
**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Active |
| Core Processor | PIC |
| Core Size | 8-Bit |
| Speed | 32MHz |
| Connectivity | I²C, LINbus, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, POR, PWM, WDT |
| Number of I/O | 18 |
| Program Memory Size | 28KB (16K x 14) |
| Program Memory Type | FLASH |
| EEPROM Size | 256 x 8 |
| RAM Size | 2K x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.3V ~ 5.5V |
| Data Converters | A/D 17x12b; D/A 1x5b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Through Hole |
| Package / Case | 20-DIP (0.300", 7.62mm) |
| Supplier Device Package | 20-PDIP |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16f18446-e-p |

Program Memory Organization
Storage Area Flash
Memory Violation

## 8.12 Determining the Cause of a Reset

Upon any Reset, multiple bits in the STATUS and PCON0 registers are updated to indicate the cause of the Reset. The following tables show the Reset conditions of these registers.

**Table 8-3. Reset Status Bits and Their Significance**

| STOVF | STKUNF | $\overline{\text{RWDT}}$ | $\overline{\text{RMCLR}}$ | $\overline{\text{RI}}$ | $\overline{\text{POR}}$ | $\overline{\text{BOR}}$ | $\overline{\text{TO}}$ | $\overline{\text{PD}}$ | $\overline{\text{MEMV}}$ | Condition |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 | 0 | x | 1 | 1 | 1 | Power-on Reset |
| 0 | 0 | 1 | 1 | 1 | 0 | x | 0 | x | u | Illegal, $\overline{\text{TO}}$ is set on $\overline{\text{POR}}$ |
| 0 | 0 | 1 | 1 | 1 | 0 | x | x | 0 | u | Illegal, $\overline{\text{PD}}$ is set on $\overline{\text{POR}}$ |
| 0 | 0 | u | 1 | 1 | u | 0 | 1 | 1 | u | Brown-out Reset |
| u | u | 0 | u | u | u | u | 0 | u | u | WWDT Reset |
| u | u | u | u | u | u | u | 0 | 0 | u | WWDT Wake-up from Sleep |
| u | u | u | u | u | u | u | 1 | 0 | u | Interrupt Wake-up from Sleep |
| u | u | u | 0 | u | u | u | u | u | 1 | $\overline{\text{MCLR}}$ Reset during normal operation |
| u | u | u | 0 | u | u | u | 1 | 0 | u | $\overline{\text{MCLR}}$ Reset during Sleep |
| u | u | u | u | 0 | u | u | u | u | u | RESET Instruction Executed |
| 1 | u | u | u | u | u | u | u | u | u | Stack Overflow Reset (STVREN = 1) |
| u | 1 | u | u | u | u | u | u | u | u | Stack Underflow Reset (STVREN = 1) |
| u | u | u | u | u | u | u | u | u | 0 | Memory violation Reset |

**Table 8-4. Reset Condition for Special Registers**

| Condition | Program Counter | STATUS Register | PCON0 Register | PCON1 Register |
|---|---|---|---|---|
| Power-on Reset | 0 | ---1 1000 | 0011 110x | ---- --1- |
| Brown-out Reset | 0 | ---1 1000 | 0011 11u0 | ---- --u- |
| $\overline{\text{MCLR}}$ Reset during normal operation | 0 | -uuu uuuu | uuuu 0uuu | ---- --1- |
| $\overline{\text{MCLR}}$ Reset during Sleep | 0 | ---1 0uuu | uuuu 0uuu | ---- --u- |
| WWDT Time-out Reset | 0 | ---0 uuuu | uuu0 uuuu | ---- --u- |
| WWDT Wake-up from Sleep | PC + 1 | ---0 0uuu | uuuu uuuu | ---- --u- |
| WWDT Window Violation Reset | 0 | ---u uuuu | uu0u uuuu | ---- --u- |

### 9.6.4 OSCSTAT

**Name:** OSCSTAT
**Address:** 0x890

Oscillator Status Register 1

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EXTOR | HFOR | MFOR | LFOR | SOR | ADOR | | PLLR |
| Access | RO | RO | RO | RO | RO | RO | | RO |
| Reset | q | q | q | q | q | q | | q |

**Bit 7 – EXTOR** EXTOSC (external) Oscillator Ready bit

| Value | Description |
|---|---|
| 1 | The oscillator is ready to be used |
| 0 | The oscillator is not enabled, or is not yet ready to be used |

**Bit 6 – HFOR** HFINTOSC Oscillator Ready bit

| Value | Description |
|---|---|
| 1 | The oscillator is ready to be used |
| 0 | The oscillator is not enabled, or is not yet ready to be used |

**Bit 5 – MFOR** MFINTOSC Oscillator Ready bit

| Value | Description |
|---|---|
| 1 | The oscillator is ready to be used |
| 0 | The oscillator is not enabled, or is not yet ready to be used |

**Bit 4 – LFOR** LFINTOSC Oscillator Ready bit

| Value | Description |
|---|---|
| 1 | The oscillator is ready to be used |
| 0 | The oscillator is not enabled, or is not yet ready to be used |

**Bit 3 – SOR** Secondary (Timer1) Oscillator Ready bit

| Value | Description |
|---|---|
| 1 | The oscillator is ready to be used |
| 0 | The oscillator is not enabled, or is not yet ready to be used |

**Bit 2 – ADOR** ADC Oscillator Ready bit

| Value | Description |
|---|---|
| 1 | The oscillator is ready to be used |
| 0 | The oscillator is not enabled, or is not yet ready to be used |

**Bit 0 – PLLR** PLL Ready bit

### 10.7.7 PIE5

**Name:** PIE5
**Address:** 0x71B

Peripheral Interrupt Enable Register 5

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CLC4IE | CLC3IE | CLC2IE | CLC1IE | | TMR5GIE | TMR3GIE | TMR1GIE |
| Access | R/W | R/W | R/W | R/W | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | | 0 | 0 | 0 |

**Bit 7 – CLC4IE** CLC4 Interrupt Enable bit

| Value | Description |
|---|---|
| 1 | Enabled |
| 0 | Disabled |

**Bit 6 – CLC3IE** CLC3 Interrupt Enable bit

| Value | Description |
|---|---|
| 1 | Enabled |
| 0 | Disabled |

**Bit 5 – CLC2IE** CLC2 Interrupt Enable bit

| Value | Description |
|---|---|
| 1 | Enabled |
| 0 | Disabled |

**Bit 4 – CLC1IE** CLC1 Interrupt Enable bit

| Value | Description |
|---|---|
| 1 | Enabled |
| 0 | Disabled |

**Bit 2 – TMR5GIE** TMR5 Gate Interrupt Enable bit

| Value | Description |
|---|---|
| 1 | Enabled |
| 0 | Disabled |

**Bit 1 – TMR3GIE** TMR3 Gate Interrupt Enable bit

| Value | Description |
|---|---|
| 1 | Enabled |
| 0 | Disabled |

**Bit 0 – TMR1GIE** TMR1 Gate Interrupt Enable bit

2. When WDTCPS in CONFIG3 = 11111, the Reset value (q) of WDTPS is '01011'. Otherwise, the Reset value of WDTPS is equal to WDTCPS in CONFIG3.
3. When WDTCPS in CONFIG3L ≠ 11111, these bits are read-only.

## 20.4 ADC Charge Pump

The ADC module has a dedicated charge pump which can be controlled through the ADCPCON0 register. The primary purpose of the charge pump is to supply a constant voltage to the gates of transistor devices in the A/D converter, signal and reference input pass-gates, to prevent degradation of transistor performance at low operating voltage.

The charge pump can be enabled by setting the CPON bit. Once enabled, the pump will undergo a start-up time to stabilize the charge pump output. Once the output stabilizes and is ready for use, the CPRDY bit will be set.

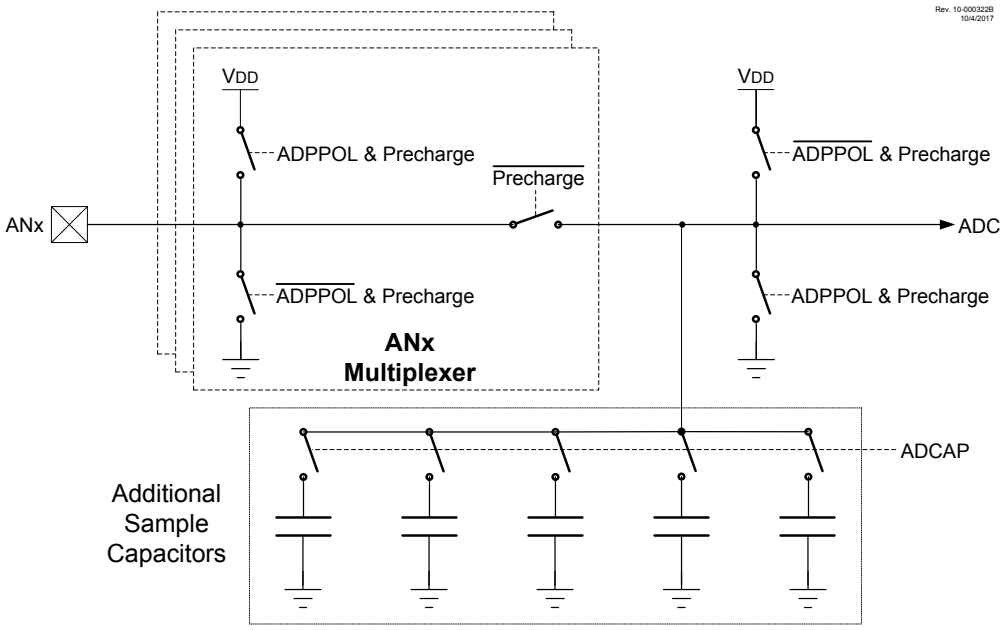
**Related Links**

ADCPCON0

## 20.5 Capacitive Voltage Divider (CVD) Features

The ADC module contains several features that allow the user to perform a relative capacitance measurement on any ADC channel using the internal ADC sample and hold capacitance as a reference. This relative capacitance measurement can be used to implement capacitive touch or proximity sensing applications. The following figure shows the basic block diagram of the CVD portion of the ADC module.

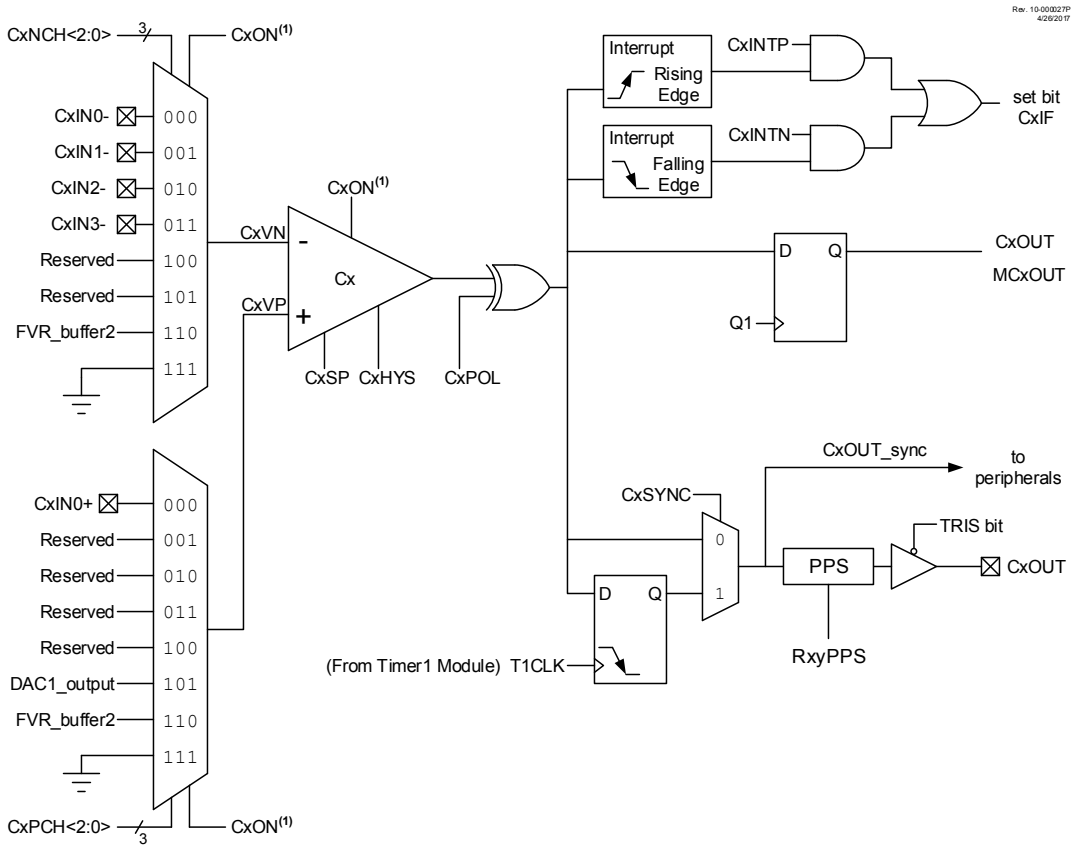
**Figure 20-7. Hardware Capacitive Voltage Divider Block Diagram**




### 20.5.1 CVD Operation

A CVD operation begins with the ADC's internal sample and hold capacitor ($C_{HOLD}$) being disconnected from the path which connects it to the external capacitive sensor node. While disconnected, $C_{HOLD}$ is precharged to $V_{DD}$ or $V_{SS}$ the sensor node is also charged to $V_{SS}$ or $V_{DD}$ respectively to the level opposite that of $C_{HOLD}$. When the precharge phase is complete, the $V_{DD}/V_{SS}$ bias paths for the two nodes are shut off and the paths between $C_{HOLD}$ and the external sensor node is re-connected, at which time the acquisition phase of the CVD operation begins. During acquisition, a capacitive voltage divider is formed between the precharged $C_{HOLD}$ and sensor nodes, which results in a final voltage level setting on

**Figure 23-2. Comparator Module Simplified Block Diagram**




**Related Links**

CMxNCH

CMxPCH

## 23.2 Comparator Control

Each comparator has two control registers: CMxCON0 and CMxCON1.

The CMxCON0 register contains Control and Status bits for the following:

- Enable
- Output
- Output polarity
- Hysteresis enable
- Timer1 output synchronization

The CMxCON1 register contains Control bits for the following:

- Interrupt on positive/negative edge enables
- Positive input channel selection
- Negative input channel selection

**Equation 24-6. Series R for V range**

$$R_{SERIES} = \frac{V_{MAX\_PEAK} + V_{MIN\_PEAK}}{7 \times 10^{-4}}$$

## 24.7 Operation During Sleep

The ZCD current sources and interrupts are unaffected by Sleep.

## 24.8 Effects of a Reset

The ZCD circuit can be configured to default to the active or inactive state on Power-on Reset (POR). When the $\overline{ZCD}$ Configuration bit is cleared, the ZCD circuit will be active at POR. When the $\overline{ZCD}$ Configuration bit is set, the SEN bit must be set to enable the ZCD module.

## 24.9 Disabling the ZCD Module

The ZCD module can be disabled in two ways:

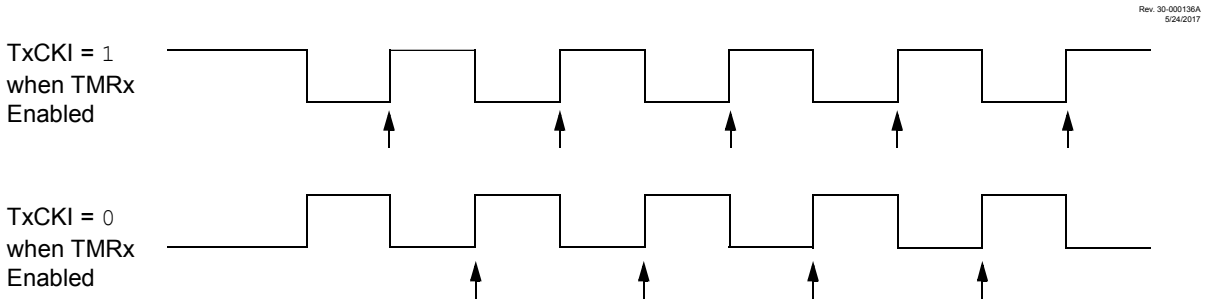
1. The $\overline{ZCD}$ configuration bit disables the ZCD module when set. When this is the case then the ZCD module will be enabled by setting the SEN bit. When the $\overline{ZCD}$ bit is clear, the ZCD is always enabled and the SEN bit has no effect.
2. The ZCD can also be disabled using the ZCDMD bit of the PMDx register. This is subject to the status of the $\overline{ZCD}$ bit.

When the $F_{OSC}$ internal clock source is selected, the Timer1 register value will increment by four counts every instruction clock cycle. Due to this condition, a 2 LSB error in resolution will occur when reading the Timer1 value. To utilize the full resolution of Timer1, an asynchronous input signal must be used to gate the Timer1 clock input.

**Important:** In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge after any one or more of the following conditions:

- Timer1 enabled after POR
- Write to TMRxH or TMRxL
- Timer1 is disabled
- Timer1 is disabled (TMRxON = 0) when TxCKI is high then Timer1 is enabled (TMRxON = 1) when TxCKI is low. Refer to the figure below.

**Figure 26-2. Timer1 Incrementing Edge**




**Note:**

1. Arrows indicate counter increments.
2. In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge of the clock.

### 26.2.2 External Clock Source

When the external clock source is selected, the Timer1 module may work as a timer or a counter.

When enabled to count, Timer1 is incremented on the rising edge of the external clock input of the TxCKIPPS pin. This external clock source can be synchronized to the system clock or it can run asynchronously.

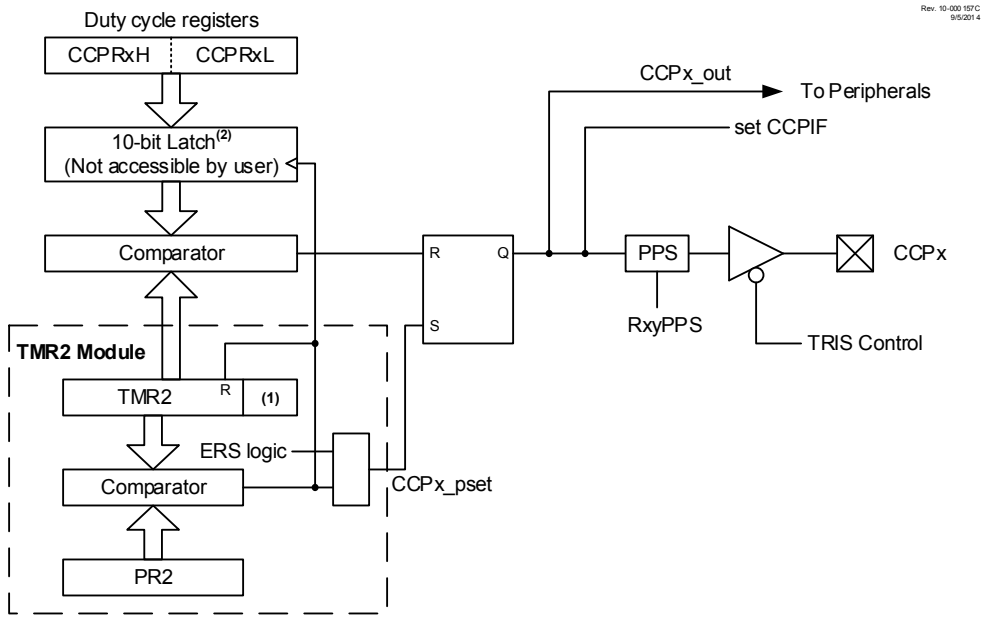
## 26.3 Timer1 Prescaler

Timer1 has four prescaler options allowing 1, 2, 4 or 8 divisions of the clock input. The CKPS bits control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMRxH or TMRxL.

## 26.4 Secondary Oscillator

A secondary low-power 32.768 kHz oscillator circuit is built-in between pins SOSCI (input) and SOSCO (amplifier output). This internal circuit is to be used in conjunction with an external 32.768 kHz crystal. The secondary oscillator is not dedicated only to Timer1; it can also be used by other modules.

**Figure 29-4. Simplified PWM Block Diagram**




**Note:**

1. 8-bit timer is concatenated with two bits generated by $F_{OSC}$ or two bits of the internal prescaler to create 10-bit time-base.
2. The alignment of the 10 bits from the CCPRx register is determined by the CCPxFMT bit.

**Important:** The corresponding TRIS bit must be cleared to enable the PWM output on the CCPx pin.

### 29.4.2 Setup for PWM Operation

The following steps should be taken when configuring the CCP module for standard PWM operation:

1. Use the desired output pin RxyPPS control to select CCPx as the source and disable the CCPx pin output driver by setting the associated TRIS bit.
2. Load the T2PR register with the PWM period value.
3. Configure the CCP module for the PWM mode by loading the CCPxCON register with the appropriate values.
4. Load the CCPRx register with the PWM duty cycle value and configure the FMT bit to set the proper register alignment.
5. Configure and start Timer2:
   - Clear the TMR2IF interrupt flag bit of the PIRx register. See Note below.
   - Select the timer clock source to be as $F_{OSC}/4$ using the TxCLKCON register. This is required for correct operation of the PWM module.
   - Configure the T2CKPS bits of the T2CON register with the Timer prescale value.
   - Enable the Timer by setting the T2ON bit.

### 31.15.9 CWGxDBF

**Name:** CWGxDBF
**Address:** 0x60F,0x619

CWG Falling Dead-Band Count Register

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | DBF[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | x | x | x | x | x | x |

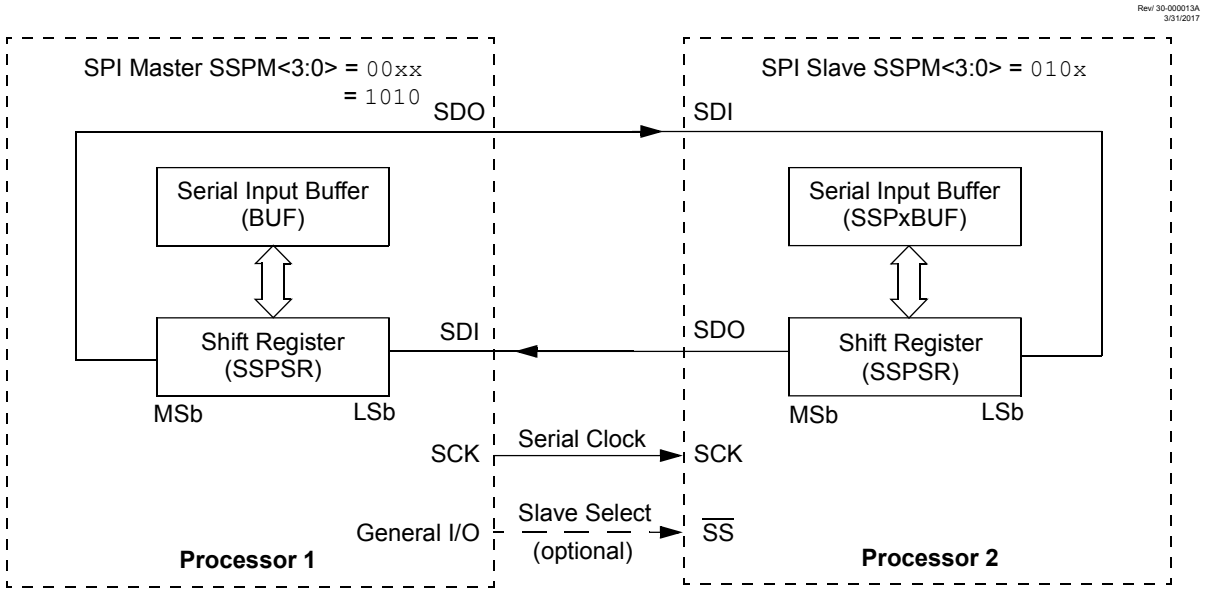
**Bits 5:0 – DBF[5:0]** CWG Falling Edge Triggered Dead-Band Count bits
Reset States: POR/BOR = xxxxxx
All Other Resets = uuuuuu

| Value | Description |
|---|---|
| n | Dead band is active no less than n, and no more than n+1, CWG clock periods after the falling edge |
| 0 | 0 CWG clock periods. Dead-band generation is bypassed |

bit (MSb) shifted out first. At the same time, a new Least Significant bit (LSb) is shifted into the same register.

The following figure shows a typical connection between two processors configured as master and slave devices.

**Figure 35-3. SPI Master/Slave Connection**




Data is shifted out of both shift registers on the programmed clock edge and latched on the opposite edge of the clock.

The master device transmits information out on its SDO output pin which is connected to, and received by, the slave's SDI input pin. The slave device transmits information out on its SDO output pin, which is connected to, and received by, the master's SDI input pin.

To begin communication, the master device first sends out the clock signal. Both the master and the slave devices should be configured for the same clock polarity.

The master device starts a transmission by sending out the MSb from its shift register. The slave device reads this bit from that same line and saves it into the LSb position of its shift register.

During each SPI clock cycle, a full-duplex data transmission occurs. This means that while the master device is sending out the MSb from its shift register (on its SDO pin) and the slave device is reading this bit and saving it as the LSb of its shift register, that the slave device is also sending out the MSb from its shift register (on its SDO pin) and the master device is reading this bit and saving it as the LSb of its shift register.

After eight bits have been shifted out, the master and slave have exchanged register values.

If there is more data to exchange, the shift registers are loaded with new data and the process repeats itself.

Whether the data is meaningful or not (dummy data), depends on the application software. This leads to three scenarios for data transmission:

- Master sends useful data and slave sends dummy data.
- Master sends useful data and slave sends useful data.

#### 35.6.7.2 SSPOV Status Flag

In receive operation, the SSPOV bit is set when eight bits are received into the SSPSR while the BF flag bit is already set from a previous reception.

#### 35.6.7.3 WCOL Status Flag

If the user writes the SSPxBUF when a receive is already in progress (i.e., SSPSR is still shifting in a data byte), the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

#### 35.6.7.4 Typical Receive Sequence:

1. The user generates a Start condition by setting the SEN bit.
2. SSPxIF is set by hardware on completion of the Start.
3. SSPxIF is cleared by software.
4. User writes SSPxBUF with the slave address to transmit and the R/W bit set.
5. Address is shifted out the SDA pin until all eight bits are transmitted. Transmission begins as soon as SSPxBUF is written to.
6. The MSSP module shifts in the $\overline{\text{ACK}}$ bit from the slave device and writes its value into the ACKSTAT bit.
7. The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPxIF bit.
8. User sets the RCEN bit and the master clocks in a byte from the slave.
9. After the eighth falling edge of SCL, SSPxIF and BF are set.
10. Master clears SSPxIF and reads the received byte from SSPUF which clears BF.
11. Master sets the $\overline{\text{ACK}}$ value to be sent to slave in the ACKDT bit and initiates the $\overline{\text{ACK}}$ by setting the ACKEN bit.
12. Master's $\overline{\text{ACK}}$ is clocked out to the slave and SSPxIF is set.
13. User clears SSPxIF.
14. Steps 8-13 are repeated for each received byte from the slave.
15. Master sends a not $\overline{\text{ACK}}$ or Stop to end communication.

**Figure 37-3. Gated Timer Mode, Repeat Acquisition Timing Diagram**

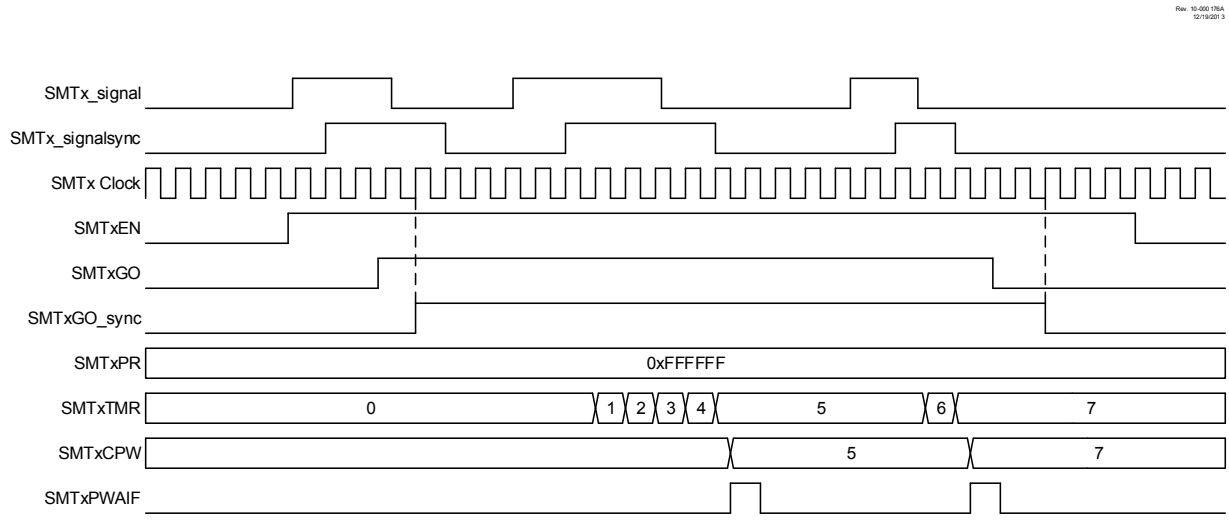
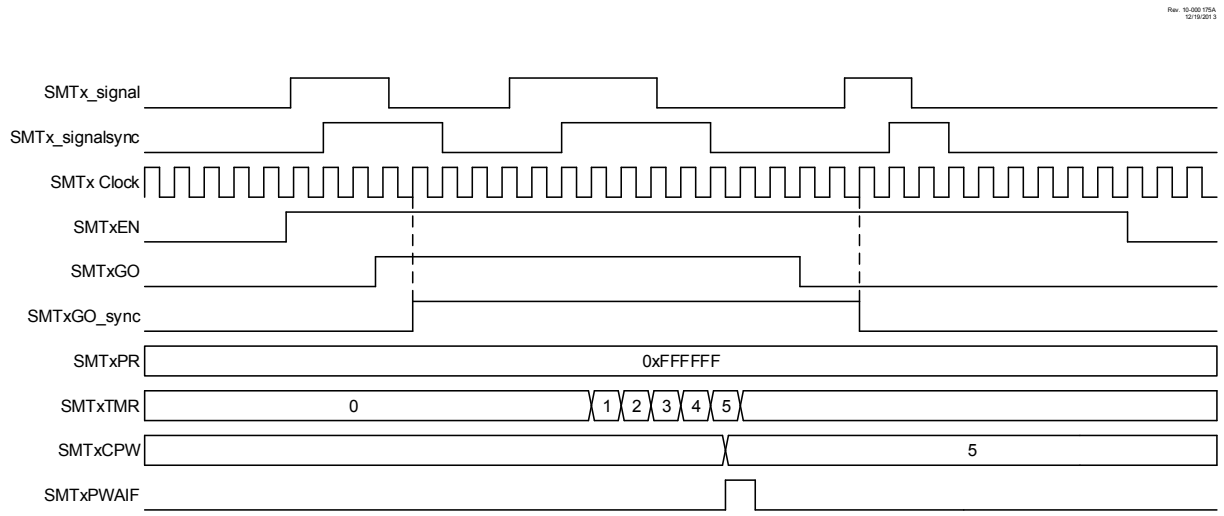



**Figure 37-4. Gated Timer Mode, Single Acquisition Timing Diagram**




### 37.1.6.3 Period and Duty Cycle Measurement Mode

In this mode, either the duty cycle or period (depending on polarity) of the input signal can be acquired relative to the SMT clock. The CPW register is updated on a falling edge of the signal, and the CPR register is updated on a rising edge of the signal, along with the SMTxTMR resetting to 0x000001. In addition, the SMTxGO bit is reset on a rising edge when the SMT is in single acquisition mode. See figures below.

| WSEL<4:0> | SMT1 Window Source |
|---|---|
| 00010 | MFINTOSC (31.25kHz) |
| 00001 | LFINTOSC (31.25kHz) |
| 00000 | Pin Selected by SMT1WINPPS |

| Offset | Name | Bit Pos. | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x088D | OSCCON1 | 7:0 | | NOSC[2:0] | | | NDIV[3:0] | | | |
| 0x088E | OSCCON2 | 7:0 | | COSC[2:0] | | | CDIV[3:0] | | | |
| 0x088F | OSCCON3 | 7:0 | CSWHOLD | SOSCPWR | | ORDY | NOSCR | | | |
| 0x0890 | OSCSTAT | 7:0 | EXTOR | HFOR | MFOR | LFOR | SOR | ADOR | | PLLR |
| 0x0891 | OSCEN | 7:0 | EXTOEN | HFOEN | MFOEN | LFOEN | SOSCEN | ADOEN | | |
| 0x0892 | OSCTUNE | 7:0 | | | HFTUN[5:0] | | | | | |
| 0x0893 | OSCFRQ | 7:0 | | | | | | HFFRQ[2:0] | | |
| 0x0894 | Reserved | | | | | | | | | |
| 0x0895 | CLKRCON | 7:0 | EN | | | DC[1:0] | | DIV[2:0] | | |
| 0x0896 | CLKRCLK | 7:0 | | | | | CLK[3:0] | | | |
| 0x0897 | MD1CON0 | 7:0 | EN | | OUT | OPOL | | | | BIT |
| 0x0898 | MD1CON1 | 7:0 | | | CHPOL | CHSYNC | | | CLPOL | CLSYNC |
| 0x0899 | MD1SRC | 7:0 | | | | SRCS[4:0] | | | | |
| 0x089A | MD1CARL | 7:0 | | | | | CLS[3:0] | | | |
| 0x089B | MD1CARH | 7:0 | | | | | CHS[3:0] | | | |
| 0x089C ... 0x08FF | Reserved | | | | | | | | | |
| 0x0900 | INDF0 | 7:0 | INDF0[7:0] | | | | | | | |
| 0x0901 | INDF1 | 7:0 | INDF1[7:0] | | | | | | | |
| 0x0902 | PCL | 7:0 | PCL[7:0] | | | | | | | |
| 0x0903 | STATUS | 7:0 | | | | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C |
| 0x0904 | FSR0 | 7:0 | FSRL[7:0] | | | | | | | |
| | | 15:8 | FSRH[7:0] | | | | | | | |
| 0x0906 | FSR1 | 7:0 | FSRL[7:0] | | | | | | | |
| | | 15:8 | FSRH[7:0] | | | | | | | |
| 0x0908 | BSR | 7:0 | | | BSR[5:0] | | | | | |
| 0x0909 | WREG | 7:0 | WREG[7:0] | | | | | | | |
| 0x090A | PCLATH | 7:0 | | PCLATH[6:0] | | | | | | |
| 0x090B | INTCON | 7:0 | GIE | PEIE | | | | | | INTEDG |
| 0x090C | FVRCON | 7:0 | FVREN | FVRRDY | TSEN | TSRNG | CDAFVR[1:0] | | ADFVR[1:0] | |
| 0x090D | Reserved | | | | | | | | | |
| 0x090E | DAC1CON0 | 7:0 | EN | | OE1 | | PSS[1:0] | | | NSS |
| 0x090F | DAC1CON1 | 7:0 | | | | DAC1R[4:0] | | | | |
| 0x0910 ... 0x091E | Reserved | | | | | | | | | |
| 0x091F | ZCDCON | 7:0 | SEN | | OUT | POL | | | INTP | INTN |
| 0x0920 ... 0x097F | Reserved | | | | | | | | | |
| 0x0980 | INDF0 | 7:0 | INDF0[7:0] | | | | | | | |
| 0x0981 | INDF1 | 7:0 | INDF1[7:0] | | | | | | | |
| 0x0982 | PCL | 7:0 | PCL[7:0] | | | | | | | |
| 0x0983 | STATUS | 7:0 | | | | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C |
| 0x0984 | FSR0 | 7:0 | FSRL[7:0] | | | | | | | |

| Offset | Name | Bit Pos. | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x0D8B | INTCON | 7:0 | GIE | PEIE | | | | | | INTEDG |
| 0x0D8C ... 0x0DFF | Reserved | | | | | | | | | |
| 0x0E00 | INDF0 | 7:0 | INDF0[7:0] | | | | | | | |
| 0x0E01 | INDF1 | 7:0 | INDF1[7:0] | | | | | | | |
| 0x0E02 | PCL | 7:0 | PCL[7:0] | | | | | | | |
| 0x0E03 | STATUS | 7:0 | | | | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C |
| 0x0E04 | FSR0 | 7:0 | FSRL[7:0] | | | | | | | |
| | | 15:8 | FSRH[7:0] | | | | | | | |
| 0x0E06 | FSR1 | 7:0 | FSRL[7:0] | | | | | | | |
| | | 15:8 | FSRH[7:0] | | | | | | | |
| 0x0E08 | BSR | 7:0 | | | BSR[5:0] | | | | | |
| 0x0E09 | WREG | 7:0 | WREG[7:0] | | | | | | | |
| 0x0E0A | PCLATH | 7:0 | | PCLATH[6:0] | | | | | | |
| 0x0E0B | INTCON | 7:0 | GIE | PEIE | | | | | | INTEDG |
| 0x0E0C ... 0x0E7F | Reserved | | | | | | | | | |
| 0x0E80 | INDF0 | 7:0 | INDF0[7:0] | | | | | | | |
| 0x0E81 | INDF1 | 7:0 | INDF1[7:0] | | | | | | | |
| 0x0E82 | PCL | 7:0 | PCL[7:0] | | | | | | | |
| 0x0E83 | STATUS | 7:0 | | | | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C |
| 0x0E84 | FSR0 | 7:0 | FSRL[7:0] | | | | | | | |
| | | 15:8 | FSRH[7:0] | | | | | | | |
| 0x0E86 | FSR1 | 7:0 | FSRL[7:0] | | | | | | | |
| | | 15:8 | FSRH[7:0] | | | | | | | |
| 0x0E88 | BSR | 7:0 | | | BSR[5:0] | | | | | |
| 0x0E89 | WREG | 7:0 | WREG[7:0] | | | | | | | |
| 0x0E8A | PCLATH | 7:0 | | PCLATH[6:0] | | | | | | |
| 0x0E8B | INTCON | 7:0 | GIE | PEIE | | | | | | INTEDG |
| 0x0E8C ... 0x0EFF | Reserved | | | | | | | | | |
| 0x0F00 | INDF0 | 7:0 | INDF0[7:0] | | | | | | | |
| 0x0F01 | INDF1 | 7:0 | INDF1[7:0] | | | | | | | |
| 0x0F02 | PCL | 7:0 | PCL[7:0] | | | | | | | |
| 0x0F03 | STATUS | 7:0 | | | | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C |
| 0x0F04 | FSR0 | 7:0 | FSRL[7:0] | | | | | | | |
| | | 15:8 | FSRH[7:0] | | | | | | | |
| 0x0F06 | FSR1 | 7:0 | FSRL[7:0] | | | | | | | |
| | | 15:8 | FSRH[7:0] | | | | | | | |
| 0x0F08 | BSR | 7:0 | | | BSR[5:0] | | | | | |
| 0x0F09 | WREG | 7:0 | WREG[7:0] | | | | | | | |
| 0x0F0A | PCLATH | 7:0 | | PCLATH[6:0] | | | | | | |
| 0x0F0B | INTCON | 7:0 | GIE | PEIE | | | | | | INTEDG |

| Offset | Name | Bit Pos. | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x1F56 ... 0x1F7F | Reserved | | | | | | | | | |
| 0x1F80 | INDF0 | 7:0 | INDF0[7:0] | | | | | | | |
| 0x1F81 | INDF1 | 7:0 | INDF1[7:0] | | | | | | | |
| 0x1F82 | PCL | 7:0 | PCL[7:0] | | | | | | | |
| 0x1F83 | STATUS | 7:0 | | | | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C |
| 0x1F84 | FSR0 | 7:0 | FSRL[7:0] | | | | | | | |
| | | 15:8 | FSRH[7:0] | | | | | | | |
| 0x1F86 | FSR1 | 7:0 | FSRL[7:0] | | | | | | | |
| | | 15:8 | FSRH[7:0] | | | | | | | |
| 0x1F88 | BSR | 7:0 | | | BSR[5:0] | | | | | |
| 0x1F89 | WREG | 7:0 | WREG[7:0] | | | | | | | |
| 0x1F8A | PCLATH | 7:0 | | PCLATH[6:0] | | | | | | |
| 0x1F8B | INTCON | 7:0 | GIE | PEIE | | | | | | INTEDG |
| 0x1F8C ... 0x1FE3 | Reserved | | | | | | | | | |
| 0x1FE4 | STATUS_SHAD | 7:0 | | | | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C |
| 0x1FE5 | WREG_SHAD | 7:0 | WREG[7:0] | | | | | | | |
| 0x1FE6 | BSR_SHAD | 7:0 | | | BSR[5:0] | | | | | |
| 0x1FE7 | PCLATH_SHAD | 7:0 | | PCLATH[6:0] | | | | | | |
| 0x1FE8 | FSR0_SHAD | 7:0 | FSRL[7:0] | | | | | | | |
| | | 15:8 | FSRH[7:0] | | | | | | | |
| 0x1FEA | FSR1_SHAD | 7:0 | FSRL[7:0] | | | | | | | |
| | | 15:8 | FSRH[7:0] | | | | | | | |

| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|---|---|---|---|---|---|---|---|
| Param No. | Sym. | Characteristic | Min. | Typ. † | Max. | Units | Conditions |
| OS56 | $T_{LFOSCST}$ | LFINTOSC Wake-up from Sleep Start-up Time | — | 0.2 | — | ms | |

† - Data in “Typ” column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note:**

1. To ensure these oscillator frequency tolerances, $V_{DD}$ and $V_{SS}$ must be capacitively decoupled as close to the device as possible. 0.1 µF and 0.01 µF values in parallel are recommended.
2. See the figure below.

**Figure 42-6. Precision Calibrated HFINTOSC Frequency Accuracy Over Device $V_{DD}$ and Temperature**