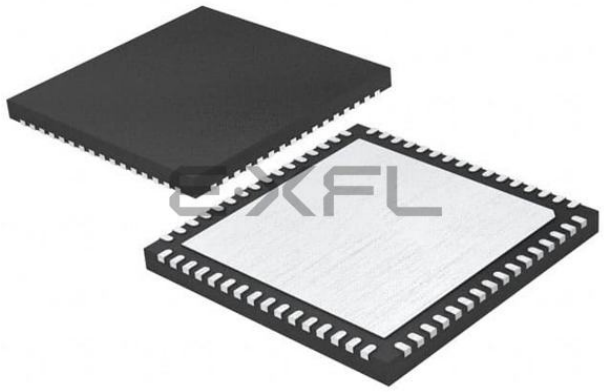


Welcome to E-XFL.COM

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	64MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	53
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 16x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad
Supplier Device Package	64-VQFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f66k22-i-mr

3.3 Clock Sources and Oscillator Switching

Essentially, PIC18F87K22 family devices have these independent clock sources:

- Primary oscillators
- Secondary oscillators
- Internal oscillator

The **primary oscillators** can be thought of as the main device oscillators. These are any external oscillators connected to the OSC1 and OSC2 pins, and include the External Crystal and Resonator modes and the External Clock modes. If selected by the FOSC<3:0> Configuration bits (CONFIG1H<3:0>), the internal oscillator block may be considered a primary oscillator. The internal oscillator block can be one of the following:

- 31 kHz LF-INTRC source
- 31 kHz to 500 kHz MF-INTOSC source
- 31 kHz to 16 MHz HF-INTOSC source

The particular mode is defined by the FOSC Configuration bits. The details of these modes are covered in **Section 3.5 "External Oscillator Modes"**.

The **secondary oscillators** are external clock sources that are not connected to the OSC1 or OSC2 pin. These sources may continue to operate, even after the controller is placed in a power-managed mode. PIC18F87K22 family devices offer the SOSC (Timer1/3/5/7) oscillator as a secondary oscillator source. This oscillator, in all power-managed modes, is often the time base for functions, such as a Real-Time Clock (RTC).

The SOSC can be enabled from any peripheral that requests it. There are eight ways the SOSC can be enabled: if the SOSC is selected as the source by any of the odd timers, which is done by each respective SOSCEN bit (TxCON<3>), if the SOSC is selected as the RTCC source by the RTCOSC Configuration bit (CONFIG3L<1>), if the SOSC is selected as the CPU clock source by the SCS bits (OSCCON<1:0>) or if the SOSCGO bit is set (OSCCON2<3>). The SOSCGO bit is used to warm up the SOSC so that it is ready before any peripheral requests it.

The secondary oscillator has three Run modes. The SOSCSEL<1:0> bits (CONFIG1L<4:3>) decide the SOSC mode of operation:

- 11 = High-power SOSC circuit
- 10 = Digital (SCLKI) mode
- 01 = Low-power SOSC circuit

If a secondary oscillator is not desired and digital I/O on port pins, RC0 and RC1, is needed, the SOSCSEL bits must be set to Digital mode.

In addition to being a primary clock source in some circumstances, the **internal oscillator** is available as a power-managed mode clock source. The LF-INTOSC source is also used as the clock source for several special features, such as the WDT and Fail-Safe Clock Monitor. The internal oscillator block is discussed in more detail in **Section 3.6 "Internal Oscillator Block"**.

The PIC18F87K22 family includes features that allow the device clock source to be switched from the main oscillator, chosen by device configuration, to one of the alternate clock sources. When an alternate clock source is enabled, various power-managed operating modes are available.

3.3.1 OSC1/OSC2 OSCILLATOR

The OSC1/OSC2 oscillator block is used to provide the oscillator modes and frequency ranges:

Mode	Design Operating Frequency
LP	31.25-100 kHz
XT	100 kHz to 4 MHz
HS	4 MHz to 25 MHz
EC	0 to 64 MHz (external clock)
EXTRC	0 to 4 MHz (external RC)

The crystal-based oscillators (XT, HS and LP) have a built-in start-up time. The operation of the EC and EXTRC clocks is immediate.

3.3.2 CLOCK SOURCE SELECTION

The System Clock Select bits, SCS<1:0> (OSCCON2<1:0>), select the clock source. The available clock sources are the primary clock defined by the FOSC<3:0> Configuration bits, the secondary clock (SOSC oscillator) and the internal oscillator. The clock source changes after one or more of the bits is written to, following a brief clock transition interval.

The OSTS (OSCCON<3>) and SOSCRUN (OSCCON<6>) bits indicate which clock source is currently providing the device clock. The OSTS bit indicates that the Oscillator Start-up Timer (OST) has timed out and the primary clock is providing the device clock in primary clock modes. The SOSCRUN bit indicates when the SOSC oscillator (from Timer1/3/5/7) is providing the device clock in secondary clock modes. In power-managed modes, only one of these bits will be set at any time. If neither of these bits is set, the INTRC is providing the clock, or the internal oscillator has just started and is not yet stable.

The IDLEN bit (OSCCON<7>) determines if the device goes into Sleep mode or one of the Idle modes when the SLEEP instruction is executed.

REGISTER 4-2: PMD2: PERIPHERAL MODULE DISABLE REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TMR10MD(1)	TMR8MD	TMR7MD(1)	TMR6MD	TMR5MD	CMP3MD	CMP2MD	CMP1MD
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **TMR10MD:** TMR10MD Disable bit(1)
1 = Peripheral Module Disable (PMD) is enabled and all TMR10MD clock sources are disabled
0 = PMD is disabled and TMR10MD is enabled

bit 6 **TMR8MD:** TMR8MD Disable bit
1 = PMD is enabled and all TMR8MD clock sources are disabled
0 = PMD is disabled and TMR8MD is enabled

bit 5 **TMR7MD:** TMR7MD Disable bit(1)
1 = PMD is enabled and all TMR7MD clock sources are disabled
0 = PMD is disabled and TMR7MD is enabled

bit 4 **TMR6MD:** TMR6MD Disable bit
1 = PMD is enabled and all TMR6MD clock sources are disabled
0 = PMD is disabled and TMR6MD is enabled

bit 3 **TMR5MD:** TMR5MD Disable bit
1 = PMD is enabled and all TMR5MD clock sources are disabled
0 = PMD is disabled and TMR5MD is enabled

bit 2 **CMP3MD:** PMD Comparator 3 Enable/Disable bit
1 = PMD is enabled for Comparator 3, disabling all of its clock sources
0 = PMD is disabled for Comparator 3

bit 1 **CMP2MD:** PMD Comparator 3 Enable/Disable bit
1 = PMD is enabled for Comparator 2, disabling all of its clock sources
0 = PMD is disabled for Comparator 2

bit 0 **CMP1MD:** PMD Comparator 3 Enable/Disable bit
1 = PMD is enabled for Comparator 1, disabling all of its clock sources
0 = PMD is disabled for Comparator 1

Note 1: Unimplemented on devices with a program memory of 32 Kbytes (PIC18FX5K22).

5.2 Master Clear ($\overline{MCLR}$)

The $\overline{MCLR}$ pin provides a method for triggering a hard external Reset of the device. A Reset is generated by holding the pin low. PIC18 extended microcontroller devices have a noise filter in the $\overline{MCLR}$ Reset path which detects and ignores small pulses.

The $\overline{MCLR}$ pin is not driven low by any internal Resets, including the WDT.

5.3 Power-on Reset (POR)

A Power-on Reset condition is generated on-chip whenever VDD rises above a certain threshold. This allows the device to start in the initialized state when VDD is adequate for operation.

To take advantage of the POR circuitry, tie the $\overline{MCLR}$ pin through a resistor (1 kΩ to 10 kΩ) to VDD. This will eliminate external RC components usually needed to create a Power-on Reset delay. A minimum rise rate for VDD is specified (Parameter D004). For a slow rise time, see Figure 5-2.

When the device starts normal operation (exiting the Reset condition), device operating parameters (such as voltage, frequency and temperature) must be met to ensure operation. If these conditions are not met, the device must be held in Reset until the operating conditions are met.

Power-on Reset events are captured by the $\overline{POR}$ bit (RCON<1>). The state of the bit is set to '0' whenever a Power-on Reset occurs and does not change for any other Reset event. $\overline{POR}$ is not reset to '1' by any hardware event. To capture multiple events, the user manually resets the bit to '1' in software following any Power-on Reset.

5.4 Brown-out Reset (BOR)

The PIC18F87K22 family has four BOR Power modes:

- High-Power BOR
- Medium Power BOR
- Low-Power BOR
- Zero-Power BOR

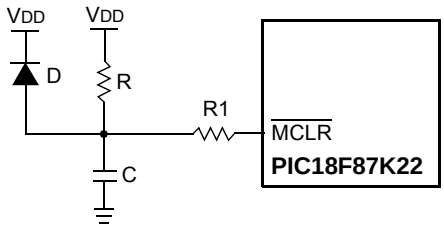
Each power Mode is selected by the BORPWR<1:0> bits setting (CONFIG2L<6:5>). For low, medium and high-power BOR, the module monitors the VDD depending on the BORV<1:0> setting (CONFIG1L<3:2>). A BOR event re-arms the Power-on Reset. It also causes a Reset, depending on which of the trip levels has been set: 1.8V, 2V, 2.7V or 3V.

BOR is enabled by the BOREN<1:0> bits (CONFIG2L<2:1>) and the SBOREN bit (RCON<6>). Typical power consumption is listed as Parameter D022A in **Section 31.0 "Electrical Characteristics"**.

In Zero-Power BOR (ZPBORMV), the module monitors the VDD voltage and re-arms the POR at about 2V. ZPBORMV does not cause a Reset, but re-arms the POR.

The BOR accuracy varies with its power level. The lower the power setting, the less accurate the BOR trip levels are. Therefore, the high-power BOR has the highest accuracy and the low-power BOR has the lowest accuracy. The trip levels (BVDD, Parameter D005), current consumption (**Section 31.2 "DC Characteristics: Power-Down and Supply Current PIC18F87K22 Family (Industrial/Extended)"**) and time required below BVDD (TBOR, Parameter 35) can all be found in **Section 31.0 "Electrical Characteristics"**.

FIGURE 5-2: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD POWER-UP)



Note 1: External Power-on Reset circuit is required only if the VDD power-up slope is too slow. The diode, D, helps discharge the capacitor quickly when VDD powers down.

2: R < 40 kΩ is recommended to make sure that the voltage drop across R does not violate the device's electrical specification.

3: R1 ≥ 1 kΩ will limit any current flowing into $\overline{MCLR}$ from external capacitor, C, in the event of $\overline{MCLR}$/VPP pin breakdown due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS).

5.4.1 DETECTING BOR

The $\overline{BOR}$ bit always resets to '0' on any Brown-out Reset or Power-on Reset event. This makes it difficult to determine if a Brown-out Reset event has occurred just by reading the state of $\overline{BOR}$ alone. A more reliable method is to simultaneously check the state of both $\overline{POR}$ and $\overline{BOR}$. This assumes that the $\overline{POR}$ bit is reset to '1' in software immediately after any Power-on Reset event. If $\overline{BOR}$ is '0' while $\overline{POR}$ is '1', it can be reliably assumed that a Brown-out Reset event has occurred.

LP-BOR cannot be detected with the $\overline{BOR}$ bit in the RCON register. LP-BOR can rearm the $\overline{POR}$ and can cause a Power-on Reset.

EXAMPLE 7-3: WRITING TO FLASH PROGRAM MEMORY (CONTINUED)

```
PROGRAM_MEMORY
                BSF     EECON1, EEPGD       ; point to Flash program memory
                BCF     EECON1, CFGS        ; access Flash program memory
                BSF     EECON1, WREN        ; enable write to memory
                BCF     INTCON, GIE         ; disable interrupts
                MOVLW   0x55
    Required    MOVWF   EECON2              ; write 55h
    Sequence    MOVLW   0xAA
                MOVWF   EECON2              ; write 0AAh
                BSF     EECON1, WR          ; start program (CPU stall)
                BSF     INTCON, GIE         ; re-enable interrupts
                BCF     EECON1, WREN        ; disable write to memory
```

7.5.2 WRITE VERIFY

Depending on the application, good programming practice may dictate that the value written to the memory should be verified against the original value. This should be used in applications where excessive writes can stress bits near the specification limit.

7.5.3 UNEXPECTED TERMINATION OF WRITE OPERATION

If a write is terminated by an unplanned event, such as loss of power or an unexpected Reset, the memory location just programmed should be verified and reprogrammed if needed. If the write operation is interrupted by a $\overline{\text{MCLR}}$ Reset or a WDT Time-out Reset during normal operation, the user can check the WRERR bit and rewrite the location(s) as needed.

7.5.4 PROTECTION AGAINST SPURIOUS WRITES

To protect against spurious writes to Flash program memory, the write initiate sequence must also be followed. See **Section 28.0 “Special Features of the CPU”** for more details.

7.6 Flash Program Operation During Code Protection

See **Section 28.6 “Program Verification and Code Protection”** for details on code protection of Flash program memory.

TABLE 7-2: REGISTERS ASSOCIATED WITH PROGRAM FLASH MEMORY

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TBLPTRU	—	—	bit 21(1)	Program Memory Table Pointer Upper Byte (TBLPTR<20:16>)				
TBPLTRH	Program Memory Table Pointer High Byte (TBLPTR<15:8>)							
TBLPTRL	Program Memory Table Pointer Low Byte (TBLPTR<7:0>)							
TABLAT	Program Memory Table Latch							
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
EECON2	EEPROM Control Register 2 (not a physical register)							
EECON1	EEPGD	CFGS	—	FREE	WRERR	WREN	WR	RD
IPR6	—	—	—	EEIP	—	CMP3IP	CMP2IP	CMP1IP
PIR6	—	—	—	EEIF	—	CMP3IF	CMP2IF	CMP1IF
PIE6	—	—	—	EEIE	—	CMP3IE	CMP2IE	CMP1IE

Legend: — = unimplemented, read as ‘0’. Shaded cells are not used during Flash/EEPROM access.

Note 1: Bit 21 of the TBLPTRU allows access to the device Configuration bits.

16.0 TIMER3/5/7 MODULES

The Timer3/5/7 timer/counter modules incorporate these features:

- Software-selectable operation as a 16-bit timer or counter
- Readable and writable eight-bit registers (TMRxH and TMRxL)
- Selectable clock source (internal or external) with device clock or SOSC oscillator internal options
- Interrupt-on-overflow
- Module Reset on ECCP Special Event Trigger

Timer7 is unimplemented for devices with a program memory of 32 Kbytes (PIC18FX5K22).

Note: Throughout this section, generic references are used for register and bit names that are the same – except for an 'x' variable that indicates the item's association with the Timer3, Timer5 or Timer7 module. For example, the control register is named TxCON and refers to T3CON, T5CON and T7CON.

A simplified block diagram of the Timer3/5/7 module is shown in Figure 16-1.

The Timer3/5/7 module is controlled through the TxCON register (Register 16-1). It also selects the clock source options for the ECCP modules. (For more information, see **Section 20.1.1 "ECCP Module and Timer Resources"**.)

The FOSC clock source should not be used with the ECCP capture/compare features. If the timer will be used with the capture or compare features, always select one of the other timer clocking options.

REGISTER 17-1: TxCON: TIMERx CONTROL REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TxOUTPS3	TxOUTPS2	TxOUTPS1	TxOUTPS0	TMRxON	TxCKPS1	TxCKPS0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **Unimplemented:** Read as '0'

bit 6-3 **TxOUTPS<3:0>:** Timerx Output Postscale Select bits
0000 = 1:1 Postscale
0001 = 1:2 Postscale
•
•
•
1111 = 1:16 Postscale

bit 2 **TMRxON**: Timerx On bit
1 = Timerx is on
0 = Timerx is off

bit 1-0 **TxCKPS<1:0>:** Timerx Clock Prescale Select bits
00 = Prescaler is 1
01 = Prescaler is 4
1x = Prescaler is 16

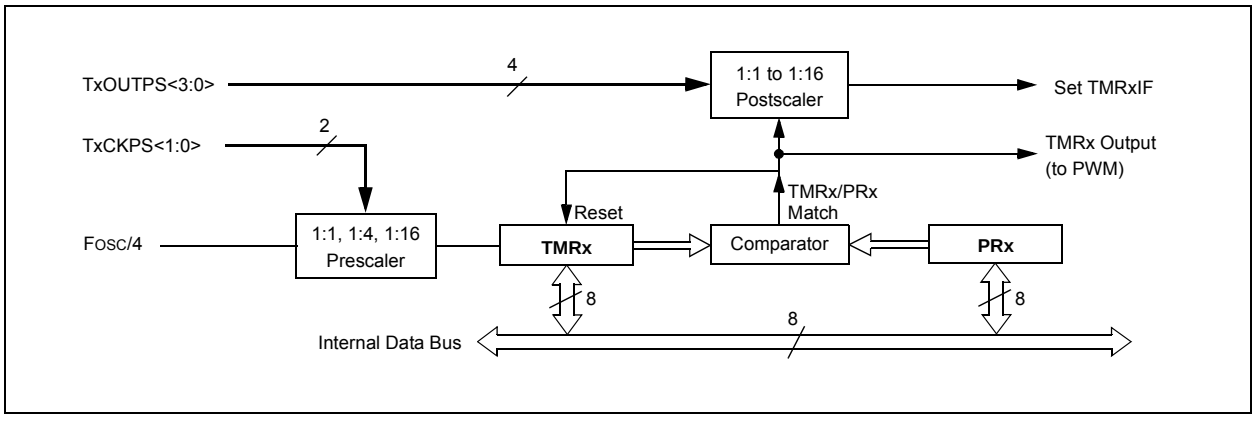
17.2 Timer4/6/8/10/12 Interrupt

The Timer4/6/8/10/12 modules have eight-bit Period registers, PRx, that are both readable and writable. Timer4/6/8/10/12 increment from 00h until they match PR4/6/8/10/12 and then reset to 00h on the next increment cycle. The PRx registers are initialized to FFh upon Reset.

17.3 Output of TMRx

The outputs of TMRx (before the postscaler) are used only as a PWM time base for the ECCP modules. They are not used as baud rate clocks for the MSSP modules as is the Timer2 output.

FIGURE 17-1: TIMER4 BLOCK DIAGRAM



19.4.2 PWM DUTY CYCLE

The PWM duty cycle is specified to use CCP4, as an example, by writing to the CCPR4L register and to the CCP4CON<5:4> bits. Up to 10-bit resolution is available. The CCPR4L contains the eight MSbs and the CCP4CON<5:4> bits contain the two LSbs. This 10-bit value is represented by CCPR4L:CCP4CON<5:4>. The following equation is used to calculate the PWM duty cycle in time:

EQUATION 19-2:

$$\text{PWM Duty Cycle} = (\text{CCPR4L:CCP4CON<5:4>}) \bullet T_{OSC} \bullet (\text{TMR2 Prescale Value})$$

CCPR4L and CCP4CON<5:4> can be written to at any time, but the duty cycle value is not latched into CCPR4H until after a match between PR2 and TMR2 occurs (that is, the period is complete). In PWM mode, CCPR4H is a read-only register.

The CCPR4H register and a two-bit internal latch are used to double-buffer the PWM duty cycle. This double-buffering is essential for glitchless PWM operation.

When the CCPR4H and two-bit latch match TMR2, concatenated with an internal two-bit Q clock or two bits of the TMR2 prescaler, the CCP4 pin is cleared.

The maximum PWM resolution (bits) for a given PWM frequency is shown in Equation 19-3:

EQUATION 19-3:

$$\text{PWM Resolution (max)} = \frac{\log\left(\frac{F_{OSC}}{F_{PWM}}\right)}{\log(2)} \text{bits}$$

Note: If the PWM duty cycle value is longer than the PWM period, the CCP4 pin will not be cleared.

TABLE 19-6: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 40 MHz

PWM Frequency	2.44 kHz	9.77 kHz	39.06 kHz	156.25 kHz	312.50 kHz	416.67 kHz
Timer Prescaler (1, 4, 16)	16	4	1	1	1	1
PR2 Value	FFh	FFh	FFh	3Fh	1Fh	17h
Maximum Resolution (bits)	10	10	10	8	7	6.58

19.4.3 SETUP FOR PWM OPERATION

To configure the CCP module for PWM operation, using CCP4 as an example:

1. Set the PWM period by writing to the PR2 register.
2. Set the PWM duty cycle by writing to the CCPR4L register and CCP4CON<5:4> bits.
3. Make the CCP4 pin an output by clearing the appropriate TRIS bit.
4. Set the TMR2 prescale value, then enable Timer2 by writing to T2CON.
5. Configure the CCP4 module for PWM operation.

TABLE 19-7: REGISTERS ASSOCIATED WITH PWM AND TIMERS

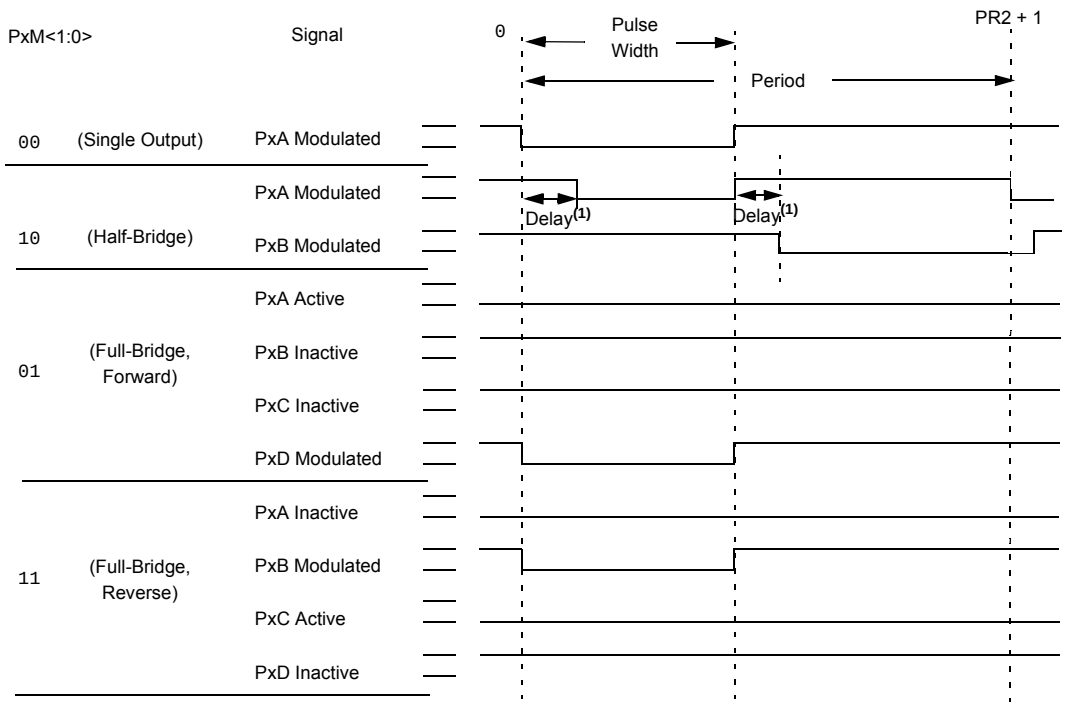
Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
RCON	IPEN	SBOREN	$\overline{CM}$	$\overline{RI}$	$\overline{TO}$	$\overline{PD}$	$\overline{POR}$	$\overline{BOR}$
PIR4	CCP10IF(1)	CCP9IF(1)	CCP8IF	CCP7IF	CCP6IF	CCP5IF	CCP4IF	CCP3IF
PIE4	CCP10IE(1)	CCP9IE(1)	CCP8IE	CCP7IE	CCP6IE	CCP5IE	CCP4IE	CCP3IE
IPR4	CCP10IP(1)	CCP9IP(1)	CCP8IP	CCP7IP	CCP6IP	CCP5IP	CCP4IP	CCP3IP
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
TRISE	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0

Legend: — = unimplemented, read as '0'. Shaded cells are not used by PWM or Timer2/4/6/8.

Note 1: Unimplemented on devices with a program memory of 32 Kbytes (PIC18F65K22 and PIC18F85K22).

2: Unimplemented on 64-pin devices (PIC18F6XK22), read as '0'.

FIGURE 20-5: EXAMPLE ENHANCED PWM OUTPUT RELATIONSHIPS (ACTIVE-LOW STATE)



Relationships:

- Period = 4 * TOSC * (PR2 + 1) * (TMR2 Prescale Value)
- Pulse Width = TOSC * (CCPRxL<7:0>:CCPxCON<5:4>) * (TMR2 Prescale Value)
- Delay = 4 * TOSC * (ECCPxDEL<6:0>)

Note 1: Dead-band delay is programmed using the ECCP1DEL register (see **Section 20.4.6 "Programmable Dead-Band Delay Mode"**).

21.4.2 OPERATION

The MSSP module functions are enabled by setting the MSSP Enable bit, SSPEN (SSPxCON1<5>).

The SSPxCON1 register allows control of the I^2C operation. Four mode selection bits (SSPxCON1<3:0>) allow one of the following I^2C modes to be selected:

- I^2C Master mode, clock
- I^2C Slave mode (7-bit address)
- I^2C Slave mode (10-bit address)
- I^2C Slave mode (7-bit address) with Start and Stop bit interrupts enabled
- I^2C Slave mode (10-bit address) with Start and Stop bit interrupts enabled
- I^2C Firmware Controlled Master mode, slave is Idle

Selection of any I^2C mode with the SSPEN bit set forces the SCLx and SDAx pins to be open-drain, provided these pins are programmed as inputs by setting the appropriate TRISC or TRISD bits. To ensure proper operation of the module, pull-up resistors must be provided externally to the SCLx and SDAx pins.

21.4.3 SLAVE MODE

In Slave mode, the SCLx and SDAx pins must be configured as inputs (TRISC<4:3> set). The MSSP module will override the input state with the output data when required (slave-transmitter).

The I^2C Slave mode hardware will always generate an interrupt on an address match. Address masking will allow the hardware to generate an interrupt for more than one address (up to 31 in 7-bit addressing and up to 63 in 10-bit addressing). Through the mode select bits, the user can also choose to interrupt on Start and Stop bits.

When an address is matched, or the data transfer after an address match is received, the hardware automatically will generate the Acknowledge ($\overline{ACK}$) pulse and load the SSPxBUF register with the received value currently in the SSPxSR register.

Any combination of the following conditions will cause the MSSP module not to give this $\overline{ACK}$ pulse:

- The Buffer Full bit, BF (SSPxSTAT<0>), was set before the transfer was received.
- The overflow bit, SSPOV (SSPxCON1<6>), was set before the transfer was received.

In this case, the SSPxSR register value is not loaded into the SSPxBUF, but bit, SSPxIF, is set. The BF bit is cleared by reading the SSPxBUF register, while bit, SSPOV, is cleared through software.

The SCLx clock input must have a minimum high and low for proper operation. The high and low times of the I^2C specification, as well as the requirement of the MSSP module, are shown in timing Parameter 100 and Parameter 101.

21.4.3.1 Addressing

Once the MSSP module has been enabled, it waits for a Start condition to occur. Following the Start condition, the 8 bits are shifted into the SSPxSR register. All incoming bits are sampled with the rising edge of the clock (SCLx) line. The value of register, SSPxSR<7:1>, is compared to the value of the SSPxADD register. The address is compared on the falling edge of the eighth clock (SCLx) pulse. If the addresses match and the BF and SSPOV bits are clear, the following events occur:

1. The SSPxSR register value is loaded into the SSPxBUF register.
2. The Buffer Full bit, BF, is set.
3. An $\overline{ACK}$ pulse is generated.
4. The MSSP Interrupt Flag bit, SSPxIF, is set (and an interrupt is generated, if enabled) on the falling edge of the ninth SCLx pulse.

In 10-Bit Addressing mode, two address bytes need to be received by the slave. The five Most Significant bits (MSbs) of the first address byte specify if this is a 10-bit address. The R/$\overline{W}$ (SSPxSTAT<2>) bit must specify a write so the slave device will receive the second address byte. For a 10-bit address, the first byte would equal '`11110 A9 A8 0`', where '`A9`' and '`A8`' are the two MSbs of the address. The sequence of events for 10-bit addressing is as follows, with Steps 7 through 9 for the slave-transmitter:

1. Receive first (high) byte of address (bits, SSPxIF, BF and UA, are set on address match).
2. Update the SSPxADD register with second (low) byte of address (clears bit, UA, and releases the SCLx line).
3. Read the SSPxBUF register (clears bit, BF) and clear flag bit, SSPxIF.
4. Receive second (low) byte of address (bits, SSPxIF, BF and UA, are set).
5. Update the SSPxADD register with the first (high) byte of address. If match releases SCLx line, this will clear bit, UA.
6. Read the SSPxBUF register (clears bit, BF) and clear flag bit, SSPxIF.
7. Receive Repeated Start condition.
8. Receive first (high) byte of address (bits, SSPxIF and BF, are set).
9. Read the SSPxBUF register (clears bit, BF) and clear flag bit, SSPxIF.

24.0 COMPARATOR MODULE

The analog comparator module contains three comparators that can be independently configured in a variety of ways. The inputs can be selected from the analog inputs and two Internal Reference Voltages. The digital outputs are available at the pin level and can also be read through the control register. Multiple output and interrupt event generation are also available. A generic single comparator from the module is shown in Figure 24-1.

Key features of the module includes:

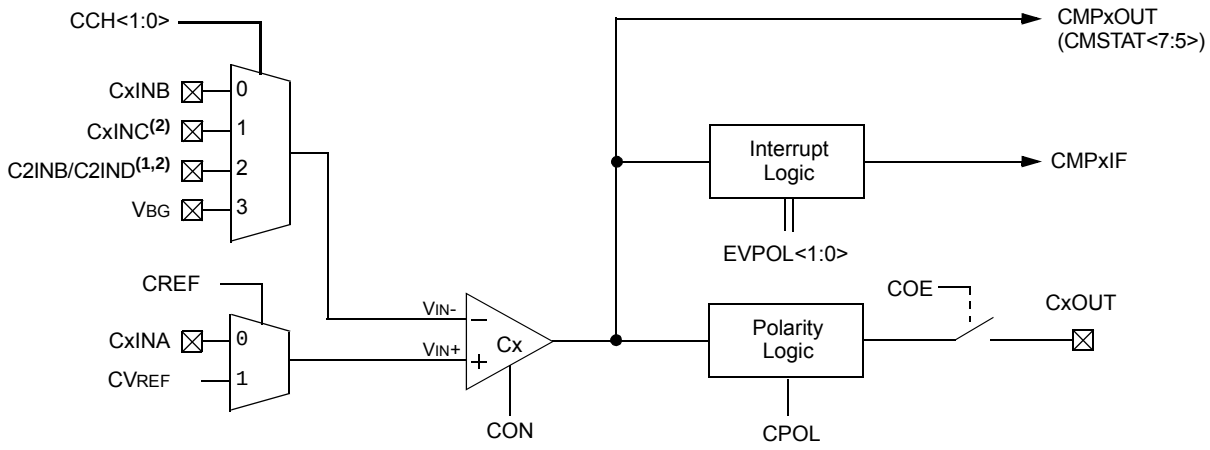
- Independent comparator control
- Programmable input configuration
- Output to both pin and register levels
- Programmable output polarity
- Independent interrupt generation for each comparator with configurable interrupt-on-change

24.1 Registers

The CMxCON registers (CM1CON, CM2CON and CM3CON) select the input and output configuration for each comparator, as well as the settings for interrupt generation (see Register 24-1).

The CMSTAT register (Register 24-2) provides the output results of the comparators. The bits in this register are read-only.

FIGURE 24-1: COMPARATOR SIMPLIFIED BLOCK DIAGRAM



Note 1: Comparators, 1 and 3, use C2INB as an input to the inverting terminal. Comparator 2 uses C2IND as an input to the inverted terminal.

2: C1INC, C2INC and C2IND are all unavailable for 64-pin devices (PIC18F6XK22).

27.2.5 INTERRUPTS

The CTMU sets its interrupt flag (PIR3<3>) whenever the current source is enabled, then disabled. An interrupt is generated only if the corresponding interrupt enable bit (PIE3<3>) is also set. If edge sequencing is not enabled (i.e., Edge 1 must occur before Edge 2), it is necessary to monitor the edge status bits and determine which edge occurred last and caused the interrupt.

27.3 CTMU Module Initialization

The following sequence is a general guideline used to initialize the CTMU module:

1. Select the current source range using the IRNGx bits (CTMUICON<1:0>).
2. Adjust the current source trim using the ITRIMx bits (CTMUICON<7:2>).
3. Configure the edge input sources for Edge 1 and Edge 2 by setting the EDG1SEL and EDG2SEL bits (CTMUCONL<3:2> and <6:5>, respectively).
4. Configure the input polarities for the edge inputs using the EDG2POL and EDG1POL bits (CTMUCONL<7,4>).

   The default configuration is for negative edge polarity (high-to-low transitions).
5. Enable edge sequencing using the EDGSEQEN bit (CTMUCONH<2>).

   By default, edge sequencing is disabled.
6. Select the operating mode (Measurement or Time Delay) with the TGEN bit.

   The default mode is Time/Capacitance Measurement.
7. Configure the module to automatically trigger an A/D conversion when the second edge event has occurred, using the CTTRIG bit (CTMUCONH<0>).

   The conversion trigger is disabled by default.
8. Discharge the connected circuit by setting the IDISSEN bit (CTMUCONH<1>).
9. After waiting a sufficient time for the circuit to discharge, clear IDISSEN.
10. Disable the module by clearing the CTMUEN bit (CTMUCONH<7>).
11. Clear the Edge Status bits, EDG2STAT and EDG1STAT (CTMUCONL<1:0>).
12. Enable both edge inputs by setting the EDGEN bit (CTMUCONH<3>).
13. Enable the module by setting the CTMUEN bit.

Depending on the type of measurement or pulse generation being performed, one or more additional modules may also need to be initialized and configured with the CTMU module:

- Edge Source Generation: In addition to the external edge input pins, CCPx Special Event Triggers can be used as edge sources for the CTMU.
- Capacitance or Time Measurement: The CTMU module uses the A/D Converter to measure the voltage across a capacitor that is connected to one of the analog input channels.
- Pulse Generation: When generating system clock independent, output pulses, the CTMU module uses Comparator 2 and the associated comparator voltage reference.

27.4 Calibrating the CTMU Module

The CTMU requires calibration for precise measurements of capacitance and time, as well as for accurate time delay. If the application only requires measurement of a relative change in capacitance or time, calibration is usually not necessary. An example of a less precise application is a capacitive touch switch, in which the touch circuit has a baseline capacitance and the added capacitance of the human body changes the overall capacitance of a circuit.

If actual capacitance or time measurement is required, two hardware calibrations must take place:

- The current source needs calibration to set it to a precise current.
- The circuit being measured needs calibration to measure or nullify any capacitance other than that to be measured.

27.4.1 CURRENT SOURCE CALIBRATION

The current source on board the CTMU module has a range of ±60% nominal for each of three current ranges. For precise measurements, it is possible to measure and adjust this current source by placing a high-precision resistor, RCAL, onto an unused analog channel. An example circuit is shown in Figure 27-2.

To measure the current source:

1. Initialize the A/D Converter.
2. Initialize the CTMU.
3. Enable the current source by setting EDG1STAT (CTMUCONL<0>).
4. Issue the settling time delay.
5. Perform the A/D conversion.
6. Calculate the current source current using $I = V/R_{CAL}$, where RCAL is a high-precision resistance and V is measured by performing an A/D conversion.

EXAMPLE 27-1: SETUP FOR CTMU CALIBRATION ROUTINES

```
#include "p18cxxx.h"
/*****************************************************************************/
/*Setup CTMU *****************************************************************/
/*****************************************************************************/
void setup(void)

{ //CTMUCON - CTMU Control register

    CTMUCONH = 0x00;              //make sure CTMU is disabled
    CTMUCONL = 0X90;
    //CTMU continues to run when emulator is stopped,CTMU continues
    //to run in idle mode,Time Generation mode disabled, Edges are blocked
    //No edge sequence order, Analog current source not grounded, trigger
    //output disabled, Edge2 polarity = positive level, Edge2 source =
    //source 0, Edge1 polarity = positive level, Edge1 source = source 0,
    // Set Edge status bits to zero


    //CTMUICON - CTMU Current Control Register
    CTMUICON = 0x01;              //0.55uA, Nominal - No Adjustment

/*****************************************************************************/
//Setup AD converter;
/*****************************************************************************/

    TRISA=0x04;                   //set channel 2 as an input

    // Configured AN2 as an analog channel
    // ANCON0
    ANCON0 = 0X04;
    // ANCON1
    ANCON1 = 0XE0;

    // ADCON2
    ADCON2bits.ADFM=1;            // Resulst format 1= Right justified
    ADCON2bits.ACQT=1;            // Acquition time 7 = 20TAD 2 = 4TAD 1=2TAD
    ADCON2bits.ADCS=2;            // Clock conversion bits 6= FOSC/64 2=FOSC/32

    // ADCON0
    ADCON1bits.VCFG0 =0;          // Vref+ = AVdd
    ADCON0bits.VCFG1 =0;          // Vref+ = AVdd
    ADCON0bits.VCFG = 0;          // Vref- = AVss
    ADCON0bits.CHS=2;             // Select ADC channel

    ADCON0bits.ADON=1;            // Turn on ADC

}
```

NOTES:

REGISTER 28-3: CONFIG2L: CONFIGURATION REGISTER 2 LOW (BYTE ADDRESS 300002h)

U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	BORPWR1(1)	BORPWR0(1)	BORV1(1)	BORV0(1)	BOREN1(2)	BOREN0(2)	$\overline{\text{PWRTEN}}$(2)
bit 7							bit 0

Legend:	P = Programmable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 7 **Unimplemented:** Read as ‘0’

bit 6-5 **BORPWR<1:0>:** BORMV Power-Level bits(1)
- 11 = ZPBORVMV instead of BORMV is selected
- 10 = BORMV is set to a high-power level
- 01 = BORMV is set to a medium power level
- 00 = BORMV is set to a low-power level

bit 4-3 **BORV<1:0>:** Brown-out Reset Voltage bits(1)
- 11 = V_{BORMV} is set to 1.8V
- 10 = V_{BORMV} is set to 2.0V
- 01 = V_{BORMV} is set to 2.7V
- 00 = V_{BORMV} is set to 3.0V

bit 2-1 **BOREN<1:0>:** Brown-out Reset Enable bits(2)
- 11 = Brown-out Reset is enabled in hardware only (SBOREN is disabled)
- 10 = Brown-out Reset is enabled in hardware only and disabled in Sleep mode (SBOREN is disabled)
- 01 = Brown-out Reset is enabled and controlled by software (SBOREN is enabled)
- 00 = Brown-out Reset is disabled in hardware and software

bit 0 **$\overline{\text{PWRTEN}}$:** Power-up Timer Enable bit(2)
- 1 = PWRT is disabled
- 0 = PWRT is enabled

Note 1: For the specifications, see **Section 31.1 “DC Characteristics: Supply Voltage PIC18F87K22 Family (Industrial/Extended)”**.

2: The Power-up Timer is decoupled from Brown-out Reset, allowing these features to be independently controlled.

REGISTER 28-5: CONFIG3L: CONFIGURATION REGISTER 3 LOW (BYTE ADDRESS 300004h)

U-1	U-1	U-1	U-1	U-1	U-0	U-0	R/P-1
WAIT(1)	BW(1)	ABW1(1)	ABW0(1)	EASHFT(1)	—	—	RTCOSC
bit 7							bit 0

Legend:	P = Programmable bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **WAIT:** External Bus Wait Enable bit(1)
1 = Wait states on the external bus are disabled
0 = Wait states on the external bus are enabled and selected by MEMCON <5:4>

bit 6 **BW:** Data Bus Width Select bit(1)
1 = 16-Bit Data Width modes
0 = 8-Bit Data Width modes

bit 5-4 **ABW<1:0>:** External Memory Bus Configuration bits(1)
11 = 8-Bit Address mode (Microcontroller mode)
10 = 12-Bit Address mode
01 = 16-Bit Address mode
00 = 20-Bit Address mode

bit 3 **EASHFT:** External Address Bus Shift Enable bit(1)
1 = Address shifting is enabled; external address is shifted to start at 000000h
0 = Address shifting is disabled; external address bus reflects the PC value

bit 2-1 **Unimplemented:** Read as '0'

bit 0 **RTCOSC:** RTCC Reference Clock Select bit
1 = RTCC uses SOSC as the reference clock
0 = RTCC uses LF-INTOSC as the reference clock

Note 1: Unimplemented on 64-pin devices (PIC18F6XK22), read as '0'.

28.2.1 CONTROL REGISTER

Register 28-16 shows the WDTCON register. This is a readable and writable register which contains a control bit that allows software to override the WDT Enable Configuration bit, but only if the Configuration bit has disabled the WDT.

REGISTER 28-16: WDTCON: WATCHDOG TIMER CONTROL REGISTER

R/W-0	U-0	R-x	R/W-0	U-0	R/W-0	R/W-0	R/W-0
REGSLP	—	ULPLVL	SRETEN(2)	—	ULPEN	ULPSINK	SWDTEN(1)
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **REGSLP:** Regulator Voltage Sleep Enable bit
1 = Regulator goes into Low-Power mode when device's Sleep mode is enabled
0 = Regulator stays in normal Operation mode when device's Sleep mode is activated

bit 6 **Unimplemented**: Read as '0'

bit 5 **ULPLVL:** Ultra Low-Power Wake-up Output bit
Not valid unless ULPEN = 1.
1 = Voltage on RA0 pin > ~ 0.5V
0 = Voltage on RA0 pin < ~ 0.5V.

bit 4 **SRETEN:** Regulator Voltage Sleep Disable bit(2)
1 = If $\overline{\text{RETEN}}$ (CONFIG1L<0>) = 0 and the regulator is enabled, the device goes into Ultra Low-Power mode in Sleep
0 = The regulator is on when the device's Sleep mode is enabled and the Low-Power mode is controlled by REGSLP

bit 3 **Unimplemented**: Read as '0'

bit 2 **ULPEN:** Ultra Low-Power Wake-up Module Enable bit
1 = Ultra Low-Power Wake-up module is enabled; ULPLVL bit indicates the comparator output
0 = Ultra Low-Power Wake-up module is disabled

bit 1 **ULPSINK:** Ultra Low-Power Wake-up Current Sink Enable bit
Not valid unless ULPEN = 1.
1 = Ultra Low-Power Wake-up current sink is enabled
0 = Ultra Low-Power Wake-up current sink is disabled

bit 0 **SWDTEN:** Software Controlled Watchdog Timer Enable bit(1)
1 = Watchdog Timer is on
0 = Watchdog Timer is off

Note 1: This bit has no effect if the Configuration bits, WDTEN<1:0>, are enabled.
2: This bit is available only when ENVREG = 1 and $\overline{\text{RETEN}}$ = 0.

TABLE 28-2: SUMMARY OF WATCHDOG TIMER REGISTERS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RCON	IPEN	SBOREN	$\overline{\text{CM}}$	$\overline{\text{RI}}$	$\overline{\text{TO}}$	$\overline{\text{PD}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$
WDTCON	REGSLP	—	ULPLVL	SRETEN	—	ULPEN	ULPSINK	SWDTEN

Legend: — = unimplemented, read as '0'. Shaded cells are not used by the Watchdog Timer.

NOTES:

CLRF **Clear f**

Syntax: CLRF f {,a}

Operands: $0 \le f \le 255$
$a \in [0,1]$

Operation: 000h → f,
1 → Z

Status Affected: Z

Encoding:

0110	101a	ffff	ffff

Description: Clears the contents of the specified register.

If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank.

If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever $f \le 95$ (5Fh). See **Section 29.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write register 'f'

Example: CLRF FLAG_REG,1

Before Instruction
FLAG_REG = 5Ah
After Instruction
FLAG_REG = 00h

CLRWDT **Clear Watchdog Timer**

Syntax: CLRWDT

Operands: None

Operation: 000h → WDT,
000h → WDT postscaler,
1 → $\overline{TO}$,
1 → $\overline{PD}$

Status Affected: $\overline{TO}$, $\overline{PD}$

Encoding:

0000	0000	0000	0100

Description: CLRWDT instruction resets the Watchdog Timer. It also resets the post-scaler of the WDT. Status bits, $\overline{TO}$ and $\overline{PD}$, are set.

Words: 1

Cycles: 1

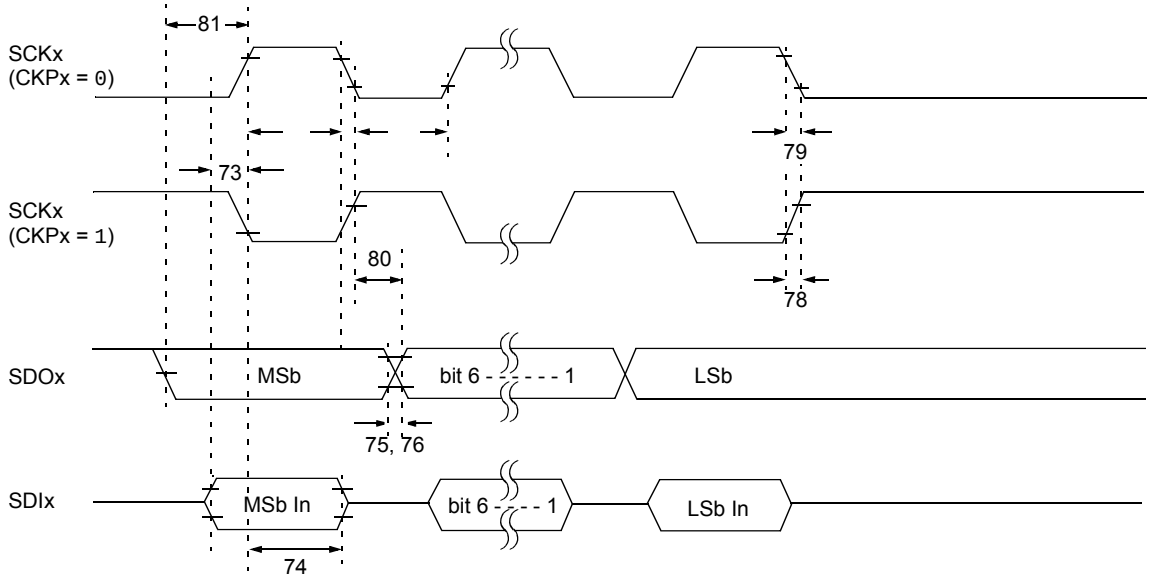
Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	Process Data	No operation

Example: CLRWDT

Before Instruction
WDT Counter = ?
After Instruction
WDT Counter = 00h
WDT Postscaler = 0
$\overline{TO}$ = 1
$\overline{PD}$ = 1

FIGURE 31-15: EXAMPLE SPI MASTER MODE TIMING (CKE = 1)



Note: Refer to Figure 31-3 for load conditions.

TABLE 31-18: EXAMPLE SPI MODE REQUIREMENTS (MASTER MODE, CKE = 1)

Param. No.	Symbol	Characteristic	Min	Max	Units	Conditions
73	TDIV2SCH, TDIV2SCL	Setup Time of SDIx Data Input to SCKx Edge	20	—	ns	
73A	TB2B	Last Clock Edge of Byte 1 to the 1st Clock Edge of Byte 2	1.5 TCY + 40	—	ns	
74	TSCH2DIL, TSCL2DIL	Hold Time of SDIx Data Input to SCKx Edge	40	—	ns	
75	TDOR	SDOx Data Output Rise Time	—	25	ns	
76	TDOF	SDOx Data Output Fall Time	—	25	ns	
78	TSCR	SCKx Output Rise Time (Master mode)	—	25	ns	
79	TSCF	SCKx Output Fall Time (Master mode)	—	25	ns	
80	TSCH2DOV, TSCL2DOV	SDOx Data Output Valid after SCKx Edge	—	50	ns	
81	TDOV2SCH, TDOV2SCL	SDOx Data Output Setup to SCKx Edge	TCY	—	ns	