragma code lowVector=0x18
void LowVector (void)
{
   _asm goto MyLowISR _endasm
}
#pragma code /* return to default code section */
```

An optimized, C18 version of the code is provided in Example 3. This example illustrates how to reduce the instruction cycle count from 10 cycles to 3.

EXAMPLE 3: INTERRUPT WORK AROUND – OPTIMIZED C18

```
#pragma code high_vector_section=0x8
void high_vector (void)
{
  _asm
    CALL high_vector_branch, 1
  _endasm
}

void high_vector_branch (void)
{
  _asm
    POP
    GOTO high_isr
  _endasm
}

#pragma interrupt high_isr
void high_isr (void)
{
  ...
}
```

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

5. Module: ECCP

When monitoring a shutdown condition using a bit test on the ECCPASE bit (ECCP1AS<7>) or performing a bit operation on the ECCPASE bit, the device may produce unexpected results.

Work around

Before performing a bit test or bit operation on the ECCPASE bit, copy the ECCP1AS register to the working register and perform the operation there.

By avoiding these operations on the ECCPASE bit in the ECCP1AS register, the module will operate normally.

In Example 4, ECCPASE bit operations are performed on the W register.

EXAMPLE 4: ECCPASE OPERATION

```
MOVF    ECCP1AS, W
BTFSC   WREG, ECCPASE
BRA     SHUTDOWN_ROUTINE
```

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

6. Module: ECCP

When the CCP1 auto-shutdown feature is configured for automatic restart (PRSEN bit (ECCP1DEL<7>) = 1), the pulse terminates immediately in a shutdown event. In addition, the pulse may restart within the period, if the shutdown condition expires. This may result in the generation of short pulses on the PWM output(s).

Work around

Configure the auto-shutdown for software restart by clearing the PRSEN bit (ECCP1DEL<7>). The PWM can be re-enabled, after the shutdown condition expires, by clearing the ECCPASE bit (ECCP1AS<7>).

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

7. Module: ECCP

When operating either Timer1 or Timer3 as a counter with a prescale value other than 1:1 and operating the ECCP in Compare mode with the Special Event Trigger (CCP1CON bits, CCP1M<3:0> = 1011), the Special Event Trigger Reset of the timer occurs as soon as there is a match between TMRxH:TMRxL and CCPR1H:CCPR1L.

This differs from the PIC18F452, where the Special Event Trigger Reset of the timer occurs on the next rollover of the prescale counter after the match between TMRxH:TMRxL and CCPR1H:CCPR1L.

Work around

To achieve the same timer Reset period on the PIC18F4550 family as the PIC18F452 family for a given clock source, add 1 to the value in CCPR1H:CCPR1L. In other words, if CCPR1H:CCPR1L = x for the PIC18F452, to achieve the same Reset period on the PIC18F4550 family, CCPR1H:CCPR1L = x + 1, where the prescale is 1, 2, 4 or 8 depending on the T1CKPS<1:0> bit values.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

8. Module: A/D

The A/D offset is greater than the specified limit in Table 28-8 of the Device Data Sheet. The updated conditions and limits are shown in **bold** text in Table 28-8.

Work around

Any of three work arounds may be used.

- Configure the A/D to use the VREF+ and VREF- pins for the voltage references. This is done by setting the VCFG<1:0> bits (ADCON1<5:4>).
- Perform a conversion on a known voltage reference voltage and adjust the A/D result in software.
- Increase system clock speed to 48 MHz and adjust A/D settings accordingly. Higher system clock frequencies decrease offset error.

TABLE 28-8: A/D CONVERTER CHARACTERISTICS:PIC18F2455/2550/4455/4550 (INDUSTRIAL) PIC18LF2455/2550/4455/4550 (INDUSTRIAL)

Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
A06A	**EOFF**	**Offset Error**	**—**	**—**	**<±2.0**	**LSb**	**VREF = VREF+ and VREF-**
A06	EOFF	Offset Error	—	—	**<±3.5**	LSb	**VREF = VSS and VDD**

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

9. Module: DC Characteristics (BOR)

When the trip point for BORV<1:0> = 11, the values for parameter D005 (VBOR) in **Section 28.1 "DC Characteristics"** of the Device Data Sheet are not applicable as the device may reset below the minimum operating voltage for the device.

Work around

None.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

10. Module: USB

When an IN endpoint is owned by USB SIE and the UCON register's PKTDIS bit is set, if a USB NAK event occurs on the IN endpoint before the PKTDIS bit is clear, then after the PKTDIS is clear, the pending IN endpoint will send out more bytes than expected. For example, if configured to send out 8 bytes, the SIE would actually send out 12 bytes of data.

Work around

The PKTDIS bit is set when a USB control transfer setup packet is received. Clear this bit as soon as possible, particularly before turning over any IN endpoint ownership to the SIE.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

11. Module: PORTD

Each of the PORTD pins has a weak internal pull-up. A single control bit, RDPU (PORTE<7>), can turn on all the pull-ups. After the pull-up has been enabled (PORTE<7> = 1), any access to the PORTE register would cause the RDPU control bit to clear, except those accesses that write a '1' to PORTE<7>.

Work around

Reassert RDPU after every access to the PORTE register, except those accesses that write a '1' to PORTE<7> or use external pull-ups.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

12. Module: MSSP

The I²C™ slave address masking feature is not supported. Therefore, SSPCON2 register bits, ADMSK<5:1>, do not exist in I²C Slave mode.

Work around

None.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

13. Module: EUSART

In the BAUDCON register, bits, RXDTP and TXCKP, do not exist. BAUDCON bit 4 is defined instead as SCKP and has the following definition:

bit 4 **SCKP**: Synchronous Clock Polarity Select bit
Asynchronous mode:
Unused in this mode.
Synchronous mode:
1 = Idle state for clock (CK) is a high level
0 = Idle state for clock (CK) is a low level

Work around

None.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

14. Module: USB

The Ping-Pong Buffer mode in which the ping-pong buffers are enabled for Endpoints 1 to 15 (UCFG (PPB<1:0) = 11) is not supported.

Work around

Use other Ping-Pong Buffer modes.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

15. Module: MSSP

The MSSP configured in SPI Slave mode will generate a write collision if SSPBUF is updated and the previous SSPBUF contents have not been transferred to the shift register.

Re-initializing the MSSP by clearing and setting the SSPEN bit (SSPCON1<5>), prior to rewriting SSPBUF, will not prevent the error condition.

Work around

Prior to updating the SSPBUF register with a new value, verify whether the previous contents were transferred by reading the BF bit (SSPSTAT<0>). If the previous byte has not been transferred, update SSPBUF and clear the WCOL bit (SSPCON1<7>) if necessary.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

16. Module: MSSP

In SPI mode, the SDO output may change after the inactive clock edge of the bit '0' output. This may affect some SPI components that read data over 300 ns after the inactive edge of SCK.

Work around

None.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

17. Module: MSSP

It has been observed that following a Power-on Reset, the I^2C mode may not initialize properly by just configuring the SCL and SDA pins as either inputs or outputs. This has only been seen in a few unique system environments.

A test of a statistically significant sample of pre-production systems, across the voltage and current range of the application's power supply, should indicate if a system is susceptible to this issue.

Work around

Before configuring the module for I^2C operation:

1. Configure the SCL and SDA pins as outputs by clearing their corresponding TRIS bits.
2. Force SCL and SDA low by clearing the corresponding LAT bits.
3. While keeping the LAT bits clear, configure SCL and SDA as inputs by setting their TRIS bits.

Once this is done, use the SSPCON1 and SSPCON2 registers to configure the proper I^2C mode as before.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

18. Module: MSSP

When the MSSP is configured for SPI mode, the Buffer Full bit, BF (SSPSTAT<0>), should not be polled in software to determine when the transfer is complete.

Work around

Copy the SSPSTAT register into a variable and perform the bit test on the variable. In Example 5, SSPSTAT is copied into the working register where the bit test is performed.

EXAMPLE 5: SSPSTAT WORK AROUND

```
loop_MSB:
    MOVF   SSPSTAT, W
    BTFSS  WREG, BF
    BRA    loop_MSB
```

A second option is to poll the Master Synchronous Serial Port Interrupt Flag bit, SSPIF (PIR1<3>). This bit can be polled and will set when the transfer is complete.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

19. Module: EUSART

In rare situations, one or more extra zero bytes have been observed in a packet transmitted by the module operating in Asynchronous mode. The actual data is not lost or corrupted; only unwanted (extra) zero bytes are observed in the packet.

This situation has only been observed when the contents of the transmit buffer, TXREG, are transferred to the TSR during the transmission of a Stop bit. For this to occur, three things must happen in the same instruction cycle:

- TXREG is written to
- The baud rate counter overflows (at the end of the bit period)
- A Stop bit is being transmitted (shifted out of TSR)

Work around

If possible, do not use the module's double-buffer capability. Instead, load the TXREG register when the TRMT bit (TXSTA<1>) is set, indicating the TSR is empty.

If double-buffering is used and back-to-back transmission is performed, load TXREG immediately after TXIF is set, or wait 1 bit time after TXIF is set. Both solutions prevent writing TXREG while a Stop bit is transmitted. The TXIF bit is set at the beginning of the Stop bit transmission.

If transmission is intermittent, do one of the following:

- Wait for the TRMT bit to be set before loading TXREG
- Use a free timer resource to time the baud period.

  Set up the timer to overflow at the end of a Stop bit, then start the timer when you load the TXREG. Do not load the TXREG when timer is about to overflow.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

20. Module: EUSART

In 9-Bit Asynchronous Full-Duplex Receive mode, the received data may be corrupted if the TX9D bit (TXSTA<0>) is not modified immediately after the RCIDL bit (BAUDCON<6>) is set.

Work around

Write to TX9D only when a reception is not in progress (RCIDL = 1). Since there is no interrupt associated with RCIDL, it must be polled in software to determine when TX9D can be updated.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

21. Module: EUSART

After the last received byte has been read from the EUSART receive buffer (RCREG), the value is no longer valid for subsequent read operations.

Work around

The RCREG register should only be read once for each byte received. After each byte is received from the EUSART, store the byte in a user variable.

To determine when a byte is available to read from RCREG, poll the RCIDL bit (BAUDCON<6>) for a low-to-high transition, or use the EUSART Receive Interrupt Flag, RCIF (PIR1<5>).

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

22. Module: EUSART

With the auto-wake-up option enabled by setting the WUE bit (BAUDCON<1>), the RCIF (PIR1<5>) bit will become set on a high-to-low transition on the RX pin. However, the WUE bit may not clear within 1 TCY of a low-to-high transition on RX.

While the WUE bit is set, reading the Receive Buffer (RCREG) will not clear the RCIF interrupt flag. Therefore, the first opportunity to automatically clear RCIF by reading RCREG may take longer than expected.

Note:	RCIF can only be cleared by reading RCREG.

Work around

Either of these work arounds can be used:

- Clear the WUE bit in software, after the wake-up event has occurred and prior to reading the receive buffer (RCREG)
- Poll the WUE bit and read RCREG, after the WUE bit is automatically cleared

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

23. Module: Timer1/3

In 16-Bit Asynchronous Counter mode (with or without use of the Timer1 oscillator), the TMR1H and TMR3H buffers do not update when TMRxL is read.

This issue only affects reading the TMRxH registers. The timers increment and set the interrupt flags as expected. The timer registers can also be written as expected.

Work around

1. Use 8-bit mode by clearing the RD16 bit (T1CON<7>).
2. Use the internal clock synchronization option by clearing the $\overline{\text{T1SYNC}}$ bit (T1CON<2>).

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

24. Module: Reset

The indicated version of silicon does not support the functionality described in Note 1 of parameter D002 in **Section 28.1 "DC Characteristics: Supply Voltage"** of the data sheet. The RAM content may be altered during a Reset event if the following conditions are met.

- Device is accessing RAM
- Asynchronous Reset (WDT, BOR or $\overline{\text{MCLR}}$) occurs when a write operation is being executed (start of a Q4 cycle) or if a RESET instruction is executed and immediately followed by a RETURN instruction

Work around

None.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

25. Module: ECCP (PWM Mode)

When configured for half-bridge operation with dead band (CCPxCON<7:6> = 10), the PWM output may be corrupted for certain values of the PWM duty cycle. This can occur when these additional criteria are also met:

- A non-zero dead-band delay is specified (PDC<6:0> > 0)
- The duty cycle has a value of 0 through 3, or 4n + 3 (n ≥ 1)

Work around

None.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X	X	X	X	X

26. Module: MSSP

With MSSP in SPI Master mode, FOSC/64 or Timer2/2 clock rate and CKE = 0, a write collision may occur if SSPBUF is loaded immediately after the transfer is complete. A delay may be required after the MSSP Interrupt Flag bit (SSPIF) is set or the Buffer Full bit (BF) is set – before writing SSPBUF. If the delay is insufficiently short, a write collision may occur as indicated by the WCOL bit being set.

Work around

Add a software delay of one SCK period after detecting the completed transfer and prior to updating the SSPBUF contents.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X	X	X	X	X

27. Module: MSSP

In an I^2C system with multiple slave nodes, an unaddressed slave may respond to bus activity when data on the bus matches its address. The first occurrence will set the BF bit. The second occurrence will set the BF and SSPOV bits. In both situations, the SSPIF bit is not set and an interrupt will not occur. The device will vector to the Interrupt Service Routine only if the interrupt is enabled and an address match occurs.

Work around

The I^2C slave must clear the SSPOV bit after each I^2C event to maintain normal operation.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

28. Module: EUSART

The EUSART auto-baud feature may periodically measure the incoming baud rate incorrectly. The rate of incorrect baud-rate measurements will depend on the frequency of the incoming synchronization byte and the system clock frequency.

Work around

None.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

29. Module: A/D

When the A/D clock source is selected as 2 TOSC or RC (when ADCS<2:0> = 000 or x11), in extremely rare cases, the EIL (Integral Linearity Error) and EDL (Differential Linearity Error) may exceed the data sheet specification at codes 511 and 512 only.

Work around

Select a different A/D clock source (4 TOSC, 8 TOSC, 16 TOSC, 32 TOSC, 64 TOSC) and avoid selecting the 2 TOSC or RC modes.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X	X	X		

30. Module: Resets (BOR)

If either the HLVD or USB modules are enabled, clearing the SBOREN bit (RCON<6>) when the software controlled BOR feature is enabled (BOREN<1:0> = 01) may cause a Brown-out Reset (BOR) event.

Work around

Before clearing the SBOREN bit, temporarily disable the HLVD and USB modules.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X				

31. Module: MSSP

When operated in I²C™ Master mode, the I²C baud rate may be somewhat slower than predicted by the following formula:

$$I^2C \text{ Master mode, clock} = \frac{F_{OSC}}{4 \bullet (SSPADD + 1)}$$

Work around

If the target application is sensitive to the baud rate and requires more precision, the SSPADD value can be adjusted to compensate.

If this work around is going to be used, it is recommended that the firmware first check the Revision ID by reading the DEVID1 value at address, 3FFFFEh. Silicon revisions, B6 and B7, will match the I²C baud rate predicted by the given formula.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X	X	X		

32. Module: MSSP

In SPI Slave mode with slave select enabled (SSPM<3:0> = 0100), the minimum time between the falling edge of the $\overline{SS}$ pin and first SCK edge is greater than specified in parameter 70 in Table 28-17 and Table 28-18. The updated specification is shown in bold in Table 3.

The minimum time between $\overline{SS}$ pin low and an SSPBUF write is also 3 TCY. If the falling edge of the $\overline{SS}$ pin occurs greater than 3 TCY, before the first SCK edge or loading SSPBUF, the peripheral will function correctly. Also, if SSPBUF is written prior to the $\overline{SS}$ pin going low, the peripheral will function correctly.

TABLE 3: EXAMPLE SPI MODE REQUIREMENTS (SLAVE MODE TIMING)

Param No.	Symbol	Characteristic	Min	Max	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SS}$ ↓ to SCK ↓ or SCK ↑ Input	**3 TCY**	—	ns	

Work around

None.

Affected Silicon Revisions

A3	B4	B5	B6	B7
	X	X	X	X

33. Module: Timer1/3

For Timer1 or Timer3, if the TMRxH and TMRxL registers are written to in consecutive instruction cycles, the timer may not be updated with the correct value if it is configured for externally clocked, 8-Bit Asynchronous mode (T1CON<7:0> or T3CON<7:0> = 0xxx x111).

For the purposes of this issue, instructions that directly affect the contents of the Timer registers are considered to be writes. This includes CLRF, SETF and MOVF instructions.

Work around

Insert a delay of one or more instruction cycles between writes to TMRxH and TMRxL. This delay can be a NOP, or any instruction that does not access the Timer registers (Example 6).

EXAMPLE 6: TIMER1/3 – CONSECUTIVE WRITES

```
CLRF    TMR1H
MOVLW   T1Offset  ; 1 Tcy delay
MOVWF   TMR1L
```

Affected Silicon Revisions

A3	B4	B5	B6	B7
	X			

34. Module: Resets (BOR)

Certain operating conditions can move the effective Brown-out Reset (BOR) threshold outside of the range specified in the electrical characteristics of the Device Data Sheet (parameter D005).

The BOR threshold has been observed to increase with high device operating frequencies, some table read operations and heavy loading on the USB voltage regulator. When all of these conditions are present, BOR has been observed with VDD 20 percent higher than the VBOR value specified for a given BORV<1:0> setting.

The BOR threshold may decrease under other conditions, such as during Sleep, where it may not occur until VDD is 120 mV below the specified minimums.

Work around

None.

Affected Silicon Revisions

A3	B4	B5	B6	B7
	X	X	X	

35. Module: Resets (BOR)

Certain operating conditions can move the effective Brown-out Reset (BOR) threshold outside of the range specified in the electrical characteristics of the Device Data Sheet (parameter D005).

The BOR threshold has been observed to increase with some table read operations. BOR has been observed with 7 percent higher VDD than the VBOR value specified for a given BORV<1:0> setting.

Work around

None.

Affected Silicon Revisions

A3	B4	B5	B6	B7
				X

36. Module: EUSART

In Synchronous Master mode, while transmitting the Most Significant data bit, the data line (DT) may change state before the bit finishes transmitting. If the receiver samples the data line later than 0.5 bit times + 1.5 TCY (of the master) after the starting edge of the MSb, the bit may be read incorrectly.

Work around

None.

Affected Silicon Revisions

A3	B4	B5	B6	B7
	X	X		

37. Module: MSSP (SPI Slave)

If configured in SPI Slave mode, the MSSP may not successfully recognize data packets generated by an external master processor. This applies to all SPI Slave modes (CKE/CKP = 1 or 0), whether or not slave select is enabled (SSPM<3:0> = 010x).

Work around

Insert a series resistor between the SPI master Serial Data Out (SDO) and the corresponding SPI slave Serial Data In (SDI) input line of the microcontroller. The required value for the resistor varies with the application system's characteristics and the process variations between the microcontrollers.

Experimentation and thorough testing are encouraged.

Affected Silicon Revisions

A3	B4	B5	B6	B7
	X	X	X	

38. Module: MSSP

If the application firmware is expecting to receive valid data, in either SPI Slave or Master mode, the firmware must read from the SSPBUF register before writing the next byte to transmit to SSPBUF.

If the firmware does not read from SSPBUF, the BF bit (SSPSTAT<0>) can still be set from the previous transaction. If the BF bit is set, the incoming data byte is blocked from transferring from the SSPSR Shift register to the SSPBUF register. If the firmware then reads from SSPBUF, the data read will not match the data most recently received on the SDI pin.

In earlier silicon revisions (A3, B4 and B5), incoming data bytes received on the SDI pin are always transferred from SSPSR to SSPBUF, regardless of the state of the BF bit.

Work around

If the firmware expects to receive valid data, always clear the BF bit by reading from SSPBUF prior to writing to SSPBUF, even when the current data in SSPBUF is not important. Sample work around code, suitable for all silicon revisions, is given in Example 7 (Assembly language) and Example 8 (C language).

Affected Silicon Revisions

A3	B4	B5	B6	B7
			X	X

EXAMPLE 7: SAMPLE ASSEMBLY CODE FOR TRANSFERRING SPI DATA

```
WriteSPI:
BCF    PIR1, SSPIF
MOVF   SSPBUF, w     ;Perform read, even if the data in SSPBUF is not important
MOVLW  0xA5          ;In this example, let's send "0xA5" to the other SPI device.
MOVWF  SSPBUF

WaitXmitComplete:
BTFSS  PIR1, SSPIF
BRA    WaitXmitComplete

MOVF   SSPBUF, w     ;The data received should be valid.
```

EXAMPLE 8: SAMPLE C CODE FOR TRANSFERRING SPI DATA

```
unsigned char WriteSPI(unsigned char ByteToSend)
{
   unsigned char TempVariable;

   PIR1bits.SSPIF = 0;
   TempVariable = SSPBUF;     // Reads from SSPBUF, ensures BF bit is clear before
   SSPBUF = ByteToSend;       // sending the next byte.

   while(!PIR1bits.SSPIF);    // Wait until the transmission is complete.
   return SSPBUF;             // The data received should be valid.
}
```

39. Module: MSSP (I^2C™ Slave)

When configured for I^2C™ slave reception, the MSSP module may not receive the correct data, in extremely rare cases. This occurs only if the Serial Receive/Transmit Buffer Register (SSPBUF) is not read after the SSPIF interrupt (PIR1<3>) has occurred, but before the first rising clock edge of the next byte being received.

Work around

The issue can be resolved in either of these ways:

- Prior to the I^2C slave reception, enable the clock stretching feature.

  This is done by setting the SEN bit (SSPCON2<0>).
- Each time the SSPIF is set, read the SSPBUF before the first rising clock edge of the next byte being received.

Affected Silicon Revisions

A3	B4	B5	B6	B7
X	X	X	X	X

40. Module: MSSP (I^2C™ Master)

When in I^2C Master mode, if the slave performs clock stretching, the first clock pulse after the slave releases the SCL line may be narrower than the configured clock width. This may result in the slave missing the first clock in the next transmission/ reception.

Work around

The clock pulse will be the normal width if the slave does not perform clock stretching.

Affected Silicon Revisions

A3	B4	B5	B6	B7
			X	X

41. Module: EUSART

In rare situations, when interrupts are enabled, unexpected results may occur if:

- The EUSART is disabled (the SPEN bit, RCSTAx<7> = 0)
- The EUSART is re-enabled (RCSTAx<7> = 1)
- A two-cycle instruction is executed immediately after setting SPEN = 1

Work around

Add a 2 TCY delay after any instruction that re-enables the EUSART module (sets SPEN = 1). See Example 9.

EXAMPLE 9: RE-ENABLING A EUSART MODULE

```
;Initial conditions: SPEN = 0 (module disabled)
;To re-enable the module:
;Re-Initialize TXSTAx, BAUDCONx, SPBRGx, SPBRGHx registers (if needed)
;Re-Initialize RCSTAx register (if needed), but do not set SPEN = 1 yet

;Now enable the module, but add a 2-Tcy delay before executing any two-cycle
;instructions
bsf    RCSTA1, SPEN  ;or RCSTA2 if EUSART2
nop    ;1 Tcy delay
nop    ;1 Tcy delay (two total)

;CPU may now execute 2 cycle instructions
```

Affected Silicon Revisions

A3	B4	B5	B6	B7
	X	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 (DS39632**D**):

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

1. Module: Timer1

The Real-Time Clock application example, cited in **Section 12.6 “Using Timer1 as a Real-Time Clock,”** has changed.

Example 12-1 is corrected to the following example.

EXAMPLE 12-1: IMPLEMENTING A REAL-TIME CLOCK USING A TIMER1 INTERRUPT SERVICE

```
RTCinit
            MOVLW   80h             ; Preload TMR1 register pair
            MOVWF   TMR1H           ; for 1 second overflow
            CLRF    TMR1L
            MOVLW   b’00001111’     ; Configure for external clock,
            MOVWF   T1CON           ; Asynchronous operation, external oscillator
            CLRF    secs            ; Initialize timekeeping registers
            CLRF    mins            ;
            MOVLW   .12
            MOVWF   hours
            BSF     PIE1, TMR1IE    ; Enable Timer1 interrupt
            RETURN
RTCisr
                                    ; Insert the next 4 lines of code when TMR1
                                    ; can not be reliably updated before clock pulse goes low
            BTFSC   TMR1L,0         ; wait for TMR1L to become clear
            BRA     $-2             ; (may already be clear)
            BTFSS   TMR1L,0         ; wait for TMR1L to become set
            BRA     $-2             ; TMR1 has just incremented
                                    ; If TMR1 update can be completed before clock pulse goes low
                                    ; Start ISR here
            BSF     TMR1H, 7        ; Preload for 1 sec overflow
            BCF     PIR1, TMR1IF    ; Clear interrupt flag
            INCF    secs, F         ; Increment seconds
            MOVLW   .59             ; 60 seconds elapsed?
            CPFSGT  secs
            RETURN                  ; No, done
            CLRF    secs            ; Clear seconds
            INCF    mins, F         ; Increment minutes
            MOVLW   .59             ; 60 minutes elapsed?
            CPFSGT  mins
            RETURN                  ; No, done
            CLRF    mins            ; clear minutes
            INCF    hours, F        ; Increment hours
            MOVLW   .23             ; 24 hours elapsed?
            CPFSGT  hours
            RETURN                  ; No, done
            CLRF    hours           ; Reset hours
            RETURN                  ; Done
```

2. Module: Timer1

The following text, **Section 12.7 "Considerations in Asynchronous Counter Mode"**, is new. It defines the proper method to update the TMR1 registers in Asynchronous mode.

Section 12.7 is located after **Section 12.6 "Using Timer1 as a Real-Time Clock"** in the data sheet.

12.7 Considerations in Asynchronous Counter Mode

Following a Timer1 interrupt and an update to the TMR1 registers, the Timer1 module uses a falling edge on its clock source to trigger the next register update on the rising edge. If the update is completed after the clock input has fallen, the next rising edge will not be counted.

If the application can reliably update TMR1 before the timer input goes low, no additional action is needed. Otherwise, an adjusted update can be performed following a later Timer1 increment. This can be done by monitoring TMR1L within the interrupt routine until it increments, and then updating the TMR1H:TMR1L register pair while the clock is low, or one-half of the period of the clock source. Assuming that Timer1 is being used as a Real-Time Clock, the clock source is a 32.768 kHz crystal oscillator; in this case, one-half period of the clock is 15.25 μs.

The Real-Time Clock application code in Example 12-1 shows a typical ISR for Timer1, as well as the optional code required if the update cannot be done reliably within the required interval.

(Example 12-1 appears on page 18 of this errata.)

3. Module: Universal Serial Bus (USB)

In **Section 17.2.2.8 "Internal Regulator,"** the following corrections should be noted (changes and added text appear in **bold** for the purposes of this errata):

- In the second paragraph, the first sentence is corrected to read, "The regulator is **dis**abled by default and can be **en**abled through the VREGEN Configuration bit."

  The sentence originally stated, "The regulator is enabled by default and can be disabled through the VREGEN Configuration bit."
- In the final note box of the section, Note 2 is corrected to read, "V_{DD} must be greater than **or equal to** V_{USB} at all times, even with the regulator disabled."

  The sentence originally stated, "V_{DD} must be greater than V_{USB} at all times, even with the regulator disabled."

4. Module: Master Synchronous Serial Port (MSSP)

In **Section 19.3.5 "Master Mode,"** the second paragraph of the second column is corrected to read, "This allows a maximum data rate (at 48 MHz) of **12.00** Mbps."

The sentence originally stated, "This allows a maximum data rate (at 48 MHz) of 2.00 Mbps."

5. Module: 10-Bit Analog-to-Digital (A/D) Converter

In **Register 21-1**, the display and the detailed bit description for bit 5 is corrected to **"VCFG1"**, rather than "VCFG0". All other bit 5 displays and descriptions are correct in the Device Data Sheet.

6. Module: Special Features of the CPU

In **Section 25.9.1 "Dedicated ICD/ICSP Port"**, the second sentence of the fourth paragraph is corrected to state, "When **V_{IHH}** is seen on the $\overline{MCLR}/V_{PP}$/RE3 pin, the state of the $\overline{ICRST}/ICV_{PP}$ pin is ignored". This refers to the high-voltage programming voltage level for ICSP™ (DC Specification D110).

The sentence originally stated, "When V_{IH} is seen on the $\overline{MCLR}/V_{PP}$/RE3 pin, the state of the ICRST/ICV_{PP} pin is ignored". That incorrectly referred to the maximum input voltage tolerated by the pin as an I/O (DC specification D040).

7. Module: Electrical Characteristics

In **Section 28.3 "DC Characteristics,"** pin-specific variations of parameters, D031 (Input Low Voltage) and D041 (Input High Voltage), are corrected as characteristic for pins, **RB0 and RB1**, not pins, RC3 and RC4.

The following relevant portion of the table indicates the corrections. (For clarity, the corrected items appear in **bold** text – all other text appears in plain text for purposes of this errata.)

28.3 DC Characteristics: PIC18F2455/2550/4455/4550 (Industrial) PIC18LF2455/2550/4455/4550 (Industrial) (Partial Presentation)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial			
Param No.	Symbol	Characteristic	Min	Max	Units	Conditions
D031	VIL	Input Low Voltage				
		with Schmitt Trigger buffer	VSS	0.2 VDD	V	
		RB0 and RB1	VSS	0.3 VDD	V	
D041	VIH	Input High Voltage				
		with Schmitt Trigger buffer	0.8 VDD	VDD	V	
		RB0 and RB1	0.7 VDD	VDD	V	

8. Module: Electrical Characteristics

In **Table 28-1**, the symbol for parameter D110 is corrected to **VIHH**, rather than VPP.

The following relevant portion of Table 28-1 indicates the correction. (For clarity, the corrected item appears in **bold** text – all other text appears in plain text for purposes of this errata.)

TABLE 28-9: MEMORY PROGRAMMING REQUIREMENTS (PARTIAL PRESENTATION)

DC Characteristics			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial				
Param No.	Sym	Characteristic	Min	Typ(†)	Max	Units	Conditions
		Internal Program Memory Programming Specifications(1)					
D110	**VIHH**	Voltage on $\overline{\text{MCLR}}$/VPP/RE3 pin	9.00	—	13.25	V	(Note 2)

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: These specifications are for programming the on-chip program memory through the use of table write instructions.

2: Required only if Single-Supply Programming is disabled.

9. Module: Electrical Characteristics

In **Table 28-5**, parameter D323 (Regulator Voltage Output) is qualified with the condition of a minimum device **$V_{DD} \geq$ 4.0V** in the Comments column.

The following relevant portion of Table 28-1 indicates the correction. (For clarity, the corrected item appears in **bold** text – all other text has been changed to plain text for purposes of this errata.)

TABLE 28-5: USB INTERNAL VOLTAGE REGULATOR SPECIFICATIONS (PARTIAL PRESENTATION)

Operating Conditions: -40°C < T_A < +85°C (unless otherwise stated).							
Param No.	Sym	Characteristics	Min	Typ	Max	Units	Comments
D323	V_{USBANA}	Regulator Output Voltage	3.0	—	3.6	V	**$V_{DD} \geq$ 4.0V**

10. Module: MSSP (SPI Master)

In **Section 19.3.5, "Master Mode,"** the following content is added:

When used in Timer2 Output/2 mode, the SPI bit rate can be configured using the PR2 Period register and the Timer2 prescaler.

To operate in this mode, firmware must first initialize and enable the Timer2 module before it can be used with the MSSP. Once enabled, the Timer2 module is free-running and mostly independent of the MSSP module.

Writing to the SSPBUF register will not clear the current TMR2 value in hardware. This can result in an unpredictable SPI transmit MSb bit width, depending on how close the TMR2 register was to the PR2 match condition at the moment that the firmware wrote to SSPBUF.

To avoid the unpredictable MSb bit width, initialize the TMR2 register to a known value when writing to SSPBUF. An example procedure, which provides predictable bit widths (only needed in the Timer2/2 mode), is given in Example 10.

EXAMPLE 10: LOADING SSPBUF WITH THE TIMER2/2 CLOCK MODE

```
TransmitSPI:
BCF     PIR1, SSPIF     ;Make sure interrupt flag is clear (may have been set from previous
                        ;transmission)
MOVF    SSPBUF, W       ;Perform read, even if the data in SSPBUF is not important
MOVWF   RXDATA          ;Save previously received byte in user RAM, if the data is meaningful
BCF     T2CON, TMR2ON   ;Turn off timer when loading SSPBUF
CLRF    TMR2            ;Set timer to a known state
MOVF    TXDATA, W       ;WREG = Contents of TXDATA (user data to send)
MOVWF   SSPBUF          ;Load data to send into transmit buffer
BSF     T2CON, TMR2ON   ;Start timer to begin transmission

WaitComplete:           ;Loop until data has finished transmitting
BTFSS   PIR1, SSPIF     ;Interrupt flag set when transmit is complete
BRA     WaitComplete
```

APPENDIX A: DOCUMENT REVISION HISTORY

Rev A Document (7/2009)

Initial release of this new, combined document. Republished earlier silicon issues from documents being discontinued (listed below). Added new silicon issues 39 (MSSP, I^2C Slave), 40 (MSSP, I^2C Master) and 41 (EUSART). Republished earlier data sheet clarifications from data sheet errata being discontinued. Added new data sheet issue 10 (MSSP, SPI Master).

This document replaces these errata documents:

- DS80220, *"PIC18F2455/2550/4455/4550 Revision A3 Silicon Errata"*
- DS80287, *"PIC18F2455/2550/4455/4550 Revision B4 Silicon Errata"*
- DS80322, *"PIC18F2455/2550/4455/4550 Revision B5 Silicon Errata"*
- DS80335, *"PIC18F2455/2550/4455/4550 Revision B6 Silicon Errata"*
- DS80388, *"PIC18F2455/2550/4455/4550 Revision B7 Silicon Errata"*
- DS80278, *"PIC18F2455/2550/4455/4550 Data Sheet Errata"*

NOTES:

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

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

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

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

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4080

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

03/26/09