PIC18F6627/6722/8627/8722 Rev. A1 Silicon Errata

The PIC18F6627/6722/8627/8722 parts you have received conform functionally to the Device Data Sheet (DS39646**B**), except for the anomalies described below. Any Data Sheet Clarification issues related to the PIC18F6627/6722/8627/8722 devices will be reported in a separate Data Sheet errata. Please check the Microchip web site for any existing issues.

All of the issues listed here will be addressed in future revisions of the PIC18F6627/6722/8627/8722 silicon.

The following silicon errata apply only to PIC18F6627/6722/8627/8722 devices with these Device/Revision IDs:

Part Number	Device ID	Revision ID
PIC18F6627	01 0011 110	00000
PIC18F6722	01 0100 000	00000
PIC18F8627	01 0011 111	00000
PIC18F8722	01 0100 001	00000

The Device IDs (DEVID1 and DEVID2) are located at addresses 3FFFFEh:3FFFFFh in the device's configuration space. They are shown in hexadecimal in the format "DEVID2 DEVID1".

1. Module: EUSART

When performing back-to-back transmission in 9-bit mode (TX9D bit in the TXSTAx 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 TXxIF. This is because any write to the TXSTAx register results in a reset of the Baud Rate Generator which will effect any ongoing transmission.

Work around

Load TX9D just after TXxIF is set, either by polling TXxIF 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).

Date Codes that pertain to this issue:

All engineering and production devices.

2. Module: Timer1/Timer3

When Timer1/Timer3 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 was written.

Work around

Do not write to TMR1H/TMR3H while Timer1/Timer3 is running, or else write to TMR1L/TMR3L immediately following a write to TMR1H/TMR3H.

Do not write to TMR1H/TMR3H and then wait for another event before also updating TMR1L/TMR3L.

Date Codes that pertain to this issue:

All engineering and production devices.

3. Module: Timer1/Timer3

When Timer1 or Timer3 is the time base for CCPx, and the associated CCPxCON register is configured with 0x0B (Compare mode, trigger special event), the assigned timer is not reset on a Special Event Trigger.

Work around

Modify firmware to reset the Timer registers (TMRxL and TMRxH) upon detection of the compare match condition.

Date Codes that pertain to this issue:

All engineering and production devices.

© 2006 Microchip Technology Inc.

4. Module: A/D

The A/D offset is greater than the specified limit in Table 28-26 of the Device Data Sheet. The updated conditions and limits are shown in **bold** text in Table 1.

Work around

Three work arounds exist:

1. 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>).
2. Perform a conversion on a known voltage reference voltage and adjust the A/D result in software.
3. Increase system clock speed and adjust A/D settings accordingly. Higher system clock frequencies decrease offset error.

Date Codes that pertain to this issue:

All engineering and production devices.

TABLE 1: A/D CONVERTER CHARACTERISTICS: PIC18F6X27/6X22/8X27/8X22 (INDUSTRIAL, EXTENDED) PIC18LF6X27/6X22/8X27/8X22 (INDUSTRIAL)

Param No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
A06A	**EOFF**	**Offset Error**	**—**	**—**	**±1.5**	**LSb**	**VREF = VREF+ and VREF-**
A06	**EOFF**	**Offset Error**	—	—	**±3.5**	**LSb**	**VREF = VSS and VDD**

5. Module: MSSP

I²C™ Receive mode should be enabled (i.e., RCEN bit should be set) only when the system is idle (i.e., ACKEN, RCEN, PEN, RSEN and SEN all equal zero). It should not be possible to set the RCEN bit when the system is not idle; however, the RCEN bit *can* be set under this circumstance.

Work around

Wait for the system to become idle before setting the RCEN bit. This requires a check for the following bits to be clear: ACKEN, RCEN, PEN, RSEN and SEN.

Date Codes that pertain to this issue:

All engineering and production devices.

6. Module: ECCP

When switching direction in Full-Bridge PWM mode, the modulated outputs will switch immediately instead of waiting for the next PWM cycle. This may generate unexpected short pulses on the modulated outputs.

Work around

Disable the PWM or set duty cycle to zero prior to switching directions.

Date Codes that pertain to this issue:

All engineering and production devices.

© 2006 Microchip Technology Inc.

7. 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; and

`MOVSS [Zs], [Zd]`
where the destination is WREG, BSR or STATUS.

Work around

1. Assembly Language Programming:

   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 and then restore WREG, BSR and STATUS via software as shown in Example 8-1 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, the following work around shown in Example 1 can be used. This example overwrites the Fast Return register by making a dummy call to `Foo` with the fast option in the high priority service routine.

2. C Language Programming:

   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, then 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. An optimized C18 version, which illustrates how to reduce the instruction cycle count to 3, is provided in Example 3.

Date Codes that pertain to this issue:

All engineering and production devices.

EXAMPLE 1: ASSEMBLY LANGUAGE INTERRUPT SERVICE

```
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
    :
    RETFIE     FAST
```

EXAMPLE 2: INTERRUPT SERVICE ROUTINE IN 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 */
```

EXAMPLE 3: OPTIMIZED INTERRUPT SERVICE ROUTINE

```
#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)
{
  ...
}
```


© 2006 Microchip Technology Inc.

8. Module: BOD

The BOD module may reset below the minimum operating voltage of the device when configured for BORV1:BORV0 = 11. The updated Reset voltage specifications are shown in **bold** in Table 2.

Work around

Use the next higher BOD voltage setting to ensure a low VDD is detected above 2.0V.

Date Codes that pertain to this issue:

All engineering and production devices.

TABLE 2: BROWN-OUT RESET VOLTAGE

D005	VBOR	**Brown-out Reset Voltage**				
		PIC18LF6627/6722/8627/8722				
		BORV1:BORV0 = 11	**N/A**	2.05	**N/A**	V

9. Module: EUSART

When performing back-to-back transmission in 9-bit mode (TX9D bit in the TXSTAx register is set), the second byte may be corrupted if it is written into TXREGx immediately after the TMRT bit is set.

Work around

Execute a software delay, at least one-half the transmission's bit time, after TMRT is set and prior to writing subsequent bytes into TXREGx.

Date Codes that pertain to this issue:

All engineering and production devices.

10. 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.

11. Module: EUSART

In Synchronous mode (SYNC = 1) with clock polarity high (SCKP = 1), the EUSART transmits a shorter than expected clock on the CKx pin for the last bit transmitted.

Work around

None.

12. Module: EUSART

In Synchronous mode, EUSART baud rates using SPBRGx values of '0' and '1' may not function correctly.

Work around

Use another baud rate configuration to generate the desired baud rate.

13. Module: EUSART

After the last received byte has been read from the EUSART receive buffer, RCREGx, the value is no longer valid for subsequent read operations. The RCREGx register should only be read once for each byte received.

Work around

After each byte is received from the EUSART, store the byte into a user variable. To determine when a byte is available to read from RCREGx, poll the RCIDL (BAUDCONx<6>) bit for a low-to-high transition or use the EUSART receive interrupt, RCxIF (PIRx<5>).

14. Module: EUSART

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

Work around

Only write to TX9D when a reception is not in progress (RCIDL = 1). No interrupt is associated with RCIDL, therefore, it must be polled in software to determine when TX9D can be updated.

15. Module: ECCP

PIC18F6XXX device's Configuration Word, CONFIG3L, is not unimplemented and will switch the ECCP2 output pin similar to the PIC18F8XXX devices if the Processor mode does not select Microcontroller mode.

Work around

In MPLAB® IDE, program the PIC18F6XXX device as its PIC18F8XXX equivalent and assign the Processor mode bits (PM<1:0>) to '11' for Microcontroller mode.

16. Module: External Memory Bus

For PIC18F8XXX devices, the Stack Pointer may incorrectly increment during a table read operation if external memory bus wait states are enabled (i.e., Configuration bit, WAIT, is clear (CONFIG3L<7> = 0) and WAIT<1:0> bits (MEMCON<5:4>) are not equal to '11').

Work around

If using the external memory bus and performing TBLRD operations with a non-zero wait state (CONFIG3L<7> = 0 and WAIT<1:0> (MEMCON<5:4> are not equal to '11'), disable interrupts by clearing the GIE/GIEH (INTCON<7>) and PEIE/GIEL (INTCON<6>) bits prior to executing any TBLRD operation.

17. Module: MSSP

In an I²C™ 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 SSPxIF 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 I²C slave must clear the SSPOV bit after each I²C address match to maintain normal operation.

18. Module: MSSP

Setting the SEN bit initiates a Start sequence on the bus, after which, the SEN bit is cleared automatically by hardware. If the SEN bit is set again (without an address byte being transmitted), a Start sequence will not commence and the SEN bit will not be cleared. This condition causes the bus to remain in an active state. The system is idle when ACKEN, RCEN, PEN, RSEN, and SEN are clear.

Work around

Set the PEN or RSEN bit to transmit a Stop or Repeated Start sequence, although the SEN bit may still be set, indicating the bus is active. After the sequence has completed, the PEN, RSEN and SEN bits will be clear, indicating the bus is idle. Clearing and setting the SSPEN bit will also reset the I²C peripheral and clear the PEN, RSEN and SEN status bits.

19. Module: MSSP

In SPI mode, the Buffer Full flag (BF bit in the SSPxSTAT register), the Write Collision Detect bit (WCOL in SSPxCON1) and the Receive Overflow Indicator bit (SSPOV in SSPxCON1) are not reset upon disabling the SPI module (by clearing the SSPEN bit in the SSPxCON1 register).

For example, if SSPxBUF is full (BF bit is set) and the MSSP module is disabled and re-enabled, the BF bit will remain set. In SPI Slave mode, a subsequent write to SSPxBUF will result in a write collision. Also, if a new byte is received, a receive overflow will occur.

Work around

Ensure that if the buffer is full, SSPxBUF is read (thus clearing the BF flag) and WCOL is clear before disabling the MSSP module. If the module is configured in SPI Slave mode, ensure that the SSPOV bit is clear before disabling the module.

20. Module: MSSP

In I²C Master mode, the BRG value of '0' may not work correctly.

Work around

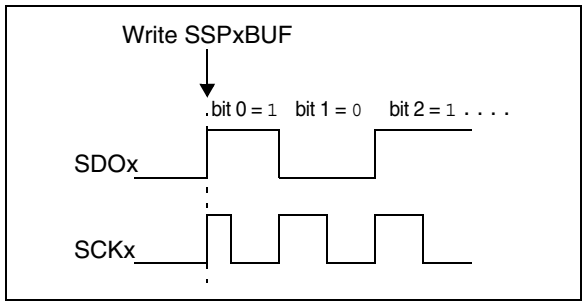
Use a BRG value greater than '0' by setting SSPxADD ≥ '1'.

© 2006 Microchip Technology Inc.

21. Module: MSSP (SPI Mode)

When the SPI is using Timer2/2 as the clock source, a shorter than expected SCKx pulse may occur on the first bit of the transmitted/received data (Figure 1).

FIGURE 1: SCKx PULSE VARIATION USING TIMER2/2



Work around

To avoid producing the short pulse, turn off Timer2 and clear the TMR2 register, load the SSPxBUF with the data to transmit and then turn Timer2 back on. Refer to Example 4 for sample code.

EXAMPLE 4: AVOIDING THE INITIAL SHORT SCK PULSE

```
LOOP BTFSS  SSPSTAT, BF     ;Data received?
                            ;(Xmit complete?)
     BRA    LOOP            ;No
     MOVF   SSPBUF, W       ;W = SSPBUF
     MOVWF  RXDATA          ;Save in user RAM
     MOVF   TXDATA, W       ;W = TXDATA
     BCF    T2CON, TMR2ON   ;Timer2 off
     CLRF   TMR2            ;Clear Timer2
     MOVWF  SSPBUF          ;Xmit New data
     BSF    T2CON, TMR2ON   ;Timer2 on
```

22. Module: Timer1

In 16-bit Asynchronous Counter mode or 16-bit Asynchronous Oscillator mode, 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

Use 8-bit mode by clearing the RD16 (T1CON<7>) bit or use the synchronization option by clearing T1SYNC (T1CON<2>).

23. Module: PORTE

The RE4 pin latch remains at a logic level zero when the ECCPMX Configuration bit is clear and selects PORTH.

Work around

This issue will be corrected in a future revision of silicon.

24. Module: Timer1 (Asynchronous Counter)

When writing to the TMR1H register, under specific conditions, it is possible that the TMR1L register will miss a count while connected to the external oscillator via the T1OSO and T1OSI pins.

When Timer1 is started, the circuitry looks for a falling edge before a rising edge can increment the counter. Writing to the TMR1H register is similar to starting Timer1; therefore, the former logic stated applies any time the TMR1H register is written. If the TMR1H register is not completely written to during the high pulse of the external clock, then the TMR1L register will miss a count due to the circuit operation stated previously. The high pulse of a 32.768 kHz external clock crystal yields a 15.25 µs window for the write to TMR1H to occur. The amount of instructions that can be executed within this window is frequency dependent, as shown in Table 3.

Work around

Operating Conditions: FOSC ≥ 4 MHz, no wake-ups from Sleep, Timer1 interrupt enabled, global interrupts enabled.

The code excerpts in Example 5 and Example 6 show how the TMR1H register can be updated while the external clock (32.768 kHz) is still on its high pulse.

The importance of the code examples is that the **bold** instructions are executed within the first 15.25 µs high pulse on the external clock after the Timer1 overflow occurred. This will allow the TMR1L register to increment correctly.

Note:	These instructions are required based on the function of the ISR. If the only code in the ISR is to reload Timer1, then they are not required, but may be required if additional code is added.

TABLE 3: FREQUENCY DEPENDENT INSTRUCTION EXECUTION AMOUNTS

FOSC	TCY (µs)	TCY within 15.25 µs
1 MHz	4	3.81
2 MHz	2	7.63
4 MHz	1	15.25
8 MHz	0.5	30.5
16 MHz	0.25	61
20 MHz	0.2	76.25
40 MHz	0.1	152.5

EXAMPLE 5: PIC18 HIGH PRIORITY INTERRUPT SERVICE ROUTINE

```
ISR @ 0x0008                    ; (3-4Tcy), fixed interrupt latency

    BRA     HIGHINT             ; (2Tcy), go to high priority interrupt routine

HIGHINT
    BTFSC   PIR1, TMR1IF        ; (1Tcy), did a Timer1 overflow occur?
    BSF     TMR1H, 7            ; (1Tcy) Yes, reload for a 1 second overflow

    RETFIE  FAST
```

Total = 7-8 TCY (if Timer1 overflow occurred)

EXAMPLE 6: PIC18 LOW PRIORITY INTERRUPT SERVICE ROUTINE

```
ISR @ 0x0018                    ; (3-4Tcy), fixed interrupt latency
    MOVFF   STATUS, STATUS_TEMP ; (2Tcy), save STATUS register
    MOVFF   WREG, WREG_TEMP     ; (2Tcy), save working register, refer to note 1
    MOVFF   BSR, BSR_TEMP       ; (2Tcy), save BSR register, refer to note 1

    BTFSS   PIR1, TMR1IF        ; (2Tcy), did a Timer1 overflow occur?
    BRA     EXIT                ; No
    BSF     TMR1H, 7            ; (1Tcy) Yes, reload for a 1 second overflow

EXIT
    MOVFF   BSR_TEMP, BSR       ;restore BSR register, refer to note 1
    MOVFF   WREG_TEMP, WREG     ;restore working register, refer to note 1
    MOVFF   STATUS_TEMP, STATUS ;restore STATUS register
    RETFIE
```

Total = 12-13 TCY (if Timer1 overflow occurred)

© 2006 Microchip Technology Inc.

25. 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, TXREGx, are transferred to the TSRx during the transmission of a Stop bit. For this to occur, three things must happen in the same instruction cycle:

- TXREGx is written to;
- the baud rate counter overflows (at the end of the bit period); and
- a Stop bit is being transmitted (shifted out of TSRx).

Work around

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

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

26. Module: EUSART

When performing back-to-back transmission in 9-bit mode (TX9D bit in the TXSTAx register is set), the second byte may be corrupted if it is written into TXREGx immediately after the TMRT bit is set.

Work around

Execute a software delay, at least one-half the transmission's bit time, after TMRT is set and prior to writing subsequent bytes into TXREGx.

Date Codes that pertain to this issue:

All engineering and production devices.

If transmission is intermittent, then do the following:

- Wait for the TRMT bit to be set before loading TXREGx
- Alternatively, use a free timer resource to time the baud period. Set up the timer to overflow at the end of the Stop bit, then start the timer when you load the TXREGx. Do not load the TXREGx when timer is about to overflow.

Date Codes that pertain to this issue:

All engineering and production devices.

27. Module: EUSART

With the auto-wake-up option enabled by setting the WUE (BAUDCONx<1>) bit, the RCxIF bit will become set on a high-to-low transition on the RXx pin. However, the WUE bit may not clear within 1 TCY of a low-to-high transition on RXx. While the WUE bit is set, reading the receive buffer, RCREGx, will not clear the RCxIF interrupt flag. Therefore, the first opportunity to automatically clear RCxIF by reading RCREGx may take longer than expected.

Note:	RCxIF can only be cleared by reading RCREGx.

Work around

There are two work arounds available:

1. Clear the WUE bit in software, after the wake-up event has occurred, prior to reading the receive buffer, RCREGx.
2. Poll the WUE bit and read RCREGx after the WUE bit is automatically cleared.

Date Codes that pertain to this issue:

All engineering and production devices.

28. Module: MSSP (SPI Mode)

In SPI mode, the SDOx 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 SCKx.

Work around

None

Date Codes that pertain to this issue:

All engineering and production devices.

29. Module: MSSP (SPI Mode)

In SPI mode, the Buffer Full Status bit, BF (SSPxSTAT<0>), should not be polled in software to determine when the transfer is complete.

Work around

Copy the SSPxSTAT register into a variable and perform the bit test on the variable. In Example 7, SSPxSTAT is copied into the working register where the bit test is performed (SSP1STAT is shown, but this process is also applicable to SSP2STAT).

EXAMPLE 7:

```
loop_MSB:
    MOVF    SSP1STAT, W
    BTFSS   WREG, BF
    BRA     loop_MSB
```

A second option is to poll the appropriate Master Synchronous Serial Port Interrupt Flag bit, SSPxIF. This bit can be polled and will set when the transfer is complete.

Date Codes that pertain to this issue:

All engineering and production devices.

30. Module: MSSP (SPI Mode)

In SPI Master mode, a write collision may occur if the SSPxBUF register is loaded immediately after a transfer is complete. This may be caused by an inadequate delay between the MSSP Interrupt Flag bit (SSPxIF) or the Buffer Full bit (BF) being set, and SSPxBUF being written to.

This has only been observed when the SPI clock is operating at FOSC/64 or ((Timer2)/2) (SSPxCON1<3:0> = 001x).

Work around

Add a software delay of one SCKx period after detecting the completed transfer, and prior to updating the contents of SSPxBUF.

Also verify that the Write Collision bit (WCOL) is clear after writing SSPxBUF. If WCOL is set, clear the bit in software and rewrite the contents of SSPxBUF.

Date Codes that pertain to this issue:

All engineering and production devices.

31. Module: MSSP (I^2C™ Mode)

In its current implementation, the I^2C Master mode operates as follows:

a) The Baud Rate Generator for I^2C in Master mode is slower than the rates specified in Table 17-3 of the Device Data Sheet.

For this revision of silicon, use the values shown in Table 4 (below) in place of those shown in Table 17-3 of the Device Data Sheet. The differences are shown in **bold** text.

b) Use the following formula in place of the one shown in Register 19-4 (SSPxCON1) of the Device Data Sheet for bit description SSPM3:SSPM0 = 1000.

$$SSPxADD = INT((F_{CY}/F_{SCL}) - (F_{CY}/1.111\ MHz)) - 1$$

Date Codes that pertain to this issue:

All engineering and production devices.

TABLE 4: I^2C™ CLOCK RATE w/BRG

FOSC	FCY	FCY * 2	BRG Value	FSCL (2 Rollovers of BRG)
40 MHz	10 MHz	20 MHz	**0Eh**	400 kHz(1)
40 MHz	10 MHz	20 MHz	**15h**	312.5 kHz
40 MHz	10 MHz	20 MHz	**59h**	100 kHz
16 MHz	4 MHz	8 MHz	**05h**	400 kHz(1)
16 MHz	4 MHz	8 MHz	**08h**	308 kHz
16 MHz	4 MHz	8 MHz	**23h**	100 kHz
4 MHz	1 MHz	2 MHz	**01h**	333 kHz(1)
4 MHz	1 MHz	2 MHz	**08h**	100 kHz
4 MHz	1 MHz	2 MHz	00h	1 MHz(1)

Note 1: The I^2C™ interface does not conform to the 400 kHz I^2C specification (which applies to rates greater than 100 kHz) in all details, but may be used with care where higher rates are required by the application.

© 2006 Microchip Technology Inc.

32. Module: MSSP (I^2C Mode)

It has been observed that, following a Power-on Reset, I^2C mode may not initialize properly by just configuring the SCLx and SDAx 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 SCLx and SDAx pins as outputs by clearing their corresponding TRIS bits.
2. Force SCLx and SDAx low by clearing the corresponding LAT bits.
3. While keeping the LAT bits clear, configure SCLx and SDAx as inputs by setting their TRIS bits.

Once this is done, use the SSPxCON1 and SSPxCON2 registers to configure the proper I^2C mode as before.

Date Codes that pertain to this issue:

All engineering and production devices.

33. Module: MSSP (I^2C Mode)

In I^2C Master mode, the RCEN bit is set by software to begin data reception, and cleared by the peripheral after a byte is received. After a byte is received, the device may take up to 80 TCY to clear RCEN and 800 TCY during emulation.

Work around

Single byte receptions are typically not affected, since the delay between byte receptions typically is long enough for the RCEN bit to clear. For multiple byte receptions, the software must wait until the bit is cleared by the peripheral before the next byte can be received.

Date Codes that pertain to this issue:

All engineering and production devices.

34. Module: ECCP (PWM Mode)

When the PWM auto-shutdown feature is configured for automatic restart by setting the PxRSEN bit (ECCPxDEL<7>), the pulse may terminate 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 PxRSEN bit. The PWM can be re-enabled by clearing the ECCPxASE bit (ECCPxAS<7>) after the shutdown condition expires.

Date Codes that pertain to this issue:

All engineering and production devices.

35. 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 (PxDC6:PxDC0 > 0); and
- the duty cycle has a value of 0 through 3, or 4n + 3 (n ≥ 1).

Work around

None.

Date Codes that pertain to this issue:

All engineering and production devices.

36. Module: CCP (PWM Mode)

Timer4 is not available as a clock source for either CCP4 or CCP5 in any PWM mode (CCPxCON<3:2> = 11). Selecting Timer4 as the module's clock source may cause the PWM output to stop generating pulses.

Work around

To use CCP4 or CCP5 in PWM mode, use only Timer2 as the clock source (T3CON<6,3> = 00).

Date Codes that pertain to this issue:

All engineering and production devices.

37. Module: Reset

This 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 (i.e., WDT, BOR or $\overline{\text{MCLR}}$ occurs when a write operation is being executed (start of a Q4 cycle).

Work around

None.

Date Codes that pertain to this issue:

All engineering and production devices.

38. Module: External Memory Bus

The A<19:16> EMB address lines and Read/Write control pins ($\overline{\text{OE}}$, $\overline{\text{WRH}}$ and $\overline{\text{WRL}}$) are released to their respective inactive states at the same time, violating the timing condition mentioned in Figure 28-8 and Figure 28-9 in the Device Data Sheet. This may result in a peripheral device on the bus detecting an address change when a write/read is initiated. The bus capacitance and signal delay on the address and control lines can affect the probability of invalid detection.

Work around

Two work arounds are available:

1. Use a latch based on the falling edge of ALE to hold the A<19:16> signals.
2. Add a delay circuit to extend the valid time for A<19:16> signals to ensure the address is valid until read/write signals go inactive.

© 2006 Microchip Technology Inc.

REVISION HISTORY

Rev A Document (12/2004)

Original version of this document. Contains silicon issues: 1 (EUSART), 2-3 (Timer1/Timer3), 4 (A/D), 5 (MSSP), 6 (ECCP), 7 (Interrupts), 8 (BOD), 9 (EUSART) and 10 (HLVD).

Rev B Document (12/2005)

Updated issue 7 (Interrupts), deleted previous issue 10 (HLVD), and added issues 10-14 (EUSART), 15 (ECCP), 16 (External Memory Bus), 17-20 (MSSP), 21 (MSSP – SPI Mode), 22 (Timer1), 23 (PORTE) and 24 (Timer1 – Asynchronous Counter).

Rev C Document (8/2006)

Updated issue 7 (Interrupts) to include new code examples. Added silicon issues 25-27 (EUSART), 28-30 (MSSP – SPI Mode), 31-33 (MSSP – I^2C Mode), 34-35 (ECCP – PWM Mode), 36 (CCP – PWM Mode), 37 (Reset) and 38 (External Memory Bus).

NOTES:

© 2006 Microchip Technology Inc.

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.

Trademarks

The Microchip name and logo, the Microchip logo, Accuron, dsPIC, KEELOQ, microID, MPLAB, PIC, PICmicro, PICSTART, PRO MATE, PowerSmart, rfPIC, and SmartShunt are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AmpLab, FilterLab, Migratable Memory, MXDEV, MXLAB, SEEVAL, SmartSensor and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, ECAN, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, Linear Active Thermistor, Mindi, MiWi, MPASM, MPLIB, MPLINK, PICkit, PICDEM, PICDEM.net, PICLAB, PICtail, PowerCal, PowerInfo, PowerMate, PowerTool, REAL ICE, rfLAB, rfPICDEM, Select Mode, Smart Serial, SmartTel, Total Endurance, UNI/O, WiperLock 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.

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

© 2006, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

Printed on recycled paper.

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 Mountain View, California. The Company's quality system processes and procedures are for its PICmicro® 8-bit MCUs, 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
Alpharetta, GA
Tel: 770-640-0034
Fax: 770-640-0307

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

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

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

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

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

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

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

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

ASIA/PACIFIC

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

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

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

China - Chengdu
Tel: 86-28-8676-6200
Fax: 86-28-8676-6599

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

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

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

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

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

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

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

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

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

ASIA/PACIFIC

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

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

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

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

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

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

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

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

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

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

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

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

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

EUROPE

Austria - Wels
Tel: 43-7242-2244-3910
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

07/21/06

© 2006 Microchip Technology Inc.